Optimizing Multi-objective Knapsack Problem using a Hybrid Ant Colony Approach within Multi Directional Framework

Imen Ben Mansour[1,2]
[1]*ESPRIT School of Engineering, Tunis, Tunisia*
[2]*ENSI-LARIA, University of Manouba, Tunisia*

Keywords: Knapsack Problem, Multi-objective Optimization, Ant Colony Optimization, Local Search Method, Multi-directional Framework.

Abstract: Balancing the convergence and diversity simultaneously is very challenging for multi-objective evolutionary algorithms on solving multi-objective optimization problems (MOPs). The proposed approach MD-HACO coupled an Ant Colony Optimization (ACO) algorithm with a multi-objective local search procedure, and evolves it into a multi-directional framework. The idea is to optimize the overall quality of Pareto set approximation by using different configurations of the hybrid approach by means of different directional vectors. During the optimization process, the artificial ants work in different search directions in the objective space trying to approximate small parts of the Pareto front. Afterward, a local search procedure is applied to each sub-region to enhance the search process toward the extreme Pareto-optimal solutions with respect to the weight vector under consideration. A multi-directional set holding the non-dominated solutions according to all directional archives is maintained. The proposed approach is tested on widely used multi-objective multi-dimensional knapsack problem (MOMKP) instances and compared with well-known state-of-the-art algorithms. Experiments highlight that the use of a multi-directional paradigm as well as a hybrid schema can lead to interesting results on the MOMKP and ensure a good balance between convergence and diversity.

SCIENCE AND TECHNOLOGY PUBLICATIONS

1 INTRODUCTION

Optimizing multi-objective problems is considered as a hard task since multiple objectives should to be optimized simultaneously while satisfying a several constraints. Indeed, the aim is to find the optimal trade-offs between the different objectives that are usually conflicting. To this end, researchers have proposed many metaheuristic algorithms in order to achieve the set of compromise solutions which is called Pareto front in a reasonable time.

In this paper, we address the multi-objective multi-dimensional knapsack problem (MOMKP) that consists to find a subset of items subject to a set of resource constraints while maximizing several objectives. Due to its NP-hard nature (Martello, 1990), and wide-range applicability (Chabane et al., 2017), (Ehrgott and Ryan, 2002), (Kellerer et al., 2004) MOMKP has been considered among the most intriguing multi-objective optimization problems, and has attracted considerable attention from the operations research community. As a result, a significant number of works have investigated the MOMKP, giving rise to various solution methods (Lust and Teghem, 2012).

Considering multi-objective metaheuristics, the main purpose behind designing a approximate approach, is balancing exploration and exploitation. The population-based methods such as ant colony optimization (ACO) are powerful techniques in the exploration of the solution space but less efficient in the exploitation of the search toward promising regions. However, local search approaches have perfect plans for intensification, but have a tendency of getting stuck in local optima due to lack of diversification. Therefore, to intensify the search and escape from being trapped into local optima, a local enhancement will be coupled with an ACO approach. Within the scope of this paper, a hybrid approach is proposed as a synergy of the ACO algorithm with a Tchebycheff-based Local Search (TLS) procedure coined as a Multi-Directional Hybrid Ant Colony Optimization approach (MD-HACO) to handle the knapsack problem within the multi-objective framework. The design of a hybrid approach is motivated by the need to enhance the convergence of the solutions to-

Ben Mansour, I.
Optimizing Multi-objective Knapsack Problem using a Hybrid Ant Colony Approach within Multi Directional Framework.
DOI: 10.5220/0010865600003116
In *Proceedings of the 14th International Conference on Agents and Artificial Intelligence (ICAART 2022) - Volume 2*, pages 409-418
ISBN: 978-989-758-547-0; ISSN: 2184-433X
Copyright © 2022 by SCITEPRESS – Science and Technology Publications, Lda. All rights reserved

ward the optimal Pareto front and to promote diversity of the search space.

The proposed approach uses an improved version of the Gw-ACO approach (Mansour et al., 2019). Indeed, this version differs from Gw-ACO mainly in the used Ant System variant and the solutions construction phase:

(i) Here, the proposed algorithm follows $\mathcal{MAX}-\mathcal{MIN}$ Ant System (Stützle and Hoos, 2000) scheme. Whereas Gw-ACO is based on an elitist version of Ant System (Dorigo et al., 1996).

(ii) Gw-ACO uses many heuristic matrices, it assigns to each objective a weighted heuristic information. In MD-HACO, all ants in the algorithm share one heuristic matrix in the process of constructing the solution and this matrix changes configuration during the optimization phase using weighted pseudo-utility ratio. The objectives were considered fairly, i.e., no objective has any precedence over the others and the evaluation of a candidate solution is rather based on its position in the objective space. Therefore, the Pareto optimal set may be found (Iredi et al., 2001).

The rest of this paper is organized as follows. In section two, we introduce briefly the context of multi-objective optimization. The studied problem MOMKP is formulated in section 3. Next, the multi-objective ACO (MOACO) approach is introduced. The hybrid approach is presented in section five. The experiments are detailed on a multi-objective multi-dimensional knapsack problem in the next section. The last section is devoted to conclusion and discussion about possible future research directions.

