Metaheuristic Optimization for Logistics in Natural Disasters

Thomai Korkou, Dimitris Souravlias, Konstantinos Parsopoulos, and Konstantina Skouri

Abstract Logistics in natural disasters or emergencies involve highly complicated optimization problems with diverse characteristics. The contribution of the present paper is twofold. First, it introduces a multi-period model aiming to minimize the shortages of different relief products in a number of affected areas. The relief products are transported via multiple modes of transportation from dispatch centers to these areas, while adhering to traffic restrictions. A test suite of benchmark problems with diverse characteristics is generated from the proposed model and solved to optimality with CPLEX. The test suite is used for benchmarking a number of established metaheuristics. Necessary modifications are introduced in the algorithms, in order to fit the special requirements of the specific problem type. The algorithms' performance is assessed in terms of solution accuracy with respect to the optimal solutions. Comparisons among the employed metaheuristics offer valuable insight regarding their ability to tackle humanitarian logistics problems.

Keywords Humanitarian logistics • Metaheuristics • Algorithm portfolios

1 Introduction

Humanitarian Logistics (HL) has attracted increasing interest due to the exponential surge in natural and man-made disasters (Özdamar et al., 2004). Ranging from earthquakes to tsunamis, natural disasters have produced startling devastation with major death tolls and economical consequences worldwide. Recent examples of severe natural disasters include the Nepal earthquake in 2015, Japan earthquake and tsunami in 2011, Haiti earthquake in 2010, Myanmar cyclone Nargis in 2008, and

Th. Korkou • D. Souravlias • K. Parsopoulos (✉)
Department of Computer Science and Engineering, University of Ioannina, GR-45110 Ioannina, Greece
e-mail: thkorkou@cs.uoi.gr; dsouravl@cs.uoi.gr; kostasp@cs.uoi.gr

K. Skouri
Department of Mathematics, University of Ioannina, GR-45110 Ioannina, Greece
e-mail: kskouri@uoi.gr

I.S. Kotsireas et al. (eds.), *Dynamics of Disasters—Key Concepts, Models, Algorithms, and Insights*, Springer Proceedings in Mathematics & Statistics 185,
DOI 10.1007/978-3-319-43709-5_7

Pakistan earthquake in 2005. All these events caused thousands of deaths and left numerous people homeless, needing emergent assistance. The long lasted, slowly progressing recovery efforts forced thousands of wounded people to continue living in refugee camps that were set up immediately after the disaster.

HL plays a crucial role in addressing disaster relief operations problems. Quoting from (Thomas and Kopczak, 2005), HL is responsible for "*planning, implementing and controlling the efficient, cost-effective flow and storage of goods and materials, as well as related information, from point of origin to point of consumption for the purpose of alleviating the suffering of vulnerable people*." It is widely perceived that HL constitutes a powerful tool, capable of making the difference between success and failure in managing disaster relief operations (Cozzolino et al., 2012; Van Wassenhove, 2006).

Moreover, according to the *Pan American Health Organization* (PAHO), HL focuses on the procurement, transportation, storage, and distribution of relief supplies (PAHO, 2001). Procurement ensures the availability of the demanded resources, while transportation is responsible for the transfer of the latter to the wounded people in the affected areas. Transportation plans for the goods must take into account the available infrastructure, which is frequently damaged from the disaster. Storage protects the relief commodities until their delivery to the points of consumption. Finally, distribution is responsible for the aid delivery to beneficiaries, avoiding problems triggered by external factors (Balcik et al., 2008).

Although HL is significant to prevent from consequences on people's health or life loss, the relevant literature is limited when compared to commercial logistics. In the latter, the main goal is typically the cost reduction (Van Wassenhove, 2006). HL promotes different priorities than cost, thereby introducing new aspects to the underlying problems. Various aspects of HL have been investigated in several studies (Diaz et al., 2013; Galindo and Batta, 2013). Transportation and routing were studied by Barbarosoglu and Arda (2004), Han et al. (2011), Huang et al. (2013), Yi and Özdamar (2007), Yi and Kumar (2007), and Yuan and Wang (2009). Specifically, Barbarosoglu and Arda (2004) proposed a two-stage stochastic programming model to plan the transportation of vital relief resources to the affected areas, taking into account the variations in demand, supply, and route capacity. Han et al. (2011) proposed a model based on Lagrangian relaxation to address a problem of delivering relief commodities.

Huang et al. (2013) studied a continuous approximation approach by utilizing aggregated instance data to explore appropriate routes for aid response teams. Yi and Özdamar (2007) addressed a dynamic coordination model for evacuation and support in disaster response situations. Yi and Kumar (2007) employed a meta-heuristic algorithm, namely ant colony optimization to explore optimal solutions for wounded people with respect to speed delivery. Yuan and Wang (2009) presented two mathematical models for path-selection, taking into account the travel speed on each route as well as hectic situations that frequently follow disasters.

Supply chain and procurement were investigated by Balcik and Beamon (2008), Clark and Culkin (2013), Falasca and Zobel (2011), Peng and Chen (2011), as well as by Tatham and Kovacs (2010), and Taylor and Pettit (2009). More specifically, Balcik and Beamon (2008) presented a model aiming to define the optimal number of distribution centers and the quantity of supplies. Clark and Culkin (2013) proposed a mathematical transshipment multi-commodity supply chain flow model to satisfy demand for affected people. Falasca and Zobel (2011) explored the need for models that support procurement through a two-stage decision model. Peng and Chen (2011) studied how transportation and information delays affect disaster relief operations. Tatham and Kovacs (2010) suggested a model of swift trust aiming to enhance disaster relief operations. Finally, Taylor and Pettit (2009) explored how lean logistics techniques such as value chain analysis perform in HL.

In addition, a number of research works are devoted to the study of distribution and supply location (Chang et al., 2007; Sheu et al., 2005; Van Hentenryck et al., 2010; Vitoriano et al., 2011; Zhang et al., 2012). Specifically, Chang et al. (2007) addressed a flood emergency logistics problem by presenting two stochastic programming models that were solved by using a sample average approximation scheme. Sheu et al. (2005) solved a disaster relief distribution problem by employing fuzzy clustering techniques and fuzzy linear programming. Van Hentenryck et al. (2010) proposed a hybrid optimization algorithm to solve the single commodity allocation problem. Vitoriano et al. (2011) solved a distribution model, taking into account performance measures such as equity of the distribution and security of routes. Zhang et al. (2012) proposed a heuristic approach based on linear programming and network optimization to solve a multiple-resource and multiple-depot disaster relief problem.

The complexity of HL optimization problems requires efficient solvers that can produce satisfactory solutions within strict time constraints. *Metaheuristics* have been recognized as valuable optimization tools for this purpose. The term *metaheuristics* mostly refers to nature-inspired algorithms with stochastic components (Glover, 1986). Such algorithms are able to offer (sub-)optimal solutions to difficult optimization problems within reasonable amount of time. However, this comes at the cost of dubious optimality of the detected solution. The dynamic of metaheuristics is governed by two major properties, namely *exploration* and *exploitation* (Blum and Roli, 2003). The first one is the ability to perform diverse search without neglecting regions of the search space. The latter is the ability to conduct more refined search in the neighborhood of already detected candidate solutions. Global optimality of the solutions is highly related to the appropriate balance of these two components.

Metaheuristics have gained increasing popularity in academia and industry due to their successful application in solving complex real-world problems. This can be attributed to their efficiency in decision making, simplicity, noise tolerance, and easy implementation (Liu and Ye, 2014). There is a significant amount of research studying the performance of metaheuristics in various problems in logistics, while recently several works appeared also in the growing area of HL (Yan and Shih, 2012; Yi and Kumar, 2007; Zheng et al., 2014).

