Κατανομή Πόρων και Εκτίμηση της Οπτικής Ποιότητας σε Ασύρματη Μετάδοση Βίντεο

Η ΔΙΔΑΚΤΟΡΙΚΗ ΔΙΑΤΡΙΒΗ

υποβάλλεται στην

ορισθείσα από τη Γενική Συνέλευση Ειδικής Σύνθεσης

του Τμήματος Μηχανικών Η/Υ & Πληροφορικής
Εξεταστική Επιτροπή

από την

Αικατερίνη Πανδρεμμένου

ως μέρος των Υποχρεώσεων για τη λήψη του

ΔΙΔΑΚΤΟΡΙΚΟΥ ΔΙΠΛΩΜΑΤΟΣ ΣΤΗΝ ΠΛΗΡΟΦΟΡΙΚΗ

Ιούνιος 2015

COMMITTEES

Advisory Committee

- Lisimachos-Paul Kondi, Associate Professor, Department of Computer Science and Engineering, University of Ioannina, Greece (Supervisor)

- Christophoros Nikou, Associate Professor, Department of Computer Science and Engineering, University of Ioannina, Greece

- Konstantinos E. Parsopoulos, Assistant Professor, Department of Computer Science and Engineering, University of Ioannina, Greece

Examination Committee

- Lisimachos-Paul Kondi, Associate Professor, Department of Computer Science and Engineering, University of Ioannina, Greece (Supervisor)

- Christophoros Nikou, Associate Professor, Department of Computer Science and Engineering, University of Ioannina, Greece

- Konstantinos E. Parsopoulos, Assistant Professor, Department of Computer Science and Engineering, University of Ioannina, Greece

- Konstantinos Blekas, Assistant Professor, Department of Computer Science and Engineering, University of Ioannina, Greece

- Aggelos Katsaggelos, Professor, Department of Electrical Engineering and Computer Science, Northwestern University, USA

- Aristidis Likas, Professor, Department of Computer Science and Engineering, University of Ioannina, Greece

- Athanassios Skodras, Professor, Department of Electrical and Computer Engineering, University of Patras, Greece

ACKNOWLEDGEMENTS

CONTENTS

LIST OF FIGURES

LIST OF TABLES

LIST OF ALGORITHMS

ABSTRACT

Katerina Ch. Pandremmenou

PhD, Department of Computer Science and Engineering, University of Ioannina, Greece.

June 2015.

Thesis Title: *Resource Allocation and Visual Quality Estimation for Wireless Video Transmission.*

Thesis Supervisor: Lisimachos P. Kondi.

Providing the desired Quality of Service (QoS) as well as the maximum Quality of Experience (QoE) or improving the efficiency of H.264/AVC video transmissions over wireless networks presents several challenges due to the characteristics of wireless networks, such as limited bandwidth, time-varying channel conditions, heterogeneous users, etc. In such networks, data are transmitted via the wireless radio medium, which is a shared medium over which many users compete for resources. Due to the existence of many users, it is important to allocate resources in a fair manner among them. Resource allocation is applied under various network infrastructures such as cellular networks, relay channels, ultrawideband networks etc.

In the present thesis, we restrict our attention to the problem of resource allocation over wireless Visual Sensor Networks (VSNs), which consist of spatially distributed video cameras that are capable of compressing and transmitting the video sequences they acquire. Our goal is to ameliorate the video quality that reaches the end-user through efficient resource management. Specifically, we consider a Direct-Sequence Code Division Multiple Access (DS-CDMA) VSN, which employs a cross-layer design, where each node has its individual requirements in compression bitrate and energy consumption, depending on the characteristics of the monitored scenes.

The constraint that holds for each node of the network is that it has an available bitrate that can be shared between source and channel coding and an available power that can be used for video sensing, processing, and transmission. Hence, the source coding rates, channel coding rates and power levels are the parameters that should be optimally determined for each node, in an effort to tradeoff the video quality of the received videos and system's efficacy. The source and channel coding rates can take discrete values, while for the power levels we assume both the cases of taking continuous and discrete values.

In order to optimally and jointly allocate system resources to all nodes, we consider

four optimization criteria. Two of them aim at video distortion minimization, while the rest seek for a distribution rule that offers fair utility allocations. The first one, called the Minimized Average Distortion (MAD), minimizes the overall average video distortion of the network, neglecting fairness among the nodes. The second criterion, called the Minimized Maximum Distortion (MMD), minimizes the maximum distortion among all nodes of the network, promoting a rather unbiased treatment of the nodes. Since the simultaneous maximization of the video qualities of all nodes is not possible, we also apply cooperative game theory. Specifically, we use the Nash Bargaining Solution (NBS) in order to pinpoint one of the infinite Pareto-optimal solutions, based on the stipulation that the solution should satisfy four fairness axioms. An additional solution extracted from the area of game theory that we utilize to the same problem of resource allocation is the Kalai-Smorodinsky Bargaining Solution (KSBS), which has also to comply with four fairness axioms, and is applied to non-convex utility spaces. For comparison purposes, we also employ a criterion that Maximizes the Total system Utility (MTU) achieved by all nodes of the network.

Special attention is also given to the solution methodology followed by all explored optimization criteria. For the case where the power levels assume continuous values, we propose the use of the Particle Swarm Optimization (PSO) algorithm, which is a computational intelligence algorithm that draws inspiration from social dynamics. Also, we introduce a hybrid algorithm, denoted as Hybrid Particle Swarm Optimization Active Set (HPSOAS), which combines PSO with Active Set (AS) and aims at exploiting the benefits of the two aforementioned methods, thereby increasing efficiency. For comparison reasons, the performance of the Interior Point (IP) and Trust Region Reflective (TRR) methods is also evaluated to the same optimization problem, when all of the optimization criteria are used, except for the KSBS. The KSBS is found directly from the graphical representations of the utility sets, by following a geometric approach. In an effort to evaluate the results offered by each optimization criterion, we invoke four different fairness metrics: the first one considers both fairness and performance issues, and the second one measures the "equality" of a resource allocation (equal utilities for the nodes). The third metric computes the total system utility, while the last one computes the total power consumption of the nodes.

Another piece of the current study is focused on the joint problem of Group Of Pictures (GOP) length determination during the encoding process along with the allocation of the nodes' transmission parameters, where the objective function is indicated by the MAD, NBS and MTU optimization criteria. In this case, we have to tackle a purely discrete optimization problem as it results from the discrete source and channel coding rates and the discrete power levels for each node of the network. In this vein, we use the SARSA algorithm from the area of reinforcement learning.

Since in video transmissions over lossy networks quality degradation is inevitable, a common practice is to ensure higher reliability of the crucial pieces of information through the application of Unequal Error Protection (UEP). Such a scenario is also con-

sidered in our research, where based on the Cumulative Mean Squared Error (CMSE) we are able to prioritize the slices of the video sequences. In order to estimate the CMSE, we propose the use of the Least Absolute Shrinkage and Selection Operator (LASSO) regression method. A number of quality-relevant features are extracted from the H.264/AVC video sequences and are given as input to LASSO. Based on the estimated CMSE values, we group the video slices into four priority classes, we assign a different channel coding rate to each of them, and simulate a video transmission scenario over a noisy environment so as to investigate the performance of our proposed approach.

In the last part of this thesis, we deal with the problem of perceptual video quality assessment. Particularly, we propose Reduced-Reference (RR) and No-Reference (NR) models so as estimate the quality of H.264/AVC video sequences, in terms of the Mean Opinion Score (MOS). A variety of perceptually-motivated features are examined to account for the impact of coding artifacts, packet losses, and video content characteristics. These features are employed for estimating video quality using the LASSO regression technique, which utilizes a subset of the input features, by selecting only those that have relatively higher impact on the process of video quality estimation. For comparison purposes, the Ordinary Least Squares (OLS) and the Ridge regression method combined with sequential Forward Feature Selection (FFS) are also applied. In addition, performance measures as recommended by Video Quality Experts Group (VQEG) are used in order to gauge the effectiveness of our proposed models.

In order to estimate subjective video quality, we usually deal with a large number of features and a small sample set. Applying regression on complex datasets may lead to imprecise solutions due to possibly irrelevant or noisy features as well as the effect of overfitting. For this purpose, our research is extended to include a robust NR model that has a good generalization capability to unseen data, for videos that are impaired by both compression artifacts and packet losses. This model is able to improve the per-sequence MOS estimation accuracy, by following a frame-level MOS estimation approach, where the MOS estimate of a sequence is obtained by averaging the per-frame MOS estimates, instead of performing regression directly at the sequence level. Since it is impractical to obtain the per-frame MOS ground truth through subjective experiments, we propose an objective metric able to do this task, which provides a reliable indicator for the quality of each frame of a video, offering an intuition about its individual contribution to the overall video quality score.

Εκτεταμενη Περιληψη στα Ελληνικα

Αικατερίνη Πανδρεμμένου του Χρήστου και της Ευδοξίας.
PhD, Τμήμα Μηχανικών Η/Υ και Πληροφορικής, Πανεπιστήμιο Ιωαννίνων, Ιούνιος 2015.
Τίτλος Διατριβής: *Κατανομή Πόρων και Εκτίμηση της Οπτικής Ποιότητας για Ασύρματη Μετάδοση Βίντεο.*
Επιβλέποντας: Λυσίμαχος-Παύλος Κόντης.

Η παροχή της επιθυμητής ποιότητας των υπηρεσιών και της καλύτερης δυνατής ποιότητας της εμπειρίας του χρήστη ή ακόμη η βελτίωση της αποδοτικότητας των μεταδόσεων σε ασύρματα δίκτυα, βιντεοακολουθιών που έχουν κωδικοποιηθεί με το πρότυπο H.264/AVC, παρουσιάζουν αρκετές προκλήσεις εξαιτίας των χαρακτηριστικών αυτών των δικτύων, όπως για παράδειγμα το περιορισμένο εύρος ζώνης, οι χρονικά μεταβαλλόμενες συνθήκες του καναλιού, οι ετερογενείς χρήστες κτλ. Σε τέτοια δίκτυα, τα δεδομένα μεταφέρονται μέσω ενός ασύρματου, κοινόχρηστου μέσου, στο οποίο έχουν πρόσβαση πολλοί χρήστες. Οι χρήστες αυτοί προσπαθούν να επωφεληθούν όσο το δυνατόν περισσότερο από τους διαθέσιμους πόρους του δικτύου και επομένως είναι απαραίτητο να κατανείμουμε δίκαια τους πόρους ανάμεσά τους. Η κατανομή πόρων είναι ένα πρόβλημα που συναντάται σε διάφορες υποδομές δικτύων όπως κυψελωτά δίκτυα, κανάλια με ενδιάμεσους, βοηθητικούς κόμβους, δίκτυα υπερευρείας ζώνης κ.α.

Στην παρούσα διατριβή, εστιάζουμε στο πρόβλημα της κατανομής πόρων σε ασύρματα δίκτυα οπτικών αισθητήρων, τα οποία απαρτίζονται από κάμερες που βρίσκονται τοποθετημένες σε διάφορα σημεία και οι οποίες συμπιέζουν και μεταδίδουν τα βίντεο που καταγράφουν. Ο στόχος μας είναι να βελτιώσουμε την ποιότητα του βίντεο που βλέπει ο χρήστης, μέσω αποδοτικής διαχείρισης των πόρων του συστήματος. Συγκεκριμένα, θεωρούμε ένα δίκτυο οπτικών αισθητήρων πολλαπλής πρόσβασης με διαίρεση κωδίκων και χρήση άμεσης ακολουθίας, το οποίο χρησιμοποιεί ένα διαστρωματικό σχεδιασμό, όπου ο κάθε κόμβος του δικτύου έχει τις δικές του απαιτήσεις σε ρυθμό bit για τη συμπίεση και σε ενέργεια (ισχύ), ανάλογα με τα χαρακτηριστικά των σκηνών που καταγράφει.

Ο ρυθμός bit σε κάθε κόμβο του δικτύου θα πρέπει να μοιραστεί για την κωδικοποίηση της πηγής και του καναλιού, ενώ η διαθέσιμη ισχύς θα χρησιμοποιηθεί για την καταγραφή, επεξεργασία και μετάδοση των βιντεοακολουθιών. Επομένως, ο ρυθμός κωδικοποίησης της πηγής, ο ρυθμός κωδικοποίησης του καναλιού και η ισχύς αποτελούν τις παραμέτρους οι οποίες θα πρέπει να εκτιμηθούν βέλτιστα για κάθε κόμβο του δικτύου, έτσι ώστε να πετύχουμε το καλύτερο δυνατό ισοζύγιο μεταξύ της ποιότητας των λαμβανόμενων

βιντεοακολουθιών και της απόδοσης του συστήματος. Οι ρυθμοί κωδικοποίησης πηγής και καναλιού παίρνουν διακριτές τιμές, ενώ για τις ισχύς θεωρούμε δύο περιπτώσεις, ότι δηλαδή μπορούνε να πάρουνε και συνεχείς και διακριτές τιμές.

Προκειμένου να κατανείμουμε βέλτιστα και από κοινού τους πόρους του συστήματος σε όλους τους κόμβους, θεωρούμε τέσσερα κριτήρια βελτιστοποίησης. Τα δύο από αυτά αποσκοπούν στην ελαχιστοποίηση της παραμόρφωσης του βίντεο, ενώ τα υπόλοιπα δύο αναζητούν έναν κανόνα που προσφέρει δίκαιες κατανομές πόρων για κάθε κόμβο. Πιο συγκεκριμένα, το πρώτο κριτήριο ελαχιστοποιεί τη μέση συνολική παραμόρφωση του δικτύου (MAD), αγνοώντας θέματα δικαιοσύνης ανάμεσα στους κόμβους. Το δεύτερο κριτήριο ελαχιστοποιεί τη μέγιστη παραμόρφωση ανάμεσα σε όλους τους κόμβους του δικτύου (MMD) και μεταχειρίζεται αμερόληπτα τους κόμβους. Επιπλέον, καθώς η ταυτόχρονη μεγιστοποίηση της ποιότητας των βίντεο όλων των κόμβων δεν είναι δυνατή, εφαρμόζουμε συνεργατική θεωρία διαπραγμάτευσης, χρησιμοποιώντας τη λύση διαπραγμάτευσης του Nash (NBS) προκειμένου να επιλέξουμε μία από τις άπειρες βέλτιστες κατά Pareto λύσεις, με την προϋπόθεση ότι θα πρέπει να ικανοποιούνται τέσσερα αξιώματα δικαιοσύνης. Ακόμη, στο ίδιο πρόβλημα χρησιμοποιούμε άλλη μία λύση από τη θεωρία παιγνίων, τη λύση διαπραγμάτευσης των Kalai-Smorodinsky (KSBS), η οποία θα πρέπει επίσης να ικανοποιεί τέσσερα αξιώματα δικαιοσύνης, και η οποία στην περίπτωσή μας εφαρμόζεται σε μη-κυρτούς χώρους ωφέλειας. Για λόγους σύγκρισης, χρησιμοποιούμε ακόμη ένα κριτήριο που μεγιστοποιεί τη συνολική ωφέλεια του συστήματος (MTU) που επιτυγχάνεται από όλους τους κόμβους του δικτύου.

Ιδιαίτερη προσοχή δίνουμε επίσης στη μεθοδολογία που ακολουθείται για την εξεύρεση λύσης, χρησιμοποιώντας κάθε ένα από τα προαναφερθέντα κριτήρια. Στην περίπτωση που οι ισχύς παίρνουν συνεχείς τιμές, προτείνουμε τη χρήση του αλγορίθμου βελτιστοποίησης σμήνους σωματιδίων (PSO), ο οποίος είναι ένας αλγόριθμος υπολογιστικής νοημοσύνης εμπνευσμένος από τη φύση. Επίσης, εισάγουμε έναν υβριδικό αλγόριθμο, τον υβριδικό αλγόριθμο βελτιστοποίησης σμήνους σωματιδίων ενεργού συνόλου (HPSOAS), ο οποίος συνδυάζει τον αλγόριθμο PSO με τον αλγόριθμο ενεργού συνόλου (AS). Ο αλγόριθμος HPSOAS εκμεταλλεύεται τα πλεονεκτήματα και των δύο προαναφερθέντων αλγορίθμων κι επομένως πετυχαίνει καλύτερη απόδοση. Για λόγους σύγκρισης, αξιολογούμε την απόδοση των μεθόδων interior point (IP) και trust region reflective (TRR) στο ίδιο πρόβλημα βελτιστοποίησης, χρησιμοποιώντας καθένα από τα κριτήρια βελτιστοποίησης, εκτός από το KSBS, το οποίο υπολογίζεται απευθείας από τις γραφικές αναπαραστάσεις των συνόλων ωφέλειας, ακολουθώντας μία γεωμετρική προσέγγιση. Σε μια προσπάθεια να αξιολογήσουμε τα αποτελέσματα που προκύπτουν χρησιμοποιώντας κάθε κριτήριο βελτιστοποίησης, χρησιμοποιούμε τέσσερις διαφορετικές μετρικές δικαιοσύνης. Η πρώτη από αυτές συνυπολογίζει θέματα δικαιοσύνης και απόδοσης, ενώ η δεύτερη μετράει την 'ισότητα' της κατανομής των πόρων (ίσες ωφέλειες για τους κόμβους). Η τρίτη μετρική υπολογίζει τη συνολική ωφέλεια του συστήματος και η τέταρτη τη συνολική κατανάλωση ενέργειας από τους κόμβους.

Ένα άλλο κομμάτι της τρέχουσας μελέτης εστιάζει στο πρόβλημα του καθορισμού του βέλτιστου μήκους της ομάδας των εικόνων (GOP) σε συνδυασμό με το πρόβλημα της

κατανομής των πόρων στους κόμβους του δικτύου. Στην περίπτωση αυτή, η συνάρτηση βελτιστοποίησης υποδεικνύεται από τα κριτήρια MAD, NBS και MTU, όπου έχουμε να επιλύσουμε ένα διακριτό πρόβλημα που προκύπτει από τις διακριτές τιμές των ρυθμών κωδικοποίησης πηγής και καναλιού καθώς επίσης και τις διακριτές τιμές των ισχύων. Για την επίλυση αυτού του προβλήματος χρησιμοποιούμε τον αλγόριθμο SARSA από την περιοχή της ενισχυτικής μάθησης.

Καθώς κατά τις μεταδόσεις βίντεο σε απωλεστικά δίκτυα η υποβάθμιση της ποιότητας είναι αναπόφευκτη, μία κοινή πρακτική είναι να εξασφαλίζουμε υψηλότερη αξιοπιστία στα σημαντικά κομμάτια πληροφορίας εφαρμόζοντας άνιση προστασία από λάθη (UEP). Στην παρούσα διατριβή θεωρούμε ένα τέτοιο σενάριο, όπου με βάση το αθροιστικό μέσο τετραγωνικό σφάλμα (CMSE) δίνουμε προτεραιότητα στα επιμέρους κομμάτια των βιντεοακολουθιών. Προκειμένου να εκτιμήσουμε το CMSE, προτείνουμε τη χρήση της least absolute shrinkage and selection operator (LASSO) μεθόδου παλινδρόμησης. Πιο συγκεκριμένα, εξάγουμε ορισμένα χαρακτηριστικά από τις βιντεοακολουθίες που έχουν κωδικοποιηθεί με το πρότυπο H.264/AVC και τα οποία σχετίζονται με την ποιότητα του βίντεο. Στη συνέχεια, τα χαρακτηριστικά αυτά τροφοδοτούν το LASSO προκειμένου να εκτιμήσουμε το CMSE, όπως αυτό προκύπτει από κάθε πιθανή απώλεια ενός τμήματος της βιντεοακολουθίας. Με βάση τις εκτιμώμενες τιμές για το CMSE, ομαδοποιούμε τα επιμέρους κομμάτια του βίντεο σε τέσσερις ομάδες προτεραιότητας και αναλόγως, αναθέτουμε διαφορετικούς ρυθμούς κωδικοποίησης του καναλιού σε κάθε μία από αυτές. Τέλος, προσομοιώνουμε ένα σενάριο μετάδοσης βιντεοακολουθιών σε ένα ενθόρυβο περιβάλλον προκειμένου να μελετήσουμε την αποτελεσματικότητα της προτεινόμενης προσέγγισης.

Στο τελευταίο τμήμα της διατριβής ασχολούμαστε με το πρόβλημα της εκτίμησης της ποιότητας του βίντεο όπως την αντιλαμβάνεται ο χρήστης. Συγκεκριμένα, προτείνουμε μοντέλα που έχουν μειωμένη ή και καθόλου πρόσβαση στην αρχική βιντεοακολουθία προκειμένου να εκτιμήσουμε την ποιότητα των βιντεοακολουθιών που φτάνουν στο χρήστη, μέσω του mean opinion score (MOS). Μελετάμε μία μεγάλη ποικιλία χαρακτηριστικών που σχετίζονται με την ποιότητα του βίντεο όπως την αντιλαμβάνεται ο χρήστης και τα οποία αντικατοπτρίζουν τις παραμορφώσεις λόγω της συμπίεσης και των απωλειών πακέτων καθώς επίσης και τις ιδιαιτερότητες του βίντεο. Κατόπιν, τα χαρακτηριστικά αυτά χρησιμοποιούνται από το LASSO για να εκτιμήσουμε την ποιότητα του βίντεο. Το LASSO έχει την ικανότητα να χρησιμοποιεί ένα υποσύνολο από τα αρχικά χαρακτηριστικά, και συγκεκριμένα εκείνα τα οποία έχουν συγκριτικά μεγαλύτερο αντίκτυπο στη διαδικασία της εκτίμησης της ποιότητας των βιντεοακολουθιών. Για λόγους σύγκρισης, εφαρμόζουμε τις μεθόδους παλινδρόμησης των ελαχίστων τετραγώνων (OLS) και τη Ridge σε συνδυασμό με μία τεχνική ακολουθιακής εμπρόσθιας επιλογής χαρακτηριστικών (FFS). Ακόμη, για να μετρήσουμε την αποδοτικότητα των μοντέλων που προτείνουμε χρησιμοποιούμε κάποιες μετρικές απόδοσης, όπως προτάθηκαν από το video quality experts group (VQEG).

Συχνά, σε προβλήματα εκτίμησης της υποκειμενικής ποιότητας του βίντεο ερχόμαστε αντιμέτωποι με ένα μεγάλο αριθμό χαρακτηριστικών και με ένα μικρό μέγεθος παρατηρήσεων. Εάν εφαρμόσουμε παλινδρόμηση σε σύνθετα σύνολα δεδομένων ενδέχεται να οδηγηθού-

με σε μη ακριβείς λύσεις εξαιτίας της ύπαρξης κάποιων άσχετων χαρακτηριστικών ή χαρακτηριστικών που αποτελούν ουσιαστικά θόρυβο για το πρόβλημά μας ή ακόμη εξαιτίας του προβλήματος της υπερπροσαρμογής. Γι' αυτό το λόγο επεκτείνουμε την έρευνα μας και εισάγουμε ένα εύρωστο μοντέλο χωρίς αναφορά, το οποίο παρουσιάζει καλή γενικευτική ικανότητα σε άγνωστα δεδομένα, για βιντεοακολουθίες που πλήττονται από παραμορφώσεις εξαιτίας της συμπίεσης και των απωλειών πακέτων. Το μοντέλο που προτείνουμε βελτιώνει την ακρίβεια εκτίμησης του MOS ολόκληρης της ακολουθίας, κάνοντας εκτιμήσεις για το MOS κάθε καρέ. Αξίζει να σημειωθεί ότι το MOS ολόκληρης της ακολουθίας προκύπτει ως ο μέσος όρος των εκτιμήσεων των MOS όλων των καρέ, κι όχι εφαρμόζοντας παλινδρόμηση απευθείας σε επίπεδο ακολουθίας. Καθώς δεν είναι πρακτικό να λάβουμε το πραγματικό MOS όλων των καρέ κάνοντας πειράματα με χρήστες, αντ' αυτού προτείνουμε μία αντικειμενική μετρική η οποία παρέχει μια αξιόπιστη ένδειξη για την ποιότητα κάθε καρέ του βίντεο δίνοντας συγχρόνως μια εικόνα για τη συνεισφορά του στο συνολικό σκορ της ποιότητας του βίντεο.

CHAPTER 1

INTRODUCTION

1.1 Overview

Nowadays, multimedia video streaming has become increasingly popular and more and more users join the end-system over a wireless environment. However, due to the characteristics of wireless networks such as heterogeneous wireless users, high error rate, limited bandwidth, multiple transmission rates, time-varying channel conditions and dynamic network users, providing the desired Quality of Service (QoS) as well as the maximum Quality of Experience (QoE) or even improving the efficiency for H.264/Advanced Video Coding (AVC) video transmissions over wireless networks presents several new challenges.

In a wireless network, data are transmitted via the wireless radio medium. Node mobility causes bandwidth fluctuations due to differences in channel quality. Even if the wireless station is stationary, its wireless bandwidth may fluctuate due to multi-path fading, co-channel interference, and noise disturbances. Bandwidth fluctuations represent the main challenges of real-time video streaming over wireless networks. For this reason, a cross-layer design is needed for video streaming in wireless networks [69].

Most wireless networking solutions are designed around the layered protocol architecture that forms the foundation of networking design, and thus, they are not always able to provide adequate support for multimedia applications. In a layered architecture, like the seven-layer Open Systems Interconnection (OSI) model, each protocol layer completes a specific task with only the information from the layer below it and

then provides the processed information only to the next layer above. This sort of layered architecture is beneficial in that it grants flexibility to the designers to modify one layer without disrupting the rest of the overall system, and it has been proved to be successful with wired networks. However, with the rapid growth of wireless networks, the question arises as to whether this architecture is still optimal for this kind of networks as well [159], due to their time-varying nature that leads to errors due to multipath fading and co-channel interference.

A wireless sensor network is a wireless network consisting of spatially distributed autonomous devices using sensors to monitor physical or environmental conditions. Initially, such networks were mainly concerned with the transmission of unidimensional signals (e.g., temperature, sound, etc). Nowadays, their applications have been expanded to the transmission of visual data, such as images or videos. This type of wireless sensor networks that conveys visual data is the well-known Visual Sensor Network (VSN), on which we spend a large portion of our research.

In VSNs, each node is equipped with a camera for imaging different fields of view and detecting events of interest. VSNs support a plethora of applications, ranging from security and teleconference systems to environmental monitoring [181]. Security monitoring includes the surveillance of large areas such as motorways, airports, banks, and other public or private places, mainly for the prevention of unpleasant events or illegal activities. Environmental monitoring is concerned with the monitoring of natural environment in remote locations where human presence is often impossible. These monitoring systems deter illegal construction, logging, hunting, etc. Concerning teleconference systems, they enable users to participate remotely in events of interest, such as attending a meeting or a lecture that is taking place in a different location, or even virtually visit an exhibition or a museum.

Therefore, it is evident that the nodes of the VSN monitor scenes with different motion activity. For example, some nodes may be imaging scenes with high levels of motion, while some other nodes may record low-motion scenes. Since VSN applications are related with video delivery, the demands for resources are particularly increased. On the contrary, wireless video communications suffer from a number of network resource constraints, including bandwidth, energy and computational complexity limitations. Data imaging, processing and transmission are recognized as power-consuming operations that can affect the performance of VSNs. Also, the available bitrate for video transmission can be limited in a wireless channel due to limited bandwidth and adverse channel conditions. Therefore, the main challenge in wireless VSNs is the coordinated behavior of each node constituting the network, such that it maximizes the overall system performance within the various resource constraints.

In the context of our research, we consider a Direct-Sequence Code Division Multiple Access (DS-CDMA) VSN, where the sensor nodes communicate directly with a central processing server, called the Centralized Control Unit (CCU), and they exchange information in order to achieve the ideal tradeoff between the transmitted video quality and
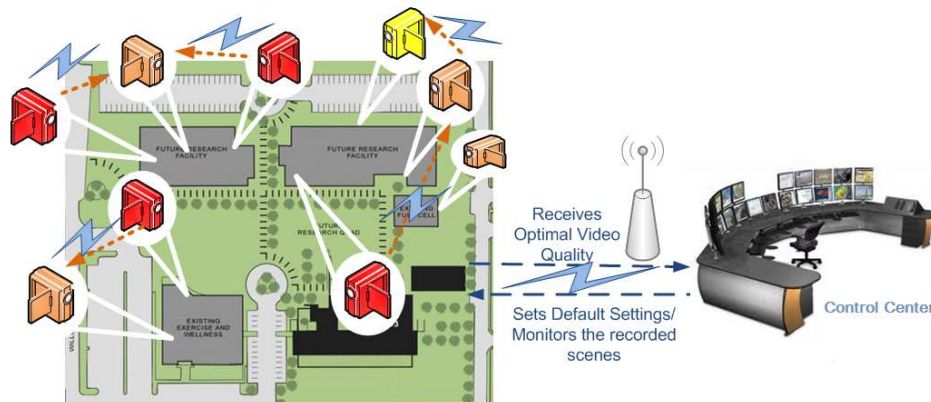
Figure 1.1: A wireless VSN environment.

energy consumption. Figure 1.1 depicts a wireless VSN environment.

Each node of the considered VSN has an available power that can be used for video sensing, processing, and transmission. Transmission powers have to be high enough to guarantee video reception with an acceptable visual quality. On the other hand, they have to be low enough to prevent increased interference and to prolong the battery lifetime of the battery-operated nodes, increasing the energy efficiency and maximizing the lifetime of the network. In light of this, an efficient allocation of the transmission parameters among all nodes of the VSN becomes crucial in order to achieve the dual goal of energy saving and visual quality assurance.

However, the question that arises is how video quality is defined for a specific application? A thorough understanding of visual quality as it is perceived by the human eye is necessary in order to meet the QoE requirements of many modern multimedia communication systems. In any processing or transmission of visual information, the ultimate judge is the human observer, and due to this, subjective assessment is considered the most reliable source of evaluation, providing the ground truth. However, this approach is not practical for applications that involve automatic control and adjustment of machine parameters as well as it entails a lot of extra limitations, resulting mainly in financial cost. Due to this, over the last decades a large effort is devoted to developing objective metrics able to evaluate video quality automatically, targeting at the reliable and accurate modeling of subjective Video Quality Assessment (VQA). Nevertheless, in spite of the great number of objective metrics developed for this purpose, the majority of them attempt to modify the Mean Squared Error (MSE) measure, which is a mathematical error measure that does not consider the Human Visual System (HVS), and is therefore not an accurate measure of perceptual quality. Consequently, it is highly desirable to derive efficient objective metrics that are consistent with subjective ratings.

A classification of the objective metrics can be made on the basis of the reference information they use for quality estimation. Some of them have access to both of the original and impaired versions of a video, some others have access to the impaired video sequence and key features of the original video, while there are also some metrics that are allowed to access only the impaired video sequence. Evidently, the more limited the available input information, the more challenging the video quality estimation can be [172].

Quality of visual media can get degraded during capture, storage, transmission, reproduction and display due to distortions that might occur at any of these stages. Nonetheless, most objective metrics model impairments related exclusively to video compression, while there are also some other metrics that are concerned with the visual effects of packet losses during wireless transmissions. Even fewer are those metrics that take into account both compression artifacts and transmission impairments, simulating a more realistic scenario of video transmission over wireless networks.

1.2 Thesis Contribution

In this thesis, we mainly elaborate on two different axes: i) resource allocation over wireless visual sensor networks and ii) objective metrics for video quality assessment. The specific contributions in each direction are described below.

First of all, we develop a flexible cross-layer design in order to overcome possible network latency problems and improve the real-time response of the considered DS-CDMA VSN. The proposed scheme operates over the whole range of layers, allowing them to exchange information, regardless of their position in the considered layered hierarchy. More specifically, the particular design operates across the physical, data link, network and application layers. The layer collaboration is controlled by the CCU, which communicates with all nodes, receiving their data and requesting changes in their transmission parameters according to their unique, content-aware needs for resources.

Each node of the network has a fixed bitrate that can be shared between source and channel coding. Source coding aims at video representation with the smallest number of bits, removing redundant information from the video bitstreams. On the contrary, channel coding attempts to increase the reliability of video transmissions, and for this purpose, redundant bits are added to the video bitstreams. Additionally, each node has also an available power in order to be used for video sensing, processing and transmission. Hence, the source coding rates, channel coding rates and power levels are the parameters that should be optimally determined, under certain modeling conditions, in an effort to tradeoff the video quality of the received videos and system efficacy.

In order to optimally and jointly allocate system resources to all nodes, we consider two quality-based optimization criteria aiming at video distortion minimization. The

first one, called the Minimized Average Distortion (MAD), minimizes the overall average video distortion of the network, neglecting fairness among the nodes. The second criterion, called the Minimized Maximum Distortion (MMD), minimizes the maximum distortion among all nodes of the network, promoting a rather unbiased treatment of the nodes. Moreover, we consider an alternative scheme that maximizes the total system utility achieved by all nodes of the network and is called the Maximized Total Utility (MTU). As it is declared by its name, the specific optimization criterion leads to the maximum sum of the wireless stations qualities, while the quality differences among the wireless stations are significant.

In addition, game theorists have proposed several bargaining solutions so as to resolve similar resource allocation problems. The Nash Bargaining Solution (NBS) was the first among many bargaining solutions. This solution, based on its set of fairness axioms, provides a fair and efficient way for distributing resources. In light of this, except for the above quality-based optimization criteria, we apply cooperative game theory to deal with the problem of resource allocation, by using the NBS. Since the simultaneous maximization of the video qualities of all nodes is not possible, we employ the NBS so as to pinpoint one of the infinite Pareto-optimal solutions (i.e., solutions that are jointly preferred by all nodes, meaning that it is impossible for a node to increase its utility, without another node to decrease its own), based on the stipulation that the solution should satisfy four fairness axioms. Specifically, this solution promises fairness for all nodes, taking into account the amounts of motion in the videos they capture. Besides NBS, we also employ another solution extracted from game theory. The Kalai-Smorodinsky Bargaining Solution (KSBS) is applied to non-convex utility spaces, in order to find a fair utility allocation resulting from the optimal determination of the nodes' transmission parameters, considering a number of assumptions and constraints.

All of the schemes we examine in this study optimize a function of the video qualities of the nodes, while the methodology applied to each of these schemes in order to provide an optimal (or near optimal) solution that complies with the considered constraints is a significant issue that has attracted our interest. In our considered problem formulation, we assume that for each node, the power level is continuous and bounded, while source coding rate retains discrete values, since channel coding rate can take values only within a finite discrete set [61]. Hence, for the purpose of solving the optimization problems, we suggest the use of the Particle Swarm Optimization (PSO) algorithm, which is a computational intelligence approach that draws inspiration from social dynamics. This methodology is applied to all of the aforementioned criteria, except for the KSBS, which is derived by following a novel geometric approach, directly from the graphical representations of the nodes' utility sets.

The performance of PSO is compared to the performance of three deterministic optimization methods, which are used as benchmarks, such as Active Set (AS), Interior Point (IP) and Trust Region Reflective (TRR). The conducted statistical tests show

that PSO greatly outperforms these classical deterministic algorithms, offering strong motivation for its use as the main optimizer in our research. In addition, motivated by the promising performance of hybrid algorithms that combine population-based approaches with deterministic schemes, we also consider a hybrid algorithm that combines PSO with AS. The new algorithm, denoted as Hybrid Particle Swarm Optimization Active Set (HPSOAS), is designed to exploit the benefits of both PSO and AS, thereby increasing efficiency.

However, there is no single scheme that maximizes the video quality of each node simultaneously. In fact, all presented schemes are able to provide Pareto-optimal solutions. Thus, it is not clear which scheme results in the best resource allocation for the whole network. Although the general feeling tends to associate fairness with equality, researchers disagree as to what should be equalized. Sometimes, it is desirable to achieve similar utilities for all nodes of the network, while some other times the goal is an equal utility penalty for all nodes relative to the maximum achievable utility. Additionally, there are cases where the challenge is a high total utility allocation, cumulatively for all nodes, considering the available network resources, channel conditions, participating nodes and video content characteristics.

To handle the resulting tradeoffs, in the current research we examine four metrics that investigate fairness and efficiency under different perspectives. Specifically, we apply a metric that considers both fairness and performance issues, and another metric that measures the "equality" of a resource allocation (equal utilities for the nodes). The third metric computes the total system utility, while the last metric computes the total power consumption of the nodes. Ideally, a desirable scheme would achieve high total utility while being equally fair to all nodes and requiring low amounts of power, at the same time.

Since the errors that occur during wireless multimedia transmissions in conjunction with the lossy source coding techniques deteriorate the quality of the video sequences at the decoder, careful treatment is required during video encoding in order to acquire high coding performance and robustness to transmission errors, so as to ensure high end-to-end video quality. In this context, another aspect of our research concerns the cross-layer resource allocation problem among the nodes of a wireless VSN, dealing with the optimal Instantaneous Decoding Refresh (IDR) frame placement during the encoding process, based on the motion level included in each video sequence, at the same time. In this case, we have to tackle a discrete optimization problem, since all nodes' transmission parameters, i.e., source coding rates, channel coding rates, power levels and Group Of Pictures (GOP) lengths can all take discrete values.

The increased problem's dimensionality that comes from some assumptions regarding the possible choices for the values of the power levels and GOP lengths as well as the various motion levels included in the scenes captured by the nodes, motivate us to adopt a Reinforcement Learning (RL) scheme, which allows the controller (CCU) to make optimal decisions in unknown environments with very large or continuous state

spaces. Specifically, we use the tabular SARSA algorithm, which constructs a map that allows us to explore the best parameters with the minimum effort, starting by any randomly selected parameters' combination. Therefore, once the optimal parameters are selected video quality enhancement is observed.

In the same context of assuring high end-to-end video quality in wireless video transmissions, we consider a scenario where H.264/AVC video sequences are transmitted over an Additive White Gaussian Noise (AWGN) channel. Evidently, slice losses occur and they negatively affect perceptual video quality. Since the contents of some parts of a video are considered to be more important than others, better protection should be provided to ensure higher reliability of the crucial pieces of information. For this purpose, Unequal Error Protection (UEP) is applied to accommodate for receivers with poor link quality and assure a more graceful video degradation. In our consideration, we apply a Quartile-Based Prioritization (QBP) scheme, classifying slices into four priorities, based on the Cumulative MSE (CMSE) index. This index is employed in order to account for the impact of individual slice losses on video quality, by accurately describing the error propagation within a GOP. It is computed by systematically discarding one video slice at each time and summing the MSE of the current and subsequent frames in the same GOP.

Having collected the actual CMSE values for each slice, we propose the use of the Least Absolute Shrinkage and Selection Operator (LASSO) regression method in order to estimate CMSE. A number of quality-relevant features extracted from the H.264/AVC video sequences are given as input to LASSO, which is able to keep a subset of the features that have the strongest effects towards video quality, producing accurate CMSE estimations, at the same. Therefore, based on the measured and estimated CMSE values, we classify the slices into four priorities, aiming at the assignment of a different channel coding rate to each differently prioritized class, so as to enhance in this way the video quality that reaches the end user.

It is commonly accepted that the automatic estimation and control of video quality is highly desirable so as to satisfy the needs for high end-to-end QoE. Thus, objective metrics have greater potential, especially for real-time quality monitoring of video communication systems. Since degradation of perceptual video quality occurs due to both lossy video encoding and transmission errors, there is the need for the development of a perceptual video quality model able to estimate possible video degradations due to both these sources of distortion.

In light of this, we propose Reduced-Reference (RR) and No-Reference (NR) models for estimating the quality of H.264/AVC video sequences, affected by both compression artifacts and transmission impairments. The proposed models are based on a feature extraction procedure, where a large number of perceptually-motivated features are examined to account for the impact of coding artifacts, packet losses, and video content characteristics. The feature observations are given as input to the LASSO regression method in order for the latter to indicate those features that have the strongest ef-

fects towards video quality, and using only them to produce video quality estimates. Additionally, for evaluating the efficiency of LASSO in terms of both model's sparsity and estimation accuracy, we also employ the Ordinary Least Squares (OLS) regression method as well as Ridge, where before applying this latter method a feature selection preprocessing takes place.

In supervised learning regression problems, we usually deal with a large number of features and a small sample set. Therefore, applying regression on complex datasets may lead to imprecise solutions due to possibly irrelevant or noisy features as well as the effect of overfitting. In this context, in an effort to establish a robust regression model that generalizes well to unknown data and to increase the estimation accuracy of visual quality, we propose a frame-level quality estimation approach, where the quality estimate of a sequence is obtained by averaging the per-frame quality estimates, instead of performing regression directly at the sequence-level. Moreover, as it is impractical to obtain the per-frame quality ground truth through subjective experiments, we propose an objective metric able to do this task.

1.3 Thesis Layout

The rest of this thesis is structured as follows: In Chapter 2 we overview the works that are related to the topics and particular problems discussed in the current study. In Chapter 3 we elaborate on the continuous resource allocation problem of a wireless VSN and how it is tackled through the use of the MAD, MMD, NBS, KSBS and MTU optimization criteria, along with their variants. Also the same chapter, discusses the optimization solvers used, including PSO, AS, IP, TRR and the proposed hybrid method HPSOAS. Furthermore, a part of this chapter is devoted to the fairness and efficiency investigation of the solutions provided by each of the optimization criteria, under different perspectives. In Chapter 4, we discuss our approach of considering the optimal GOP length during encoding in combination with a discrete resource allocation problem. The SARSA algorithm is analyzed as well, as it is used for tackling the specific problem. Chapter 5 presents the scenario of video transmission over an AWGN channel, where UEP is applied based on the estimated CMSE, which results through the use of LASSO. The proposed RR and NR metrics for estimating perceptual video quality are introduced in Chapter 6, where the LASSO regression framework is also presented. In the same chapter, OLS and Ridge are also discussed. Chapter 7 presents an approach for further improving perceptual video quality estimates as well as stepwise regression is meticulously described. Finally, Chapter 8 summarizes this thesis and overviews directions for future work.

CHAPTER 2

BACKGROUND AND RELATED WORK

R ecent advances in micro-electro-mechanical systems, wireless communications, and digital electronics have enabled the development of low-cost, low-power, multifunctional sensor nodes that are small in size and communicate untethered in short distances [13]. As the use of wireless local area networks spreads beyond simple data transfer to bandwidth intense, delay-sensitive, and loss-tolerant multimedia applications, addressing QoS issues become extremely important [198].

In the current research, we consider a Direct-Sequence Code Division Multiple Access (DS-CDMA) [49] wireless Visual Sensor Network (VSN), under a centralized topology, where the nodes are spatially distributed. They contain low-weight smart cameras capable of processing and fusing images of a scene from a variety of viewpoints into some form more useful than the individual images. A Centralized Control Unit (CCU) collects all the video data and network statistics and makes the resource allocation decision for each node. The nodes send only input and receive output, since all data processing is performed at the central computer.

The policy applied in our study is based on a cross-layer multi-node optimization design that accounts for the overall system performance through all network layers and differs from usual layered network architectures like the Open Systems Interconnection (OSI) [39], and the Transmission Control Program/Internet Protocol (TCP/IP) models [190]. Particularly, it is responsible for the optimal determination of the source coding rates, channel coding rates, and power levels at the application layer, data link layer, and physical layer, respectively, while the CCU lies at the network layer.

2.1   Network Layer Designs

Networks in general, and the Internet in particular, make use of what are sometimes called protocol stacks or layered protocols. The motivating factor for the use of a layered architecture is the incredible diversity of systems and physical devices. The basic idea is that each layer is responsible for a particular kind of functionality. Each layer depends on the layers below it for other functions, and provides services to the layers above it. One major goal of a layered architecture is to factor out various services so that a given service can be used by multiple versions of the layer above it and can make use of multiple versions of the layer below it. The OSI reference model [39] consists of seven layers that is, the physical, data link, network, transport, session, presentation and application layers, from lowest to highest. Each of these successive layers includes wrapping of the functionalities of the lower layers that are made available as information to the above layer.

The internet protocol suite is the set of communication protocols used for the Internet and similar networks, and it is generally the most popular protocol stack for wide area networks. It is commonly known as TCP/IP [190], because of its most important protocols: transmission control protocol and internet protocol, which were the first networking protocols defined in this standard. TCP/IP provides end-to-end connectivity specifying how data should be formatted, addressed, transmitted, routed and received at the destination. It has four abstraction layers, which are used to sort all internet protocols according to the scope of networking involved [2, 3]. From lowest to highest, these layers are: network access, internet, transport and application.

In a layered network design, each layer acts independently of the way that the lower layer is implemented and how its services perform [39]. Owing to this characteristic, it is clear that each layer could be optimized independently. However, there is evidence that the joint optimization of the layers, namely the cross-layer optimization in wireless networks helps to significantly improve performance.

Cross-layer optimization is an escape from the pure waterfall-like concept of the OSI communication model with virtually strict boundaries between layers. The cross-layer approach transports feedback dynamically via the layer boundaries to enable the compensation for e.g. overload, latency or other mismatch of requirements and

resources by any control input to another layer. Such a scheme allows communication between layers by permitting one layer to access the data of another layer to exchange information and enable interaction. For example, having knowledge of the current physical state will help a channel allocation scheme or Automatic Repeat Request (ARQ) strategy at the Medium Access Control (MAC) layer in optimizing tradeoffs and achieving throughput maximization. Recent research has shown that cross-layer optimization leads to improved multimedia performance over existing wireless networks [198, 96, 100, 98, 80, 75, 196, 191, 128, 19, 20, 97, 157].

Our study adopts a cross-layer design, where the physical layer, data link layer, network layer, and application layer are allowed to cooperate with each other with the goal of overcoming the delay difficulties and improve the QoS of real-time applications. Indeed, the majority of VSN applications require real-time data from sensor nodes [181], i.e., data shall be transmitted from nodes to the end-user within an extremely limited time frame, such that the total transmission delay is imperceivable. Yet, there are some negative factors that affect real-time system performance. Data imaging, processing, and transmission, in conjunction with the constraints that govern wireless channels (e.g., available bandwidth or bitrate, modulation scheme, video coding standard, wireless access method) can dramatically slow down real-time system response, thereby deteriorating the overall efficiency of the VSN.

Wireless channels are generally unreliable due to frequent errors that occur during wireless data transmissions. Thus, if our primary concern is to maintain a good level of video quality, we should aim at maintaining a low transmission Bit Error Rate (BER). To this end, suitable dynamic adjustment of sensor nodes' transmission parameters is required to maximize VSN performance. Specifically, each sensor node has a bitrate that can be used for both source coding and channel coding, while it also has an amount of power necessary for sensing, processing, and transmission of the captured data. Hence, the source coding rate, channel coding rate, and power constitute the transmission parameters of each node. Naturally, each node compresses the captured data at a different source coding rate according to the detected amount of motion in each scene. Thus, channel coding rate shall be different for each node. Under a total bitrate constraint, a higher source coding rate results in a lower channel coding rate, and consequently, higher levels of power are required for data transmission due to the lower protection from channel errors.

Data transmission is scourged mainly due to multi-path fading, shadowing at the physical layer, and co-channel interference at the MAC layer [13]. Hence, channel coding is used to increase the reliability of the transmissions. ARQ and Forward Error correction (FEC) are two widely used schemes, suitable for error-correction over wireless channel transmissions [216]. The former scheme is based on the retransmission of missing data packets. Its weak points are the need for a feedback channel and the time required for recovering missing packets, which impose significant limitations in real-time applications. Instead, FEC schemes can tolerate some amount of losses, allowing

data transmissions under lower power consumption. However, in the special cases where there are too many losses, they can be handled by usual ARQ techniques.

In addition, given that sensor nodes are battery-operated systems, energy control determines the lifetime of their battery. The high complexity of data processing and analysis that accompanies large amounts of video information, along with the real-time application requirements for data transmissions, result in rapid battery drain of the sensor. Energy conservation can be achieved through proper power control in order to maximize the sensors' energy-efficiency [102, 114, 70]. Besides that, multimedia content transmissions that have high bandwidth demands shall be taken into consideration. In [221] a bandwidth management framework has been proposed to coordinate multiple video flows in order to overcome wireless channel resource limitations. Further VSN challenges are highlighted in [225].

2.2 Resource Allocation using Quality-Based Criteria

As it was mentioned earlier, in our study we consider a wireless DS-CDMA VSN, where we assume that the nodes in the network are deployed to survey a large area and are equipped with a video camera. Some of the nodes are imaging relatively stationary fields, while others are imaging scenes with a high level of motion. In a DS-CDMA VSN, low-motion scenes can be source encoded at a lower bitrate, thus a larger bitrate may be used for channel coding. Therefore, nodes that image such scenes can afford to use a lower transmission power. It is important for a node to transmit data at low power, since it both increases battery life and reduces interference to the transmissions of the rest of the nodes. Actually, increasing the transmission power of a node improves the quality of the transmitted video, but it also degrades the video quality of the other nodes due to the increased interference. This effect can be alleviated by properly determining the transmission parameters of all nodes, such that the resulting distortions adhere to the application requirements. Hence, the necessity for a joint optimization of the parameters of all nodes becomes evident.

The nodes communicate directly with a CCU, which receives data from all source nodes, performs channel and source decoding to obtain the received video from each node, transmits information to the nodes and asks for adjustments to their parameters, considering their needs for both compression and error protection during transmissions. For example, it can request that the video of specific nodes be transmitted at a lower picture quality and bitrate, if the content of the video is deemed of secondary importance.

The source coding rates, channel coding rates and power levels are the transmission parameters of the nodes, considered in our problem formulation. Source coding rate determines the compression rate of a video sequence, while channel coding rate defines the relative protection of a transmitted video sequence. Regarding the power level, on one hand, it should be adequately high to permit data transmission and maintain the
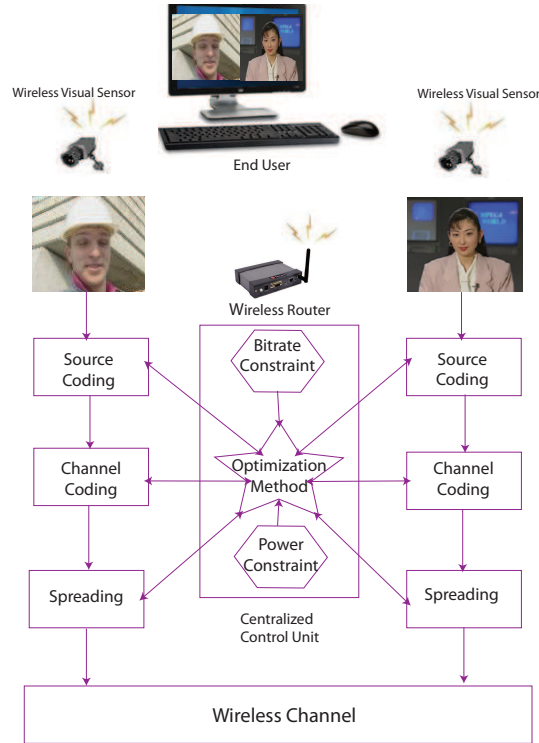
Figure 2.1: A wireless VSN with two diverse video priorities.

quality of the video reception. On the other hand, it needs to be adequately low in order to prolong battery lifetime, keep interference at low levels among the nodes, and efficiently exploit channel capacity, resulting in high system QoS. Figure 2.1 gives an insight of a wireless VSN with two diverse video priorities.

In order to optimally and jointly allocate system resources to all nodes, we consider two quality-based optimization criteria aiming at video distortion minimization. The first one, called the Minimized Average Distortion (MAD), minimizes the overall average video distortion of the network, neglecting fairness among the nodes. The second one, called the Minimized Maximum Distortion (MMD), minimizes the maximum distortion among all nodes of the network, promoting a rather unbiased treatment of the nodes. It should be noted that for both criteria, we require that the total bitrate is identical for all nodes.

Both of these schemes have been previously used in a number of similar resource allocation problems. In [119] source-controlled resource allocation has been investigated for orthogonal transmission of heterogeneous sources over a multi-user system. Different scenarios are considered, where the system resources rate, bandwidth, and power are shared among all users in such a way that their average distortion is minimized. In [228] the authors have considered the issue of how diverse receivers of a video stream should be grouped, where each group receives a Multiple Description with Forward Error Correction (MD-FEC) coded bitstream optimized for that group, so that the average video distortion is minimized across all receivers. Additionally, in [206] the

problem of bit allocation among shape, texture and motion in object-based video coding has been studied and solutions are provided based on the criteria that minimize the average and maximum video distortion.

In [168] the MMD criterion for the optimal bit allocation among dependent quantizers has been described, while also the MAD criterion has been used for the same purpose. Similarly, a scheme that maintains a fixed probability of packet loss and optimally allocates bits in order to minimize the maximum expected distortion has been proposed in [46]. In that work, a situation where a video sequence is to be compressed and transmitted over a wireless channel is considered, with the goal of limiting the amount of distortion in the received video sequence.

Moreover, we have also suggested the use of both MAD and MMD criteria in some of our earlier works [157, 20, 19]. However, in these works, MAD and MMD have been applied to purely discrete optimization problems, since the source coding rates, channel coding rates, and power levels assumed discrete values solely. Contrary to this, in our current research, we consider continuous power level values within a reasonable prespecified range, while source coding rates retain discrete values, since channel coding rates can take values only within a finite discrete set [61]. Thus, the resulting problem is modeled as a mixed-integer optimization task.

Some promising preliminary results of tackling the aforementioned problem led to a publication appeared in [131]. Additionally, this work ([131]) has been extended to include more extensive experimentation as well as the application of metaheuristic optimization algorithms. Thus, a more comprehensive version of [131] was published to the Applied Soft Computing journal [136]. More details on the novel aspects of this paper ([136]) can be found in Section 2.5, where we discuss about the optimization methodologies employed in our research. Also, we have applied the MAD and MMD criteria to the following research works [85, 84, 86, 82].

Other than MAD and MMD, there is also another well-known scheme that is commonly used in order to address similar resource allocation problems, called the Maximized Total Utility (MTU) criterion. In our study, we use the particular scheme so as to compare the solutions obtained by MAD and MMD as well as by some game-theoretic schemes as we discuss in Section 2.3. In the literature, MTU is usually encountered into two versions, the *unweighted* and *weighted* ones. The works presented in [50, 187, 151, 147, 47, 108, 12] make use of the unweighted version of the MTU.

Specifically, the work presented in [50] has solved the resource allocation problem of maximizing the sum of transmitter utilities subject to a minimum and a maximum data rate constraints per link and peak power constraints per node in a wireless multihop network. The study in [187] has addressed the opportunistic spectrum access problem, by developing a model so as to analyse a scenario in which the nodes of a wireless network seek to agree on a fair and efficient allocation of spectrum. In that work, the solution given by the MTU criterion (called as MaxSum in [187]) has been compared

for its effectiveness with the solutions provided by other three optimization criteria. The objective of [151] has been to schedule uplink transmissions in order to maximize the overall system utility, under explicit fairness constraints. In [147] the maximum total system quality has been one of the strategies used for solving the problem of optimal resource distribution among autonomous wireless stations. The maximum total system quality, as it is declared by its name, leads to the maximum sum of the wireless stations' qualities, while the quality differences among the wireless stations is significant. The work in [47] has analyzed scenarios in which self-interested agents negotiate with each other in order to agree on deals to exchange resources. In the same paper, the authors have identified the welfare enjoyed by a society of agents with the sum of the values ascribed by the individual agents in that society to the resources they hold in a particular situation. The behavior modeling and analysis of the dynamics in a colluders' social network in order to achieve different fairness of collusion has been studied in [108]. In that work, human behavior has been analyzed by four bargaining criteria. According to the max-sum solution, all the members in the colluders' social network have the same goal so that they are willing to maximize the total utility over the whole social network.

On the other hand, some of the works that utilize the weighted version of the MTU criterion can be found in [55, 208, 212, 12]. In [55] an optimal feedback allocation policy for cellular uplink systems has been proposed, where the base station has a limited feedback budget. The optimal allocation policy of this paper involves solving a weighted sum-rate maximization problem at every scheduling instant. The work presented in [208] has studied the multi-user resource allocation problem in Orthogonal Frequency Division Multiple Access (OFDMA) networks, extending the traditional network utility maximization problem into a more general framework of weighted network utility maximization. Furthermore, in [212] a joint subcarrier and power allocation algorithm has been proposed for maximizing the weighted sum rate in multiuser OFDMA downlink systems. Lastly, in [12] the scheduling and resource allocation problem for the downlink in a CDMA-based wireless network has been considered. This problem reduces to maximizing the weighted throughput over the state-dependent downlink capacity region, while taking into account the system-wide and individual user constraints.

2.3 Resource Allocation using Game-Theory

The main challenge in VSNs is the coordinated behavior of each node constituting the network, such that it maximizes the overall system performance within the various resource constraints. For this purpose, in order to tackle the resource allocation problem of this study, except for the use of quality-based criteria, we also resort to game-theoretic solutions.

Game theory is a branch of mathematics with extensive applications to social and

formal sciences, ranging from economics to biology to computer science and other disciplines [21]. Specifically, game theory studies the interactions of different factors in order to investigate matters such as monetary distributions in economics, the emergence of animal communication in biology, or even multi-agent cooperation in artificial intelligence and resource allocation issues in various network infrastructures.

The agents that participate in a considered game sometimes coordinate their behavior to achieve a common goal, while some other times each player may follow an independent, selfish policy, aiming exclusively at its own profit. Therefore, game theory can be considered as the umbrella that encompasses two distinct branches: *non-cooperative* and *cooperative* game theory. The first one attempts to rationalize the selfish actions of the players, while the second one studies the collective rationality of the players.

In more detail, non-cooperative game-theory is mainly concerned with the mutual interactions among intelligent individuals, striving to achieve their own goals. Several applications of non-cooperative game theory in wireless networks are provided in [120, 14, 63, 62]. In all these works, the Nash equilibrium appears as the solution achieved when the players compete with each other. A Nash equilibrium is reached when the strategy of each player is the best response to the strategies of the other players. However, such selfish behaviors often lead to suboptimal solutions, in the sense that user collaboration could promote an improved outcome, favorable for all players of the game.

In cooperative game theory models, the players coordinate their strategies by forming coalitions, in order to agree on a mutually acceptable division of the payoff. This aspect of game theory is also used in wireless networks for obtaining unbiased and efficient resource allocation schemes, avoiding disproportional allocations or resource depletion. Cooperative game theory concepts have been used to solve the opportunistic spectrum access problem [187]. Also, the issue of multi-radio resource allocation in generic heterogeneous wireless networks has been addressed based on the idea of network technologies cooperation [96].

Bargaining theory is an area of cooperative game theory that includes the notion of negotiation. Through the bargaining process, players are encouraged to choose one among many other possible outcomes, following the strategy indicated by the rational model. The outcome that determines the final share among all candidate parties constitutes a *bargaining solution.* Bargaining theory has been applied to distribute resources at a relay node to multiple source nodes [221]. In [29] the problem of downlink resource allocation for a multi-user Multiple-Input Multiple-Output (MIMO) OFDMA system has been considered through the bargaining perspective, while different fairness policies targeting at efficient resource management have been proposed in [147, 143, 117].

Game theorists have suggested several bargaining solutions so as to resolve resource allocation problems. The Nash Bargaining Solution (NBS) was the first among many bargaining solutions. A game-theoretic model developed in [187] has been used to

analyze a scenario in which the nodes of a wireless network seek to agree on a fair and efficient allocation of spectrum. For this purpose, the NBS has been applied, satisfying this dual requirement. In [218] a joint subcarrier assignment and relay power allocation problem has been formulated as a Nash bargaining problem with fairness consideration and practical constraints, in order to enhance system efficiency by exploiting multi-user diversity. The problem of fair and optimal bandwidth allocation among multiple collaborative video users has also been solved with the help of the NBS in [143]. However, in [143] no specific network setup has been assumed.

In [97] we have applied axiomatic bargaining game theory [153], which belongs to the broader category of cooperative games. Axiomatic bargaining defines the properties (axioms) that shall be adhered to by the optimal solution, and they serve as criteria for rejecting other candidate solutions, until a unique optimal solution is finally selected. Specifically, in [97] we have proposed the use of the Nash bargaining solution in the game of resource allocation in a wireless VSN that uses DS-CDMA. The objective has been the amelioration of the quality of the videos received by the CCU from each node, taking into account the fact that different nodes image videos with varying amounts of motion. Since the simultaneous maximization of the video qualities of all nodes is not possible, we have employed the NBS in order to pinpoint one of the infinite Pareto-optimal solutions, based on the stipulation that the solution should satisfy four fairness axioms. Specifically, this solution promises fairness for all nodes, taking into account the amounts of motion in the videos they capture.

In that work ([97]), the *disagreement point*, which is the vector of minimum utilities that each node expects by joining the game without cooperating with the other nodes, has been assumed to be the Nash equilibrium. Furthermore, finding the NBS has involved solving an optimization problem where the Nash product is maximized. In the same paper, all parameters to be optimized (source coding rates, channel coding rates, power levels) have been assumed to take values from discrete sets. Thus, a discrete optimization problem has been formulated and solved.

In our more recent research, we have extended the work of [97] by assuming that the power levels can take values from a continuous set. Clearly, this option for the power levels offers flexibility to the CCU to perform better management of the nodes' transmission parameters, achieving in this way better end-to-end video quality for each node. Since the source-channel coding rate combination can only take discrete values, the resulting optimization problem is a mixed-integer problem, solved using intelligent optimization. It is worth mentioning that a constraint is imposed to each node of the network; the total available bitrate can be shared between source and channel coding. Additionally, driven by the fact that in a considered game, users' collaboration promotes improved outcomes favorable for all players participating in the game, we take careful treatment to the optimal setting of the disagreement point, while we also consider a different definition for the utility function. A preliminary version of the specific research work has been presented in [132], where the disagreement point is set by the system

designer and it does not correspond to the Nash equilibrium.

More light on the specific problem is shed in the journal paper we have published in [137], where we introduce two versions of the NBS that differ in the definition of the bargaining powers, which declare the relative advantage that each class of nodes has in the negotiation. The first variant (NNBS) treats equally each individual node of the VSN, while the second variant (CNBS) provides equal treatment to each class of nodes. The proposed optimization schemes can be used for any wireless VSN with a centralized topology that uses DS-CDMA for data transmission. They keep low computational complexity, especially after the assumption of node clustering, based on the amounts of the detected motion in the videos they record. Given this assumption, fewer parameters need to be estimated and thus, less time is required for their computation. Additionally, the specific schemes not only provide Pareto-optimal solutions, but also guarantee fairness for all nodes of the VSN, as their fairness axioms state. Also, we have applied the NBS in different resource allocation scenarios in the following works [85, 84, 86, 82, 38, 83].

Besides NBS, another solution that is widely used for addressing resource allocation problems is the Kalai-Smorodinsky Bargaining Solution (KSBS). It is extracted from the field of game theory and has been proposed by Kalai and Smorodinsky in 1975 [77]. In our research, we have explored the specific solution to a video-quality optimization problem in order to ensure the QoS required by wireless DS-CDMA VSN applications, allowing continuous values for the power levels and discrete values for the source and channel coding rates. In the considered resource allocation game, the nodes play the role of the *players*. Increasing the power level of one node will improve its received video quality. However, the increased interference will reduce the video quality of the other nodes. In order to tackle the aforementioned problem, we apply axiomatic bargaining game theory [153], with the goal of maximizing the quality of the transmitted video that reaches the end-user. Axiomatic bargaining defines a set of axioms that the optimal solution should satisfy. In this way, all but one candidate solutions are rejected, since they fail to satisfy all axioms. Thus, a unique optimal solution is finally selected.

This study has resulted in a conference publication presented in [133], where the promising preliminary results regarding the performance of the KSBS have offered strong motivation for the further investigation of the aforementioned resource allocation problem. In light of this, an extension of [133] has resulted in a journal publication in IEEE Transactions on Circuits and Systems for Video Technology [135]. Specifically, in [135] we apply the KSBS to non-convex utility spaces according to [36], relaxing the requirement of convex feasible sets, in order to find a fair utility allocation resulting from the optimal determination of the nodes' transmission parameters, considering the assumption of a fixed available bitrate for each node that should be shared between source and channel coding. The KSBS is derived geometrically, directly from the graphical representations of the nodes' utility sets. In the same work, our attention is focused on the reliable evaluation of the KSBS performance in the quality and resource

domains.

The specific game-theoretic solution has also been applied in the past to find appropriate rules for the allocation of the available resources in various network architectures. In [187] besides the NBS, the KSBS has also been used to explore an efficient spectrum sharing for the nodes of the wireless network. The scheduling of multiple users to access channels has been discussed in [221], taking into account a maximum rate constraint of each source as well as a minimum rate requirement. In that paper, both the NBS and KSBS have been applied to address the problem. The problem of fair and optimal bandwidth allocation among multiple collaborative video users discussed in [143] has been resolved using both NBS and KSBS. The KSBS ensures that all users incur the same utility penalty relative to the maximum achievable utility. Additionally, in [147] the KSBS has been used to distribute the resources optimally and fairly among autonomous wireless stations, considering their current channel conditions, content characteristics, and cross-layer strategies.

In [95] a game-theoretic approach for resource allocation using cooperative games has been presented, where available network technologies cooperate to simultaneously allocate resources to the application requests. In that work, the KSBS determines the amount of allocation by each network technology. The application of KSBS to the problem of downlink resource allocation for multi-user MIMO OFDMA systems has been proposed in [29]. Additionally, [117] has presented a new system resource allocation framework for multimedia systems that perform multiple simultaneous video decoding tasks. The available system resources and the video decoding task's characteristics have been jointly considered in order to determine a fair and optimal resource allocation using the KSBS.

