Journal of King Saud University – Computer and Information Sciences 31 (2019) 378–384

Contents lists available at ScienceDirect

Journal of King Saud University – Computer and Information Sciences

journal homepage: www.sciencedirect.com

Hybrid Elitist-Ant System for Nurse-Rostering Problem

Ghaith M. Jaradat [a,*], Anas Al-Badareen [b], Masri Ayob [c], Mutasem Al-Smadi [d], Ibrahim Al-Marashdeh [d], Mahmoud Ash-Shuqran [a], Eyas Al-Odat [a]

[a] Department of Computer Science, Faculty of Computer Science and Information Technology, Jerash University, 26150-311 Jerash, Jordan
[b] Department of Software Engineering, Aqaba University of Technology, Aqaba, Jordan
[c] Data Mining and Optimization Group, Centre of Artificial Intelligence, Faculty of Information Science and Technology, National University of Malaysia, 43600 B. B. Bangi, Selangor, Malaysia
[d] College of Applied Studies and Community Service, Imam Abdurrahman Bin Faisal University, Al-Dammam, Saudi Arabia

ARTICLE INFO

Article history:
Received 12 January 2018
Revised 5 February 2018
Accepted 22 February 2018
Available online 26 February 2018

Keywords:
Metaheuristics, Elitist-Ant System
External memory
Diversification
Intensification
Nurse Rostering Problem

ABSTRACT

The diversity and quality of high-quality and diverse-solution external memory of the hybrid Elitist-Ant System is examined in this study. The Elitist-Ant System incorporates an external memory for preserving search diversity while exploiting the solution space. Using this procedure, the effectiveness and efficiency of the search may be guaranteed which could consequently improve the performance of the algorithm and it could be well generalized across diverse problems of combinatorial optimization. The generality of this algorithm through its consistency and efficiency is tested using a Nurse-Rostering Problem. The outcomes demonstrate the competitiveness of the hybrid Elitist-Ant System's performance within numerous datasets as opposed to those by other systems. The effectiveness of the external memory usage in search diversification is evidenced in this work. Subsequently, such usage improves the performance of the hybrid Elitist-Ant System over diverse datasets and problems.

1. Introduction

The problems ofpersonnel rostering are increasingly a great concern among researchers on scheduling and timetabling ever since the last decade. Among others, Nurse Rostering Problems (NRPs) in particular, are complex and challenging to be resolved to optimality (Brucker et al., 2011). At healthcare institutions, scheduling the nurse shifts and distributions has been a tough task. In fact, considering that approximately 40% of hospital budgets go to healthcare personnel, additional care should be taken (Özcan, 2005). Duty rosters that are inflexible or poor can impact the personal life of nurses, escalate job dissatisfaction and could cause high staff turnover rates, which consequently will adversely affect the nursing services received by patients (Özcan, 2005; Burke et al., 2003). As reported in the literature, many leading healthcare organizations all over the world, still prepare the duty roster of nurses manually (Burke et al., 2003; Özcan, 2007; De Causmaecker and Berghe, 2011). Such phenomenon becomes a motivation for researchers to recommend and examine automated solution methodologies as resolution to this problem.

The ongoing research for establishing a successful generic problem-solver makes available the potential for establishing intelligent systems that can create practical solutions for many Combinatorial Optimization Problems (COPs). In the search for computational effectiveness and efficiency, there has been a rapid evolution in the research in meta-heuristics as an effort in finding good quality solutions to these problems within the time frame desired (Osman and Kelly, 1996). Many other researchers in different fields such as (Alsmadi, 2016; Alsmadi et al., 2011, 2012; Alsmadi, 2017a,b; 2018; Badawi and Alsmadi, 2013, 2014) hybridized the meta-heuristics algorithms with other algorithms in order to enhance the performance in finding good quality solutions.

The selection of ant systems (AS) in resolving the aforementioned problems is because it offers (Glover et al., 2003; Dorigo and Stützle, 2010): a probabilistic selection; valuable information; indirect representation; and elitism, and yet still inadequate in establishing balance between diversification and intensification.

* Corresponding author.
E-mail addresses: ghaith_jaradat@yahoo.com (G.M. Jaradat), masri@ukm.edu.my (M. Ayob).

Peer review under responsibility of King Saud University.

Production and hosting by Elsevier

https://doi.org/10.1016/j.jksuci.2018.02.009
1319-1578/

In this study, we present a variant of AS known as the hybrid Elitist-Ant System (Elitist-AS). This study will address the following research question:

"Does the usage of external memory (a pool of varied and high-quality solutions) improve the performance of the Elitist-AS, as opposed to usage of just the diverse pool?"