Recently, Liu and Ye (2014) studied a multi-period problem, taking into account limited supply and transportation capacity that aims to minimize losses caused by (i) the mismatch between supplies and demand and (ii) the transportation time due to logistics processes. In the present study, we consider a similar model where the objective is the minimization of losses caused by the mismatch between supply and demand of relief resources in the affected areas, while taking into account the already existing quantities (if any) and the importance of the different resources. Furthermore, apart from constraints related to the number, volume, and load capacity of vehicles, we consider also road capacity constraints. The latter is the source of bottleneck in supply chain due to the increase of relief vehicles and possible decrease in transportation capacity, defined by the authorities (Besiou et al., 2011).

A test suite of benchmark problems is produced for the proposed model and they are solved to optimality with the commercial CPLEX solver. In addition, a number of established metaheuristics, namely differential evolution (DE) (Price et al., 2005) and particle swarm optimization (PSO) (Parsopoulos and Vrahatis, 2010), are considered and their efficiency is studied on the test suite. In order to fit the special requirements of the test problems, appropriate modifications are made in their basic operations. Moreover, we consider an enhanced DE (eDE) variant (Mohamed, 2014), which is shown to produce significant performance improvement. All the aforementioned algorithms are also utilized in a parallel algorithm portfolio (AP) framework (Gomes and Selman, 1997, 2001; Huberman et al., 1997; Peng et al., 2010; Tang et al., 2014), according to a recently proposed scheme (Souravlias et al., 2014).

The rest of the paper is organized as follows: in Sect. 2 the mathematical formulation of the proposed model is given. The employed algorithms are comprehensively discussed in Sect. 3, while Sect. 4 contains details on the experimental setting and presentation of the obtained results. Sect. 5 concludes the paper.

2 Problem Formulation

In our model, we consider a set J of affected areas (AAs) and a set I of dispatch centers (DCs). Relief resources (commodities) are transported from DCs to AAs through a number of vehicles of different type and mode. In our case, ground and aerial vehicles of two sizes (big and small) are considered. We henceforth denote as C the set of commodities, M the set of transportation modes, and O_m the set of vehicles of mode $m \in M$. The planning horizon is finite and denoted as T. The complete notation used in our model is presented in Table 1.

The main optimization goal lies in specifying the optimal delivered quantities s_{cijm}^t per commodity $c \in C$ from DC i to AA j, using vehicles of transportation mode m, for each time period t. Moreover, we need to specify the optimal number v_{cijmo}^t of type o, mode m vehicles that are used to transport the commodities at each time period t. All decision variables assume integer values. The corresponding minimization problem is defined as follows:

$$\min \sum_{t\in T} \sum_{j\in J} \sum_{c\in C} b_{cj} \left(d_{cj}^t - \sum_{i\in I} \sum_{m\in M} s_{cijm}^t - L_{cj}^{t-1} \right)^2, \qquad (1)$$

Table 1 Notation used in the proposed model

Model parameters	*Description*
T	Planning horizon
I	Set of dispatch centers (DCs)
J	Set of affected areas (AAs)
C	Set of commodities
M	Set of transportation modes
m	Index denoting the transportation mode (ground, air)
O_m	Set of vehicle types of transportation mode m
o	Index denoting the vehicle type (big vehicle, small vehicle)
b_{cj}	Importance weight of commodity c in AA j
w_c	Unit weight of commodity c
$volume_c$	Unit volume of commodity c
cap_{mo}	Capacity of type o, mode m vehicle
vol_{mo}	Volume capacity of type o, mode m vehicle
d_{cj}^t	Demand for commodity c in AA j at time period t
k_{ijm}^t	Traffic restriction for mode m vehicles from DC i to AA j at time t
v_{imo}^t	Number of type o, mode m vehicles at DC i at time t
L_{cj}^t	Inventory level of commodity c in AA j at time t
Decision variables	*Description*
s_{cijm}^t	Delivered quantity of commodity c from DC i to AA j through transportation mode m at time t
v_{cijmo}^t	Number of type o, mode m vehicles used at period t to transport commodity c from DC i to AA j

where b_{cj} is a scalar weight of importance of commodity c at AA j. The model is subject to the following constraints:

$$L_{cj}^0 = Y_{cj}, \qquad \forall\, c \in C,\ \forall\, j \in J, \tag{2}$$

$$L_{cj}^t = \sum_{i \in I} \sum_{m \in M} s_{cijm}^t - d_{cj}^t + L_{cj}^{t-1}, \qquad \forall\, t \in T,\ \forall\, c \in C,\ \forall\, j \in J, \tag{3}$$

$$\sum_{c \in C} \sum_{j \in J} s_{cijm}^t\, w_c \leqslant \sum_{o \in O_m} v_{imo}^t\, cap_{mo}, \qquad \forall\, t \in T,\ \forall\, i \in I,\ \forall\, m \in M, \tag{4}$$

$$\sum_{c \in C} \sum_{j \in J} s_{cijm}^t\, vol_c \leqslant \sum_{o \in O_m} v_{imo}^t\, vol_{mo}, \qquad \forall\, t \in T,\ \forall\, i \in I,\ \forall m \in M, \tag{5}$$

$$s_{cijm}^t \leqslant \min \left\{ \frac{\sum_{o \in O_m} v_{cijmo}^t\, cap_{mo}}{w_c},\ \frac{\sum_{o \in O_m} v_{cijmo}^t\, vol_{mo}}{volume_c} \right\}, \tag{6}$$

$$\sum_{c \in C} \sum_{o \in O_m} v_{cijmo}^t \leqslant k_{ijm}^t, \qquad \forall\, t \in T,\ \forall\, i \in I,\ \forall\, m \in M,\ \forall j \in J, \tag{7}$$

$$\sum_{c \in C} \sum_{j \in J} \upsilon^t_{cijmo} \leqslant \upsilon^t_{imo}, \qquad \forall\, t \in T,\ \forall\, i \in I,\ \forall\, m \in M,\ \forall\, o \in O, \tag{8}$$

$$s^t_{cijm}, \upsilon^t_{cijmo}, L^t_{cj} \in \mathbb{N}, \quad \text{for all } t, c, i, j, m, o, \tag{9}$$

$$\upsilon^t_{imo}, d^t_{cj}, vol_{mo}, cap_{mo}, volume_c, w_c \in \mathbb{N}^+, \quad \text{for all } t, c, i, j, m, o, \tag{10}$$

$$\sum_{c \in C} b_{cj} = 1, \quad b_{cj} \in [0, 1], \qquad \forall j \in J. \tag{11}$$

Equation (2) accounts for the initial inventory level of commodity c that pre-exists at DC j. Equation (3) determines the inventory balance, which takes into account the demand of the commodity c and the replenishment quantity. Equations (4) and (5) refer to capacity and volume constraints, respectively. Equation (6) defines upper limits of the delivered quantity s^t_{cijm}, which is useful for bounding the decision variables. Equation (7) stands for traffic flow restrictions expected in natural disasters, e.g., roads that are partially damaged or destroyed, thereby reducing traffic capacity. Equation (8) ensures that the number of vehicles transporting the commodities in a particular AA does not exceed the total number of vehicles. Equations (9)–(11) define the appropriate domains of the decision variables and problem parameters.

The squared error in Eq. (1) can be replaced by the absolute error if metaheuristics are used. Nevertheless, the quadratic form is selected in order to render the problem solvable by CPLEX. Note that the objective function is also convex since it constitutes the sum of convex functions.

3 Employed Algorithms

In the following paragraphs, we briefly present the main features of the employed metaheuristics. For presentation purposes, we assume that the considered minimization problem is defined in the general form,

$$\min_{\mathbf{x} \in X \subset \mathbb{R}^n} f(\mathbf{x}), \tag{12}$$

where X is the (real-valued) search space. The only requirement on the objective function is the availability of $f(\mathbf{x})$ at any $\mathbf{x} \in X$. Appropriate modifications of the algorithms to handle the integer search spaces of the considered HL model are given later.