Continuing, a fully-centralized scheme based on the KSBS has dealt with the problem of resource allocation in wireless CDMA communication networks in [52]. A fully centralized scheme requires the base station to know all details including the users' utility, which may not always be possible. The problem of optimal allocation of bandwidth to multimedia applications within the operator network and the distribution of excess bandwidth among operators has also been confronted using the KSBS in [94]. A brokerage-based decentralized resource management scheme for multi-user multimedia transmission over networks has been presented in [144]. In that work, the autonomous behavior of multimedia users that stream video over the networks has been addressed with the Kalai-Smorodinsky approach, which explicitly considers the utility impact for different resource allocation schemes. Moreover, [71] has introduced the KSBS as well as three other game-theoretic solutions, which have been applied as OFDMA schedulers. They have been compared in two scenarios with respect to sum throughput, per user throughput, frequency band sharing and scaling with the number of users.

Furthermore, the work in [145] has proposed a utility-based resource management scheme for multi-user multimedia transmissions over networks. To manage the avail-

able resources, the resource manager deploys bargaining solutions from economics in order to explicitly consider the utility impact for different resource allocation schemes, while the authors mainly focus on the KSBS, because it can successfully model relevant non-collaborative utility-aware fairness policies for multimedia users. Lastly, in [146] decentralized solutions for resource negotiation have been proposed, where multiple autonomous users self-organize into a coalition, which share the same network resources and negotiate the division of these resources by exchanging information about their requirements. In that work, the KSBS has been considered as it can provide a fair division of resources for autonomous multimedia users.

2.4 Fairness Investigation

All of the schemes we consider in our research so as to solve the problem of optimal resource allocation, i.e, the MAD, MMD, the unweighted and weighted versions of the MTU and NBS, as well as the KSBS are able to provide Pareto-optimal solutions. Therefore, there is no single scheme that would be selected by all nodes to be the best. Considering the tradeoffs between video quality and power consumption that result after using a specific scheme, we engage in an effort to evaluate each examined scheme under different fairness aspects.

On the grounds that an ideal scheme offers high amounts of total utility cumulatively for all nodes, behaves equally fairly to all of them by assigning similar utilities and also consumes low amounts of power for all nodes, for the results evaluation obtained from all presented schemes, we investigate four different fairness notions (considering that the nodes of the network are clustered into two classes, based on the amount of motion in the captured scenes).

Firstly, we apply a metric [189] that captures both performance and fairness issues, assuming that the total utility varies per scheme. Secondly, we compute the Jain's fairness index [74] in order to investigate the "equality" of the resource allocations achieved by each considered scheme (equal utilities for the nodes). Thirdly, we calculate the overall gained utility cumulatively for all nodes and fourthly, we measure the total power required by all nodes of each considered scheme. In this direction, we juxtapose the total consumed power relatively with the total utility gain in order to evaluate the results in the resource domain. Ideally, a desirable scheme would achieve high total utility, while being equally fair to all nodes and requiring low amounts of power. The specific fairness investigation of the solutions provided by all examined optimization criteria has resulted in a conference publication appeared in [134].

Continuing, various fairness metrics have been proposed in the literature to weigh the video quality impact of using different resource allocation policies [117, 147, 187]. Each metric studies performance and efficiency from a different point of view, considering different kinds of fairness for the nodes. In [117], different resource allocation policies have been used to determine a quality-fair resource allocation for decoding

tasks sharing a single resource constrained processor. In the same work, a metric that captures the quality requirements for each task has been used to compare these policies. Specifically, a factor of 0 indicates that a task achieves its minimum desired quality and a factor of 100 indicates that a task achieves its maximum desired quality. A negative value for this factor indicates that a task achieves below its minimum required quality, while a positive value indicates that the task achieves higher quality than the minimum.

In [147] different resource management strategies have been compared in terms of the maximum quality drop, while the optimal strategy minimizes this drop. A metric defined as the ratio of the largest quality drop among wireless stations in the network using each considered scheme to the quality drop incurred by the KSBS for the wireless stations has been proposed in that paper ([147]). For the KSBS, the quality drop is the same for all wireless stations. In addition, in [187] the metrics of average, minimum and standard deviation of channel capacity, the KSBS score and the NBS score have been used to investigate the spectrum allocation achieved by the bargaining solutions. Furthermore, in the same paper, the bargaining solutions have been compared with the allocation that maximizes the sum of channel capacities.

2.5   Metaheuristics in Resource Allocation

As it has been previously discussed, in the resource allocation problem we consider, the power levels are allowed to take continuous values, while the source and channel coding rates retain discrete values. Therefore, a mixed-integer optimization problem is formulated, where the solution methodology is a significant issue for all examined optimization criteria, requiring careful treatment.

Deterministic mixed-integer programming methods can be used to tackle such problems. However, traditional optimization algorithms on mixed-integer problems may exhibit declining performance. For example, traditionally, infeasible primal-dual interior-point methods have two main perceived deficiencies, i.e., lack of infeasibility detection capabilities, and poor performance after a warmstart [18]. Additionally, branch and bound approaches combined with outer approximation algorithms have been proposed in the literature [54] for the same purpose. Despite the robustness that usually accompanies deterministic algorithms, various issues may arise. For instance, they may require significant implementation effort and expertise and their required memory and running time can be exponentially increased with the number of integer variables. In addition, such methods usually require the existence of derivatives and they are sensitive to the initial conditions provided by the user.

On the other hand, established population-based optimization algorithms can offer satisfactory solutions at the cost of reasonable computational requirements and minor implementation effort. Also, they concurrently evolve a population of candidate solutions that may constitute useful suboptimal alternatives with slightly different

characteristics than the optimal one. Applications of population-based metaheuristics such as evolutionary algorithms, ant colony, greedy randomized adaptive search procedure, and particle swarm optimization in wireless sensor networks can be found in [42, 179, 158, 101]. The highly appreciable properties of metaheuristic algorithms have triggered our interest in using such methods to the resource allocation problems under investigation.

In particular, we employ the Particle Swarm Optimization (PSO) algorithm, which is a computational intelligence approach that draws inspiration from social dynamics. The stochastic nature of PSO along with its ability to efficiently work in highly-complex environments with uncertainties, relieves the user from the burden of presenting an appropriate initialization to the algorithm. The particular method has been introduced in 1995 by Eberhart and Kennedy [92, 44] as a stochastic algorithm for numerical optimization tasks. It is a population-based algorithm, based on models that simulate flocking behavior, while it has close ties with the concurrent concepts of *emergent* and *collective behavior* [121]. Its dynamic is governed by fundamental laws encountered in swarms in nature, hence, it is categorized as a *swarm intelligence* algorithm within the wider field of intelligent optimization [150, 48]. Its ongoing popularity can be attributed to its efficiency in tackling a plethora of scientific and technological applications and complex engineering problems as well as to its easy implementation, which renders it accessible to researchers from various disciplines [150]. In the context of the current study, the PSO algorithm has been applied to the following research works [131, 132, 136, 137], while applications of PSO in similar problems to the ones of the current research are encountered to the works presented in [85, 83, 84, 38, 86, 81, 82].

Apart from PSO, for comparison purposes, we also investigate the performance of three deterministic algorithms that are used as benchmarks, including Active Set (AS) [60, 113, 123, 166], Interior Point (IP) [25, 26, 202] and Trust Region Reflective (TRR) [34, 35]. AS is an iterative method that is used for solving a sequence of quadratic subproblems, guaranteeing the feasibility of the final solution. The IP approach to constrained minimization is applied so as to solve a sequence of approximate minimization problems. The TRR algorithm is a subspace trust-region method and is based on the interior-reflective Newton method described in [34] and [35]. Each iteration involves the approximate solution of a large linear system using the method of preconditioned conjugate gradients.

In addition, motivated by the promising performance of hybrid algorithms that combine population-based approaches with deterministic schemes, often called *memetic algorithms* [126], we also consider a hybrid algorithm that combines PSO with AS in order to solve the mixed-integer problem. Our proposed algorithm, called Hybrid Particle Swarm Optimization Active Set (HPSOAS), aims at exploiting the benefits of both PSO and AS, thereby increasing efficiency. It employs AS as a local optimizer for further improving the findings of PSO [136].

Therefore, PSO and HPSOAS are used as optimization solvers to the problems for-

mulated by the MAD, MMD, NBS and MTU criteria, where each of them defines a specific optimization objective. On the contrary, since axiomatic bargaining game theory is rather descriptive, we follow an equivalent geometric approach so as to obtain the KSBS, where this solution is reached directly from the graphical representations of the considered feasible sets. This novel approach can be found in [133, 135].

2.6 Reinforcement Learning in Resource Allocation

A great concern in most VSN applications is to provide mechanisms able to guarantee high levels of QoS in the real-time delivery of multimedia content. The time-varying nature and error-prone environment of wireless networks as opposed to the delay-sensitive and bandwidth-intensive real-time multimedia applications poses the need for the optimal configuration of the wireless transmission systems. Furthermore, the errors that occur during wireless multimedia transmissions in conjunction with the lossy source coding techniques deteriorate the quality of the video sequences at the decoder. Thus, careful treatment is also required during video encoding in order to acquire high coding performance and coding robustness to transmission errors.

In our previous works [19, 20, 97], we have assumed a wireless VSN, where an application-driven cross-layer optimization scheme has been proposed for the dynamic adjustment of the sensor nodes' transmission parameters across all network layers. Such a scheme provides the opportunity for increased network resource usage and user profit maximization, at the same time. In these works [19, 20, 97], the nodes' transmission parameters, that is the source-channel coding rates and the power levels, have assumed discrete values and the resulting optimization problems have been tackled using the brute force problem-solving technique.

A literature review demonstrates that, whereas joint source and channel coding as well as energy consumption minimization have been the main objectives in wireless VSN research [224, 209, 27], little evidence is available for the investigation of efficient coding techniques by applying adaptive Group Of Pictures (GOP) length, at the same time. This latter approach aims at the enhancement of video resiliency to channel errors during wireless transmissions. The works presented in [226] and [105] have proposed GOP structures adaptive to video content, without addressing resource allocation issues, at the same time.

The H.264/Advanced Video Coding (AVC) video coding standard defines three frame types for video coding: intra frames (IDR of I), predictive frames (P) and bidirectionally predictive frames (B). Intra frames are coded without reference to other frames, while the difference between P-frames and B-frames is the number of reference frames they are allowed to use for coding. An Instantaneous Decoding Refresh (IDR) frame is a regular I-frame with the constraint that pictures appearing after it in the bitstream cannot use the pictures appearing before it as references. A GOP, which is a group of successive pictures within a coded video stream, always begins with an IDR-frame,

and therefore any errors within the GOP structure are corrected by the next IDR-frame. However, scene changes or large motion variations between frames can be located anywhere within a GOP, creating the need for an IDR-frame placement with the goal of increasing coding performance.

In light of this, in the current research, we also deal with the dual problem of cross-layer resource allocation among the nodes of a wireless DS-CDMA VSN along with the optimal IDR-frame placement during the encoding process, based on the motion level included in each video sequence. The particular optimization problem is formulated through the use of the MAD, NBS and MTU criteria. In this case, we have to address a discrete optimization problem, since all nodes' transmission parameters, i.e., source coding rates, channel coding rates and power levels can take discrete values. Discrete optimization problems have been also resolved in our previous works [19, 20, 97]. However, as compared to our previous works, in our more recent research, we consider that each node is able to select among more possible values for the power levels. Furthermore, more motion levels are assumed so as to simulate a more realistic video transmission scenario. Combining these two considerations about more possible choices for the nodes' power levels and more levels of motion included in the scenes captured by the nodes, it is clear that the problem's dimensionality significantly increases, rendering the use of the traditional brute-force search algorithm rather impractical. Hence, in this direction, we abandon the Exhaustive Search (ES) algorithm used in [19, 20, 97], and enjoy the benefits of other innovative optimization methods extracted from the area of Reinforcement Learning (RL) [188].

RL provides an elegant framework for making decisions under uncertainty based on the maximization of the expected utility functions. A significant contribution of our study is the incorporation of an RL scheme in the resource allocation problem, which allows the controller (CCU) to make optimal decisions in unknown environments with very large or continuous state spaces. RL has been used extensively in control strategies for video quality processing [217], surpassing a lot of difficulties in the particular field. More specifically, RL discovers an optimal or near-optimal policy in the early stages of the learning process, while at the same time is able to adapt in potential changes of the environment. The benefit of the latter feature clearly emerges in the online case, where the environment changes dynamically over the time.

In our research, we use the tabular SARSA algorithm which is a model-free on-policy algorithm that belongs to the family of Temporal Difference (TD) algorithms [188]. The examined resource allocation problem is modeled appropriately as a Markov Decision Process (MDP) [156]. The particular approach exploits the received raw experience, discovering the optimal combination of the nodes' transmission parameters in a more efficient way. Roughly speaking, SARSA constructs a map that allows us to explore the best parameters with the minimum effort, starting by any randomly selected parameters' combination. Last but not least, the specific RL approach gives the opportunity to the proposed scheme to be used in an online mode. Hence, the specific research effort

has resulted to a conference publication presented in [141].

2.7 Unequal Error Protection for Video Quality Enhancement

In our current study, we consider the issue of resource allocation aiming at the assurance of high system QoS. In the same context, we also try to achieve this outcome by applying Unequal Error Protection (UEP) to H.264/AVC video sequences transmitted over noisy environments. UEP consists of allocating coding redundancy over the sequence of progressively transmitted packets, depending on the importance of the information bits [24]. Thus, better protection is provided to specific video parts so as to accommodate for receivers with poor link quality and assure a more graceful video degradation [201, 230]. The goal is the minimization of the expected video distortion at reconstruction.

In the literature, several studies have attempted to evaluate the packet loss effect on video quality. For the estimation of the distortion caused by packet losses and the resulting error propagation, the authors in [23] have proposed a model that evaluates video quality degradation in terms of Peak Signal to Noise Ratio (PSNR). In the same work, a distortion-based video packet prioritization mechanism for streaming over networks, which can be used for UEP, has also been introduced. Moreover, P. Pérez et al. [152] have proposed a model for measuring the packet loss effect, where it is shown that packet priority and packet loss prediction models significantly improve the network Quality of Experience (QoE).

In [162] the effect of packet loss in terms of Mean Squared Error (MSE) on video quality directly from the video bitstream has been studied through the use of three different methods; a model that exploits only network-level measurements, another that extracts the spatiotemporal extent of the impact of the loss and yet another model that extracts sequence-specific information including spatiotemporal activity and effects of error propagation. The goal of the work presented in [107] has been to develop a packet loss visibility model, applicable to different GOP structures. The effectiveness of the proposed model has been validated on a packet prioritization scenario, where the network gets congested at an intermediate router and the router has to decide about which packets to drop such that the visual quality of the video is minimally impacted. Furthermore, in the same work, a Generalized Linear Model (GLM) is employed in order to predict the probability that a packet loss will be visible to a viewer.

In [130] a GLM-based model has been presented, targeting at slice prioritization of video flows for real-time H.264/AVC streaming, based on the estimated values of Cumulative MSE (CMSE) incurred by individual slice losses. In addition, the GLM-based CMSE prediction model developed in [129] has been evaluated in a large variety of GOP structures and lengths, as well as in different encoding bitrates. Also, the same work has examined both the cases of GOP-level slice prioritization and frame-level slice prioritization.

In our research, we focus on improving the accuracy of CMSE estimations provided by [130, 129], through the use of the Least Absolute Shrinkage and Selection Operator Regression (LASSO) tool [192, 193, 127]. More precisely, we calculate the true CMSE values as they result from the loss of each individual slice. In the following, a number of quality-relevant features are extracted from each slice of a video sequence during the encoding process, in order to be used for the estimation of the CMSE values. LASSO regression is performed so as to indicate the most important features of the dataset, providing accurate CMSE estimations at the same time. Based on the CMSE measured and estimated values, we group the slices into four priority classes, by applying a Quartile Based Prioritization (QBP) scheme, and a scenario where video sequences are transmitted over an Additive White Gaussian Noise (AWGN) channel, by applying UEP, is also considered. This work led to a conference publication presented in [142].

To the best of our knowledge, LASSO regression is applied for the first time to address slice prioritization issues, principally aiming at precise CMSE estimations, which will next guide the quartile-based slice classification. The specific linear regression method is able to select the features that have the strongest effects towards video quality, rejecting those that do not essentially capture the effect incurred by each individual slice loss. In this way, the problem's complexity is significantly reduced, since in this case a smaller set of features is necessary for CMSE estimation, something especially important in time-critical applications. Moreover, LASSO approach is simpler compared to regression techniques that require additional methods for feature selection prior to the estimation of a response variable (for example, combining Principal Component Analysis (PCA) [99] for feature selection with Ordinary Least Squares (OLS) [22] for the estimation of the response variable [163]).

Particularly, we study the LASSO regression through two different architectures; Global LASSO (G.LASSO) and Local LASSO (L.LASSO). In G.LASSO, a single regression model is trained for all slice types together. Moreover, motivated by the fact that the values for some features are closely dependent on the considered slice type, in L.LASSO we examine the case where each slice type has its own sparse regression model, in an effort to capture more precisely the effect of a slice loss. In addition, in L.LASSO the estimation results for the separate models are combined so as to compute the performance statistics for all slice types together.

2.8 Automatic Video Quality Assessment

Telecommunication networks are faced with increased demand for resources due to continuous growth in data traffic. The usage of bandwidth-intensive applications is increasing, leading to congestion in the access networks. On the other hand, the user expectations are growing over time. Any slight disturbances in the network result in a decay in the user satisfaction level, which is an undesirable situation for the network operators. Due to the availability of several competitors in the market, user satisfaction

has become the most critical component for gaining competitive edge in the market.

QoE is the overall acceptability of an application or a service, as perceived subjectively by the end user [73]. Therefore, real-time assessment of QoE is becoming the primary tool for network operators in managing the networks [173]. Although human observer is the most reliable source for Video Quality Assessment (VQA), the collection of video subjective scores implicates a series of constraints. In subjective quality assessment tests, a number of human subjects are required to rate the video quality of the presented content and the product of such assessments is typically a Mean Opinion Score (MOS) for each test sample, which corresponds to the average value of the scores given by the panel. Such tests have to be carefully designed and performed and require a significant number of viewers available to perform the specific task [41]. An alternative approach to subjective tests is crowdsourcing [88], where the testing procedure is conducted through the Internet. By following this method, one can access a wider range of evaluators, while keeping the financial cost low and obtaining results similar to those of lab-based subjective tests. Even in such a case, these tests are time-consuming and cannot be used in real-time applications.

In the last two decades, many modern methods of perceptual VQA have been developed and they can be implemented automatically, based on quality-relevant features of a video. The goal of such objective methods is the computation of a perceptual quality estimate that correlates well with the results of subjective assessment. A classification of the objective methods can be made on the basis of the reference information used for quality estimation as given in the following [214]:

- Full-Reference (FR) metrics have full access to both of the original and impaired versions of a video.

- Reduced-Reference (RR) metrics have access to some key features extracted from the original video sequence and full access to the impaired video sequence.

- No-Reference (NR) metrics have access only to the impaired video sequence.

Generally, FR methods have the capacity to provide the most accurate estimations of video quality since they can compare the pixels of original and impaired video sequences on a frame-by-frame basis. Because of the dependence on the original video, FR methods are mostly suitable for offline applications, such as encoder performance comparisons. In addition to the impaired video data, RR metrics can also access selected features of the original video. These features can be sent to the receiver through an ancillary channel [154] or alternatively, they can be embedded in the video content itself, by using techniques such as watermarking [227].

For quality estimation, NR methods make use of either the encoded bitstream of the impaired video or they access the decoded pixels of the impaired video, or both of them. The methods that fall in the former type are called NR Bitstream-based (NR-B) metrics, while those that fall in the latter type are called NR Pixel-based (NR-P) metrics.
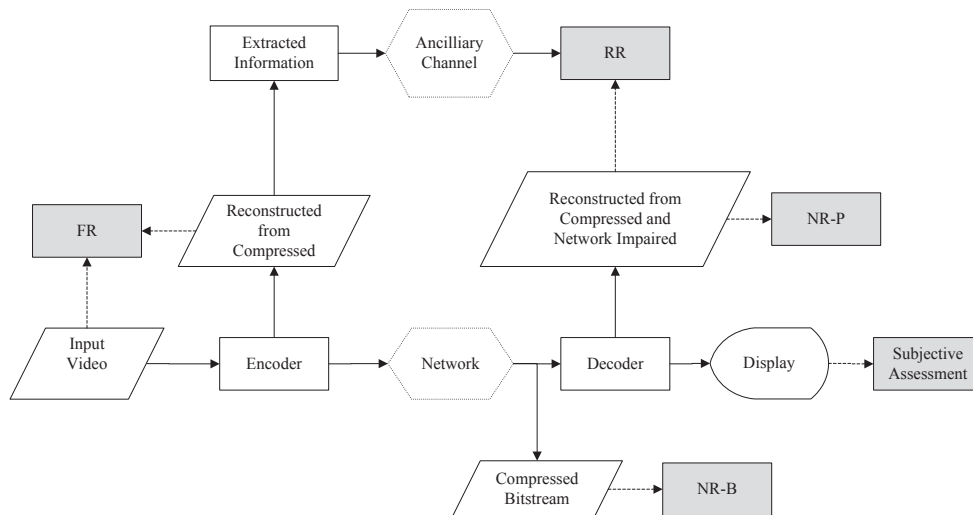
Figure 2.2: Categories of VQA methods.

Because of the limited or no dependence of the original video, RR and NR methods are suitable for real-time applications and online quality monitoring of the streaming videos [162], though quality estimation with limited available input information can be challenging [172]. Figure 2.2 highlights the use-case scenarios of the aforementioned reference-based approaches of VQA, by showing the types of the data being accessed.

During the last years, a considerable part of the scientific community has focused its interest on efforts for the development of objective video quality metrics that target at reliable and accurate modeling of subjective VQA. However, some of these metrics are of FR type [110, 111], which are rather impractical in most cases. Some RR metrics of VQA have been presented in [223, 182, 116]. In [223] the authors have designed an RR metric that is targeted for applications related to wireless communications. It is built based on the principle that humans tend to have different impairment perceptibility according to the spatial and temporal affected regions of a video sequence. The work in [182] has presented a family of RR VQA models that differ in the amount of reference information required for video quality measurement, while [116] has proposed a wavelet-based video distortion metric that can operate in FR or RR mode, as required. Actually, RR metrics can be an alternative to FR metrics when the original video is not accessible. However, in some cases, the cost of maintaining an ancillary channel may be high for an RR approach, while such metrics may not meet the requirements of quality estimation in the event of a failure in RR data delivery to the receiver's end.

For these reasons, NR metrics are the most broadly applicable solution for VQA. A NR metric tested on MPEG-4 compressed video that estimates the PSNR on MacroBlock (MB) level has been proposed in [174] and a similar method that estimates the Structural SIMilarity (SSIM) index has been introduced in [175]. The study presented in [180] has described a PSNR estimator that considers only the compressed bitstream of an H.264/AVC coded video. However, the estimation of perceptual quality in terms

of MOS could be an applicable improvement for the works presented in [174, 175, 180].

A set of bitstream-based features related to slice coding type, coding modes, various statistics of motion vectors, and Quantization Parameter (QP) value have been employed in [90] with the goal of quality estimation of high definition television video, encoded by H.264/AVC. For the same purpose, statistics of boundary strength values of the deblocking filter, QP, and average bitrates have been used in [106] for H.264/AVC encoded videos. Also, a motion-based quality metric has been explored in [163] for H.264/AVC encoded videos as well. For this metric, some statistical features related to motion vectors along with the bitrate and frame rate have been calculated, and the PCA method [99] has been used to identify the parameters that can be the most influential in quality value. Similarly, a low complexity solution of VQA based on bitstream features has been proposed in [164]. An improvement of this approach has been included in [170], where the required number of features has been reduced so as to promote computational efficiency. In that work, an improvement has been noted in estimation accuracy by the virtue of the usage of an artificial neural network. A further improvement of [170] can be found in [171], where a larger set of parameters has been used and the estimation of subjective MOS has also been considered. However, the models built in [90, 106, 163, 164, 170, 171] are oriented towards capturing distortions due to lossy source coding only, and thus, they cannot be applied in the case of packet-loss impaired videos.

In [78] the authors have extracted a set of features from the MPEG-2 bitstream and have proposed two different modeling approaches: i) a tree classifier to decide if a packet loss is visible or invisible and ii) a GLM to estimate the probability that a packet loss is visible. In [79] the GLM approach has been extended for H.264/AVC bitstreams to model the visibility of individual and multiple packet losses. An application of the proposed GLM scheme to packet prioritization of a video stream, considering factors not only within a packet but also in its vicinity has been suggested in [107]. The visual effect of whole B-frame losses has been investigated in [28]. For this purpose, a GLM has been used to estimate the probability of the visibility of a B-frame loss and a router was able to decide about which frames to drop in a video transmission scenario, where the incoming bitrate is higher than the outgoing rate. However, the methods presented in [78, 79, 107, 184, 28] classify packets in a binary mode as visible or invisible based on the viewers' responses to the glitches they spot. For example, a packet loss has been assumed to be visible when the percentage of the viewers that identified an impairment has been over a threshold and invisible when this percentage has been under a threshold. On the contrary, [16] has introduced an NR bitstream-based model that predicts continuous estimates for the visibility of packet losses, and the impact of the lost packets on perceptual video quality has also been studied. However, most of these metrics mainly target the visibility of packet losses and a direct estimation of perceptual quality is not made by including also the features related to video coding.

Since degradation of perceptual video quality is incurred due to both lossy video

encoding and transmission impairments, there is the need for the development of a perceptual video quality model able to estimate possible video degradations due to both sources of distortion. The NR method presented in [220] estimates the quality of videos transmitted over wireless networks, using information from MBs of inter-frame encoded pictures of a video. The proposed method analyzes the impact of both encoding and channel conditions to the video quality degradation by using motion vectors and residual error from the received P-frame and/or B-frame. In addition, in [178] a QoE evaluation model has been proposed to estimate the end-users' perception on a video streaming service considering different video content types. This QoE model extracts key parameter information directly from degraded video frames in order to estimate the video QoE. A similar NR quality metric for networked video has been introduced in [219] using information extracted from the compressed bitstream only. This metric accounts for picture distortion caused by quantization, quality degradation due to packet losses and error propagation, and temporal effects of the Human Visual System (HVS).

In the context of the aforementioned related works, in our research we propose an approach that directly estimates video quality by employing perceptually-motivated video features, extending our previous studies presented in [164, 170, 171]. Particularly, we develop RR and NR models in order to estimate the perceptual quality of H.264/AVC video sequences, which are affected by packet losses. In more detail, a NR bitstream-based model is built so as to estimate the MOS as well as the SSIM [210] and Video Quality Metric (VQM) [1] metrics, which are known for their good correlation with subjective assessment. The RR and NR models utilize a different set of extracted features and are both oriented in estimating solely the actual MOS values. A variety of features that may have an effect on perceptual video quality are collected in order to be used for building the proposed models, using the LASSO regression technique. The RR set of features as a whole and the NR set of features as a whole are employed for the first time in our study, while we also introduce the utilization of 11 new features.

LASSO regression [192, 193, 127] is utilized in order to indicate the most useful features for making video quality estimations. To the best of our knowledge, this is the first time that LASSO is employed in video quality estimation problems. Additionally, in order to evaluate the efficiency of LASSO in terms of both model's sparsity and estimation accuracy, we employ the OLS regression method [22], as well as Ridge regression [22, 115, 67] on a preselected set of features, having performed sequential feature selection and particularly Forward Feature Selection (FFS) [53] on the complete RR and NR sets of features, as Ridge is not able to perform any feature selection.

Moreover, for assessing our models' performance, we emphasize on measuring the estimation accuracy, monotonicity, and error. In addition, we explore the performance statistics for two FR metrics that are oriented towards measuring video quality of digital video systems, that is VQM [1] and Perceptual Evaluation of Video Quality (PEVQ) [72], which are known for their good correlation with subjective assessment. Some preliminary results of our NR model has been presented in [138], while the largest part of the

specific research is included in [140].

Furthermore, a performance comparison of our proposed approaches is made with the following related works [124, 125, 31, 17, 109, 7, 183]. The study presented in [124] has proposed an FR method that uses both singular values and singular vectors as visual features, and a machine learning technique for feature pooling is also introduced. The work presented in [125] has proposed an RR metric that compares the phase and magnitude of the 2-D discrete Fourier transform of the reference and distorted images in order to compute visual quality. An NR bitstream-based quality metric that considers both the effects of lossy H.264/AVC video encoding and packet losses over internet protocol networks has been proposed in [31]. In [17] an NR video quality metric for H.264/AVC video transmissions in packet-based networks has been introduced, which uses features from the headers that encapsulate compressed video data. Similarly, in [109] an enhanced algorithm based on the G.1070 model [7] has been developed that compensates for the impact of varying video content characteristics on encoding bitrate. Lastly, genetic programming-based symbolic regression has been used in [183] in order to build a bitstream-based NR model. The used features characterize encoding settings, parameters related to network distortions and video content.

As it has been mentioned earlier, subjective quality assessment is the most reliable way of evaluating the quality of a video communication product. Nonetheless, the carrying out of subjective tests faces many challenges. For example, a large number of viewers are required and the viewers are not always available or willing to rate a large variety of video sequences with different kinds of impairments. In addition, through subjective experiments we are unable to get instantaneous measurements of video quality due to many practical limitations. Thus, a single quality value for the whole video does not provide any information about the individual quality of the video frames, making it impossible to know which parts of the video have the greatest influence in forming the viewer's judgement.

An alternative approach to subjective VQA is to automatically get video quality scores, through the use of objective metrics [30, 211], where an ideal objective metric is able to provide quality scores that highly correlate with human ratings. In the literature, many works construct objective metrics through the use of various machine learning techniques, such as Partial Least Squares Regression (PLSR) [89], Neural Networks (NN) [32], Support Vector Machines (SVM) [203], and Support Vector Regression (SVR) [124]. Each considered regression model takes as input a number of quality-relevant features that account for different types of distortion and influence the accuracy of MOS estimations [17, 109]. Theoretically, the larger the number of features, the better the estimation power of the regression model. However, having a large number of features along with a possibly small number of observations (sample set) involves the risk of fitting the model to the noise of the training data, being unable to generalize well to unseen data (testing data) [57]. For this reason, a feature selection procedure often takes place before video quality estimation [204, 37, 176].

In the context of the current research, we propose an NR quality estimation method for videos that are impaired by both compression artifacts and packet losses, having a two-fold goal: i) to improve the per-sequence MOS estimation accuracy through the development of a model which is robust and has a good generalization capability, and ii) to provide a reliable indicator for the quality of each frame of a video, offering an intuition about their individual contribution to the overall video quality score. In order to accomplish our goal we develop a new metric, which is able to provide quality estimates for each individual frame. The requirement imposed on this metric is that its average MOS value over the whole video sequence should highly correlate with the actual MOS of the video sequence. The results produced by the developed metric are used as the ground truth, which plays the role of the target variable in the regression procedure. Thus, we aim at the development of a NR method for the estimation of the MOS for each frame using features of the received video bitstream, which, when averaged over the whole video sequence, give an accurate estimation of the MOS of the video sequence. This piece of study led to a conference publication presented in [139].

The works presented in [87, 112] have elaborated on the concept of considering frame quality measurements and measurements over small parts of video sequences that guide the overall video quality rating. However, the goal of both [87, 112] is different from the one of our research mentioned earlier. In [87] a NR objective metric that provides two video quality scores per second so as to align with the subjective results of a Single Stimulus Continuous Quality Evaluation (SSCQE) method [15] has been introduced, and in [112] the authors have applied a fusion mechanism in order to integrate the scores from some video intervals into a final one, increasing in this way the correlation with the MOS.

CHAPTER 3

RESOURCE ALLOCATION OVER WIRELESS VISUAL SENSOR NETWORKS

3.1 Source Coding

3.2 Channel Coding

3.3 Direct-Sequence Code Division Multiple Access

3.4 Modeling and Estimation of the Expected Distortion

3.5 Optimization Criteria

3.6 Optimization Algorithms

3.7 Evaluation Metrics

3.8 Experimental Results

Visual sensor networks are comprised of typically low-weight distributed sensor networks that can communicate directly with a CCU. In our research, the nodes utilize DS-CDMA in order to access the wireless VSN. The CCU performs source and channel decoding to obtain the received video from each node and it transmits information to the nodes in order to request changes in their transmission parameters, such as source coding rates, channel coding rates and power levels, depending on the amount of motion detected in each video sequence. For instance, if the CCU considers that a node is imaging scenes of great interest, it tries to maximize the picture quality of the specific video by appropriately adjusting its transmission parameters.

3.1 Source Coding

Video compression is essential in communications due to limitations in the bandwidth of the communication channel, meaning that data representation with the smallest possible number of bits is an imperative need. Due to this, source coding is applied before data are conveyed through the network. Different video sequences have different bitrate requirements for their compression. Certainly, the compression requirements vary per video sequence, while the end-user can also determine which video sequences are source encoded at a higher rate, for each considered application. For example, if the end-user feels that some sensor nodes monitor more interesting scenes compared to some others, he/she will require these nodes to enhance the quality of the video they record. This means that more bits will be spent for compression in order to avoid significant degradation of the video quality.

In addition, video sequences with less motion can be source encoded at a lower bitrate while still maintaining good perceptual quality. On the other hand, video sequences that contain intense motion activity shall be compressed at a higher bitrate in order to avoid significant degradation of the video quality. Thus, assuming that the total bitrate is fixed, if a node needs a higher source coding rate, a lower percentage of the total bitrate is assigned to channel coding for error correction. Hence, in order to keep the BER at acceptable levels, the power level used for transmission must be increased. In our study, we assume that each node has the power required for video transmission over the VSN. Inevitably, the energy consumed for data transmission leads to shortening of the battery life and, due to the nature of DS-CDMA, to increased interference imposed to the other nodes of the network.

In our research, we employ the H.264/AVC video coding standard to compress the video sequences imaged by the nodes, using the Main profile, and High profile for $4:2:0$ format video for chroma sampling as well. This standard is an extremely efficient tool for coding and is targeted at many applications such as video telephony, storage, broadcast, and streaming [207] that can provide good video quality at substantially lower bitrates than earlier standards [186]. The Main profile was designed so as to provide high coding efficiency. Therefore, it includes B-pictures, context-adaptive binary arithmetic coding, context-adaptive variable-length coding and interlaced coding tools. As the error rates after FEC are not expected to be high, the error resilience tools, i.e., flexible macroblock ordering, arbitrary slice order, and redundant slices, are not included in this profile. The High profile has proved to be extremely efficient in coding, taking into consideration the available coding tools for the encoder. It is a super set of the Main profile and in addition to all the tools used in the Main profile, it includes 8×8 transform, 8×8 intra prediction and downloadable quantization/weighting tables. On average, this profile is reported to provide about 10% higher coding efficiency in comparison to the Main profile for 720p (1280×720 pixels) formats [207].

The coded video data are organized in Network Abstraction Layer (NAL) units, which are packets containing an integer number of bytes. These units are grouped into Video

34

Coding Layer (VCL) NAL units and non-VCL NAL units. The VCL NAL units contain the data that represent the values of the pixels in the video pictures, while non-VCL NAL units contain parameter sets and supplemental enhancement information [213].

3.2 Channel Coding

Channel coding is used to increase the communication channel reliability by increasing resistance to channel errors. Specifically, it adds redundancy in the video bitstream, unlike source coding which intends to represent data with the smallest possible number of bits. In cases where a video sequence uses fewer bits for error protection, it is necessary to increase the transmission power in order to keep the bit error rate at acceptable levels. In our study, an adaptive FEC scheme using Rate Compatible Punctured Convolutional (RCPC) codes is utilized for channel coding [61]. These are families of codes with different rates, which can be decoded by the same Viterbi decoder. However, we could also have used other channel coding schemes.

The use of RCPC codes allows the use of Viterbi's upper bounds on the bit error probability [27, 61, 200, 194], P_{b}, which is described by the inequality [61]

$$P_{\mathrm{b}} \leqslant \frac{1}{P} \sum_{d=d_{\mathrm{free}}}^{\infty} c_d P_d, \tag{3.1}$$

where P is the period of the code; d_{free} is the free distance of the code; c_d is the information error weight; and P_d is the probability that the wrong path at distance d is selected.

Let us assume that information is sent over a channel subjected to AWGN. Also, let Binary Phase Shift Keying (BPSK) be the employed modulation scheme. Then, the probability P_d becomes [61]

$$P_d = \frac{1}{2} \operatorname{erfc} \left(\sqrt{\frac{d R_{\mathrm{c}} E_k}{N_0}} \right), \tag{3.2}$$

where

$$\operatorname{erfc}(x) = \frac{2}{\sqrt{\pi}} \int_x^{\infty} \exp\left(-t^2\right) \, dt, \tag{3.3}$$

is the *complementary error function*; R_{c} is the channel coding rate; and E_k/N_0 is the energy per bit to Multiple Access Interference (MAI) ratio. The index k denotes the corresponding node of the network.

3.3 Direct-Sequence Code Division Multiple Access

DS-CDMA is the wireless VSN access method adopted in our study. This method allows all nodes to transmit over the same channel, sharing the same bandwidth. An

advantage of such systems is the lack of a fixed limit on the number of nodes accessing the same bandwidth. However, since all nodes transmit over the same channel, transmissions are affected by generated interference from the other nodes, mainly due to non-orthogonal spreading codes, possible asynchronous transmissions, and multipath fading. The target is to limit the interference as much as possible in order to ameliorate the video quality, retain low power consumption, and achieve effective exploitation of the system's capacity without affecting the integrity of the data transmission procedure. Therefore, we are interested in achieving the ideal tradeoff between power consumption and video viewing quality. On one hand, spending less power will limit interference but, on the other hand, low power amounts cannot guarantee sufficient video quality. Hence, power control is considered indispensable for a successful DS-CDMA system.

After source and channel coding, each data signal is assigned a unique spreading code, usually orthogonal or pseudo-random to the codes assigned to the other signals, such that the interference between the two signals is minimized. In order to transmit a single bit, a node actually transmits L *chips*, where L is the spreading code length, measured in chips. Usually, the *chip rate* (number of transmitted chips per second), R_{chip}, measured in chips per second, is identical for all nodes. We assume that the spreading code length is identical for all nodes. This, since the bitrate R_k, for node k, is equal to

$$R_k = \frac{R_{\text{chip}}}{L}, \tag{3.4}$$

a constraint on the chip rate corresponds to a constraint on the bitrate. DS-CDMA systems are usually interference-limited systems and therefore, it is common for the thermal noise and background noise to be neglected.

The power level S_k, for node $k = 1, 2, \ldots, N$ is given by

$$S_k = E_k R_k, \tag{3.5}$$

and is measured in Watts (W). The quantity E_k is the energy per bit, and R_k is the total bitrate used for both source and channel coding.

In fact, S_k refers to the power that is received by the CCU after the transmission of node k. Therefore, for given power levels, the required transmission powers for the nodes can be determined by a power control algorithm that is present in all practical DS-CDMA systems [59, 56]. Power control can track the attenuation due to the distance between transmitter and receiver, as well as the effects of fading. Assuming the Two-Ray Ground Reflection (TRGR) model as the propagation model, the transmission power, $S_{k_{\text{trans}}}$, for node k, is given by [160]

$$S_{k_{\text{trans}}} = \frac{S_k \, d_{\text{tr}}^4}{G_{\text{t}} \, G_{\text{r}} \, h_{\text{t}}^2 \, h_{\text{r}}^2}, \tag{3.6}$$

where d_{tr} is the distance between the transmitter (node) and the receiver (CCU), G_{t} is the transmitter antenna gain, G_{r} is the receiver antenna gain, h_{t} is the height of the transmitter antenna, and h_{r} is the height of the receiver antenna.

Concerning the total bitrate R_k, for node k, it is defined as

$$R_k = \frac{R_{s,k}}{R_{c,k}}, \tag{3.7}$$

and it is measured in bits per second (bps). The quantity $R_{s,k}$ represents the source coding rate of node k, also measured in bps, while $R_{c,k}$ is the channel coding rate of k-th node. Obviously, since $R_{c,k}$ is the ratio of the number of information bits over the total number of bits, it is a dimensionless number within the range $[01]$ [98].

A significant constraint considered in our problem setup refers to the total bitrate, R_k, used by each node for both source and channel coding. Specifically, each node shall transmit data using the same maximum bitrate. This constraint results from a fixed overall transmission chip rate, R_{chip}, and the use of the same spreading code length L, for all nodes (see Eq (3.4)). From the definition of R_k, it is clear that source and channel coding rates are inversely related quantities, i.e., higher source coding rates imply fewer bits available for channel coding, and vice versa. Hence, for a fixed R_k, the higher the source coding rate for a video sequence, the lower the video sequence protection from channel errors, and vice versa. Therefore, source and channel coding rates are two interdependent variables that can be considered as a pair.

In our investigation, we follow the assumption that interference can be approximated by AWGN [59, 27]. Thus, the energy per bit to MAI ratio is given by

$$\frac{E_k}{I_0} = \frac{\frac{S_k}{R_k}}{\sum\limits_{j \neq k}^{N} \frac{S_j}{W_t}}, \qquad k = 1, 2, \ldots, N, \tag{3.8}$$

where $I_0/2$ is the two-sided noise power spectral density due to MAI, measured in Watts/Hertz (W/Hz), and W_t is the total available bandwidth, measured in Hertz (Hz). Again, k refers to the corresponding node, while j refers to each interfering node.

In Eq. (3.8) the following fundamental assumptions are made

(a) The thermal and background noise are ignored.

(b) The spreading codes used are random and do not have any special properties.

(c) Interference suppression filters are not used.

Assumptions (b) and (c) suggest that no means is used to suppress or limit the co-channel interference, implying that each node admits the power of the other nodes totally as interference. If we drop assumption (a), i.e., assuming that thermal and background noise are rather significant, then, instead of E_k/I_0, we use the following energy per bit to MAI and noise ratio

$$\frac{E_k}{I_0 + N_0} = \frac{\frac{S_k}{R_k}}{\sum\limits_{j \neq k}^{N} \frac{S_j}{W_t} + N_0}, \qquad k = 1, 2, \ldots, N, \tag{3.9}$$

where $N_0/2$ is the power spectral density of the AWGN.

3.4 Modeling and Estimation of the Expected Distortion

The video sequences received by the CCU are degraded by both the lossy compression and the errors introduced by the channel. Clearly, there is a direct relationship between the bit error rate (bit error probability), P_b, and the distortion of the video sequences. In our research, for the estimation of the expected video distortion, $E[D_{s+c,k}]$, for node k, we use Universal Rate-Distortion Characteristics (URDCs) [98]. These characteristics show the expected distortion, $E[D_{s+c,k}]$, as a function of the bit error probability, P_b, after channel decoding. However, since video encoded with the H.264/AVC codec is designed to handle packet errors as opposed to bit errors, we need to calculate the resulting Packet Loss Rate (PLR). We assume that the video bitstream is packetized using the Real-time Transport Protocol (RTP). RTP provides a packet format for real-time data transmissions [167]. We calculate an RTP PLR from a certain BER, drop packets from the H.264 bitstream according to the RTP PLR, and pass the corrupted H.264 bitstream to the H.264 decoder to calculate the distortion of the video [19].

We assume that each RTP packet consists of a number of Link Layer (LL) packets. The link layer packet size is LL_{size}, measured in bits. Thus, the link layer PLR is

$$PLR_{LL} = 1 - (1 - BER)^{LL_{size}},\tag{3.10}$$

where PLR_{LL} is the PLR for a link layer packet of size LL_{size}. Similarly, we calculate the RTP PLR with

$$PLR_{RTP} = 1 - (1 - PLR_{LL})^{RTP_{size}},\tag{3.11}$$

where PLR_{RTP} is the PLR for an RTP packet of size RTP_{size}, measured in the number of link layer packets. We assume that we know when a packet has an error and we manually drop packets with any errors from the H.264 encoded video stream, in accordance with the PLR_{RTP} calculated from the BER.

Since channel errors are random, the video distortion $D_{s+c,k}$ of node k, which is due to both the lossy compression and channel errors, is a random variable. Thus, it does not suffice to calculate the video distortion for just one realization of the channel. Therefore, we consider the expected value of the distortion, $E[D_{s+c,k}]$. Alternatively, instead of running repeated simulations in order to estimate the expected distortion at the receiver, it is also possible to estimate it using the recursive optimal per-pixel estimate algorithm [229].

The URDC model used in this research to estimate the expected distortion is given by [19, 132]

$$E[D_{s+c,k}] = \alpha \left[\log_{10} \left(\frac{1}{P_b} \right) \right]^{-\beta},\tag{3.12}$$

for node k. The parameters α and β are positive and they are highly dependent on the video content characteristics as well as the source coding rate. Their values are determined in a preprocessing phase by using mean squared error optimization for some $(E[D_{s+c,k}], P_b)$ pairs [19].

With regard to the values that the source coding rates and channel coding rates can assume, it suffices to mention that the channel coding rates, $R_{c,k}$, $k = 1, 2, \ldots, N$, can only take discrete values from a set $\mathbf{R}_c$ [61]. Combining this assumption with the definition of R_k (which is fixed in our problem), this implies that source coding rates, $R_{s,k}$, $k = 1, 2, \ldots, N$, can also take discrete values from a set $\mathbf{R}_s$. Namely

$$R_{c,k} \in \mathbf{R}_c, \qquad R_{s,k} \in \mathbf{R}_s, \qquad k = 1, 2, \ldots, N.$$

Let the index $cb = 1, 2, \ldots, CB$, denote the admissible source coding rate-channel coding rate combinations. Then, $(R_{s,k}, R_{c,k})$ assumes discrete values from a set

$$\mathbf{R}_{s+c} = \left\{ (R_{s,k,1}, R_{c,k,1}), \ldots, (R_{s,k,cb}, R_{c,k,cb}), \ldots, (R_{s,k,CB}, R_{c,k,CB}) \right\}.$$

The cardinality of $\mathbf{R}_{s+c}$ is CB. Evidently, the cardinalities of the sets $\mathbf{R}_s$, $\mathbf{R}_c$, and $\mathbf{R}_{s+c}$ shall be equal. Increasing the cardinality of these sets, results in a significant augmentation of the search space with a consequent impact on the corresponding problem's complexity.

Regarding the power levels of the nodes, we assume that they can take real values within a predetermined continuous range

$$S_k \in \mathbf{S} = [s_{\min}, s_{\max}] \subset \mathbb{R}, \qquad k = 1, 2, \ldots, N.$$

Since the parameters α and β in the URDC model (Eq. (3.12)) are functions of the source coding rate, they are immediately dependent on cb. Furthermore, the aforementioned parameters are also closely related to the motion detected in each video sequence. Higher motion levels detected in a video sequence and/or higher source coding rates correspond to higher values for the parameter α. The free distance of the code, d_{free}, and the information error weight, c_d, in P_b's relation depend on the channel coding rate and, thus, they are also dependent on cb. Viterbi's upper bound of Rel. (3.1) is considered to be tight [194], and thus, it can be used as an approximation of the bit error rate P_b [27]. It is to be noted that taking Rel. (3.1) with equality refers to a worst case analysis.

Substituting P_d (Eq. (3.2)) into Rel. (3.1) (assuming that it holds as equality), and then P_b into Eq. (3.12), $E[D_{s+c,k}]$ becomes

$$E[D_{s+c,k}](R_{s,k}, R_{c,k}, S) = \alpha(cb) \left[\log_{10} \frac{1}{\frac{1}{P} \sum_{d=d_{\text{free}}(cb)}^{\infty} \left(c_d(cb) \frac{1}{2} \operatorname{erfc} \left(\sqrt{d R_{c,k} \left(\frac{S_k/R_k}{\sum_{j=1, j \neq k}^{N} S_j/W_t} \right)} \right) \right)} \right]^{-\beta(cb)},$$

(3.13)

for node k. Evidently, the expected video distortion is a function of the source coding rate, $R_{s,k}$, and the channel coding rate, $R_{c,k}$, for node k, as well as of the power levels, $S = (S_1, S_2, \ldots, S_N)^{\top}$, of all N nodes participating in the network. Therefore, we eventually need to determine the source-channel coding rate combination for node k, and the power levels of all nodes, in order to compute the expected video distortion.

3.5 Optimization Criteria

We assume that the sensor nodes participating in the network image scenes that include various motion levels. This is a common feature for the majority of real-time VSN applications. The optimization criteria that we consider in order to tackle the problem of optimal resource allocation among the nodes of a wireless DS-CDMA VSN and are based on the concepts of minimizing video distortion, maximizing total system utility, as well as finding optimal resource distributions using game-theory are analyzed in the following.

3.5.1 Minimized Average Distortion (MAD)

According to the minimized average distortion criterion, we need to determine the optimal vectors of source coding rates, $R_{\mathrm{s}} = (R_{\mathrm{s},1}, R_{\mathrm{s},2}, \ldots, R_{\mathrm{s},N})^{\top}$, channel coding rates, $R_{\mathrm{c}} = (R_{\mathrm{c},1}, R_{\mathrm{c},2}, \ldots, R_{\mathrm{c},N})^{\top}$, and power levels, $S = (S_1, S_2, \ldots, S_N)^{\top}$, such that the overall *average distortion* $D_{\mathrm{ave}}(R_{\mathrm{s}}, R_{\mathrm{c}}, S)$ of the network is minimized, subject to the constraint of equal target bitrate, R_{target}, for all nodes. This problem can be formally given as follows

$$\min_{R_{\mathrm{s}}, R_{\mathrm{c}}, S} D_{\mathrm{ave}}(R_{\mathrm{s}}, R_{\mathrm{c}}, S), \tag{3.14}$$

$$\text{subject to}\quad R_1 = R_2 = \cdots = R_N = R_{\mathrm{target}},$$

where $D_{\mathrm{ave}}(R_{\mathrm{s}}, R_{\mathrm{c}}, S)$ is defined as

$$D_{\mathrm{ave}}(R_{\mathrm{s}}, R_{\mathrm{c}}, S) = \frac{1}{N} \sum_{k=1}^{N} E[D_{\mathrm{s+c},k}](R_{\mathrm{s},k}, R_{\mathrm{c},k}, S), \tag{3.15}$$

where k is the node's index and N is the total number of nodes in the VSN. Obviously, this criterion does not assert fairness among the nodes. Hence, distortion is allowed to vary significantly from node to node as far as the average distortion is kept to minimal levels.

3.5.2 Minimized Maximum Distortion (MMD)

The minimized maximum distortion criterion requires the determination of the optimal vectors of source coding rates, $R_{\mathrm{s}} = (R_{\mathrm{s},1}, R_{\mathrm{s},2}, \ldots, R_{\mathrm{s},N})^{\top}$, channel coding rates, $R_{\mathrm{c}} = (R_{\mathrm{c},1}, R_{\mathrm{c},2}, \ldots, R_{\mathrm{c},N})^{\top}$, and power levels, $S = (S_1, S_2, \ldots, S_N)^{\top}$, such that the *maximum distortion* $D_{\mathrm{max}}(R_{\mathrm{s}}, R_{\mathrm{c}}, S)$ among all nodes is minimized subject to the constraint of equal target bitrate, R_{target}, for all nodes, i.e.,

$$\min_{R_{\mathrm{s}}, R_{\mathrm{c}}, S} D_{\mathrm{max}}(R_{\mathrm{s}}, R_{\mathrm{c}}, S), \tag{3.16}$$

$$\text{subject to}\quad R_1 = R_2 = \cdots = R_N = R_{\mathrm{target}},$$

where $D_{\mathrm{max}}(R_{\mathrm{s}}, R_{\mathrm{c}}, S)$ is defined as

$$D_{\mathrm{max}}(R_{\mathrm{s}}, R_{\mathrm{c}}, S) = \max_{k \in \{1,2,\ldots,N\}} E[D_{\mathrm{s+c},k}](R_{\mathrm{s},k}, R_{\mathrm{c},k}, S), \tag{3.17}$$

where k denotes the corresponding node. The MMD criterion may also exhibit deviations of the distortion from node to node, but, in contrast to the MAD criterion, it guarantees that all distortions are kept within acceptable ranges.

3.5.3 Maximized Total Utility (MTU)

The maximized total utility criterion needs to determine the optimal vectors of source coding rates, $R_{\mathrm{s}} = (R_{\mathrm{s},1}, R_{\mathrm{s},2}, \ldots, R_{\mathrm{s},N})^{\top}$, channel coding rates, $R_{\mathrm{c}} = (R_{\mathrm{c},1}, R_{\mathrm{c},2}, \ldots, R_{\mathrm{c},N})^{\top}$, and power levels, $S = (S_1, S_2, \ldots, S_N)^{\top}$, such that the *total system utility* $\sum_{k=1}^{N} U_k(R_{\mathrm{s},k}, R_{\mathrm{c},k}, S)$ of all nodes is maximized subject to the constraint of equal target bitrate, R_{target}, for all nodes, i.e.,

$$\max \sum_{k=1}^{K} U_k(R_{\mathrm{s},k}, R_{\mathrm{c},k}, S_k), \tag{3.18}$$

$$\text{subject to} \quad R_1 = R_2 = \cdots = R_N = R_{\mathrm{target}}.$$

The utility function, U_k, constitutes a measure of relative satisfaction for each node k. In our problem, it is defined equivalently to the PSNR

$$U_k = 10 \log_{10} \frac{255^2}{E[D_{\mathrm{s+c},k}]}, \qquad k = 1, 2, \ldots, N, \tag{3.19}$$

and thus, it is measured in decibel (dB). The quantity $E[D_{\mathrm{s+c},k}]$ represents the expected video distortion for each node k, given by Eq. (3.13). Clearly, higher values of the utility function correspond to higher received video qualities.

3.5.4 Nash Bargaining Solution (NBS)

The Nash bargaining solution offers a distribution rule in order to achieve a mutually agreeable, fair and efficient allocation of the nodes' transmission parameters. Specifically, the NBS, denoted as $F(\mathbf{U}, dp)$ for the feasible set $\mathbf{U}$ and the disagreement point dp, shall adhere to a number of axioms as discussed later in this section.

The vector $U = (U_1, U_2, \ldots, U_N)^{\top}$ contains the utilities for all N nodes, given by Eq. (3.19). The *feasible set*, $\mathbf{U}$, encompasses all possible vectors U that result from all possible combinations of the source and channel coding rates as well as the power levels of all nodes, when pure strategies are allowed. (A pure strategy defines a deterministic action of a player). Also, it shall satisfy the following conditions [187]

i) $\mathbf{U} \subset \mathbb{R}^{C}$ is comprehensive, closed and bounded-above.

ii) Free disposal is allowed.

The first condition stipulates that a set $\mathbf{U} \subset \mathbb{R}^{C}$ shall be comprehensive. This means that if X is in $\mathbf{U}$ and $Y \leqslant X$, then Y is in $\mathbf{U}$ as well [21]. Additionally, the same set shall also include all its boundary points (i.e., be closed) and be bounded from above. A set $\mathbf{U}$ is bounded-above, if there exists X such that $Y \leqslant X$ for all $Y \in \mathbf{U}$.

Regarding the second condition, free disposal means that each player is permitted to dispose of utility, if required. The physical meaning in the case of video is that a node is allowed to purposely add noise to its video to degrade the video quality. Obviously, this is an irrational decision and will never be chosen. However, we should not restrict the possible choices of the players regarding the handling of their resources, unless they lead to cases that are impossible to be implemented. Specifically, if $Y \leqslant X$, and X is a feasible point for all nodes, it follows that Y can be achieved by the players also by mutually agreeing to dispose of utility, unilaterally or multilaterally. In the current research, we assume that free disposal is allowed for the feasible set and therefore, this statement clearly implies that the feasible set $\mathbf{U}$ is also comprehensive [187, 21].

Participating in a game, each player expects that he/she will receive at least as high a utility as he/she would get without joining the game (without collaborating). This fact constitutes an incentive for the players to negotiate. The *disagreement point* is the vector of minimum utilities that each player expects by joining the game without cooperating with the other players, and it is what each player will get even in cases of negotiation failure. It is defined as $dp = (dp_1, dp_2, ..., dp_N)^\top$, for all N players (nodes), where dp_k is the minimum acceptable utility for node k, and $dp \in \mathbf{U}$, that is it also belongs to the feasible set. Evidently, the value of the disagreement point has a profound impact on the outcome of the negotiations, even if it never comes to pass.

A resource allocation outcome is *strongly Pareto-optimal* if there cannot be another feasible outcome that is strictly preferred by at least one node, and weakly preferred by the other nodes. In other words, this means that all nodes maintain the payoff they hold and at least one node increases its utility. Instead, a *weakly Pareto-optimal* allocation of resources is strictly preferred by all the nodes of the network, meaning that all of them increase their utilities [21]. All the points that are characterized as Pareto-optimal (strongly and/or weakly) give each node a utility that is greater than or equal to the disagreement point. They are points of the feasible set and consist the *bargaining set*, which is thus, a subset of the feasible set. Fig. 3.1 below illustrates the feasible set, bargaining set, disagreement point as well as the NBS.

With regard to the previous discussion about the axioms that the NBS should satisfy, they are given as follows [21]

1. *Individual Rationality*: $F(\mathbf{U}, dp) \geqslant dp$.

2. *Strong Pareto-Optimality*: $X > F(\mathbf{U}, dp) \Rightarrow X \notin \mathbf{U}$.

3. *Invariance to Affine Transformations*: Given any strictly increasing affine transformation $\tau()$, it holds that $F(\tau(\mathbf{U}), \tau(dp)) = \tau(F(\mathbf{U}, dp))$.

4. *Independence of Irrelevant Alternatives*: If $dp \in \mathbf{Y} \subseteq \mathbf{U}$, then
   $F(\mathbf{U}, dp) \in \mathbf{Y} \Rightarrow F(\mathbf{Y}, dp) = F(\mathbf{U}, dp)$.

The first two axioms imply that the NBS belongs to the bargaining set and the third axiom stipulates that the NBS is unaffected by affine transformation scalings of the
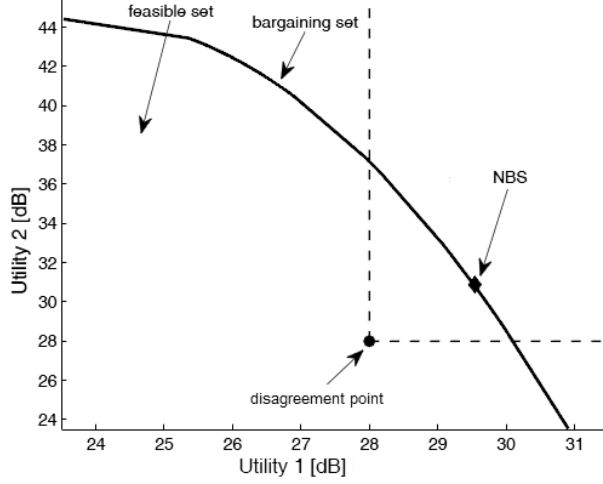
Figure 3.1: Illustration of a feasible set.

utility function. The last axiom states that, if the bargaining solution, $F(\mathbf{U}, dp)$, for the feasible set $\mathbf{U}$ also belongs to a subset $\mathbf{Y}$ of the feasible set, then $F(\mathbf{Y}, dp)$ shall be the same as $F(\mathbf{U}, dp)$, since none of the extra elements of $\mathbf{U}$ were chosen as a solution when they were available. Thus, their unavailability in $\mathbf{Y}$ should be irrelevant.

Provided that the aforementioned conditions are satisfied, the NBS maximizes the Nash product [187, 132, 137, 21]

$$F(\mathbf{U}, dp) = \arg \max_{U \geqslant dp} \prod_{k=1}^{N} (U_k(R_{\mathrm{s},k}, R_{\mathrm{c},k}, S) - dp_k)^{bp_k}, \tag{3.20}$$

subject to the following constraints

(a) $R_k = R_{\mathrm{target}}$ (fixed bitrate).

(b) $S_k \in \mathbf{S} = [s_{\min}, s_{\max}] \subset \mathbb{R}$ (bounded power).

(c) $\sum_{k=1}^{N} bp_k = 1$, $bp_k \geqslant 0$, $k = 1, 2, \ldots, N$.

The parameter bp_k assigned to each factor of the Nash product is called *bargaining power* and declares the advantage of each player in the considered game. Higher bargaining powers imply more advantaged players, and vice versa.

Apart from the bargaining powers, another component that directly affects the Nash product is the disagreement point, as derived from Eq. (3.20). For this reason, we shall pay attention to the appropriate determination of this vector. In [97] we have assumed that the disagreement point corresponds to the vector of utilities that the nodes get if they behave selfishly, without collaborating with each other. Following this reasoning, a node that desires to achieve the best possible received video quality regardless of the intentions of the other nodes, will have to transmit using the maximum power. However, if all nodes adopt this strategy, they will all select to transmit at maximum

power, thereby reaching a Nash equilibrium. This occurs since each node adopts the strategy that is the best response to the strategies followed by the other nodes.

However, such a selection for the disagreement point heavily favors the nodes that capture videos with low motion, which get high utility and have no incentive to collaborate [97]. For this reason, in our research we assume that the disagreement point is imposed by the system designer and expresses the minimum acceptable video quality for each node, for the particular application.

3.5.5 Kalai-Smorodinsky Bargaining Solution (KSBS)

The Kalai-Smorodinsky bargaining solution, $F(\mathbf{U}, dp)$, for the feasible set $\mathbf{U}$, and the disagreement point dp, is the solution that satisfies the following axioms [133, 21, 36]

1. *Individual Rationality*: $F(\mathbf{U}, dp) \geqslant dp$.

2. *Weak Pareto-Optimality*: $X \gg F(\mathbf{U}, dp) \Rightarrow X \notin \mathbf{U}$.

3. *Invariance to Affine Transformations*: Given any strictly increasing affine transformation $\tau()$, it holds that $F(\tau(\mathbf{U}), \tau(dp)) = \tau(F(\mathbf{U}, dp))$.

4. *Strong Individual Monotonicity*: Suppose that $dp \in \mathbf{U}' \subseteq \mathbf{U}$ and U_{MAX} is identical for both $(\mathbf{U}, dp)$ and $(\mathbf{U}', dp)$. Then, if $F(\mathbf{U}', dp)$ is a Pareto-optimal point of $\mathbf{U}$, it holds that $F(\mathbf{U}, dp) = F(\mathbf{U}', dp)$.

The first two axioms state that the bargaining solution lies in the bargaining set. Particularly, the second one specifies that the solution $F(\mathbf{U}, dp)$ is weakly Pareto-optimal, i.e., if there is a point X that is strictly preferred by all nodes, then X does not belong to the feasible set. The third condition stipulates that if the utility function or the disagreement point are scaled by an affine transformation, the bargaining solution remains unaffected. The axiom of *strong individual monotonicity*, described by the fourth axiom, presents the circumstances under which two sets have the same solution.

According to [77], the KSBS is found by taking the maximal element of the feasible set (lying on the bargaining set), on the line connecting the disagreement point and the utopian point. *Utopian point* is called the vector that consists of the maximum achievable utilities that each node can get by participating in the resource allocation game, and is defined as

$$U_{MAX}(\mathbf{U}, dp) = (\max U_1, \max U_2, \ldots, \max U_N)^\top \geqslant dp. \qquad (3.21)$$

The maximum possible utility, $\max U_k$, for node k, has to be greater or at least equal to the utility that node k can get at its disagreement point, dp_k. Since the available resources are usually limited, it is impossible for all nodes to benefit at the same time. Therefore, the utopian point does not belong to the feasible set.

It should be stressed that the KSBS can be applied either to convex or to non-convex feasible sets, satisfying the aforementioned conditions. The only difference lies in the

weak/strong Pareto-optimality axiom, which holds for non-convex/convex feasible sets, respectively. As it was mentioned before, weak Pareto-optimality declares that all nodes prefer the payoff they get with X more than the payoff they get with $F(\mathbf{U}, dp)$. Strong Pareto-optimality means that all nodes like X at least as much as $F(\mathbf{U}, dp)$ and that at least one node likes X strictly more than $F(\mathbf{U}, dp)$. In our study, experimentation proved that the examined feasible sets are all slightly non-convex sets, and due to this, the KSBS has to satisfy the condition of weak Pareto-optimality [187, 133, 36].

Node Clustering

An assumption made in our study is that the N nodes of the network are clustered into C classes of nodes, based on the amount of motion in the detected scenes. The motivation behind node clustering is based on the fact that after such an assumption much less parameters need to be estimated, significantly reducing the problem's complexity as well as the time required for problem solving, a feature particularly important in time-critical applications. In addition, each class of nodes has its own set of parameters α and β (see Eq. (3.13)), since they are affected by the amount of motion of each considered video sequence.

A reasonable question that follows the aforementioned assumption, is what happens in case of a possible change in the motion level of a scene. For example, what happens if the relatively stationary scenes of a forest-monitoring application are disturbed by an unexpected passage of an animal or, in a motorway-surveillance application, the scenes that capture intense traffic succeed scenes with infrequent vehicle passing? In such cases, a new classification of the scenes into motion classes is required, corresponding to a new resource allocation problem that is adjusted to the current state of the observed system.

In the context of node clustering, in the current study we consider two variants for the MTU criterion as well as the NBS criterion, which lie on the way that the *weights* and the bargaining powers are determined, respectively. The MTU criterion in its standard form (alternatively called *unweighted* MTU), as it is described in Section 3.5.3, considers an equal weight, w_{cl}, assigned to the utility U_{cl} of each class of nodes, cl, given by $w_{cl} = 1/C$. However, this definition for the weights leads to the same result given by Eq. (3.18), and thus, for simplicity, we omit the presentation of the weights from this equation (Eq. (3.18)). On the contrary, the Maximized Weighted Total Utility criterion, denoted as w.MTU, makes a different consideration for the weights. Specifically, assuming an equally fair game for all nodes of the network, it follows that each class of nodes cl is assigned a weight equal to $w_{cl} = 1/N_{cl}$, with N_{cl} representing the cardinality of class cl.