2 MULTI-OBJECTIVE OPTIMIZATION PROBLEMS

In multi-objective optimization problems, a set of objective functions must be maximized or minimized, subjected to a set of constraints. In the following, we assume that m objective functions $F = (f_1, \cdots, f_m)$ are to be maximized. Each objective function f_k is defined as $f_k : D_1 \times \cdots \times D_n \rightarrow \mathcal{R}$, where the D_i are the domains of the decision variables. An optimization problem is defined by an objective space $Z = \{z = (z_1, \cdots, z_m) | \forall 1 \le k \le m, z_k \in \mathcal{R}\}$ and a decision space $X = \{x = (x_1, \cdots, x_n) | \forall 1 \le i \le n, x_i \in D_i\}$. Each decision vector $x \in X$ is associated to an objective vector $z = (f_1(x), \cdots, f_m(x)) \in Z$. In order to compare solutions in a multi-objective context, the concept of Pareto dominance is used. The definition 1 formally defines the Pareto dominance relation in a maximization context without loss of generality.

Definition 2.1. *A decision vector $x \in X$ is said to dominate another decision vector $x' \in X$ (written as $x \succ x'$), if $\forall 1 \le k \le m, f_k(x) \ge f_k(x')$ and $\exists 1 \le j \le m, f_j(x) > f_j(x')$.*

Accordingly, a decision vector $x \in X$ is said to be Pareto optimal (resp. non-dominated) if and only if it is not dominated by any other solution of the search space (resp. within a set of solutions).

3 MULTI-OBJECTIVE MULTI-DIMENSIONAL KNAPSACK PROBLEM

The multi-objective multidimensional knapsack problem could be formulated as follows:

$$Maximize \sum_{j=1}^{n} p_j^k x_j \qquad k = 1, ..., m \tag{1}$$

$$Subject\,to \sum_{j=1}^{n} w_j^i x_j \le b_i \qquad i = 1, ..., q \tag{2}$$

$$x_j \in \{0, 1\} \qquad j = 1, ..., n$$

n denotes the number of items I_j, x_j corresponds to the decision variable for the item I_j i.e., the item is taken or not taken since we consider a 0/1 knapsack problem. Each item I_j has a profit p_j^k relatively to the objective k and a weight w_j^i relatively to the resource i and b_i is the total quantity available for the resource i.

4 MULTI-OBJECTIVE ANT COLONY OPTIMIZATION

ACO metaheuristic, initially developed by Dorigo (Dorigo et al., 1991b), is a cooperative population-based construction algorithm inspired from the behavior of real ants while searching for a food source. The colony of ants cooperates to perform some tasks for the whole group using an indirect form of communication, called pheromone, deposited by the member of the colony while building their solutions. From an initial point, each ant moves on the search space through a finite sequence of points. Its movements depend on a stochastic construction policy directed by two types of information: (i) the ancient ant movements recorded in his memory (ii) pheromone traces and heuristic information specific to the problem being treated. These informations are used by ants to help them to converge on the most interesting areas of the search space. The first example of such an algorithm was Ant System and introduced by Dorigo, Maniezzo and Colorni in (Dorigo et al., 1991b),(Dorigo et al., 1991a),(Dorigo et al., 1996) and Colorni, Dorigo and Maniezzo in (Colorni et al., 1991) and (Colorni et al., 1992) using Traveling Salesman Problem (TSP) as a sample application.

Due to the success of ACO for mono-objective problems (Dorigo and Blum, 2005) many authors investigated the capabilities of this metaheuristic to handle multi-objective problems. In (Gambardella et al., 1999), a multiple ant colony system is proposed to solve the vehicle routing problems with time windows (MACS-VRPTW). The algorithm is organized with a hierarchy of ant colonies designed to successively optimize a multi-objective function: the first colony minimizes the number of vehicles while the second optimizes the feasible solutions found by the first. Each colony uses an independent pheromone structure for its specific purpose and they collaborate by sharing the best overall solution found. In (Alaya et al., 2007), a generic multi-objectif ACO approach named m-ACO, is proposed and instantiated with four variants. This algorithm is parameterized by the number of ant colonies and the number of the pheromone structures. In (Mansour and Alaya, 2015), an ACO algorithm called IBACO was introduced to slove MOMKP. The IBACO algorithm employs binary indicators. It uses hypervolume and epsilon indicators to guide the artificial ants to find the best solutions by laying pheromone trails relatively to the indicator values. (Yagmahan and Yenisey, 2010) solved the flow shop scheduling problem with a multi-objective ant colony system algorithm combined to a local search strategy. proposed a MOACO for the mixed-model assembly line balancing problem to minimize the balance delay and the smoothness index for a given cycle time.

On the above approaches, the objectives were considered fairly where they foster search in specific directions in the objective space. Iredi, Merkle and Middendorf proposed using different weight vectors in order to force ants to search in different regions of the optimal Pareto set. They introduced the BicriterionAnt algorithm in (Iredi et al., 2001). Since the success of BicriterionAnt, several ant-based approaches following the multi-directional idea were developed. (Angus, 2007) solved the multi-objective TSP using population-based ACO algorithm with an average-rank-weight method to determine weightings for each objective. A MOACO approach based on decomposition method called MOEA/D-ACO is introduced in (Ke et al., 2013). In MOEA/D-ACO, each ant targets a particular point in the Pareto front by means of uniformly distributed weight vectors, each group of ants tries to approximate a particular part of the Pareto front. Recently, (Mansour et al., 2019) conceived an ant colony approach based on multiple search directions to optimize the MOMKP. The algorithm uses a generation-based weight assignment strategy to enhance population diversity.