3.1 Differential Evolution

Differential Evolution (DE) was introduced by Storn and Price (1997). Being a population-based optimization algorithm, it proceeds by iteratively improving a population of candidate solutions,

$$S = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\},$$

consisting of N search points, called the *individuals*. The population is randomly and uniformly initialized over the search space. DE applies biologically inspired operators, namely *mutation*, *crossover*, and *selection*, on each individual in order to produce new candidate solutions by combining existing ones.

At each iteration (also called *generation*) g of the algorithm, a mutant vector $\mathbf{v}_i$ is generated for each individual $\mathbf{x}_i$, $i = 1, 2, \ldots, N$. This vector is produced by combining existing individuals from the population according to various *mutation operators*. The following are among the most popular ones:

$$\text{DE1}: \quad \mathbf{v}_i^{(g+1)} = \mathbf{x}_{\text{best}}^{(g)} + F\left(\mathbf{x}_{r_1}^{(g)} - \mathbf{x}_{r_2}^{(g)}\right), \tag{13}$$

$$\text{DE2}: \quad \mathbf{v}_i^{(g+1)} = \mathbf{x}_{r_1}^{(g)} + F\left(\mathbf{x}_{r_2}^{(g)} - \mathbf{x}_{r_3}^{(g)}\right), \tag{14}$$

$$\text{DE3}: \quad \mathbf{v}_i^{(g+1)} = \mathbf{x}_i^{(g)} + F\left(\mathbf{x}_{\text{best}}^{(g)} - \mathbf{x}_i^{(g)}\right) + F\left(\mathbf{x}_{r_1}^{(g)} - \mathbf{x}_{r_2}^{(g)}\right), \tag{15}$$

$$\text{DE4}: \quad \mathbf{v}_i^{(g+1)} = \mathbf{x}_{\text{best}}^{(g)} + F\left(\mathbf{x}_{r_1}^{(g)} - \mathbf{x}_{r_2}^{(g)}\right) + F\left(\mathbf{x}_{r_3}^{(g)} - \mathbf{x}_{r_4}^{(g)}\right), \tag{16}$$

$$\text{DE5}: \quad \mathbf{v}_i^{(g+1)} = \mathbf{x}_{r_1}^{(g)} + F\left(\mathbf{x}_{r_2}^{(g)} - \mathbf{x}_{r_3}^{(g)}\right) + F\left(\mathbf{x}_{r_4}^{(g)} - \mathbf{x}_{r_5}^{(g)}\right), \tag{17}$$

where $\mathbf{x}_{\text{best}}^{(g)}$ denotes the individual with the lowest objective value in the population at iteration g. The randomly selected indices $r_j \in \{1, 2, \ldots, N\} \setminus \{i\}, j = 1, 2, \ldots, 5$, are taken to be mutually different. The user-defined parameter $F \in [0, 2]$ is called the *scale factor* and defines the size of the steps taken towards the search directions defined by the differences between existing individuals.

After generating the mutant vectors, crossover takes place. A *trial vector*, $\mathbf{u}_i = (u_{i1}, u_{i2}, \ldots, u_{in})^\top$, is generated for each individual $\mathbf{x}_i^{(g)}$, as follows:

$$u_{ij}^{(g+1)} = \begin{cases} v_{ij}^{(g+1)}, & \text{if } R_j \leqslant CR \text{ or } j = RI(i), \\ x_{ij}^{(g)}, & \text{otherwise,} \end{cases} \tag{18}$$

where $j = 1, 2, \ldots, n$; R_j is the j-th evaluation of a uniform random number generator in the range $[0, 1]$; $CR \in [0, 1]$ is a user-defined *crossover rate*; and $RI(i)$ is a randomly selected index in $\{1, 2, \ldots, n\}$.

Eventually, the selection operator is applied, where the trial vectors compete against their original individuals. If the trial vector achieved a better objective value, it replaces the original individual in the population, as follows:

$$\mathbf{x}_i^{(g+1)} = \begin{cases} \mathbf{u}_i^{(g+1)}, & \text{if } f\left(\mathbf{u}_i^{(g+1)}\right) < f\left(\mathbf{x}_i^{(g)}\right), \\ \mathbf{x}_i^{(g)}, & \text{otherwise.} \end{cases} \tag{19}$$

The parameters F, CR, and N must be carefully selected due to their impact on DE's performance. A comprehensive presentation of DE-related research can be found in Price et al. (2005).

3.2 Enhanced Differential Evolution

Recently, Mohamed (2014) proposed an enhanced DE (eDE) variant. It defines an alternative mutation scheme, while crossover is based on probabilistic selection between the new and the DE2 scheme of Eq. (14). Moreover, the algorithm is enhanced by using restart to alleviate local minima.

Putting it formally, eDE is based on the mutation scheme,

$$\mathbf{w}_i^{(g+1)} = \mathbf{x}_{r_1}^{(g)} + F_1\left(\mathbf{x}_{\text{best}}^{(g)} - \mathbf{x}_{r_1}^{(g)}\right) + F_2\left(\mathbf{x}_{r_1}^{(g)} - \mathbf{x}_{\text{worst}}^{(g)}\right), \tag{20}$$

where $\mathbf{x}_{r_1}^{(g)}$ is a randomly selected individual; $F_1, F_2 \in [0, 2]$ are scalar parameters called the *differential weights*; and $\mathbf{x}_{\text{best}}^{(g)}$, $\mathbf{x}_{\text{worst}}^{(g)}$, denote the best and worst individuals at iteration g, respectively. The trial vector is given as follows:

$$u_{ij}^{(g+1)} = \begin{cases} w_{ij}^{(g+1)}, & \text{if } \left(R_j \leqslant CR \text{ or } j = RI(i)\right) \text{ and } R \geqslant \left(1 - \frac{g}{g_{\max}}\right), \\ v_{ij}^{(g+1)}, & \text{if } \left(R_j \leqslant CR \text{ or } j = RI(i)\right) \text{ and } R < \left(1 - \frac{g}{g_{\max}}\right), \\ x_{ij}^{(g)}, & \text{otherwise,} \end{cases} \tag{21}$$

where $g_{\max}$ is the total number of iterations and R is a uniform random number generator in the range [0,1]. The rest of the parameters are identical to the standard DE. Also, note that v_{ij} is the j-th component of the mutation vector $\mathbf{v}_i$ produced through Eq. (14).

A restart mechanism is also incorporated in eDE to avoid premature convergence. The restart mechanism is applied on each individual except for the best one, which is kept unaltered. In our case, we adopt restarts from mild perturbations $\mathbf{x}_i'$ of current individuals $\mathbf{x}_i$, as follows:

$$x_{ij}' = x_{ij} \pm 1. \tag{22}$$

According to this scheme, the probability of plunging into local minima is drastically decreased and the local search capability is enhanced through the perturbation of individuals into their imminent neighborhood. The sign "+" or "−" in Eq. (22) is randomly selected with equal probability for each j. The bias is selected equal to 1 since it constitutes the smallest step size in integer search spaces as the ones in the proposed model.

3.3 Particle Swarm Optimization

Particle swarm optimization (PSO) was introduced by Kennedy and Eberhart (1995). Similarly to DE it is a population-based algorithm, although with special emphasis in cooperation. PSO does not have any direct selection operator. Instead, a population (called a *swarm*) of candidate solutions (called the *particles*) probes the search space. Each particle retains in memory the best position it has ever visited. This position, along with information shared with the rest of the swarm, is used to bias the move of the particles.