Similarly, since the determination of the bargaining powers is crucial for the performance and efficiency of the NBS, in this study, we propose two versions of the NBS, the NNBS and the CNBS. For the NNBS, we assume that each node has the same advantage in the resource allocation game. Practically, given the constraint that the sum of all bargaining powers is equal to 1, and considering an equally fair game for all

45

nodes, it follows that each class of nodes cl is assigned a bargaining power equal to $bp_{cl} = 1/N_{cl}$, with N_{cl} representing the cardinality of class cl. For the CNBS, we assume that each class of nodes is put in a similar position by the rules of the considered game. Therefore, assuming C motion classes, and considering the constraint for the total sum of the bargaining powers, it is implied that $bp_{cl} = 1/C$, for the cl class of nodes.

3.6 Optimization Algorithms

3.6.1 Particle Swarm Optimization (PSO)

Particle swarm optimization uses a population, called *swarm*, of search points, called *particles*, to probe the search space. The particles are randomly initialized (usually uniformly) in the search space. Each particle has three essential features: its *current position* in the search space, a *memory*, which retains the best position it has ever visited, and an adaptable *velocity* (position shift) that iteratively defines its new position. Also, it assumes a *neighborhood* consisting of other particles, i.e., a subset of the swarm, with which it interacts by means of information exchange. The information originating from the particle's own experience as well as the collective experience, are the main sources of influence for its move in the search space.

Let the general minimization problem

$$\min_{x \in \mathbf{X} \subset \mathbb{R}^n} f(x),$$

with $f(x)$ being the objective function. Let the set $I = \{1, 2, \ldots, P\}$ denote the indices of the N particles of the swarm. Then, the swarm can be represented as a set of search points

$$\mathcal{S} = \{x_1, x_2, \ldots, x_P\}.$$

Each particle is an n-dimensional vector

$$x_i = (x_{i1}, x_{i2}, \ldots, x_{in})^\top \in \mathbf{X}, \qquad i \in I,$$

and its velocity is defined as

$$v_i = (v_{i1}, v_{i2}, \ldots, v_{in})^\top, \qquad i \in I.$$

Its best position is also an n-dimensional vector

$$p_i = (p_{i1}, p_{i2}, \ldots, p_{in})^\top \in \mathbf{X}, \qquad i \in I,$$

stored in the memory and iteratively updated as long as the particle moves in $\mathbf{X}$.

The neighborhood, $\mathcal{N}_i$, of the i-th particle can be defined in various ways. A straight-forward approach considers as neighbors the closest particles in the search space. However, this approach has been shown to produce clusters of particles that rapidly

collapse on local minimizers, thereby reducing the (collective) exploration ability of the swarm. An alternative idea is the determination of neighborhoods in abstract spaces instead of the actual search space. An instance that has proved to be very efficient assumes that the particles are ordered on a *ring* based on their indices. In this case, the neighborhoods consist of particles with neighboring indices, having the form

$$\mathcal{N}_i = \{i - r, \ldots, i - 1, i, i + 1, \ldots, i + r\} \subseteq I,$$

where $r \in \left\{1, 2, \ldots, \frac{P}{2}\right\}$ is called the neighborhood's *radius*. The indices are assumed to recycle at the ends, i.e., index 1 follows immediately after index P. Evidently, increasing r results in neighborhoods that approximate the whole swarm. Different neighborhood topologies have been proposed in the literature [91, 185]. The neighborhoods control the information flow among the particles as well as the available information that influences the particles' position shifts at each iteration. Therefore, they can have a tremendous impact on PSO's performance.

Let g_i denote the index of the best particle in the neighborhood $\mathcal{N}_i$ of the i-th particle, i.e.,

$$g_i = \arg \min_{j \in \mathcal{N}_i} f(p_j), \qquad i \in I, \tag{3.22}$$

and let t denote the iteration number. Then, the swarm is updated at each iteration as follows [33]

$$v_{ij}^{(t+1)} = \chi \left[v_{ij}^{(t)} + c_1 \mathcal{R}_1 \left(p_{ij}^{(t)} - x_{ij}^{(t)} \right) + c_2 \mathcal{R}_2 \left(p_{g_i j}^{(t)} - x_{ij}^{(t)} \right) \right], \tag{3.23}$$

$$x_{ij}^{(t+1)} = x_{ij}^{(t)} + v_{ij}^{(t+1)}, \tag{3.24}$$

where $i \in I$; $j = 1, 2, \ldots, n$; and χ is a parameter called the *constriction coefficient*, which can deter the *swarm explosion* effect, i.e., the rapid divergence of the particles due to excessively large velocities [33, 45, 195]. Regarding c_1 and c_2, they are two positive acceleration parameters called the *cognitive* and *social* parameter, respectively. These parameters control the influence of the personal and collective experience (memory) on the particle's move, with equal values promoting a fair tradeoff between them. Finally, $\mathcal{R}_1$ and $\mathcal{R}_2$ are random variables uniformly distributed in the range $[0, 1]$. They introduce stochasticity in PSO and assume a different value for each i and j. Evidently, PSO's update is inherently parallel, since it is performed componentwise.

After updating and evaluating the swarm, memory update takes place in two stages. In the first stage, the personal best position of each particle is updated as follows

$$p_i^{(t+1)} = \begin{cases} x_i^{(t+1)}, & \text{if } f\left(x_i^{(t+1)}\right) < f\left(p_i^{(t)}\right), \\ p_i^{(t)}, & \text{otherwise,} \end{cases} \qquad i \in I.$$

The determination of new best positions is followed, in the second stage, by the update of all indices g_i, $i \in I$, according to Eq. (3.22). This completes a PSO iteration. The procedure is repeated until a stopping criterion is satisfied, such as exceeding a prespecified number of function evaluations or reaching a target function value.

Clerc and Kennedy [33] have extensively studied the stability of PSO. Their analysis offered significant mathematical evidence on its proper parameter settings. Based on their analysis, the parameter values

$$\chi = 0.729, \quad c_1 = 2.05, \quad c_2 = 2.05,$$

have been shown to be a satisfactory starting choice, considered as the default parameter set of the constriction coefficient variant of PSO. Further information and alternative settings can be found in [33, 195].

PSO belongs among the most studied metaheuristics. In addition to [33, 195], theoretical analyses can also be found in [197, 76, 169]. Its theoretical background, well-understood dynamics, as well as its frequently verified efficiency renders PSO a very appealing optimizer.

Tackling Discrete Variables

Although PSO was primarily designed to handle continuous variables, it has been successfully applied also on integer optimization problems [93, 51, 148, 149, 104]. This can be achieved by introducing integer, arithmetic-based operators in PSO. However, in most cases, the resulting PSO variants barely resemble the original PSO dynamics. Alternatively, discrete values can be tackled by solving an extended version of the problem in the continuous space and rounding the candidate solutions to the nearest integers prior to their evaluation with the objective function. The latter procedure has minor effect on the algorithm. Also, it has been shown to work efficiently in various problems, offering motivation for selecting the latter approach in our study.

In the mixed-integer optimization problems of the present research, each particle should normally consist of integer and continuous components, corresponding to the discrete and continuous variables as described earlier. Instead, we also consider the integer parameters to be continuous (retaining their bounds) and apply the presented PSO scheme. However, whenever a particle is evaluated with the objective function, its corresponding components are rounded to the nearest integers as follows

$$x_{ij} = \lfloor x_{ij} + 0.5 \rfloor,$$

where $\lfloor . \rfloor$ is the floor function.

Estimation of Maximum Velocity

A feature usually neglected in PSO implementations is that of *maximum velocity*. Specifically, whenever the velocities are computed by Eq. (3.23), they undergo a magnitude-restriction test as follows

$$v_{ij}^{(t+1)} = \begin{cases} v_j^{\max}, & \text{if} \quad v_{ij}^{(t+1)} > v_j^{\max}, \\ -v_j^{\max}, & \text{if} \quad v_{ij}^{(t+1)} < -v_j^{\max}, \\ v_{ij}^{(t+1)}, & \text{otherwise}, \end{cases} \qquad \forall \, i, j, t,$$

where $v_j^{\max}$ is a predefined positive value, possibly different for each $j = 1, 2 \ldots, n$. Obviously, this procedure restricts the velocity components within the corresponding ranges $[-v_j^{\max}, v_j^{\max}]$, preventing the particles from taking large steps that could lead to wide-range oscillations around the best positions or frequently escaping out of the search space. Naturally, this can have a considerable impact on PSO's convergence speed. We can easily infer that large values of $v_j^{\max}$ are more appropriate for search spaces with wide flat or low-curvature regions, while significantly smaller values may be required in steep functions with large number of minimizers, especially when they are closely concentrated.

Typically, maximum velocity is determined as the maximum absolute distance allowed to be traveled by the particle in a single step at each component direction. For this purpose, it is usually defined as a fraction of the corresponding search space's range in the specific component direction. For example, if the search space is defined as

$$\mathbf{X} = [x_1^{\min}, x_1^{\max}] \times \cdots \times [x_n^{\min}, x_n^{\max}],$$

then the following restriction is commonly used

$$v_j^{\max} = \gamma_j \left(x_j^{\max} - x_j^{\min} \right), \qquad \gamma_j \in (0, 1], \quad j = 1, 2 \ldots, n. \tag{3.25}$$

Available information on the form of the objective function may dictate larger or smaller values of the parameter γ_j. For example, the Lipschitz property [64] can provide useful insight regarding the degree of variation of the objective function in the whole search space. However, in most cases, such information is either unavailable or very laborious to be computed.

In such cases, we can approximately estimate the Lipschitz constant by considering its Modulus of Continuity (MoC), $\delta > 0$, which is locally defined in a subset $\mathbf{B} \subset \mathbf{X}$ of the search space as follows

$$|f(x) - f(y)| \leqslant \delta \|x - y\|, \qquad \forall \, x, y \in \mathbf{B}.$$

Estimating the MoC around solutions obtained in preliminary experiments as well as on randomly selected points in the search space can partially reveal the local behavior of the objective function. In turn, this can lead to a more appropriate selection of the maximum velocity thresholds described above. The estimation can be easily conducted through Monte Carlo sampling within the corresponding region $\mathbf{B}$.

In our preliminary experiments, we have observed that PSO's performance in terms of convergence speed exhibited large deviations per optimization criterion for some cases. Thorough examination of the corresponding landscapes revealed the importance of proper velocity setting. Figure 3.2 refers to the MMD criterion and illustrates the contour plot of the corresponding landscape for a fixed source-channel coding rate combination (the one that corresponds to the best solution). The two axes stand for the real-valued power levels, when $C = 2$ motion classes are considered. Darker lines denote lower objective values. As we can see, the function has extremely steep
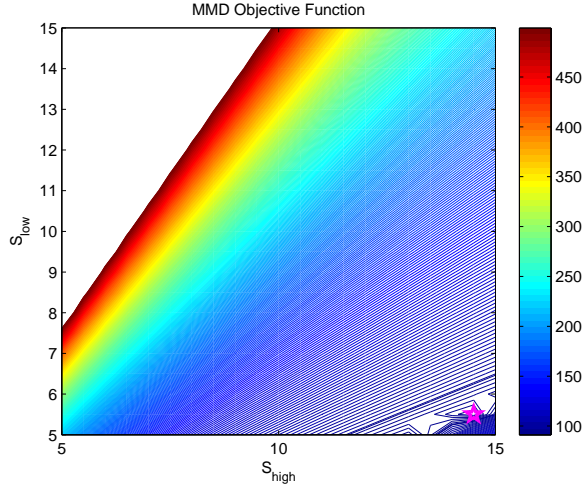
Figure 3.2: Contour plot of the objective function for the MMD criterion.

regions (upper left part of the figure) as well as almost flat regions (middle to lower part). The star mark denotes the globally optimal power level vector (for the specific source-channel coding rate combination), which lies in a small region near the right lower bound of the search space.

Note that the optimization algorithm optimizes the source-channel coding rate combinations as well as the power levels, concurrently. It is easily conceived that the interplay of the discrete variables along with the remarkable changes in slope for the real-valued variables, as well as the discontinuities that may be produced by criteria as the MMD can impose serious difficulties for any optimization algorithm. Thus, we employed the procedure described above to obtain estimations of the maximum velocities for each optimization criterion based on Monte Carlo approximations of the MoC.

Last but not least, it is to be noted that the PSO algorithm is the main optimization solver for all but one of the optimization criteria considered in our research. Specifically, except for the KSBS, which is found by following a geometric approach, all the rest schemes utilize PSO, while its performance is evaluated with that of the AS, IP, and TRR optimization methods. Also, the HPSOAS algorithm applied on the MAD and MMD criteria proves its advantage over all examined optimization methods, including PSO.

3.6.2 Active Set (AS)

Constrained optimization problems are usually tackled by splitting the initial problem into simpler subproblems than can be solved and used as the basis of an iterative process. The active set is an iterative method that is used for solving a sequence of quadratic subproblems, guaranteeing the feasibility of the final solution [113, 123]. The main mechanism is based on the solution of the Karush-Kuhn-Tucker (KKT) equations, which guarantee the optimality for a constrained optimization problem.

50

Let us assume again the minimization problem as declared in Section 3.6.1

$$\min_{x \in \mathbf{X} \subset \mathbb{R}^n} f(x),$$

subject to m constraints (these constraints may be implicitly given as defining relations of the search space $\mathbf{X}$)

$$G_i(x) = 0, \qquad i = 1, \ldots, m_e, \tag{3.26}$$

$$G_i(x) \leq 0, \qquad i = m_e + 1, \ldots, m. \tag{3.27}$$

The vector function $G(x) = (G_1(x), \ldots, G_m(x))^\top$ returns a vector of length m that includes the equality and inequality constraint values at x. The corresponding KKT equations are given by

$$\nabla f(x) + \sum_{i=1}^{m} \lambda_i \, \nabla G_i(x) = 0, \tag{3.28}$$

$$\lambda_i \, G_i(x) = 0, \qquad i = 1, \ldots, m_e, \tag{3.29}$$

$$\lambda_i \geq 0, \qquad i = m_e + 1, \ldots, m. \tag{3.30}$$

Eq. (3.28) depicts the canceling process of the gradients between the objective function $f(x)$ and the active constraints $G_i(x)$ at x, through the use of the Lagrange multipliers λ_i, $i = 1, \ldots, m$. Lagrange multipliers are used in order to balance the deviations in magnitude of the objective function and constraint gradients. Due to the fact that only active constraints are included in the gradients canceling, non-active constraints are assigned $\lambda_i = 0$, as it is stated implicitly by Eqs. (3.29)-(3.30).

Thus, the AS method is based on the solution of the KKT equations and attempts to compute the Lagrange multipliers directly. It searches solutions in the feasible sets and if a minimizer is found during each iteration followed by a decrease in the value of the objective function, the algorithm terminates after a user-defined stopping criterion. Such a criterion can be the maximum iteration number, the maximum number of function evaluations, the function tolerance, the tolerance of the optimal point x etc.

In our problem, we use AS for comparison purposes, and the same is also true for the IP and TRR methods. Specifically, we adopt the robust implementations of the original Matlab$^{\circledR}$ Optimization Toolbox. Further details on all these implementations can be found in [11].

3.6.3 Hybrid Particle Swarm Optimization Active Set (HPSOAS)

Motivated by the benefits of both PSO and AS optimization algorithms, we combine their features introducing a hybrid PSO-AS approach, which is denoted as HPSOAS. This hybrid approach can be categorized as a *memetic algorithm* [126] and employs AS as local optimizer for further improving the findings of PSO.

Algorithm 1 HPSOAS Description

Require: Initialize PSO algorithm.

 1: **loop**
 2:    **if** (not stopping) **then**
 3:        Update swarm and best positions.
 4:        **if** (new overall best position is found) **then**
 5:            Apply AS on the new best position.
 6:            Make AS's solution the new overall best position.
 7:        **end if**
 8:    **end if**
 9: **end loop**

In more detail, when the overall best position of PSO changes, a local search procedure with AS is initiated from this point, in order to further improve it. The procedure is sketched in Alg. 1. In our experiments, although PSO and in many cases also AS are capable of successfully approximating the optimal solution, the HPSOAS scheme is significantly more time-efficient, yet retaining the solutions' quality. Its success lies on the fact that AS is rapidly improving the PSO's best findings, thereby providing better attractors (best positions) for the particles, while at the same time it surmounts the sensitivity of AS on the initial conditions (starting point). Our experimentation adopts this approach as the optimization solver to the MAD and MMD optimization criteria.

3.6.4 Geometric Solution

Let us now describe the procedure that we follow in order to solve the resource allocation problem using the KSBS. We seek the rule that allocates fairly and efficiently the discrete source and channel coding rates, and the continuous power levels among all the nodes of the network. This rule is defined by the KSBS, which is calculated at the CCU. Since the KSBS is found by taking the element of the bargaining set that also lies on the line that connects the disagreement point and the utopian point, we approach the problem of resource allocation under a geometric perspective. The bargaining solutions are derived geometrically, directly from the graphical representation of each considered feasible set.

Figure 3.3 gives an intuition about the feasible set and the KSBS. Specifically, it depicts the feasible set $\mathbf{U}$, when there are $C = 2$ classes of nodes in the network. U_1 declares the utility for the first class of nodes and U_2 for the second class of nodes. In our problem, these quantities represent the corresponding PSNR values. The utopian point U_{MAX} lies outside the feasible set. In the same figure, the diamond represents the KSBS, $F(\mathbf{U}, dp)$, for the feasible set $\mathbf{U}$ and the disagreement point dp.

A feasible point results from a combination of the nodes' transmission parameters. Thus, considering all possible combinations of the nodes' transmission parameters, having assumed a node clustering into $C = 2$ motion classes, the feasible set is formed.
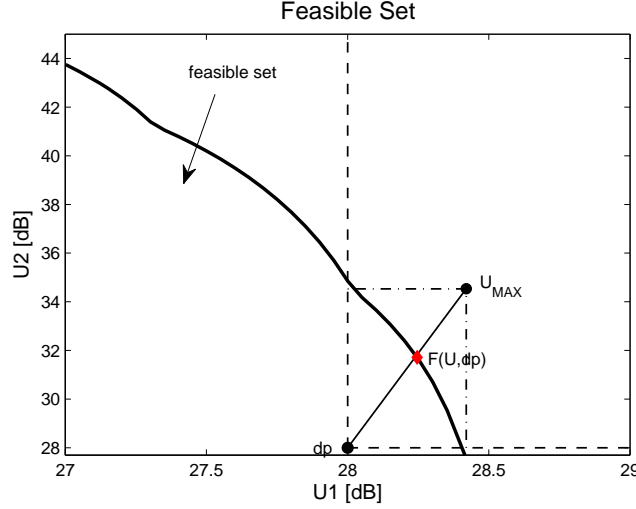
Figure 3.3: A feasible set and the KSBS.

In the following, in order to determine the bargaining set, namely the Pareto-optimal points of the feasible set, we partition the x-axis of the feasible set in small, equal segments. For each segment of the x-axis, we keep the point with the highest value in the y-axis. The set that is formed including the points with the highest values in the y-axis, for each segment of the x-axis, forms the bargaining set. In order to find the equation that describes the bargaining set, a polynomial of second degree is used, since it is a good approximation for this set. Specifically, we have the equation

$$U_2 = \alpha_1 U_1^2 + \alpha_2 U_1 + \alpha_3, \tag{3.31}$$

where the coefficients $\alpha_1, \alpha_2, \alpha_3$ are estimated in a least squares sense for a few (U_1, U_2) pairs.

Continuing, we set the disagreement point at a specific value and compute the vector of the utopian point, $U_{MAX}(\mathbf{U}, dp)$, which corresponds to the maximum achievable utilities for each class of nodes. We connect the disagreement point with the utopian point with a straight line and find the equation of this line. Specifically, we have the equation

$$U_2 = \frac{\max U_2 - dp_2}{\max U_1 - dp_1}(U_1 - dp_1) + dp_2. \tag{3.32}$$

Therefore, having a set of equations, Eqs. (3.31) and (3.32), we solve the system. The point that results from solving the system is the intersection point of the curve and the straight line and corresponds to the Kalai-Smorodinsky bargaining solution that we seek to find. This point belongs to the bargaining set and is unique, as we can see in Fig. 3.3. Therefore, it is a Pareto-optimal point. Concisely, the steps for the calculation of the KSBS are described in Alg. 2.

As it was previously mentioned, each feasible point comes from a combination of the nodes' transmission parameters. In our problem, this corresponds to a power

level value and a combination of source and channel coding rate values, assuming a maximum power constraint and a fixed bitrate constraint. Allowing continuous values for the power levels, we have an infinite number of points in the feasible set. Thus, in order to graphically determine the KSBS, discretization of the power levels is necessary. Therefore, we constrain the power levels to take values within a set of predetermined range.

Algorithm 2 KSBS Calculation
---
1. Determine all feasible points (U_1, U_2).
2. Determine the feasible points (U_1, U_2) that form the bargaining set.
3. Determine the coefficients $\alpha_1, \alpha_2, \alpha_3$ of $U_2 = \alpha_1 U_1^2 + \alpha_2 U_1 + \alpha_3$, in a least square sense for a few (U_1, U_2) feasible points.
4. Determine $dp = (dp_1, dp_2)^\top$.
5. Determine $U_{MAX}(\mathbf{U}, dp) = (\max U_1, \max U_2)^\top$.
6. Connect dp and $U_{MAX}(\mathbf{U}, dp)$ with a straight line.
7. Find $U_2 = \frac{\max U_2 - dp_2}{\max U_1 - dp_1}(U_1 - dp_1) + dp_2$.
8. Solve the system of $U_2 = \alpha_1 U_1^2 + \alpha_2 U_1 + \alpha_3$ and $U_2 = \frac{\max U_2 - dp_2}{\max U_1 - dp_1}(U_1 - dp_1) + dp_2$.
9. $F(\mathbf{U}, dp)$ is the intersection point of the system, corresponding to the KSBS.

Clearly, the smaller the step size, the higher the computational complexity of the problem and vice versa. Due to this fact, in this study we chose the value of 10^{-1} for the step size of the power levels. Extensive experimentation reveals that this assumption has minor and trivial effects on the achieved performance for the nodes, incurring solutions for the PSNR values with differences to the third or fourth decimal digit compared to the PSNR values obtained after assuming a smaller step size i.e., 10^{-3} or 10^{-4}. In our opinion, these utility differences are negligible compared to the great gain of the problem's complexity reduction and clearly, this quality difference cannot be perceived by the human eye.

3.7 Evaluation Metrics

The metrics we apply in our study in order to evaluate the results obtained from the different resource allocation schemes are presented below. Specifically, we use four metrics, each of which investigates fairness under a different point of view, considering different fairness and performance aspects at each time.

1. *Performance to Fairness (PF) Metric*

   This metric captures both relative performance and relative fairness issues. It assumes that the total utility achieved by all motion classes using a specific optimization scheme is higher under one scheme compared to the utility achieved by all other competing schemes. Also, we consider that none of the examined

schemes is simultaneously preferred by all motion classes compared to the other schemes.

Assuming that the criterion that maximizes the unweighted total system utility, namely the MTU, is used as the reference criterion for this metric, we define the performance to fairness metric as [189]

$$PF(MTU, Cons) = \frac{\sum_{cl=1}^{C}(U_{cl}^{MTU}(R_{\mathsf{s},cl}, R_{\mathsf{c},cl}, S) - U_{cl}^{Cons}(R_{\mathsf{s},cl}, R_{\mathsf{c},cl}, S))}{\sum_{cl=1}^{C} \max(0, U_{cl}^{Cons}(R_{\mathsf{s},cl}, R_{\mathsf{c},cl}, S) - U_{cl}^{MTU}(R_{\mathsf{s},cl}, R_{\mathsf{c},cl}, S))}, \tag{3.33}$$

where $Cons$ refers to each considered scheme. The numerator of Eq. (3.33) quantifies the total performance gain of using the MTU over $Cons$, while the denominator quantifies the unfairness of using the MTU over $Cons$. Since the MTU criterion is considered as the reference criterion, the PF values for this scheme are not defined.

2. *Jain's Index* (JI)

   This index measures how close to equal is a resource allocation for all motion classes. Specifically, it is defined as [74]

   $$JI_{Cons}(U) = \frac{\left| \sum_{cl=1}^{C} U_{cl}^{Cons}(R_{\mathsf{s},cl}, R_{\mathsf{c},cl}, S) \right|^2}{C \cdot \sum_{cl=1}^{C}(U_{cl}^{Cons}(R_{\mathsf{s},cl}, R_{\mathsf{c},cl}, S))^2}, \tag{3.34}$$

   where $Cons$ refers to each considered optimization scheme and U corresponds to the vector of utilities of all C motion classes. It takes values between 0 and 1 and this boundedness helps us to understand intuitively the fairness index. The closer the JI value is to unity, the more "equal" the resource allocation is for the motion classes. Therefore, this metric provides a quantitative value to the fairness of the allocation.

3. *Total Utility (TU) Metric*

   This metric examines the total utility that a scheme will bring cumulatively from all motion classes. According to this metric, the most efficient scheme is the one that gathers the highest overall system utility, without examining how close are the utilities achieved by each class of nodes, but only checking the sum of all utilities as a whole. Specifically, this metric computes

   $$TU = \sum_{cl=1}^{C} U_{cl}, \tag{3.35}$$

   for the C motion classes.

4. *Total Power (TP) Metric*

   This metric investigates the major issue of power consumption by each optimization scheme. Each node of the VSN spends an amount of power in order to assure

a reliable video transmission and to maintain the quality of the video reception. On the other hand, it is necessary to keep low amounts of power consumption, since the sensor nodes are battery-operated systems and the prolongation of the battery lifetime is an important issue. Furthermore, in a DS-CDMA system, increased power for a node implies increased interference to the other nodes. Thus, a low power level is required in order to avoid degradation of the video qualities of the other nodes. Therefore, the TP metric calculates the total amount of consumed power, cumulatively for all C motion classes, as given by

$$TP = \sum_{cl=1}^{C} S_{cl},$$ (3.36)

where S_{cl} represents the power level of class cl.

3.8 Experimental Results and Discussion

This section exposes the configuration of the system parameters, experiments and a relative discussion on the obtained results. It is worth mentioning that we have tried various experimental settings so as to check the performance and generalization capabilities of our schemes under different conditions and constraints. Initially, we describe the procedure that we have followed for computing the expected video distortion, which is needed in Eq. (3.13) in order to estimate the parameters α and β of the same equation (Eq. (3.13)).

3.8.1 Estimation of α and β

For given BER, we determine the rate of the packet loss according to the RTP (see Section 3.4). Then, packets are dropped from the video bitstream under investigation. We continue decoding the corrupted video sequence with the H.264/AVC video codec and finally, the expected video distortion is obtained. Figure 3.4 below shows graphically the process of obtaining the distortion for a specific BER for a packet-based video.

Due to the existence of random channel errors in VSNs, the same procedure is repeated for 300 times and the expected video distortion is averaged over all these experiments to offer a more reliable estimation. After the computation of the expected video distortion, the parameters α and β of Eq. (3.13) are determined using least squares optimization from data obtained using a few BERs. Specifically, we consider the BER values of 10^{-7}, 10^{-6}, and 10^{-5}, while it is necessary to have one set of URDCs for each level of motion included in the captures scenes. For this purpose, various video sequences are considered (we present them later in this section), downloaded from [9], so as to simulate the different motion levels. All video sequences used are at Quarter Common Intermediate Format (QCIF) resolution (176×144 pixels), consisting of 300
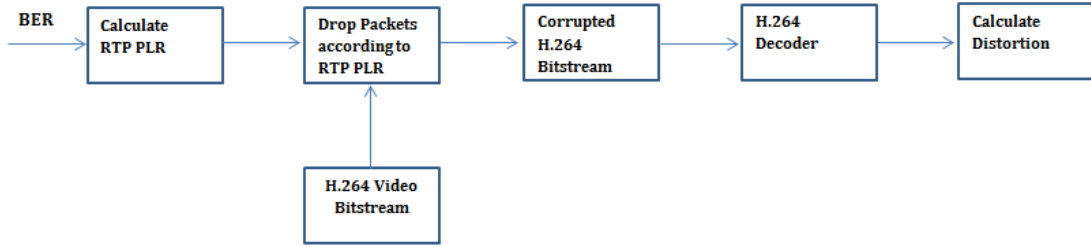
Figure 3.4: Obtaining distortion for a specific BER [19].

frames. The characteristics for all examined video sequences are obtained at a rate of 15 frames per second.

3.8.2 Parameters Configuration

All our experiments have been obtained using simulations, conducted on an Intel® Core™ 2 Quad CPU @ 2.50 GHz with 4.00 GB RAM, using the Matlab environment. In all the experiments, we have employed the BPSK modulation scheme, while RCPC codes with mother code rate of $1/4$ have been used for channel decoding [61]. Also, the considered target bitrate constraints have been equal to 96 kbps and 144 kbps. The power levels could take continuous values from a set $\mathbf{S} = [5.0, 15.0]$ W. The source-channel coding rates could take various pair values, given a target bitrate constraint at each time. More information about this issue is given in the following of this section.

It is worth referring that the video quality for each motion class cl has been estimated by using the PSNR, which is measured in decibel (dB), given by

$$PSNR_{cl} = 10 \, \log_{10} \left(\frac{255^2}{E[D_{\text{s+c},cl}]} \right), \tag{3.37}$$

with $E[D_{\text{s+c},cl}]$ representing the expected video distortion due to source and channel coding for the cl class of nodes given by Eq. (3.13).

For the PSO, the default parameter set as defined in Section 3.6.1 has been selected along with a ring topology of radius $r = 1$. Moreover, since PSO is a stochastic algorithm, its performance has been evaluated over 30 independent experiments. As it is clarified later in this section, some of our experiments have been conducted under the presence of thermal and background noise, while in some other experiments this noise has been considered negligible compared to the interference. In this latter case, PSO algorithm is able to detect a number of optimal solutions for the power levels of all motion classes, all of which can attain the optimum value for the Nash product. Indeed, from Eq. (3.8), it follows that the multiplication of all power levels with the

same constant leaves the ratio E_k/I_0 unaffected. However, the source-channel coding rate combinations are unique. On the contrary, when AWGN is considered, PSO reports unique solutions for all transmission parameters. Moreover, talking about the complexity of PSO, we should refer that taking into account that the number of particles (P) and the number of iterations $(Iter)$ that the algorithm is executed both depend on the number of motion classes (C), the complexity of PSO is given by $O(C \cdot P \cdot Iter)$.

Regarding the number of nodes constituting the VSN as well as the clusters they form, we have considered two different scenarios.

(a) N $= 100$ nodes assigned to C $= 2$ motion classes.
A high-motion class consisting of the nodes that detect high levels of motion and a low-motion class consisting of the nodes that detect low levels of motion have been formed in this case. The "Foreman" video sequence has been used to represent the class of nodes that detect high motion levels, while the "Akiyo" video sequence has been used to represent the class of nodes that capture more stationary fields. The nodes have been assigned to the two motion classes in different proportions, called *node distributions*, denoted as "$N_{\text{high}} - N_{\text{low}}$", where

$$N_{\text{high}}, N_{\text{low}} \in \{10, 30, 50, 70, 90\}, \quad N_{\text{high}} + N_{\text{low}} = N = 100,$$

meaning that the corresponding classes consist of a number of N_{high} nodes capturing high-motion scenes and N_{low} nodes capturing low-motion scenes, respectively. All considered node distributions are summarized bellow:

- **Case 1:** $N_{\textbf{high}} = 90, N_{\textbf{low}} = 10.$

- **Case 2:** $N_{\textbf{high}} = 70, N_{\textbf{low}} = 30.$

- **Case 3:** $N_{\textbf{high}} = 50, N_{\textbf{low}} = 50.$

- **Case 4:** $N_{\textbf{high}} = 30, N_{\textbf{low}} = 70.$

- **Case 5:** $N_{\textbf{high}} = 10, N_{\textbf{low}} = 90.$

Given the bitrate constraints of $R_{\text{target}} = 96$ kbps and $R_{\text{target}} 144$ kbps, the following source-channel coding rate combinations, cb_{cl}, for each motion class $cl \in \{\text{high}, \text{low}\}$ have been considered:

(i) $R_{\text{target}} = 96$ kbps that results in

$$\mathbf{R}_{\text{s+c}} = \left\{(32, 1/3), (48, 1/2), (64, 2/3)\right\}$$

$$cb_{cl} = 1 \longrightarrow (32, 1/3)$$
$$cb_{cl} = 2 \longrightarrow (48, 1/2)$$
$$cb_{cl} = 3 \longrightarrow (64, 2/3).$$

(ii) $R_{\text{target}} = 144$ kbps that results in

$$\mathbf{R}_{\text{s+c}} = \Big\{ (48, 1/3), (72, 1/2), (96, 2/3) \Big\}$$

$$cb_{cl} = 1 \longrightarrow (48, 1/3)$$
$$cb_{cl} = 2 \longrightarrow (72, 1/2)$$
$$cb_{cl} = 3 \longrightarrow (96, 2/3).$$

For the bandwidth W_t two different values have been examined, namely $W_t = 20$ MHz and $W_t = 15$ MHz.

Regarding PSO, a swarm of $P = 40$ particles has been employed. In the specific formulation, each particle x_i has been 4-dimensional, defined as

$$x_i = \big(S_{\text{high}}, S_{\text{low}}, cb_{\text{high}}, cb_{\text{low}} \big)^\top, \qquad i = 1, 2, \ldots, 40,$$

containing the power level and the source-channel coding rate combination for each motion class, denoted as "high" and "low", respectively. Furthermore, the discrete parameters, i.e., source and channel coding rate combinations, have been represented in the particle with continuous values within the range $\mathbf{R} = [0.6, 3.4]$. However, as explained in Section 3.6.1, they have been rounded to the nearest integer for the particle's evaluation.

Besides that, the maximum velocities for the particles have been set based on the MoC estimation procedure described in Section 3.6.1. In our problem, the corresponding values of γ in Eq. (3.25) for the 4 component directions have been set as follows

$$\gamma^{\text{MAD}} = (0.1, 0.1, 0.03, 0.03)^\top, \qquad \gamma^{\text{MMD}} = (1, 1, 1, 1)^\top.$$

For each problem instance (out of 30 in total), PSO has been executed for a maximum number of $Iter = 1000$ iterations and the best solution has been recorded.

Regarding the AS method [60, 166], it has also been applied to each problem instance 30 times, for different initial conditions. However, AS has been capable of providing only the power levels, since it works only on continuous search spaces. For the discrete source-channel coding rate combinations, we have used exhaustive search among all possible pair values, i.e., 3 admissible values for each of the two motion classes, resulting in $3 \times 3 = 9$ cases. Thus, a single application of AS required 9 optimization runs, one for each discrete combination, and the final solution has been selected as the best one among the obtained 9 solutions.

Finally, identical experiments have been conducted using the proposed hybrid algorithm HPSOAS. The parameter setting of HPSOAS has been the same with that of PSO. HPSOAS has exhibited similar performance with PSO in terms of solutions quality for each problem instance. Note that all algorithms (PSO, AS, HPSOAS) have been

equipped with exactly the same total computational budget in terms of function evaluations, namely 40000 function evaluations, in order to achieve fair comparisons among them.

(b) $N = 102$ **nodes assigned to** $C = 6$ **motion classes.**

Six motion classes have been formed in this case, describing a different amount of motion in the captured scenes. For this purpose, six different video sequences have been used so as to represent each motion class. For the node distributions we have considered the following cases:

- **Case 1:** $N_\mathbf{a} = N_\mathbf{md} = N_\mathbf{s} = N_\mathbf{h} = N_\mathbf{f} = N_\mathbf{c} = 17.$

- **Case 2:** $N_\mathbf{a} = N_\mathbf{md} = 25, N_\mathbf{s} = N_\mathbf{h} = N_\mathbf{f} = N_\mathbf{c} = 13.$

- **Case 3:** $N_\mathbf{a} = N_\mathbf{md} = N_\mathbf{f} = N_\mathbf{c} = 13, N_\mathbf{s} = N_\mathbf{h} = 25.$

- **Case 4:** $N_\mathbf{a} = N_\mathbf{md} = N_\mathbf{s} = N_\mathbf{h} = 13, N_\mathbf{f} = N_\mathbf{c} = 25.$

N_a denotes the cardinality of the class that is represented by the "Akiyo" video sequence, while $N_\mathrm{md}, N_\mathrm{s}, N_\mathrm{h}, N_\mathrm{f}$ and N_c denote the cardinality of the class that is represented by the "Mother&Daughter", "Salesman", "Hall", "Foreman" and "Coastguard" video sequences, respectively. In Case 1, all classes include exactly the same number of nodes; in Case 2 more nodes describe low amounts of motion; in Case 3 more nodes describe medium amounts of motion and in Case 4 more nodes describe high amounts of motion. Figure 3.5 below depicts the video sequences used as well as the amount of motion in each of them.
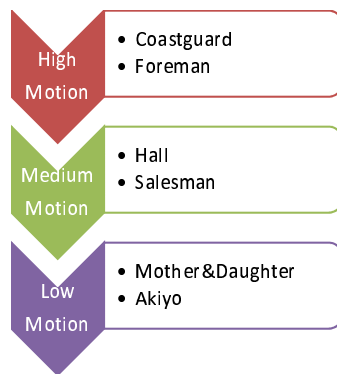
Figure 3.5: The used video sequences and the amount of motion described by each of them.

From the "Salesman" video sequence we have kept only the first 300 frames in order to have the same length for all video sequences, that is 300 frames.

Taking into account the bitrate constraints of $R_{\text{target}} = 96$ kbps and $R_{\text{target}} = 144$ kbps, it follows that the source-channel coding rate combinations can take the following discrete values:

(i) $R_{\text{target}} = 96$ kbps that results in

$$\mathbf{R}_{\text{s+c}} = \left\{ (32, 1/3), (38.4, 4/10), (48, 1/2), (64, 2/3), (76.8, 4/5) \right\}$$

$$
\begin{aligned}
cb_{cl} = 1 &\longrightarrow (32, 1/3) \\
cb_{cl} = 2 &\longrightarrow (38.4, 4/10) \\
cb_{cl} = 3 &\longrightarrow (48, 1/2) \\
cb_{cl} = 4 &\longrightarrow (64, 2/3) \\
cb_{cl} = 5 &\longrightarrow (76.8, 4/5).
\end{aligned}
$$

(ii) $R_{\text{target}} = 144$ kbps that results in

$$\mathbf{R}_{\text{s+c}} = \left\{ (48, 1/3), (57.6, 4/10), (72, 1/2), (96, 2/3), (115.2, 4/5) \right\}$$

$$
\begin{aligned}
cb_{cl} = 1 &\longrightarrow (48, 1/3) \\
cb_{cl} = 2 &\longrightarrow (57.6, 4/10) \\
cb_{cl} = 3 &\longrightarrow (72, 1/2) \\
cb_{cl} = 4 &\longrightarrow (96, 2/3) \\
cb_{cl} = 5 &\longrightarrow (115.2, 4/5).
\end{aligned}
$$

For the bandwidth W_t we have examined the following values per bitrate constraint:

(a) $R_{\text{target}} = 96$ kbps $\longrightarrow W_t = 20\,\text{MHz}, W_t = 15\,\text{MHz}$.

(b) $R_{\text{target}} = 144$ kbps $\longrightarrow W_t = 30\,\text{MHz}, W_t = 22.5\,\text{MHz}$.

Since $R_{\text{target}} = 144$ kbps is 1.5 times the $R_{\text{target}} = 96$ kbps, the same reasoning has been followed for the bandwidth values assignment. Specifically, $W_t = 30$ MHz is 1.5 times the $W_t = 20$ MHz and $W_t = 22.5$ MHz is 1.5 times the $W_t = 15$ MHz. This corresponds to keeping the spreading code length the same for all cases.

In the specific formulation, each particle x_i has been 12-dimensional, defined as

$$x_i = \left(S_{\text{a}},\ S_{\text{md}},\ S_{\text{s}},\ S_{\text{h}},\ S_{\text{f}},\ S_{\text{c}},\ cb_{\text{a}},\ cb_{\text{md}},\ cb_{\text{s}},\ cb_{\text{h}},\ cb_{\text{f}},\ cb_{\text{c}} \right)^{\top}, \qquad i = 1, 2, \ldots, 100,$$

containing the power levels and the source-channel coding rate combinations for all motion classes. The discrete components of the particle, i.e., the source and channel coding rate combinations, cb_{a}, cb_{md}, cb_{s}, cb_{h}, cb_{f}, cb_{c}, have assumed continuous values within the range $\mathbf{R} = [0.6, 5.4]$. However, as explained in Section 3.6.1, they have been rounded to the nearest integer for the particle's evaluation. At each of the 30 experiments, PSO has been executed for $Iter = 700$ iterations, which correspond to 70000 function evaluations, given that $P = 100$ particles, and the best solution has been recorded.

3.8.3 Presentation and Analysis of Results

A large number of experiments have been conducted, which can be coarsely grouped into four sets. Each one of these sets is meticulously presented in the following of this section.

1. **First Set of Experiments**

In this first group of experiments, we have been mainly interested in evaluating MAD and MMD performance under the system setup as described in case (a) of Section 3.8.2, taking into account both the cases of considering and neglecting thermal and background noise, namely $N_0 = 0$ W/Hz, and $N_0 = 10^{-7}$ W/Hz, respectively. In addition, in the same set of experiments, we have examined the performance of the HPSOAS algorithm as well as that of the AS, in terms of efficiency in finding the optimal solution as well as execution time.

For each problem instance, PSO and HPSOAS converge on the same solutions in all 30 experiments. Specifically, the obtained values of the objective functions of the MAD and MMD criteria are identical up to $15 - 20$ decimal digits, and therefore, they are considered to be essentially identical. However, this is not the case for AS, which produces inferior solutions in some cases due to its dependency on the initial conditions and the peculiarities of the objective function landscape. Nevertheless, in the rest problem instances AS achieves the same solutions as PSO and HPSOAS.

PSO detects a multitude of solutions for the power levels of both motion classes, all of which have the same ratio $S_{\text{high}}/S_{\text{low}}$, up to $4 - 5$ decimal digits. In contrast to the power levels, the corresponding optimal source and channel coding rate combinations are unique in all solutions provided by PSO. Moreover, when thermal and background noise is added, PSO reports unique solutions for all transmission parameters.

Tables 3.1-3.4 report the obtained direction components of the solution vectors (transmission parameters) that correspond to the best objective values of both MAD and MMD, for the case where thermal and background noise equals $N_0 = 0$ W/Hz. Tables 3.5-3.8 report the corresponding results under the assumption that thermal and background noise is equal to $N_0 = 10^{-7}$ W/Hz. The corresponding tables per combination of bitrate and bandwidth are summarized below:

(a) Bitrate 96 kbps and bandwidth 20 MHz: Tables 3.1 and 3.5.

(b) Bitrate 96 kbps and bandwidth 15 MHz: Tables 3.2 and 3.6.

(c) Bitrate 144 kbps and bandwidth 20 MHz: Tables 3.3 and 3.7.

(d) Bitrate 144 kbps and bandwidth 15 MHz: Tables 3.4 and 3.8.

More specifically, the tables report the obtained source-channel coding rate, the power level and the PSNR of the high-motion class, which are represented as $(R_{\text{s,high}}, R_{\text{c,high}})$,

Table 3.1: $R_{\text{target}} = 96\,\text{kbps}$, $W_t = 20\,\text{MHz}$, $N_0 = 0\,\text{W/Hz}$.

			MAD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10-90$	$(64, 2/3)$	10.7080	$(64, 2/3)$	5.9845	32.9787	36.7642
$30-70$	$(64, 2/3)$	8.6240	$(64, 2/3)$	5.0000	31.3844	35.1131
$50-50$	$(64, 2/3)$	15.0000	$(32, 1/3)$	7.0428	30.9419	32.8537
$70-30$	$(48, 1/2)$	15.0000	$(32, 1/3)$	7.6070	29.2296	32.2582
$90-10$	$(48, 1/2)$	9.8451	$(32, 1/3)$	5.0000	28.2705	31.2943
			MMD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10-90$	$(64, 2/3)$	13.3488	$(64, 2/3)$	5.0000	35.7218	35.7218
$30-70$	$(64, 2/3)$	12.6814	$(32, 1/3)$	5.0000	33.4049	33.4049
$50-50$	$(64, 2/3)$	13.0847	$(32, 1/3)$	5.0000	31.6114	31.6114
$70-30$	$(64, 2/3)$	13.4344	$(32, 1/3)$	5.0000	29.7737	29.7737
$90-10$	$(48, 1/2)$	15.0000	$(32, 1/3)$	5.7234	28.3919	28.3919

S_{high} and $PSNR_{\text{high}}$, respectively, as well as the same quantities for the low-motion class of nodes, denoted as $(R_{\text{s,low}}, R_{\text{c,low}})$, S_{low}, and $PSNR_{\text{low}}$, respectively. The same tables present the results for both optimization criteria and different node distributions. Since in this case we are interested in the optimal power level ratios, rather than the determination of the specific values of S_{high} and S_{low}, we cite indicative S_{high} and S_{low} values that correspond to the best objective value.

A close inspection of the results, demonstrates that the MAD criterion works favorably for the low-motion class of nodes, equipping it with better image quality than the high-motion class. Concerning the MMD criterion, it is rather unbiased, offering identical PSNR values to both motion classes. These remarks are derived from all combinations of bitrate-bandwidth considered in the specific set of experiments.

Moreover, the MMD criterion assigns higher PSNR values to the high-motion class of nodes than the MAD criterion in the corresponding cases. Indeed, the PSNR differences between the two criteria is inversely proportional to the cardinality of the high-motion class of nodes. Thus, the MMD criterion can be considered as the most appropriate choice in cases where we are interested in the amelioration of the high-motion scenes rather than improving the quality of the low-motion scenes. Surveillance applications are typical examples of such cases. On the contrary, the MAD criterion appears to be more suitable in cases where high video quality of low-motion scenes is desirable.

Considering the noisy case, in general, no significant changes in PSNR values are observed after the addition of noise, except for a marginal reduction of no more than 0.01 dB, which has imperceptible impact on video quality. Similarly to the noiseless case, the high-motion class of nodes requires more power than the low-motion class. More specifically, the power levels of the high-motion class are not only higher than those of the low-motion class, but also they actually take their maximum possible values.

Special attention shall be paid to the case of distributing 10 nodes in the high-

Table 3.2: $R_{\text{target}} = 96\,\text{kbps}$, $W_t = 15\,\text{MHz}$, $N_0 = 0\,\text{W/Hz}$.

			MAD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10-90$	$(64, 2/3)$	10.5000	$(32, 1/3)$	5.0000	31.4521	33.3203
$30-70$	$(48, 1/2)$	9.8486	$(32, 1/3)$	5.0000	29.0488	32.0672
$50-50$	$(48, 1/2)$	9.8262	$(32, 1/3)$	5.0000	27.7763	30.7930
$70-30$	$(32, 1/3)$	11.2833	$(32, 1/3)$	7.4632	26.7596	31.6151
$90-10$	$(32, 1/3)$	15.0000	$(32, 1/3)$	10.1943	26.4203	31.1774

			MMD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10-90$	$(64, 2/3)$	12.7296	$(32, 1/3)$	5.0000	33.0047	33.0047
$30-70$	$(64, 2/3)$	13.2312	$(32, 1/3)$	5.0000	30.7191	30.7191
$50-50$	$(48, 1/2)$	13.1569	$(32, 1/3)$	5.0000	28.5991	28.5991
$70-30$	$(32, 1/3)$	12.3831	$(32, 1/3)$	5.0213	27.1521	27.1521
$90-10$	$(32, 1/3)$	15.0000	$(32, 1/3)$	6.7090	26.5321	26.5321

Table 3.3: $R_{\text{target}} = 144\,\text{kbps}$, $W_t = 20\,\text{MHz}$, $N_0 = 0\,\text{W/Hz}$.

			MAD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10-90$	$(72, 1/2)$	15.0000	$(48, 1/3)$	7.7859	30.3384	32.7780
$30-70$	$(48, 1/3)$	15.0000	$(48, 1/3)$	9.3816	28.0307	31.8708
$50-50$	$(48, 1/3)$	7.7427	$(48, 1/3)$	5.0000	27.1339	30.9034
$70-30$	$(48, 1/3)$	15.0000	$(48, 1/3)$	9.9501	26.3745	30.0843
$90-10$	$(48, 1/3)$	15.0000	$(48, 1/3)$	10.1822	25.7146	29.3731

			MMD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10-90$	$(72, 1/2)$	15.0000	$(48, 1/3)$	6.3505	32.3213	32.3213
$30-70$	$(48, 1/3)$	11.6678	$(48, 1/3)$	5.0000	29.6328	29.6328
$50-50$	$(48, 1/3)$	10.5707	$(48, 1/3)$	5.0000	28.0449	28.0449
$70-30$	$(48, 1/3)$	9.8198	$(48, 1/3)$	5.0000	26.8317	26.8317
$90-10$	$(48, 1/3)$	9.2654	$(48, 1/3)$	5.0000	25.8461	25.8461

motion class and 90 nodes in the low-motion class, which is reported on Table 3.7, for the MAD criterion. Despite the addition of noise, an increase of PSNR is observed for the low-motion class compared to the corresponding case in Table 3.3, for the same criterion. This is attributed to the fact that under the influence of noise, the power level of the low-motion class is increased. Thus, the ratio E_k/I_0 also increases as follows from Eq. (3.8).

Additional information regarding the performance of the proposed schemes is graphically illustrated in figures. Specifically, Fig. 3.6(a) depicts the differences of the received PSNR between the MAD and MMD criteria for both the high- and low-motion class of nodes, for all node distributions, and refers to the case of $R_{\text{target}} = 96$ kbps, $W_t = 20$ MHz, and $N_0 = 0$ W/Hz. The last column of the same figure shows the accumulated PSNR difference between MAD and MMD, also for all node distributions. This figure manifests that, cumulatively for all node distributions, the decrease in PSNR achieved

Table 3.4: $R_{\text{target}} = 144\,\text{kbps}$, $W_t = 15\,\text{MHz}$, $N_0 = 0\,\text{W/Hz}$.

			MAD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10 - 90$	$(48, 1/3)$	15.0000	$(48, 1/3)$	10.0856	26.0091	29.6842
$30 - 70$	$(48, 1/3)$	15.0000	$(48, 1/3)$	10.4477	24.9842	28.5810
$50 - 50$	$(48, 1/3)$	15.0000	$(48, 1/3)$	10.7484	24.1375	27.6704
$70 - 30$	$(48, 1/3)$	15.0000	$(48, 1/3)$	11.0045	23.4146	26.8932
$90 - 10$	$(48, 1/3)$	15.0000	$(48, 1/3)$	11.2265	22.7829	26.2145
			MMD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10 - 90$	$(48, 1/3)$	12.3251	$(48, 1/3)$	5.6041	28.7266	28.7266
$30 - 70$	$(48, 1/3)$	9.6409	$(48, 1/3)$	5.0000	26.5538	26.5538
$50 - 50$	$(48, 1/3)$	15.0000	$(48, 1/3)$	8.4876	25.0406	25.0406
$70 - 30$	$(48, 1/3)$	15.0000	$(48, 1/3)$	9.0465	23.8711	23.8711
$90 - 10$	$(48, 1/3)$	11.0000	$(48, 1/3)$	6.9700	22.9148	22.9148

Table 3.5: $R_{\text{target}} = 96\,\text{kbps}$, $W_t = 20\,\text{MHz}$, $N_0 = 10^{-7}\,\text{W/Hz}$.

			MAD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10 - 90$	$(64, 2/3)$	15.0000	$(64, 2/3)$	8.4148	32.9280	36.7442
$30 - 70$	$(64, 2/3)$	15.0000	$(64, 2/3)$	8.7092	31.3549	35.0942
$50 - 50$	$(64, 2/3)$	15.0000	$(32, 1/3)$	7.0445	30.9207	32.8401
$70 - 30$	$(48, 1/2)$	15.0000	$(32, 1/3)$	7.6110	29.2156	32.2485
$90 - 10$	$(48, 1/2)$	15.0000	$(32, 1/3)$	7.6210	28.2575	31.2847
			MMD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10 - 90$	$(64, 2/3)$	15.0000	$(64, 2/3)$	5.6298	35.6857	35.6857
$30 - 70$	$(64, 2/3)$	15.0000	$(32, 1/3)$	5.9121	33.3853	33.3853
$50 - 50$	$(64, 2/3)$	15.0000	$(32, 1/3)$	5.7300	31.5918	31.5918
$70 - 30$	$(64, 2/3)$	15.0000	$(32, 1/3)$	5.5814	29.7539	29.7539
$90 - 10$	$(48, 1/2)$	15.0000	$(32, 1/3)$	5.7257	28.3789	28.3789

by the MMD criterion for the low-motion class of nodes is considerably higher than the corresponding gain for the high-motion class of nodes. For the case of a bitrate of $96\,\text{kbps}$ and a bandwidth of $20\,\text{MHz}$, the MMD increases the total PSNR for all members of the high-motion class by $6.0986\,\text{dB}$, while the total PSNR for all members of the low-motion class decreases by $9.3798\,\text{dB}$. Therefore, despite the fact that the MMD offers equal PSNR values to both motion classes, it is proved to be less fair than it was initially perceived, since it disfavors the low-motion class of nodes.

An additional piece of information is provided by Fig. 3.6(b) that depicts the optimal ratios $S_{\text{high}}/S_{\text{low}}$ for the two optimization criteria when the number of nodes imaging high levels of motion increases, while that of the nodes imaging low levels of motion decreases. The specific figure refers to the case of $R_{\text{target}} = 96$ kbps, $W_t = 20$ MHz, and $N_0 = 0$ W/Hz and offers strong evidence that the MAD criterion requires much less power than the MMD. Spending less power for data transmission means that a stronger

Table 3.6: $R_{\text{target}} = 96\,\text{kbps}$, $W_t = 15\,\text{MHz}$, $N_0 = 10^{-7}\,\text{W/Hz}$.

			MAD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10 - 90$	$(64, 2/3)$	15.0000	$(32, 1/3)$	7.1567	31.4147	33.3082
$30 - 70$	$(48, 1/2)$	15.0000	$(32, 1/3)$	7.6223	29.0315	32.0576
$50 - 50$	$(48, 1/2)$	15.0000	$(32, 1/3)$	7.6366	27.7619	30.7834
$70 - 30$	$(32, 1/3)$	15.0000	$(32, 1/3)$	9.9299	26.7537	31.6109
$90 - 10$	$(32, 1/3)$	15.0000	$(32, 1/3)$	10.2012	26.4153	31.1736
			MMD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10 - 90$	$(64, 2/3)$	15.0000	$(32, 1/3)$	5.8898	32.9863	32.9863
$30 - 70$	$(64, 2/3)$	15.0000	$(32, 1/3)$	5.6668	30.7003	30.7003
$50 - 50$	$(48, 1/2)$	15.0000	$(32, 1/3)$	5.7028	28.5851	28.5851
$70 - 30$	$(32, 1/3)$	15.0000	$(32, 1/3)$	6.0882	27.1463	27.1463
$90 - 10$	$(32, 1/3)$	15.0000	$(32, 1/3)$	6.7141	26.5271	26.5271

Table 3.7: $R_{\text{target}} = 144\,\text{kbps}$, $W_t = 20\,\text{MHz}$, $N_0 = 10^{-7}\,\text{W/Hz}$.

			MAD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10 - 90$	$(48, 1/3)$	15.0000	$(48, 1/3)$	9.0621	29.0841	33.0421
$30 - 70$	$(48, 1/3)$	15.0000	$(48, 1/3)$	9.3985	28.0076	31.8577
$50 - 50$	$(48, 1/3)$	15.0000	$(48, 1/3)$	9.6979	27.1164	30.8916
$70 - 30$	$(48, 1/3)$	15.0000	$(48, 1/3)$	9.9590	26.3597	30.0736
$90 - 10$	$(48, 1/3)$	15.0000	$(48, 1/3)$	10.1896	25.7016	29.3630
			MMD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10 - 90$	$(72, 1/2)$	15.0000	$(48, 1/3)$	6.3510	32.2887	32.2887
$30 - 70$	$(48, 1/3)$	15.0000	$(48, 1/3)$	6.4367	29.6111	29.6111
$50 - 50$	$(48, 1/3)$	15.0000	$(48, 1/3)$	7.1026	28.0276	28.0276
$70 - 30$	$(48, 1/3)$	15.0000	$(48, 1/3)$	7.6444	26.8170	26.8170
$90 - 10$	$(48, 1/3)$	15.0000	$(48, 1/3)$	8.1007	25.8331	25.8331

channel coding is used, capable of correcting a higher number of channel errors. From Eq. (3.7), it follows that the channel coding rate is inversely proportional to the data compression rate, highlighting the importance of considering the special characteristics of the video sequence (high- against low-motion), when determining the optimal power levels.

Moreover, the power received by the CCU from the low-motion class of nodes is always less than that of the high-motion class for both optimization criteria, as it is confirmed from the tables for all examined bitrate-bandwidth combinations. Hence, since the nodes that image high levels of motion need higher power levels than those of the low-motion class, it is reasonable that the ratio E_k/I_0 of Eq. (3.8) exhibits further decrease when the cardinality of the high-motion class is larger than that of the low-motion one. Naturally, this incurs a reduction to the PSNR values for both motion classes.
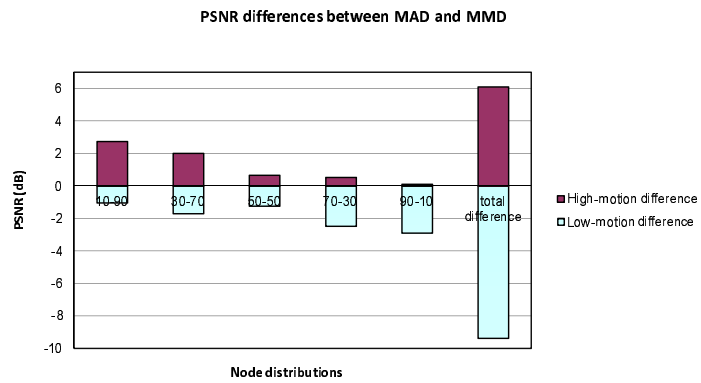
(a)
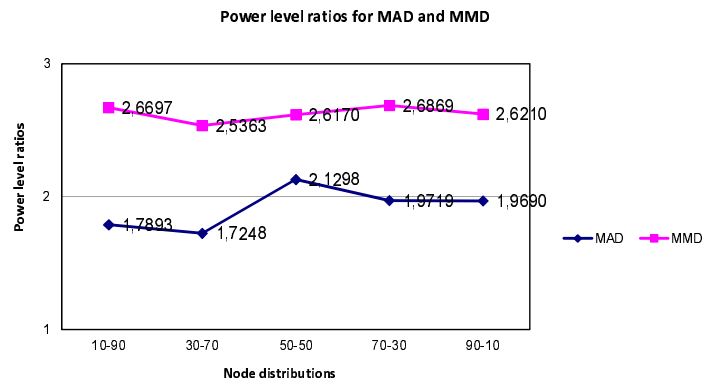
(b)

Figure 3.6: PSNR differences (a) and power level ratios (b) for the MAD and MMD criteria, for various node distributions, when $R_{\text{target}} = 96$ kbps, $W_t = 20$ MHz, and $N_0 = 0$ W/Hz.

Table 3.8: $R_{\text{target}} = 144\,\text{kbps}$, $W_t = 15\,\text{MHz}$, $N_0 = 10^{-7}$ W/Hz.

			MAD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10-90$	$(48, 1/3)$	15.0000	$(48, 1/3)$	10.1179	25.9716	29.6706
$30-70$	$(48, 1/3)$	15.0000	$(48, 1/3)$	10.4608	24.9646	28.5689
$50-50$	$(48, 1/3)$	15.0000	$(48, 1/3)$	10.7574	24.1222	27.6593
$70-30$	$(48, 1/3)$	15.0000	$(48, 1/3)$	11.0116	23.4013	26.8828
$90-10$	$(48, 1/3)$	15.0000	$(48, 1/3)$	11.2325	22.7709	26.2046
			MMD			
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$10-90$	$(48, 1/3)$	15.0000	$(48, 1/3)$	6.8309	28.7016	28.7016
$30-70$	$(48, 1/3)$	15.0000	$(48, 1/3)$	7.7878	26.5355	26.5355
$50-50$	$(48, 1/3)$	15.0000	$(48, 1/3)$	8.4948	25.0255	25.0255
$70-30$	$(48, 1/3)$	15.0000	$(48, 1/3)$	9.0528	23.8579	23.8579
$90-10$	$(48, 1/3)$	15.0000	$(48, 1/3)$	9.5102	22.9029	22.9029

Actually, this fact explains the downward trend of the PSNR values illustrated in Figs. 3.7(a) and 3.7(b) for both motion classes, for all examined bitrate and bandwidth combinations. Specifically, Fig. 3.7(a) includes the PSNR variations for the high-motion class of nodes, while Fig. 3.7(b) illustrates the same results for the low-motion class of nodes, for the MAD criterion. Figure 3.8 offers the corresponding information for the MMD criterion. Note that for the MMD case, the PSNR values for both motion classes are identical.

Another issue that has gained our interest is the efficiency of PSO against that of AS in terms of the number of iterations required for obtaining solutions of same quality with AS. Figure 3.9 offers this information. In particular, it illustrates the number of iterations required by PSO, averaged over the 30 independent experiments, to detect the same solution with AS for different levels of precision, namely 3, 4, 5, and 6 decimal digits. In each subfigure, both criteria are compared for all examined bitrate and bandwidth combinations. Figures 3.9(a) and 3.9(b) present the aforementioned results for the node distribution "30-70", while Figs. 3.9(c) and 3.9(d) for the node distribution "70-30". Moreover, Figs. 3.9(a) and 3.9(c) refer to the noiseless case ($N_0 = 0$ W/Hz), and Figs. 3.9(b) and 3.9(d) to the noisy case ($N_0 = 10^{-7}$ W/Hz).

From Fig. 3.9 we corroborate that PSO requires fewer iterations to achieve same quality solutions with AS for lower precisions. As expected, increasing precision is accompanied by a higher number of iterations. Furthermore, it is obvious that the MMD criterion needs more function evaluations than MAD to achieve the optimal solution, while the addition of noise incurs an increase to the number of iterations for both optimization criteria. The aforementioned confirmations derive from all considered bitrate-bandwidth combinations and node distributions of the specific group of experiments.

Figure 3.10 presents the success ratio of AS in achieving the optimal solution, for both MAD and MMD criteria. Particularly, each of the cases as referred to this
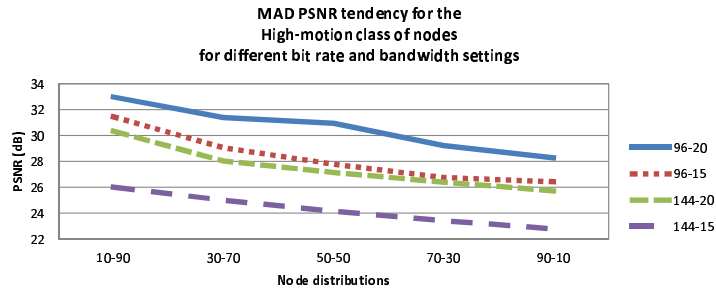
(a)
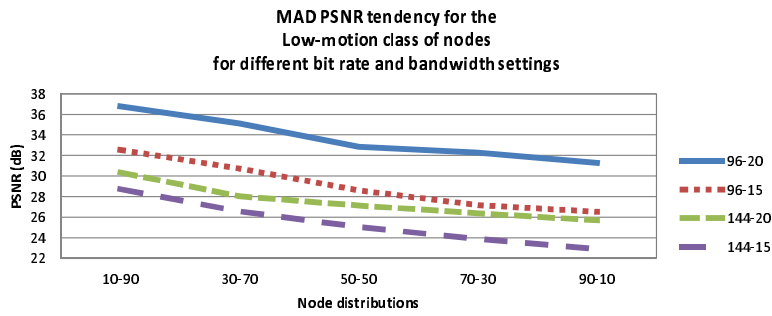
(b)

Figure 3.7: PSNR tendency for the MAD criterion (a) for the high-motion, and (b) low-motion class of nodes, for various node distributions, when $R_{\text{target}} = 96$ kbps, $W_t = 20$ MHz, and $N_0 = 0$ W/Hz.
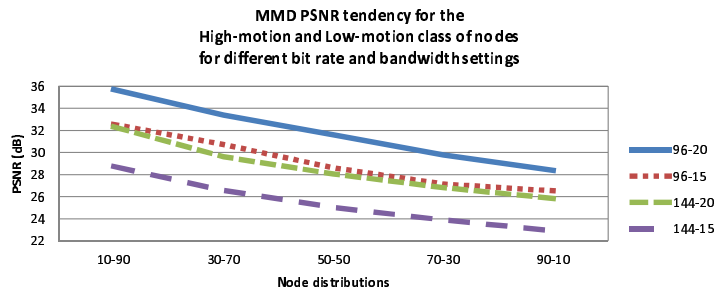
Figure 3.8: PSNR tendency for the MMD criterion, for both motion classes, for various node distributions, when $R_{\text{target}} = 96$ kbps, $W_t = 20$ MHz, and $N_0 = 0$ W/Hz.
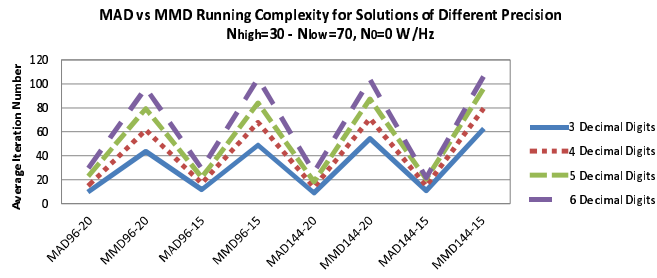
(a)
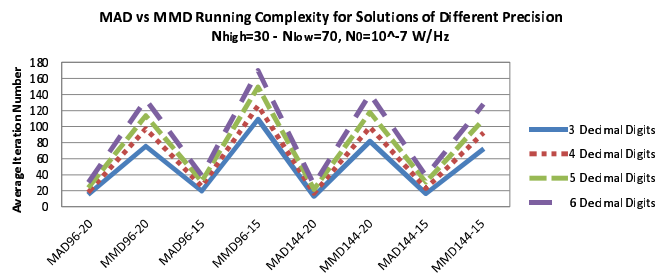
(b)
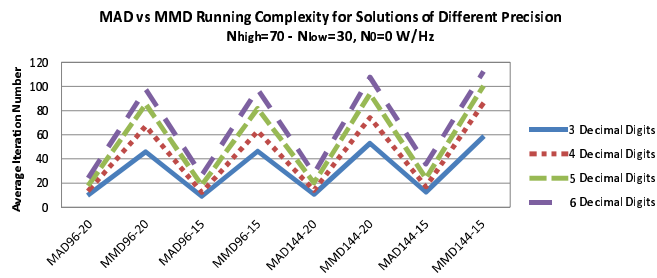
(c)
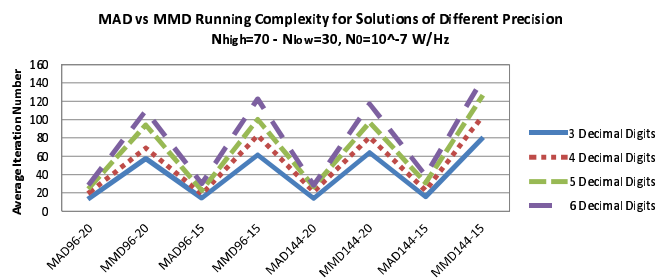
(d)

Figure 3.9: Average number of iterations required by PSO to obtain the solution given by AS, for both criteria, and precision of 3, 4, 5 and 6 decimal digits, for all examined bitrate-bandwidth combinations, for two different node distributions. Figures 3.9(a) and 3.9(c) correspond to the noiseless case, while Figs. 3.9(b) and 3.9(d) refer to the noisy case.