5 MULTI-DIRECTIONAL HYBRID ANT COLONY OPTIMIZATION ALGORITHM FOR MOMKP

Since ant colony should be able to generate a set of diverse new solutions when solving large and hard optimization problems, the use of directional vectors seems to be an interesting paradigm. A possible option for multi-directional metaheuristics consists in using different combinations of weight vector during the optimization process. The directional model is used to guide the optimization on exploiting new and different parts of the objective space.

Our algorithm follows the $\mathcal{MAX}-\mathcal{MIN}$ Ant System (Stützle and Hoos, 2000) scheme. As a first step, the pheromone traces are initialized to an upper bound τ_{max}. Then, the Multi-directional optimization process is launched. The idea is to decompose the objective space into several sub-regions i.e search directions by means of different Λ vectors. Here, we use the method that has been proposed in (Mansour et al., 2018) and (Mansour et al., 2019). This method, called Gradual weights generation method (Gw), is interesting because it uses not only a different weight vector at each search step, but also gradually distributed weights in the objective space (progressively generated from 0 to 1 or/and from 1 to 0) more details can be found in (Mansour et al., 2019) (especially concerning the weight assignment process). Each sub-region has its own directional Pareto set A holding all non-dominated solutions of the current direction i.e. small part of the whole Pareto front. At each generation, the colony focuses in one single sub-region and constructs solutions according to the search direction under consideration. Therefore, it can compute a particular part of the Pareto front. Once the construction phase is done, all solutions are added to a local set Sol and the directional Pareto set A is update with the non-dominated solutions of Sol. Then, the Multi-directional iterated local search procedure is applied to each solution S^h in Sol (see Section 5.3). Once the directional optimization phase is carried out, a sharing step is performed. First, the pheromone structure is update as detailed in Section 5.2 and then the multi-directional Pareto set called P manages the update of the non-dominated solutions sent by the directional Pareto sets A. This stage constitutes the general information sharing between the different sub-regions, since the multi-directional Pareto set benefits from the respective potential contribution of each directional Pareto set.

The proposed Multi-directional Hybrid Ant Colony Optimization approach is described in algorithm 1. To have a clear understanding of the process of MD-HACO, Figure 1 provides an illustration of flowchart of the proposed approach.

Algorithm 1: Multi-directional Hybrid Ant Colony Optimization Algorithm.

Input: $Nbants$ (number of ants)
g_{max} (maximum number of generations)
$\lambda = (\lambda_1, \cdots, \lambda_m)$ (weight vector)

Output: **P** (multi-directional Pareto set P)

Initialize the traces of pheromone to τ_{max}
Initialize the multi-directional Pareto set $P \leftarrow \emptyset$
ForEach direction $\lambda(g), g \in \{1, \cdots, g_{max}\}$ do
Step 0: Initialize $A \leftarrow \emptyset$, $Sol \leftarrow \emptyset$
Step 1: Compute λ_k for each objective f_k
Step 2: Construct solution S^h, $h \in \{1, \cdots, Nbants\}$ and $Sol \leftarrow Sol \cup S^h$
Step 3: $A \leftarrow$ Non-dominated solutions of $A \cup Sol$
Step 4: Execute the Multi-directional iterated local search procedure for each S^h in Sol
EndForEach
Update pheromone structure
$\mathbf{P} \leftarrow$ Non-dominated solutions of $(\mathbf{P} \bigcup \mathbf{A})$. If g_{max} is reached then stop and return **P**; else performs another directional optimization step.

5.1 Solutions Construction

Each ant h constructs one feasible solution by applying repeatedly a stochastic function named a state transition rule. This function is used to select the most appropriate item I_j to be added to the solution S^h among a set of feasible items $Feas$ (candidate vertices). This set is updated by including items not yet added and that don't violate any constraint. The transition rule p_{S^h} is directed by the pheromone value τ_{S^h} and the heuristic information η_{S^h}:

$$p_{S^h}(I_j) = \frac{[\tau_{S^h}(I_j)]^{\alpha}.[\eta_{S^h}(I_j)]^{\beta}}{\sum\limits_{I_l \in Feas} [\tau_{S^h}(I_l)]^{\alpha}.[\eta_{S^h}(I_l)]^{\beta}} \tag{3}$$

The heuristic information is used to guide the search process of artificial ants. In order to orientate ants to look in different regions of the non-dominated front, different configurations of the heuristic information matrix are executed, i.e. using different weight vectors. For that, η_{S^h} for a given ant h, is set as:

$$\eta_{S^h}(I_j) = \frac{\sum_{k=1}^{m} \lambda_k(g) p_j^k}{\sum_{i=1}^{q} \left(\frac{w_j^i}{R_{S^h}(i)} \right)} \tag{4}$$

where $R_{S^h}(i) = b_i - \sum_{t \in S^h} r_{it}$ is the remaining amount of the resource i when an ant h is currently building its solution S^h. p_j^k and w_j^i are respectively the profit and the weight of the candidate item.