Let $S = \{\mathbf{x}_1, \mathbf{x}_2, \ldots, \mathbf{x}_N\}$ be a swarm of N particles, each one being an n-dimensional vector in the search space, $\mathbf{x}_i = (x_{i1}, x_{i2}, \ldots, x_{in}) \in X$, $i = 1, 2, \ldots, N$. The particle moves by adding to its current position an adaptable bias vector, called the *velocity*,

$$\mathbf{v}_i = (v_{i1}, v_{i2}, \ldots, v_{in})^\top ,$$

while it has also a personal memory, called the *best position*,

$$\mathbf{p}_i = (p_{i1}, p_{i2}, \ldots, p_{in})^\top \in X.$$

Apart from its own best position, each particle assumes a *neighborhood* in the form of a set of particle indices. The particle exchanges information with other particles from its neighborhood by adopting their best findings. In the case where the particle's neighborhood is the entire swarm, the best position in the neighborhood is referred as *global best* particle, and the resulting algorithm is denoted as *gbest* PSO. On the other hand, when smaller neighborhoods are used the algorithm is denoted as *local best* PSO (*lbest* PSO) (Parsopoulos and Vrahatis, 2010). The size of the neighborhood has crucial impact on the dissemination of information in the swarm and, hence, on convergence speed.

In literature, various neighborhood topologies have been proposed. Most commonly used are the star, Von Neumann, and the ring topology. According to the ring, all particles are assumed to lie on a ring with respect to their indices. Then, for a given particle, its neighborhood is defined by its immediate neighbors in the ring. The considered number of neighbors per neighborhood is called the neighborhood's *radius*. Thus, a ring neighborhood of radius r for a particular particle $\mathbf{x}_i$ is defined as follows:

$$NB_i^r = \{i - r, i - r + 1, \ldots, i, \ldots, i + r - 1, i + r\} . \quad (23)$$

Let best_i be the index of the best position found so far by any individual in the neighborhood NB_i^r of $\mathbf{x}_i$, i.e.,

$$\text{best}_i = \arg\min_{j \in NB_i^r} f\left(\mathbf{p}_j\right). \tag{24}$$

Also, let g denote the iteration counter. Then, the swarm is updated according to,

$$v_{ij}^{(g+1)} = \chi \left[v_{ij}^{(g)} + c_1 R_1 \left(p_{ij}^{(g)} - x_{ij}^{(g)} \right) + c_2 R_2 \left(p_{\text{best}_i j}^{(g)} - x_{ij}^{(g)} \right) \right], \tag{25}$$

$$x_{ij}^{(g+1)} = x_{ij}^{(g)} + v_{ij}^{(g+1)}, \tag{26}$$

where χ is the *constriction coefficient* parameter; c_1, c_2, are positive acceleration parameters; and R_1, R_2, are random variables, uniformly distributed in the range $[0, 1]$. The constriction coefficient promotes convergence by reducing the magnitude of the velocities. For the reader's convenience, we mention the typical values $\chi = 0.729$, $c_1 = c_2 = 2.05$, which are widely accepted as the default parameter set on the basis of the PSO's stability analysis due to Clerc and Kennedy (2002).

The best positions of the particles are updated at each iteration according to,

$$\mathbf{p}_i^{(g+1)} = \begin{cases} \mathbf{x}_i^{(g+1)}, & \text{if } f\left(\mathbf{x}_i^{(g+1)}\right) < f\left(\mathbf{p}_i^{(g)}\right), \\ \mathbf{p}_i^{(g)}, & \text{otherwise.} \end{cases} \tag{27}$$

A compendium of PSO-related research can be found in Parsopoulos and Vrahatis (2010).

3.4 Algorithm Portfolios

The term *Algorithm Portfolio* (AP) refers to a framework where different algorithms (*heterogeneous AP*) or different copies of the same algorithm (*homogeneous AP*) are combined in a single algorithmic scheme (Huberman et al., 1997). APs constitute a modern approach for solving challenging optimization problems. Their computational efficiency against common metaheuristics has led to a constantly increasing research production (Gomes and Selman, 1997; Huberman et al., 1997; Peng et al., 2010; Souravlias et al., 2014, 2015; Tang et al., 2014).

The performance of APs depends on the selection of appropriate constituent algorithms. The constituent algorithms can either interact or run independently. In the present work, the AP framework proposed by Souravlias et al. (2015) is used to define interactive algorithmic schemes consisting of the metaheuristics described in the previous sections. The AP's algorithms interact with each other and employ a typical *master-slave* parallelization model. Each metaheuristic runs on one of M

slave nodes for a pre-specified budget of total running time. The budget is divided into *execution time*, T_{exec}, i.e., time used by the algorithm for its own execution, and *investment time*, T_{inv}, i.e., time used to buy elite solutions from the other algorithms.

Slave nodes can communicate via a *master node*, which is responsible for two basic operations. Firstly, it maintains an archive of the M elite solutions detected by the algorithms. Secondly, it assigns *prices* to the elite solutions whenever solution trading takes place. For this purpose, the solutions are sorted in descending order with respect to their objective values. Then, the price of each one is determined according to its corresponding position ρ_i in the ranking, i.e.,

$$C_i = \frac{\rho_i \times BC}{M}, \tag{28}$$

where $BC = \beta\, T_{\text{inv}}$ is a fixed *base cost* and β is a constant that takes values in $[0, 1]$ as suggested by Souravlias et al. (2014).

When an algorithm (slave node) fails to improve its solution for some predefined amount of time, it requests from the master node to buy an elite solution from the archive. For the buyer algorithm, this comes at the cost of a fraction of its investment time, which is immediately credited to the seller algorithm that offered the purchased solution. The master node proposes to the slave node its archived elite solutions that are better than its own current best solution. Then, the buyer algorithm gets the one that maximizes the *Return on Investment* (ROI) index, among the solutions it can afford. The ROI index comes from trading theory and in our case is defined as follows:

$$ROI_j = \frac{f - f_j}{C_j}, \quad j \in \{1, 2, \ldots, M\}, \tag{29}$$

where f denotes the objective value of the buyer's best solution, f_j denotes the objective value of the seller's elite solution, and C_j is the assigned price (Souravlias et al., 2014). The paid investment time from the buyer algorithm is then added to the total execution time of the seller algorithm. The purchased solution replaces the worst solution in the population of the buyer. In case of no affordable solution, the buyer algorithm simply restarts, retaining only its best solution.

Apparently, better algorithms of the AP sell solutions more frequently and, consequently, gain additional execution time. It is worth mentioning that the total execution time assigned to the AP remains constant, since time portions are only dynamically transferred from inferior to the most promising constituent algorithms. This is a significant property in modern high-performance platforms where usage and execution time have a significant cost. Also, the final distribution of execution time of the constituent algorithms offers useful insight regarding the best-performing one for the problem at hand (Souravlias et al., 2014, 2015).

3.5 Further Applicability Issues

Two main issues need to be addressed prior to the application of the presented metaheuristics on the problem of Sect. 2. The first one is related to the discrete nature of the search space, while the second one refers to constraint handling.

Regarding the first issue, simple rounding to the nearest integer is used. Specifically, the algorithms are applied on the corresponding real search space and, for the function evaluation, the vectors are rounded to the nearest integer ones. In DE and eDE, the rounded vectors are also retained in the population. In PSO, rounded vectors replace best positions solely. Rounding is a common approach successfully applied in similar problems (Piperagkas et al., 2012; Parsopoulos et al., 2015).

The constraint handling problem is tackled with the widely used *penalty function* approach, combined with a set of preference rules between feasible and infeasible solutions:

1. Between two infeasible solutions, the one that violates fewer constraints is selected.
2. Between a feasible and an infeasible solution, the feasible one is preferred.
3. Between two feasible solutions, the one with the lowest objective value is preferred.