Grounded on the question above, the current study will attempt to achieve the objectives below:

1. To include a memory structure to generate balance between diversification and intensification;
2. To integrate two assistant systems in order to reinforce the search process, and impose control on the search diversity;
3. To construct an interaction between the two mechanisms and the memory structure to assist pheromone's evaporation and update so that the search exploration and exploitation can be controlled well; *and*
4. To test the performance of the hybrid Elitist-AS by comparing it with similar meta-heuristics over NRPs.

NRP is a problem commonly seen in real life and it has been extensively studied. It thus becomes an appropriate platform for researchers to test the effect of the external memory on the performance of the hybrid Elitist-AS presented in this study.

This paper focuses on the NRP three tracks datasets recommended by the First International Nurse Rostering Competition (INRC2010) prepared by the CODeS research group at the University of Udine in Italy. The NRP is solvable via the assignment of a set of nurses possessing diverse skills and work contracts to a set of types of shift over a specified period of scheduling (Burke et al., 2003). The roster is bound by hard and soft constraints. In the roster, the hard constraints have to be satisfied. Meanwhile, it is desirable to satisfy the soft constraints and such degree of satisfaction dictates the roster's quality. Here, the primary objective is to hit upon a roster that satisfies each hard constraint while minimally violating the soft constraints. Establishing a roster that satisfies every constraint is virtually impossible as it is regarded as a NP-hard COP (Bartholdi, 1981; Millar and Kiragu, 1998). Rostering is mainly the placing, subject to constraints, of resources into slots within a pattern. The portrayal of hard and soft constraints of the INRC2010 datasets and its mathematical formulation and the objective function are provided by (Lü and Hao, 2012).

Following the introduction, the structure of this paper can be found as such: Section 2 discusses the related works; Section 3 elaborates the algorithm and its design; Section 4 explains the experimental structure for the hybrid Elitist-AS; Section 5 presents the computational outcomes generated by the hybrid Elitist-AS and; Section 6 ends the paper with the derived conclusions.

2. The background

First proposed by (Dorigo et al., 1991), the AS is a type of population-based foraging behavior (*aka* swarm intelligence) meta-heuristics. The working of AS is based upon the foraging behavior of actual ants that engage in indirect communication by way of the scattering and dynamic changing of information, also termed as pheromone trails. The load of these trails mirrors the shared search experience, which future ants will exploit in order to more effectively solve a given problem. AS have been used in many COPs and successes have been documented.

External memory refers to an adaptive memory structure that keeps valuable information about the global optima in the shape of a varied and elite set of solutions. Using this structure enables the search process to recombine samples from the elite set allowing the valuable information about the global optima to be exploited.

Further, adaptive memory structure is a primary component of efficient and effective hybrid meta-heuristic (Resende et al., 2010) with the emphasis on the notion of memory, intensification -exploitation- and diversification -exploration- (Črepinšek et al., 2013). A memory refers to the information gathered by the algorithm on the distribution of objective function. Memory is representable as complex structures including pheromone trails in the Elitist-AS. The obtained information is exploited by intensification and such exploitation enhances the current solutions. Typically, this intensification comprises a local search routine (Črepinšek et al., 2013).

Several meta-heuristics have recently been used for the NRP. These include the local search-based and population-based approaches such as (Lü and Hao, 2012; Dorigo et al., 1991; Resende et al., 2010; Črepinšek et al., 2013; Hadwan et al., 2013; Santos et al., 2016; Valouxis et al., 2012; Awadallah and Bolaji, 2015; Rajeswari et al., 2017; Awadallah et al., 2017; Bilgin et al., 2012; Burke and Curtois, 2014; Nonobe, 2010). Thus, a hybrid variant of the Elitist-AS is brought forth in this study alongside the incorporation of external memory, iterated local search (ILS) routine, as well as two mechanisms for intensification and controlling of diversification. The proposed method is underpinned by the successful approaches for the creation of the hybrid Elitist-AS and it is then applied to the NRP.

3. Proposed algorithm

This study expands the scrutiny of the effect of an external memory on the performance and generality of the hybrid Elitist-AS from (Jaradat et al., 2016; Jaradat and Ayob, 2010) through conducting tests on the datasets from the INRC2010 using this system. The major advantages of our hybrid Elitist-AS over the conventional AS are:

1. employing an ILS routine, to further enhance solutions while maintaining diversity;
2. adding an intensification mechanism, to perform a significant enhancement by exploring the neighbors of elite solutions more effectively;
3. adding a diversification mechanism, to trigger the search process again when it stagnates; and
4. adding an external memory, which differs from the pheromone matrices in contents, to store elite solutions to act as a guidance toward the global solution.