5.2 Pheromone Update

The pheromone matrix is defined as an aggregation of objectives. Once all ants have constructed their solutions and the set Sol is updated with all created solutions, the pheromone update phase will be needed:

$$\tau(I_j) \leftarrow (1-\rho) * \tau(I_j) + \Delta\tau(I_j) \tag{5}$$

To exploit the most promising parts, ants that are allowed to update the pheromone trails are only those that have found non-dominated solutions S_{ND} during the actual iteration. Furthermore, to give every generation of ants the same influence in a particular part of the Pareto front, ants that are authorized to update, lay the same amount of pheromone $\Delta\tau(I_j)$ equal to the current archive set size $|A|$.

5.3 Enhancement by Multi-directional Iterated Local Search

Local search aims to improve the quality of solutions generated by the Ant Colony algorithm. To that end, the enhancement procedure follows the directional model using by the proposed multi-directional ACO approach. Since our proposed approach uses a multi-directional framework, the multi-directional Ant Colony Optimization algorithm explicitly decomposes the objective space in sub-regions. Indeed, the weight vector is necessary here for the computation of the pseudo-utility ratio (Mansour et al., 2017a) and (Mansour et al., 2017b), it evaluates the performance of the potentially efficient solution relatively to the different objectives. That is, the selection of the solution for a given vector is based on the position of the solution in the objective space. Where each ant attempts to find a new solution according to this position.

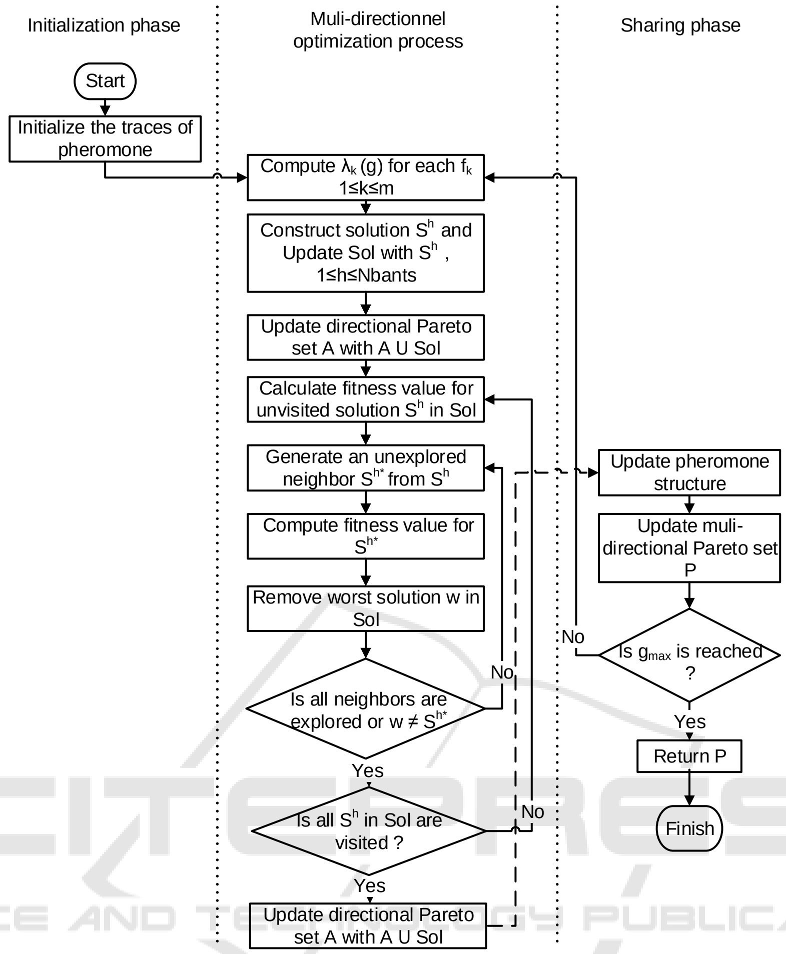


Figure 1: Flowchart of the proposed Multi-directional Hybrid Ant Colony Optimization Algorithm: MD-HACO.

SCITEPRESS SCIENCE AND TECHNOLOGY PUBLICATIONS

5.3.1 Algorithm Description

The local search steps are performed on each sub-region and for each solution found by the colony. Although the method proposed here is inspired from the one introduced in (Mansour et al., 2018), here only one local search procedure is executed using multiple search directions. The proposed method operates as outlined in Algorithm 2: Before performing a local search step, the fitness value of each solution in *Sol* is calculated. This step involves the exploring of the neighborhood of each solution S^h in *Sol* until we find a solution S^{h*} that is better than the worst solution w of *Sol* regarding the search direction λ under consideration. Then, S^{h*} is added to *Sol* and replaces the solution w. The neighborhood exploration process stops once the first improving neighbor is found.

To evaluate a solution S^h against the whole population, we use the augmented weighted Tchebycheff method. This scalarization approach measure the

Algorithm 2: Multi-directional iterated local search.