These rules have been previously used with PSO and DE (Parsopoulos et al., 2015). The employed penalty function has a simple form,

$$P(\mathbf{x}) = f(\mathbf{x}) + \sum_{i \in VC(\mathbf{x})} |V(i)| \,, \tag{30}$$

where $f(\mathbf{x})$ is the actual objective value of $\mathbf{x}$; $V(i)$ is the violation magnitude of the i-th constraint; and $VC(\mathbf{x})$ is the set of constraints violated by $\mathbf{x}$. Note that the penalty for a violated constraint depends on the magnitude of violation. Apparently, in absence of violated constraints the penalty function is equal to the original objective function.

4 Performance Assessment

In this section we expose the experimental settings and the obtained results from the application of the described algorithms on the test suite produced for the proposed HL model.

Table 2 Capacity and volume information for vehicle types I (small) and II (big)

	Transportation mode			
	Ground		Air	
	I	II	I	II
Load capacity (ton)	3	10	4	9
Load volume (m^3)	20	44	35	75

Table 3 Commodities information

	Water	Medicines	Food
Importance weight	0.35	0.35	0.30
Unit weight (kg)	650	20	200
Unit volume (m^3)	1.44	0.125	0.60

Table 4 Number of vehicles per DC

	Transportation mode			
	Ground		Air	
	I	II	I	II
DC_1	4	5	1	1
DC_2	4	5	1	1

4.1 Experimental Setting

The main goal in our model is the minimization of losses caused by the mismatch between supply and demand, as well as the determination of the optimal number of vehicles for the transportation of relief resources to the stricken areas. In our experiments we considered three life-essential commodities, namely *water*, *medicines*, and *food*. Among them, the first two were assumed to have slightly higher importance weights than the third one.

Moreover, we assumed the existence of two DCs responsible to supply two AAs, and two modes of transportation, ground and aerial, using trucks and helicopters, respectively. For each transportation mode, two vehicle types were considered, namely small and big vehicles, henceforth denoted as type I and II, respectively. Tables 2 and 3 report all relevant information regarding vehicles and commodities, respectively. Note that, motivated by the Kefalonia island earthquake in 2014, the reported data are based on real-world values (e.g., palettes of water bottles, typical transportation boxes for medication, etc). Also, Table 4 reports the number of available vehicles per DC.

In the context of the proposed model, a test suite of 10 benchmark problems with diverse characteristics was generated. The test problems are henceforth denoted as P1-P10. The problems were initially solved to optimality with the CPLEX solver. Subsequently, extensive experiments were conducted with the following algorithms: PSO, DE, eDE, as well as APs consisting of PSO+DE, PSO+eDE, DE+DE, DE+eDE, eDE+eDE, and PSO+DE+eDE.

In a preprocessing phase, all five mutation operators of Eqs. (13)–(17) were considered for DE and eDE, along with all combinations of their parameters $F \in [0, 2]$ and $CR \in [0, 1]$, discretized with step size 0.05. Various population sizes were also investigated. Preliminary experiments provided clear evidence that DE2 with,

$$F = F_1 = F_2 = 0.4, \qquad CR = 0.05,$$

was the most promising setting. The PSO algorithm was considered in its lbest model with ring topology of radius $r = 1$, and the default parameter set,

$$\chi = 0.729, \qquad c_1 = c_2 = 2.05.$$

The population size for all algorithms was set to $N = 150$, which was identified as a promising value for our 144-dimensional optimization problem. The boundaries for the decision variables were the ones imposed by the given data (for the vehicles) and the constraints given in Sect. 2 (for the delivered quantities).

In order to statistically validate each algorithm, 30 independent experiments were performed per problem instance. The experiments were conducted on Intel i7 servers with 8GB RAM. The running time for each experiment was set to 10 min in order to be comparable with the time needed by CPLEX to provide accurate lower bounds for the solutions. Note that, on average CPLEX required around 10 min to find good solution approximations, although their optimality guarantee required even hours. The algorithms were run and analyzed also for 5 and 20 min in order to investigate their sensitivity with respect to running time. The time-variation of the solution error is illustrated in Fig. 1.

For each algorithm and experiment, the best solution $\mathbf{x}^*_{\text{alg}}$ and its value f^*_{alg} were recorded, along with the *solution error* from the global minimum detected by CPLEX, i.e.,

$$\textit{solution error} = f^*_{\text{alg}} - f^*_{\text{cplex}}.$$

The plain solution error is utilized instead of the relative error, because the optimal objective value of some problems was zero. Average values of solution error over the 30 experiments, along with standard deviation, minimum, and maximum values, were also recorded for performance comparison purposes.

4.2 Results and Discussion

A summary of all recorded results is reported in Table 5, where the best-performing approach is boldfaced. Also, the results are graphically illustrated to facilitate visual comparisons. The average solution error from the global minimum is presented in

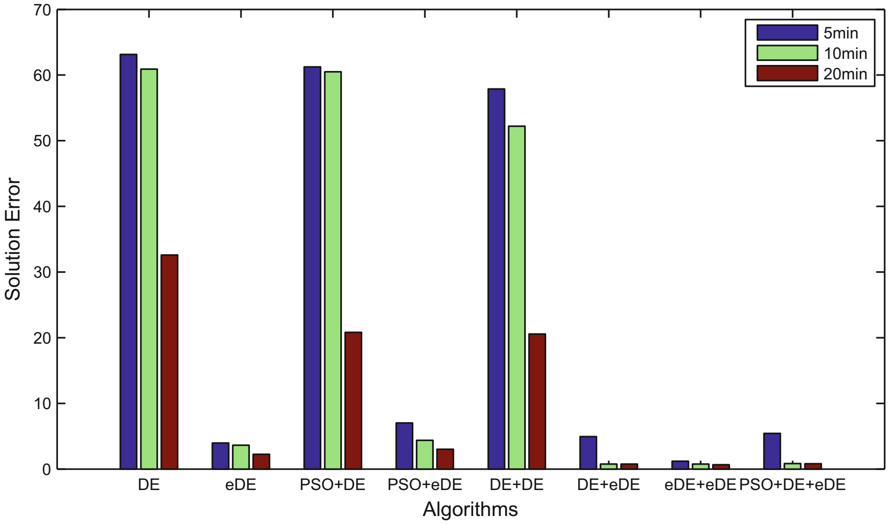

Fig. 1 Solution error averaged over all problem instances per algorithm and running time

Table 5 Mean, standard deviation, minimum, and maximum solution error values for all algorithms, averaged over all problems. Best values are boldfaced. The "+" symbol denotes AP approach constituting of the corresponding algorithms

Algorithm	Mean	St.D.	Min	Max
PSO	513.80	235.85	197.00	2442.20
DE	63.31	40.45	26.97	160.21
eDE	3.54	3.42	0.29	11.80
PSO+DE	52.28	31.11	27.01	129.42
PSO+eDE	4.14	3.99	0.16	13.77
DE+DE	59.65	55.36	21.15	193.81
DE+eDE	0.76	0.91	0.00	2.91
eDE+eDE	**0.75**	**0.85**	**0.00**	**2.27**
PSO+DE+eDE	0.84	1.18	0.00	3.74

the upper part of Fig. 2, per problem and algorithm. In the lower part of Fig. 2, the central region around the origin is zoomed, exposing the corresponding curves of the most competitive algorithms. Similarly, in the upper and lower part of Fig. 3, we illustrate the averaged standard deviation per problem and algorithm. Note that in all figures we excluded the results of PSO due to scaling reasons.