70
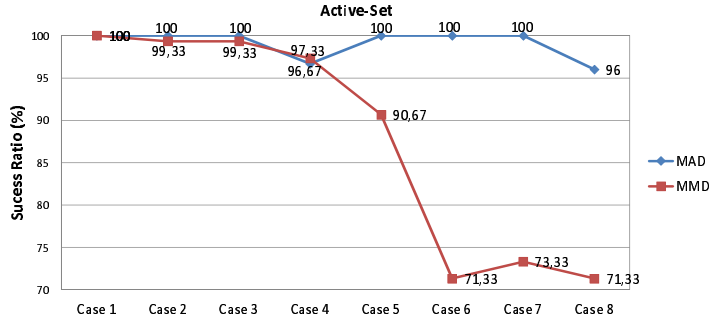
Figure 3.10: Success ratios of AS in finding the optimal solution.

figure corresponds to the results of each corresponding table. For example, Case 3 corresponds to the results of Table 3.3, when $R_{\text{target}} = 144\,\text{kbps}$, $W_t = 20\,\text{MHz}$, and $N_0 = 0\,\text{W/Hz}$. For this case, we observe that the MAD has a success ratio of 100% and the MMD of 99.33%. For the MAD criterion, this means that each of the node distributions manages to achieve the same best solution in all 30 experiments. In contrast, the success ratio for the MMD implies that one experiment out of 30, for a specific node distribution, failed to reach the best solution.

Interpreting the AS efficiency for MAD and MMD, we infer that the MAD criterion is capable of detecting the best solution in most of the cases and exhibits better performance than the MMD. The superiority for MAD is clearer in noisy cases (Cases $5-8$) where for MMD it presents success ratios of nearly 70%. Although it may seems high, we indicatively refer that for the case of $N_{\text{high}} = 90 - N_{\text{low}} = 10$ of Table 3.8, only $7/30$ experiments reach the optimal solution. Therefore, this figure sheds light on the weakness of AS in detecting always the best solution, while it also demonstrates that its efficiency depends on both the initially supplied starting point as well as the objective function to be optimized.

Lastly, Table 3.9 offers an intuition regarding the execution times required by each optimization algorithm averaged over 30 experiments per problem instance. Also, the total running times needed for the execution of all cases of node distributions for each specific bitrate and bandwidth combination are presented. The same table includes results for both criteria and all cases of considered bitrate and bandwidth, for both the noiseless and noisy cases. We shall note that AS has been set to execute 40000 function evaluations (as PSO and HPSOAS), with the function tolerance set to 10^{-15} and the tolerance of the candidate optimal solution to 10^{-15}. We have set the function and optimal point tolerances to such levels since we have confirmed that the PSNR results provided by the AS have been different from those of PSO even to the first decimal digit for some cases. At this point, we should recall that the AS method was run sequentially 9 times for each tested case in order to evaluate all possible combinations of bitrate and bandwidth. The execution times of this method are averaged over the number of successful experiments per case.

71

An overall inspection of the results denotes that despite PSO is able to reach the optimal solutions efficiently in all 30 experiments of each considered case, as opposed to AS, it still needs more time to be executed compared to both AS and HPSOAS. Particularly, more time is needed as more nodes are assigned to the low-motion class of nodes. Moreover, it follows that MAD takes less time to reach the optimal point compared to MMD, while in the noisy case MMD needs double the time to achieve the optimal solution compared to the noiseless case using PSO.

Commenting on the AS execution times, from Table 3.9 it is clear that it requires less time compared to PSO, with two exceptions for the noiseless case of the MMD criterion, when the bitrate is set to $R_{\text{target}} = 144$ kbps. This probably means that the randomly supplied starting points are far away from the optimal solution and that the shape of the objective function complicates further the finding of the solution. Additionally, we may think that the AS method follows the brute force approach testing all possible cases of source and channel coding rate pair values, increasing the overall problem's complexity. Therefore, this method is impractical to applications where problems with higher dimensions exist. For example, if we had made the assumption for a node clustering into 10 motion classes, the AS should be executed 3^{10} times in order to determine the optimal source and channel coding rate combinations for each motion class, instead of 3^2 times of our problem.

Therefore, as it was also mentioned in Section 3.6.3 in order to surmount all weaknesses of both PSO and AS keeping their benefits at the same time, we have resorted to the development of the HPSOAS optimization method. Table 3.9 demonstrates that the optimization of MMD criterion using the HPSOAS method requires significantly less time compared to both PSO and AS. Similarly, the MAD optimized under the use of the HPSOAS saves much execution time compared to PSO and the same holds also in half the cases compared to AS.

Interesting enough are also the conclusions drawn by Fig 3.11. The goal of the specific illustration is to give us an insight regarding the overall execution times for each of the MAD and MMD criteria, when we aggregate the total times for each considered bitrate and bandwidth combination, for both the noiseless and noisy cases. Evidently, it follows that the MMD takes longer to converge compared to the MAD, for all considered optimization methods. Moreover, for the MMD it is clear that HPSOAS needs about $1/6^{th}$ and $1/4^{th}$ of the corresponding time of PSO and AS, respectively, in order to converge to the optimal point. Further is the gain in time by adopting the HPSOAS to the MAD criterion instead of PSO. In this case, HPSOAS reaches the optimal solution in about $1/12^{th}$ of PSO's time. Comparable execution times are required cumulatively from both AS and HPSOAS for the MAD criterion.

Summarizing, HPSOAS keeps the following strong points that renders its use very appealing: i) it is not sensitive to initial conditions, ii) there is no need to execute an exhaustive search in order to determine the discrete parameters of this problem, iii) it can be applied to problems of higher dimensions without a tremendous impact on

Table 3.9: Average execution times (in seconds).

$N_{\text{high}} - N_{\text{low}}$	$R_{\text{target}} = 96$ kbps, $W_t = 20$ MHz, $N_0 = 0$ W/Hz						$R_{\text{target}} = 96$ kbps, $W_t = 20$ MHz, $N_0 = 10^{-7}$ W/Hz					
	MAD			MMD			MAD			MMD		
	PSO	AS	HPSOAS	PSO	AS	HPSOAS	PSO	AS	HPSOAS	PSO	AS	HPSOAS
$10 - 90$	148.78	4.67	5.43	154.34	7.36	9.71	154.83	4.93	10.70	319.55	7.08	13.24
$30 - 70$	117.27	4.91	3.74	125.29	97.94	16.41	124.11	5.20	11.90	245.21	51.37	41.91
$50 - 50$	81.74	6.43	3.59	94.55	53.01	32.22	87.08	5.61	0.85	185.45	47.82	48.75
$70 - 30$	51.77	6.27	0.79	53.56	43.78	8.78	56.75	5.95	1.23	122.51	53.13	23.14
$90 - 10$	25.44	6.47	0.61	29.01	125.44	13.29	28.15	8.88	4.04	58.87	70.79	50.42
Total time	425.00	28.75	14.16	456.75	327.53	80.41	450.92	30.57	28.72	931.59	230.19	177.46

$N_{\text{high}} - N_{\text{low}}$	$R_{\text{target}} = 96$ kbps, $W_t = 15$ MHz, $N_0 = 0$ W/Hz						$R_{\text{target}} = 96$ kbps, $W_t = 15$ MHz, $N_0 = 10^{-7}$ W/Hz					
	MAD			MMD			MAD			MMD		
	PSO	AS	HPSOAS	PSO	AS	HPSOAS	PSO	AS	HPSOAS	PSO	AS	HPSOAS
$10 - 90$	182.11	3.32	15.89	183.22	12.38	37.24	169.20	4.64	2.85	352.83	28.09	27.30
$30 - 70$	151.86	6.32	6.32	129.84	20.86	21.51	141.15	5.76	13.33	295.61	51.79	13.05
$50 - 50$	121.44	6.17	18.32	99.39	131.27	28.98	103.18	5.58	10.96	243.32	92.02	27.58
$70 - 30$	68.89	6.63	5.52	63.67	165.39	40.93	61.64	6.00	0.61	133.04	86.56	17.84
$90 - 10$	29.69	9.53	1.18	26.83	114.67	17.44	29.10	8.97	2.65	52.54	90.14	31.35
Total time	553.99	25.45	47.23	502.95	444.57	146.10	504.27	30.95	30.40	1077.34	348.60	117.12

$N_{\text{high}} - N_{\text{low}}$	$R_{\text{target}} = 144$ kbps, $W_t = 20$ MHz, $N_0 = 0$ W/Hz						$R_{\text{target}} = 144$ kbps, $W_t = 20$ MHz, $N_0 = 10^{-7}$ W/Hz					
	MAD			MMD			MAD			MMD		
	PSO	AS	HPSOAS	PSO	AS	HPSOAS	PSO	AS	HPSOAS	PSO	AS	HPSOAS
$10 - 90$	183.50	4.51	19.78	176.71	26.23	13.75	148.45	5.50	7.07	359.43	48.07	85.15
$30 - 70$	137.70	5.60	4.72	147.39	106.24	12.33	120.21	5.16	10.33	284.55	89.94	47.53
$50 - 50$	107.84	6.17	12.90	119.47	116.33	22.95	88.51	4.46	1.24	217.15	95.29	26.18
$70 - 30$	69.08	6.38	22.96	86.76	198.52	59.87	60.34	5.18	21.44	166.85	109.61	24.80
$90 - 10$	26.34	7.57	4.26	29.97	201.45	24.68	25.05	6.65	2.35	90.57	105.50	51.44
Total time	524.46	30.23	64.62	560.30	648.77	133.58	442.56	26.95	42.43	1118.55	448.41	235.10

| $N_{\text{high}} - N_{\text{low}}$	$R_{\text{target}} = 144$ kbps, $W_t = 15$ MHz, $N_0 = 0$ W/Hz						$R_{\text{target}} = 144$ kbps, $W_t = 15$ MHz, $N_0 = 10^{-7}$ W/Hz					
	MAD			MMD			MAD			MMD		
	PSO	AS	HPSOAS	PSO	AS	HPSOAS	PSO	AS	HPSOAS	PSO	AS	HPSOAS
$10 - 90$	152.84	3.65	2.44	155.06	176.00	13.98	166.51	4.60	1.67	343.07	77.41	25.07
$30 - 70$	122.65	6.08	0.54	127.42	101.08	8.32	142.24	4.92	16.71	299.31	75.09	17.36
$50 - 50$	93.27	7.39	10.83	100.01	178.71	50.09	106.31	5.22	22.97	243.20	98.60	23.78
$70 - 30$	57.95	6.85	0.95	59.56	333.43	17.66	61.56	5.54	12.71	172.14	102.95	53.07
$90 - 10$	32.31	9.38	4.95	31.83	313.93	27.19	28.04	6.77	3.44	75.48	106.72	30.97
Total time	459.02	33.35	19.71	473.88	1103.15	117.24	504.66	27.05	57.5	1133.20	460.77	150.25
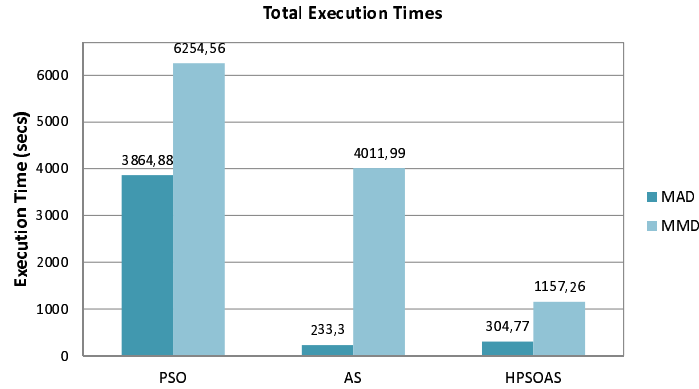
Figure 3.11: Total execution times over all node distributions and bitrate-bandwidth combinations of both the noiseless and noisy cases, for MAD and MMD, using all optimization methods.

problem's complexity, and iv) it takes much less time for its execution compared to other stochastic methods.

Regarding the key points of the specific set of experiments, we should mention that the MAD criterion works favorably for the nodes that image low motion. On the other hand, the MMD assigns equal PSNR values to both motion classes, although it is not sufficiently clear how fair it can be for the nodes that record low motion, taking into consideration the significant PSNR reduction observed in this case, for this class of nodes. Furthermore, our results confirm that the CCU receives less power with the MAD than with the MMD, implying that MAD requires less power for data transmission.

The experiments conducted under the presence of thermal and background noise verify the conclusions derived for the noiseless case. The main impact of noise is a marginal reduction of the PSNR of both motion classes and optimization criteria, with only a minor exception. Also, the nodes that detect high levels of motion require considerably higher power levels than the nodes that detect low levels of motion to accomplish data transmission. Regarding the performance of the PSO algorithm, it has been assessed in comparison with the performance of the AS method, justifying its comparative advantage in tackling such problems. Additionally, experimental results using HPSOAS, PSO ans AS highlight the superiority of HPSOAS over both AS and PSO, under various aspects.

2. **Second Set of Experiments**

In this second set of experiments, our attention has been focused on the evaluation of the NBS criterion, and particularly of its two variants, that is NNBS and CNBS, where we assume that thermal and background noise is negligible compared with the interference, under the system setup as described in case (b) of Section 3.8.2. In the same group of experiments the results from both NNBS and CNBS are assessed in comparison with the results from the MAD and MMD criteria, while also a great emphasis is given on PSO's efficiency against AS, IP and TRR optimization algorithms.

In the specific group of experiments, we have to maximize the Nash product. Due to this, we have used PSO in order to minimize its negative, given by

$$f(x) = -\Big((U_a(cb_a, S_a) - dp_a)^{a_a} \cdot (U_{md}(cb_{md}, S_{md}) - dp_{md})^{a_{md}} \cdot (U_s(cb_s, S_s) - dp_s)^{a_s}$$
$$\cdot (U_h(cb_h, S_h) - dp_h)^{a_h} \cdot (U_f(cb_f, S_f) - dp_f)^{a_f} \cdot (U_c(cb_c, S_c) - dp_c)^{a_c}\Big),$$
(3.38)

and identical parameter settings have been utilized for both NNBS and CNBS variants.

First of all, we have explored the effect of the value of the disagreement point vector on the results of the NBS-based criteria. Tables 3.10-3.13 show the achieved PSNR values for the NNBS and CNBS, using PSO as the optimization solver. We have tested different selections of the dp vector, for the same bitrate and bandwidth combinations. The tested values for the vector of the dp have also been different for each bitrate and bandwidth combination, since they must be feasible values using the available bitrate and bandwidth, at each time. Additionally, all the elements of a dp vector have been equal. Although this is not obligatory, we have made this assumption in an effort to be equally fair to all motion classes.

In practice, video that includes high amounts of motion is particularly important in surveillance applications, since this is where the action occurs. From this round of experiments, we confirm that increasing the values of the dp vector results in favoring the nodes that image high motion levels more than the rest of the nodes, a fact that is expounded by the achieved PSNR values for each motion class. This is what the values in bold denote on Tables 3.10-3.13. In addition, the amelioration of video quality becomes more perceivable in videos with poor quality rather than in videos with good quality. Therefore, we have assigned to dp the highest values among the tested ones for each combination of bitrate and bandwidth. All our experiments with the NNBS and CNBS have been conducted using:

(i) dp $= 28$ dB for the cases

    (a) $R_{target} = 96$ kbps and $W_t = 20$ MHz

    (b) $R_{target} = 144$ kbps and $W_t = 30$ MHz

(ii) dp $= 26$ dB for the cases

(a) $R_{\text{target}} = 96$ kbps and $W_t = 15$ MHz

(b) $R_{\text{target}} = 144$ kbps and $W_t = 22.5$ MHz.

Table 3.10: PSNR values for different dp, using PSO, for $R_{\text{target}} = 96$ kbps and $W_t = 20$ MHz.

		NNBS				CNBS			
	Sequences	dp = 28	dp = 27	dp = 26	dp = 25	dp = 28	dp = 27	dp = 26	dp = 25
Case 1	"Akiyo"	36.5112	36.6797	36.8173	36.9319	36.5112	36.6797	36.8173	36.9319
	"Mother&Daughter"	34.5452	34.6279	34.6957	34.7521	34.5452	34.6279	34.6957	34.7521
	"Salesman"	33.0986	33.0871	33.0777	33.0697	33.0986	33.0871	33.0777	33.0697
	"Hall"	33.8855	33.9285	33.9637	33.9929	33.8855	33.9285	33.9637	33.9929
	"Foreman"	33.5774	33.6097	33.6378	33.6621	33.5774	33.6097	33.6378	33.6621
	"Coastguard"	**31.7825**	31.6143	31.4685	31.3413	**31.7825**	31.6143	31.4685	31.3413
Case 2	"Akiyo"	36.9809	36.8912	37.1221	37.2022	36.9424	36.9382	36.9356	36.9334
	"Mother&Daughter"	34.9393	34.8361	34.9994	35.0243	33.6715	33.6401	33.6110	33.5875
	"Salesman"	33.3904	33.2421	33.3045	33.2733	33.6637	33.6380	33.6148	33.5940
	"Hall"	34.2718	35.2623	35.4899	35.5408	35.8310	35.9345	36.0190	36.0895
	"Foreman"	35.0184	34.8631	35.1263	35.1739	**35.3444**	35.3347	35.3289	35.3239
	"Coastguard"	**32.8463**	32.6105	31.7981	31.6474	**32.7816**	32.7745	32.7702	32.7665
Case 3	"Akiyo"	36.9558	36.8917	37.0031	37.1001	37.4139	37.5529	37.6663	37.7609
	"Mother&Daughter"	34.8905	34.8361	34.8806	34.9213	35.4221	35.4903	35.5448	35.5896
	"Salesman"	33.3543	33.2422	33.2158	33.1963	32.8240	32.8028	32.7937	32.7862
	"Hall"	34.2239	34.1341	35.3394	35.4096	33.1350	33.0680	33.0095	32.9582
	"Foreman"	34.0659	34.8633	33.9198	33.9284	**35.7753**	35.7898	35.7631	35.7409
	"Coastguard"	**32.8041**	32.6106	31.6680	31.5307	**33.1541**	33.1083	33.0887	33.0725
Case 4	"Akiyo"	35.8036	36.1377	36.3882	36.5839	37.0979	37.5082	37.8145	38.0531
	"Mother&Daughter"	33.8701	34.0992	34.2704	34.4030	35.1134	35.4458	35.6941	35.8869
	"Salesman"	32.5969	32.6923	32.7593	32.8080	33.5191	33.6951	33.8216	33.9162
	"Hall"	33.2280	33.4080	33.5419	33.6446	35.5853	36.0320	36.3737	36.6444
	"Foreman"	32.6798	32.8640	33.0096	33.1267	**30.7193**	30.5880	30.4715	30.3670
	"Coastguard"	**30.1005**	29.8555	29.6511	29.4783	**29.8901**	29.6107	29.3772	29.1796

On Tables 3.14-3.17 we report the results for the four different node distributions and all considered criteria, solved by PSO, for all bitrate and bandwidth combinations. The reported quantities at each line of the tables correspond to the power level, S, the combination of source and channel coding rate, (R_s,R_c), and the achieved utility, PSNR, per class. The values in bold refer to the total power levels and achieved PSNR values for all node distributions and criteria.

Studying the performance of NNBS and CNBS, we observe that when all motion classes have the same cardinality (Case 1), both NBS approaches offer exactly the same solution, i.e, the same PSNR values to all motion classes. In Case 2, the NNBS offers higher PSNR values compared to the CNBS to the nodes that describe low amounts of motion. In Case 3, the same criterion (NNBS) assigns higher PSNR values compared to the CNBS to the nodes that describe medium amounts of motion and in Case 4, higher PSNR values are assigned to the nodes that describe high amounts of motion, using also the NNBS compared to the CNBS. In all other cases, the CNBS "beats" the NNBS, by assigning higher PSNR values compared to the latter.

Continuing, the performance of the NNBS and CNBS criteria has been compared with the performance achieved by the MAD and MMD criteria. For the Cases 2 and

Table 3.11: PSNR values for different dp, using PSO, for $R_{\text{target}} = 96$ kbps and $W_t = 15$ MHz.

	Sequences	NNBS				CNBS			
		dp = 26	dp = 25	dp = 24	dp = 23	dp = 26	dp = 25	dp = 24	dp = 23
Case 1	"Akiyo"	35.6945	35.9260	36.1113	36.2636	35.6945	35.9260	36.1113	36.2636
	"Mother&Daughter"	33.5979	33.7564	33.8830	33.9863	33.5979	33.7564	33.8830	33.9863
	"Salesman"	32.2508	32.3180	32.3695	32.4100	32.2508	32.3180	32.3695	32.4100
	"Hall"	32.8823	33.0063	33.1049	33.1850	32.8823	33.0063	33.1049	33.1850
	"Foreman"	**30.2644**	30.1824	30.1099	30.0453	**30.2644**	30.1824	30.1099	30.0453
	"Coastguard"	**29.2214**	29.0370	28.8818	28.7496	**29.2214**	29.0370	28.8818	28.7496
Case 2	"Akiyo"	36.1434	36.3211	36.4657	36.5859	35.3258	35.3934	35.2294	35.4980
	"Mother&Daughter"	34.0389	34.1504	34.2403	34.3142	32.5165	32.5537	33.0002	32.6025
	"Salesman"	32.5823	32.6143	32.6382	32.6564	32.7412	32.7807	33.1431	32.8332
	"Hall"	33.3179	33.3982	33.4622	33.5142	33.5278	33.6192	34.1380	33.7512
	"Foreman"	**30.6672**	30.5615	30.4680	30.3847	32.5214	32.6343	31.1627	32.8104
	"Coastguard"	**29.5215**	29.3229	29.1539	29.0087	**29.6692**	29.4868	29.6795	29.1970
Case 3	"Akiyo"	36.0155	36.2113	36.3704	36.5026	36.3045	36.4973	36.6539	36.7839
	"Mother&Daughter"	33.9130	34.0407	34.1441	34.2294	34.1980	34.3268	34.4307	34.5161
	"Salesman"	32.4877	32.5319	32.5659	32.5927	31.5356	31.5615	31.5818	31.5980
	"Hall"	33.1933	33.2889	33.3659	33.4290	31.7751	31.7760	31.7738	31.7697
	"Foreman"	**30.5507**	30.4550	30.3709	30.2964	32.8400	32.8961	32.9403	32.9756
	"Coastguard"	**29.4348**	29.2427	29.0802	28.9414	**29.6321**	29.4531	29.3008	29.1699
Case 4	"Akiyo"	34.9081	35.2441	35.5041	35.7124	35.8943	36.2671	36.5527	36.7796
	"Mother&Daughter"	32.8337	33.0818	33.2741	33.4280	33.7938	34.0965	34.3282	34.5117
	"Salesman"	31.6736	31.8085	31.9101	31.9892	32.3982	32.5737	32.7042	32.8046
	"Hall"	32.1320	32.3380	32.4979	32.6256	33.0756	33.3444	33.5502	33.7129
	"Foreman"	**29.5999**	29.5560	29.5162	29.4799	**29.0024**	28.8656	28.7452	28.6382
	"Coastguard"	**28.7254**	28.5630	28.4288	28.3160	**28.2780**	28.0378	27.8366	27.6658

Table 3.12: PSNR values for different dp, using PSO, for $R_{\text{target}} = 144$ kbps and $W_t = 30$ MHz.

	Sequences	NNBS				CNBS			
		dp = 28	dp = 27	dp = 26	dp = 25	dp = 28	dp = 27	dp = 26	dp = 25
Case 1	"Akiyo"	36.4608	36.7424	36.9780	37.1782	36.4608	36.7424	36.9780	37.1782
	"Mother&Daughter"	34.5607	34.7048	34.8257	34.9284	34.5607	34.7048	34.8257	34.9284
	"Salesman"	33.6832	33.7449	33.7959	33.8382	33.6832	33.7449	33.7959	33.8382
	"Hall"	34.0215	34.1143	34.1915	34.2564	34.0215	34.1143	34.1915	34.2564
	"Foreman"	33.5108	33.5653	33.6114	33.6504	33.5108	33.5653	33.6114	33.6504
	"Coastguard"	**30.9488**	30.6709	30.4247	30.2056	**30.9488**	30.6709	30.4247	30.2056
Case 2	"Akiyo"	36.9372	37.4978	37.6764	37.8302	35.3652	35.4299	35.4863	35.5358
	"Mother&Daughter"	34.9906	35.4015	35.4793	35.5459	33.5877	33.5149	33.4510	33.3944
	"Salesman"	34.0740	34.3847	34.4003	34.4124	34.9117	34.9451	34.9743	34.9999
	"Hall"	34.4212	34.7646	34.8034	34.8357	35.2744	35.3324	35.3829	35.4272
	"Foreman"	33.9734	34.3446	34.3610	34.3739	34.9981	35.0481	35.0932	35.1340
	"Coastguard"	**32.7052**	31.1199	30.8668	30.6401	**33.5649**	33.4763	33.3962	33.3237
Case 3	"Akiyo"	36.9433	37.1866	37.3925	37.5692	37.6623	37.9145	38.1293	38.3148
	"Mother&Daughter"	34.9962	35.1135	35.2129	35.2982	35.6522	35.7891	35.9063	36.0077
	"Salesman"	34.0791	34.1199	34.1538	34.1819	32.6142	32.4898	32.3800	32.2823
	"Hall"	34.4263	34.4957	34.5540	34.6033	32.9227	32.8314	32.7509	32.6794
	"Foreman"	33.9795	34.0190	34.0526	34.0812	34.7077	34.7907	34.8639	34.9287
	"Coastguard"	**31.2108**	30.9324	30.6851	30.4646	**33.3202**	33.2569	33.1990	33.1457
Case 4	"Akiyo"	35.4595	35.9111	36.2769	36.5806	36.6970	37.2631	37.7171	38.0907
	"Mother&Daughter"	33.6705	33.9479	34.1759	34.3666	34.7734	35.1841	35.5176	35.7938
	"Salesman"	32.8784	33.0533	33.1971	33.3171	33.8763	34.1849	34.4358	34.6434
	"Hall"	33.1952	33.4086	33.5836	33.7294	34.2192	34.5617	34.8392	35.0684
	"Foreman"	**31.6048**	31.5892	31.5741	31.5590	**30.8078**	30.6425	30.4941	30.3591
	"Coastguard"	**30.4360**	30.2016	29.9977	29.8190	**29.8923**	29.5397	29.2286	28.9526

Table 3.13: PSNR values for different dp, using PSO, for $R_{\text{target}} = 144$ kbps and $W_t = 22.5$ MHz.

		NNBS				CNBS			
	Sequences	dp = 26	dp = 25	dp = 24	dp = 23	dp = 26	dp = 25	dp = 24	dp = 23
Case 1	"Akiyo"	34.1645	34.4186	34.5889	34.6786	34.1645	34.4186	34.5889	34.6786
	"Mother&Daughter"	32.0453	32.1855	32.3602	32.5980	32.0453	32.1855	32.3602	32.5980
	"Salesman"	31.1217	31.1968	31.2682	31.3454	31.1217	31.1968	31.2682	31.3454
	"Hall"	31.5210	31.6229	31.8874	31.9786	31.5210	31.6229	31.8874	31.9786
	"Foreman"	**29.2876**	29.2015	29.0012	28.9201	**29.2876**	29.2015	29.0012	28.9201
	"Coastguard"	**28.0057**	27.7981	27.5629	27.3057	**28.0057**	27.7981	27.5629	27.3057
Case 2	"Akiyo"	34.8552	35.0496	35.2546	35.5378	33.1191	33.2324	33.3648	33.5678
	"Mother&Daughter"	32.6833	32.7767	32.9886	33.2501	31.0925	31.0872	31.2674	31.3897
	"Salesman"	31.7107	31.7459	31.8761	32.0452	32.6571	32.7461	32.8964	33.1562
	"Hall"	32.1199	32.1790	32.3564	32.5648	33.0788	33.1889	33.3617	33.4879
	"Foreman"	**29.8588**	29.7500	29.6420	29.5273	**30.8089**	30.7803	30.6348	30.2468
	"Coastguard"	**28.4254**	28.2047	28.1036	28.0010	**29.1200**	28.9639	28.7464	28.5560
Case 3	"Akiyo"	34.5609	34.7806	34.8978	35.1762	35.7362	36.0013	36.2564	36.5963
	"Mother&Daughter"	32.4107	32.5241	32.7896	32.9968	33.5056	33.6763	33.7888	34.2165
	"Salesman"	31.4588	31.5112	31.7862	31.9689	30.0895	30.0402	30.2658	30.5986
	"Hall"	31.8640	31.9414	32.3619	32.6891	30.4669	30.4470	30.6790	30.8970
	"Foreman"	**29.6126**	29.5140	29.3456	29.2165	**30.6207**	30.6106	30.0631	29.7532
	"Coastguard"	**28.2447**	28.0299	27.8962	27.6031	**28.9827**	28.8393	28.6866	28.4650
Case 4	"Akiyo"	33.0309	33.3978	33.5698	33.8938	34.4020	34.8576	35.1395	35.4443
	"Mother&Daughter"	31.0128	31.2392	31.0012	30.8964	32.2639	32.5963	32.3363	32.1161
	"Salesman"	30.1727	30.3207	30.6896	30.8896	31.3233	31.5783	31.8622	32.1116
	"Hall"	30.5520	30.7328	30.9863	31.3489	31.7262	32.0093	32.4658	32.6874
	"Foreman"	**28.4022**	28.3533	28.0130	27.7985	**27.4835**	27.3023	27.1542	26.9868
	"Coastguard"	**27.3516**	27.1656	26.8543	26.6366	**26.6678**	26.3748	26.2010	26.0012

3, a wise selection between NNBS and CNBS can always give higher PSNR results compared to the MAD, to all motion classes. For Cases 1 and 4 the same can also be done, but with some exceptions. Regarding the MMD, typically this criterion offers the same utilities to all nodes. From the obtained results, we observe that the MMD in all examined cases assigns exactly the same PSNR values to both classes of nodes that describe high amounts of motion. Also, in the large majority of the cases, the same PSNR values are also assigned to the sequence/sequences that describe medium amounts of motion. Last but not least, there are also some cases where all motion classes enjoy exactly the same PSNR values.

However, these results reveal that there are some cases where some nodes receive a lower distortion than that of the other nodes. At the same time, in these cases, these nodes need the lowest possible power level. Specifically, they need 5.0 W, which is the low bound of the considered power level range. As a result, these nodes achieve lower distortions than the rest of the nodes, despite the fact that they use the least possible power. Evidently, if we had allowed a lower bound for the power level range, the specific nodes would use even less power and thus, all nodes would receive exactly the same distortion (thus, the same PSNR).

Comparing the performance of the MMD criterion with the performance of the NNBS and CNBS criteria, the latters can be wisely used so as to assign higher PSNR values to the low and medium video sequences, with some exceptions for the case of $R_{\text{target}} = 96$

kbps and $W_t = 20$ MHz. Additionally, the higher the amounts of motion included in a sequence, the higher the power level that is required and also, the lower the PSNR value that is achieved. In our results, we have normalized the power levels so that the lowest allocated power is equal to 5.0 W. The nodes that describe high motion usually use more bits to compress their data and leave fewer bits that can be used to protect the sequence from transmission errors. In the following, Tables 3.14-3.17 confirm our conviction that decreasing the bandwidth while keeping the bitrate constant, the value of Eq. (3.8) decreases, incurring a PSNR decrease to all motion classes.

For the total consumed power, there is no specific scheme that requires the highest amounts of power levels in all cases of a specific bitrate and bandwidth combination, except for the case of $R_{\text{target}} = 144$ kbps and $W_t = 22.5$ MHz, where the MMD criterion clearly needs far more power compared to the other schemes. Furthermore, it seems that Case 2 is the most demanding in resources (in terms of power) compared to the other three cases, when $R_{\text{target}} = 96$ kbps and $W_t = 20$ MHz, and $R_{\text{target}} = 144$ kbps and $W_t = 30$ MHz. The same holds also in half the cases of $R_{\text{target}} = 96$ kbps and $W_t = 15$ MHz, and $R_{\text{target}} = 144$ kbps and $W_t = 22.5$ MHz. Last but not least, the highest total PSNR values are achieved in Case 2 of all examined bitrate and bandwidth combinations, with only an isolated exception.

All of the optimization criteria examined in this round of experiments, i.e., the Nash bargaining solution in terms of its two different approaches, that is NNBS and CNBS, the MAD and the MMD provide Pareto-optimal solutions. Specifically for the NBS, let us assume that the provided solution (the solutions from the NNBS and CNBS), i.e., the solution that maximizes the Nash product is not Pareto-optimal. This means that there is another solution where at least one node strictly increases its utility and no node decreases its utility. However, such a solution would lead to an even greater Nash product, thus contradicting the fact that the NBS maximizes the Nash product. Therefore, the solution provided by the NBS criterion is Pareto-optimal.

Similar reasoning applies also to the MAD criterion. If we assume that the solution given by the MAD is not Pareto-optimal, this means that there is another solution where at least one node receives lower distortion and no node increases its distortion. However, such a solution would lead to an even smaller average distortion, thus contradicting the fact that the MAD criterion minimizes the average distortion. Therefore, the solution provided by the MAD criterion is Pareto-optimal.

As mentioned earlier, the MMD solution occurs when an "equilibrium" is reached, i.e., when all nodes have the same distortion (except for some cases where a node, i.e., motion class, uses a power level at the lower end of the considered power level range), and increasing a node's transmission power will increase the distortions of the other nodes, thus, leading to a higher maximum distortion.

If the solution given by the MMD is not Pareto-optimal, this would mean that there exists another solution where at least one node receives a lower distortion and no node increases its distortion. Such a solution would result in a deviation from the "equilib-

rium". Thus, the lower distortion of one node would be a result of an increase of its power level. This would lead to an increase of the distortions of the other nodes. Therefore, an alternative solution where at least one node receives a lower distortion and no node increases its distortion cannot exist and the MMD leads to a Pareto-optimal solution. For comparison reasons, each examined criterion, except for PSO has also been run using three competing optimization algorithms, for all cases of node distributions as described in case (b) of Section 3.8.2. Specifically, PSO's performance has been compared with that of the deterministic algorithms AS [60, 166], IP [25, 26] and TRR [34, 35]. Each of these methods has been run for the same maximum number of function evaluations as PSO, i.e., 70000 function evaluations. Furthermore, 30 independent experiments have also been conducted for each one of the aforementioned deterministic methods, starting from a different, random starting point at each experiment, within the range $[5.0, 15.0]$.

In the following, Tables 3.18-3.21 provide statistical information regarding the performance of all the aforementioned algorithms, over all independent trials. Specifically, each table presents the results for a particular bitrate and bandwidth combination. The column "Case" refers to a specific node distribution. The column "Success" shows how many times each algorithm succeeds in finding the optimal solution to a precision of six decimal digits, out of 30 independent trials. Columns "Min", "Mean", "Max" and "Median" report the min, mean, max and median value of the function, respectively, over the 30 experiments. The standard deviation of the 30 values of the function is presented under the column "Std". We should note that for the NNBS and CNBS criteria the objective function is the one given by Eq. (3.38), assuming different bargaining powers for each of them. Similarly, the objective functions of the MAD and MMD criteria are described in Sections 3.5.1 and 3.5.2, respectively.

The last column of the tables presents the results of the Wilcoxon rank sum hypothesis tests [58, 68], having set the significance level at 1%. More specifically, the obtained values of these tests can be either 1 or 0. A value equal to 1 indicates rejection of the null hypothesis at the 1% significance level, while a value equal to 0 indicates a failure to reject the null hypothesis at the 1% significance level. For each case of nodes' distribution, PSO has been compared with the respective case of the AS, IP and TRR algorithms, and the results of the two-sided rank sum tests have been reported under the "Ranksum" column, next to AS, IP and TRR algorithms, respectively. For example, for the case of $R_{\text{target}} = 96$ kbps and $W_t = 20$ MHz, for the NNBS criterion, having "1" under the "Ranksum" column to Case 1 of AS means that the test rejects the null hypothesis of equal medians for the 30 values of the NNBS function using the PSO and the 30 values of the NNBS function using the AS, at 1% significance level.

Observing the successes of each optimization method over the 30 experiments for all considered bitrate and bandwidth combinations, we can see that PSO far exceeds the other methods, where in many cases its success rates is 100%. The great advantage of PSO compared to the other methods is more obvious in the MMD criterion. While PSO's

Table 3.14: Experimental results using PSO for $R_{\text{target}} = 96$ kbps and $W_t = 20$ MHz.

	Sequences	NNBS			CNBS			MAD			MMD		
		S	(R_s, R_c)	PSNR	S	(R_s, R_c)	PSNR	S	(R_s, R_c)	PSNR	S	(R_s, R_c)	PSNR
Case 1	"Akiyo"	5.0000	(32,1/3)	36.5112	5.0000	(32,1/3)	36.5112	5.0000	(32,1/3)	36.2808	5.0000	(32,1/3)	37.0690
	"Mother&Daughter"	6.1970	(32,1/3)	34.5452	6.1970	(32,1/3)	34.5452	6.3500	(32,1/3)	34.4621	5.0000	(32,1/3)	33.7349
	"Salesman"	5.8840	(32,1/3)	33.0986	5.8840	(32,1/3)	33.0986	6.2000	(32,1/3)	33.1756	5.2662	(32,1/3)	32.9946
	"Hall"	6.8140	(32,1/3)	33.8855	6.8140	(32,1/3)	33.8855	7.0500	(32,1/3)	33.8988	5.4199	(32,1/3)	32.9946
	"Foreman"	14.0260	(48,1/2)	33.5774	14.0260	(48,1/2)	33.5774	14.6000	(48,1/2)	33.6466	12.2326	(48,1/2)	32.9946
	"Coastguard"	14.4270	(48,1/2)	31.7825	14.4270	(48,1/2)	31.7825	15.0000	(48,1/2)	31.8157	15.0000	(64,2/3)	32.9946
	Total	**52.3480**		**203.4004**	**52.3480**		**203.4004**	**54.2118**		**203.2796**	**47.9187**		**202.7823**
Case 2	"Akiyo"	5.0000	(32,1/3)	36.9809	5.0000	(32,1/3)	36.9424	5.0000	(32,1/3)	36.9572	5.0000	(38.4,4/10)	35.0808
	"Mother&Daughter"	6.1387	(32,1/3)	34.9393	5.0555	(32,1/3)	33.6715	6.1150	(32,1/3)	34.8931	5.0000	(32,1/3)	34.2562
	"Salesman"	5.8296	(32,1/3)	33.3904	6.2451	(32,1/3)	33.6637	6.0468	(32,1/3)	33.5344	5.8396	(32,1/3)	33.8069
	"Hall"	6.7451	(32,1/3)	34.2718	8.3800	(38.4,4/10)	35.8310	6.8460	(32,1/3)	34.3374	5.6838	(32,1/3)	33.8069
	"Foreman"	14.5715	(64,2/3)	35.0184	15.0000	(64,2/3)	35.3444	14.5421	(64,2/3)	34.9341	12.0802	(48,1/2)	33.8069
	"Coastguard"	15.0000	(64,2/3)	32.8463	15.0000	(64,2/3)	32.7816	15.0000	(64,2/3)	32.8065	15.0000	(64,2/3)	33.8069
	Total	**53.2849**		**207.4471**	**54.6806**		**208.2346**	**53.5499**		**207.4627**	**48.6036**		**204.5646**
Case 3	"Akiyo"	5.0000	(32,1/3)	36.9558	5.2132	(32,1/3)	37.4139	5.0000	(32,1/3)	36.9399	5.0000	(32,1/3)	37.4428
	"Mother&Daughter"	6.1139	(32,1/3)	34.8905	6.4668	(32,1/3)	35.4221	6.0062	(32,1/3)	34.7674	5.0000	(32,1/3)	34.1410
	"Salesman"	5.8058	(32,1/3)	33.3543	5.0000	(32,1/3)	32.8240	5.9207	(32,1/3)	33.4298	5.7001	(32,1/3)	33.6265
	"Hall"	6.7184	(32,1/3)	34.2239	5.4524	(32,1/3)	33.1350	6.7194	(32,1/3)	34.2095	5.6214	(32,1/3)	33.6265
	"Foreman"	13.6258	(48,1/2)	34.0659	14.9177	(64,2/3)	35.7753	13.6744	(48,1/2)	34.0777	12.1153	(48,1/2)	33.6265
	"Coastguard"	15.0000	(64,2/3)	32.8041	15.0000	(64,2/3)	33.1541	15.0000	(64,2/3)	32.7774	15.0000	(64,2/3)	33.6265
	Total	**52.2636**		**206.2945**	**52.0501**		**207.7244**	**52.3207**		**206.2017**	**48.4368**		**206.0898**
Case 4	"Akiyo"	5.0000	(32,1/3)	35.8036	5.0000	(32,1/3)	37.0979	5.0000	(32,1/3)	35.6996	5.0000	(32,1/3)	36.0568
	"Mother&Daughter"	6.1890	(32,1/3)	33.8701	6.2010	(32,1/3)	35.1134	6.3910	(32,1/3)	33.9843	5.0000	(32,1/3)	32.6348
	"Salesman"	5.8750	(32,1/3)	32.5969	5.8895	(32,1/3)	33.5191	6.1889	(32,1/3)	32.7776	5.0000	(32,1/3)	31.9587
	"Hall"	6.8130	(32,1/3)	33.2280	7.9245	(38.4,4/10)	35.5853	7.1161	(32,1/3)	33.4128	5.1498	(32,1/3)	31.5385
	"Foreman"	14.4575	(48,1/2)	32.6798	9.4885	(32,1/3)	30.7193	15.0000	(48,1/2)	32.9506	12.6278	(32,1/3)	31.5385
	"Coastguard"	12.6450	(32,1/3)	30.1005	9.9345	(32,1/3)	29.8901	12.2593	(32,1/3)	29.8803	15.0000	(48,1/2)	31.5385
	Total	**50.9795**		**198.2789**	**44.4380**		**201.9251**	**51.9553**		**198.7052**	**47.7776**		**195.2658**

Table 3.15: Experimental results using PSO for $R_{\text{target}} = 96$ kbps and $W_t = 15$ MHz.

	Sequences	NNBS			CNBS			MAD			MMD		
		S	(R_s, R_c)	PSNR	S	(R_s, R_c)	PSNR	S	(R_s, R_c)	PSNR	S	(R_s, R_c)	PSNR
Case 1	"Akiyo"	5.0000	(32,1/3)	35.6945	5.0000	(32,1/3)	35.6945	5.0000	(32,1/3)	35.0168	5.0000	(32,1/3)	35.2312
	"Mother&Daughter"	6.0230	(32,1/3)	33.5979	6.0230	(32,1/3)	33.5979	6.4010	(32,1/3)	33.3981	5.0000	(32,1/3)	31.7377
	"Salesman"	5.5620	(32,1/3)	32.2508	5.5620	(32,1/3)	32.2508	6.1295	(32,1/3)	32.2877	5.0000	(32,1/3)	31.3134
	"Hall"	6.5455	(32,1/3)	32.8823	6.5455	(32,1/3)	32.8823	7.0970	(32,1/3)	32.8212	5.1860	(32,1/3)	30.6941
	"Foreman"	10.8655	(32,1/3)	30.2644	10.8655	(32,1/3)	30.2644	12.2990	(32,1/3)	30.4978	12.3134	(32,1/3)	30.6941
	"Coastguard"	10.6320	(32,1/3)	29.2214	10.6320	(32,1/3)	29.2214	11.9025	(32,1/3)	29.3379	15.0000	(48,1/2)	30.6941
	Total	**44.6280**		**193.9113**	**44.6280**		**193.9113**	**48.8290**		**193.3595**	**47.4994**		**190.3646**
Case 2	"Akiyo"	5.0000	(32,1/3)	36.1434	5.0000	(32,1/3)	35.3258	5.0000	(32,1/3)	35.2539	5.0000	(32,1/3)	35.8514
	"Mother&Daughter"	6.0360	(32,1/3)	34.0389	5.4467	(32,1/3)	32.5165	6.3586	(32,1/3)	33.5627	5.0000	(32,1/3)	32.4116
	"Salesman"	5.5710	(32,1/3)	32.5823	6.4488	(32,1/3)	32.7412	6.1069	(32,1/3)	32.4249	5.0000	(32,1/3)	31.7982
	"Hall"	6.5600	(32,1/3)	33.3179	7.5948	(32,1/3)	33.5278	7.0575	(32,1/3)	32.9887	5.2294	(32,1/3)	31.4334
	"Foreman"	10.8100	(32,1/3)	30.6672	15.0000	(48,1/2)	32.5214	15.0000	(48,1/2)	32.4018	12.6330	(32,1/3)	31.4334
	"Coastguard"	10.5830	(32,1/3)	29.5215	12.2136	(32,1/3)	29.6692	11.9233	(32,1/3)	29.5011	15.0000	(48,1/2)	31.4334
	Total	**44.5600**		**196.2712**	**51.7039**		**196.3019**	**51.4463**		**196.1331**	**47.8624**		**194.3614**
Case 3	"Akiyo"	5.0000	(32,1/3)	36.0155	5.5936	(32,1/3)	36.3045	5.0000	(32,1/3)	35.2806	5.0000	(32,1/3)	35.8273
	"Mother&Daughter"	6.0325	(32,1/3)	33.9130	6.7573	(32,1/3)	34.1980	6.4400	(32,1/3)	33.6732	5.0000	(32,1/3)	32.3854
	"Salesman"	5.5685	(32,1/3)	32.4877	5.0000	(32,1/3)	31.5356	6.1980	(32,1/3)	32.5171	5.0000	(32,1/3)	31.7793
	"Hall"	6.5560	(32,1/3)	33.1933	5.7320	(32,1/3)	31.7751	7.1535	(32,1/3)	33.1013	5.2272	(32,1/3)	31.4039
	"Foreman"	10.8230	(32,1/3)	30.5507	15.0000	(48,1/2)	32.8400	12.4530	(32,1/3)	30.8146	12.6201	(32,1/3)	31.4039
	"Coastguard"	10.5970	(32,1/3)	29.4348	11.8194	(32,1/3)	29.6321	12.1445	(32,1/3)	29.6109	15.0000	(48,1/2)	31.4039
	Total	**44.5770**		**195.5950**	**49.9023**		**196.2853**	**49.3890**		**194.9977**	**47.8473**		**194.2037**
Case 4	"Akiyo"	5.0000	(32,1/3)	34.9081	5.0000	(32,1/3)	35.8943	5.0000	(32,1/3)	34.3861	5.0000	(32,1/3)	33.7426
	"Mother&Daughter"	5.9965	(32,1/3)	32.8337	6.0290	(32,1/3)	33.7938	6.3065	(32,1/3)	32.7405	5.0000	(32,1/3)	30.1199
	"Salesman"	5.5440	(32,1/3)	31.6736	5.5665	(32,1/3)	32.3982	5.9745	(32,1/3)	31.7394	5.0000	(32,1/3)	30.1496
	"Hall"	6.5155	(32,1/3)	32.1320	6.5520	(32,1/3)	33.0756	6.9585	(32,1/3)	32.1522	5.5550	(32,1/3)	29.7558
	"Foreman"	10.9590	(32,1/3)	29.5999	8.8385	(32,1/3)	29.0024	11.9135	(32,1/3)	29.7447	12.8151	(32,1/3)	29.7558
	"Coastguard"	10.7140	(32,1/3)	28.7254	8.6335	(32,1/3)	28.2780	11.3330	(32,1/3)	28.6895	15.0000	(32,1/3)	29.7558
	Total	**44.7290**		**189.8727**	**40.6195**		**192.4423**	**47.4860**		**189.4522**	**48.3701**		**183.2795**

Table 3.16: Experimental results using PSO for $R_{\text{target}} = 144$ kbps and $W_t = 30$ MHz.

		NNBS			CNBS			MAD			MMD		
	Sequences	S	(R_s, R_c)	PSNR	S	(R_s, R_c)	PSNR	S	(R_s, R_c)	PSNR	S	(R_s, R_c)	PSNR
Case 1	"Akiyo"	5.0000	(48,1/3)	36.4608	5.0000	(48,1/3)	36.4608	5.0000	(48,1/3)	35.9137	5.0000	(48,1/3)	34.5798
	"Mother&Daughter"	6.0859	(48,1/3)	34.5607	6.0889	(48,1/3)	34.5607	6.3540	(48,1/3)	34.5109	6.3054	(48,1/3)	33.2093
	"Salesman"	6.5475	(48,1/3)	33.6832	6.5475	(48,1/3)	33.6832	6.9400	(48,1/3)	33.7930	7.3095	(48,1/3)	33.2093
	"Hall"	6.2033	(48,1/3)	34.0215	6.2033	(48,1/3)	34.0215	6.5390	(48,1/3)	34.0764	6.7042	(32,1/3)	33.2093
	"Foreman"	10.9137	(72,1/2)	33.5108	10.9137	(72,1/2)	33.5108	11.5525	(72,1/2)	33.6666	12.5330	(48,1/3)	33.2093
	"Coastguard"	10.4261	(48,1/3)	30.9488	10.4261	(48,1/3)	30.9488	10.9800	(48,1/3)	30.9864	15.0000	(48,1/3)	33.2093
	Total	45.1795		203.1858	45.1795		203.1858	47.3655		202.9470	52.8521		200.6263
Case 2	"Akiyo"	5.0000	(48,1/3)	36.9372	5.0000	(48,1/3)	35.3652	5.0000	(48,1/3)	36.3332	5.0000	(48,1/3)	35.5691
	"Mother&Daughter"	6.0765	(48,1/3)	34.9906	6.1183	(48,1/3)	33.5877	6.3770	(48,1/3)	34.9365	6.4225	(48,1/3)	34.3115
	"Salesman"	6.5275	(48,1/3)	34.0740	8.1458	(48,1/3)	34.9117	6.9825	(48,1/3)	34.2124	7.5287	(48,1/3)	34.3115
	"Hall"	6.1895	(48,1/3)	34.4212	7.7352	(48,1/3)	35.2744	6.5760	(48,1/3)	34.4928	6.8920	(48,1/3)	34.3115
	"Foreman"	10.7665	(72,1/2)	33.9734	13.1513	(72,1/2)	34.9981	11.4865	(72,1/2)	34.1553	12.3809	(72,1/2)	34.3115
	"Coastguard"	11.8530	(72,1/2)	32.7052	14.4449	(72,1/2)	33.5649	12.8185	(72,1/2)	33.0510	15.0000	(72,1/2)	34.3115
	Total	46.4130		207.1016	54.5955		207.7020	49.2405		207.1812	53.2241		207.1266
Case 3	"Akiyo"	5.0000	(48,1/3)	36.9433	5.0000	(48,1/3)	37.6623	5.0000	(48,1/3)	35.9637	5.0000	(48,1/3)	35.1208
	"Mother&Daughter"	6.0765	(48,1/3)	34.9962	6.0590	(48,1/3)	35.6522	6.3570	(48,1/3)	34.5617	6.3659	(48,1/3)	33.8065
	"Salesman"	6.5275	(48,1/3)	34.0791	5.2935	(48,1/3)	32.6142	6.9450	(48,1/3)	33.8430	7.4243	(48,1/3)	33.8065
	"Hall"	6.1890	(48,1/3)	34.4263	5.0040	(48,1/3)	32.9227	6.5435	(48,1/3)	34.1261	6.8021	(48,1/3)	33.8065
	"Foreman"	10.7650	(72,1/2)	33.9795	10.5535	(72,1/2)	34.7077	11.5450	(72,1/2)	33.7248	12.4494	(72,1/2)	33.8065
	"Coastguard"	10.3110	(48,1/3)	31.2108	11.5990	(72,1/2)	33.3202	12.8355	(72,1/2)	32.6268	15.0000	(72,1/2)	33.8065
	Total	44.8690		205.6352	43.5090		206.8793	49.2260		204.8461	53.0417		204.1533
Case 4	"Akiyo"	5.0000	(48,1/3)	35.4595	5.0000	(48,1/3)	36.6970	5.0000	(48,1/3)	35.1774	5.0000	(48,1/3)	32.9847
	"Mother&Daughter"	6.1155	(48,1/3)	33.6705	6.0830	(48,1/3)	34.7734	6.3125	(48,1/3)	33.7643	6.2398	(48,1/3)	31.6621
	"Salesman"	6.5910	(48,1/3)	32.8784	6.5375	(48,1/3)	33.8763	6.8640	(48,1/3)	33.0576	7.1215	(48,1/3)	31.6621
	"Hall"	6.2340	(48,1/3)	33.1952	6.1965	(48,1/3)	34.2192	6.4740	(48,1/3)	33.3461	6.5542	(48,1/3)	31.6621
	"Foreman"	10.3350	(48,1/3)	31.6048	8.7005	(48,1/3)	30.8078	10.7305	(48,1/3)	31.7761	12.5792	(48,1/3)	31.6621
	"Coastguard"	10.6795	(48,1/3)	30.4360	9.0040	(48,1/3)	29.8923	10.7025	(48,1/3)	30.2586	15.0000	(48,1/3)	31.6621
	Total	44.9550		197.24444	41.5215		200.2660	46.0835		197.3801	52.4947		191.2952

Table 3.17: Experimental results using PSO for $R_{\text{target}} = 144$ kbps and $W_t = 22.5$ MHz.

		NNBS			CNBS			MAD			MMD		
	Sequences	S	(R_s, R_c)	PSNR	S	(R_s, R_c)	PSNR	S	(R_s, R_c)	PSNR	S	(R_s, R_c)	PSNR
Case 1	"Akiyo"	5.0000	(48,1/3)	34.1645	5.0000	(48,1/3)	34.1645	5.0000	(48,1/3)	33.1807	5.0000	(48,1/3)	30.1623
	"Mother&Daughter"	5.8885	(48,1/3)	32.0453	5.8885	(48,1/3)	32.0453	6.1850	(48,1/3)	31.7519	6.6545	(48,1/3)	30.1623
	"Salesman"	6.1985	(48,1/3)	31.1217	6.1985	(48,1/3)	31.1217	6.6445	(48,1/3)	31.0805	7.4745	(48,1/3)	30.1623
	"Hall"	5.9185	(48,1/3)	31.5210	5.9185	(48,1/3)	31.5210	6.2885	(48,1/3)	31.3786	6.9120	(48,1/3)	30.1623
	"Foreman"	9.4195	(48,1/3)	29.2876	9.4195	(48,1/3)	29.2876	10.4010	(48,1/3)	29.6193	13.2165	(48,1/3)	30.1623
	"Coastguard"	8.8895	(48,1/3)	28.0057	8.8895	(48,1/3)	28.0057	9.9280	(48,1/3)	28.3462	14.8490	(48,1/3)	30.1623
	Total	41.3145		186.1458	41.3145		186.1458	44.4470		185.3572	54.1065		180.9738
Case 2	"Akiyo"	5.0000	(48,1/3)	34.8552	5.0000	(48,1/3)	33.1191	5.0000	(48,1/3)	33.8512	5.0000	(48,1/3)	31.7686
	"Mother&Daughter"	5.8840	(48,1/3)	32.6833	5.8950	(48,1/3)	31.0925	6.2265	(48,1/3)	32.4322	6.4705	(48,1/3)	31.0979
	"Salesman"	6.1905	(48,1/3)	31.7107	7.7055	(48,1/3)	32.6571	6.7150	(48,1/3)	31.7506	7.3337	(48,1/3)	31.0979
	"Hall"	5.9135	(48,1/3)	32.1199	7.3670	(48,1/3)	33.0788	6.3480	(48,1/3)	32.0442	6.7650	(48,1/3)	31.0979
	"Foreman"	9.3350	(48,1/3)	29.8588	11.4845	(48,1/3)	30.8089	10.5020	(48,1/3)	30.3640	12.8977	(48,1/3)	31.0979
	"Coastguard"	8.8210	(48,1/3)	28.4254	10.8730	(48,1/3)	29.1200	10.1715	(48,1/3)	29.0056	15.0000	(48,1/3)	31.0979
	Total	41.1440		189.6533	48.2950		189.8764	44.9630		189.4478	53.4669		187.2581
Case 3	"Akiyo"	5.0000	(48,1/3)	34.5609	5.2395	(48,1/3)	35.7362	5.0000	(48,1/3)	33.5194	5.0000	(48,1/3)	31.0367
	"Mother&Daughter"	5.8860	(48,1/3)	32.4107	6.1590	(48,1/3)	33.5056	6.2060	(48,1/3)	32.0955	6.5748	(48,1/3)	30.7007
	"Salesman"	6.1940	(48,1/3)	31.4588	5.2415	(48,1/3)	30.0895	6.6800	(48,1/3)	31.4189	7.4234	(48,1/3)	30.7007
	"Hall"	5.9155	(48,1/3)	31.8640	5.0000	(48,1/3)	30.4669	6.3185	(48,1/3)	31.7148	6.8547	(48,1/3)	30.7007
	"Foreman"	9.3705	(48,1/3)	29.6126	9.6730	(48,1/3)	30.6207	10.4530	(48,1/3)	29.9952	13.0894	(48,1/3)	30.7007
	"Coastguard"	8.8500	(48,1/3)	28.2447	8.7360	(48,1/3)	28.9827	10.0510	(48,1/3)	28.6790	15.0000	(48,1/3)	30.7007
	Total	41.2160		188.1517	40.0490		189.4016	44.7085		187.4228	53.9423		184.5402
Case 4	"Akiyo"	5.0000	(48,1/3)	33.0309	5.0000	(48,1/3)	34.4020	5.0000	(48,1/3)	32.2495	5.0000	(48,1/3)	28.7813
	"Mother&Daughter"	5.8955	(48,1/3)	31.0128	5.8870	(48,1/3)	32.2639	6.1265	(48,1/3)	30.8076	6.5060	(48,1/3)	28.7813
	"Salesman"	6.2120	(48,1/3)	30.1727	6.1960	(48,1/3)	31.3233	6.5465	(48,1/3)	30.1503	7.2135	(48,1/3)	28.7813
	"Hall"	5.9260	(48,1/3)	30.5520	5.9170	(48,1/3)	31.7262	6.2070	(48,1/3)	30.4547	6.6985	(48,1/3)	28.7813
	"Foreman"	9.5655	(48,1/3)	28.4022	8.0570	(48,1/3)	27.4835	10.2450	(48,1/3)	28.5880	12.8005	(48,1/3)	28.7813
	"Coastguard"	9.0075	(48,1/3)	27.3516	7.5680	(48,1/3)	26.6678	9.5950	(48,1/3)	27.4337	13.6900	(48,1/3)	28.7813
	Total	41.6065		180.5222	38.6250		183.8667	43.7200		179.6838	51.9085		172.6878

success rates are in many cases 100%, the other methods fail nearly always to reach the optimal solution. However, there are a few cases where PSO's successes are less than 30. In these cases, if we examine the other statistic values of the tables, we will observe that the min value of the function differs from the max value (of the 30 values) in the third, second or first decimal digit. This claim is also confirmed by the small values of the standard deviation or by the fact that the min function value is equal to the median function value or have a slight difference in the fourth or third decimal digit. However, even in cases where PSO achieves a near-optimal solution, this solution is acceptable in practice, since it has only a slight impact on the utilities achieved by the nodes. Thus, all this statistical information reinforces our view about the efficiency of PSO in solving such optimization tasks.

Also, PSO, AS and IP behave better with the MAD criterion, noting better performance. As it has been previously referred, PSO far exceeds the other competing methods, being able to nearly always reach the optimal solution. Among the benchmarks that we have used for comparison with the PSO, the IP algorithm is the most efficient one, followed in performance by the AS, and finally, by the TRR, which fails always (in all examined cases) to reach the optimal solution.

The considerably low success rates of the deterministic algorithms can be probably attributed to the shape of the corresponding objective functions. Specifically, if they include steep hills as well as large flat areas, this can trap the deterministic approaches if they are initialized within these regions. This means that the selection of the starting point is very important for the performance of each method. For example, if the functions are flat in a large part, a starting point in this area does not lead any of the three above methods to find the optimal solution. This fact motivated us to use the PSO algorithm as the optimization solver in this set of experiments.

Table 3.18: Statistical results for PSO, AS, IP, TRR, $R_{\text{target}} = 96$ kbps and $W_t = 20$ MHz.

Criterion	Algorithm	Case	Success	Min	Mean	Max	Median	Std	Ranksum
NNBS	PSO	1	30	-5.726552	-5.726552	-5.726552	-5.726552	0.00	–
		2	30	-6.759974	-6.759974	-6.759974	-6.759974	0.00	–
		3	29	-6.138522	-6.138506	-6.138032	-6.138522	0.00	–
		4	24	-4.275103	-4.272522	-4.232132	-4.275103	0.01	–
	AS	1	4	-5.726552	-0.121092	1.000000	1.000000	2.55	1
		2	2	-6.759974	0.482668	1.000000	1.000000	1.97	1
		3	4	-6.138522	0.048197	1.000000	1.000000	2.47	1
		4	7	-4.275103	-0.582531	1.000000	1.000000	2.46	1
	IP	1	6	-5.726552	-0.345310	1.000000	1.000000	2.74	1
		2	1	-6.759974	0.482668	1.000000	1.000000	1.97	1
		3	3	-6.138522	0.286148	1.000000	1.000000	2.18	1
		4	9	-4.275103	-0.582531	1.000000	1.000000	2.46	1
	TRR	1	0	-5.330930	-0.213350	1.000000	1.000000	2.28	1
		2	0	-5.908247	0.565973	1.000000	1.000000	1.66	1
		3	0	-4.873453	0.277376	1.000000	1.000000	1.88	1
		4	0	-3.367513	-0.163746	1.000000	1.000000	1.82	1
CNBS	PSO	1	30	-5.726552	-5.726552	-5.726552	-5.726552	0.00	–
		2	30	-6.550393	-6.550393	-6.550393	-6.550393	0.00	–
		3	30	-6.409900	-6.409900	-6.409900	-6.409900	0.00	–
		4	30	-4.904941	-4.904941	-4.904941	-4.904941	0.00	–
	AS	1	3	-5.726552	0.327345	1.000000	1.000000	2.05	1
		2	2	-6.550393	0.496640	1.000000	1.000000	1.92	1
		3	1	-6.409900	0.259011	1.000000	1.000000	2.26	1
		4	7	-4.904941	-2.149301	1.000000	-4.904939	3.00	1
	IP	1	4	-5.726552	0.103126	1.000000	1.000000	2.33	1
		2	0	-6.550392	0.748320	1.000000	1.000000	1.38	1
		3	1	-6.409900	0.259010	1.000000	1.000000	2.26	1
		4	14	-4.904941	-1.755639	1.000000	1.000000	3.00	1
	TRR	1	0	-4.964087	0.515564	1.000000	1.000000	1.51	1
		2	0	1.000000	1.000000	1.000000	1.000000	0.00	1
		3	0	-5.078231	0.442476	1.000000	1.000000	1.71	1
		4	0	-4.420913	-0.777876	1.000000	1.000000	2.41	1
MAD	PSO	1	29	27.877986	27.903443	28.641710	27.877986	0.14	–
		2	30	22.105961	22.105961	22.105961	22.105961	0.00	–
		3	30	25.335557	25.335557	25.335557	25.335557	0.00	–
		4	28	38.152870	38.195643	38.921273	38.152872	0.17	–
	AS	1	30	27.877986	27.877986	27.877986	27.877986	0.00	0
		2	30	22.105961	22.105961	22.105961	22.105961	0.00	0
		3	14	25.335557	27.840128	100.472675	25.335558	13.72	1
		4	30	38.152872	38.152872	38.152872	38.152872	0.00	0
	IP	1	27	27.877986	27.877986	27.877987	27.877986	0.00	0
		2	30	22.105961	22.105961	22.105961	22.105961	0.00	0
		3	14	25.335557	25.335558	25.335558	25.335558	0.00	1
		4	30	38.152872	38.152872	38.152872	38.152872	0.00	0
	TRR	1	0	35.459611	84.873294	590.658891	55.698445	105.48	1
		2	0	29.444823	31922142.648722	957609654.693354	198.978964	174834459.25	1
		3	0	29.759017	649.290198	13790.047768	66.250344	2528.78	1
		4	0	45.894814	97.197020	428.296977	60.885126	103.30	1
MMD	PSO	1	26	32.630183	32.681779	33.238361	32.630183	0.16	–
		2	30	27.063638	27.063638	27.063638	27.063638	0.00	–
		3	20	28.211844	28.375334	29.236985	28.211844	0.33	–
		4	29	45.627729	45.673158	46.990611	45.627729	0.25	–
	AS	1	0	32.630184	32.727708	33.837196	32.630313	0.28	1
		2	0	27.063640	27.214074	28.781947	27.064162	0.40	1
		3	0	28.211845	28.277313	29.324319	28.214876	0.21	1
		4	1	45.627729	45.627799	45.628617	45.627735	0.00	1
	IP	1	0	32.893982	32.921179	33.593664	32.894431	0.13	1
		2	0	27.286547	27.343641	27.381377	27.346057	0.01	1
		3	0	28.475167	28.483026	28.513347	28.482314	0.01	1
		4	0	45.992175	46.004366	46.080940	46.001779	0.01	1
	TRR	1	0	63.809690	1473.410378	11798.985887	735.239348	2557.98	1
		2	0	50.968639	871.311987	3926.881004	561.712215	990.35	1
		3	0	54.714794	5969.861550	162460.812440	324.760820	29562.79	1
		4	0	63.999563	166.903248	491.288336	128.784483	96.83	1

Table 3.19: Statistical results for PSO, AS, IP, TRR, $R_{\text{target}} = 96$ kbps and $W_t = 15$ MHz.