Input: *Sol* (Solutions set)
Λ (the weight vectors set)

Output: *A* (Directional Pareto set)

For Each $S^h \in Sol$
$Update\ reference\ point\ z^*$
$Calculate\ Fit(S^h|\lambda,z^*) = g^{AT}(S^h|\lambda,z^*)$
Repeat
$Generate\ neighbor\ S^{h*}\ from\ S^h$
$Calculate\ Fit(S^{h*}|\lambda,z^*)$
$Find\ w\ with\ the\ worst\ fitness\ value\ Fit(w|\lambda,z^*)$
If $Fit(S^{h*}|\lambda,z^*) > Fit(w|\lambda,z^*)$
$Replace\ w\ with\ S^{h*}\ in\ Sol$
EndIf
Until $S^{h*} \neq w\ or\ all\ neighbors\ are\ explored$
End For Each
$A \leftarrow Non\text{-}dominated\ solutions\ of\ A \cup Sol$

population diversity without sacrificing convergence with respect to the number of objectives in terms of computational time(L'opez et al., 2013) and (Dächert et al., 2012):

$$g^{AT}(S^h|\lambda, z^*) = max_{k=1...m}\{\lambda_k.|z_k^* - f_k(S^h)|\} + \varepsilon \sum_{k=1}^{m} |z_k^* - f_k(S^h)| \quad (6)$$

where $\varepsilon \geq 0$ is usually chosen as a small positive number and where z^* is the ideal point and update it during the execution of the algorithm as: $z_k^* = max_{k=1...m} f_k(S^h)$

In this paper, we examine the augmented weighted Tchebycheff to calculate the fitness value of each solution. Therefore, good solutions are emphasized in each sub-region to maintain a balance between convergence and diversity using the distance between the solutions and the reference vectors.

5.3.2 Neighborhood Structure

Since the 0/1 multi-objective knapsack problem is a constrained problem, all individuals should satisfy all resource constraints. Indeed, we have to define an efficient neighborhood structure for this problem to both handle constraints and increases the solutions quality.

A solution S^h to the MOMKP can be presented as a double list $I = (I_l^+, I_{\bar{l}}^-)$. The first list I_l^+ corresponds to the taken item (belonging to the solution) $I_l^+ = \{I_1^+, I_2^+, ..., I_T^+\}$, where T is the size of the list. The second list $I_{\bar{l}}^-$ is the list of the remaining items $I_{\bar{l}}^- = \{I_1^-, I_2^-, ..., I_{NT}^-\}$, where NT is the number of unselected items. The transition from one solution S^h to $S^{h'}$ is referred as a move. Two neighborhood operators are performed in sequential order in this paper for the MOMKP:

(1) $U(l^+)$: The extraction ratio is calculated for all items in list I_l^+, which measures the utility value of each item. The lower this ratio is, the worst the item is.

(2) $\overline{U}(\bar{l})$: The insertion ratio is calculated for all unselected items in list $I_{\bar{l}}^-$, the ratio measures the quality of the candidate item according to the solution S^h where the higher this ratio is, the better the item is.

6 SIMULATION AND EVALUATION

We examine the performance of the proposed algorithm, Multi-Directional Hybrid Ant Colony Optimization algorithm MD-HACO, by means of computational experiments in an enlarged sampling size scheme to solve the MOMKP. An empirical comparison to powerful decomposition-based multiobjective algorithms (Gw-ACO: (Mansour et al., 2019), MOEA/D: (Zhang and Li, 2007) and MOEA/D-ACO: (Ke et al., 2013)) and state-of-the-art reference approaches (2PPLS: (Lust and Teghem, 2012)) is investigated.

In order to evaluate the efficiency of a proposed approach we use the following performance metrics: The hypervolume difference (Zitzler and Thiele, 1999) and the summary attainment surface (Da Fonseca et al., 2001). Moreover, we use the non-parametric Mann-Whitney statistical test to verify if the difference between the tested algorithms is statistically significant with a confidence level greater than 95% (p-value $\leq$ 0.05). Note that we obtain similar results using other statistical tests.

6.1 Parametrization of the MD-HACO

Metaheuristic algorithms require a crucial decision about the values of numerous parameters. The solution quality and speed may be affected by the parameter settings. Indeed, we have been striven to determine an appropriate set of parameter values for the MD-HACO. The FQ (frequency in the Gw method) and g_{max} are determinate according to the number of objectives. For instances with 2 objectives, FQ and g_{max} are respectively 800 and 200. For instances with 3 objectives, they are equal to 40 and 100 and for instances with 4 objectives, FQ and g_{max} are set to 20 and 125.

The significance weights for pheromone trail α is set to 1, the significance weights for heuristic information β to 10, the pheromone evaporation rate ρ to 0.90, the number of ants set to 10. The lower and upper bound of pheromone τ_{min} and τ_{max} are respectively set to 1 and 5 and the epsilon parameter of g^{AT} function ε is set to 10^{-3}.

6.2 Experimental Results

Table 1 reports the mean CPU (M-CPU) running time (in second) spent for each instance by Gw-ACO, 2PPLS, MOEA/D, MOEA/D-ACO and MD-HACO. From this table, it is clear that the proposed algorithm is faster (bold values) than the other compared approaches on large instances with 3 and 4 objectives.