Furthermore, we also recorded the success rate per algorithm, i.e., the percentage of experiments where it succeeded to reach the optimal solution within the available execution time. Figure 4 presents the resulting success rates per problem instance

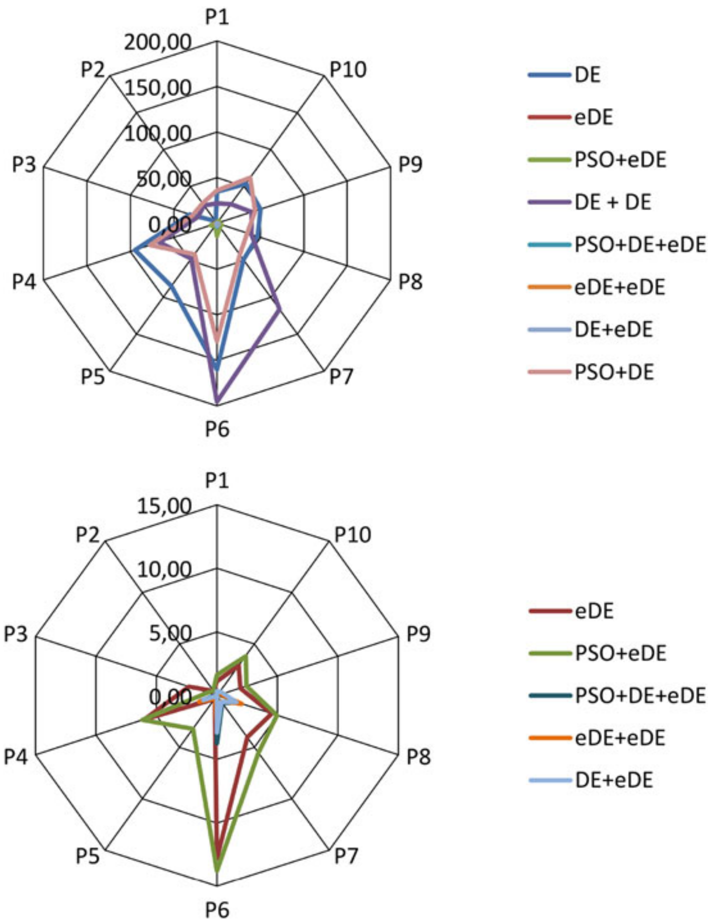

Fig. 2 Averaged solution error per algorithm and problem (*upper part*) and zoom in center area (*lower part*)

for the most promising algorithms. Finally, the boxplots of Fig. 5 illustrate the distribution of the obtained solution error values in all experiments.

The reported results offer interesting conclusions. Firstly, we can easily see that the homogeneous AP approach eDE+eDE as well as the heterogeneous PSO+DE+eDE outperformed the rest of the algorithms, yielding higher success rates. Also, these two approaches exhibited almost equivalent performance. However, in problems P6-P8, which were proved to be the most difficult ones with respect to the success rates of the algorithms, the eDE+eDE approach dominated in terms of efficiency.

In order to quantitatively study this behavior, we further analyzed the solution purchases between the algorithms of the AP approaches. The analysis verified that, especially for the aforementioned problems, the number of purchases between the

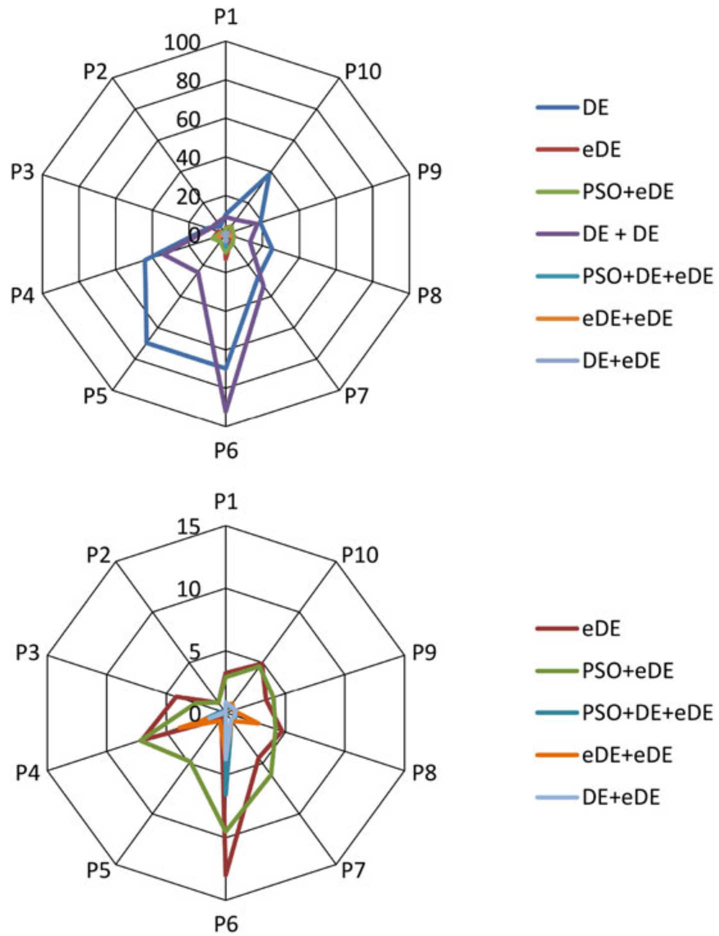

Fig. 3 Standard deviation of the solution error per algorithm and problem (*upper part*) and zoom in center area (*lower part*)

algorithms was remarkably high. This leads to the conclusion that, due to the complexity of these problems, the constituent algorithms of the AP experienced severe difficulties in reaching the optimal solution. Therefore, they were more prone to exchange information in order to improve their performance.

Also, in the case of PSO+DE+eDE, the assigned execution time per algorithm was shorter than that of each eDE instance in eDE+eDE, because in the first case the total time of the AP is divided by 3, while in the latter one it is divided in 2 equal parts. Since PSO was proved to be less efficient than eDE, the assigned time in PSO+DE+eDE was consequently proved to be insufficient.

Regarding the standalone algorithms, eDE was clearly the dominant one, exhibiting undoubtful advantages against the rest. This can also explain the superiority of the eDE-based AP approaches. Obviously, the special probabilistic operator of eDE as well as the restart mechanism with mild perturbations (see Sect. 3.2)