Criterion	Algorithm	Case	Success	Min	Mean	Max	Median	Std	Ranksum
NNBS	PSO	1	30	-5.931049	-5.931049	-5.931049	-5.931049	0.00	–
		2	30	-6.884919	-6.884919	-6.884919	-6.884919	0.00	–
		3	29	-6.357387	-6.357375	-6.357014	-6.357387	0.00	–
		4	30	-4.644677	-4.644677	-4.644677	-4.644677	0.00	–
	AS	1	10	-5.931049	-1.310350	1.000000	1.000000	3.32	1
		2	11	-6.884919	-2.679628	1.000000	1.000000	4.00	1
		3	2	-6.357387	-1.942954	1.000000	1.000000	3.67	1
		4	2	-4.644677	-2.386805	1.000000	-4.644673	2.81	1
	IP	1	14	-5.931049	-2.234490	1.000000	1.000000	3.52	1
		2	13	-6.884919	-2.416798	1.000000	1.000000	3.97	1
		3	18	-6.357387	-3.414432	1.000000	-6.357387	3.67	1
		4	11	-4.644677	-1.069715	1.000000	1.000000	2.77	1
	TRR	1	0	-5.622618	-2.655088	1.000000	-4.322057	2.89	1
		2	0	-6.486487	-1.237290	1.000000	1.000000	3.24	1
		3	0	-6.035897	-2.238168	1.000000	-3.305525	3.14	1
		4	0	-4.130423	-1.386533	1.000000	-2.321361	2.17	1
CNBS	PSO	1	30	-5.931049	-5.931049	-5.931049	-5.931049	0.00	–
		2	19	-6.476440	-6.466387	-6.449022	-6.476440	0.01	–
		3	16	-6.374505	-6.357740	-6.338579	-6.374505	0.02	–
		4	30	-5.366147	-5.366147	-5.366147	-5.366147	0.00	–
	AS	1	8	-5.931049	-1.541384	1.000000	1.000000	3.40	1
		2	6	-6.476440	-0.744503	1.000000	1.000000	3.22	1
		3	5	-6.374505	-0.229084	1.000000	1.000000	2.80	1
		4	5	-5.366147	-1.970868	1.000000	1.000000	3.23	1
	IP	1	16	-5.931049	-2.696559	1.000000	-5.931049	3.52	1
		2	4	-6.476440	0.003141	1.000000	1.000000	2.58	1
		3	9	-6.374505	-1.212352	1.000000	1.000000	3.44	1
		4	14	-5.366147	-1.970869	1.000000	1.000000	3.23	1
	TRR	1	0	-5.497558	-2.238326	1.000000	-3.878174	2.92	1
		2	0	-5.926976	-0.693331	1.000000	1.000000	2.67	1
		3	0	-5.030588	0.261632	1.000000	1.000000	1.92	1
		4	0	-5.104637	-1.385651	1.000000	1.000000	2.80	1
MAD	PSO	1	30	42.715334	42.715334	42.715334	42.715334	0.00	–
		2	18	34.740266	34.741492	34.743331	34.740266	0.00	–
		3	30	38.678888	38.678888	38.678888	38.678888	0.00	–
		4	30	56.482854	56.482854	56.482854	56.482854	0.00	–
	AS	1	8	42.715334	42.715335	42.715336	42.715335	0.00	1
		2	27	34.740266	34.740266	34.740267	34.740266	0.00	1
		3	17	38.678888	38.678888	38.678889	38.678888	0.00	1
		4	8	56.482854	56.482855	56.482855	56.482855	0.00	1
	IP	1	30	42.715334	42.715334	42.715334	42.715334	0.00	0
		2	11	34.740266	34.740267	34.740267	34.740267	0.00	0
		3	30	38.678888	38.678888	38.678888	38.678888	0.00	0
		4	30	56.482854	56.482854	56.482854	56.482854	0.00	0
	TRR	1	0	48.597923	67.903578	124.637931	59.695320	21.60	1
		2	0	41.072661	2562336.995613	76818946.290372	69.551001	14024836.44	1
		3	0	41.140134	55.932004	84.065159	55.987149	9.81	1
		4	0	59.823066	84.709830	142.269987	81.943355	21.17	1
MMD	PSO	1	26	55.420774	55.516563	56.139190	55.420774	0.25	–
		2	30	46.745971	46.745971	46.745971	46.745971	0.00	–
		3	27	47.064096	47.260824	50.356854	47.064096	0.67	–
		4	30	68.786486	68.786486	68.786486	68.786486	0.00	–
	AS	1	0	55.420775	213734.132136	6410416.730469	55.420789	1170366.50	1
		2	2	46.745971	427404.773826	6410416.730469	46.746009	1626362.98	1
		3	0	47.064097	47.076131	47.414143	47.064110	0.06	1
		4	0	68.786488	68.954052	72.913638	68.786535	0.75	1
	IP	1	0	55.788466	213734.511989	6410416.730469	55.809086	1170366.43	1
		2	0	47.116362	47.135636	47.155817	47.136172	0.01	1
		3	0	47.373010	47.450356	47.459722	47.452354	0.01	1
		4	0	69.178991	69.180494	69.198662	69.179764	0.00	1
	TRR	1	0	93.225285	214326.061791	6410416.730469	337.734698	1170255.19	1
		2	0	78.373678	4090.934508	89078.591611	343.627444	16255.22	1
		3	0	105.814688	433327.625626	6410416.730469	279.925460	1624953.58	1
		4	0	96.411897	181.725259	339.864507	149.604006	71.05	1

Table 3.20: Statistical results for PSO, AS, IP, TRR, $R_{\text{target}} = 144$ kbps and $W_t = 30$ MHz.

Criterion	Algorithm	Case	Success	Min	Mean	Max	Median	Std	Ranksum
NNBS	PSO	1	28	-5.600859	-5.599407	-5.579073	-5.600859	0.01	–
		2	22	-6.723017	-6.719601	-6.710207	-6.723017	0.01	–
		3	30	-6.062459	-6.062459	-6.062459	-6.062459	0.00	–
		4	25	-4.144447	-4.121988	-3.845342	-4.144447	0.07	–
	AS	1	5	-5.600859	-0.760229	1.000000	1.000000	2.97	1
		2	0	-6.723015	0.742566	1.000000	1.000000	1.41	1
		3	3	-6.062459	-0.177076	1.000000	1.000000	2.68	1
		4	0	-4.144446	-0.200370	1.000000	1.000000	2.21	1
	IP	1	10	-5.600859	-1.200286	1.000000	1.000000	3.16	1
		2	1	-6.723017	0.742566	1.000000	1.000000	1.41	1
		3	5	-6.062459	-0.177076	1.000000	1.000000	2.68	1
		4	7	-4.144447	-0.200371	1.000000	1.000000	2.21	1
	TRR	1	0	-5.020738	0.461282	1.000000	1.000000	1.65	1
		2	0	-4.769009	0.807700	1.000000	1.000000	1.05	1
		3	0	-5.326840	0.004166	1.000000	1.000000	2.27	1
		4	0	-3.502565	0.078349	1.000000	1.000000	1.71	1
CNBS	PSO	1	28	-5.600859	-5.599407	-5.579073	-5.600859	0.01	–
		2	30	-6.572136	-6.572136	-6.572136	-6.572136	0.00	–
		3	29	-6.255745	-6.253124	-6.177124	-6.255745	0.01	–
		4	25	-4.746744	-4.729444	-4.584500	-4.746744	0.05	–
	AS	1	5	-5.600859	-0.760229	1.000000	1.000000	2.97	1
		2	1	-6.572136	0.747595	1.000000	1.000000	1.38	1
		3	1	-6.255745	0.274426	1.000000	1.000000	2.21	1
		4	4	-4.746744	-0.340907	1.000000	1.000000	2.47	1
	IP	1	10	-5.600859	-1.200286	1.000000	1.000000	3.16	1
		2	1	-6.572136	0.747595	1.000000	1.000000	1.38	1
		3	3	-6.255745	0.274425	1.000000	1.000000	2.21	1
		4	7	-4.746744	-0.340907	1.000000	1.000000	2.47	1
	TRR	1	0	-5.020738	0.461282	1.000000	1.000000	1.65	1
		2	0	-3.896261	0.836791	1.000000	1.000000	0.89	1
		3	0	-4.803903	0.481003	1.000000	1.000000	1.59	1
		4	0	-4.448528	-0.132917	1.000000	1.000000	2.10	1
MAD	PSO	1	30	28.671136	28.671136	28.671136	28.671136	0.00	–
		2	29	22.197505	22.203320	22.371955	22.197505	0.03	–
		3	19	25.781431	25.795943	25.821008	25.781431	0.02	–
		4	30	39.537574	39.537574	39.537574	39.537574	0.00	–
	AS	1	29	28.671136	28.671136	28.671137	28.671136	0.00	0
		2	30	22.197505	22.197505	22.197505	22.197505	0.00	0
		3	23	25.781431	25.781431	25.781432	25.781431	0.00	0
		4	2	39.537574	39.537575	39.537575	39.537575	0.00	1
	IP	1	30	28.671136	28.671136	28.671136	28.671136	0.00	0
		2	30	22.197505	22.197505	22.197505	22.197505	0.00	0
		3	30	25.781431	25.781431	25.781431	25.781431	0.00	1
		4	30	39.537574	39.537574	39.537574	39.537574	0.00	0
	TRR	1	0	38.174917	80.084232	437.191990	54.975283	82.08	1
		2	0	33.022878	115.806950	658.762104	61.160962	131.73	1
		3	0	35.043037	136.701257	806.049917	74.006241	186.53	1
		4	0	46.219864	65.138385	125.127617	59.559891	18.31	1
MMD	PSO	1	14	31.056458	31.151369	31.355355	31.057992	0.13	–
		2	26	24.095459	24.129881	24.612998	24.095459	0.12	–
		3	15	27.066388	27.242809	28.630456	27.066426	0.40	–
		4	29	44.347172	44.347172	44.347173	44.347172	0.00	–
	AS	1	0	31.056462	31.078239	31.525284	31.056759	0.09	0
		2	0	24.095463	24.135839	24.835710	24.095768	0.14	1
		3	0	27.066398	27.224033	29.311025	27.072111	0.45	0
		4	0	44.347187	44.419303	45.299474	44.352565	0.23	1
	IP	1	0	31.241349	31.244034	31.245254	31.244358	0.00	1
		2	0	24.279991	24.285232	24.287873	24.285217	0.00	1
		3	0	27.228921	27.248421	27.259118	27.249201	0.00	1
		4	0	44.524078	44.535475	44.598444	44.532140	0.01	1
	TRR	1	0	94.109723	576.252212	3386.972994	288.327944	715.21	1
		2	0	87.636164	605.994373	3050.645396	269.316773	740.79	1
		3	0	45.631121	2812.207090	36087.939245	627.444041	6679.12	1
		4	0	61.519355	208.647801	439.313149	185.546088	93.59	1

Table 3.21: Statistical results for PSO, AS, IP, TRR, $R_{\text{target}} = 144$ kbps and $W_t = 22.5$ MHz.

Criterion	Algorithm	Case	Success	Min	Mean	Max	Median	Std	Ranksum
NNBS	PSO	1	26	-6.728280	-6.706684	-6.317561	-6.728280	0.08	–
		2	28	-7.817586	-7.817496	-7.814898	-7.817586	0.00	–
		3	27	-7.198682	-7.167147	-6.771945	-7.198682	0.11	–
		4	20	-5.259108	-5.158823	-4.178432	-5.259108	0.24	–
	AS	1	3	-6.728280	-2.348921	1.000000	1.000000	3.90	1
		2	7	-7.817586	-2.820954	1.000000	1.000000	4.44	1
		3	5	-7.198682	-3.372630	1.000000	-7.198680	4.16	1
		4	13	-5.259108	-2.129554	1.000000	-2.129553	3.18	0
	IP	1	13	-6.728280	-2.348921	1.000000	1.000000	3.90	1
		2	13	-7.817586	-2.820954	1.000000	1.000000	4.44	1
		3	16	-7.198682	-3.372630	1.000000	-7.198682	4.16	1
		4	15	-5.259108	-2.129554	1.000000	-2.129554	3.18	0
	TRR	1	0	-5.920830	-1.691694	1.000000	1.000000	3.03	1
		2	0	-7.072670	-2.070224	1.000000	1.000000	3.61	1
		3	0	-6.461379	-2.447709	1.000000	-4.086208	3.33	1
		4	0	-4.428599	-1.096080	1.000000	-0.021107	2.28	1
CNBS	PSO	1	26	-6.728280	-6.706684	-6.317561	-6.728280	0.08	–
		2	29	-7.494610	-7.494609	-7.494581	-7.494610	0.00	–
		3	30	-7.247564	-7.247564	-7.247564	-7.247564	0.00	–
		4	24	-5.968381	-5.879394	-5.345384	-5.968381	0.18	–
	AS	1	3	-6.728280	-2.348921	1.000000	1.000000	3.90	1
		2	7	-7.494610	-2.680997	1.000000	1.000000	4.28	1
		3	14	-7.247564	-3.398701	1.000000	-7.247563	4.18	1
		4	2	-5.968381	-2.484190	1.000000	-2.484189	3.54	1
	IP	1	13	-6.728280	-2.348921	1.000000	1.000000	3.90	1
		2	13	-7.494610	-2.680998	1.000000	1.000000	4.28	1
		3	16	-7.247564	-3.398701	1.000000	-7.247564	4.18	1
		4	15	-5.968381	-2.484190	1.000000	-2.484191	3.54	0
	TRR	1	0	-5.920830	-1.691694	1.000000	1.000000	3.03	1
		2	0	-6.353936	-1.683354	1.000000	1.000000	3.18	1
		3	0	-6.700017	-2.361050	1.000000	-3.812727	3.26	1
		4	0	-5.471234	-1.591104	1.000000	-0.546628	2.75	1
MAD	PSO	1	30	56.480888	56.480888	56.480888	56.480888	0.00	–
		2	30	44.424576	44.424576	44.424576	44.424576	0.00	–
		3	30	50.564936	50.564936	50.564936	50.564936	0.00	–
		4	30	78.125561	78.125561	78.125561	78.125561	0.00	–
	AS	1	28	56.480888	56.480888	56.480889	56.480888	0.00	0
		2	28	44.424576	44.424576	44.424577	44.424576	0.00	0
		3	25	50.564936	50.564936	50.564937	50.564936	0.00	0
		4	29	78.125561	78.125561	78.125562	78.125561	0.00	0
	IP	1	30	56.480888	56.480888	56.480888	56.480888	0.00	0
		2	30	44.424576	44.424576	44.424576	44.424576	0.00	0
		3	30	50.564936	50.564936	50.564936	50.564936	0.00	0
		4	30	78.125561	78.125561	78.125561	78.125561	0.00	0
	TRR	1	0	70.058420	114.105190	305.226320	100.026179	49.94	1
		2	0	55.440766	95.548555	265.978810	83.776112	42.75	1
		3	0	62.454791	102.904933	297.469295	87.282984	50.16	1
		4	0	94.335021	142.020580	313.279410	126.272882	52.33	1
MMD	PSO	1	26	62.640217	62.674556	63.670340	62.640217	0.19	–
		2	30	50.499656	50.499656	50.499656	50.499656	0.00	–
		3	29	55.336279	55.446682	58.648373	55.336279	0.60	–
		4	25	86.090121	86.138088	86.839018	86.090121	0.16	–
	AS	1	0	62.640227	62.709359	64.231017	62.641429	0.29	1
		2	0	50.499664	50.597425	52.681553	50.501584	0.40	1
		3	0	55.336303	55.451090	56.332457	55.337953	0.28	1
		4	0	86.090138	86.298447	88.117830	86.106936	0.49	1
	IP	1	0	62.667073	62.689967	62.702033	62.691908	0.01	1
		2	0	50.666813	50.843280	55.099470	50.695189	0.80	1
		3	0	55.484306	56.219901	76.256342	55.529304	3.78	1
		4	0	86.090126	86.249038	90.661932	86.090335	0.83	1
	TRR	1	0	105.389738	661.950070	1853.695807	472.654046	473.79	1
		2	0	104.554625	684.388187	1739.286364	533.403263	513.81	1
		3	0	88.154391	650.196306	2273.759910	411.834949	534.75	1
		4	0	131.077527	515.742949	1396.314830	392.958246	297.15	1

Lastly, experimental results of the PSO's convergence speed are presented on Table 3.22. Specifically, this table shows the time that PSO requires to find the optimal solution with a precision of 6 decimal digits, in terms of the number of iterations that the swarm is updated. Thus, these statistics concern only the experiments where PSO reaches the optimal solution. In this table are included results for all node distributions of all considered bitrate and bandwidth combinations, and for all tested criteria. Due to the fact that PSO is a stochastic algorithm, we do not only cite the average performance, i.e., the mean number of iterations that the swarm is updated in order to reach the solution (Mean), but we also present the best case, i.e., the minimum number of iterations (Min) and the worst case, i.e., the maximum number of iterations (Max).

The criterion that presents the fastest PSO's convergence, requiring less iterations on average over all 30 experiments per case, is the MAD, which behaves better than all the other competing schemes. On the contrary, PSO confronts the biggest challenge in convergence, using the MMD criterion. With an exception for the case of $R_{\text{target}} = 96$ kbps and $W_t = 15$ MHz, in all other bitrate and bandwidth combinations, the MMD requires much more iterations on average compared to the other criteria. Especially when the bitrate is equal to $R_{\text{target}} = 144$ kbps, the lowest average iteration number equals 684 out of 700 iterations.

Table 3.22: PSO convergence speed in terms of best (Min), average (Mean) and worst (Max) swarm update iterations.

		NNBS			CNBS			MAD			MMD		
	Case	Min	Mean	Max	Min	Mean	Max	Min	Mean	Max	Min	Mean	Max
$R_{\text{target}} = 96$ **kbps** $W_t = 20$ **MHz**	1	465	559.4	696	443	561.4	682	366	482.7	650	518	611.7	700
	2	336	558.6	700	302	523.3	686	288	437.8	547	497	587.7	660
	3	437	552.6	700	385	503.1	666	307	402.1	646	536	620.6	699
	4	500	588.8	692	404	582.3	700	294	405.6	694	373	429.6	498
$R_{\text{target}} = 96$ **kbps** $W_t = 15$ **MHz**	1	490	660.9	700	470	597.9	697	310	421.9	695	389	463.8	561
	2	391	593.5	700	405	576.6	700	281	401.2	641	323	390.2	522
	3	414	605.7	694	495	563.9	681	361	532.1	680	364	424.9	546
	4	447	621.9	700	374	598.6	699	318	390.5	471	364	413.8	498
$R_{\text{target}} = 144$ **kbps** $W_t = 30$ **MHz**	1	461	630.5	700	518	625.1	700	383	505.3	679	687	697.6	700
	2	433	599.7	697	442	606.0	689	414	548.0	694	636	684.0	700
	3	426	594.1	700	420	604.7	697	418	592.2	700	691	698.4	700
	4	500	659.8	700	421	592.5	700	322	488.4	672	677	695.2	700
$R_{\text{target}} = 144$ **kbps** $W_t = 22.5$ **MHz**	1	437	649.2	700	427	627.2	699	376	506.3	681	691	697.9	700
	2	454	636.4	700	393	616.0	700	386	552.5	700	645	691.8	700
	3	454	626.1	700	482	619.0	700	396	560.0	686	667	693.5	700
	4	509	658.2	700	453	652.8	700	366	540.2	690	692	698.2	700

Summarizing the key points of this set of experiments, we confirm that both NBS-based approaches keep low computational complexity and a wise selection between NNBS and CNBS according to the needs of each application and the node distribution, produces worthwhile results that are preferable to those of MAD and MMD. Regarding the optimization solver, PSO algorithm is proved the best choice among other conventional optimization methods, including AS, IP and TRR, for solving the mixed-integer tasks of the considered problem formulation, under the constraints of a fixed bitrate and a bounded power level for each node.

3. **Third Set of Experiments**

This third set of experiments has elaborated on the resource allocation problem, when the KSBS is employed. In this case, we have modeled the thermal and background noise as AWGN with $N_0 = 10^{-7}$ W/Hz, under the system setup as described in case (a) of Section 3.8.2. For values of N_0 smaller than 10^{-7} W/Hz, a marginal PSNR increase is anticipated, which is trivial and unperceivable by the human eye. In order to compare KSBS performance, we have employed the NNBS, MTU and w.MTU criteria. At this point, we should recall that only the KSBS has been derived geometrically, while the NNBS, MTU and w.MTU have used the PSO algorithm to solve the resulting mixed-integer optimization problems, using the parameter set described in Section 3.6.1. Moreover, a significant part of the experiments of this group have focused on the reliable evaluation of the achieved results on both the quality and resource domains.

Table 3.23 explores the effect of assigning different dp values to the results of the KSBS. The terms $PSNR_{\text{high}}$ and $PSNR_{\text{low}}$ refer to the PSNR achieved by the high- and low-motion class, respectively. It can be seen that higher dp values favor the high-motion class and lower dp values favor the low-motion class. Videos with more intense motion activity are generally considered as more important compared to more stationary videos, since such videos image scenes of interest. Therefore, aiming at better video quality for high-motion scenes, we have chosen to initialize dp with the highest values among the tested ones for each bitrate and bandwidth combination. Specifically, for $R_{\text{target}} = 96$ kbps and $W_t = 20$ MHz, the selected dp vector is dp $= 28$ dB, while for $R_{\text{target}} = 96$ kbps and $W_t = 15$ MHz, the selected dp vector is dp $= 26$ dB. It is worth mentioning that it is not necessary for the dp to have the same value for both motion classes. However, we have made this assumption in an effort to be equally fair to all of them.

Also, from Table 3.23, we infer that the PSNR values for both motion classes are reduced when the bandwidth is reduced, while the bitrate, N_0 and the disagreement point remain the same. This is attributed to the fact that when the bandwidth W_t is reduced, the term I_0, which is equal to $I_0 = \sum_{j \neq k}^{N} S_j / W_t$, increases. Thus, the energy per bit to MAI and noise ratio of Eq. (3.9) becomes lower, which leads to reduced PSNR values.

Table 3.23: PSNR results for 3 different dp assignments per bitrate and bandwidth combination.

| | $R_{\text{target}} = 96$ kbps, $W_t = 20$ MHz | | | | | | $R_{\text{target}} = 96$ kbps, $W_t = 15$ MHz | | | | | |
| | dp $= 28$ | | dp $= 26$ | | dp $= 24$ | | dp $= 26$ | | dp $= 25$ | | dp $= 24$ | |
$N_{\text{high}} - N_{\text{low}}$	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$90 - 10$	28.2248	31.7811	27.6533	38.7737	27.4387	40.4599	26.3072	33.3874	26.0796	37.7415	25.9756	39.1441
$70 - 30$	29.0590	32.8873	28.3531	35.5374	28.0505	36.7516	26.7322	31.7317	26.3766	33.6255	26.1257	34.6615
$50 - 50$	30.3679	33.5810	29.5671	34.3620	29.2021	35.1549	27.6806	30.9220	27.2761	31.6966	27.0460	32.4253
$30 - 70$	32.0458	34.3732	31.5431	34.8620	31.2067	35.2058	29.5953	31.5920	29.2774	31.8243	29.0081	32.0654
$10 - 90$	34.9841	36.0284	34.7502	36.1288	34.5811	36.1992	32.9591	32.9897	32.8731	33.0132	32.7973	33.0311

Continuing, Table 3.24 includes the results for the NNBS, KSBS, MTU and w.MTU, when $R_{\text{target}} = 96$ kbps and $W_t = 20$ MHz, for all considered node distributions. In this case, NNBS and KSBS assume dp $= 28$ dB. The same results for the aforementioned criteria are also depicted on Table 3.25, but for the case of $R_{\text{target}} = 96$ kbps and $W_t = 15$ MHz. In this case, NNBS and KSBS assume dp $= 26$ dB. The combination of the source-channel coding rate, and the power level of the high-motion class are represented as $(R_{\text{s,high}}, R_{\text{c,high}})$, and S_{high}, respectively, while $(R_{\text{s,low}}, R_{\text{c,low}})$ and S_{low} are the corresponding parameters for the low-motion class.

First of all, all four criteria give a higher PSNR to the low-motion class of nodes compared to the high-motion class, with an exception observed for the MTU criterion, in cases where more nodes belong to the low-motion class. In such cases, the high-motion class achieves higher PSNR values than the low-motion class. The KSBS is a promising criterion, since it assigns close enough values to the PSNR of the two motion classes. Compared to the other schemes, the KSBS favors the high-motion class clearly more than the w.MTU and in many cases more than the NNBS and MTU. This fact plays an important role considering the significance of the scenes that include high levels of motion.

The MTU criterion guarantees the highest levels of total utility, cumulatively for both motion classes, compared to all other schemes. However, in cases where the cardinality of the low-motion class is smaller than that of the high-motion class, a large discrepancy between $PSNR_{\text{high}}$ and $PSNR_{\text{low}}$ is observed. Interpreting the results for the w.MTU, it favors eminently the low-motion class of nodes, offering clearly higher PSNR values compared to the NNBS and KSBS, and even in some cases compared to the MTU. Interesting are the cases where the two motion classes include the same number of nodes. In such cases, both MTU and w.MTU offer exactly the same solution, i.e., the same PSNR values to both motion classes.

Regarding the power levels, for the NBS and KSBS we observe that the high-motion class of nodes requires higher power levels compared to the low-motion class, unlike w.MTU where the low-motion class maintains the highest power levels. For the MTU, we infer that the class that has the highest power level, achieves the highest PSNR. Also, for the source and channel coding rate combinations, since the total bitrate is assumed to be constant, a higher source coding rate means that fewer bits are available for channel coding, resulting in lower error protection. Therefore, a higher power level is required in order to increase channel reliability, increasing at the same time the interference to the transmissions of the other nodes.

The PF values for the NNBS, KSBS and w.MTU are included in Table 3.26. Since the MTU is used as the reference criterion in Eq. (3.33), the PF values for this criterion are not defined. Moreover, as it has been previously implied, in cases where the nodes are equally assigned to both motion classes, the w.MTU solutions coincide with the solutions of the MTU. Hence, in such cases the PF values are not defined either for the w.MTU.

Table 3.24: Results for $R_{\text{target}} = 96\,\text{kbps}$, $W_t = 20\,\text{MHz}$. For the NNBS and KSBS dp = 28 dB.

	NNBS						KSBS					
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$90 - 10$	$(48, 1/2)$	15.0000	$(32, 1/3)$	6.1023	28.3548	29.1082	$(48, 1/2)$	11.7000	$(32, 1/3)$	6.3000	28.2248	31.7811
$70 - 30$	$(64, 2/3)$	15.0000	$(32, 1/3)$	6.3135	29.5326	30.8287	$(48, 1/2)$	9.5000	$(32, 1/3)$	5.3000	29.0590	32.8873
$50 - 50$	$(64, 2/3)$	15.0000	$(32, 1/3)$	6.9364	30.9757	32.7535	$(64, 2/3)$	9.8000	$(32, 1/3)$	5.3000	30.3679	33.5810
$30 - 70$	$(64, 2/3)$	15.0000	$(64, 2/3)$	8.9106	31.2109	35.2367	$(64, 2/3)$	9.7000	$(32, 1/3)$	5.0000	32.0458	34.3732
$10 - 90$	$(64, 2/3)$	15.0000	$(64, 2/3)$	9.1006	32.2861	36.9037	$(64, 2/3)$	15.0000	$(64, 2/3)$	6.3000	34.9841	36.0284

	MTU						w.MTU					
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$90 - 10$	$(32, 1/3)$	5.0000	$(64, 2/3)$	15.0000	26.7244	44.9688	$(48, 1/2)$	13.6537	$(64, 2/3)$	15.0000	27.6931	38.6067
$70 - 30$	$(32, 1/3)$	5.0246	$(64, 2/3)$	15.0000	25.3578	43.1664	$(32, 1/3)$	8.3514	$(64, 2/3)$	15.0000	26.5871	41.2879
$50 - 50$	$(32, 1/3)$	8.1044	$(64, 2/3)$	15.0000	25.9290	40.3410	$(32, 1/3)$	8.1044	$(64, 2/3)$	15.0000	25.9290	40.3410
$30 - 70$	$(64, 2/3)$	15.0000	$(32, 1/3)$	5.6892	33.5507	33.2252	$(32, 1/3)$	7.9229	$(64, 2/3)$	15.0000	25.2663	39.3116
$10 - 90$	$(64, 2/3)$	15.0000	$(64, 2/3)$	5.0000	36.3876	35.2762	$(32, 1/3)$	7.7983	$(64, 2/3)$	15.0000	24.6122	38.2269

Table 3.25: Results for $R_{\text{target}} = 96\,\text{kbps}$, $W_t = 15\,\text{MHz}$. For the NNBS and KSBS dp = 26 dB.

	NNBS						KSBS					
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$90 - 10$	$(32, 1/3)$	15.0000	$(32, 1/3)$	7.8222	26.4914	28.5181	$(32, 1/3)$	9.1000	$(32, 1/3)$	8.2000	26.3072	33.3874
$70 - 30$	$(32, 1/3)$	15.0000	$(32, 1/3)$	8.1418	26.9339	30.0794	$(32, 1/3)$	8.9000	$(32, 1/3)$	6.0000	26.7322	31.7317
$50 - 50$	$(48, 1/2)$	15.0000	$(32, 1/3)$	7.1911	27.9503	30.3836	$(48, 1/2)$	9.6000	$(32, 1/3)$	5.0000	27.6806	30.9220
$30 - 70$	$(48, 1/2)$	15.0000	$(32, 1/3)$	7.6311	29.0261	32.0622	$(64, 2/3)$	11.2000	$(32, 1/3)$	5.1000	29.5953	31.5920
$10 - 90$	$(64, 2/3)$	15.0000	$(32, 1/3)$	7.4228	31.0886	33.3624	$(64, 2/3)$	13.2000	$(32, 1/3)$	5.2000	32.9591	32.9897

	MTU						w.MTU					
$N_{\text{high}} - N_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$	$(R_{\text{s,high}}, R_{\text{c,high}})$	S_{high}	$(R_{\text{s,low}}, R_{\text{c,low}})$	S_{low}	$PSNR_{\text{high}}$	$PSNR_{\text{low}}$
$90 - 10$	$(32, 1/3)$	5.0000	$(64, 2/3)$	15.0000	25.3543	43.2224	$(32, 1/3)$	8.1158	$(64, 2/3)$	15.0000	25.8689	40.2343
$70 - 30$	$(32, 1/3)$	5.0263	$(64, 2/3)$	15.0000	23.7017	41.0550	$(32, 1/3)$	7.8819	$(64, 2/3)$	15.0000	25.0542	38.9792
$50 - 50$	$(32, 1/3)$	7.7475	$(64, 2/3)$	15.0000	24.2297	37.5805	$(64, 2/3)$	7.7475	$(64, 2/3)$	15.0000	24.2297	37.5805
$30 - 70$	$(64, 2/3)$	15.0000	$(32, 1/3)$	5.5879	30.7752	30.6264	$(32, 1/3)$	7.6648	$(64, 2/3)$	15.0000	23.3987	36.0942
$10 - 90$	$(64, 2/3)$	15.0000	$(32, 1/3)$	5.0000	34.1215	32.6676	$(32, 1/3)$	11.1948	$(32, 1/3)$	15.0000	24.9082	34.3075

The obtained PF results can be explained as follows: for every unit of utility lost by a class of nodes using the MTU, there are PF units of utility gained cumulatively from both motion classes, using also the MTU. The tendency of PF values for each scheme is similar for both considered combinations of bitrate and bandwidth, from node distribution to node distribution. Specifically, as the cardinality of the high-motion class decreases against the cardinality of the low-motion class, a PF decrease is observed, except for the case of "$N_{\text{high}} = 10 - N_{\text{low}} = 90$" where the PF value is increased. Additionally, no specific scheme offers the highest or the lowest PF values in all node distributions. This always depends on the achieved PSNR values in each case. However, the conclusion derived using this metric is that the lower the PF value for a scheme, the smaller the discrepancy between the total achieved PSNR of the considered scheme and the MTU. Therefore, when the number of nodes that belong to the high-motion class increases, the utility gained cumulatively from both motion classes decreases.

Additional pieces of information are also provided by the graphical illustration of the results. Figure 3.12 depicts the relation between the total achieved utility and the total consumed power, for all examined criteria. Each subfigure refers to a specific node distribution and presents the results for both considered bitrate and bandwidth combinations. As we observe, the tendency of the total utility as well as that of the total
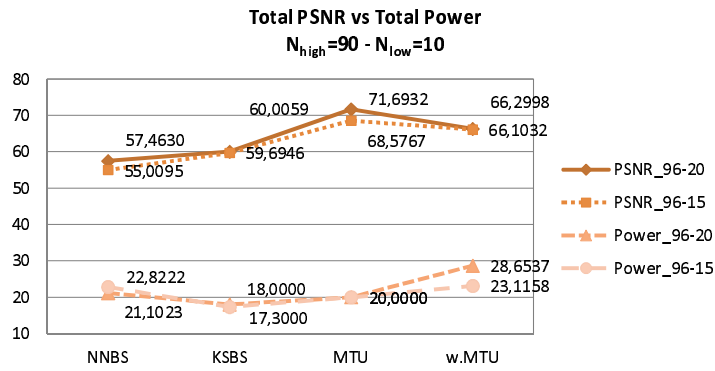
Table 3.26: PF values per bitrate and bandwidth combination.

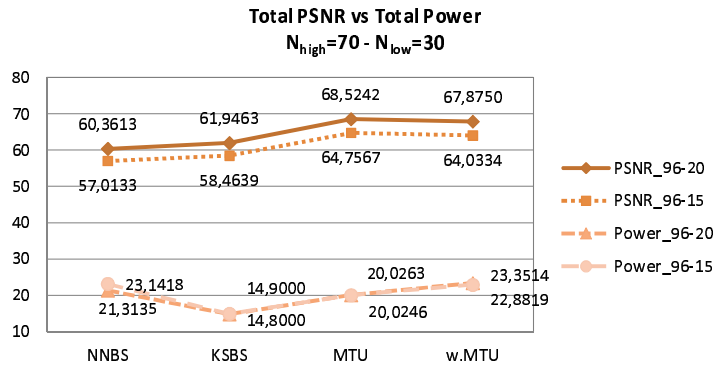| $N_{\text{high}} - N_{\text{low}}$	$R_{\text{target}} = 96$ kbps, $W_t = 20$ MHz			$R_{\text{target}} = 96$ kbps, $W_t = 15$ MHz		
	NNBS	KSBS	w.MTU	NNBS	KSBS	w.MTU
$90 - 10$	8.7280	7.7895	5.5677	11.9314	9.3211	4.8066
$70 - 30$	1.9553	1.7772	0.5281	2.3957	2.0765	0.5348
$50 - 50$	0.5035	0.5229	–	0.9343	0.9295	–
$30 - 70$	0.1632	0.3109	0.3611	0.2182	0.2219	0.3491
$10 - 90$	1.5201	0.8659	2.9907	3.3651	2.6088	4.6182

power consumption is similar for both considered bitrate and bandwidth combinations of this set of experiments. Specifically, the sum of the PSNR values is reduced in all criteria and node distributions, when the bandwidth is reduced (keeping the bitrate constant), since in such a case the value of Eq. (3.9) decreases. For the sum of the power levels, there is no noticeable difference between the two considerations for the bandwidth.

From Fig. 3.12, we also observe that no scheme simultaneously holds the desired features of achieving the highest levels of utility and consuming the lowest levels of power, cumulatively for both motion classes. Clearly, such a scheme would be a preferable scheme. Although the MTU assures the highest levels of utility, it is an unfair scheme if we consider the amounts of consumed power as well as the high discrepancy that is often observed in the PSNR values of the two motion classes. Alternatively, if we are interested in achieving similar PSNR values for both motion classes, we could say that in some cases the NBS is the most suitable criterion, while in some other cases the MTU meets this requirement. Despite all these, neither the NBS nor the MTU can be considered as equally fair criteria to both motion classes, if we also consider the power levels required by each motion class. In cases of similar PSNR values, the high-motion class is undoubtedly more demanding in resources.

Summarizing the key points of this set of experiments, we have to mention that no scheme of the NNBS, KSBS, MTU and w.MTU gathers all desired features of being equally fair to all nodes, assuring the highest total utility, and requiring the lowest levels of power, at the same time. Nevertheless, comparisons have led us to the conclusion that the KSBS is the criterion that is closer to our demands, since it is a compromise to all our requirements. The main strength of this method is that it guarantees the lowest power level values, cumulatively for both motion classes, far exceeding the other competing methods. Additionally, it assigns close PSNR values to both motion classes as compared to the values assigned by the MTU and w.MTU and even by the NNBS, in cases where the cardinality of the low-motion class is greater than that of the high-motion one. Also, the KSBS clearly outperforms the NNBS in terms of the total utility gained by both motion classes, and in cases where more nodes belong to the low-motion class, it also outperforms the w.MTU, keeping low running complexity.

Total PSNR vs Total Power
N_high=90 - N_low=10

(a)

Total PSNR vs Total Power
N_high=70 - N_low=30

(b)

Total PSNR vs Total Power
N_high=50 - N_low=50
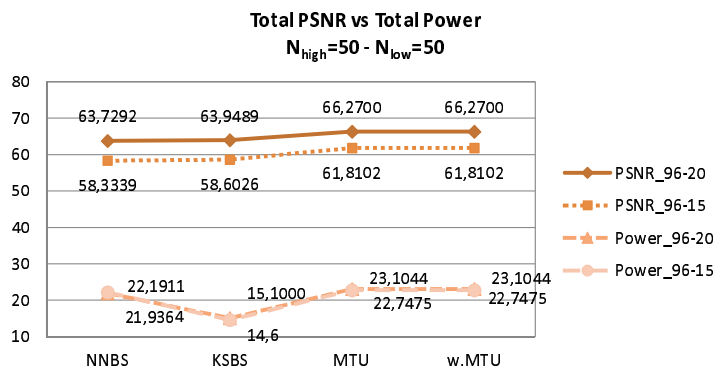
(c)

(d)

(e)

Figure 3.12: Total PSNR gain versus total power consumption.
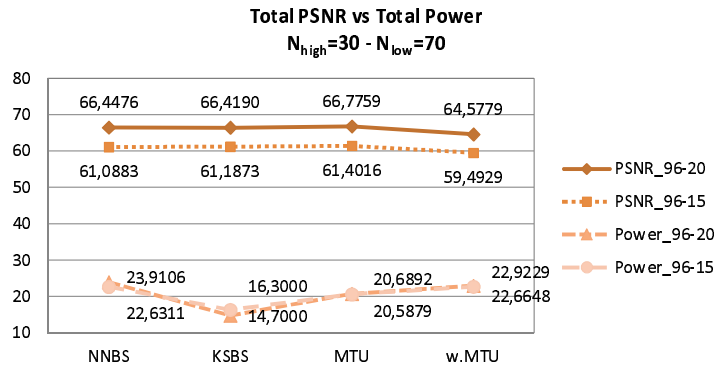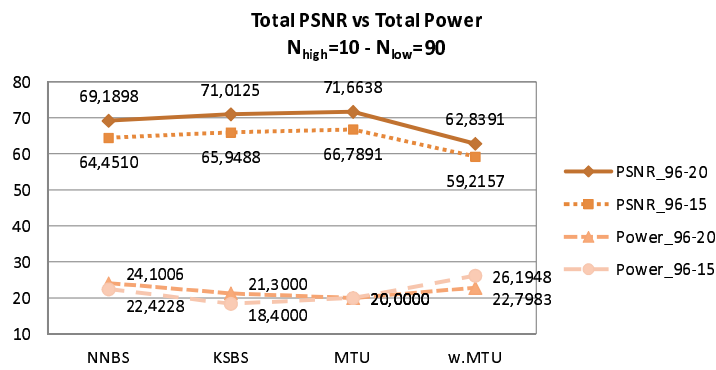
4. **Fourth Set of Experiments**

Last but not least, this last group of conducted experiments has dealt with the investigation of the most fair and efficient optimization scheme among the MAD, MMD, NNBS, CNBS, KSBS, MTU and w.MTU, under the system setup as described in case (a) of Section 3.8.2, for the case of $R_{\text{target}} = 96$ kbps and $W_t = 20$ MHz, when the thermal and background noise is not neglected and equals $N_0 = 10^{-7}$ W/Hz. All of the examined schemes optimize a function of the video qualities of the nodes. However, there is no single scheme that maximizes the video quality of each node simultaneously. In fact, all presented schemes are able to provide a Pareto-optimal solution, meaning that there is no other solution that is simultaneously preferred by all nodes, and thus, there is no single scheme that would be selected by all nodes to be the best. Indicatively, Fig. 3.13 graphically depicts the Pareto-optimal solutions achieved by each of the considered schemes assuming that the high-motion class includes $N_{\text{high}} = 70$ nodes and the low-motion class $N_{\text{low}} = 30$ nodes.
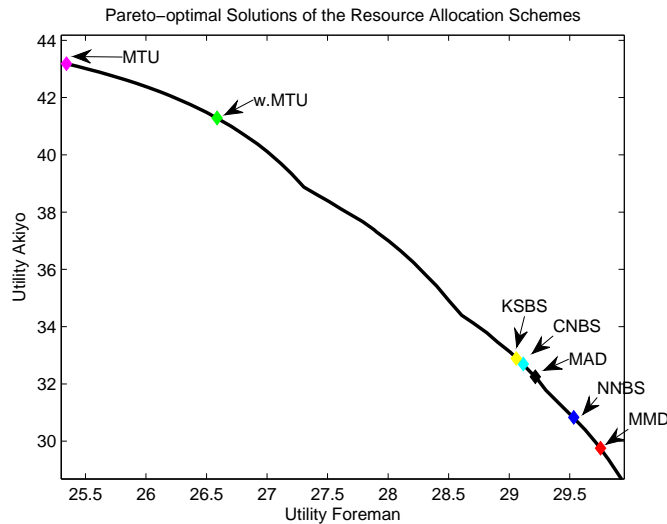
Figure 3.13: Pareto-optimality of the solutions.

Four different metrics have been employed in the specific set of experiments in order to handle the resulting tradeoffs, investigating fairness and efficiency under different perspectives. Tables 3.27-3.31 present the results for all considered fairness metrics. Each of the tables refers to a different node distribution and each line of the tables refers to a specific scheme. The term N_{high} declares the cardinality of the high-motion class of nodes and N_{low} the cardinality of the low-motion class of nodes. The first column of each table shows the schemes, the second column shows the PF values of each scheme and the third column cites the JI values of the nodes' utilities. The fourth column depicts the total utility achieved by each scheme, and the last column shows the total consumed power for all schemes. Since a fair and efficient scheme guarantees high

amounts of total utility, is equally fair to all motion classes and is not demanding in resources (in our case power levels), we have used bold type for the lowest PF value, the highest JI value, the highest total utility and the lowest total power among all schemes, for each considered node distribution.

Table 3.27: Fairness metrics for the case of $N_{\text{high}} = 90 - N_{\text{low}} = 10$.

Scheme	PF	JI	Total Utility	Total Power
MAD	7.9258	0.9974	59.5422	22.6210
MMD	9.0271	**1.0000**	56.7578	20.7257
NNBS	8.7280	0.9998	57.4630	21.1023
CNBS	7.7732	0.9965	59.9914	23.0548
KSBS	7.7895	0.9965	60.0059	**18.0000**
MTU	–	0.9392	**71.6932**	20.0000
w.MTU	**5.5677**	0.9736	66.2998	28.6537

Table 3.28: Fairness metrics for the case of $N_{\text{high}} = 70 - N_{\text{low}} = 30$.

Scheme	PF	JI	Total Utility	Total Power
MAD	1.8301	0.9976	61.4641	22.6110
MMD	2.0510	**1.0000**	59.5078	20.5814
NNBS	1.9553	0.9995	60.3613	21.3135
CNBS	1.7870	0.9967	61.8099	23.1208
KSBS	1.7772	0.9962	61.9463	**14.8000**
MTU	–	0.9365	**68.5241**	20.0000
w.MTU	**0.5281**	0.9552	67.8748	23.3521

Table 3.29: Fairness metrics for the case of $N_{\text{high}} = 50 - N_{\text{low}} = 50$.

Scheme	PF	JI	Total Utility	Total Power
MAD	**0.5027**	0.9991	63.7608	22.0445
MMD	0.5450	**1.0000**	63.1836	20.7300
NNBS	0.5035	0.9992	63.7292	21.9364
CNBS	0.5035	0.9992	63.7292	21.9364
KSBS	0.5229	0.9975	63.9489	**15.1000**
MTU	–	0.9548	**66.2700**	23.1044
w.MTU	–	0.9548	**66.2700**	23.1044

Moreover, in Tables 3.32-3.36, we present the PSNR of the high-motion class, the PSNR of the low-motion class, the power level of the high-motion class and the power level of the low-motion class, respectively. Of course, each line of the tables refers to a specific scheme, while each of the tables refers to a different node distribution.

Table 3.30: Fairness metrics for the case of $N_{\text{high}} = 30 - N_{\text{low}} = 70$.

Scheme	PF	JI	Total Utility	Total Power
MAD	0.1749	0.9968	66.4491	23.7092
MMD	0.0331	**1.0000**	66.7706	20.9121
NNBS	0.1632	0.9963	66.4476	23.9106
CNBS	**0.0165**	0.9999	66.7746	20.7968
KSBS	0.3109	0.9988	66.4190	**14.7000**
MTU	−	0.9999	**66.7758**	20.6893
w.MTU	0.3611	0.9548	64.5780	22.9236

Table 3.31: Fairness metrics for the case of $N_{\text{high}} = 10 - N_{\text{low}} = 90$.

Scheme	PF	JI	Total Utility	Total Power
MAD	1.3567	0.9970	69.6722	23.4148
MMD	**0.7140**	**1.0000**	71.3714	20.6298
NNBS	1.5201	0.9956	69.1898	24.1006
CNBS	0.8418	0.9999	71.4211	21.9862
KSBS	0.8659	0.9998	71.0125	21.3000
MTU	−	0.9998	**71.6638**	**20.0000**
w.MTU	2.9907	0.9552	62.8391	22.7983

Table 3.32: PSNR values and power level values for the case of $N_{\text{high}} = 90 - N_{\text{low}} = 10$.

Scheme	PSNR	Power
MAD	$[28.2575, 31.2847]$	$[15.0000, 7.6210]$
MMD	$[28.3789, 28.3789]$	$[15.0000, 5.7257]$
NNBS	$[28.3548, 29.1082]$	$[15.0000, 6.1023]$
CNBS	$[28.2298, 31.7616]$	$[15.0000, 8.0548]$
KSBS	$[28.2248, 31.7811]$	$[11.7000, 6.3000]$
MTU	$[26.7244, 44.9688]$	$[5.0000, 15.0000]$
w.MTU	$[27.6931, 38.6067]$	$[13.6537, 15.0000]$

Table 3.33: PSNR values and power level values for the case of $N_{\text{high}} = 70 - N_{\text{low}} = 30$.

Scheme	PSNR	Power
MAD	$[29.2156, 32.2485]$	$[15.0000, 7.6110]$
MMD	$[29.7539, 29.7539]$	$[15.0000, 5.5814]$
NNBS	$[29.5326, 30.8287]$	$[15.0000, 6.3135]$
CNBS	$[29.1152, 32.6947]$	$[15.0000, 8.1208]$
KSBS	$[29.0590, 32.8873]$	$[9.5000, 5.3000]$
MTU	$[25.3434, 43.1807]$	$[5.0000, 15.0000]$
w.MTU	$[26.5873, 41.2875]$	$[8.3521, 15.0000]$

Table 3.34: PSNR values and power level values for the case of $N_{\text{high}} = 50 - N_{\text{low}} = 50$.

Scheme	PSNR	Power
MAD	$[30.9207, 32.8401]$	$[15.0000, 7.0445]$
MMD	$[31.5918, 31.5918]$	$[15.0000, 5.7300]$
NNBS	$[30.9757, 32.7535]$	$[15.0000, 6.9364]$
CNBS	$[30.9757, 32.7535]$	$[15.0000, 6.9364]$
KSBS	$[30.3679, 33.5810]$	$[9.8000, 5.3000]$
MTU	$[25.9290, 40.3410]$	$[8.1044, 15.0000]$
w.MTU	$[25.9290, 40.3410]$	$[8.1044, 15.0000]$

Table 3.35: PSNR values and power level values for the case of $N_{\text{high}} = 30 - N_{\text{low}} = 70$.

Scheme	PSNR	Power
MAD	$[31.3549, 35.0942]$	$[15.0000, 8.7092]$
MMD	$[33.3853, 33.3853]$	$[15.0000, 5.9121]$
NNBS	$[31.2109, 35.2367]$	$[15.0000, 8.9106]$
CNBS	$[33.4708, 33.3038]$	$[15.0000, 5.7968]$
KSBS	$[32.0458, 34.3732]$	$[9.7000, 5.0000]$
MTU	$[33.5506, 33.2252]$	$[15.0000, 5.6893]$
w.MTU	$[25.2666, 39.3114]$	$[7.9236, 15.0000]$

Table 3.36: PSNR values and power level values for the case of $N_{\text{high}} = 10 - N_{\text{low}} = 90$.

Scheme	PSNR	Power
MAD	$[32.9280, 36.7442]$	$[15.0000, 8.4148]$
MMD	$[35.6857, 35.6857]$	$[15.0000, 5.6298]$
NNBS	$[32.2861, 36.9037]$	$[15.0000, 9.1006]$
CNBS	$[35.8566, 35.5645]$	$[15.0000, 6.9862]$
KSBS	$[34.9841, 36.0284]$	$[15.0000, 6.3000]$
MTU	$[36.3876, 35.2762]$	$[15.0000, 5.0000]$
w.MTU	$[24.6122, 38.2269]$	$[7.7983, 15.0000]$

Regarding the results from Tables 3.27-3.31, one way to interpret the PF values obtained using Eq. (3.33) is that for every unit of utility lost by a class of nodes using the MTU instead of the considered scheme, there are PF units of utility gained cumulatively from both motion classes using also the MTU instead of the considered scheme, as it was also explained in the third set of experiments. Additionally, the lower the PF value for a scheme, the smaller the discrepancy between the total achieved PSNR by the considered scheme and the MTU. Therefore, if we desire to have a high total utility, the scheme that offers the lowest PF value is the preferred one. However, no specific scheme holds the lowest PF values for all considered node distributions. This always depends on the achieved PSNR values in each case. Moreover, since the MTU criterion has been

considered as the reference criterion in Eq. (3.33), the PF values for this scheme are not defined. Additionally, in cases where both motion classes include the same number of nodes, the w.MTU solutions coincide with the solutions of the MTU. Hence, in such a case the PF values are not defined either for the w.MTU. From the JI values, we observe that all schemes promise quite fair utility allocations for both motion classes, since the JI values in all examined cases are greater than 0.93. However, the MMD criterion assures absolutely equal allocations for both motion classes, guaranteeing JI values equal to unity. This means that the MMD is fair to the 100% of the nodes, as it results from the definition of the Jain's index [74], and thus, it is the most fair scheme among all as it regards the equality of the utility allocations.

Additionally, if we consider that a high-performance scheme provides high amounts of utility cumulatively from both motion classes, the MTU is the scheme that can assure this requirement, as it is declared by its name. Indeed, as we see from Tables 3.27-3.31, this scheme offers the highest total utility in all considered node distributions. Finally, if the system resources are limited (as it is usual in wireless VSNs), it is necessary to have a scheme that is able to optimally allocate the transmission parameters among the nodes, while spending low amounts of power for the video transmission over the network, and guaranteeing adequate levels of viewing quality. In such a case, our choice is the KSBS criterion, since in four out of five node distributions of this group of experiments, it assures the lowest power consumption compared to all other schemes.

Generalizing, no scheme holds all desired characteristics of achieving the highest total utility, while assigning similar utilities to the two motion classes, and spending the lowest overall power, at the same time. Clearly, such a scheme would be a preferable scheme. Each proposed metric investigates fairness under a different perspective and it is rather impossible for a single metric to gather all aspects of fairness, at the same time.

Specifically, if we are interested in a scheme that gathers the highest amounts of utility compared to all other schemes, our choice would be the MTU criterion. Although the MTU assures the highest levels of utility, it is an unfair scheme if we consider the amounts of consumed power as well as the high discrepancy that is often observed between the PSNR values of the motion classes. The PF values indicate the scheme that approaches in performance the MTU. However, no specific scheme keeps the lowest PF values in all considered node distributions. From another point of view, if our priority is a scheme that assigns as close utilities as possible to both motion classes, surely the MMD criterion would be our selection. However, the total utility gained by the MMD is quite low relative with the total utility gained by the MTU. From another aspect, we would select the KSBS criterion, if we were looking for a scheme that consumes low amounts of power, while guaranteeing adequate levels of video viewing quality, at the same time. Nevertheless, this criterion fails to gather high amounts of total utility compared to the MTU, and also there is a large discrepancy between the utilities of the two motion classes, up to approximately 4 dB. Hence, the selection of the appropriate

scheme depends on the particular application in combination with the users' desires.

CHAPTER 4

ADAPTIVE GOP LENGTH AND RESOURCE ALLOCATION OVER WIRELESS VISUAL SENSOR NETWORKS

T his piece of our research adds an extra element to the issue of optimal resource allocation over wireless DS-CDMA VSNs, as studied in the previous section. Specifically, we investigate the combinatorial problem of optimal GOP length determination along with the optimal resource allocation, based on the amount of motion detected in the scenes captured by the nodes of the VSN.

4.1 Adaptive GOP Length

The H.264/AVC codec has the flexibility to determine the frequency of IDR-frames on the encoding side. Since IDR-frames are independently coded frames, the errors that occur within a GOP propagate to the following frames until the next IDR-frame is found. Generally, the more IDR-frames included in a video stream, the more editable it is and the greater its size is. Since predictive coding techniques are applied during encoding, the effect of channel errors on the video can have a tremendous impact after video transmission over error-prone environments. Thus, it is important to apply techniques that ensure a tolerable level of QoS.

It is widely accepted that scene changes or large variations can happen at any location in a video stream. This means that it is important to consider the video content in order to wisely arrange each of the IDR-, P- and B-frames in a GOP. Clearly, in the beginning of a new scene or after an abrupt scene change, an IDR-frame insertion is required in order to prohibit poor prediction for the next frames, since this type of intra frames does not allow the following frames to use frames appearing before it as references. Alternatively, when low levels of motion are included in a video stream, it is more efficient to use more P- or B-frames, instead of IDR-frames, to enhance video coding performance.

In our current research, before tackling the problem of optimal resource allocation over wireless DS-CDMA VSNs, we deal with the optimal IDR-frame placement during video sequences encoding. More precisely, we experiment on the GOP length, i.e., the distance between two consecutive IDR-frames, during the encoding process of the video sequences, assuming that the nodes of the considered VSN record four different levels of motion, that is low, low-medium, medium-high and high. Thus, they are clustered in four motion classes, based on the motion level included in the scenes they record.

Our previous experience with URDCs has shown that it is an efficient tool that can be used to express the expected distortion $E[D_{\text{s+c},k}]$ of node k, as a function of the bit error probability (bit error rate), P_{b}, after channel decoding, and thus, we again make use of the URDCs given by Eq. (3.12). As explained in Section 3, the parameters α and β are positive and they are highly dependent on the source coding rate as well as the motion level included in a video sequence. In addition, the same parameters depend on the considered GOP length. Hence, each considered video sequence, which is used to represent a different motion level, has its own set of α and β parameters and in our problem, it is compressed using four different GOP lengths.

Having compressed each video bitstream using four different GOP lengths, we test all possible GOP length combinations of all video bitstreams. Each different combination results in different values for the α and β parameters assigned to each motion class. For each (α, β) pairwise values, we run the optimization procedure as described in Section 3.8.1 using each of the MAD, NNBS and MTU criteria (see Section 3). The α and β values that satisfy the objective of each scheme are chosen as optimal. The GOP length for each motion class that produces the optimal values for the α and β parameters is proved to be the most efficient one, since it leads to the ultimate video quality enhancement.

4.2 Resource Allocation

Regarding the sub-problem of resource allocation, the same assumptions as described in the previous section, have also been made in this piece of work. Specifically, a wireless DS-CDMA VSN under a centralized topology has also been considered. Similarly, a target bitrate constraint is set and we assume that interference can be approximated

by AWGN, where the thermal and background noise is ignored. Thus, the energy per bit to MAI ratio is given by Eq. (3.8). For the channel coding, we have assumed RCPC codes [61], which allow the use of Viterbi's upper bounds on the bit error probability P_b. Assuming BPSK as the employed modulation scheme, P_b satisfies the inequality given by Rel. (3.1), where P_d is given by Eq. (3.2) and the complementary error function, $\mathrm{erfc}()$, is described by Eq. (3.3).

Given that the N nodes of the VSN are grouped into C motion classes and substituting P_d (Eq. (3.2)) into Rel. (3.1) (assuming it holds as equality), and then P_b into Eq. (3.12), $E[D_{\mathrm{s+c},cl}]$, for the motion class cl, takes the form of Eq. (3.13). Then, it follows that the expected distortion $E[D_{\mathrm{s+c},cl}]$ is a function of the source coding rate and channel coding rate, for class cl, and power levels of all motion classes. The utility function U_{cl}, for the motion class cl is defined as given in Eq. (3.19) and expresses the PSNR (see Eq. (3.37)).

Last, the optimization criteria used in this piece of study in order to determine the nodes' transmission parameters include the MAD described in Section 3.5.1, NBS described in Section 3.5.4 and MTU described in Section 3.5.3. It is worth mentioning that for the NBS the disagreement point is imposed by the designer of the system and each node of the network is equally advantaged. Thus, given the node clustering into C motion classes, the bargaining power bp_{cl} assigned to each motion class cl is proportional to its cardinality N_{cl}. Therefore, $bp_{cl} = N_{cl}/N$, which means that the NNBS variant of NBS is employed. In contrast, for the MTU, the unweighted version is used, which corresponds to the case where each class of nodes cl has the same weight, w_{cl}, that is $w_{cl} = 1/C$.

4.3 Reinforcement Learning

In this subsection, the formulation of the resource allocation problem as a Markov decision process [156] is introduced and we also present the reinforcement learning scheme, which is incorporated in the controller. The resource allocation problem considered in this particular study is treated as a discrete optimization problem (discrete nodes' transmission parameters). Although our previous works have tackled this problem using the heuristic optimization methodology of exhaustive search [19, 97], this is not feasible in the particular study. In our case, the controller has to select among a considerably larger set of possible variable combinations compared with our previous work. The major handicap of the ES algorithm is its computational complexity, which renders its use prohibitive in the online mode.

According to our proposed methodology, the learning optimization problem is formulated in a sequential decision framework and is modeled as an MDP [156]. Roughly speaking, an MDP involves a decision agent (controller) that repeatedly observes the current state of the controlled system, takes a decision among the ones allowed in that state, and then observes a new state as well as a reward that will drive its future

decisions. The MDP is typically denoted as a tuple

$$\{\mathcal{X}, \mathcal{U}, \mathcal{R}, \mathcal{P}, \gamma\},$$

where $\mathcal{X}$ and $\mathcal{U}$ are the state and action spaces, respectively; $\mathcal{R}$ is the reward function that specifies the importance of each transition; $\mathcal{P}$ is the state transition distribution; and $\gamma \in [0, 1]$ is the discount factor that determines the importance of the future rewards.

In the specific learning problem, we consider the state space as the Cartesian product of eight sets

$$\mathcal{X} \triangleq cb_{\mathrm{c}} \times cb_{\mathrm{f}} \times cb_{\mathrm{s}} \times cb_{\mathrm{a}} \times S_{\mathrm{c}} \times S_{\mathrm{f}} \times S_{\mathrm{s}} \times S_{\mathrm{a}}.$$

In this way, a state is represented as an eight-dimensional vector. Each of the first four variables denotes the source coding rate-channel coding rate combination, cb_{cl}, and each of the remaining variables denotes the power level, S_{cl}, for the motion class $cl \in \{c, f, s, a\}$. Moreover, the action space consists of 17 actions, two for each dimension plus one action that corresponds to the neutral case. At each time step, the controller can increase or decrease one of the state variables. Additionally, we give the ability to the controller to leave the state variables unchanged, by remaining in the same state. Regarding the reward function, it specifies the gain obtained during a transition from the current state $\boldsymbol{x}$ to the next state $\boldsymbol{x}'$, as given by the difference between the values of the objective functions corresponding to the specific states.

A stationary policy $\pi : \mathcal{X} \rightarrow \mathcal{U}$ is a mapping from states to actions and denotes a mechanism for choosing actions appropriately. The notion of *value function* is of central interest in RL tasks. Given a policy π, the value $V^\pi(\boldsymbol{x})$ of a state $\boldsymbol{x}$ is defined as the expected discounted sum of rewards, obtained starting from this state until the termination of the current episode

$$V^\pi(\boldsymbol{x}) = E_\pi \left[\mathcal{R}(\boldsymbol{x}_t) + \gamma V^\pi(\boldsymbol{x}_{t+1}) | \boldsymbol{x}_t = \boldsymbol{x} \right]. \tag{4.1}$$

This is actually a Bellman equation, which expresses a relationship between the value of a state and the values of its successor states. Similarly, the state-action value function $Q(\boldsymbol{x}, u)$ denotes the expected cumulative reward as received by taking action u in state $\boldsymbol{x}$, and following policy π

$$Q^\pi(\boldsymbol{x}, u) = E_\pi \left[\mathcal{R}(\boldsymbol{x}_t) + \gamma V^\pi(\boldsymbol{x}_{t+1}) | \boldsymbol{x}_t = \boldsymbol{x}, u_t = u \right]. \tag{4.2}$$

The objective of an RL task is to estimate an optimal policy π^* by choosing actions that yield the optimal state-action value function

$$\pi^*(\boldsymbol{x}) = \arg \max_u Q^*(\boldsymbol{x}, u).$$

The temporal difference family of algorithms [188] provides an elegant framework for solving prediction problems. The main advantage of this class of algorithms is its ability to learn directly from raw experience, without any further information. One of

the most popular TD algorithms is the SARSA algorithm [165], which is a *bootstrapping* technique. More specifically, this is an on-policy control method, which is based on the state-action value function estimation. Particularly, the predicted Q value of the new visited state-action pair and the received reward are used to calculate an improved estimate for the Q value of the previous visited state-action pair

$$\delta_t = r_t + \gamma Q(\boldsymbol{x}_{t+1}, u_{t+1}) - Q(\boldsymbol{x}_t, u_t). \qquad (4.3)$$

The above quantity is known as the one-step TD error and is used for adjusting the weights of the policy, by performing a stochastic gradient descent scheme

$$Q(\boldsymbol{x}_t, u_t) \leftarrow Q(\boldsymbol{x}_t, u_t) + \eta \delta_t, \qquad (4.4)$$

where the parameter η is the learning rate that controls the update rule. Moreover, we can combine the SARSA algorithm with the *eligibility traces*, SARSA(λ) [165], allowing the update rule to propagate the TD error backward over the current trajectory of states. It has been proved that TD algorithms are able to find an optimal policy with probability 1 [40]. This fact gives us the opportunity to find the optimal variable combination with certainty, starting from each initial state and following the learned policy.