Figures 2 plot respectively the median attainment surfaces obtained by comparing Gw-ACO, 2PPLS, MOEA/D, MOEA/D-ACO and MD-HACO on bi-objective instances. The figures illustrate that the

Table 2: Average and standard deviation values of the hypervolume difference metric.

Instance	Gw-ACO		2PPLS		MOEA/D		MOEA/ D-ACO		MD-HACO	
_	Avg	Std	Avg	Std	Avg	Std	Avg	Std	Avg	Std
2_250	4.33E-01	2.72E-02	1.84E-01	1.27E-01	3.05E-01	8.82E-02	2.91E-01	1.49E-01	**1.72E-01**	3.74E-02
2_500	4.41E-01	2.85E-02	1.76E-01	1.19E-01	3.26E-01	9.08E-02	**1.73E-01**	1.59E-01	2.05E-01	1.71E-02
2_750	4.28E-01	3.18E-02	2.19E-01	1.40E-01	3.63E-01	7.03E-02	2.51E-01	1.30E-01	**1.57E-01**	8.56E-03
3_250	5.35E-01	1.85E-02	_	_	4.54E-01	2.91E-02	**1.91E-01**	1.74E-01	2.07E-01	1.95E-02
3_500	5.47E-01	1.73E-02	_	_	4.30E-01	6.72E-02	3.65E-01	1.39E-01	**2.04E-01**	1.34E-02
3_750	5.63E-01	1.00E-02	_	_	4.20E-01	6.79E-02	2.87E-01	1.69E-01	**2.09E-01**	9.75E-03
4_250	5.38E-01	2.53E-02	_	_	4.79E-01	2.39E-02	2.96E-01	8.64E-02	**2.53E-01**	1.17E-02
4_500	5.25E-01	2.67E-02	_	_	4.36E-01	6.69E-02	3.67E-01	9.16E-02	**1.84E-01**	1.54E-02
4_750	5.58E-01	2.42E-02	_	_	4.59E-01	4.66E-02	3.87E-01	5.76E-02	**1.70E-01**	1.21E-02

Table 3: p-values of the Mann-Whitney statistical test.

Instance	2_250				2_500			
Algorithm	Gw-ACO	2PPLS	MOEA/D	MOEA/D-ACO	Gw-ACO	2PPLS	MOEA/D	MOEA/D-ACO
MD-HACO	**≤0.05**	0.440	**≤0.05**	**≤0.05**	**≤0.05**	0.518	**≤0.05**	0.674

Instance	2_750			
Algorithm	Gw-ACO	2PPLS	MOEA/D	MOEA/D-ACO
MD-HACO	**≤0.05**	0.092	**≤0.05**	0.075

Instance	3_250			3_500		
Algorithm	Gw-ACO	MOEA/D	MOEA/D-ACO	Gw-ACO	MOEA/D	MOEA/D-ACO
MD-HACO	**≤0.05**	**≤0.05**	0.900	**≤0.05**	**≤0.05**	**≤0.05**

Instance	3_750		
Algorithm	Gw-ACO	MOEA/D	MOEA/D-ACO
MD-HACO	**≤0.05**	**≤0.05**	**≤0.05**

Instance	4_250			4_500		
Algorithm	Gw-ACO	MOEA/D	MOEA/D-ACO	Gw-ACO	MOEA/D	MOEA/D-ACO
MD-HACO	**≤0.05**	**≤0.05**	**≤0.05**	**≤0.05**	**≤0.05**	**≤0.05**

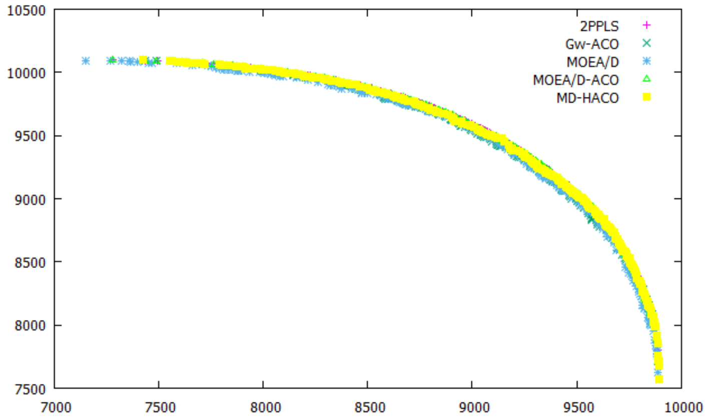
Instance	4_750		
Algorithm	Gw-ACO	MOEA/D	MOEA/D-ACO
MD-HACO	**≤0.05**	**≤0.05**	**≤0.05**

Table 1: CPU running time (seconds).