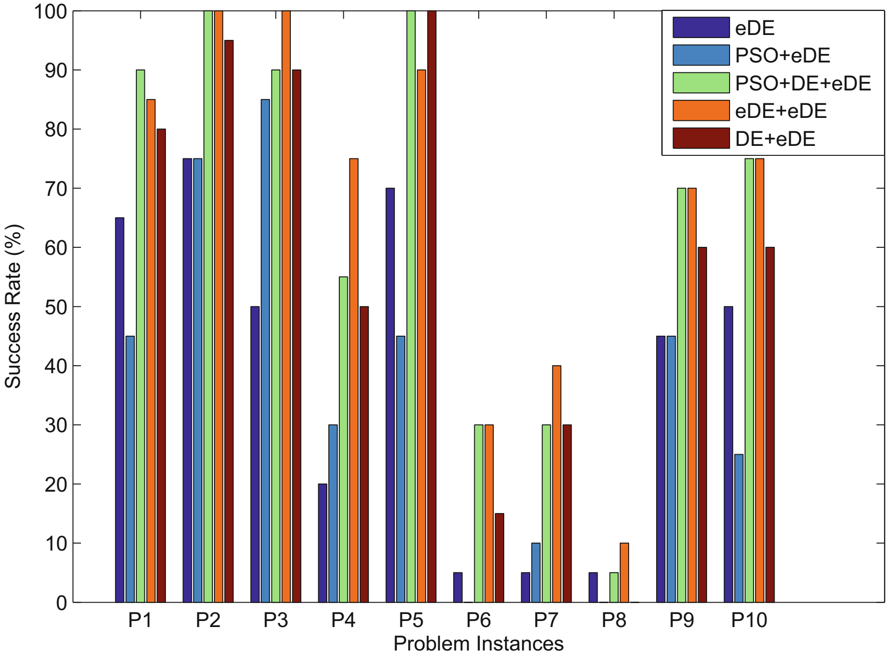

Fig. 4 Success rates of the most promising algorithms per problem

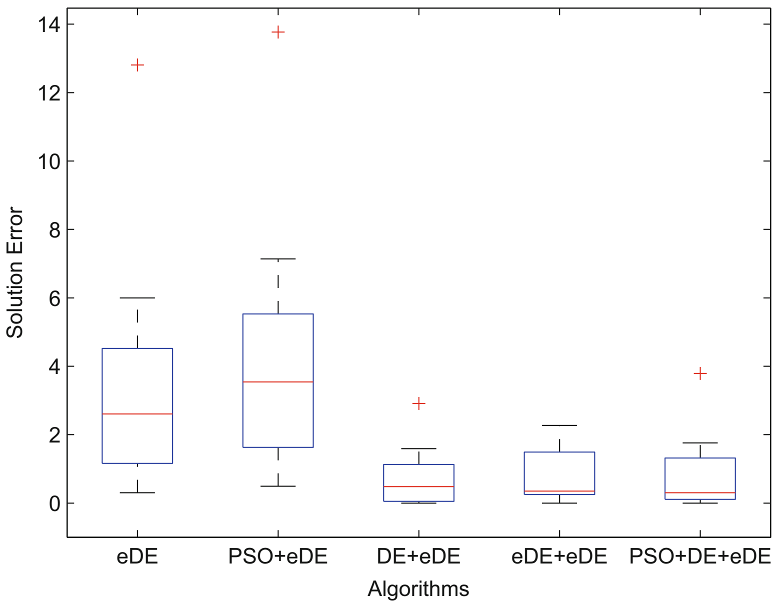

Fig. 5 Solution error distribution of the most promising algorithms for all test problems

were beneficial for the algorithm. Experimental evidence suggested that this can be attributed to the alleviation of search stagnation caused by the rounding of the real-valued vectors to their nearest integers. Moreover, this can be related also to the domination of the DE2 operator, which offers the necessary diversity to avoid stagnation. These properties were also identified in Souravlias et al. (2014).

Although there is a clear advantage of some algorithms against the rest, there are marginal differences among the most promising approaches. In order to investigate whether the observed differences were the outcome of random fluctuations, we conducted statistical significance tests among the most competitive algorithms. Specifically, pairwise comparisons of the algorithms were conducted using the Wilcoxon ranksum tests at 95 % confidence level for all test problems. Whenever an algorithm was statistically superior to another, we counted it as *win* of the algorithm. On the other hand, if it was statistically inferior, we counted it as a *loss*. The lack of statistical significance was counted as a *draw* for both algorithms.

The results concerning wins/losses/draws are presented in Table 6 and the corresponding graphical illustration is given in Fig. 6 for all problem instances. The superiority of DE+eDE, eDE+eDE, and PSO+DE+eDE was anew confirmed. In almost all comparisons, these approaches were prevalent against the rest. Yet, most of the comparisons among them resulted in draws, despite the marginal differences reported in Table 6. Especially for DE+eDE and eDE+eDE, no losses were reported. Thus, our initial assumption regarding the superiority of eDE-based approaches was corroborated by the statistical evidence, placing these AP approaches in a salient position among the most promising solvers.

5 Conclusions

The contribution of the present work was twofold. On one hand, we introduced a model that aims at minimizing the losses caused by the mismatch between supply and demand, while concurrently determining the number of different types of vehicles used to transport relief commodities from dispatch centers to stricken areas. A number of test problems with diverse characteristics was generated for the proposed model and solved to optimality using CPLEX.

Table 6 Wins/losses/draws of row versus column algorithms for all problem instances

	eDE	PSO+eDE	DE+eDE	eDE+eDE	PSO+DE+eDE
eDE	–	1 / 1 / 8	0 / 5 / 5	0 / 5 / 5	0 / 8 / 2
PSO+eDE		–	0 / 7 / 3	0 / 9 / 1	1 / 7 / 2
DE+eDE			–	0 / 0 / 10	0 / 0 / 10
eDE+eDE				–	0 / 0 / 10
PSO+DE+eDE					–

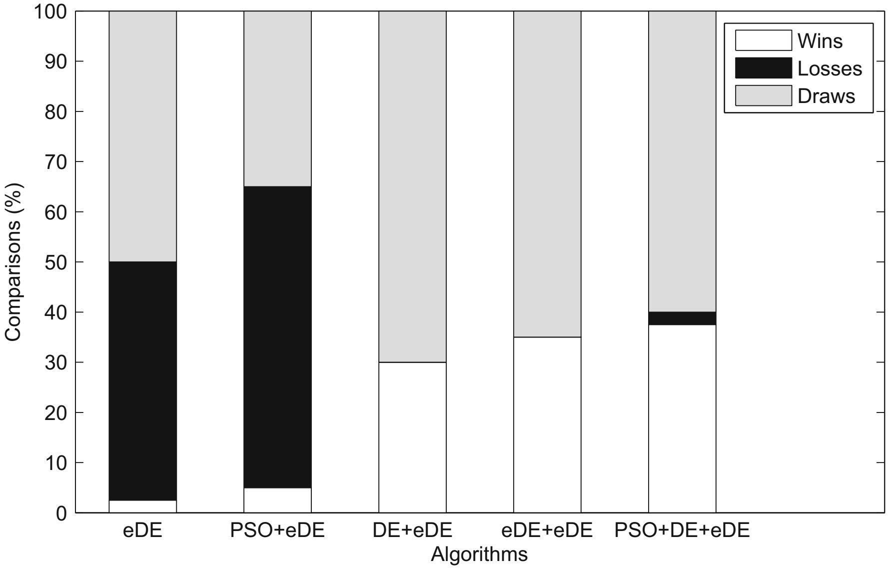

Fig. 6 Results of the pairwise statistical comparisons among the most competitive algorithms for all test problems

On the other hand, prevalent modern metaheuristics were studied in solving Humanitarian Logistics problems. Our approach was based on DE, eDE, PSO, and heterogeneous/homogeneous APs consisting of combinations of these algorithms. Proper modifications and refinements were introduced to tackle the special requirements of the test problems.

From the extracted results, we concluded that APs based on eDE offer remarkable performance efficiency and solution quality. Also, it became evident that APs can offer crucial insight in gathering information regarding the most appropriate metaheuristic for the problem at hand.

Future work will extend the test suite, aiming at an abundant set of test problems with a multitude of different characteristics and peculiarities. Also, the study of APs will be enriched by employing larger and diverse collections of metaheuristics, in order to efficiently deal with problems of higher complexity.

References

Balcik, B., Beamon, B.M.: Facility location in humanitarian relief. Int. J. Log. Res. Appl. **11**(2), 101–121 (2008)

Balcik, B., Beamon, B.M., Smilowitz, K.: Last mile distribution in humanitarian relief. J. Intell. Transp. Syst. **12**(2), 51–63 (2008)

Barbarosoglu, G., Arda, Y.: A two-stage stochastic programming framework for transportation planning in disaster response. J. Oper. Res. Soc. **55**(1), 43–53 (2004)

Besiou, M., Stapleton, O., Van Wassenhove, L.N.: System dynamics for humanitarian operations. J. Humanitarian Logist. Supply Chain Manag. **1**(1), 78–103 (2011)

Blum, C., Roli, A.: Metaheuristics in combinatorial optimization: Overview and conceptual comparison. ACM Comput. Surv. **35**(3), 268–308 (2003)

Chang, M.S., Tseng, Y.L., Chen, J.W.: A scenario planning approach for the flood emergency logistics preparation problem under uncertainty. Transp. Res. E Logist. Transp. Rev. **43**(6), 737–754 (2007)

Clark, A., Culkin, B.: A network transshipment model for planning humanitarian relief operations after a natural disaster. In: Decision Aid Models for Disaster Management and Emergencies, Atlantis Computational Intelligence Systems, vol. 7, pp. 233–257. Atlantis Press, Paris (2013)