4.4 Experimental Results and Discussion

In the specific study, we have considered that there are $N = 100$ nodes in the VSN, which are clustered into $C = 4$ motion classes, with the following proportion

- $N_a = N_s = N_f = N_c = 25$.

N_a denotes the cardinality of the class that is represented by the "Akiyo" video sequence, while N_s, N_f and N_c denote the cardinality of the class that is represented by the "Salesman", "Foreman" and "Coastguard" video sequences, respectively. Figure 4.1 below presents the different motion levels described by each considered video sequence.

Figure 4.1: Motion level represented by each video sequence.

As mentioned in Section 3, all employed video sequences are at QCIF resolution, each of 300 frames. They have been downloaded from [9] and the H.264/AVC High profile for $4:2:0$ color format video has been selected for their compression. The RCPC codes have had a mother rate of $1/4$ [61]. The target bitrate has been set to

$R_{\text{target}} = 96$ kbps that results in

$$\mathbf{R}_{\text{s+c}} = \Big\{ (32, 1/3), (48, 1/2), (64, 2/3) \Big\}$$

$$cb_{cl} = 1 \longrightarrow (32, 1/3)$$
$$cb_{cl} = 2 \longrightarrow (48, 1/2)$$
$$cb_{cl} = 3 \longrightarrow (64, 2/3).$$

We have to recall that the index cb_{cl}, denotes the admissible source coding rate-channel coding rate combinations of the cl class of nodes. For the bandwidth, the value of $W_t = 20$ MHz has been examined and the power levels have assumed discrete values from the set

$$\mathbf{S} = \{5, 7, 9, 11, 13, 15\},$$

in milliWatts (mW). Therefore, we have had to deal with a purely discrete optimization problem.

Also, the tested GOP lengths have been

$$\mathbf{GOP} = \{3, 5, 10, 30\}.$$

The vector of the disagreement point has been set to dp $= 25$ dB. For encouraging exploration in the adopted RL scheme, the initial state-action value functions have been selected optimistically. The specific optimization problem was treated as a continuous task, where we have supposed that the optimal solution is reached, in case that the controller remains in the same state for a predetermined, max number of steps (stopping criterion).

In the following, Table 4.1 presents the optimal GOP length as well as the optimal transmission parameters for all considered criteria. Although all possible combinations for the GOP length for each video sequence have been tested, we cite the following three cases.

- **Case 1:** All video sequences are compressed with GOP length 30 (relatively infrequent IDR-frame placement).

- **Case 2:** All video sequences are compressed with GOP length 3 (relatively frequent IDR-frame placement).

- **Case 3:** Each motion class selects the optimal GOP length.

As previously stated taking about the cardinality of the motion classes, index "c" denotes the "Coastguard" video sequence, index "f" the "Foreman" video sequence, index "s" the "Salesman" video sequence and index "a" the "Akiyo" video sequence. Thus, GOP_{cl}, S_{cl} and cb_{cl} refer to the GOP, power level and source coding rate-channel coding rate combination for the class cl, $cl \in \{$c,f,s,a$\}$. From the obtained results, we observe that when the optimal GOP length is selected for each motion class, we can

Table 4.1: Optimal GOP length and transmission parameters for all considered criteria.

| MAD													
Case	GOP_c	GOP_f	GOP_s	GOP_a	S_c	S_f	S_s	S_a	cb_c	cb_f	cb_s	cb_a	Total PSNR
1	30	30	30	30	15	15	5	5	3	3	1	1	118.3214
2	3	3	3	3	15	11	5	5	1	1	1	1	112.9643
3	30	3	30	30	15	13	5	5	3	1	1	1	122.5263
NNBS													
Case	GOP_c	GOP_f	GOP_s	GOP_a	S_c	S_f	S_s	S_a	cb_c	cb_f	cb_s	cb_a	Total PSNR
1	30	30	30	30	11	11	7	5	3	3	1	1	120.4887
2	3	3	3	3	15	9	13	7	1	1	1	1	118.9107
3	30	3	30	30	15	11	13	7	3	1	2	1	124.4835
MTU													
Case	GOP_c	GOP_f	GOP_s	GOP_a	S_c	S_f	S_s	S_a	cb_c	cb_f	cb_s	cb_a	Total PSNR
1	30	30	30	30	5	7	15	13	1	1	3	3	125.2693
2	3	3	3	3	11	11	13	11	1	1	1	1	121.2006
3	30	3	30	30	5	11	15	13	1	1	3	3	129.0177

receive an increase in the total PSNR (sum of the PSNRs) of all motion classes up to 4.2 dB compared to the case when GOP length is 30 and up to 9.6 dB when GOP length is 3.

Furthermore, as Fig. 4.2 shows the increase in the PSNR can be interpreted differently for each criterion. Specifically, for the MAD criterion, when optimal GOP length is selected, all video sequences increase their own utilities compared to the other two GOP length considerations, and the same holds also for the NNBS criterion. For the MTU criterion, only the "Foreman" video sequence augments its utility compared to the other two GOP length considerations. However, the total PSNR increase achieved using the optimal GOP length is 7.8 dB compared to the case of GOP length 3, and 3.7 dB compared to the case of GOP length 30, which is a considerable PSNR increase.

In the following, Fig. 4.3 compares the PSNR values achieved by each considered criterion, for all tested video sequences. The MAD favors the video sequences including high and medium-high amounts of motion, while the MTU is preferred by the nodes that capture low and low-medium amounts of motion. Regarding the NNBS, it is the criterion that presents the smallest discrepancy between the PSNR values of all video sequences, being a compromise between the values of MAD and MTU, for all video sequences.

Last but not least, Fig. 4.4 depicts the mean number of steps that the SARSA algorithm requires compared to the ES algorithm. It is obvious that SARSA needs a significantly smaller number of steps and hence less time, in order to discover the optimal combination of nodes' transmission parameters, for all considered criteria. This is attributed to the efficient way that the particular algorithm uses the received information from the environment. These two approaches, i.e., SARSA and ES, will become non comparable in the case of the online processing.
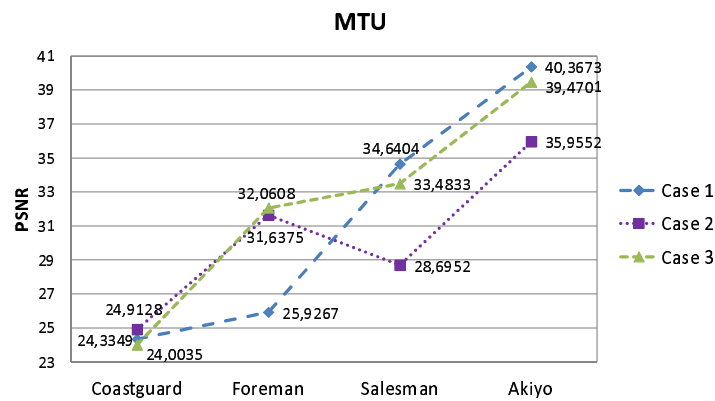
Summarizing the key points of the specific study, we have to highlight that allowing
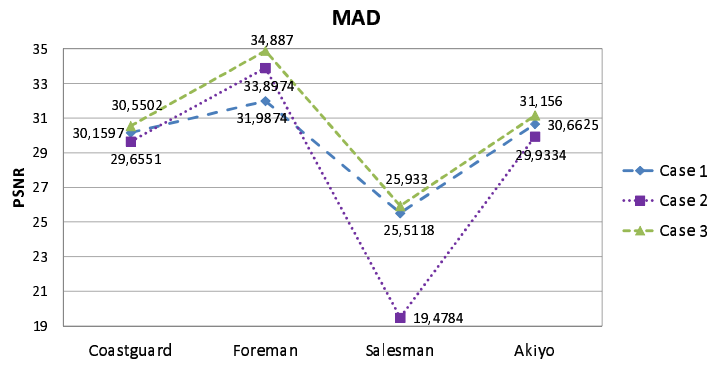
(a)
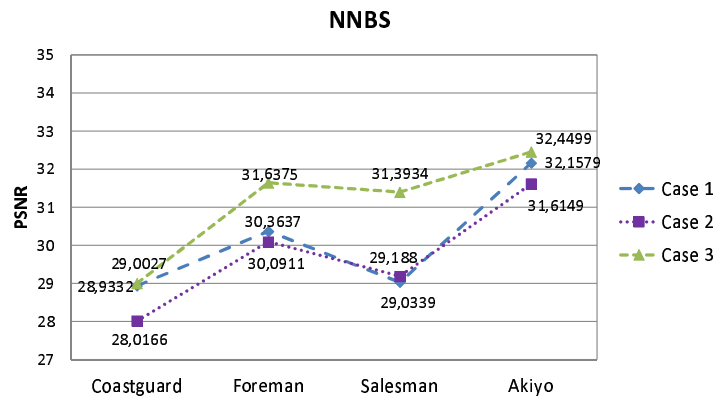
(b)

(c)

Figure 4.2: PSNR achieved by all video sequences for 3 different GOP lengths.
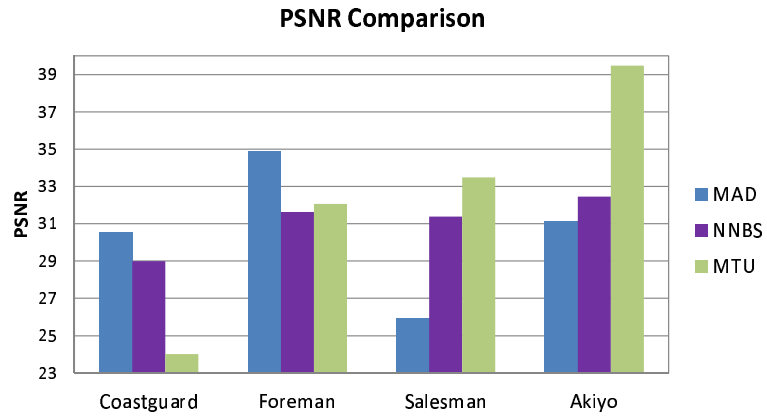
Figure 4.3: PSNR achieved by each criterion for all video sequences.
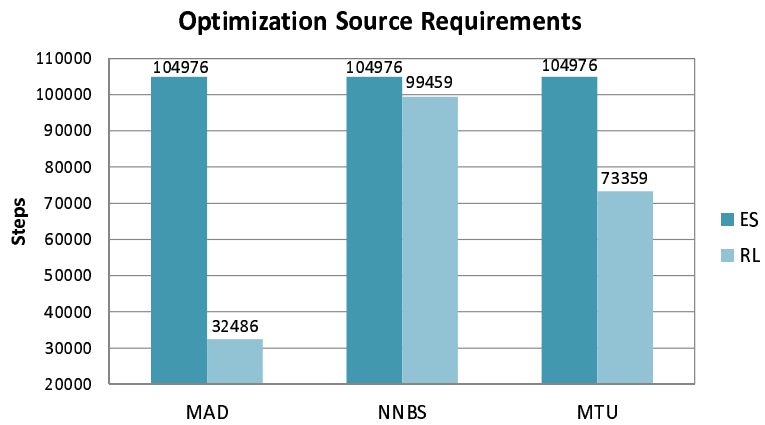
Figure 4.4: Steps required by ES and RL to reach the solution.

the nodes to select among various GOP lengths for the encoding of the video they capture, considering the motion level included in those scenes, video quality enhancement is observed as opposed to fixed GOP length considerations. Furthermore, the RL approach adopted in this study so as to tackle the discrete optimization problem is proved extremely efficient compared to the brute-force ES algorithm. Although both ES and SARSA algorithms are able to reach to the optimal solution, SARSA requires far fewer steps, making the proposed methodology applicable in online form.

CHAPTER 5

SLICE PRIORITIZATION FOR UEP TARGETING VIDEO QUALITY ENHANCEMENT

I n this chapter, we deal with a scenario where H.264/AVC video sequences are transmitted over noisy environments and the slice losses that occur degrade video quality. In an effort to guarantee as high end-to-end video quality as possible, we apply UEP based on a set of prioritized slices. Special emphasis is placed on the prioritization procedure, where the CMSE index is used as the quality "ground truth", and a number of quality-relevant features extracted from the H.264/AVC video sequences are utilized next so as to estimate this index.

5.1   CMSE and Features capturing Slice Distortion

It is well-known that the loss of a slice can introduce distortion not only in the frame where the slice loss occurs, but also in the subsequent frames belonging in the same GOP, due to error propagation. In this vein, we employ the **CMSE** index in order to account for the impact of individual slice losses on video quality, by accurately describing the error propagation within a GOP. It is computed by systematically discarding one video slice at each time and summing the MSE of the current and subsequent frames in the same GOP. Having calculated the actual values of this index by following the aforementioned procedure, we extract from the H.264/AVC video sequences the features

that are described below. Next, the feature observations as well as the actual CMSE values are given to a regression model so as to estimate the CMSE values. Therefore, based on the estimated CMSE values, which are indicators of the importance of each slice loss, we prioritize the slices accordingly.

It is worth mentioning that CMSE correlates reasonably well with subjective assessment and it does not involve the conduction of subjective tests. In fact, subjective assessment is a time-consuming and costly process that should be carefully designed and performed under specific conditions, and the human viewers are not always willing or available to perform the specific task. Therefore, in our problem CMSE is invoked so as to provide the "ground truth" of video distortion, and based on its measurements, we are able to prioritize the slices.

In this context, we summarize here the features we extract from each slice of the video sequences and which affect perceptual video quality. Specifically:

- **motx, moty** represent the mean motion vectors for the x and y directions, respectively, averaged over all the MBs in a slice. These features are calculated so as to represent the magnitude of the slice distortion in both x and y directions.

- **avginterparts** refers to the number of MB sub-partitions averaged over all the MBs in a slice. The higher the motion of a video scene, the higher the "avginterparts" value and vice versa.

- **maxresengy** is equal to the maximum residual energy of a MB, over all the MBs included in a slice. The residual energy for a MB is computed by taking the sum of squares of all its integer transform coefficients, after motion compensation. It is to be noted that if a video scene includes high motion, the "maxresengy" value is high.

- **sigmean, sigvar** correspond to the mean and variance, respectively, of the Y-component of the signal.

- **tmdr** captures the error propagation length due to a slice loss, which is heavily dependent on the considered slice type.

- **imse** captures the exact error measurement in terms of the MSE between the corresponding slices of the reconstructed frames without a slice loss and the reconstructed frames with possible slice losses, after applying error concealment at the decoder.

- **issim** captures the exact error measurement in terms of the SSIM index [210], between the corresponding slices of the reconstructed frames without a slice loss and the reconstructed frames with possible slice losses, after applying error concealment at the decoder.

In addition, it is worth mentioning that we label each slice as of IDR, P (predictive) and B (bidirectionally predictive) type.

5.2 Sparse Regression Modeling

In order to produce the CMSE estimations, we need to apply a regression model able to perform the particular task. Through the literature, there are various types of regression models that have been used in several applications, including hidden Markov models, polynomial and spline regression models, autoregressive moving average models or even Gaussian processes. However, these methods suffer from the drawback of not automatically addressing the problem of model order selection, which is a very important issue in regression. If the order of the regression model is too high, it overfits the observations and does not generalize well. On the other hand if it is too low, it might miss trends in the data.

Sparse Bayesian regression offers a convenient solution to the problem of CMSE estimation, by introducing an l_1 penalty term on the model parameters. Enforcing sparsity is a fundamental machine learning regularization principle and has been used to tackle several problems, such as feature selection. LASSO [192, 193, 127] is such a penalized regression method for simultaneous feature selection and regression coefficients estimation that has received a great deal of attention in recent years due to its generalization capabilities. The key idea is that during the training process, the least important features are assigned regression coefficients that are equal to zero, and only a few of them are retained as significant. Also, two additional LASSO features are that it is able to improve the estimation accuracy of ill-posed problems, and produces interpretable models like subset selection, by exhibiting the stability of Ridge regression [115] at the same time [192].

Let us first describe a linear regression model. A linear regression model is a model of the form

$$\hat{y}_i = w_0 + \sum_{j=1}^{m-1} w_j \phi_j(x_i) = w^\top \phi(x_i), \quad \text{for } i = 1, \ldots, n, \tag{5.1}$$

where n is the total number of observations, namely the total number of examined slices of all frames of the examined video sequences; $\hat{y}_i$ is the estimated value of CMSE at observation x_i; the basis function $\phi(x_i)$ is a vector of $m \times 1$ values at observation x_i, which includes the values for all examined features for a particular slice, and w is an $m \times 1$ vector of regression coefficients including the intercept factor w_0. Such a model is *linear* in the coefficients w.

Feature selection in regression is crucial when a variety of input features are available and we wish to select only the most important of them for the efficient estimation of a response variable. LASSO is able to simultaneously select features and produce estimations, while it also features the benefit of not only shrinking some regression coefficients close to zero, but also setting some others exactly to zero, producing interpretable models. The specific method minimizes the residual sum of squares subject to the sum of the absolute value of the regression coefficients being less than a constant. In other words, for a given positive λ value, LASSO solves the following minimization

problem

$$\min_w \left(\frac{1}{2} \sum_{i=1}^{n} \left(y_i - w^\top \phi(x_i) \right)^2 + \frac{\lambda}{2} \sum_{j=1}^{m} |w_j| \right). \tag{5.2}$$

The regression coefficients w for the LASSO methodology have no closed form and the solution involves quadratic programming techniques from convex optimization. The tuning parameter λ controls the amount of regularization, meaning that the larger the λ values are, the more regression coefficients are driven to zero, leading to a sparse model representation. Alternatively, for $\lambda = 0$, no shrinkage is performed. It is worth mentioning that in the specific study, a set of regularization coefficients λ within a predefined range have been examined in a preliminary dataset, where the $\lambda > 0$ value that corresponds to the lowest MSE for each of the models has been selected for the rest of our experiments.

Particularly, we have studied the LASSO regression through two different architectures: Global LASSO (G.LASSO) and Local LASSO (L.LASSO).

i) **G.LASSO:** A single regression model is trained for all slice types together.

ii) **L.LASSO:** Each slice type has its own regression model.

In more detail, motivated by the fact that the values for some features are closely dependent on the considered slice type, in L.LASSO we have examined the case where each slice type has its own sparse regression model, in an effort to capture more precisely the effect of a slice loss. In addition, in L.LASSO the estimation results for the separate models have been combined so as to compute the performance statistics for all slice types together.

For our experiments, we have used the database of [130]. This database includes a wide variety of scenes, such as a bird's eye view of a city, crowded areas, portraits and still water. These videos have been compressed using the JM 14.2 reference software of H.264/AVC [6]. The GOP structure has been IDR B P B with a GOP length of 20 frames. The frames have been encoded using dispersed Flexible Macroblock Ordering (FMO) and a fixed slice configuration mode, where the size of the slice (in bytes) has been predetermined by the user. At the decoder, Motion Copy Error Concealment (MCEC) has been used to conceal any slice losses in P and B frames, and spatial interpolation has been used to conceal losses in IDR frames.

As it has been stated in Section 2, the problem studied in this section has also been discussed in [130, 129]. However, the principal goal of the current study is to improve the accuracy of the CMSE estimations provided by [130, 129], through the use of LASSO, such that the video QoE to be further increased. Table 5.1 shows the regression coefficient estimates obtained by G.LASSO, L.LASSO and GLM [130]. In addition, the same table includes the selected λ values for the two LASSO-based approaches. From Table 5.1, LASSO's sparsity is obvious, since it keeps only a small subset of the features required by the GLM model [130] in order to produce CMSE estimations. We observe that only six features out of the 13 in total are employed

Table 5.1: Regression coefficients and λ values.

Features	G.LASSO	L.LASSO			GLM [130]
		IDR	P	B	
0.intercept	15.4	1.2	-2.8	22.1	94.5
1.motx	0.053	**0**	0.152	**0**	-0.298
2.moty	**0**	**0**	-0.057	-0.024	-1.15
3.avginterparts	-6.30	**0**	-3.29	-0.02	-10.8
4.maxresengy	**0**	**0**	-3.64×10^{-9}	**0**	-9.88×10^{-9}
5.sigmean	**0**	-0.0222	**0**	**0**	-0.303
6.sigvar	2.1×10^{-3}	**0**	9×10^{-4}	**0**	-2.86×10^{-3}
7.slice type f2	**0**	**0**	**0**	**0**	12.0
8.slice type f3	**0**	**0**	**0**	**0**	-19.5
9.tmdr	-0.466	**0**	-0.008	**0**	-1.30
10.imse	0.624	12.637	4.101	0.530	0.190
11.issim	**0**	**0**	**0**	-21.9	-19.1
12.imse$\times$tmdr	0.560	0.239	0.384	0.443	0.754
13.imse$\times$maxresengy	**0**	**0**	-2.40×10^{-11}	**0**	1.40×10^{-9}
λ	2.2742	2.0932	2.3002	0.8802	$-$

Table 5.2: Performance statistics.

	"Foreman"			"Akiyo"			"Tennis"		
	G.LASSO	L.LASSO	GLM	G.LASSO	L.LASSO	GLM	G.LASSO	L.LASSO	GLM
PCC	0.8108	**0.8282**	0.7217	0.9789	**0.9908**	0.9634	**0.8546**	0.8514	0.8158
SROCC	0.8745	**0.8930**	0.7586	0.7199	**0.8671**	0.7323	0.8951	**0.9038**	0.7122
RMSE	29.5563	**28.3504**	43.0587	21.7261	**12.9547**	14.5369	39.7847	**36.7304**	41.3099

by G.LASSO and similarly, six features on average for all slice types are employed by L.LASSO as well.

In the following, in order to evaluate the effectiveness of each regression model, we have employed the Pearson Correlation Coefficient (PCC), the Spearman Rank Order Correlation Coefficient (SROCC) and the Root MSE (RMSE) measures of performance, as recommended by Video Quality Experts Group (VQEG) [5]. Table 5.2 summarizes these results for the "Foreman", "Akiyo" and "Tennis" video sequences, each of 300 frames. The first two sequences are encoded at 1 Mbps and the "Tennis" video sequence at 2 Mbps, all at Common Intermediate Format (CIF) resolution (352×288 pixels). The slice size has been 300 bytes for the "Foreman" video sequence and 600 bytes for the "Akiyo" and "Tennis" video sequences, while the GOP structure has been IDR B P B, IDR P P P and IDR B P B, respectively. It is worth mentioning that the specific sequences have not been a part of the database used for models' training. In contrast, they have been used solely for models' testing.

A close inspection of the results of Table 5.2 for the tested video sequences reveals

that both LASSO architectures are able to provide accurate CMSE estimations, as it is clear from the provided PCC values. Also, the SROCC results show that the CMSE estimations are monotonically related with the measured values, following an increasing monotonic trend, and the small RMSE values signify that the CMSE estimations closely follow the measured ones. A comparison between G.LASSO and L.LASSO models justifies our choice of building a regression model, separately for each considered slice type. As it is evident from the same table (Table 5.2), improved performance statistics are obtained in such a case, as compared to the coarser approach of building a single regression model for all slice types together. In addition, it is to be noted that our proposed models behave better when the "Akiyo" video sequence is assessed compared to the "Foreman" and "Tennis" video sequences, in terms of the PCC and RMSE results, while "Tennis" gathers better SROCC results, meaning that there is a higher monotonic relationship between the measured and estimated CMSE results as compared to both "Foreman" and "Akiyo". In addition, both LASSO approaches are more efficient in estimating CMSE compared to GLM, in terms of all examined measures of performance for the "Foreman" and "Tennis" and the PCC for "Akiyo" video sequences. More obvious is this difference when "Foreman" video sequence is investigated, while even for the case of "Akiyo", L.LASSO is able to guarantee improved results as compared to GLM.

In the following, Figs. 5.1, 5.2 and 5.3 graphically illustrate the measured CMSE values as well as the estimated CMSE values achieved by each regression model, for the "Foreman", "Akiyo" and "Tennis" video sequences, respectively, when each of the slices consisting a video sequence is assumed to get lost. All presented figures verify our conclusions about the efficiency of the proposed schemes drawn earlier by examining the performance statistics for these video sequences.

5.3   Slice Prioritization

Based on the measured and estimated CMSE values using each regression model, we classify the slices into four priority classes applying a quartile-based prioritization scheme. Our ultimate goal is to assign different channel coding rates to each differently prioritized class in order to enhance the video quality that reaches the end-user.

With regard to the QBP procedure, it can be described as follows. The CMSE values (measured and estimated) are sorted in ascending order and we calculate the median value, which is the middle value of the dataset. The same procedure is also followed with the lower and upper half of the dataset. Therefore, the 75th, 50th and 25th percentiles that result, split the CMSE values into four priority classes. The class including the highest CMSE values corresponds to the 1st priority class, the class including the CMSE values between the 75th and 50th percentiles corresponds to the 2nd priority class, the 3rd priority class includes the CMSE values that fall within the 50th and 25th percentiles, while the class with the lowest CMSE values represents the 4th priority class. Figure 5.4 below presents a typical QBP scheme.
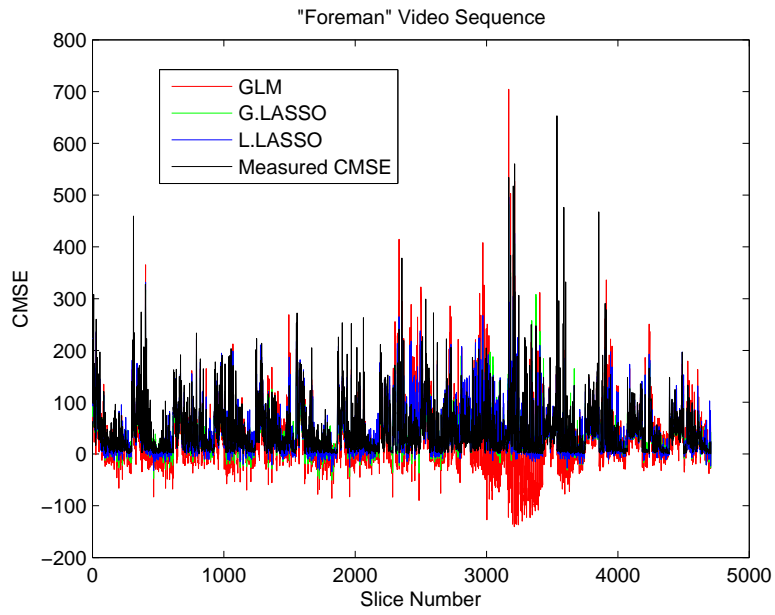
Figure 5.1: Measured versus estimated CMSE for "Foreman" video sequence.
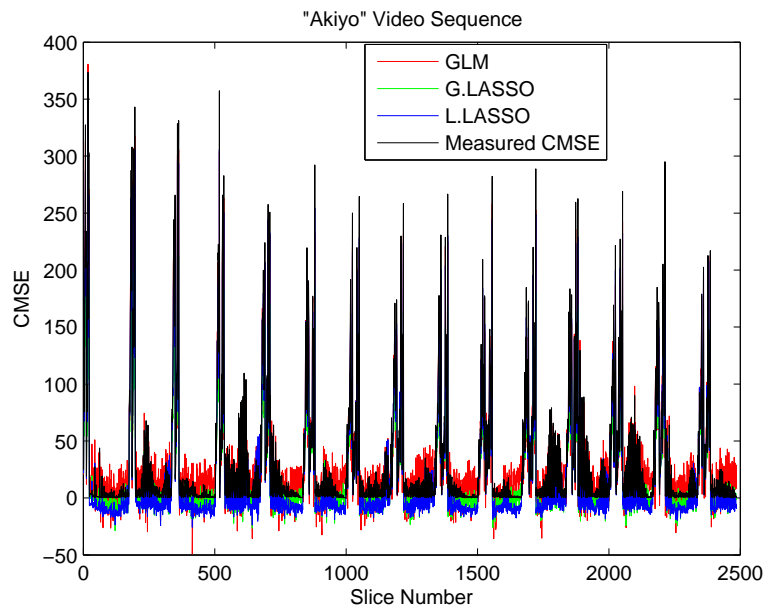

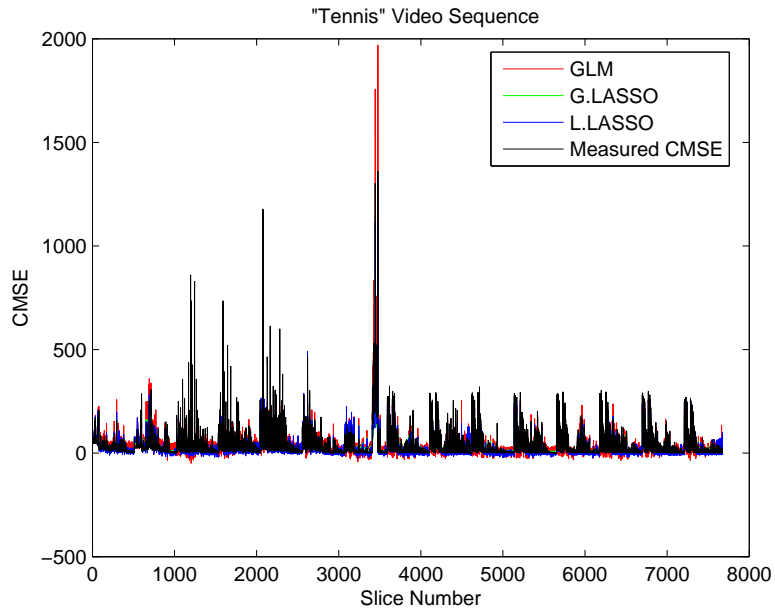Figure 5.2: Measured versus estimated CMSE for "Akiyo" video sequence.

117

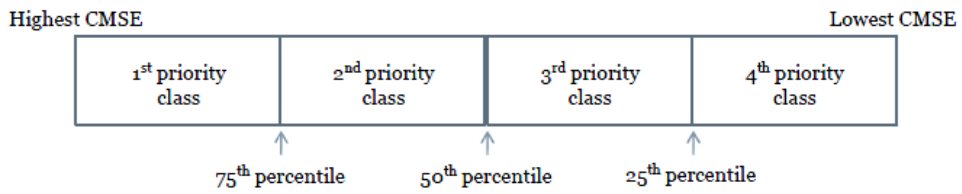Figure 5.3: Measured versus estimated CMSE for "Tennis" video sequence.

Figure 5.4: A quartile-based prioritization scheme.

118

Since QBP is conducted on both the measured and estimated CMSE values, the performance comparison between measured CMSE values and estimated CMSE values using the GLM as well as both LASSO approaches is straightforward. Let us assume that a slice is assigned a priority p, for $p = 1, 2, 3, 4$, based on its measured CMSE value. We consider a first degree misclassification error if the priority assigned to the corresponding slice based on its estimated CMSE value is $p \pm 1$ and second and third degree misclassification errors, when the assigned slice priorities based on the estimated CMSE values are $p \pm 2$ and $p \pm 3$, respectively. Two interesting remarks are the following: i) a first degree slice misclassification error is less important than a second or third degree misclassification error, since it represents a moderate CMSE estimation error compared to the other cases where the estimated values are considerably different from the measured ones and ii) the minimization of the CMSE estimation error of higher priority slices is more essential, since a possible loss of the specific slices incurs a stronger impact on perceptual quality.

Tables 5.3-5.5 depict the percentages of slice misclassifications, for the "Foreman", "Akiyo" and "Tennis" video sequences, respectively. From these tables, we confirm that the misclassification errors are low and the most common case of misclassification is observed between "neighboring" priority classes, meaning that our CMSE estimations do not differ significantly from the corresponding measured CMSE values. Additionally, the misclassification errors of the 1st priority class are considerably lower compared to those of the other priority classes, irrespectively of the followed regression approach. This fact is especially important since by providing a stronger channel coding rate during wireless transmissions to the slices belonging to the specific priority class, we assure an improved end-to-end video quality. Furthermore, as it has been expected by the results of Table 5.2, L.LASSO achieves lower slice misclassification percentages compared to G.LASSO. However, both LASSO approaches are able to gather significantly lower misclassification errors compared to the GLM, with only three isolated exceptions for the "Akiyo" video sequence.

5.4 Video Transmission Scenario

In order to assess the performance of the developed CMSE estimation models, we consider a scenario, where the measured and estimated CMSE-based prioritized bitstreams are transmitted over an AWGN channel. The goal is the optimal determination of the channel coding rate, R_p, for each of the four priorities, $p = 1, 2, 3, 4$, in each GOP of each considered bitstream that could lead to average PSNR enhancement by applying UEP [130, 129].

Let R_{target} be the transmission bitrate of the channel, in bits per second. The video is encoded at a frame rate of f_s frames per second, and the total outgoing bit budget B

Table 5.3: Percentages of slice misclassifications for the "Foreman" video sequence.

	"Foreman"					
	Priority 1			Priority 2		
	$1^{\text{st}} \to 2^{\text{nd}}$	$1^{\text{st}} \to 3^{\text{rd}}$	$1^{\text{st}} \to 4^{\text{th}}$	$2^{\text{nd}} \to 1^{\text{st}}$	$2^{\text{nd}} \to 3^{\text{rd}}$	$2^{\text{nd}} \to 4^{\text{th}}$
G.LASSO	4.59%	0.55%	0.00%	4.50%	5.35%	0.19%
L.LASSO	4.08%	0.96%	0.00%	4.63%	3.99%	0.64%
GLM	5.44%	1.17%	0.11%	5.37%	5.67%	2.55%
	Priority 3			Priority 4		
	$3^{\text{rd}} \to 1^{\text{st}}$	$3^{\text{rd}} \to 2^{\text{nd}}$	$3^{\text{rd}} \to 4^{\text{th}}$	$4^{\text{th}} \to 1^{\text{st}}$	$4^{\text{th}} \to 2^{\text{nd}}$	$4^{\text{th}} \to 3^{\text{rd}}$
G.LASSO	0.55%	4.65%	7.24%	0.08%	0.81%	6.54%
L.LASSO	0.38%	4.74%	4.74%	0.02%	0.45%	4.91%
GLM	0.87%	5.99%	8.26%	0.47%	2.17%	8.28%

Table 5.4: Percentages of slice misclassifications for the "Akiyo" video sequence.

	"Akiyo"					
	Priority 1			Priority 2		
	$1^{\text{st}} \to 2^{\text{nd}}$	$1^{\text{st}} \to 3^{\text{rd}}$	$1^{\text{st}} \to 4^{\text{th}}$	$2^{\text{nd}} \to 1^{\text{st}}$	$2^{\text{nd}} \to 3^{\text{rd}}$	$2^{\text{nd}} \to 4^{\text{th}}$
G.LASSO	1.73%	0.00%	0.00%	1.69%	4.26%	5.43%
L.LASSO	1.33%	0.00%	0.00%	1.33%	3.90%	1.01%
GLM	2.41%	0.04%	0.12%	2.33%	5.71%	5.59%
	Priority 3			Priority 4		
	$3^{\text{rd}} \to 1^{\text{st}}$	$3^{\text{rd}} \to 2^{\text{nd}}$	$3^{\text{rd}} \to 4^{\text{th}}$	$4^{\text{th}} \to 1^{\text{st}}$	$4^{\text{th}} \to 2^{\text{nd}}$	$4^{\text{th}} \to 3^{\text{rd}}$
G.LASSO	0.04%	5.71%	10.70%	0.00%	3.94%	12.19%
L.LASSO	0.00%	4.02%	9.25%	0.00%	0.88%	9.37%
GLM	0.24%	7.72%	7.80%	0.00%	3.50%	10.02%

Table 5.5: Percentages of slice misclassifications for the "Tennis" video sequence.

	"Tennis"					
	Priority 1			Priority 2		
	$1^{\text{st}} \to 2^{\text{nd}}$	$1^{\text{st}} \to 3^{\text{rd}}$	$1^{\text{st}} \to 4^{\text{th}}$	$2^{\text{nd}} \to 1^{\text{st}}$	$2^{\text{nd}} \to 3^{\text{rd}}$	$2^{\text{nd}} \to 4^{\text{th}}$
G.LASSO	4.03%	0.30%	0.00%	4.03%	4.77%	0.29%
L.LASSO	4.05%	0.34%	0.00%	3.95%	5.15%	0.36%
GLM	5.58%	0.64%	0.05%	4.89%	5.75%	3.69%
	Priority 3			Priority 4		
	$3^{\text{rd}} \to 1^{\text{st}}$	$3^{\text{rd}} \to 2^{\text{nd}}$	$3^{\text{rd}} \to 4^{\text{th}}$	$4^{\text{th}} \to 1^{\text{st}}$	$4^{\text{th}} \to 2^{\text{nd}}$	$4^{\text{th}} \to 3^{\text{rd}}$
G.LASSO	0.26%	4.43%	7.49%	0.04%	0.63%	7.11%
L.LASSO	0.38%	4.82%	5.43%	0.07%	0.59%	5.15%
GLM	1.08%	5.28%	10.34%	0.30%	3.47%	10.32%

for a GOP of length L_G, is given by

$$B = \frac{R_{\text{target}} L_G}{f_s}.$$

The RCPC code rates [61] are chosen from a candidate set $\mathbf{R}_c$ of punctured code rates, while the expected video distortion within a GOP is the sum of the slice loss distortion over the AWGN channel. The expected distortion of the jth slice depends on the CMSE due to its loss, $D(j)$, and the slice loss probability for a given channel Signal to Noise Ratio (SNR). The slice loss probability depends on i) the slice size, $S(j)$, and ii) the bit error probability, p_b, after channel decoding. Likewise, the p_b depends on the channel SNR and the channel coding rate, $R_p \in \mathbf{R}_c$, for a given priority p.

Accordingly, the optimization problem is formulated as follows [130, 129]

$$\{R_1^*, R_2^*, R_3^*, R_4^*\} = \arg \min_{R_1, R_2, R_3, R_4} \left\{ \sum_{p=1}^{4} \sum_{j=1}^{n_p} \left[1 - (1 - p_b(SNR, R_p))^{\frac{S(j)}{R_p}} \right] D(j) \right\} \quad (5.3)$$

subject to the following constraints

(1) $\sum_{p=1}^{4} \sum_{j=1}^{n_p} \frac{S(j)}{R_p} \leqslant B$

(2) $R_1^* \leqslant R_2^* \leqslant R_3^* \leqslant R_4^*.$

Here n_p is the number of slices of priority p. The first constraint is the channel bitrate constraint and the second constraint guarantees that a higher priority CMSE class is assigned at least an equal or a stronger channel coding rate compared to a lower CMSE class. The resulting optimization problems are tackled using the genetic algorithms toolbox available in Matlab [118], while the optimization is performed separately for each GOP of the video sequence, in order to avoid overflow at each second (one GOP lasts 2/3 second).

The channel bitrate has been set equal to $R_{\text{target}} = 2$ Mbps, and the GOP consisted of 20 frames. The RCPC code rates that each priority class could be assigned are given by the set

$$\mathbf{R}_c = \{8/9,\, 8/10,\, 8/12,\, 8/14,\, 8/16,\, 8/18,\, 8/20,\, 8/22,\, 8/24,\, 8/26,\, 8/28,\, 8/30,\, 8/32\}.$$

The mother code rate has been $1/4$ with memory $\text{M} = 4$ and puncturing period $\text{P} = 8$.

Figure 5.5 presents the average PSNR values for the "Foreman" video sequence computed over 100 realizations, for each given SNR value $[0, 5]$ dB. Apparently, as we observe from the simulation results, the PSNR reduces for a lower SNR channel consideration, due to a higher slice loss probability caused by more channel errors. In addition, the higher the channel SNR the closer the estimated CMSE values are to the measured ones. Moreover, we can see that when UEP is applied to the CMSE estimations achieved by L.LASSO, the average PSNR values are more close to the measured ones, as opposed to G.LASSO case, while this is more obvious for low channel SNR (0

dB and 1 dB). It is to be noted that for channel SNR 4 dB and 5 dB, both G.LASSO and L.LASSO achieve virtually identical performance with the measured CMSE case, as Fig. 5.5 shows. Compared to the GLM, it seems that for low SNR L.LASSO behaves better, offering CMSE estimations which are more close to the measured ones, while for larger SNR values a competitive performance is observed for all applied regression models.
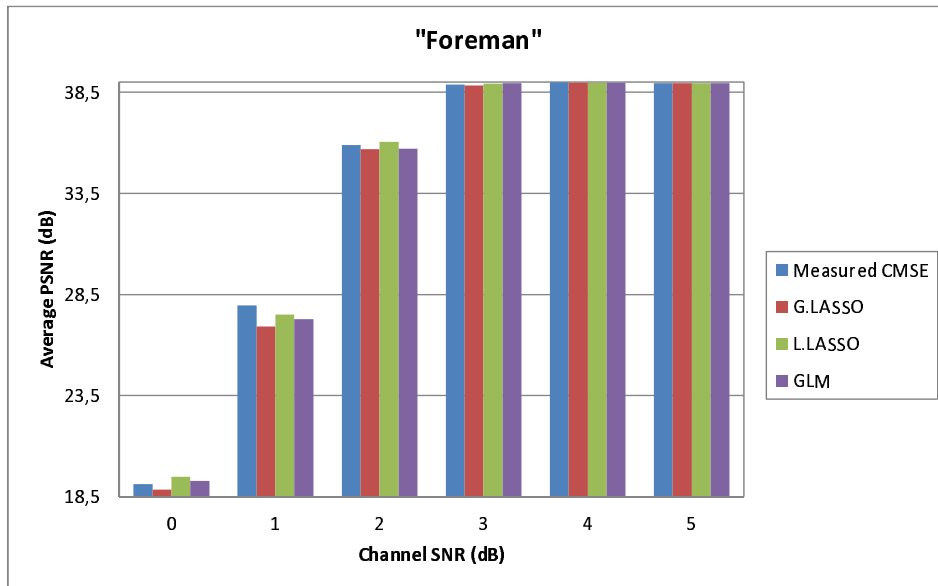
Figure 5.5: Average PSNR performance for "Foreman" video sequence transmitted over an AWGN channel.

Summarizing the key points of the problem studied in this section, we infer that all employed features used for making CMSE estimations are extremely suitable as well as both LASSO approaches and especially the L.LASSO variant is very efficient towards the same purpose. As it was expected, L.LASSO is proved to be a better choice than G.LASSO and mainly than GLM, for enhancing further the accuracy of CMSE estimations. The slice misclassification percentages are low, the performance statistics high, and the average PSNR values of the video transmission scenario achieved by the proposed models are virtually identical to the PSNR values resulting from the measured CMSE case, especially for a channel SNR of 3 dB or higher.

CHAPTER 6

PERCEPTUAL VIDEO QUALITY ESTIMATION

6.1 Features related to Perceptual Video Quality

6.2 Video Quality Estimation using Linear Regression

6.3 Test Stimuli

6.4 Performance Measures

6.5 Experimental Results and Discussion

T he video portion of the global mobile data traffic has increased tremendously and it is estimated to exceed 67% by 2018, from being 53% in 2013 [8]. Therefore, with this growing usage of videos, it is believed that the end-users are becoming more aware of the perceptual quality characteristics of video services. A required amount of compression of the raw (original) videos has to be performed in order to meet the practical limits of data storage devices and transmission channels. Depending upon its intensity, the compression can introduce different visual artifacts in a video that may decrease its perceptual quality as compared to its original version. Besides compression, video quality can also suffer from degradations due to transmission over lossy networks. Losses of video data in a network can occur for various reasons such as network fluctuations, buffer overflows, and any operational management procedures. For this purpose, in this piece of our research, we have extracted a large number of video features that are related to both sources of distortion so as to guide us in making accurate perceptual quality estimations.

6.1   Features related to Perceptual Video Quality

In the following, we describe the video features that we have used in order to model the impact of various impairments on video quality. These features are related to video

content characteristics, signal factors, error factors, motion factors, as well as to the effectiveness of the error concealment. Besides description of the examined features, we also discuss their type and the attributes through which they can be related towards video quality. In addition, the motivation for extracting the specific features is also analysed.

Examined Features

1) **Intra[%]** is the percentage of I coded MBs in a slice. It is of NR type and is related to the video content structure.

2) **I4 × 4inIslice[%]** is the percentage of MBs of size 4×4 in an I slice. It is of NR type and is related to the video content structure.

3) **I16 × 16inIslice[%]** is the percentage of MBs of size 16×16 in an I slice. It is of NR type and is related to the video content structure.

4) **IinPslice[%]** is the percentage of I coded MBs in a P slice. It is of NR type and is related to the video content structure.

5) **P[%]** is the percentage of P coded MBs in a slice. It is of NR type and is related to the video content structure.

6) **PSkip[%]** is the percentage of P MBs coded as PSkip in a slice. It is of NR type and is related to the video content structure.

7) **P16 × 16[%]** is the percentage of P MBs coded with no sub-partition of MBs in a slice. It is of NR type and is related to the video content structure.

8) **P8 × 16[%]** is the percentage of P MBs coded with 8×16 and 16×8 partition of MBs in a slice. It is of NR type and is related to the video content structure.

9) **P8 × 8[%]** is the percentage of P MBs coded with 8×8 partition of MBs in a slice. It is of NR type and is related to the video content structure.

10) **P8 × 8Sub[%]** is the percentage of P MBs coded with 8×8 in a sub-partition of MBs in a slice. It is of NR type and is related to the video content structure.

11) **P4 × 8[%]** is the percentage of P MBs coded with 4×8 and 8×4 sub-partition of MBs in a slice. It is of NR type and is related to the video content structure.

12) **P4 × 4[%]** is the percentage of P MBs coded with 4×4 sub-partition of MBs in a slice. It is of NR type and is related to the video content structure.

13) - 20) **B_modes** correspond to the same features as given in features 5 to 12, but for B MBs. They are of NR type and are related to the video content structure.

21) - 22) $\mathbf{\Delta MV_x}$, $\mathbf{\Delta MV_y}$ are the average measures of motion vector difference values for x and y direction in a slice. They are of NR type and are related to the video content motion.

23) - 24) $\mathbf{avg(MV_x)}$, $\mathbf{avg(MV_y)}$ are the average measures of motion vector values for x and y directions in a slice. They are of NR type and are related to the video content motion.

25) $\mathbf{MV_0}[\%]$ is the percentage of motion vector values equal to zero for x and y direction in a slice. It is of NR type and is related to the video content motion.

26) $\mathbf{\Delta MV_0}[\%]$ is the percentage of motion vector difference values equal to zero in a slice. It is of NR type and is related to the video content motion.

27) Motion Intensity_1 is given by

$$\sum_{i=1}^{N} \sqrt{MV_{x_i}{}^2 + MV_{y_i}{}^2},$$

where MV_a, $a \in [x, y]$ represents the average value of motion vectors in an MB in a-direction and N is the total number of MBs in a slice. It is of NR type and is related to the video content motion.

28) Motion Intensity_2 is given by

$$\sqrt{avg(MV_x)^2 + avg(MV_y)^2}.$$

It is of NR type and is related to the video content motion.

29) - 30) $|\mathbf{avg(MV_x)}|$, $|\mathbf{avg(MV_y)}|$ are the absolute values of average motion vector values for x and y direction in a slice. They are of NR type and are related to the video content motion.

31) Motion Intensity_3 is given by

$$\sum_{i=1}^{N} \sqrt{|(MV_x)|_i^2 + |(MV_y)|_i^2},$$

where $|(MV_a)|$ represents the absolute value of motion vectors in an MB in a-direction. It is of NR type and is related to the video content motion.

32) Motion Intensity_4 is given by

$$\sqrt{|avg(MV_x)|^2 + |avg(MV_y)|^2}.$$

It is of NR type and is related to the video content motion.

33) **NotStill** is a boolean variable, which is true, if the value of "Motion Intensity_2" feature is over $1/10^{th}$ of the highest magnitude value of all sequences. It is of NR type and is related to the video content motion.

34) **HighMot** is a boolean variable, which is true, if the value of "Motion Intensity_2" feature is over $8/10^{th}$ of the highest magnitude value of all sequences. It is of NR type and is related to the video content motion.

35) - 36) **MaxResEngy, MeanResEngy** are the maximum and mean residual energy values over all the MBs of a slice. The residual energy for an MB is computed as the sum of squares of its transform coefficients. They are both of NR type and include information of the video signal.

37) **LR** is a boolean variable, which is true, if a slice is lost. It is of NR type and describes a network error.

38) **LostSinFrm** is the number of lost slices in a frame. It is of NR type and describes network errors.

39) **Height** is the vertical location of a lost slice within a frame. It is of NR type and describes the network error.

40) **TMDR** is the number of frames affected by a lost slice, due to error propagation. It is of NR type and describes the network error.

41) **SpatialExtend** is the number of consecutive lost slices in a frame. It is of NR type and describes the network error.

42) **SpatialExtend2** is a boolean variable, which is true, if SpatialExtend$= 2$. It is of NR type and describes the network error.

43) **SpatialExtendFrm** is a boolean variable, which is true, if all slices of a frame are lost. It is of NR type and describes the network error.

44) **Error1Frm** is a boolean variable, which is true, if TMDR$= 1$. It is of NR type and describes the network error.

45) **DistToRef** is the distance in frames between the current frame and the reference frame used for concealment. Based on our considered GOP pattern, P frames are concealed using images from a temporal distance of three frames ago, while both I frames and B frames are concealed using images from a temporal distance of one frame ago. It is of NR type and is related to the error concealment technique.

46) **FarConceal** is a boolean variable, which is true, if $|\text{DistToRef}| \geq 3$. It is of NR type and is related to the error concealment technique.

47) **SBM** describes the impact of the impairment on slice boundaries. It is of NR type and captures the network error.

48) - 49) **SigMean, SigVar** correspond to the mean and variance of the slice luminance. They are both of RR type and include information of the video signal.

50) - 51) **MeanIMSE, MaxIMSE** represent the mean and maximum MSE, over all MBs of a slice. They are both of RR type and they capture the error introduced in the network.

52) - 53) **MeanISSIM, MinISSIM** represent the mean and minimum SSIM, over all MBs of a slice. They are both of RR type and they capture the error introduced in the network.

Motivation Behind Specific Feature Extraction

In H.264/AVC based coding, several coding modes are typically dependent on the content of a video. Mainly, the coding starts with the prediction of one part (block) of a video frame from its adjacent frames so as to eliminate any temporal redundancies. The first frame is intra (I) coded, followed by a predetermined sequence of forward predictive (P) and bi-directional predictive (B) frames, with a periodic recurrence of I frames if required. These predictions can be applied on an MB, i.e., a 16×16 block of pixels, or on its sub-sized blocks. The available information regarding these coding modes provides an estimation of the structural content of a video. The features that we compute from the lossy bitstream and can be grouped in this category are listed from 1 to 20 as referred above. Features 1 and 5 are useful for providing relative information on the percentage of blocks whose loss can be more significant as they might be used in the prediction of other blocks. Moreover, more flexibility on the usage of bi-predictive coding leads to better compression performance. In this context, we have employed feature 13. The percentage of *intra* coded blocks in an *inter* slice may represent rapid change of spatial content in a video and it is captured through feature 4. The percentage of blocks coded as *Skip* indicates the possibility of no need for any residual or motion vector information data that in turn represents the level of structural similarity of the content between various frames of a video. Encoding blocks of size 16×16 are preferable as compared to 4×4 because generally the use of higher block sizes exhibit better compression performance. Accordingly, features $2, 3, 7 - 12$, and $15 - 20$ represent the percentages of different block sizes chosen for encoding.

In addition, inter frame prediction, which takes advantage of the temporal redundancy between neighboring frames, involves the determination of motion vector information. This information can be used to estimate the relative motion found in the blocks of different frames of a video. Besides using the absolute values of the motion vectors, a number of related statistics have been computed so as to better represent the motion content of a video (features 21 to 32 as listed above). Except for the features

$13 - 20$ and $31 - 32$, which are firstly proposed in this research, the others have been inspired from the study presented in [171].

Driven by the fact that a packet loss is significantly less visible in still video scenes [107], we propose the use of feature 33, in order to define if a video slice includes motion or not. Using the motion vector magnitude values as they have been computed by feature 28, we assume that a slice includes motion (NotStill=1), if its magnitude value is greater than $1/10^{th}$ of the highest magnitude value of all slices. Similarly, we assume that a slice includes high levels of motion (feature 34) if its magnitude value is greater than $8/10^{th}$ of the highest magnitude value of all slices. Additionally, features $35 - 36$ represent the maximum and mean residual energy over all the MBs of a slice, where the residual energy for an MB is computed as the sum of squares of its transform coefficients. These additional parameters are used in order to validate whether the calculated motion vectors represent the underlying scene motion well or not. A higher residual energy value implies that the motion vectors probably do not represent the actual scene motion well. If a slice is lost, then even after applying a concealment strategy in order to accurately estimate the lost motion vectors, the resultant slice still differs from the original. Thus, residual energy is one way to assess the magnitude of this difference [79].

Continuing with the features $37 - 47$ mentioned above, they capture the effect of a packet loss in a video sequence, under various aspects. They are all computed from the lossy bitstream, except for feature 47, which is calculated from the reconstructed video sequence after error concealment. Specifically, features $37 - 44$ [107] model the impact of a packet loss based on its frequency, location, duration etc. The use of features $37 - 38$ is proposed for a first time in the context of this research as a means of quantifying the severity of distortion introduced within a frame due to possible slice losses. The vertical location of the lost slice in a frame is represented by feature 39, where its use for quality estimation is motivated by the fact that a lost slice in the middle of a frame can have different perceptual impact as compared to a lost slice in the top or bottom of a video frame.

Except for feature 39, another content-independent feature that is used to characterize the duration of time an error persists is feature 40, and features $41 - 44$ are also video content-independent and are used since intuitively, they may help in better describing the effect of losing consecutive slices. Moreover, features $45 - 46$ are related to the concealment strategy applied to the decoder and particularly, they deal with the distance from the frame that is used as reference for the concealment of a frame impaired with a slice loss. Thus, these features take into account the considered GOP structure and size. The motivation behind extracting the specific features lies in the fact that when the image used for concealment is temporally closer to the current image, fewer temporal artifacts occur and thus, reduced impairment visibility is observed compared to the case where the image used for concealment is far away from the current image [161].

As it has been discussed earlier, even after applying an error concealment technique, imperfections in the concealed parts of a video cannot be avoided. Thus, when a slice loss occurs, we may have temporal and horizontal discontinuities between the correctly received and concealed slices, which increase the visibility of the impairment [107]. In this piece of research, having detected the location of the lost slice, we apply the Slice Boundary Mismatch (SBM) metric (feature 47), as it is described in [161], with the goal of capturing the mismatch on the boundaries between correctly received and concealed slices in the decoded frames, on a pixel-by-pixel basis.

Lastly, features $48-53$ described above are calculated on a pixel-by-pixel basis. Particularly, features $48-49$ are computed from the compression-and-network-impaired videos, while features $50-53$ from the compression-impaired and compression-and-network-impaired versions of a video. The magnitude of distortion induced by a slice loss is also influenced by the presence of luminance masking, that is the sensitivity of the human visual system to the distortion introduced in darker and brighter image areas. For this purpose, we utilize features $48-49$ in order to model the mean and variance of the luminance of the signal.

Features $50-53$ model the MSE and SSIM metrics, which are commonly used in order to characterize the error amplitude and perceptual quality. In our study, we have precomputed the MSE and SSIM values for each MB at the server side. Since it is considered that human attention is mostly drawn to worst-case errors, except for the mean MSE and SSIM values, we have also kept the maximum MSE and minimum SSIM values over all MBs in a slice. Afterwards, all MSE and SSIM block values have been averaged to obtain a representative value for each of them over each slice, and the resulting values, along with the maximum MSE and minimum SSIM for each slice have next been sent to the client's side. Thus, once it is known which slices are actually lost, we are able to know the corresponding MSE and SSIM values. This process of pre-computing and transmitting the values from the server to the client renders these features of RR type [107].

However, both MSE and SSIM metrics, where in this study play the role of some of the RR features used for model's development, present a number of weaknesses [161]. MSE cannot quantify the spatio-temporal frequency characteristics of the error and it implicitly calculates the error size and duration, being unable to capture any information about error location or pattern. Moreover, MSE captures the error between the compression-impaired and compression-and-network-impaired versions of a video, but it does not give any information about the encoded and decoded signals individually. Similarly to MSE, SSIM does not offer any information about the error size or duration. Although it gives an intuition about the signal at the location of the impairment, it does not directly measure the decoded impairment attributes. Therefore, we confirm that the extraction of each of the network-error-related features presented earlier is prudent, as each of them focuses on a different aspect of the effect incurred by a lost slice.

With regard to comparison of the proposed approach of RR-based VQA, our method has some advantages over the standardized RR model called ITU-T J.342 [10] in the following ways. J.342 is based on the edge PSNR measurement, which is performed on the edge pixels of the video being transmitted over the ancillary channel. In our case, it is required to transmit a single MSE and a single SSIM values for the whole sequence and hence, it may require less bandwidth. Our RR features are not dependent on the video content; on the other hand, edge pixels may vary for different contents (spatial details, frame resolution etc.), requiring less or more bandwidth.

In the test stimuli that we have used for our experiments, a slice of a video frame corresponds to a packet. Therefore, considering the impact of a packet loss in terms of data loss on the test stimuli, it is noted that an integral number of slices are lost as a result of a packet-loss event. In light of this, in our study, the features that are related to the occurrence of a packet loss are computed at the slice level. On the other hand, some features, such as those related to motion vectors, are more suitably computed at the MB level. Hence, we have found it reasonable to follow a bottom-up approach for computing most of these features at the MB level and subsequently, an average value is obtained at the slice level. Henceforth, we compute the average values of the slice level features to obtain their values at the frame level. Moreover, the frame-level feature values are averaged further to obtain their values at the video sequence level. For frame-level data to video-level data conversion, we have tried Minkowski summation [222] by investigating a large number of Minkowski exponents. However, we have confirmed that the overall performance of the estimation models has not been significantly improved, and thus, we eventually employ the simple arithmetic mean.

6.2 Video Quality Estimation using Linear Regression

The problem of perceptual video quality estimation based on a set of quality-relevant features is solved by building computational models that take the given set of feature values as input and produce appropriate quality estimates. The choice of a particular solution to be used for regression, linear or non-linear, depends upon the requirements of the problem under consideration as well as the tradeoff preferences between the complexity and performance of a method. However, the theory associated with linear regression is well-understood and allows for the construction of different types of easily-interpretable, stable and sparse regression models. In this piece of our research, we propose the use of the LASSO method [192, 193, 127], while OLS [22], as well as Ridge [22, 67, 115] in combination with a feature selection technique [53] have also been used in our experiments for comparison purposes.

A linear regression model is a model of the form

$$\hat{y}_i = w_0 + \sum_{j=1}^{m-1} w_j \phi_j(x_i) = w^\top \phi(x_i), \quad \text{for } i = 1, \dots, n,$$

where n is the total number of observations, namely the total number of examined slices; $\hat{y}_i$ is the estimated value of perceptual quality at observation x_i; the basis function $\phi(x_i)$ is a vector of $m \times 1$ values at observation x_i, which includes the values for all examined features for a particular slice, and w is an $m \times 1$ vector of regression coefficients including the intercept term w_0. Such a model is *linear* in the coefficients w.

6.2.1 Ordinary Least Squares Regression

Ordinary least squares regression [22] is by far the most widely used method for regression because of its ease of implementation as well as its simplicity. It determines the regression coefficients w, by solving the following minimization problem

$$\min_w \left(\frac{1}{2} \sum_{i=1}^{n} \left(y_i - w^\top \phi(x_i) \right)^2 \right) = \min_w \frac{1}{2} \|y - \Phi w\|^2. \tag{6.1}$$

From Eq. (6.1) we observe that the fitted coefficients minimize the mean squared difference between the $n \times 1$ vector y of measured perceptual quality values and the estimation vector Φw. Specifically, Φ is the $n \times m$ design matrix of the model, i.e., the matrix that includes the values for all examined features of all slices, and w is the vector of regression coefficients as it results from the solution of the problem of Eq. (6.1). Particularly, it is given by

$$w = (\Phi^\top \Phi)^{-1} \Phi^\top y. \tag{6.2}$$

In the following, the calculated weights (regression coefficients), as they result from the training phase, are applied to the testing data so as to get the estimated perceptual quality values $\hat{y}$.

However, there are two major disadvantages with this method that make its use problematic. The first is the estimation accuracy. There are cases where many of the features of the design matrix are highly correlated and thus, we can get inaccurate results for any regression coefficient assigned to an estimator. Additionally, a high degree of multi-collinearity means that the matrix Φ is not of full rank and hence, neither is $\Phi^\top \Phi$. Therefore, the inversion of $\Phi^\top \Phi$ is infeasible or the results after such an inversion may be imprecise. The other major concern of OLS is that of interpretation. Having a large number of features, we often desire to select a small subset of them, by keeping the features that capture the strongest effects towards video quality. In fact, some of the computed features may be irrelevant or noise, leading to estimation harming.

6.2.2 Ridge Regression

Data analysts often concern about the accuracy of the OLS estimates. Actually, OLS estimates often have low bias but large variance, meaning that the built model is sensitive to small changes in the training set. Hence, the estimation accuracy of the

OLS estimates could be improved by sacrificing a little bias to reduce the variance of the estimated values, achieving a smaller MSE. For this purpose, we have also employed the Tikhonov regularization, known as Ridge regression [22, 115, 67], in its standard form, which is a simple extension of the OLS regression method, easy to be computed. In order to circumvent the ill-conditioning of the design matrix, Ridge calculates the mean and standard deviation of each column vector. Next, the mean is subtracted from each observation value and then, the resulting observation values are divided by the corresponding standard deviation. In this way, the variance inflation factors of the coefficient estimates are reduced achieving more stable regression coefficients.

In essence, the coefficient estimates for the model described in Eq. (6.2) rely on the independence of the model terms. However, when terms are correlated and the columns of the design matrix Φ have an approximate linear dependence, the matrix $(\Phi^\top \Phi)^{-1}$ becomes close to singular. Thus, Eq. (6.2) becomes highly sensitive to random errors in the estimated vector $\hat{y}$. In light of this, it has been suggested in [66] that to control the inflation and general instability associated with the OLS estimates, the following minimization problem should be solved

$$\min_{w} \left(\frac{1}{2} \sum_{i=1}^{n} (y_i - w^\top \phi(x_i))^2 + \frac{\lambda}{2} \|w\|^2 \right),$$ (6.3)

which is an extension of Eq. (6.1). Thus, Ridge attempts to tradeoff the goodness of fit, as it is described by the first term of Eq. (6.3), and the penalty, as it is described by the second term of the same equation. The shrinkage parameter λ is a nonnegative regularization parameter that governs the importance of the regularization term compared to the sum of squared error and basically shrinks regression coefficient values towards zero. For $\lambda = 0$, no shrinkage is performed and the solution of the OLS is obtained, while for larger λ values, the closer to zero the regression coefficient estimates are.

In a more compact form, Eq. (6.3) can be written as

$$w = (\Phi^\top \Phi + \lambda I)^{-1} \Phi^\top y,$$ (6.4)

where I is the identity matrix. Therefore, a number of biased estimators w are obtained by augmenting the diagonal matrix $\Phi^\top \Phi$ with small positive quantities, as indicated by the λ value. In this way, the system behaves more like an orthogonal system and regression coefficient values with smaller MSEs are obtained.

Nonetheless, although Ridge (in its standard form) tackles with success the possible ill-posed problems that frequently exist in OLS regression, producing more stable models, it does not set any of the regression coefficients equal to zero; it has the tendency to make them equal so as to minimize the squared norm. This means that this technique is not appropriate for producing interpretable models, namely models that keep only the most important features, rendering them easier to use and understand. Thus, there is the risk of harming the estimations, when irrelevant or noisy features are employed. For this purpose, we have combined Ridge with a feature selection procedure [53]. This

procedure precedes Ridge so as to keep only those features that are the most influential towards making perceptual quality estimations.

Feature Selection in Ridge

We have applied sequential feature selection and particularly, forward feature selection [53] in order to select the appropriate subset of features from the initial RR and NR sets, respectively, that best estimate the actual quality values. Starting from an empty feature set, we create candidate subsets by sequentially adding each of the features not yet selected, in order of importance. For each candidate feature subset, Ridge regression is applied in order to estimate the output values, and finally, it returns the MSE value between the actual and estimated quality value. This process continues until adding more features does not further decrease MSE. Thus, the specific feature subset that results to the minimum MSE is the one that is chosen as optimal. Applications of FFS using linear regression models in similar problems of video quality estimation can be found in [204, 205].

6.2.3 Least Absolute Shrinkage and Selection Operator Regression

LASSO is a regression method that can be used for both feature selection and computation of regression coefficients. Feature selection is useful when a collection of input features is available, from which we expect to select a small subset for the efficient estimation of a response variable, e.g., the perceptual quality of a video. The particular regression technique is able to effectively address possible issues that arise when the matrix of observations is not of full rank and thus, it is infeasible to be inverted using the OLS method [22]. In addition, LASSO has the benefit of not only shrinking some coefficients close to zero, but also setting some others equal to zero, offering feature selection and producing interpretable models, at the same time. Thus, it combines the stability of Ridge [22, 115, 67] and interpretability of subset selection, at the same time.

Practically, it minimizes the residual sum of squares subject to the sum of the absolute value of the coefficients being less than a constant. Therefore, it solves the following minimization problem

$$\min_{w} \left(\frac{1}{2} \sum_{i=1}^{n} \left(y_i - w^\top \phi(x_i) \right)^2 + \frac{\lambda}{2} \sum_{j=1}^{m} |w_j| \right). \tag{6.5}$$

The shrinkage parameter λ is a nonnegative parameter that controls the amount of the regularization. As λ is increased, an increasing number of regression coefficients become equal to zero, while for $\lambda = 0$ no shrinkage is obtained. For the LASSO methodology, the regression coefficients w have no closed form and the solution involves quadratic programming techniques using convex optimization.