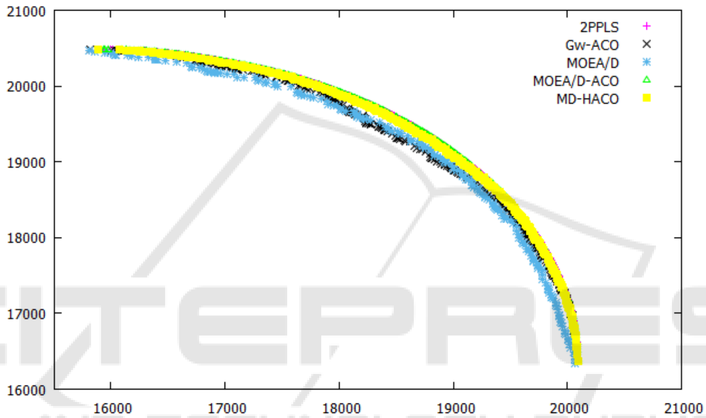
Instance	Gw-ACO	2PPLS	MOEA/D	MOEA/ D-ACO	MD-HACO
2_250	13.2	**3.1**	4.8	5.1	4.6
2_500	56.6	**14.8**	**14.8**	15.2	15.1
2_750	126.1	25.1	28.6	**24.7**	24.8
3_250	34.1	_	9.2	8.7	**3.5**
3_500	135.9	_	24.5	20.7	**12.4**
3_750	310.3	_	48.4	37.9	**26.3**
4_250	51.0	_	22.1	11.8	**8.7**
4_500	206.0	_	46.6	29.9	**20.9**
4_750	471.8	_	130.8	50.5	**48.0**

differences between the attainment surfaces returned by MD-HACO, 2PPLS and MOEA/D-ACO are small and therefore it is difficult to distinguish visually. Nevertheless, it is evident from figures 2 (b) and 2 (c) that the surfaces obtained by MD-HACO are better than those obtained by Gw-ACO and MOEA/D. Indeed, we can clearly observe that the solutions generated by these compared algorithms are below those obtained by the proposed approach.

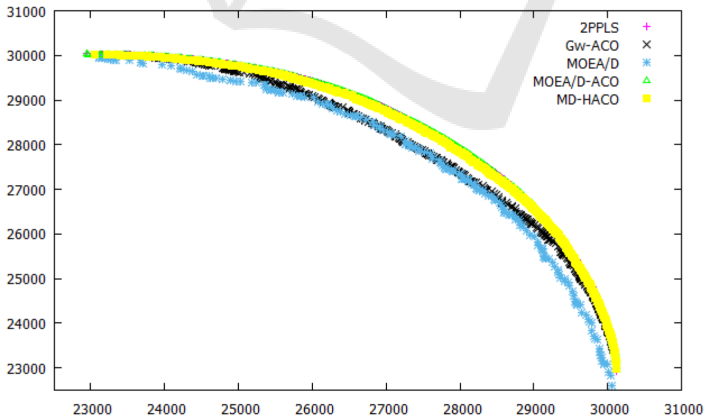
Table 2 presents the mean and standard deviation of the hypervolume difference of MD-HACO and the state-of-the-art reference approaches. Outputs clearly highlight that the proposed hybrid approach is competitive compared to Gw-ACO, 2PPLS, MOEA/D and MOEA/D-ACO. By analyzing the results, one can say that MD-HACO obtains the best performance for 7 out of the 9 instances and MOEA/D-ACO find the best results for 2 out of the 9 instances. However, Gw-ACO, 2PPLS and MOEA/D fail to achieve best values on all tested instances. Moreover, the differences between values obtained by MD-HACO and those returned by Gw-ACO and MOEA/D are very important on the 9 instances. These gaps are smaller between MD-HACO, 2PPLS and MOEA/D-ACO only on the bi-objective instances which confirm the curves relatively close obtained in figures 2. But these gaps grow significantly on the instances with 3 and 4 objectives. Table 3 summarizes the p-values returned by the Mann-Whitney statistical test when comparing MD-HACO against Gw-ACO, 2PPLS, MOEA/D and MOEA/D-ACO. These values are confirmed those of the previous table. Indeed, we can clearly see that on the 2_250 instance, MD-HACO statistically outperforms Gw-ACO, MOEA/D and MOEA/D-ACO. On the 2_500 and 2_750 instances, the proposed approach performs statistically better than Gw-ACO and MOEA/D. When comparing MD-HACO and 2PPLS,



(a)



(b)



(c)

Figure 2: The median attainment surfaces obtained by Gw-ACO, 2PPLS, MOEA/D, MOEA/D-ACO and MD-HACO for bi-objective instances with (a) representes 2_250 instance, (b) representes 2_500 instance and (c) representes 2_750 instance.

SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS

on the bi-objectives instances, the difference is statistically insignificant. On instances with 3 and 4 objectives, the table clearly highlight the performance of MD-HACO.

7 CONCLUSION

This paper presented a new hybrid method for solving the MOMKP based on ACO and local search. The objective space was explored through the use of a multi-directional ant colony optimization approach to search for a promising path in the region considered as well as the contribution of the local search procedure to drive the search toward the Pareto optimal front. From the viewpoints of both the computational expenses and solution quality, the proposed hybrid multi-directional ant colony optimization approach is efficient for MOMKP and performs consistently well especially for high-dimensional problems such as those frequently encountered in real-world applications. An interesting direction of future research would be to investigate a self-adapting version of MD-HACO. In particular, at the level of the optimization part by ant colony, it will be interesting to adopt a self-configuration mechanism where the parameters related to diversification and intensification are automatically configured during the construction process and research.

REFERENCES

Alaya, I., Solnon, C., and Ghedira, K. (2007). Ant colony optimization for multi-objective optimization problems. In *Tools with Artificial Intelligence, 2007. ICTAI 2007. 19th IEEE International Conference on*, volume 1, pages 450–457. IEEE.