Clerc, M., Kennedy, J.: The particle swarm-explosion, stability, and convergence in a multidimensional complex space. IEEE Trans. Evol. Comput. **6**(1), 58–73 (2002)

Cozzolino, A., Rossi, S., Conforti, A.: Agile and lean principles in the humanitarian supply chain. J. Humanitarian Logist. Supply Chain Manag. **2**(1), 16–33 (2012)

Diaz, R., Behr, J., Toba, A.L., Giles, B., Ng, M., Longo, F., Nicoletti, L.: Humanitarian/emergency logistics models: A state of the art overview. In: Proceedings of the 2013 Summer Computer Simulation Conference (SCSC13), pp. 24:1–24:8. Society for Modeling & Simulation International, Vista (2013)

Falasca, M., Zobel, C.W.: A two-stage procurement model for humanitarian relief supply chains. J. Humanitarian Logist. Supply Chain Manag. **1**(2), 151–169 (2011)

Galindo, G., Batta, R.: Review of recent developments in OR/MS research in disaster operations management. Eur. J. Oper. Res. **230**(2), 201–211 (2013)

Glover, F.: Future paths for integer programming and links to artificial intelligence. Comput. Oper. Res. **13**(5), 533–549 (1986)

Gomes, C.P., Selman, B.: Algorithm portfolio design: Theory vs. practice. In: Proceedings of the 13th Conference on Uncertainty in Artificial Intelligence (UAI'97), pp. 190–197. Morgan Kaufmann Publishers, San Francisco (1997)

Gomes, C.P., Selman, B.: Algorithm portfolios. Artif. Intell. **126**(1-2), 43–62 (2001)

Han, Y., Guan, X., Shi, L.: Optimization based method for supply location selection and routing in large-scale emergency material delivery. IEEE Trans. Autom. Sci. Eng. **8**(4), 683–693 (2011)

Huang, M., Smilowitz, K., Balcik, B.: A continuous approximation approach for assessment routing in disaster relief. Transp. Res. B Methodol. **50**, 20–41 (2013)

Huberman, B.A., Lukose, R.M., Hogg, T.: An economics approach to hard computational problems. Science **27**, 51–53 (1997)

Kennedy, J., Eberhart, R.: Particle swarm optimization. In: Proceedings on IEEE International Conference on Neural Networks, vol. 4, pp. 1942–1948 (1995)

Liu, N., Ye, Y.: Humanitarian logistics planning for natural disaster response with Bayesian information updates. J. Ind. Manag. Optim. **10**(3), 665–689 (2014)

Lourenço, H.: Logistics management: An opportunity for metaheuristics. In: Metaheuristics Optimization via Memory and Evolution, vol. 30, pp. 329–356. Springer, New York (2005)

Mohamed, A.W.: RDEL: Restart differential evolution algorithm with local search mutation for global numerical optimization. Egypt. Inform. J. **15**(3), 175–188 (2014)

Özdamar, L., Ekinci, E., Kucukyazici, B.: Emergency logistics planning in natural disasters. Ann. Oper. Res. **129**(1-4), 217–245 (2004)

PAHO: Humanitarian Supply Management and Logistics in the Health Sector. Pan American Health Organization, Washington D.C. (2001)

Parsopoulos, K.E., Konstantaras, I., Skouri, K.: Metaheuristic optimization for the single-item dynamic lot sizing problem with returns and remanufacturing. Comput. Ind. Eng. **83**, 307–315 (2015)

Parsopoulos, K.E., Vrahatis, M.N.: Particle Swarm Optimization and Intelligence: Advances and Applications. Information Science Publishing (IGI Global), Hershey (2010)

Peng, F., Tang, K., Chen, G., Yao, X.: Population-based algorithm portfolios for numerical optimization. IEEE Trans. Evol. Comput. **14**(5), 782–800 (2010)

Peng, M., Chen, H.: System dynamics analysis for the impact of dynamic transport and information delay to disaster relief supplies. In: 2011 International Conference on Management Science and Engineering (ICMSE 2011), pp. 93–98 (2011)

Piperagkas, G.S., Konstantaras, I., Skouri, K., Parsopoulos, K.E.: Solving the stochastic dynamic lot-sizing problem through nature-inspired heuristics. Comput. Oper. Res. **39**(7), 1555–1565 (2012)

Price, K., Storn, R.M., Lampinen, J.A.: Differential Evolution: A Practical Approach to Global Optimization. Springer, New York (2005)

Sheu, J., Chen, Y., Lan, L.: A novel model for quick response to disaster relief distribution. In: Proceedings of the Eastern Asia Society for Transportation Studies, vol. 5, pp. 2454–2462 (2005)

Souravlias, D., Parsopoulos, K.E., Alba, E.: Parallel algorithm portfolio with market trading-based time allocation. In: International Conference on Operations Research 2014 (OR2014). Aachen, Germany (2014)

Souravlias, D., Parsopoulos, K.E., Kotsireas, I.S.: Circulant weighing matrices: A demanding challenge for parallel optimization metaheuristics. Optim. Lett. **10**(6), 1303–1314 (2015)

Storn, R., Price, K.: Differential evolution - a simple and efficient heuristic for global optimization over continuous spaces. J. Global Optim. **11**(4), 341–359 (1997)

Tang, K., Peng, F., Chen, G., Yao, X.: Population-based algorithm portfolios with automated constituent algorithms selection. Inform. Sci. **279**, 94–104 (2014)

Tatham, P., Kovacs, G.: The application of swift trust to humanitarian logistics. Int. J. Prod. Econ. **126**(1), 35–45 (2010)

Taylor, D., Pettit, S.: A consideration of the relevance of lean supply chain concepts for humanitarian aid provision. Int. J. Serv. Technol. Manag. **12**(4), 430–444 (2009)

Thomas, A., Kopczak, L.: From Logistics to Supply Chain Management - The Path Forward to the Humanitarian Sector. Fritz Institute, San Francisco (2005)

Van Hentenryck, P., Bent, R., Coffrin, C.: Strategic planning for disaster recovery with stochastic last mile distribution. In: Integration of AI and OR Techniques in Constraint Programming for Combinatorial Optimization Problems, Lecture Notes in Computer Science, vol. 6140, pp. 318–333. Springer, Berlin-Heidelberg (2010)

Van Wassenhove, L.N.: Humanitarian aid logistics: Supply chain management in high gear. J. Oper. Res. Soc. **57**(5), 475–489 (2006)

Vitoriano, B., Ortuño, M., Tirado, G., Montero, J.: A multi-criteria optimization model for humanitarian aid distribution. J. Glob. Optim. **51**(2), 189–208 (2011)

Yan, S., Shih, Y.L.: An ant colony system-based hybrid algorithm for an emergency roadway repair time-space network flow problem. Transportmetrica **8**(5), 361–386 (2012)

Yi, W., Kumar, A.: Ant colony optimization for disaster relief operations. Transp. Res. E Logist. Transp. Rev. **43**(6), 660–672 (2007)

Yi, W., Özdamar, L.: A dynamic logistics coordination model for evacuation and support in disaster response activities. Eur. J. Oper. Res. **179**(3), 1177–1193 (2007)

Yuan, Y., Wang, D.: Path selection model and algorithm for emergency logistics management. Comput. Ind. Eng. **56**(3), 1081–1094 (2009)

Zhang, J.H., Li, J., Liu, Z.P.: Multiple-resource and multiple-depot emergency response problem considering secondary disasters. Expert Syst. Appl. **39**(12), 11,066–11,071 (2012)

Zheng, Y.J., Ling, H.F., Xue, J.Y., Chen, S.Y.: Population classification in fire evacuation: A multiobjective particle swarm optimization approach. IEEE Trans. Evol. Comput. **18**(1), 70–81 (2014)