In addition, it is worth mentioning that the use of LASSO avoids the problem of overfitting because: i) it builds a simple model and ii) it performs regularization. It

is generally admitted that too complex models are prone to overfitting and thus, they give poor estimations. In this context, LASSO regression is able to autonomously perform feature selection within its learning process, producing estimations at the same time. Also, apart from feature selection, the specific method is able to perform l_1 regularization. Regularization works well when we have a lot of features, each of which contributes a bit to the response variable estimation and it deters overfitting since the magnitude of the regression coefficients is reduced and thus, a smoother curve for fitting the data is obtained.

6.3 Test Stimuli

As test stimuli is of paramount significance for the evaluation of any methodology under consideration, we have carefully acquired a quite targeted set of test video sequences. Specifically, the set of features described in Section 6.1 has been extracted from the test stimuli of the Ecole Polytechnique Fédérale de Lausanne (EPFL) and Politecnico di Milano (PoliMi) database [41]. The process of subjective quality assessment of the selected stimuli has been performed in the aforementioned two independent laboratories and interestingly, there has been a high consistency and correlation between the obtained MOS values. This fact highlights the useability of these data and also the reproducibility of the related results that is generally a point of concern in the research of VQA algorithms.

The original SouRCe videos (SRCs) have been selected for the representation of a variety of spatiotemporal perceptual information, as suggested by ITU-T Rec. P.910 [4]. The selected SRCs have been in raw progressive format, sampled at $4 : 2 : 0$ ratio of luma and chroma components, and have been encoded using the H.264/AVC reference software, version JM 14.2 [6], with High Profile setting. A GOP structure of IBBP with a size of 16 has been used, while each video has a duration of 10 seconds in length.

The video sequences comprising the EPFL-PoliMi's database of CIF resolution (352×288 pixels) are "Mother", "Foreman", "Paris", "News", "Mobile", "Hall", each of 298 frames at 30 frames per second (fps), and of 4CIF resolution (704×576 pixels) are "Harbour" and "Soccer" of 298 frames at 30 fps, "Parkjoy", "Crowdrun" and "Duckstakeoff" of 250 frames at 25 fps, and "Ice" of 238 frames at 30 fps. For each video sequence, a full row of MBs has been coded as a separate slice, while the bitstreams of the coded videos have been impaired by a PLR of $0.1\%, 0.4\%, 1\%, 3\%, 5\%$ and 10%. For each PLR and content, two decoded video sequences have been obtained, by reading an error pattern from a different starting point. At the decoder, MCEC has been applied. It should be noted that this database also includes the MOS values as they have been collected after subjective experiments separately conducted at EPFL and PoliMi. Further details on the generation of this dataset as well as the testing conditions can be found in [41].

It is worth mentioning that before building our models, firstly, we have standardized the values of the input features by calculating their *zscore* values; that is we have sub-

tracted the *mean* from each feature vector and the obtained values have been divided
by the *standard deviation* of the particular feature vector.

6.4 Performance Measures

According to VQEG Phase I report on the validation of reduced-reference and no-reference objective models for standard definition television [5], the performance of a quality estimation model can be evaluated by the following parameters.

1. **Pearson Correlation Coefficient** (PCC) is used to describe the accuracy of the estimation, and more particularly it measures the linear relationship between estimated and measured video quality values. It is given by

$$\text{PCC} = \frac{\sum_{i=1}^{n} (\hat{y}_i - \tilde{y})(y_i - \bar{y})}{\sqrt{\sum_{i=1}^{n} (\hat{y}_i - \tilde{y})^2 \sum_{i=1}^{n} (y_i - \bar{y})^2}}, \tag{6.6}$$

   where $\hat{y}_i, y_i$ represent the estimated and measured video quality values, respectively; $\tilde{y}, \bar{y}$ represent the mean of the estimated and measured video quality values, respectively, and n is the total number of each such value. It holds that PCC values close to 0 declare bad or no correlation and values close to 1 denote high positive correlation [65].

2. **Root Mean Squared Error** (RMSE) is used to describe the error of the estimation by calculating the difference between the estimated and measured quality values. It is given by

$$\text{RMSE} = \sqrt{\frac{1}{n} \sum_{i=1}^{n} (\hat{y}_i - y_i)^2}. \tag{6.7}$$

3. **Outlier Ratio** (OR), which measures the consistency of an objective metric and is expressed as

$$\text{OR} = \frac{\text{Number of outliers}}{\text{Total number of data points}},$$

   where

$$\text{Number of outliers} : |perr(i)| > k_2 \frac{stdDMOS(i)}{\sqrt{Nsubj}}. \tag{6.8}$$

   The amount $perr(i)$ is the estimation error between the corresponding estimated and measured value of a video sequence i. The constant k_2 is equal to 1.96 to account for 95% confidence interval, $stdDMOS(i)$ is the standard deviation of the individual scores associated with a video sequence i, and $Nsubj$ is the number of viewers per video sequence i. In the database we have used, 16 and 20 viewers have evaluated the CIF video sequences and 17 and 17 viewers have evaluated the 4CIF video sequences, after outliers removal on the results collected by EPFL

and PoliMi, respectively [41]. Thus, a data point is considered as an outlier if the absolute value of $perr(i)$ is higher than the right term of Rel. (6.8).

Furthermore, we have examined two additional measures of performance that offer information about the monotonicity and the error of the estimations in relation with the measured values, respectively. Particularly:

4. **Spearman Rank Order Correlation Coefficient** (SROCC) [65] is used to describe the monotonicity between the estimated and measured video quality and is given by

$$\text{SROCC} = 1 - \frac{6\sum_{i=1}^{n} d_i{}^2}{n(n^2 - 1)}, \tag{6.9}$$

where d_i denotes the difference in ranks between each pair of estimated and measured video quality values and n is the total number of each such value. It holds that $-1 \le \text{SROCC} \le 1$, where $\text{SROCC} = 1$ denotes a perfect positive Spearman correlation, with the estimated values being a perfect monotone function of the measured values, and vice versa for $\text{SROCC} = -1$.

5. **Mean Absolute Error** (MAE), which is given by

$$\text{MAE} = \frac{1}{n}\sum_{i=1}^{n} |\hat{y}_i - y_i|, \tag{6.10}$$

and computes the average error in video quality estimations.

6.5   Experimental Results and Discussion

A large number of experiments have been conducted in order to estimate perceptual video quality using the observations of the features described in Section 6.1 as input to the regression models described in Section 6.2. Coarsely, our experiments are split into two sets. In the first set, we have built an NR bitstream-based model, while the second group includes both NR and RR models with a slightly different setup as compared to the one of the first set. The next two subsections elaborate on each considered case.

6.5.1   NR Bitstream-based Model

The test material used in the specific set of experiments has been a subset of the database presented in Section 6.3. It includes the "Foreman", "Mother", "Paris", "Ice", "Harbour" and "Parkjoy" video sequences, corrupted with a PLR of $0.1\%, 0.4\%, 1\%, 3\%$ and 5%. All of the aforementioned video sequences have been used for models' training, except for the "Ice" video sequence, which has been used for testing the performance of the models. In the particular set of experiments, the features $1-46$ as described in Section 6.1 have been employed and thus, a NR bitstream-based model has been developed through the use of the LASSO regression model.

For comparison purposes, we have utilized the OLS regression method and except for MOS, we have built models able to estimate the SSIM [210] and VQM [1], which are known for their good correlation with subjective assessment. However, in order to evaluate the behavior of the proposed models, it was necessary to calculate the true values for each of these metrics. The computation of the actual SSIM and VQM values has been a trivial task, while we have been supplied with the MOS values from [41], where more specifically, we have employed the results obtained from EPFL.

Another issue that triggered our interest has been the determination of the λ value in Eq. (6.5). A number of 100 different λ values have been tested, while we have selected the $\lambda > 0$ value that corresponds to the lowest MSE of the first term of Eq. (6.5), for each of the MOS, SSIM and VQM models. Furthermore, in all considered regression models, we have included the intercept term, in order to absorb the bias, since we have empirically observed that its inclusion greatly improved the convergence of each considered regression model.

Tables 6.1 and 6.2 include the intercept values as well as the regression coefficient values assigned to each feature, listed from $1-46$, for the estimation of MOS, SSIM and VQM quality metrics, when OLS and LASSO regression is applied, respectively. Also, in Table 6.2 the row "λ" depicts the λ values for each model, used in Eq. (6.5). From the provided results of these tables we can see that LASSO is a much sparser approach compared to OLS. It keeps less than $1/3$ of the input features for each model, assigning zero regression coefficient values to the rest, and thus, it renders them useless. On the contrary, the OLS method assigns non zero regression coefficients to all of the features, without eliminating any possible redundancies in the input data. Therefore, it poses the risk of deteriorating the estimations of the quality values. The performance results of our proposed models in terms of all 5 indices described in Section 6.4 are tabulated in Tables 6.3 and 6.4, when OLS and LASSO regression is applied, respectively. It is to be noted that for this group of experiments, we have selected to normalize the RMSE values given by Eq. (6.7), since MOS, SSIM and VQM scores have different scales. Hence, we have computed the Normalized RMSE (NRMSE), by dividing RMSE with the range of the estimated scores of each quality metric, respectively, as NRMSE $=$ RMSE$/\hat{y}_{max} - \hat{y}_{min}$ [174]. In the same tables (Tables 6.3-6.4), the row "# Features" presents the number of features used by each regression technique for making quality estimations; that is the number of features that have been assigned a non zero regression coefficient value.

A close inspection of the results reveals that the performance statistics achieved using the LASSO method are improved compared to the results achieved by OLS. The difference between the results of the two methods is more obvious in terms of the NRMSE and MAE measures. With only an isolated exception for the PCC of SSIM where OLS offers marginally better estimation accuracy, in all other cases the results of Table 6.4 highlight the ability of LASSO in producing more precise quality estimations, carefully selecting the most influential features towards video quality, at the same time.

137

Table 6.1: Intercept and regression coefficient values achieved by OLS.

Features	MOS	SSIM	VQM
0) Intercept	-13.5897	0.4593	14.3036
1) Intra[%]	-0.0686	-0.0020	0.0781
2) I4 $\times$ 4inIslice[%]	0.9315	0.0101	-0.3498
3) I16 $\times$ 16inIslice[%]	-0.0643	-0.0058	-0.1963
4) IinPslice[%]	0.0294	-2.7338×10^{-4}	-0.0382
5) P[%]	0.0260	0.0017	-0.0598
6) PSkip[%]	0.0452	4.2590×10^{-4}	-0.0350
7) P16 $\times$ 16[%]	0.0409	-0.0013	-0.0194
8) P8 $\times$ 16[%]	0.2989	0.0046	-0.2508
9) P8 $\times$ 8[%]	-0.3785	-0.0022	0.2563
10) P8 $\times$ 8Sub[%]	0.1773	-0.0011	-0.0400
11) P4 $\times$ 8[%]	0.0603	-4.2319×10^{-4}	0.0913
12) P4 $\times$ 4[%]	-0.4568	-0.0087	0.5975
13) B[%]	0.1297	0.0059	-0.1203
14) BSkip[%]	0.0221	-1.3470×10^{-4}	-0.0060
15) B16 $\times$ 16[%]	-0.0141	-2.0760×10^{-5}	0.0241
16) B8 $\times$ 16[%]	-0.0548	4.7298×10^{-4}	6.8552×10^{-4}
17) B8 $\times$ 8[%]	0.0250	-7.1516×10^{-4}	0.0503
18) B8 $\times$ 8Sub[%]	-0.0788	-4.6022×10^{-4}	0.0657
19) B4 $\times$ 8[%]	0.2776	2.6528×10^{-5}	-0.0151
20) B4 $\times$ 4[%]	1.6525	-0.0020	-0.1985
21) ΔMV_x	0.3324	-0.0016	-0.1912
22) ΔMV_y	-5.7547	-0.0445	3.9846
23) $avg(MV_x)$	0.0627	1.8509×10^{-4}	-0.0429
24) $avg(MV_y)$	0.2392	1.9322×10^{-4}	-0.1718
25) MV_0[%]	25.2009	-0.0110	-10.7282
26) ΔMV_0[%]	-4.7468	-0.0447	4.9602
27) Motion Intensity_1	0.0011	7.6397×10^{-7}	-4.7568×10^{-4}
28) Motion Intensity_2	-0.0028	-1.8478×10^{-4}	-0.0107
29) $|avg(MV_x)|$	-0.0014	-2.2317×10^{-4}	3.5045×10^{-4}
30) $|avg(MV_y)|$	0.0036	-7.2850×10^{-4}	-0.0363
31) Motion Intensity_3	-0.0015	-9.3395×10^{-6}	4.7530×10^{-4}
32) Motion Intensity_4	-0.0337	-3.5414×10^{-4}	0.0187
33) NotStill	-430.7387	-6.8888	885.7327
34) HighMot	-93.6475	-1.4840	223.0036
35) MaxResEngy	697.6703	12.1826	-1.2948×10^3
36) MeanResEngy	-8.9787	0.0533	1.3253
37) LR	17.7861	-0.1709	0.3835
38) LostSinFrm	-8.5533	0.1025	-29.0802
39) Height	15.0957	-0.7480	-5.6862
40) TMDR	119.6991	-1.0521	131.5921
41) SpatialExtend	2.6052	0.8778	3.1276
42) SpatialExtend2	425.1855	11.9312	-1.1407×10^3
43) SpatialExtendFrm	3.8337×10^{-9}	8.4789×10^{-11}	-2.4944×10^{-8}
44) Error1Frm	-5.7420×10^{-9}	-1.5256×10^{-12}	3.1181×10^{-9}
45) DistToRef	0.8024	-0.0284	-0.0715
46) FarConceal	83.2314	-3.0833	129.9600

Table 6.2: Intercept and regression coefficient values achieved by LASSO.

Features	MOS	SSIM	VQM
0) Intercept	2.6052	0.0832	-0.1886
1) Intra[%]	**0.0000**	**0.0000**	**0.0000**
2) I4 × 4inIslice[%]	**0.0000**	2.1534×10^{-4}	0.2482
3) I16 × 16inIslice[%]	-0.0831	-0.0044	**0.0000**
4) IinPslice[%]	**0.0000**	**0.0000**	**0.0000**
5) P[%]	**0.0000**	**0.0000**	**0.0000**
6) PSkip[%]	**0.0000**	**0.0000**	**0.0000**
7) P16 × 16[%]	**0.0000**	**0.0000**	**0.0000**
8) P8 × 16[%]	**0.0000**	0.0047	**0.0000**
9) P8 × 8[%]	**0.0000**	**0.0000**	**0.0000**
10) P8 × 8Sub[%]	**0.0000**	**0.0000**	**0.0000**
11) P4 × 8[%]	**0.0000**	**0.0000**	**0.0000**
12) P4 × 4[%]	**0.0000**	**0.0000**	1.9112
13) B[%]	**0.0000**	0.0124	**0.0000**
14) BSkip[%]	**0.0000**	**0.0000**	**0.0000**
15) B16 × 16[%]	**0.0000**	**0.0000**	**0.0000**
16) B8 × 16[%]	**0.0000**	**0.0000**	**0.0000**
17) B8 × 8[%]	**0.0000**	**0.0000**	**0.0000**
18) B8 × 8Sub[%]	**0.0000**	**0.0000**	**0.0000**
19) B4 × 8[%]	**0.0000**	**0.0000**	**0.0000**
20) B4 × 4[%]	**0.0000**	**0.0000**	**0.0000**
21) ΔMV_x	**0.0000**	**0.0000**	**0.0000**
22) ΔMV_y	**0.0000**	**0.0000**	2.7188
23) $avg(MV_x)$	**0.0000**	**0.0000**	**0.0000**
24) $avg(MV_y)$	**0.0000**	**0.0000**	**0.0000**
25) $MV_0[\%]$	**0.0000**	**0.0000**	**0.0000**
26) $\Delta MV_0[\%]$	2.4405	**0.0000**	2.1070
27) Motion Intensity_1	**0.0000**	**0.0000**	**0.0000**
28) Motion Intensity_2	**0.0000**	-1.3768×10^{-6}	**0.0000**
29) $|avg(MV_x)|$	**0.0000**	**0.0000**	**0.0000**
30) $|avg(MV_y)|$	**0.0000**	**0.0000**	**0.0000**
31) Motion Intensity_3	**0.0000**	**0.0000**	**0.0000**
32) Motion Intensity_4	**0.0000**	**0.0000**	**0.0000**
33) NotStill	-1.1180	-0.2925	**0.0000**
34) HighMot	**0.0000**	**0.0000**	**0.0000**
35) MaxResEngy	3.0031×10^{-9}	**0.0000**	**0.0000**
36) MeanResEngy	-7.3627×10^{-9}	-7.6440×10^{-11}	-2.8375×10^{-8}
37) LR	-400.7000	-0.1382	951.9403
38) LostSinFrm	-8.1379	**0.0000**	1.3923
39) Height	16.4351	-0.0525	**0.0000**
40) TMDR	-14.3535	-0.5829	-19.2934
41) SpatialExtend	13.9946	-0.2504	-3.8692
42) SpatialExtend2	115.6684	1.6991	109.4864
43) SpatialExtendFrm	**0.0000**	**0.0000**	**0.0000**
44) Error1Frm	325.8795	1.8654	-985.2688
45) DistToRef	**0.0000**	0.0604	**0.0000**
46) FarConceal	452.2033	5.2607	-792.4988
λ	9.9335×10^{-5}	4.3801×10^{-6}	1.2340×10^{-4}

Table 6.3: Performance results for "Ice" using OLS.

Performance Statistics	MOS	SSIM	VQM
PCC	0.9152	0.9992	0.8786
NRMSE	1.2627	3.7919	0.4312
OR	0.2000	0.2000	0.0000
SROCC	1.0000	0.9000	1.0000
MAE	4.4618	0.1141	1.5382
# Features	46	46	46

Table 6.4: Performance results for "Ice" using LASSO.

Performance Statistics	MOS	SSIM	VQM
PCC	0.9173	0.9982	0.8800
NRMSE	0.1476	3.1699	0.2720
OR	0.0000	0.2000	0.0000
SROCC	1.0000	1.0000	1.0000
MAE	0.4287	0.0951	0.8066
# Features	13	15	12

Compared to OLS, a significantly reduced number of features is used, contributing in this way to the reduction of the problem's complexity. Hereinafter, only the specific feature subset can be used to estimate video quality in any real-time application.

Summarizing the key points of this set of experiments, we infer that LASSO is able to achieve high performance statistics using only a few features. Indicatively, we point out that only 13 features are able to produce a PCC of 0.92 with the MOS. Equally good are also the performance statistics that we have computed in order to assess our models that estimate SSIM and VQM. Despite the fact that OLS presents a competitive performance with LASSO in terms of the examined measures of performance, it employs a much larger number of features increasing the problem's complexity. Moreover, the validity of the features used by LASSO leads to a very good correlation of our models with the actual MOS, SSIM and VQM. Hence, LASSO is a good choice for building a simple and low-complexity model that offers high correlation with subjective ratings and FR metrics, when it is supplied with appropriate, quality-relevant features.

6.5.2 RR and NR Models

The test material used in the specific set of experiments includes the whole test stimuli described in Section 6.3. This means that the available data used for models' training and testing consists of 144 sequences. Specifically, for each of the 12 SRC videos, 12 different realizations of a packet-loss environment have been simulated. In addition, features $1 - 53$ as described in Section 6.1, except for the features 37 and 43, have been employed. More specifically, the NR model is based on the features $1 - 36$, $38 - 42$ and

$44-47$, while for the RR model we incorporate features $48-53$ as well. The observations of the aforementioned features are given as input to the LASSO regression method so as to estimate the MOS. For the sake of comparison with the MOS estimations given by LASSO, we have also applied Ridge regression in combination with FFS.

In order to validate the robustness of our RR and NR estimation models, we have ensured a clear distinction between the training and testing data such that no content is common between the sets. The testing set comprises all distorted versions of one SRC sequence (12 sequences) and the training set comprises of the data of test stimuli generated from the distorted versions of 11 SRCs (132 sequences). This process of splitting the dataset into training and testing sets is iterated such that each impaired sequence set from each SRC takes its place on the test set. This procedure is the well-known k-fold Cross Validation (CV) [215] (in our case 12-fold CV), where data are partitioned into k equally sized subsets and an iterative procedure is repeated k times such that $k-1$ subsets are used for training and the remaining one subset is used for testing (validation).

Once the training data are selected, the next important step is the initialization of the regularization parameter λ in Eq. (6.5). A number of 100 different λ values slightly above 0 are tested, where each different λ results in a different number of selected features. Also, the MSEs between the subjective and estimated MOS values are calculated. Therefore, by simultaneously examining the sparsity as well as the estimation accuracy in terms of MSE on each considered training set, we select the λ value among all 100 values of λ for each tested sequence that gives the best tradeoff of these conditions. Thus, using the chosen λ value, we train our models and obtain values for the regression coefficients. The obtained regression coefficient values are applied to the data of the testing set in order to get the MOS estimations. At this point, it is worth noting that in our regression models, we include the intercept term, in order to absorb the bias, since we have empirically observed that its inclusion greatly improved their convergence. In the following, using the estimated values, we are able to evaluate models' performance in comparison with the subjective MOS values. Algorithm 3 summarizes the methodology adopted in the particular set of experiments in order to develop our proposed model.

Algorithm 3 Model Development

---
loop
    if (a SRC is not tested) **then**
        Split the dataset in training set and testing set, such that the testing set includes all impaired versions of the same SRC.
        Execute exclusively on the training set.
            a. Perform LASSO regression.
            b. Determine the optimal λ value of Eq. (6.5).
            c. Using the optimal λ value, train the whole training set of LASSO model.
            d. Get regression coefficient estimates.
        Apply the regression coefficient estimates on the testing set.
        Get video quality estimations.
        Evaluate performance.
    end if
end loop

As a result of the test setup described above, a number of simulations have been performed for the RR and NR sets of video features, using the MOS values collected by both PoliMi and EPFL [41]. It holds that subjective MOS values are usually compressed at the ends of the rating scale (0 and 5), while this is not the case for objective video quality models that are unable to mimic this weakness of subjective data. Therefore, following the VQEG report on validation of objective video quality models [199], a third order monotonic mapping function has been applied on the estimated values of our models before the computation of the performance measures.

Tables 6.5 and 6.6 present the obtained results for both RR and NR models using Ridge and LASSO, for the CIF and 4CIF resolution video sequences, respectively, when the PoliMi MOS values are used. The measures of PCC, RMSE and SROCC described in Section 6.4 have been used so as to gauge the performance of our proposed models. Besides the performance measures, the chosen λ values and the number of used features (# Features) for each test sequence and proposed model are also mentioned. It is to be noted that for Ridge, we have set $\lambda = 10^{-5}$ as being a typical small positive value able to improve the conditioning of the problem and reduce the variance of the MOS estimates. Each cell of the tables cites the results when a specific SRC sequence is used to generate its 12 impaired versions and the bottom cell shows the arithmetic mean (average) of the performance over all the SRCs (of all cells) with resolution CIF or 4CIF separately. On the same tables, the related performance of PEVQ and VQM metrics is also mentioned. PEVQ is an FR metric that is a part of the ITU-T Recommendation J.247 [72]. VQM is also an FR metric that has been largely adopted in the research community for taking quality estimates [1]. In an effort to be fair when comparing the estimated and subjective MOS, we have scaled PEVQ and VQM values in the range [0, 5]. In addition, since for the VQM the smaller the value the better the video quality, we have "reversed" these values to follow the trend of MOS. It holds that comparing RR and NR models against FR metrics is a challenging task, as FR metrics have far more data to process for estimating quality. Nonetheless, as it turns out, the proposed models perform equally well, or somewhat better than the considered FR metrics. The advantage of our proposed models is more evident mainly in terms of fairly lower values of estimation error (RMSE).

One salient aspect of comparing the performance of Ridge and LASSO is the level of accuracy and sparsity offered by each solution. Observing the performance results of Tables 6.5 and 6.6, we do not confirm an advantage of a particular regression method over the other, since their statistics is similar in both CIF and 4CIF resolutions. However, for all examined cases, it can be construed that LASSO models use far less than half of the features employed by Ridge for making quality estimations.

In fact, the values of the regression coefficients are considered as an indication of feature selection or not. If the coefficient associated with a certain feature acquires a zero value, this means that the specific feature is excluded from the estimation process. On the contrary, a feature is selected, if it is assigned a non-zero regression

Table 6.5: Performance of LASSO and Ridge models and reference FR metrics for MOS collected by PoliMi [41] for CIF resolution sequences.

Method	Test Sequence	λ	# Features	PCC	SROCC	RMSE
		CIF				
NR Ridge		$1e-05$	9	0.977	0.958	0.297
NR LASSO		0.2674	2	0.979	0.972	0.284
RR Ridge	"Foreman"	$1e-05$	9	0.986	0.986	0.230
RR LASSO		0.7014	2	0.985	0.986	0.243
PEVQ		—	—	0.983	0.963	0.792
VQM		—	—	0.971	0.979	1.548
NR Ridge		$1e-05$	8	0.945	0.937	0.420
NR LASSO		0.2711	3	0.965	0.979	0.337
RR Ridge	"Hall"	$1e-05$	11	0.976	0.923	0.278
RR LASSO		0.7099	2	0.980	0.944	0.258
PEVQ		—	—	0.944	0.818	0.753
VQM		—	—	0.940	0.895	1.064
NR Ridge		$1e-05$	5	0.960	0.979	0.356
NR LASSO		0.2697	2	0.979	0.972	0.259
RR Ridge	"Mobile"	$1e-05$	7	0.993	0.965	0.153
RR LASSO		0.7070	2	0.993	0.979	0.153
PEVQ		—	—	0.969	0.937	0.404
VQM		—	—	0.979	0.972	0.575
NR Ridge		$1e-05$	7	0.945	0.937	0.318
NR LASSO		0.2757	3	0.929	0.937	0.359
RR Ridge	"Mother"	$1e-05$	16	0.962	0.965	0.265
RR LASSO		0.7381	2	0.966	0.944	0.252
PEVQ		—	—	0.967	0.944	0.803
VQM		—	—	0.963	0.930	2.003
NR Ridge		$1e-05$	14	0.965	0.965	0.345
NR LASSO		0.2704	1	0.969	0.972	0.328
RR Ridge	"News"	$1e-05$	9	0.987	0.979	0.212
RR LASSO		0.7093	2	0.992	0.979	0.164
PEVQ		—	—	0.976	0.972	0.343
VQM		—	—	0.980	0.979	1.078
NR Ridge		$1e-05$	8	0.972	0.916	0.320
NR LASSO		0.2677	2	0.978	0.944	0.286
RR Ridge	"Paris"	$1e-05$	13	0.990	0.972	0.188
RR LASSO		0.7035	2	0.989	0.972	0.205
PEVQ		—	—	0.976	0.951	0.587
VQM		—	—	0.970	0.951	0.833
NR Ridge		—	9	0.961	0.949	0.343
NR LASSO		—	2	0.967	0.963	0.309
RR Ridge	Average	—	11	0.982	0.965	0.221
RR LASSO		—	2	0.984	0.967	0.213
PEVQ		—	—	0.969	0.930	0.614
VQM		—	—	0.967	0.951	1.183

Table 6.6: Performance of LASSO and Ridge models and reference FR metrics for MOS collected by PoliMi [41] for 4CIF resolution sequences.

		4CIF				
Method	Test Sequence	λ	# Features	PCC	SROCC	RMSE
NR Ridge		$1e-05$	11	0.981	0.972	0.223
NR LASSO		0.2677	3	0.969	0.986	0.283
RR Ridge	"Crowdrun"	$1e-05$	10	0.988	0.986	0.179
RR LASSO		0.7178	2	0.987	0.986	0.187
PEVQ		–	–	0.966	0.986	0.317
VQM		–	–	0.991	0.986	0.343
NR Ridge		$1e-05$	4	0.968	0.993	0.313
NR LASSO		0.2714	2	0.973	1.000	0.289
RR Ridge	"Duckstakeoff"	$1e-05$	6	0.996	1.000	0.113
RR LASSO		0.7070	2	0.991	1.000	0.163
PEVQ		–	–	0.994	0.996	0.329
VQM		–	–	0.988	0.996	0.489
NR Ridge		$1e-05$	6	0.978	0.916	0.203
NR LASSO		0.2754	2	0.978	0.930	0.204
RR Ridge	"Harbour"	$1e-05$	5	0.973	0.923	0.224
RR LASSO		0.7239	1	0.981	0.930	0.186
PEVQ		–	–	0.960	0.930	0.419
VQM		–	–	0.974	0.930	0.509
NR Ridge		$1e-05$	14	0.962	0.958	0.338
NR LASSO		0.2706	3	0.970	0.965	0.303
RR Ridge	"Ice"	$1e-05$	10	0.977	0.979	0.267
RR LASSO		0.7133	2	0.979	0.979	0.256
PEVQ		–	–	0.977	0.972	0.368
VQM		–	–	0.975	0.972	0.633
NR Ridge		$1e-05$	6	0.954	0.951	0.317
NR LASSO		0.2749	2	0.972	0.965	0.247
RR Ridge	"Parkjoy"	$1e-05$	3	0.975	0.979	0.234
RR LASSO		0.7180	2	0.984	0.979	0.188
PEVQ		–	–	0.979	0.972	0.446
VQM		–	–	0.981	0.979	0.550
NR Ridge		$1e-05$	4	0.991	0.979	0.162
NR LASSO		0.2712	3	0.987	0.993	0.192
RR Ridge	"Soccer"	$1e-05$	4	0.996	1.000	0.108
RR LASSO		0.7120	2	0.996	0.993	0.103
PEVQ		–	–	0.989	0.996	0.382
VQM		–	–	0.991	0.986	0.672
NR Ridge		–	8	0.973	0.962	0.261
NR LASSO		–	3	0.975	0.973	0.253
RR Ridge	Average	–	6	0.984	0.978	0.188
RR LASSO		–	2	0.986	0.978	0.181
PEVQ		–	–	0.977	0.976	0.377
VQM		–	–	0.983	0.975	0.533

coefficient value. As the values of the input features are normalized to the same scale, a higher value of a coefficient implies higher significance of the related feature, and vice versa. Moreover, the features that are associated with positive-signed coefficients are considered to cause an increase in quality if their values are increased. In contrast, the features associated with negative-signed coefficients are considered to decrease quality if their values are increased.

Continuing, it is intriguing to investigate the specific features that are actually selected using LASSO and Ridge combined with FFS. For the LASSO case, the number of selected features when each different SRC is tested, ranges between one and three for the NR case and between one and two for the RR case. The corresponding lists with all different selected features include only features $13, 40$ and 41 (see Section 6.1) for the NR case and features 40 and 53 (see Section 6.1) for the RR case, respectively. This means that only three out of 45 initially extracted NR features and only two out of 51 initially extracted RR features are eventually used for making video quality estimations.

Regarding the case where Ridge along with FFS is applied, the number of selected features when each different SRC is tested, ranges between four and 14 for the NR case and between three and 16 for the RR case. The corresponding lists with all different selected features include the features $1-4, 6-8, 10, 12-27, 30-36, 38-42, 44-47$ (see Section 6.1) for the NR case and the features $1-3, 5-8, 11-13, 15-26, 28, 30, 34, 36, 38-41, 44-45, 47-53$ (see Section 6.1) for the RR case, respectively. This means that 40 out of 45 initially extracted NR features and 39 out of 51 initially extracted RR features are used for making video quality estimations.

Therefore, we confirm that LASSO keeps the advantage of providing much more sparse solutions as compared to Ridge combined with FFS as well as the same three and two specific features that are selected in the NR and RR case, respectively, are able to make estimations irrespectively of the considered training set. Hence, it is clear that LASSO uses a significantly smaller variety of features for making MOS estimations as compared to feature selection combined with Ridge regression. Additionally, it is interesting to highlight that all three features selected by NR LASSO are extracted from the lossy bitstream and do not require its decoding. Therefore, the lower computational complexity of LASSO is another benefit over Ridge using FFS and hence, it can be used as an efficient solution for video quality estimation as well as feature selection, maintaining impressively good performance statistics.

Regarding the performance comparison of the RR and NR models, from the individual results for each SRC it is observed that RR models have slightly better performance than the corresponding NR models, as expected, and it is indicated through the correlation coefficients as well as the estimation errors, regardless of the regression method used, with only some isolated exceptions. This inference about the effectiveness of the RR models is also verified by the average results, depicted on the bottom cells of Tables 6.5 and 6.6, respectively. However, we have to mention that the estimation accuracy in NR cases is very promising as well and hence, the proposed list of NR
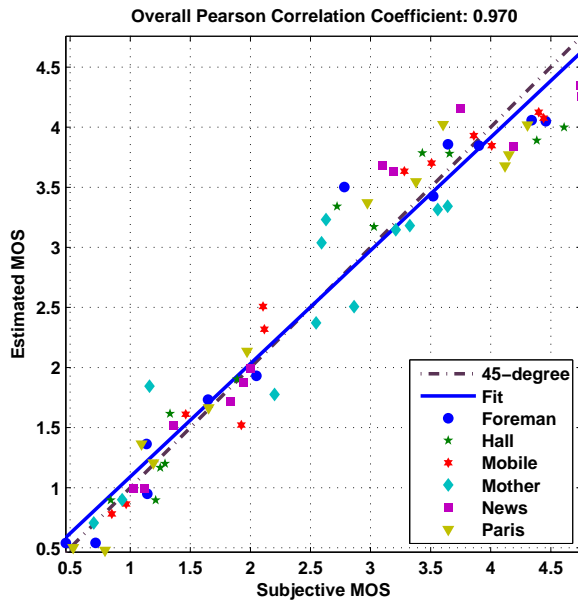
features presents an acceptable solution for reference-free quality estimation of video transmissions over lossy networks.

In addition to the earlier mentioned detailed statistics of Tables 6.5 and 6.6, the overall performance of RR LASSO and NR LASSO models is shown as scatter plots in Fig. 6.1, for both CIF and 4CIF resolutions. The values of the "overall" performance, as indicated in these plots, are obtained by comparing the values of estimated and subjective MOS, when all examined sequences of a specific spatial resolution are considered as a whole. The scatter plots not only indicate a very high overall performance in each case, and especially in the RR case, but they also show that the quality estimation for the "Mother" sequence seems to be difficult for NR LASSO, while RR LASSO manages its estimation in a more efficient way. Similarly, "Crowdrun", "Ice" and "Parkjoy" sequences are relatively challenging in the case of NR LASSO-based estimation. However, as it can be seen in Fig. 6.1(d) for the case of RR LASSO, the MOS estimation is more precise for these sequences. Thus, the slight advantage of using an RR model over an NR model in terms of performance is more obvious from these plots, at the cost of maintaining an ancillary channel and the risk of a possible failure in RR data delivery to the receiver's end.
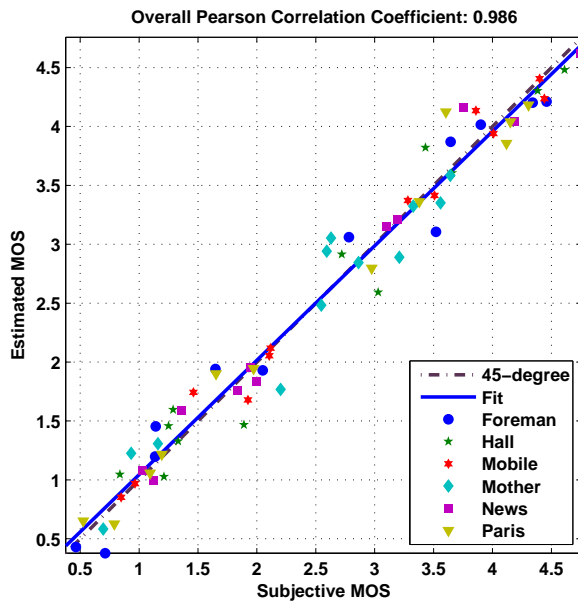
Comparison with Related Works

The results produced in this piece of study have been compared with the results of existing publications that address video quality estimation problems in FR, RR, and NR modes. It is to be noted that Table 6.7 includes the results of the PCC, SROCC, and RMSE performance measures as they have been calculated in "overall" fashion, that is, by considering estimated and subjective MOS values for all examined sequences as a whole, for each particular resolution. Also, the same table gives an intuition about the average number of features (Ave. # Features) required by each employed model for the estimation of MOS.
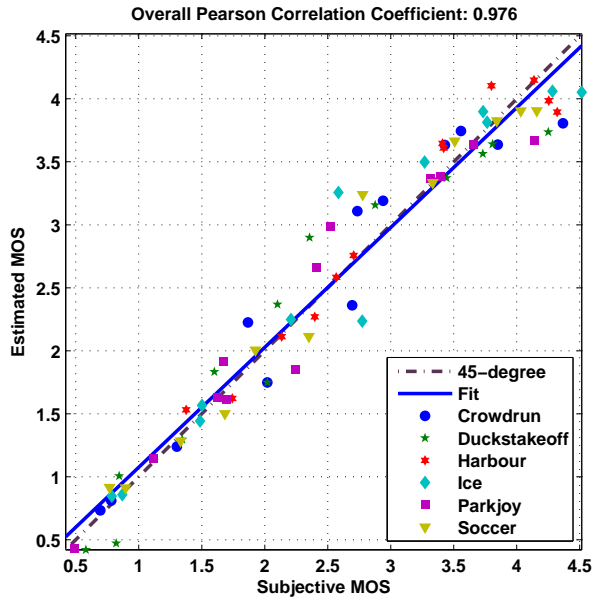
From Fig. 8(b) of the work presented in [124], we observe that the PCC and SROCC values using the EPFL-PoliMi video database [41] are between 0.85 and 0.95, for all proposed Q-mentioned FR metrics (Qvector, Qcsiq, Qtid, Qlive). These results have been generated by using the singular values and vectors so as to quantify the visual distortions and training the aforementioned models on the CSIQ [103], TID [155] and LIVE [177] databases, while the same models have been tested on the CIF sequences of EPFL-PoliMi database. Therefore, due to the fact that the models of [124] are of FR type and they have been trained on image databases only, the comparison with the results of this piece of study is not completely fair. Nonetheless, for the purpose of completeness we point out that, despite the fact that we propose RR and NR models, and thus the task of making estimations is more challenging as compared to a FR model, from Table 6.7 we infer that our NR LASSO model offers PCC values equal to or higher than 0.960 and SROCC values equal to or higher than 0.970, and our RR LASSO model offers 0.981 and 0.974 as the least values of PCC and SROCC.

Overall Pearson Correlation Coefficient: 0.970
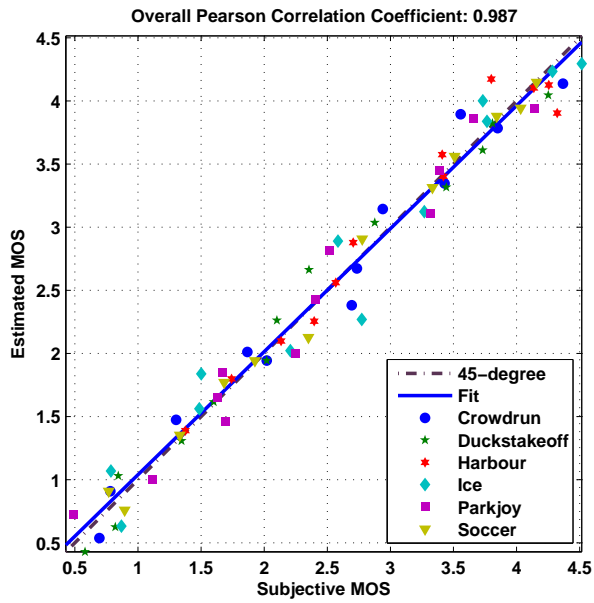
(a)

Overall Pearson Correlation Coefficient: 0.986

(b)

(c)

(d)

Figure 6.1: Overall performance of the proposed NR LASSO and RR LASSO models for CIF resolution in 6.1(a) and 6.1(b) and for 4CIF resolution in 6.1(c) and 6.1(d).

Moreover, the performance of the Fourier transform-based RR model proposed in [125] for its variant Q_{combined} that offers the best results has been compared against our proposed models in the same table (Table 6.7). In [125] the basic idea has been the comparison of the phase and magnitude of the reference and distorted images so as to compute the quality score. We realize that the model presented in [125] can be considered more general as it has been trained and tested on different media contents in contrast to the models of our work where test stimuli from one source has been employed. Thus, this implies that the worse performance of the models in [125] for the CIF resolution sequences of the EPFL-PoliMi's database [41] may be justified as they have used a broader dataset for models' development. However, from the provided experimental results, we confirm the superiority of the RR LASSO model that we propose, in all examined measures of performance. Interestingly, the estimation error of our model is nearly equal to half of the corresponding amount of Q_{combined} using on average over all tested sequences the same number of features (two) with the features used for the development of Q_{combined}. In addition, Table 6.7 depicts also the results of the RR Ridge model when FFS is preceded. In this case, we can see that Ridge achieves marginally (negligibly) better performance as compared to LASSO, but it requires six times more features as compared with the corresponding LASSO case.

Furthermore, another work that utilizes the EPFL-PoliMi database [41] and specifically, the CIF resolution sequences to assess the performance of the proposed NR metric is the one presented in [31]. In that work, the MOS values collected by PoliMi have been used, while the best proposed model is called "Frame-type and Error pattern dependent Packet-Loss model", as denoted by "FE-PLM" in Table 6.7, which uses five features in total for making perceptual quality estimations. It can be seen that our proposed NR LASSO model is better in terms of all examined statistics compared to those of [31]. One of the underlying reasons behind this difference in performance can be the fact that we have used a variety of features to capture various characteristics of a video including the impact of packet losses. On the other hand, the models in [31] are based on the assumption that visual quality can always be exponentially related to the PLR which, in practice, may not hold in varying bitrates and different contents [122]. Also in this case, the proposed NR LASSO model not only gathers better performance statistics as compared to FE-PLM model, but also it requires on average less than half the number of the features required by FE-PLM. Moreover, it is worth mentioning that the NR Ridge model combined with FFS also gathers better statistics as compared to the FE-PLM model, but it requires much more features for making estimations. In contrast, NR LASSO outperforms NR Ridge in terms of both performance and number of features used for estimating video quality.

Similarly, the NR model presented in [17] has been evaluated using MOS values collected at PoliMi [41]. In order to design the model, the authors have assumed that PLR and MOS can be characterized by a two-region piecewise linear relationship. Based on this assumption, a number of variants of the basic NR model have been

Table 6.7: Comparison of the overall performance of the proposed models with Ridge models and other related works.

	Based on MOS values by PoliMi				Based on MOS values by EPFL		
	CIF resolution						
Metric	NR Ridge	NR LASSO	FE-PLM [31]		RR Ridge	RR LASSO	Q_{Combined} [125]
PCC	0.963	**0.970**	0.95		0.983	0.981	0.944
SROCC	0.957	**0.970**	0.95		0.978	0.974	0.930
RMSE	0.345	**0.311**	0.43		0.244	0.259	0.446
Ave. # Features	9	**2**	5		12	2	2
	4CIF resolution						
Metric	NR Ridge	NR LASSO	G.1070E [109]	$SLR_{IP}+SLR_B$ [17]	NR Ridge	NR LASSO	G.1070E [109]
PCC	0.973	**0.976**	0.93	0.963	0.962	0.960	0.926
SROCC	0.974	**0.977**	0.91	−	0.970	0.976	0.93
RMSE	0.268	**0.256**	0.373	0.337	0.314	0.325	0.533
Ave. # Features	8	**3**	3	2	6	2	3

proposed, which differ mainly on the type of data used for estimating losses introduced by the network. The results that we have considered from [17] are based on the quality estimation using the $SLR_{IP}+SLR_B$ model variant (based on slice loss rate of I/P slices and B slices) that offers the best results. The conclusion derived after looking at the results is that also in this case, the NR LASSO model achieves better performance in terms of all examined measures of performance. However, it is to be noted that LASSO utilizes one feature more for making estimations as compared to the model proposed in [17]. On the other hand, we should notice that in this study we propose an one-region linear model in contrast to [17] where a two-region piecewise linear model has been employed. Similarly, the NR Ridge model is able to estimate video quality more accurately as compared to [17], while it requires significantly more features for making estimations. Therefore, also in this case, the NR LASSO model's advantage is obvious.

An enhanced version of the ITU-T Recommendation G.1070: Opinion model for video-telephony applications [7], called G.1070E can be found in [109]. The estimation accuracy of the G.1070E model has been validated using the 4CIF resolution sequences of the database presented in [41], with the MOS data collected from the subjective tests, conducted both at the EPFL and PoliMi institutions. Specifically, in [109] the estimation models have taken into account the video bitrate, frame rate and packet loss rate so as to measure video quality and they have been trained on a large variety of CIF resolution sequences, other than those included in the EPFL-PoliMi's database, which have been compressed at various bitrates and have been impaired with different PLRs. Comparing the performance of our proposed NR LASSO model with the G.1070E [109] model, we easily perceive a clear advantage of LASSO and a considerably better performance in terms of all presented measures of performance, either the PoliMi or the EPFL MOS values are used. Especially for the case when the EPFL MOS is employed, our proposed model is able to produce better performance statistics, requiring on average fewer features compared to G.1070E, at the same time. Regarding the NR Ridge model, it also surmounts in perfomance the model proposed in [109], but it has the disadvantage of requiring a much larger number of features as compared to [109]. Therefore,

the advantage of LASSO is prominent also in this case, despite the fact that for the case of EPFL's output Ridge offers slightly better MOS estimations.

Lastly, we have studied the performance achieved by a genetic programming-based NR regression model presented in [183]. In that work, the authors have validated the performance of their proposed model, exploiting eight different features that are influential for modeling perceptual video quality, by considering the video quality estimates and subjective MOS values together, for both the CIF and 4CIF resolution sequences of the EPFL-PoliMi's database [41]. Accordingly in this case, our NR LASSO model offers PCC and SROCC values equal to 0.973 and 0.975, respectively, as compared to the corresponding values offered by [183], which are equal to 0.881 and 0.883, respectively. Besides the better statistics achieved by our proposed model, it is important to point out that NR LASSO uses less than half the number of the features employed by the model of [183].

Summarizing the key points of this set of experiments, we confirm that LASSO is able to achieve exceptionally high and marginally better performance statistics as compared to Ridge using FFS, utilizing on average only two and three features, for the RR and NR cases, respectively. Interestingly, all the features required by the NR LASSO model for estimating MOS are extracted from the lossy bitstream, without the need for its decoding. This means that significantly less computational complexity is involved in the feature extraction process, rendering the model practical in real-time applications. In addition, the proposed LASSO models outperform a number of existing FR, RR, and NR techniques used for video quality estimation, in terms of both performance statistics and required features.

FRAMEWORK FOR IMPROVING PERCEPTUAL VIDEO QUALITY ESTIMATION

Continuous technological advancements have enabled the proliferation of streaming video services and consequently, the matter of VQA has become very popular. In this chapter, we present a novel framework for enhancing the accuracy of perceptual video quality estimations, adopting a frame-level estimation approach before producing sequence-level estimations. Moreover, a promising metric that provides values for each individual frame of a video sequence is also developed.

7.1 Frame Quality Ground Truth

In order to estimate the MOS per frame, it is necessary to obtain the ground truth for each frame, MOS_{fr}. For this purpose, we propose a mathematical tool to help us achieve our goal. Although we use the term "MOS per frame", it is clear that ground truth for each frame cannot be obtained using subjective tests where the viewers look at the whole video sequence. Thus, the proposed MOS per frame metric can be seen as a FR objective metric, which produces quality scores for each individual frame. It is important to mention that the average MOS value over all frames of a sequence, $\overline{\text{MOS}_{fr}}$, highly correlates with the actual MOS for each sequence, MOS_s, obtained via subjective tests.

The research conducted in [208] has proved that an exponential function can map PSNR to MOS with high accuracy, considering the temporal and spatial activity levels

of the video sequence in question. This function is given by

$$\text{MOS}_s = \exp\left(\frac{\text{PSNR}_s - a}{b}\right) - 1, \qquad\qquad (7.1)$$

where the parameters a and b represent the vertical shift and the steepness of the curve, respectively, and MOS_s, PSNR_s correspond to the MOS and PSNR, respectively, for each sequence.

In the specific study, we propose the use of Eq. (7.1) at the frame level in order to obtain the ground truth for the MOS per frame. For our experiments, we have employed the database of [41] for the 4CIF resolution sequences (704×576 pixels), described in Section 6.3. Thus, our dataset consists of 72 distorted versions of the six original video sequences (6 original sequences $\times$ 6 PLRs $\times$ 2 channel realizations). In addition, in [41] subjective MOS results from the EPFL and PoliMi are also provided. It is to be noted that, in this study, we have conducted experiments using both EPFL and PoliMi MOS values and a very similar performance has been observed. Due to this, we present results using only the PoliMi MOS.

Since the metric proposed in [208] has only been tested on CIF resolution sequences with a frame rate of 30 fps, it is necessary to verify if Eq. (7.1) fits also well to 4CIF resolution sequences at both 30 fps and 25 fps. Experiments carried out on each separate video sequence have shown that the coefficient of determination R^2, which is an indicator of how well the observed outcomes are replicated by the model, is always greater than 0.92. Based on these results, it is clear that the exponential shape of Eq. (7.1) accurately describes the MOS-PSNR relationship also for 4CIF sequences.

In this direction, for each of the 72 video sequences, we calculate their PSNR values, PSNR_s (for the whole sequence), and using the MOS values of [41], MOS_s, we apply OLS optimization in order to solve for the parameters a and b of Eq. (7.1). Afterwards, we calculate the PSNR values for each frame of all considered 4CIF sequences, PSNR_{fr}, and next we compute MOS_{fr} by applying Eq. (7.1). In other words, we use Eq. (7.1) for each frame (instead of the whole sequence as originally proposed in [208]) in order to obtain the ground truth of the MOS per frame. For the validation of the quality of the obtained values, we average the per-frame MOS values to obtain a representative value for each sequence, $\overline{\text{MOS}_{fr}}$, and check for correlation with the corresponding measured values, MOS_s.

MOS results obtained using subjective tests are typically compressed at the ends of the 5-point rating scale [41]. In order to impose the same behavior to our estimates, we apply a non-linear mapping on the $\overline{\text{MOS}_{fr}}$ values, before computing any of the PCC, SROCC and RMSE performance metrics (see Section 6.4). Specifically, we use the cubic polynomial function given by [199]

$$\overline{\text{MOS}'_{fr}} = a_1 \overline{\text{MOS}_{fr}}^3 + a_2 \overline{\text{MOS}_{fr}}^2 + a_3 \overline{\text{MOS}_s} + a_4, \qquad\qquad (7.2)$$

which is found to perform well empirically. The weights a_1, a_2, a_3 and the constant

a_4 are calculated by fitting the function to the data $(\overline{\mathrm{MOS}_{fr}}, \mathrm{MOS}_s)$ with the goal of maximizing their correlation.

The relationship between $\overline{\mathrm{MOS}'_{fr}}$ and MOS_s is given by the scatter plot of Fig. 7.1, for all 4CIF sequences. From this figure, we confirm a nearly perfect linear relationship in terms of the PCC [199], which is equal to 0.99, since a PCC of 1 indicates a perfect positive correlation between the measured and estimated data.
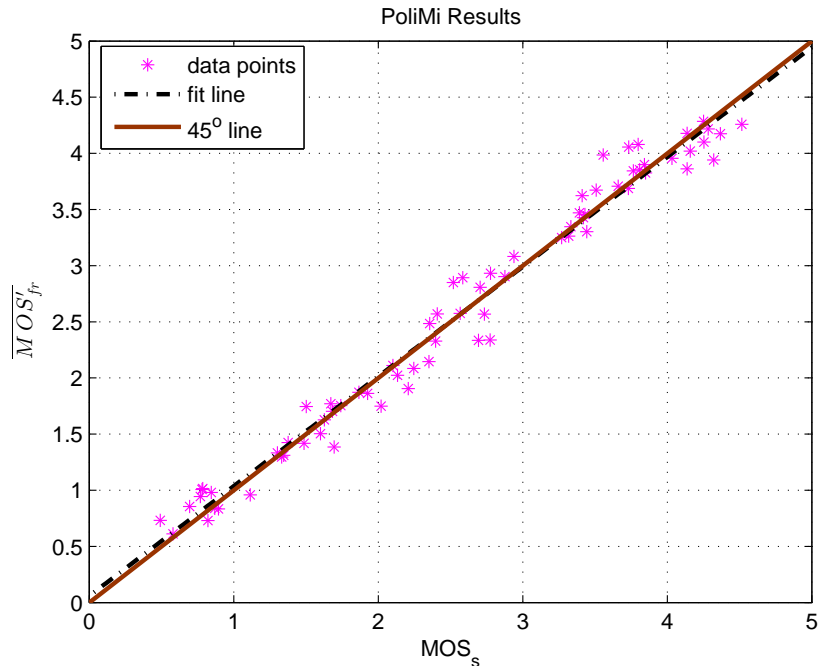

Figure 7.1: Correlation results.

The estimated a and b parameters of Eq. (7.1) as well as the PCC for each separate sequence are depicted on Table 7.1. As we can see from this table, if we apply the parameters a and b to Eq. (7.1), the subjective results can be estimated with a nearly perfect precision.

Table 7.1: Estimated a and b values.

			PoliMi MOS			
	"Crowdrun"	"Duckstakeoff"	"Harbour"	"Ice"	"Parkjoy"	"Soccer"
a	15.62	15.35	12.08	16.55	15.39	15.97
b	9.99	9.06	14.26	13.44	9.79	12.16
PCC	0.98	1.00	0.99	0.98	0.98	1.00

More light about the efficiency of the aforementioned FR metric is shed by the results presented on Table 7.2. A comparison of the employed metric used for taking per-frame MOS quality values with the state-of-the-art PEVQ [72] and VQM [1] FR metrics reveals that although all of these metrics correlate well with MOS, the metric proposed in our study is even more efficient in terms of PCC [199], SROCC [65] and RMSE [199]. Thus,

154

our experimental results verify the suitability of the proposed MOS per frame metric for estimating the actual MOS per sequence.

Table 7.2: Comparison of FR objective metrics.

	Proposed	PEVQ	VQM
PCC	**0.99**	0.96	0.96
SROCC	**0.99**	0.97	0.96
RMSE	**0.18**	0.38	0.54

7.2 MOS Estimation

Except for the ground truth, which is required in supervised learning regression problems and plays the role of the target variable that needs to be estimated, a design matrix that includes the values for the explanatory variables is also assumed so as to be taken as input to a considered regression model. In this piece of our research, we use a large number of bitstream-based features, 45 in total, that are expected to affect perceptual video quality. The specific features account for compression artifacts and packet-loss impairments and are listed from $1 - 46$, except for feature 37, described in Section 6.1. Some of the aforementioned features are related to the occurrence of a packet loss and thus, they are computed at the slice level, while the features that are related to motion vectors are computed at the MB level. For our frame-level regression problem, all feature observations calculated at the MB- or slice-level are averaged to obtain representative values for each frame. On the contrary, for the regression problem at the sequence-level, the frame-level feature observations are averaged further to get a feature value for each separate sequence.

Having collected the feature observations as well as the MOS ground truth, we proceed with applying regression. For the frame-level case, we construct a regression model that generates per-frame MOS estimates, EMOS_{fr}, which are next averaged to provide an overall MOS value for each sequence, $\overline{\text{EMOS}_{fr}}$. For comparison purposes, we also develop a regression model that operates directly on the sequence-level domain and produces per-sequence MOS estimates, EMOS_s. Our frame-level dataset includes 19008 feature observations (one feature value for each frame) and the sequence-level dataset consists of 72 feature observations (one feature value for each sequence). Next, we apply Ridge regression [22, 67, 115] (see Section 6.2.2) on both the frame- and sequence-level domains, where we set $\lambda = 10^{-5}$ in the regression models of both domains.

The column "Raw dataset" of Table 7.3 presents the PCC, SROCC and RMSE statistics, described in Section 6.4, when regression is applied on both the frame- and sequence-level domains, by employing all 45 features extracted from the bitstreams. For the frame-level case, the MOS estimates for each frame obtained from Ridge are averaged to obtain a MOS value for each video sequence, $\overline{\text{EMOS}_{fr}}$ and next, we compare

155

these results with the actual sequence-level MOS values, MOS_s. For the sequence-level case, the actual MOS values, MOS_s, are compared with the MOS values obtained from Ridge, EMOS_s. Examining the "Raw dataset" results of Table 7.3, it is clear that regression on the frame-level dataset guarantees exceptionally good performance statistics that are definitely better than the statistics achieved by performing regression on the sequence-level dataset.

Table 7.3: Performance statistics.

	Raw dataset		Processed dataset		Related works	
	Sequence-level	Frame-level	Sequence-level	Frame-level	G.1070E [17]	$\mathrm{SLR_{IP}} + \mathrm{SLR_B}$ [109]
PCC	0.87	0.96	0.96	**0.97**	0.93	0.96
SROCC	0.85	0.96	0.96	**0.97**	0.91	−
RMSE	0.58	0.34	0.31	**0.28**	0.37	0.34

Feature Selection in Ridge

In an effort to enhance the strength of the sequence-level regression model in making precise estimations, we apply *stepwise regression* [43] so as to elaborate on the extracted features and keep only the most beneficial of them as well as to make use of their most favorable pairwise interactions. The specific method starts with an initial dataset and then compares the explanatory power of incrementally larger or smaller datasets. Algorithm 4 below summarizes the basic idea of this methodology.

Algorithm 4 Stepwise Regression

Initialize: No predictors in the model.
1: **repeat**
2:    **if** F-test p-value ≤ 0.05 **then** // predictors not in the model
3:       Add the predictor with the smallest p-value;
4:    **else if** F-test p-value ≥ 0.10 **then** // predictors in the model
5:       Remove the predictor with the largest p-value;
6:    **else**
7:       **return** ;
8:    **end if**
9: **until return**

This procedure is performed separately on the sequence- and frame-level domains, and thus, some of our initial features are eliminated from each corresponding dataset. At the same time, pairwise feature interactions are added in order to ameliorate the precision of the estimations. After conducting stepwise regression, we end up with a sequence-level dataset of dimension 72×26 and a frame-level dataset of dimension 19008×95. This means that the number of features has been significantly reduced in the sequence-level dataset and considerably increased in the frame-level dataset.

On the processed sequence-level and frame-level datasets, we apply again Ridge regression (Eq. (6.3)) to obtain the new MOS estimates. The column "Processed dataset" of Table 7.3 presents the performance statistics, when regression is performed, by employing only the features and their interactions as indicated after applying Alg. 4 on both the sequence- and frame-level domains. The provided results reinforce our claim that a more stable model is achieved when regression is applied at the frame-level, which offers more precise MOS estimations compared to the model built at the sequence-level. Interestingly, only a minor improvement is observed compared to the already high frame-level performance statistics of the raw dataset. Regarding the sequence-level case, the correlation measures as well as the RMSE are considerably improved in this case, compared to the case of performing regression on the raw dataset.

Continuing, Table 7.3 not only includes the comparison of our proposed regression model that operates on the frame-level domain with the regression procedure performed directly on the sequence-level domain, but also presents comparison with recent related works [17, 109] (column "Related works") that develop NR video quality metrics, based on a similar rationale of using perceptually-driven features. Both [17] and [109] use the database of [41], and estimate MOS directly for each sequence. An overall look at the results of this table makes clear that our approach of exploiting a much larger number of feature observations as well as the concept of making estimations at the frame level lead to very good performance statistics that outperform the results of competing approaches.

Summarizing the key points of this study, we corroborate that the proposed frame-domain regression approach used for estimating MOS in an NR way provides more accurate estimates of the actual MOS of a video sequence than a similar sequence-domain regression approach, and also outperforms recently-proposed competing methods. Moreover, the proposed method offers the additional benefit of providing an indication of the quality of each individual frame, something that sequence-domain approaches cannot do.

Chapter 8

Conclusions and Future Work

8.1 Concluding Remarks

8.2 Directions for Future Work

T his study elaborates mainly on two different research directions. The first one is the resource allocation of the transmission parameters of the nodes constituting a wireless VSN and the second one is the estimation of perceptual video quality of video sequences transmitted over wireless networks in a reduced-reference and a no-reference manner. In this chapter, we summarize the material presented in this thesis and present the derived conclusions. In addition, we discuss suggestions for future research in both the aforementioned research topics.

8.1 Concluding Remarks

In Chapter 2 we have presented the recent advances in resource allocation problems similar to the one analysed in this thesis as well as the achievements in estimating perceptual video quality. Specifically, we have discussed the various network layer designs proposed in the recent literature and we have mentioned works where quality-based criteria as well as schemes extracted from game theory have been applied to address resource allocation problems. An overview of works that examine various indices in order to evaluate the fairness and efficiency of different optimization criteria has also been presented. In addition, a brief introduction to the application of metaheuristics as well as reinforcement learning algorithms in resource allocation tasks has also been discussed. In the following, we have presented a number of works that attempt to capture the effect of a packet loss on video sequences during wireless transmissions over noisy channels. Last, we have extensively presented related works that address perceptual video quality estimation problems in a RR and a NR manner. The novel

aspects and contributions that the present thesis brings over existing publications in each particular topic have been well highlighted in the same chapter.

Chapter 3 has studied the problem of optimal resource allocation among the nodes of a wireless DS-CDMA VSN, under the constraints of a fixed bitrate and a bounded power level for each node. More specifically, we have optimally allocated the source coding rates, channel coding rates, and power levels to all nodes of the VSN, based on the detected amount of motion per node. For the source and channel coding rates, discrete values have been considered, while for the power levels, we have assumed continuous values within a predetermined range. The MAD criterion that minimizes the average network distortion and the MMD criterion that minimizes the maximum distortion among all nodes of the network, both aiming at achieving the highest possible video quality, have been employed in order to solve the problem of resource allocation. The underlying mixed-integer optimization problems have been solved using the established PSO algorithm. The performance of this algorithm has been assessed in comparison with the performance of the AS method, justifying its comparative advantage in tackling such problems. Additionally, we have developed a hybrid optimization method that is based on PSO, using the AS as a local optimizer. The experimental results using all optimization methods highlight the superiority of HPSOAS over both AS and PSO, under various aspects.

Having clustered the nodes into two motion classes based on the amount of the detected motion per node, extensive experimentation has shown that the MAD criterion works favorably for the nodes that image low motion, since they are offered considerably higher PSNR values. On the other hand, the MMD assigns equal PSNR values to both motion classes. Nevertheless, it is not sufficiently clear how fair the MMD can be for the nodes that record low motion, taking into consideration the significant PSNR reduction observed in this case, for this class of nodes. Furthermore, our results confirm that the CCU receives less power with the MAD than with the MMD, implying that MAD requires less power for data transmission. Experiments conducted under the presence of thermal and background noise have verified the conclusions derived for the noiseless case. The main impact of noise is a marginal reduction of the PSNR of both motion classes and optimization criteria, with only a minor exception. Also, the nodes that detected high levels of motion require considerably higher power levels than the nodes that detect low levels of motion to accomplish data transmission.

Except for the MAD and MMD criteria that have been used to solve the problem of resource allocation, we have also assumed that the nodes of the VSN negotiate with the help of the CCU, and the result of the negotiation is the NBS, which aims at a fair distribution of system resources among the nodes. The NBS utilizes a disagreement point, which corresponds to the minimum acceptable video quality for each node. Based on the NBS, we have proposed two optimization criteria, which differ in the way that the bargaining powers are determined for the nodes. The first approach (NNBS) treats each node as equally advantaged, while the second one (CNBS) assumes the same advantage

for each class of nodes. The PSO algorithm is proved the best choice among other conventional optimization methods for solving the mixed-integer problems, resulting from the continuous values for the power levels and the discrete values for the source and channel coding rates.

The performance of the NBS-based criteria has been compared with the performance of the MAD and MMD. The MAD criterion minimizes the average video distortion of the nodes without regard to fairness. The MMD criterion typically results in the same video distortion for all nodes, at the cost of a very high power consumption compared with the other schemes. This is a significant drawback of the MMD that could prohibit its use in practical applications. On the contrary, we confirm that the NNBS and CNBS keep low computational complexity and can be used to any wireless VSN with a centralized topology that uses DS-CDMA. Additionally, a wise selection between NNBS and CNBS according to the needs of each application and the node distribution, produces worthwhile results that are preferable to those of MAD and MMD.

Continuing, the same problem of resource allocation has been solved using the Kalai-Smorodinsky bargaining solution. This scheme, based on its fairness axioms, provides a fair and efficient rule that assigns the transmission parameters to each node. In our study, this solution is derived geometrically, based on the graphical representation of each considered feasible set, implying low running complexity. The performance of the KSBS has been assessed in comparison with the NNBS and two other methods that attempt to maximize an unweighted (MTU) and a weighted (w.MTU) version of the total system utility, respectively.

For the quality evaluation of the methods, we have used a metric that captures both fairness and performance issues. This metric expresses the total utility gain achieved by all nodes using the MTU, that is attributed to every unit of utility lost by an isolated node using also the MTU. Additionally, we have studied the total utility achieved cumulatively from all nodes in combination with the total power consumption, for each scheme. In this context, we infer that no scheme gathers all desired features of being equally fair to all nodes, assuring the highest total utility, and requiring the lowest levels of power, at the same time. Nevertheless, comparisons have led us to the conclusion that the KSBS is the criterion that is closer to our demands. The main strength of this method is that it assures quite low levels of power consumption, while assigning close enough PSNR values to all nodes and having low running complexity at the same time.