Angus, D. (2007). Crowding population-based ant colony optimisation for the multi-objective travelling salesman problem. In *Computational Intelligence in Multicriteria Decision Making, IEEE Symposium on*, pages 333–340. IEEE.

Chabane, B., Basseur, M., and Hao, J.-K. (2017). R2-ibmols applied to a practical case of the multiobjective knapsack problem. *Expert Systems with Applications*, 71:457–468.

Colorni, A., Dorigo, M., and Maniezzo, V. (1991). Distributed optimization by ant colonies.

Colorni, A., Dorigo, M., Maniezzo, V., et al. (1992). An investigation of some properties of an" ant algorithm". In *PPSN*, volume 92.

Da Fonseca, V. G., Fonseca, C. M., and Hall, A. O. (2001). Inferential performance assessment of stochastic optimisers and the attainment function. In *International Conference on Evolutionary Multi-Criterion Optimization*, pages 213–225. Springer.

Dächert, K., Gorski, J., and Klamroth, K. (2012). An augmented weighted tchebycheff method with adaptively chosen parameters for discrete bicriteria optimization problems. *Computers & Operations Research*, 39(12):2929–2943.

Dorigo, M. and Blum, C. (2005). Ant colony optimization theory: A survey. *Theoretical computer science*, 344(2-3):243–278.

Dorigo, M., Maniezzo, V., and Colorni, A. (1991a). The ant system: An autocatalytic optimizing process.

Dorigo, M., Maniezzo, V., and Colorni, A. (1991b). Positive feedback as a search strategy.

Dorigo, M., Maniezzo, V., Colorni, A., et al. (1996). Ant system: optimization by a colony of cooperating agents. *IEEE Transactions on Systems, man, and cybernetics, Part B: Cybernetics*, 26(1):29–41.

Ehrgott, M. and Ryan, D. M. (2002). Constructing robust crew schedules with bicriteria optimization. *Journal of Multi-Criteria Decision Analysis*, 11(3):139–150.

Gambardella, L. M., Éric Taillard, and Agazzi, G. (1999). Macs-vrptw: A multiple colony system for vehicle routing problems with time windows. In *New Ideas in Optimization*, pages 63–76. McGraw-Hill.

Iredi, S., Merkle, D., and Middendorf, M. (2001). Bi-criterion optimization with multi colony ant algorithms. In *Evolutionary Multi-Criterion Optimization*, pages 359–372. Springer.

Ke, L., Zhang, Q., and Battiti, R. (2013). Moea/d-aco: A multiobjective evolutionary algorithm using decomposition and antcolony. *IEEE transactions on cybernetics*, 43(6):1845–1859.

Kellerer, H., Pferschy, U., and Pisinger, D. (2004). Multidimensional knapsack problems. pages 235–283.

L'opez, A., Coello, C. A. C., Oyama, A., and Fujii, K. (2013). An alternative preference relation to deal with many-objective optimization problems. In *International Conference on Evolutionary Multi-Criterion Optimization*, pages 291–306. Springer.

Lust, T. and Teghem, J. (2012). The multiobjective multidimensional knapsack problem: a survey and a new approach. *International Transactions in Operational Research*, 19(4):495–520.

Mansour, I. B. and Alaya, I. (2015). Indicator based ant colony optimization for multi-objective knapsack problem. *Knowledge-Based and Intelligent Information & Engineering Systems 19th Annual Conference*, 60:448–457.

Mansour, I. B., Alaya, I., and Tagina, M. (2017a). A min-max tchebycheff based local search approach for momkp. In *ICSOFT*, pages 140–150.

Mansour, I. B., Alaya, I., and Tagina, M. (2017b). Solving multiobjective knapsack problem using scalarizing function based local search. In *International Conference on Software Technologies*, pages 210–228. Springer.

Mansour, I. B., Alaya, I., and Tagina, M. (2019). A gradual weight-based ant colony approach for solving the multiobjective multidimensional knapsack problem. *Evolutionary Intelligence*, pages 1–20.

Mansour, I. B., Basseur, M., and Saubion, F. (2018). A multi-population algorithm for multi-objective knapsack problem. *Applied Soft Computing*, 70:814–825.

Martello, S. (1990). Knapsack problems: algorithms and computer implementations. *Wiley-Interscience series in discrete mathematics and optimiza tion*.

Stützle, T. and Hoos, H. H. (2000). Max–min ant system. *Future generation computer systems*, 16(8):889–914.

Yagmahan, B. and Yenisey, M. M. (2010). A multi-objective ant colony system algorithm for flow shop scheduling problem. *Expert Systems with Applications*, 37(2):1361–1368.

Zhang, Q. and Li, H. (2007). Moea/d: A multiobjective evolutionary algorithm based on decomposition. *IEEE Transactions on evolutionary computation*, 11(6):712–731.

Zitzler, E. and Thiele, L. (1999). Multiobjective evolutionary algorithms: A comparative case study and the strength pareto approach. *IEEE transactions on Evolutionary Computation*, 3(4):257–271.

SCITEPRESS
SCIENCE AND TECHNOLOGY PUBLICATIONS