Studying the behavior modeling and analysis of all the aforementioned resource allocation schemes we have employed four metrics that examine fairness under a different point of view: i) the PF metric, which quantifies the relationship between fairness and performance, ii) the Jain's index, which measures how "equal" is an allocation for all users, using the same scheme, iii) the utility gained cumulatively by all nodes of the same scheme and iv) the total consumed power by all nodes, also under the same scheme. All the solutions provided by the schemes are Pareto-optimal solutions and

thus, the choice about the most fair and efficient scheme is not evident. There is no scheme that holds all desired characteristics of achieving the highest total utility, while being equally fair to all nodes (equal utility allocations for the nodes), and spending the lowest total power, at the same time. Therefore, the selection of the appropriate scheme depends on the particular application in combination with the users' desires. In other words, the application's special characteristics and requirements shall dictate the methodology to use, inhibiting the possibility of a panacea that would simultaneously favor all nodes.

In Chapter 4 we have also presented the problem of cross-layer resource allocation among the nodes of a wireless DS-CDMA VSN. Additionally, the optimal GOP structure for the encoding of each video sequence captured by the nodes has been the other main objective. In this case, all of the nodes' transmission parameters could take discrete values and for their determination the MAD, NNBS and MTU criteria have been utilized. Allowing the nodes to select among various GOP lengths for the encoding of the video they capture by considering the motion level included in those scenes, video quality enhancement has beenis observed compared to fixed GOP length considerations. Furthermore, the adopted RL approach used for tackling the discrete optimization problem is proved extremely efficient compared to the brute-force ES algorithm. Although both ES and SARSA algorithms are able to reach to the optimal solution, SARSA requires far less steps, making the proposed methodology applicable in online form.

In Chapter 5 we have focused on the problem of accurately estimating the CMSE within each GOP of a video sequence, incurred due to possible slice losses during wireless video transmissions over noisy channels. Initially, we have calculated the actual CMSE values, assuming that each individual slice is lost, and have used this index as the "ground truth" of perceptual distortion. Continuing, we have extracted a number of features from a large collection of H.264/AVC video sequences and have used all feature observations as input to the LASSO model. The specific regression technique is able to provide accurate CMSE estimations, by selecting only a subset of the features; the ones that are the most influential towards CMSE estimations. In more detail, we have considered two different LASSO architectures; G.LASSO and L.LASSO. In G.LASSO a single regression model is built for all slice types, while in L.LASSO a separate regression model is considered for each different slice type. Based on the measured and estimated CMSE values, we have grouped the slices into four priority classes for each case, by using a QBP scheme. Last, in order to evaluate the performance of our proposed scheme, we have assumed a video transmission scenario over an AWGN channel, where UEP has been applied to each priority class, within each GOP of a video sequence. The provided experimental results verify the suitability of the employed video features as well as the efficiency of the proposed regression models in producing precise CMSE estimations. In addition, as it has been expected, L.LASSO is proved to be a better choice than G.LASSO and mainly than GLM for enhancing further the accuracy of CMSE estimations. The slice misclassification percentages are low, the

performance statistics high, and the average PSNR values of the video transmission scenario achieved by the proposed models and GLM are virtually identical to the PSNR values resulting from the measured CMSE case, especially for a channel SNR of 3 dB or higher.

A NR bitstream-based video quality model, which accounts for the impact of compression artifacts as well as the impairments due to possible packet losses has been presented in Chapter 6. A large set of features has been extracted from the impaired bitstreams and the LASSO regression model has been used to perform feature selection in order to reject the features that harm the quality estimates, and produce video quality estimations that correlate well with subjective assessment. For comparison purposes, we have utilized the OLS regression method and except for MOS, we have built models able to estimate SSIM and VQM. The experimental results signify that LASSO achieves high performance statistics using only a few features, as opposed to OLS. Despite the fact that OLS presents a competitive performance with LASSO in terms of the examined measures of performance, it employs a much larger number of features increasing problem's complexity. Moreover, the validity of the features used by LASSO lead to a very good correlation of our model with MOS, SSIM and VQM. Hence, LASSO is a good choice for building a simple and low-complexity model that offers high correlation with subjective ratings and FR metrics, when it is supplied with appropriate, quality-relevant features.

In the same chapter, we have also studied the problem of perceptual video quality estimation by developing novel RR and NR models. Specifically, the vast majority of the features employed for building the models can be computed without any access to the original video and hence, they are applicable to design a NR model of quality estimation. These features have a very large overlap with the ones employed earlier to the proposed NR bitstream-based model. The rest of the features can be pre-computed and sent to the client's end for providing RR information of the original video. Based on these features, we have proposed the RR and NR models of quality estimation, by employing the LASSO regression method. LASSO has been investigated for its capability to make MOS estimations as well as feature selection at the same time. For comparison purposes, we have applied sequential forward feature selection, using Ridge as the regression method, so as to get a baseline performance. The simulation results reveal that LASSO is able to achieve exceptionally high and marginally better performance statistics as compared to Ridge using FFS, utilizing on average only two and three features, for the RR and NR cases, respectively. Interestingly, all the features required by the NR LASSO model for estimating MOS have been extracted from the lossy bitstream, without the need for its decoding. This means that significantly less computational complexity is involved in the feature extraction process, rendering the model practical in real-time applications. In addition, the proposed LASSO models outperform a number of existing FR, RR, and NR techniques used for video quality estimation, in terms of both performance statistics and required features.

Last, in Chapter 7 we have also proposed a novel NR method for the quality estimation of videos that are impaired by both compression artifacts and packet losses. Initially, we have introduced a FR metric, which is able to provide the quality ground truth for each frame and next, we have developed a regression method for the perceptual estimation of the quality of each frame, using a number of features extracted from the received video bitsteam. The MOS of the video sequence has been estimated as the average of the estimated MOS per frame values. The presented experimental results show that the proposed frame-domain approach provides more accurate estimates of the actual MOS of a video sequence than a sequence-domain approach, and also outperforms recently-proposed competing methods. Moreover, the proposed method offers the additional benefit of providing an indication of the quality of each individual frame, something that sequence-domain approaches cannot do.

8.2   Directions for Future Work

In the following, we present some promising directions for future research that elaborate on a number of open issues related to the problems that we have tackled in this thesis so far. To begin with the problem of resource allocation in wireless VSNs, the optimization problem could be formulated under a different perspective so as to align more with the human visual system, by adopting appropriate optimization objectives that take into account the video quality as it is perceived by the users.

In more detail, we aim to focus on techniques that provide perceptual quality guarantees for video transmissions over wireless channels. MSE is a widely used full reference objective measure in modern block-based video compression algorithms such as H.264/AVC. It is employed by the rate-distortion optimized mode selection process as a quality measure for choosing the best compression option that gives an optimal tradeoff between picture quality and data rate. While the general approach is to use MSE to choose the best coding option, MSE is a mathematical error measure which does not consider the human visual system and is therefore not an accurate measure of perceptual quality. It may be possible to improve the subjective quality performance of a rate-constrained video codec by replacing MSE with a distortion metric that correlates more closely with subjective quality. Thus, we will attempt to formulate an optimization problem that will eventually result in the QoE-driven network performance optimization, assessing the perceptual quality of the transmitted video. Also, we plan to consider the use of scalable video coding and, specifically, the scalable extension of H.264/AVC.

In our future plans, we aim to study the design, development and validation of a similar cross-layer resource management system for Mobile Ad-hoc NETworks (MANET). This means that we will consider a case where the nodes of the network are not fixed, but in contrast they are able to move independently in any direction, and therefore can change their links to other devices frequently. For instance, video transmission over a MANET could be applied in a case of a natural disaster. Let us assume that two

Unmanned Aerial Vehicles (UAV) cover an area from the air, while a "flock" of robots equipped with video cameras explore the same area from the ground. The robots transmit the recorded scenes using multi-hop routing, succeeding simultaneously two goals: to cover a larger area and to deliver the best possible video quality. The UAV communicate with each other and with the robots achieving multiple tasks, e.g. robot localization or guidance.

Moreover, the work presented in Chapter 5 could be extended by replacing the CMSE index with another index that correlates even better with the human perception. In other words, we target to assign priorities to the different video slices based on the estimated values of a metric, other than the CMSE, following also the rationale of feature extraction from compression-impaired and packet-loss affected video sequences so as to obtain the feature observations, which will next be given as input to a regression model.

Regarding the studies of Chapters 6 and 7 they could be extended by testing the proposed models on higher resolution test stimuli with more variety in compression and network related features. In view of the recently approved standard of video compression, i.e., High Efficiency Video Coding (HEVC), the proposed techniques could also be applied on the HEVC encoded test stimuli. As it is very important to model the perceptual video quality for different types of scalability, similar models could be developed in the scalable extension of the H.264/AVC, for the various types of scalability (quality, temporal, spatial) as well as their combinations. For example, is it perceptually better to increase the frame rate of a video or its spatial resolution and/or PSNR? With the successful development of perceptual quality estimation methods, the receiver of the video signal is able to automatically estimate the video quality that would be perceived by a human, without having an actual human look at the video and without access to the original uncompressed video. Thus, the QoE can be continuously monitored, which is the case in many modern applications that include video transmission. For example, the involvement of multiple parties between content providers and the end-users gives rise to establish service level agreements under which an agreed level of quality has to be guaranteed.

BIBLIOGRAPHY

[1] http://www.its.bldrdoc.gov/resources/video-quality-research/software.aspx.

[2] Requirements for internet hosts - communication layers, 1989. R. Braden, RFC Editor, United States.

[3] Requirements for IP version 4 routers, 1995. F. Baker, RFC Editor, United States.

[4] Subjective video quality assessment methods for multimedia applications, Sept. 1999. ITU-T, Recommendation ITU-R P910.

[5] Full reference television phase II subjective test plans, 2002. Objective quality model evaluation criteria.

[6] H.264/AVC reference software JM 14.2. http://iphome.hhi.de/suehring/tml/download/old_jm/, 2011. [Online; Accessed 25-April-2015].

[7] Recommendation G.1070: Opinion model for videotelephony applications. http://www.itu.int/rec/T-REC-G.1070, 2012. [Online; Accessed 04-April-2015].

[8] Cisco visual networking index: Global mobile data traffic forecast update, 2013-2018. *Cisco white paper*, 2014.

[9] http://trace.eas.asu.edu/yuv/, 2015. [Online; Accessed 16-April-2015].

[10] J.342 : Objective multimedia video quality measurement of HDTV for digital cable television in the presence of a reduced reference signal. http://www.itu.int/rec/T-REC-J.342-201104-I/en, 2015.

[11] MathWorks: Constrained non-linear optimization algorithms: R2012a documentation. http://www.mathworks.com/help/optim/ug/constrained-nonlinear-optimization-algorithms.html, 2015. [Online; Accessed 13-April-2015].

[12] R. Agrawal, V. Subramanian, and R. Berry. Joint scheduling and resource allocation in CDMA systems. In *Proc. of the 2nd Workshop on Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks (WiOpt '04)*, pages 24–26, 2004.

[13] I. F. Akyildiz, T. Melodia, and K. R. Chowdhury. A survey on wireless multimedia sensor networks. *Computer Networks*, 51(4):921–960, 2007.

[14] T. Alpcan, T. Başar, R. Srikant, and E. Altman. CDMA uplink power control as a noncooperative game. *Wireless Networks*, 8(6):659–670, 2002.

[15] T. Alpert and J.-. P. Evain. Subjective quality evaluation - The SSCQE and DSCQE methodologies. 1997. EBU Technical Review.

[16] S. Argyropoulos, A. Raake, M. N. Garcia, and P. List. No-reference video quality assessment for SD and HD H.264/AVC sequences based on continuous estimates of packet loss visibility. In *3rd International Workshop on Quality of Multimedia Experience (QoMEX)*, pages 31–36, Sept. 2011.

[17] J. Ascenso, H. Cruz, and P. Dias. Packet-header based no-reference quality metrics for H.264/AVC video transmission. In *International Conference on Telecommunications and Multimedia (TEMU)*, pages 174–151, Jul. 2012.

[18] H. Y. Benson. Using interior-point methods within an outer approximation framework for mixed integer nonlinear programming. In *Mixed Integer Nonlinear Programming*, volume 154 of *The IMA Volumes in Mathematics and its Applications*, pages 225–243. Springer New York, 2012.

[19] E. S. Bentley, L. P. Kondi, J. D. Matyjas, M. J. Medley, and B. W. Suter. Spread spectrum visual sensor network resource management using an end-to-end cross-layer design. *IEEE Transactions on Multimedia*, 13(1):125–131, Feb. 2011.

[20] E. S. Bentley, J. D. Matyjas, M. J. Medley, and L. P. Kondi. Wireless visual sensor network resource allocation using cross-layer optimization. In *Visual Communications and Image Processing*, volume 7257. SPIE, 2009.

[21] K. Binmore. *Playing for Real: A Text on Game Theory*. Oxford University Press, 2007.

[22] C. M. Bishop. *Pattern Recognition and Machine Learning (Information Science and Statistics)*. Springer-Verlag New York, Inc., Secaucus, NJ, USA, 2006.

[23] I. Bouazizi. Estimation of packet loss effects on video quality. In *1st International Symposium on Control, Communications and Signal Processing*, pages 91–94, 2004.

[24] O. Y. Bursalioglu and G. Caire. Is unequal error protection useful? In *IEEE International Symposium on Information Theory Proceedings (ISIT)*, pages 1402–1406, Jul. 2011.

[25] R. H. Byrd, J. C. Gilbert, and J. Nocedal. A trust region method based on interior point techniques for nonlinear programming. *Mathematical Programming*, 89(1):149–185, 2000.

[26] R. H. Byrd, M. E. Hribar, and J. Nocedal. An interior point algorithm for large scale nonlinear programming. *SIAM Journal on Optimization*, 9:877–900, 1997.

[27] Y. S. Chan and J. W. Modestino. A joint source coding-power control approach for video transmission over CDMA networks. *IEEE Journal on Selected Areas in Communications*, 21(10):1516–1525, 2006.

[28] Y.-L. Chang, T.-L. Lin, and P. C. Cosman. Network-based H.264/AVC whole-frame loss visibility model and frame dropping methods. *IEEE Transactions on Image Processing*, 21(8):3353–3363, Aug. 2012.

[29] J. Chen and A. L. Swindlehurst. Downlink resource allocation for multi-user MIMO-OFDMA systems: The Kalai-Smorodinsky bargaining approach. In *3rd IEEE International Workshop on Computational Advances in Multi-Sensor Adaptive Processing (CAMSAP)*, pages 380–383, 2009.

[30] S. Chikkerur, V. Sundaram, M. Reisslein, and L. J. Karam. Objective video quality assessment methods: A classification, review, and performance comparison. *IEEE Transactions on Broadcasting*, 57(2):165–182, Jun. 2011.

[31] M. Chin, T. Brandao, and M. P. Queluz. Bitstream-based quality metric for packetized transmission of H.264 encoded video. In *19th International Conference on Systems, Signals and Image Processing (IWSSIP), 2012*, pages 312–315, Apr. 2012.

[32] J. Choe, J. Lee, and C. Lee. No-reference video quality measurement using neural networks. In *16th International Conference on Digital Signal Processing*, pages 1–4, Jul. 2009.

[33] M. Clerc and J. Kennedy. The particle swarm-explosion, stability, and convergence in a multidimensional complex space. *IEEE Transactions on Evolutionary Computation*, 6(1):58–73, 2002.

[34] T. F. Coleman and Y. Li. An interior trust region approach for nonlinear minimization subject to bounds. Technical report, Cornell University, Ithaca, NY, USA, 1993.

[35] T. F. Coleman and Y. Li. On the convergence of reflective Newton methods for large-scale nonlinear minimization subject to bounds. volume 7 of *Mathematical Programming*, pages 189–224, 1994.

[36] J. P. Conley and S. Wilkie. The bargaining problem without convexity: extending the Egalitarian and Kalai–Smorodinsky solutions. *Economic Letters*, 36(4):365–369, Aug. 1991.

[37] D. Culibrk, M. Mirkovic, V. Zlokolica, M. Pokric, V. Crnojevic, and D. Kukolj. Salient motion features for video quality assessment. *IEEE Transactions on Image Processing*, 20(4):948–958, Apr. 2011.

[38] E. G. Datsika, A. V. Katsenou, L. P. Kondi, E. Papapetrou, and K. E. Parsopoulos. Priority-based cross-layer optimization for multihop DS-CDMA visual sensor networks. In *IEEE International Conference on Image Processing (ICIP)*, pages 1101–1104, Sept. 2012.

[39] J. D. Day and H. Zimmermann. The OSI reference model. *In Proceedings of the IEEE*, 71(12):1334–1340, Dec. 1983.

[40] P. Dayan. The convergence of TD(λ) for general lambda. *Machine Learning*, 8:341–362, 1992.

[41] F. De Simone, M. Naccari, M. Tagliasacchi, F. Dufaux, S. Tubaro, and T. Ebrahimi. Subjective quality assessment of H.264/AVC video streaming with packet losses. *EURASIP Journal on Image and Video Processing*, 2011 Article ID 190431, 2011.

[42] B. Dieber, C. Micheloni, and B. Rinner. Resource-aware coverage and task assignment in visual sensor networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 21(10):1424–1437, Oct. 2011.

[43] N. Draper and H. Smith. *Applied regression analysis*. A Wiley-Interscience publication. Wiley, New York, NY, 3. ed. edition, 1998.

[44] R. C. Eberhart and J. Kennedy. A new optimizer using particle swarm theory. In *Proceedings of the Sixth Symposium on Micro Machine and Human Science*, pages 39–43, Piscataway, NJ, 1995. IEEE Service Center.

[45] R. C. Eberhart and Y. Shi. Comparing inertia weights and constriction factors in particle swarm optimization. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 84–88, Piscataway, NJ, 2000.

[46] Y. Eisenberg, C. E. Luna, T. N. Pappas, R. Berry, and A. K. Katsaggelos. Joint source coding and transmission power management for energy efficient wireless video communications. *IEEE Transactions on Circuits and Systems for Video Technology*, 12(6):411–424, Jun. 2002.

[47] U. Endriss, N. Maudet, F. Sadri, and F. Toni. On optimal outcomes of negotiations over resources. In *Proceedings of the 2nd International Joint Conference*

on Autonomous Agents and Multiagent Systems, pages 177–184, New York, NY, USA, 2003. ACM.

[48] A. P. Engelbrecht. *Fundamentals of Computational Swarm Intelligence.* Wiley, 2006.

[49] H. Eren. *Wireless sensors and instruments: networks, design, and applications.* CRC Press, 2005.

[50] J. C. Fang and R. R. Rao. An integrated and distributed scheduling and power control algorithm for maximizing network utility for wireless multihop networks. In *IEEE Military Communications Conference, 2003. MILCOM '03.*, volume 2, pages 1011–1017, Oct. 2003.

[51] A. Farmahini-Farahani, S. Vakili, S. M. Fakhraie, S. Safari, and C. Lucas. Parallel scalable hardware implementation of asynchronous discrete particle swarm optimization. *Engineering Applications of Artificial Intelligence*, 23(2):177–187, 2010.

[52] A. R. Fattahi and F. Paganini. New economic perspectives for resource allocation in wireless networks. In *Proceedings of the 2005 American Control Conference*, pages 3960–3965, vol. 6, Jun. 2005.

[53] F. J. Ferri, P. Pudil, M. Hatef, and J. Kittler. Comparative study of techniques for large-scale feature selection. *Pattern Recognition in Practice IV, Elsevier Science B.V.*, pages 403–413, 1994.

[54] C. A. Floudas. *Nonlinear and Mixed-Integer Optimization.* Oxford University Press, New York, 1995.

[55] H. Ganapathy, S. Banerjee, N. Dimitrov, and C. Caramanis. Optimal feedback allocation algorithms for multi-user uplink. In *47th Annual Allerton Conference on Communication, Control, and Computing (Allerton)*, pages 947–954, 2009.

[56] V. K. Garg. *IS-95 CDMA and CDMA2000.* Prentice Hall PTR, 2000.

[57] P. Gastaldo, R. Zunino, and J. Redi. Supporting visual quality assessment with machine learning. *EURASIP Journal on Image and Video Processing*, 2013(1):54, 2013.

[58] J. D. Gibbons and S. Chakraborti. *Nonparametric Statistical Inference (Statistics: a Series of Textbooks and Monogrphs).* CRC, 4 edition, May 2003.

[59] K. S. Gilhousen, I. M. Jacobs, R. Padovani, A. J. Viterbi, L. A. Weaver, and C. E. Wheatley. On the capacity of a cellular CDMA system. *IEEE Transactions on Vehicular Technology*, 40(20):303–312, May 1991.

[60] P. E. Gill, W. Murray, and M. H. Wright. *Practical optimization*. Academic Press, 1981.

[61] J. Hagenauer. Rate-compatible punctured convolutional codes (RCPC codes) and their applications. *IEEE Transactions on Communications*, 36(4):389–400, 1988.

[62] Z. Han, Z. Ji, and K. J. R. Liu. Power minimization for multi-cell OFDM networks using distributed non-cooperative game approach. In *Global Telecommunications Conference (GLOBECOM)*, pages 3742–3747. IEEE, 2004.

[63] Z. Han and K. J. R. Liu. Noncooperative power-control game and throughput game over wireless networks. *IEEE Transactions on Communications*, 53(10):1625–1629, 2005.

[64] J. Heinonen. Lectures on Lipschitz analysis. pages 1–77, 2005.

[65] D. E. Hinkle, W. Wiersma, and S. G. Jurs. *Applied statistics for the behavioral sciences*. Boston, Mass: Houghton Mifflin, 2003.

[66] A. E. Hoerl. Application of Ridge analysis to regression problems. *Chemical Engineering Progress*, 58(3):54–59, 1962.

[67] A. E. Hoerl and R. W. Kennard. Ridge regression: Biased estimation for nonorthogonal problems. *Technometrics*, 12:55–67, 1970.

[68] M. Hollander and D. A. Wolfe. *Nonparametric Statistical Methods*. Wiley-Interscience, 2 edition, 1999.

[69] Y. M. Hsiao, J. F. Lee, J. S. Chen, and Y. S. Chu. H.264 video transmissions over wireless networks: challenges and solutions. *Computer Communications*, 34(14):1661–1672, 2011.

[70] W. J. Huang, Y. W. P. Hong, and C. C. J. Kuo. Lifetime maximization for amplify-and-forward cooperative networks. *IEEE Transactions on Wireless Communications*, 7(5-2):1800–1805, 2008.

[71] A. Ibing and H. Boche. Fairness vs. efficiency: comparison of game theoretic criteria for OFDMA scheduling. In *Conference Record of the Forty-First Asilomar Conference on Signals, Systems and Computers*, pages 275–279, Nov. 2007.

[72] ITU-T Rec. J.247. Objective perceptual multimedia video quality measurement in the presence of a full reference. 2008.

[73] ITU-T Recommendation P.10. Vocabulary for performance and quality of service, Amendment 2: New definitions for inclusion in Recommendation ITU-T P.10/G.100, 2008.

[74] R. K. Jain, D.-M. W. Chiu, and W. R. Hawe. A quantitative measure of fairness and discrimination for resource allocation in shared computer systems. Technical Report TR-301, Digital Equipment Corporation, Sept. 1984.

[75] H. Jiang and W. Zhuang. Resource allocation with service differentiation for wireless video transmission. *IEEE Transactions on Wireless Communications*, 5(6):1456–1468, Jun. 2006.

[76] M. Jiang, Y. P. Luo, and S. Y. Yang. Stochastic convergence analysis and parameter selection of the standard particle swarm optimization algorithm. *Information Sciences*, 102(1):8–16, 2007.

[77] E. Kalai and M. Smorodinsky. Other solutions to Nash's bargaining problem. *Econometrica*, 43(3):513–518, 1975.

[78] S. Kanumuri, P. C. Cosman, A. R. Reibman, and V. A. Vaishampayan. Modeling packet-loss visibility in MPEG-2 video. *IEEE Transactions on Multimedia*, 8(2):341–355, Apr. 2006.

[79] S. Kanumuri, S. G. Subramanian, P. C. Cosman, and A. R. Reibman. Predicting H.264 packet loss visibility using a generalized linear model. In *IEEE International Conference on Image Processing (ICIP)*, pages 2245–2248, Oct. 2006.

[80] A. K. Katsaggelos, Y. Eisenberg, F. Zhai, R. Berry, and T. N. Pappas. Advances in efficient resource allocation for packet–based real–time video transmission. *In Proceedings of the IEEE*, 93:135–147, 2005.

[81] A. V. Katsenou, E. G. Datsika, L. P. Kondi, E. Papapetrou, and K. E. Parsopoulos. Power-aware QoS enhancement in multihop DS-CDMA visual sensor networks. In *18th International Conference on Digital Signal Processing (DSP)*, pages 1–6, Jul. 2013.

[82] A. V. Katsenou, L. P. Kondi, and E. Papapetrou. Distortion-aware joint scheduling and resource allocation for wireless video transmission. In *Proceedings of the 21st European Signal Processing Conference (EUSIPCO)*, pages 1–5, Sept. 2013.

[83] A. V. Katsenou, L. P. Kondi, and K. E. Parsopoulos. Resource management for wireless visual sensor networks based on individual video characteristics. In *IEEE International Conference on Image Processing (ICIP)*, pages 149–152, Sept. 2011.

[84] A. V. Katsenou, L. P. Kondi, and K. E. Parsopoulos. On the use of clustering for resource allocation in wireless visual sensor networks. In *Proceedings of SPIE*, volume 8305, pages 83050S–83050S–13, 2012.

[85] A. V. Katsenou, L. P. Kondi, and K. E. Parsopoulos. Motion-related resource allocation in dynamic wireless visual sensor network environments. *IEEE Transactions on Image Processing*, 23(1):56–68, Jan. 2014.

[86] A. V. Katsenou, L. P. Kondi, K. E. Parsopoulos, and E. S. Bentley. Quality-driven power control and resource allocation in wireless multi-rate visual sensor networks. In *IEEE International Conference on Image Processing (ICIP)*, pages 1117–1120, Sept. 2012.

[87] Y. Kawayokeita and Y. Horita. NR objective continuous video quality assessment model based on frame quality measure. In *15th IEEE International Conference on Image Processing (ICIP)*, pages 385–388, Oct. 2008.

[88] C. Keimel, J. Habigt, and K. Diepold. Challenges in crowd-based video quality assessment. In *4th International Workshop on Quality of Multimedia Experience (QoMEX)*, pages 13–18, Jul. 2012.

[89] C. Keimel, J. Habigt, M. Klimpke, and K. Diepold. Design of no-reference video quality metrics with multiway partial least squares regression. In *3rd International Workshop on Quality of Multimedia Experience (QoMEX)*, pages 49–54, Sept. 2011.

[90] C. Keimel, M. Klimpke, J. Habigt, and K. Diepold. No-reference video quality metric for HDTV based on H.264/AVC bitstream features. In *IEEE International Conference on Image Processing (ICIP)*, pages 3325–3328, Sept. 2011.

[91] J. Kennedy. Small worlds and mega-minds: Effects of neighborhood topology on particle swarm performance. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 1931–1938, Washington, D.C., USA, 1999. IEEE Press.

[92] J. Kennedy and R. C. Eberhart. Particle swarm optimization. In *Proceedings of the IEEE International Conference on Neural Networks*, volume IV, pages 1942–1948, Piscataway, NJ, 1995. IEEE Service Center.

[93] J. Kennedy and R. C. Eberhart. A discrete binary version of the particle swarm algorithm. In *Proceedings of the Conference on Systems, Man and Cybernetics*, pages 4104–4109, 1997.

[94] M. A. Khan, A. C. Toker, C. Troung, F. Sivrikaya, and S. Albayrak. Cooperative game theoretic approach to integrated bandwidth sharing and allocation. In *Game Theory for Networks, 2009. International Conference on GameNets '09.*, pages 1–9, May 2009.

[95] M. A. Khan, C. Truong, T. Geithner, F. Sivrikaya, and S. Albayrak. Network level cooperation for resource allocation in future wireless networks. In *Proceedings of Wireless Days, WD '08, 1st IFIP*, pages 1–5, Nov. 2008.

[96] S. Khan, Y. Peng, E. Steinbach, M. Sgroi, and W. Kellerer. Application-driven cross-layer optimization for video streaming over wireless networks. *IEEE Communications Magazine*, 44(1):122–130, Jan. 2006.

[97] L. P. Kondi and E. S. Bentley. Game-theory-based cross-layer optimization for wireless DS-CDMA visual sensor networks. In *IEEE International Conference on Image Processing (ICIP)*, pages 4485–4488, Sept. 2010.

[98] L. P. Kondi, F. Ishtiaq, and A. K. Katsaggelos. Joint source-channel coding for motion-compensated DCT-based SNR scalable video. *IEEE Transactions on Image Processing*, 11:1043–1052, 2002.

[99] W. J. Krzanowski, editor. *Principles of Multivariate Analysis: A User's Perspective*. Oxford University Press, Inc., New York, NY, USA, 1988.

[100] A. Ksentini, M. Naimi, and A. Gueroui. Toward an improvement of H.264 video transmission over IEEE 802.11e through a cross-layer architecture. *IEEE Communications Magazine*, 44(1):107–114, Jan. 2006.

[101] R. V. Kulkarni and G. K. Venayagamoorthy. Particle swarm optimization in wireless-sensor networks: A brief survey. *IEEE Transactions on Systems, Man, and Cybernetics, Part C: Applications and Reviews*, 41(2):262–267, Mar. 2011.

[102] H. Kwon, T. H. Kim, S. Choi, and B. G. Lee. A cross-layer strategy for energy efficient reliable delivery in wireless sensor networks. *IEEE Transactions on Wireless Communications*, 5(12):3689–3699, 2006.

[103] E. C. Larson and D. M. Chandler. Most apparent distortion: full-reference image quality assessment and the role of strategy. *Journal of Electronic Imaging*, 19(1), 2010.

[104] E. C. Laskari, K. E. Parsopoulos, and M. N. Vrahatis. Particle swarm optimization for integer programming. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 1582–1587, Hawaii (HI), USA, 2002. IEEE Press.

[105] J. Lee, I. Shin, and H. Park. Adaptive intra-frame assignment and bit-rate estimation for variable GOP length in H.264. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(10):1271–1279, Oct. 2006.

[106] S.-O. Lee, K.-S. Jung, and D.-G. Sim. Real-time objective quality assessment based on coding parameters extracted from H.264/AVC bitstream. *IEEE Transactions on Consumer Electronics*, 56(2):1071–1078, May 2010.

[107] T.-L. Lin, S. Kanumuri, Y. Zhi, D. Poole, P. C. Cosman, and A. R. Reibman. A versatile model for packet loss visibility and its application to packet prioritization. *IEEE Transactions on Image Processing*, 19(3):722–735, Mar. 2010.

[108] W. S. Lin, H. V. Zhao, and K. J. R. Liu. Fairness dynamics in multimedia colluders' social networks. In *IEEE International Conference on Image Processing (ICIP)*, volume 1-5, 2008.

[109] T. Liu, N. Narvekar, B. Wang, R. Ding, D. Zou, G. Cash, S. Bhagavathy, and J. Bloom. Real-time video quality monitoring. *EURASIP Journal on Advances in Signal Processing*, 2011(1), 2011.

[110] T. Liu, Y. Wang, J. M. Boyce, H. Yang, and Z. Wu. A novel video quality metric for low bit-rate video considering both coding and packet-loss artifacts. *IEEE Journal of Selected Topics in Signal Processing*, 3(2):280–293, Apr. 2009.

[111] T. Liu, H. Yang, A. Stein, and Y. Wang. Perceptual quality measurement of video frames affected by both packet losses and coding artifacts. In *1st International Workshop on Quality of Multimedia Experience, (QoMEX)*, pages 210–215, Jul. 2009.

[112] T. J. Liu, W. Lin, and C. C. J. Kuo. A fusion approach to video quality assessment based on temporal decomposition. In *Signal Information Processing Association Annual Summit and Conference (APSIPA ASC)*, pages 1–5, Dec. 2012.

[113] D. G. Luenberger and Y. Ye. *Linear and Nonlinear Programming.* Springer, New York, 2008.

[114] R. Madan, S. Cui, S. Lal, and A. Goldsmith. Cross-layer design for lifetime maximization in interference-limited wireless sensor networks. *IEEE Transactions on Wireless Communications*, 5(11):3142–3152, 2006.

[115] D. W. Marquardt and R. D. Snee. Ridge Regression in Practice. *The American Statistician*, 29(1):3–20, Feb. 1975.

[116] M. Masry, S. S. Hemami, and Y. Sermadevi. A scalable wavelet-based video distortion metric and applications. *IEEE Transactions on Circuits and Systems for Video Technology*, 16(2):260–273, Feb. 2006.

[117] N. H. Mastronarde and M. Van der Schaar. A bargaining theoretic approach to quality-fair system resource allocation for multiple decoding tasks. *IEEE Transactions on Circuits and Systems for Video Technology*, 18(4):453–466, 2008.

[118] MathWorks. Global Optimization Toolbox: Users' Guide (R2011b). Nov. 2011.

[119] M. Mecking and T. Stockhammer. Minimizing distortion via multiuser resource allocation. In *Data Compression Conference (DCC)*, page 464, 2002.

[120] I. Menache and N. Shimkin. Noncooperative power control and transmission scheduling in wireless collision channels. In *Proceedings of the 2008 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems*, SIGMETRICS '08, pages 349–358, New York, NY, USA, 2008. ACM.

[121] G. Minati and E. Pessa. *Collective Beings*. Springer, New York, 2010.

[122] D. C. Mocanu, A. Liotta, A. Ricci, M. T. Vega, and G. Exarchakos. When does lower bitrate give higher quality in modern video services? In *IEEE Network Operations and Management Symposium (NOMS)*, pages 1–5, May 2014.

[123] K. G. Murty. *Linear Complementarity, Linear and Nonlinear Programming*. Sigma Series in Applied Mathematics 3, Berlin: Heldermann Verlag, 1988.

[124] M. Narwaria and W. Lin. SVD-based quality metric for image and video using machine learning. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 42(2):347–364, Apr. 2012.

[125] M. Narwaria, W. Lin, I. V. McLoughlin, S. Emmanuel, and L.-T. Chia. Fourier transform-based scalable image quality measure. *IEEE Transactions on Image Processing*, 21(8):3364–3377, Aug. 2012.

[126] F. Neri, C. Cotta, and P. Moscato. *Handbook of Memetic Algorithms*. Springer-Verlag, Berlin Heidelberg, 2012.

[127] M. R. Osborne, B. Presnell, and B. A. Turlach. A new approach to variable selection in least squares problems. *IMA journal of numerical analysis*, 20(3):389, 2000.

[128] T. Ozcelebi, M. O. Sunay, A. M. Tekalp, and M. R. Civanlar. Cross-layer optimized rate adaptation and scheduling for multiple-user wireless video streaming. *IEEE Journal on Selected Areas in Communications*, 25(4):760–769, 2007.

[129] S. Paluri, K. K. R. Kambhatla, B. A. Bailey, P. C. Cosman, J. D. Matyjas, and S. Kumar. A low complexity model for predicting slice loss distortion for prioritizing H.264/AVC video. *Multimedia Tools and Applications*, pages 1–25, 2014.

[130] S. Paluri, K. K. R. Kambhatla, S. Kumar, B. Bailey, P. C. Cosman, and J. Matyjas. Predicting slice loss distortion in H.264/AVC video for low complexity data prioritization. In *IEEE International Conference on Image Processing (ICIP)*, pages 689–692, Sept. 2012.

[131] K. Pandremmenou, L. P. Kondi, and K. E. Parsopoulos. Optimal power allocation and joint source-channel coding for wireless DS-CDMA visual sensor networks. In *Proceedings of SPIE*, volume 7882, pages 788206–788206–10, 2011.

[132] K. Pandremmenou, L. P. Kondi, and K. E. Parsopoulos. Optimal power allocation and joint source-channel coding for wireless DS-CDMA visual sensor networks using the Nash bargaining solution. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, pages 2340–2343, May 2011.

[133] K. Pandremmenou, L. P. Kondi, and K. E. Parsopoulos. Kalai-Smorodinsky bargaining solution for optimal resource allocation over wireless DS-CDMA visual sensor networks. In *Proceedings of SPIE*, volume 8305, pages 83050T–83050T–11, 2012.

[134] K. Pandremmenou, L. P. Kondi, and K. E. Parsopoulos. Fairness issues in resource allocation schemes for wireless visual sensor networks. In *Proceedings of SPIE*, volume 8666, pages 866601–866601–12, 2013.

[135] K. Pandremmenou, L. P. Kondi, and K. E. Parsopoulos. Geometric bargaining approach for optimizing resource allocation in wireless visual sensor networks. *IEEE Transactions on Circuits and Systems for Video Technology*, 23(8):1388–1401, Aug. 2013.

[136] K. Pandremmenou, L. P. Kondi, and K. E. Parsopoulos. A study on visual sensor network cross-layer resource allocation using quality-based criteria and meta-heuristic optimization algorithms. *Applied Soft Computing*, 26:149–165, 2015.

[137] K. Pandremmenou, L. P. Kondi, K. E. Parsopoulos, and E. S. Bentley. Game-theoretic solutions through intelligent optimization for efficient resource management in wireless visual sensor networks. *Signal Processing: Image Communication*, 29(4):472–493, 2014.

[138] K. Pandremmenou, M. Shahid, L. P. Kondi, and B. Lövström. A no-reference bitstream-based perceptual model for video quality estimation of videos affected by coding artifacts and packet losses. In *Proceedings of SPIE*, volume 9394, pages 93941F–93941F–12, 2015.

[139] K. Pandremmenou, M. Shahid, L. P. Kondi, and B. Lövström. On the improvement of no-reference mean opinion score estimation accuracy by following a frame-level regression approach. In *IEEE International Conference on Image Processing (ICIP)*, Sept. 2015. (to appear).

[140] K. Pandremmenou, M. Shahid, L. P. Kondi, A. Rossholm, and B. Lövström. Perceptual quality estimation of H.264/AVC videos using reduced-reference and no-reference models. Submitted.

[141] K. Pandremmenou, N. Tziortziotis, L. P. Kondi, and K. Blekas. Resource allocation in visual sensor networks using a reinforcement learning framework. In *18th International Conference on Digital Signal Processing (DSP)*, pages 1–6, Jul. 2013.

[142] K. Pandremmenou, N. Tziortziotis, S. Paluri, W. Zhang, K. Blekas, L. P. Kondi, and S. Kumar. Quality optimization of H.264/AVC video transmission over noisy environments using a sparse regression framework. volume 9410, pages 94100D–94100D–10, 2015.

[143] H. Park and M. Van der Schaar. Bargaining strategies for networked multimedia resource management. *IEEE Transactions on Signal Processing*, 55(7):3496–3511, 2007.

[144] H. Park and M. Van der Schaar. Congestion game modeling for brokerage based multimedia resource management. In *Packet Video 2007*, pages 18–25, Nov. 2007.

[145] H. Park and M. Van der Schaar. Fairness strategies for multi-user multimedia applications in competitive environments using the Kalai-Smorodinsky bargaining solution. In *IEEE International Conference on Acoustics, Speech and Signal Processing (ICASSP)*, volume 2, Apr. 2007.

[146] H. Park and M. Van der Schaar. Coalition-based resource negotiation for multimedia applications in informationally decentralized networks. *IEEE Transactions on Multimedia*, 11(4):765–779, Jun. 2009.

[147] H. Park and M. Van der Schaar. Fairness strategies for wireless resource allocation among autonomous multimedia users. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(2):297–309, Feb. 2010.

[148] K. E. Parsopoulos and M. N. Vrahatis. Recent approaches to global optimization problems through particle swarm optimization. *Natural Computing*, 1(2-3):235–306, 2002.

[149] K. E. Parsopoulos and M. N. Vrahatis. Studying the performance of unified particle swarm optimization on the single machine total weighted tardiness problem. In A. Sattar and B. H. Kang, editors, *Lecture Notes in Artificial Intelligence (LNAI)*, volume 4304, pages 760–769. Springer, 2006.

[150] K. E. Parsopoulos and M. N. Vrahatis. *Particle Swarm Optimization and Intelligence: Advances and Applications*. Information Science Publishing (IGI Global), 2010.

[151] I. C. Paschalidis, W. Lai, and D. Starobinski. On maximizing the utility of uplink transmissions in sensor networks under explicit fairness constraints. In *43rd IEEE Conference on Decision and Control, 2004.*, volume 1, pages 1010–1015, Dec. 2004.

[152] P. Pérez, J. Macias, J. J. Ruiz, and N. Garcia. Effect of packet loss in video quality of experience. *Bell Labs Technical Journal*, 16(1):91–104, Jun. 2011.

[153] H. J. M. Peters. *Axiomatic Bargaining Game Theory*. Theory and decision library. Kluwer Academic Publishers, Dordrecht, 1992.

[154] M. H. Pinson and S. Wolf. A new standardized method for objectively measuring video quality. *IEEE Transactions on Broadcasting*, 50(3):312–322, Sept. 2004.

[155] N. Ponomarenko, V. Lukin, K. Egiazarian, J. Astola, M. Carli, and F. Battisti. Color image database for evaluation of image quality metrics. In *2008 IEEE 10th Workshop on Multimedia Signal Processing*, pages 403–408, Oct. 2008.

[156] M. L. Puterman. *Markov Decision Processes: Discrete Stochastic Dynamic Programming*. John Wiley & Sons, Inc., 1994.

[157] E. S. Pynadath and L. P. Kondi. Cross-layer optimization with power control in DS-CDMA visual sensor networks. In *IEEE International Conference on Image Processing (ICIP)*, pages 25–28, Oct. 2006.

[158] A. Raiconi and M. Gentili. Exact and metaheuristic approaches to extend lifetime and maintain connectivity in wireless sensors networks. In *INOC*, volume 6701 of *Lecture Notes in Computer Science*, pages 607–619. Springer, 2011.

[159] V. T. Raisinghani and S. Iyer. Cross-layer feedback architecture for mobile device protocol stacks. *IEEE Communications Magazine*, 44(1):85–92, Jan. 2006.

[160] T. Rappaport. *Wireless Communications: Principles and Practice*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 2nd edition, 2001.

[161] A. R. Reibman and D. Poole. Predicting packet-loss visibility using scene characteristics. In *Packet Video*, pages 308–317, Nov. 2007.

[162] A. R. Reibman, V. A. Vaishampayan, and Y. Sermadevi. Quality monitoring of video over a packet network. *IEEE Transactions on Multimedia*, 6(2):327–334, Apr. 2004.

[163] M. Ries, O. Nemethova, and M. Rupp. Motion based reference-free quality estimation for H.264/AVC video streaming. In *2nd International Symposium on Wireless Pervasive Computing (ISWPC)*, Feb. 2007.

[164] A. Rossholm and B. Lövström. A new low complex reference free video quality predictor. In *10th IEEE Workshop on Multimedia Signal Processing*, pages 765–768, Oct. 2008.

[165] G. A. Rummery and M. Niranjan. On-line Q-learning using connectionist systems. Technical report, 1994.

[166] K. Schittkowski. An active set strategy for solving optimization problems with up to 200,000,000 nonlinear constraints. *Applied Numerical Mathematics*, 59(12):2999–3007, 2009.

[167] H. Schulzrinne, S. L. Casner, R. Frederick, and V. Jacobson. RTP: A transport protocol for real-time applications. IETF Request for Comments: RFC 3550, Jul. 2003.

[168] G. M. Schuster, G. Melnikov, and A. K. Katsaggelos. A review of the minimum maximum criterion for optimal bit allocation among dependent quantizers. *IEEE Transactions on Multimedia*, 1(1):3–17, Mar. 1999.

[169] M. Scmitt and R. Wanka. Particle swarm optimization almost surely finds local optima. *Theoretical Computer Science*, 561, Part A(0):57–72, 2015. Genetic and Evolutionary Computation.

[170] M. Shahid, A. Rossholm, and B. Lövström. A reduced complexity no-reference artificial neural network based video quality predictor. In *4th International Congress on Image and Signal Processing*, volume 1, pages 517–521, Oct. 2011.

[171] M. Shahid, A. Rossholm, and B. Lövström. A no-reference machine learning based video quality predictor. In *5th International Workshop on Quality of Multimedia Experience (QoMEX)*, pages 176–181, 2013.

[172] M. Shahid, A. Rossholm, B. Lövström, and H.-J. Zepernick. No-reference image and video quality assessment: A classification and review of recent approaches. *EURASIP Journal on Image and Video Processing*, 2014(40), 2014.

[173] J. Shaikh. *Non-Intrusive Network-Based Estimation of Web Quality of Experience Indicators*. PhD thesis, Blekinge Institute of Technology (BTH), Sweden, 2012.

[174] T. Shanableh. No-reference PSNR identification of MPEG video using spectral regression and reduced model polynomial networks. *IEEE Signal Processing Letters*, 17(8):735–738, Aug. 2010.

[175] T. Shanableh. Prediction of structural similarity index of compressed video at a macroblock level. *IEEE Signal Processing Letters*, 18(5):335–338, May 2011.

[176] T. Shanableh. A regression-based framework for estimating the objective quality of HEVC coding units and video frames. *Signal Processing: Image Communication*, 34(0):22–31, 2015.

[177] H. Sheikh, Z. Wang, A. Bovik, and L. Cormack. In *Image and Video Quality Assessment Research at LIVE*, 2015. [Online:]. Available: http://live.ece.utexas.edu/research/quality/.

[178] Y. Shen, Y. Liu, N. Qiao, L. Sang, and D. Yang. QoE-based evaluation model on video streaming service quality. In *IEEE Globecom Workshops, 2012*, pages 1314–1318, Dec. 2012.

[179] M. Shindarov, S. Fidanova, and P. Marinov. Wireless sensor positioning algorithm. In *6th IEEE International Conference on Intelligent Systems (IS)*, pages 419–424, Sept. 2012.

[180] M. Slanina, V. Ricny, and R. Forchheimer. A novel metric for H.264/AVC no-reference quality assessment. In *14th International Workshop on Systems, Signals and Image Processing, 2007 and 6th EURASIP Conference focused on Speech and Image Processing, Multimedia Communications and Services*, pages 114–117, Jun. 2007.

[181] S. Soro and W. Heinzelman. A survey of visual sensor networks. *Advances in Multimedia*, 2009:1–22, 2009.

[182] R. Soundararajan and A. C. Bovik. Video quality assessment by reduced reference spatio-temporal entropic differencing. *IEEE Transactions on Circuits and Systems for Video Technology*, 23(4):684–694, Apr. 2013.

[183] N. Staelens, D. Deschrijver, E. Vladislavleva, B. Vermeulen, T. Dhaene, and P. Demeester. Constructing a no-reference H.264/AVC bitstream-based video quality metric using genetic programming-based symbolic regression. *IEEE Transactions on Circuits and Systems for Video Technology*, 23(8):1322–1333, Aug. 2013.

[184] N. Staelens, N. Vercammen, Y. Dhondt, B. Vermeulen, P. Lambert, R. Van De Walle, and P. Demeester. Viqid: A no-reference bit stream-based visual quality impairment detector. In *2nd International Workshop on Quality of Multimedia Experience (QoMEX)*, pages 206–211, Jun. 2010.

[185] P. N. Suganthan. Particle swarm optimizer with neighborhood operator. In *Proceedings of the IEEE Congress on Evolutionary Computation*, pages 1958–1961, Washington, D.C., USA, 1999.

[186] G. J. Sullivan, P. N. Topiwala, and A. Luthra. The H.264/AVC advanced video coding standard: Overview and introduction to the fidelity range extensions. In *SPIE Conference on Applications of Digital Image Processing XXVII*, volume 5558, pages 454–474, 2004.

[187] J. E. Suris, L. A. Dasilva, Z. Han, A. B. Mackenzie, and R. S. Komali. Asymptotic optimality for distributed spectrum sharing using bargaining solutions. *IEEE Transactions on Wireless Communications*, 8(10):5225–5237, Oct. 2009.

[188] R. S. Sutton and A. G. Barto. *Reinforcement Learning: An Introduction*. MIT Press Cambridge, USA, 1998.

[189] G. Tan. *Improving aggregate user utilities and providing fairness in multi-rate wireless LANs*. PhD thesis, Massachusetts Institute of Technology (MIT), Cambridge, MA, USA, 2006.

[190] A. Tanenbaum. *Computer Networks*. Prentice Hall Professional Technical Reference, 4th edition, 2002.

[191] J. Tang and X. Zhang. Cross-layer resource allocation over wireless relay networks for Quality of Service provisioning. *IEEE Journal on Selected Areas in Communications*, 25(4):645–656, 2007.

[192] R. Tibshirani. Regression shrinkage and selection via the LASSO. *Journal of the Royal Statistical Society, Series B*, 58:267–288, 1994.

[193] R. Tibshirani. The LASSO method for variable selection in the Cox model. *Statist. Med.*, 16(4):385–395, 1997.

[194] W. H. Tranter, D. P. Taylor, R. E. Ziemer, N. F. Maxemchuk, and J. W. Mark, editors. *The Best of the Best: Fifty Years of Communications and Networking Research*. Wiley-IEEE Press, 2007.

[195] I. C. Trelea. The particle swarm optimization algorithm: convergence analysis and parameter selection. *Information Processing Letters*, 85(6):317–325, 2003.

[196] D. Triantafyllopoulou, N. Passas, E. Zervas, and L. Merakos. Optimized cross-layer adaptation for improved resource management in wireless networks. In *Proceedings of the WiOpt'10: Modeling and Optimization in Mobile, Ad Hoc, and Wireless Networks*, pages 380–386, Avignon France, 2010.

[197] F. Van den Bergh and A. P. Engelbrecht. A study of particle swarm optimization particle trajectories. *Information Sciences*, 176(8):937–971, 2006.

[198] M. Van der Schaar and S. S. Nandagopalan. Cross-layer wireless multimedia transmission: challenges, principles, and new paradigms. *IEEE Wireless Communications*, 12(4):50–58, 2005.

[199] Video Quality Experts Group. VQEG report on validation of video quality models for high definition video content, 2010.

[200] A. Viterbi. Convolutional codes and their performance in communication systems. *IEEE Transactions on Communication Technology*, 19(5):751–772, 1971.

[201] A. Vosoughi, P. C. Cosman, and L. B. Milstein. Joint source-channel coding and unequal error protection for video plus depth. *IEEE Signal Processing Letters*, 22(1):31–34, Jan. 2015.

[202] R. A. Waltz, J. L. Morales, J. Nocedal, and D. Orban. An interior algorithm for nonlinear optimization that combines line search and trust region steps. *Math. Program.*, 107(3):391–408, Jul. 2006.

[203] B. Wang, D. Zou, and R. Ding. Support vector regression based video quality prediction. In *IEEE International Symposium on Multimedia (ISM)*, pages 476–481, Dec. 2011.

[204] C. Wang, T.-L. Lin, and P. C. Cosman. Network-based model for video packet importance considering both compression artifacts and packet losses. In *IEEE Global Telecommunications Conference (GLOBECOM 2010)*, pages 1–5, Dec. 2010.

[205] C. Wang, T.-L. Lin, and P. C. Cosman. Packet dropping for H.264 videos considering both coding and packet-loss artifacts. In *18th International Packet Video Workshop (PV)*, pages 165–172, Dec. 2010.

[206] H. Wang and A. K. Katsaggelos. Resource-distortion optimal video coding and communications. In *1st International Conference on Multimedia Services Access Networks*, pages 10–14, Jun. 2005.

[207] H. Wang, L. P. Kondi, A. Luthra, and S. Ci. *4G Wireless Video Communications*. Wiley Publishing, 2009.

[208] J. Wang, T. Korhonen, and Y. Zhao. Weighted network utility maximization aided by combined queueing priority in OFDMA systems. In *IEEE International Conference on Communications (ICC)*, pages 3323–3327, May 2008.

[209] W. Wang, D. Peng, H. Wang, H. Sharif, and H.-H. Chen. Energy-constrained distortion reduction optimization for wavelet-based coded image transmission in wireless sensor networks. *IEEE Transactions on Multimedia*, 10(6):1169–1180, Oct. 2008.

[210] Z. Wang, A. C. Bovik, H. R. Sheikh, and E. P. Simoncelli. Image quality assessment: from error visibility to structural similarity. *IEEE Transactions on Image Processing*, 13(4):600–612, Apr. 2004.

[211] Z. Wang, H. R. Sheikh, and A. C. Bovik. Objective video quality assessment. In *The Handbook of Video Databases: Design and Applications*, pages 1041–1078. CRC Press, 2003.

[212] C. Weeraddana, W. Li, M. Codreanu, and M. Latva-aho. Weighted sum-rate maximization for downlink OFDMA systems. In *42nd Asilomar Conference on Signals, Systems and Computers*, pages 990–994, Oct. 2008.

[213] T. Wiegand, G. J. Sullivan, G. Bjontegaard, and A. Luthra. Overview of the H.264/AVC video coding standard. *IEEE Transactions on Circuits and Systems for Video Technology*, 13(7):560–576, 2003.

[214] S. Winkler. Video quality and beyond. *in Proceedings of European Signal Processing Conference*, 446(3):150–153, 2007.

[215] I. H. Witten and E. Frank. *Data mining: Practical machine learning tools and techniques*. Morgan Kaufmann, third edition, 2011.

[216] M. Wu and C. W. Chen. Multiple bitstream image transmission over wireless sensor networks. In *Proceedings of the IEEE Sensors*, volume 2, pages 727–731, 2003.

[217] C. C. Wüst, L. Steffens, W. F. Verhaegh, R. J. Bril, and C. Hentschel. QoS control strategies for high-quality video processing. *Real-Time Systems*, (1-2), May 2005.

[218] B. Yang, Y. Shen, G. Feng, and X. Guan. Fair resource allocation using bargaining over OFDMA relay networks. In *Proceedings of the 48th IEEE Conference on Decision and Control, 2009 held jointly with the 28th Chinese Control Conference (CDC/CCC)*, pages 585–590, 2009.

[219] F. Yang, S. Wan, Q. Xie, and H. R. Wu. No-reference quality assessment for networked video via primary analysis of bit stream. *IEEE Transactions on Circuits and Systems for Video Technology*, 20(11):1544–1554, Nov. 2010.

[220] Y. Yang, X. Wen, W. Zheng, L. Yan, and A. Zhang. A no-reference video quality metric by using inter-frame encoding characters. In *14th International Symposium on Wireless Personal Multimedia Communications (WPMC), 2011*, pages 1–5, Oct. 2011.

[221] Z. Yang and K. Nahrstedt. A bandwidth management framework for wireless camera array. In *Proceedings of the International Workshop on network and operating systems support for digital audio and video*, pages 147–152. ACM, 2005.

[222] J. You, J. Korhonen, and A. Perkis. Spatial and temporal pooling of image quality metrics for perceptual video quality assessment on packet loss streams. In *IEEE International Conference on Acoustics Speech and Signal Processing (ICASSP)*, pages 1002–1005, Mar. 2010.

[223] F. Yuan and E. Cheng. Reduced-reference metric design for video quality measurement in wireless application. In *11th IEEE International Conference on Communication Technology*, pages 641–644, Nov. 2008.

[224] J. Yuan and W. Yu. Joint source coding, routing and power allocation in wireless sensor networks. *IEEE Transactions on Communications*, 56(6):886–896, 2008.

[225] T. Zahariadis and S. Voliotis. Open issues in wireless visual sensor networking. In *Systems, Signals and Image Processing, 2007 and 6th EURASIP Conference focused on Speech and Image Processing*, pages 335–338, Jun. 2007.

[226] B. Zatt, M. Porto, J. Scharcanski, and S. Bampi. GOP structure adaptive to the video content for efficient H.264/AVC encoding. In *IEEE International Conference on Image Processing (ICIP)*, pages 3053–3056, Sept. 2010.

[227] K. Zeng and Z. Wang. Quality-aware video based on robust embedding of intra- and inter-frame reduced-reference features. In *IEEE International Conference on Image Processing (ICIP)*, pages 3229–3232, Sept. 2010.

[228] H. Zhang, A. K. Ramasubramonian, K. Kar, and J. W. Woods. Distortion-optimal receiver grouping for MD-FEC coded video streaming. In *Visual Communications and Image Processing (VCIP)*, pages 1–6. IEEE, 2012.

[229] R. Zhang, S. L. Regunathan, and K. Rose. Video coding with optimal inter/intra-mode switching for packet loss resilience. *IEEE Journal on Selected Areas in Communications*, 18(6):966–976, 2000.

[230] C. Zhou, C.-W. Lin, X. Zhang, and Z. Guo. A novel JSCC scheme for UEP-based scalable video transmission over MIMO systems. *IEEE Transactions on Circuits and Systems for Video Technology*, PP(99):1–1, 2014.

GLOSSARY

ARQ	Automatic Repeat Request
AS	Active Set
AVC	Advanced Video Coding
AWGN	Additive White Gaussian Noise
B	Bidirectionally predictive
BER	Bit Error Rate
BPSK	Binary Phase Shift Keying
CCU	Centralized Control Unit
CIF	Common Intermediate Format
CMSE	Cumulative Mean Squared Error
CNBS	Class Nash Bargaining Solution
CV	Cross Validation
DS-CDMA	Direct Sequence-Code Division Multiple Access
EPFL	Ecole Polytechnique Fédérale de Lausanne
ES	Exhaustive Search
FEC	Forward Error Correction
FFS	Forward Feature Selection
FMO	Flexible Macroblock Ordering
FR	Full-Reference
GLM	Generalized Linear Model
GOP	Group Of Pictures
G.LASSO	Global Least Absolute Shrinkage and Selection Operator
HEVC	High Efficiency Video Coding
HPSOAS	Hybrid Particle Swarm Optimization Active Set
HVS	Human Visual System
I	Intra
IDR	Instantaneous Decoding Refresh
IP	Interior Point
JI	Jain's Index
KKT	Karush-Kuhn-Tucker
KSBS	Kalai-Smorodinsky Bargaining Solution
LASSO	Least Absolute Shrinkage and Selection Operator

LL	Link Layer
L.LASSO	Local Least Absolute Shrinkage and Selection Operator
MAC	Medium Access Control
MAD	Minimized Average Distortion
MAE	Mean Absolute Error
MAI	Multiple Access Interference
MANET	Mobile Ad-hoc NETworks
MB	MacroBlock
MCEC	Motion Compensated Error Concealment
MDP	Markov Decision Process
MD-FEC	Multiple Description-Forward Error Correction
MIMO	Multiple Input Multiple Output
MMD	Minimized Maximum Distortion
MoC	Modulus of Continuity
MOS	Mean Opinion Score
MSE	Mean Squared Error
MTU	Maximized Total Utility
NAL	Network Abstraction Layer
NBS	Nash Bargaining Solution
NN	Neural Network
NNBS	Node Nash Bargaining Solution
NR	No-Reference
NRMSE	Normalized Root Mean Squared Error
NR-B	No-Reference-Bitstream
NR-P	No-Reference-Pixel
OFDMA	Orthogonal Frequency Division Multiple Access
OLS	Ordinary Least Squares
OR	Outlier Ratio
OSI	Open Systems Interconnection
P	Predictive
PCA	Principal Component Analysis
PCC	Pearson Correlation Coefficient
PEVQ	Perceptual Evaluation of Video Quality
PF	Performance to Fairness
PLR	Packet Loss Rate
PLSR	Partial Least Squares Regression
PoliMi	Politecnico di Milano
PSNR	Peak Signal to Noise Ratio
PSO	Particle Swarm Optimization
QBP	Quartile Based Prioritization

QCIF	Quarter Common Intermediate Format
QoE	Quality of Experience
QoS	Quality of Service
QP	Quantization Parameter
RCPC	Rate Compatible Punctured Convolutional Codes
RL	Reinforcement Learning
RMSE	Root Mean Squared Error
RR	Reduced Reference
RTP	Real-time Transport Protocol
SBM	Slice Boundary Mismatch
SNR	Signal to Noise Ratio
SRCs	Sources
SROCC	Spearman Rank Order Correlation Coefficient
SSCQE	Single Stimulus Continuous Quality Evaluation
SSIM	Structural SIMilarity
SVM	Support Vector Machines
SVR	Support Vector Regression
TCP/IP	Transmission Control Program/Internet Protocol
TD	Temporal Difference
TP	Total Power
TRGR	Two-Ray Ground Reflection
TRR	Trust Region Reflective
TU	Total Utility
UAV	Unmanned Aerial Vehicle
UEP	Unequal Error Protection
URDC	Universal Rate Distortion Characteristics
VCL	Video Coding Layer
VQA	Video Quality Assessment
VQEG	Video Quality Experts Group
VQM	Video Quality Metric
VSN	Visual Sensor Network
w.MTU	weighted Maximized Total Utility
bps	bits per second
dB	deciBel
fps	frames per second
Hz	Hertz
mW	milliWatt
W	Watt

AUTHOR'S PUBLICATIONS

Journal Papers

J1. M. Shahid, K. Pandremmenou, L. P. Kondi, A. Rossholm, B. Lövström, "Perceptual Quality Estimation of H.264/AVC Videos Using Reduced-Reference and No-Reference Models", submitted.

J2. K. Pandremmenou, L. P. Kondi, K. E. Parsopoulos, "A Study on Visual Sensor Network Cross-Layer Resource Allocation Using Quality-Based Criteria and Meta-heuristic Optimization Algorithms", Applied Soft Computing, vol. 26, pp. 149-165, Jan. 2015.

J3. K. Pandremmenou, L. P. Kondi, K. E. Parsopoulos, E. S. Bentley, "Game-Theoretic Solutions through Intelligent Optimization for Efficient Resource Management in Wireless Visual Sensor Networks", Signal Processing: Image Communication, vol.29, issue 4, pp. 472-493, Apr. 2014.

J4. K. Pandremmenou, L. P. Kondi, K. E. Parsopoulos, "Geometric Bargaining Approach for Optimizing Resource Allocation in Wireless Visual Sensor Networks", IEEE Transactions on Circuits and Systems for Video Technology, vol.23, no.8, pp. 1388-1401, Aug. 2013.

Conference Papers

C1. K. Pandremmenou, M. Shahid, L. P. Kondi, B. Lövström, "On the Improvement of No-Reference Mean Opinion Score Estimation Accuracy by Following a Frame-Level Regression Approach", in the 22nd IEEE International Conference on Image Processing (ICIP 2015), Quebec, Canada, Sept. 2015 (to appear).

C2. K. Pandremmenou, M. Shahid, L. P. Kondi, B. Lövström, "A No-Reference Bitstream-based Perceptual Model for Video Quality Estimation of Videos Affected by Coding Artifacts and Packet Losses", in Human Vision and Electronic Imaging XX, Proceedings of SPIE-IS&T Electronic Imaging, SPIE Vol. 9394, San Francisco, CA, Feb. 2015.

C3. K. Pandremmenou, N. Tziortziotis, S. Paluri, W. Zhang, K. Blekas, L. P. Kondi, S. Kumar, "Quality Optimization of H.264/AVC Video Transmission over Noisy Environments Using a Sparse Regression Framework", in Visual Information Processing and Communication VI, Proceedings of SPIE-IS&T Electronic Imaging, SPIE Vol. 9410, San Francisco, CA, Feb. 2015.

C4. K. Pandremmenou, N. Tziortziotis, L. P. Kondi, K. Blekas, "Resource Allocation in Visual Sensor Networks Using a Reinforcement Learning Framework", in the 18th IEEE International Conference on Digital Signal Processing (DSP 2013), Santorini, Greece, Jul. 2013.

C5. K. Pandremmenou, L. P. Kondi, K. E. Parsopoulos, "Fairness Issues in Resource Allocation Schemes for Wireless Visual Sensor Networks", in Visual Information Processing and Communication IV, Proceedings of SPIE-IS&T Electronic Imaging, SPIE Vol. 8666, San Francisco, CA, Feb. 2013.

C6. K. Pandremmenou, L. P. Kondi, K. E. Parsopoulos, "Kalai-Smorodinsky Bargaining Solution for Optimal Resource Allocation over Wireless DS-CDMA Visual Sensor Networks", in Visual Information Processing and Communication III, Proceedings of SPIE-IS&T Electronic Imaging, SPIE Vol. 8305, San Francisco, CA, Jan. 2012.

C7. K. Pandremmenou, L. P. Kondi, K. E. Parsopoulos, "Optimal Power Allocation and Joint Source-Channel Coding for Wireless DS-CDMA Visual Sensor Networks Using the Nash Bargaining Solution", in the 36th IEEE International Conference on Accoustics Speech and Signal Processing (ICASSP 2011), Prague, Czech Republic, pp. 2340-2343, May 2011.

C8. K. Pandremmenou, L. P. Kondi, K. E. Parsopoulos, "Optimal Power Allocation and Joint Source-Channel Coding for Wireless DS-CDMA Visual Sensor Networks", in Visual Information Processing and Communication II, Proceedings of SPIE-IS&T Electronic Imaging, SPIE Vol. 7882, San Francisco, CA, Jan. 2011.

SHORT VITA

Katerina Pandremmenou was born in Larissa, Greece, in 1985. She received the B.Sc in Computer Science in 2008, from the Computer Science Department of the University of Crete. In 2010, she received the M.Sc (Technologies and Applications) from the Computer Science Department of the University of Ioannina and since 2011 she has been a PhD student in the same Department under the supervision of Prof. Lisimachos P. Kondi. She is a member of the Information Processing and Analysis research group (I.P.AN.) and she has been involved in 2 research projects. Her research interests are in the areas of wireless communications and video quality assessment.