The capacity of our Elitist-AS should be improved by having it hybridized with an ILS alongside the application of the mechanisms of diversification and intensification, and the provision of an external memory for elite solutions' storage. A generic pseudo code and a flowchart of the hybrid Elitist-AS can be viewed in Fig. 1 and Fig. 2 (See Appendix A).

The algorithm starts by initializing all parameters (Step 1). Then (Step 2), each ant constructs a solution from scratch (empty roster) using a probabilistic rule guided by two pheromone trails matrices which are represented as the lists of available shifts and skills for a nurse. In the probabilistic rule, i.e. the pairs of nurse-shift and nurse-skill assignments are weighted with the help of the largest degree (LD) of difficulty ordering heuristic. In other words, the unscheduled nurses are ordered based on the largest number of nurses in conflict. The selection of a skill and a shift is based on the pheromone information; a number of hard violations exist if the nurse is assigned into those skills and shift. The probabilistic rule ensures a level of randomness; in which it may reduce the

```
Step 1: Initialization phase
While Stopping Criterion is not met do
Step 2: Construction phase
            For each ant
                        Assign all elements into feasible allocations using a probabilistic rule;
            End for
Step 3: Improvement phase
    While non-improvement stopping criterion is not met do
                  Locally improve each constructed solution; //employ ILS
                  Update size & content of external memory;
    End While
    If the best solution is updated then
Step 4: Intensification phase;
            Randomly explore the neighbors of the best solution found so far (elite solution);
Step 5: Global Pheromone update phase;
          Update pheromone trails for assignments appearing in solution;
    Else
Step 6: Diversification phase;
            Pheromone evaporation; // diversity control
            Reinitialize pheromone trails;
            Generate new population of ant solutions using elite solutions in the external memory by performing some perturbations;
    End If
End While
Step 7: Return Best ant // best solution
```

Fig. 1. A generic pseudo code of the hybrid Elitist-AS (Jaradat et al., 2016).

probability of premature convergence. The values of the pheromone trails are simply presented by the cost of the nurse-shift and nurse-shift assignments. After Step 2, the ant with the lowest number of unscheduled nurses will add certain amount of pheromone to guide the subsequent ant searching for a better solution.

During the construction phase, we represent two pheromone matrices as the weight of the preferable least penalty cost assignment of a nurse into a skill and a shift based on the experience of previous ants. As for the improvement phase, we represent pheromone trails as the estimation of a significant perturbation improvement of a solution. The pheromone trails matrix is updated as follows. Each ant constructs a roster using only local information available for each nurse, which consists of a heuristic function and a pheromone trail. Additionally, an ant uses a list to keep track of the nurses it has assigned into shift and skill and to store the partial roster constructed so far. This list is also used to prevent assigning the already assigned nurses to force the ant to construct feasible assignments.

Initially, each ant selects a nurse and assigns it into feasible shift and skill. Then, starting from that nurse, an ant assigns another nurse into shift and skill until a complete roster is constructed. According to the probability distribution from (Dorigo and Stützle, 2010), an ant probabilistically prefers to assign nurse which cause minimal hard and soft constraint violations. Then two parameters (exploration ratio and weight of hard constraints violations) determine the relative influence of heuristic function and pheromone trail on the roster construction process. If the exploration ratio equals zero, the construction algorithm corresponds to a randomized greedy assignment construction, and the higher the weight of hard constraints violations the closer the constructed assignment will be to those returned by the LD heuristic. The ants repeat these steps until a roster is completed and then they calculate the cost of their constructed roster. Pheromone trails are updated once all ants have constructed their assignments, as a complete roster. This is done by first lowering the pheromone trail of all shifts and skills by a constant factor and then allowing each ant to add pheromone on the shifts and skills it has used.

Steps 3 and 4 are then employed, where we improve solutions using the ILS guided by the pheromone trails matrix. The ILS has four types of neighborhood structures:

1. Hard constraint related neighborhood
   - o Single Shift per Day.
2. Soft constraint related neighborhoods
   - o Weekend.
   - o Overtime-Undertime.
   - o Alternative Qualifications.
   - o Personal Requests.
   - o The most Violated Constraint.
3. Swapping large sections of personal schedules
   - o Shuffle.
   - o Greedy Shuffling.
   - o Core Shuffle.
4. Shaking the solution
   - o Shake a shift.
   - o Shake weekends.
   - o Shake two people.

We use a simple descent heuristic which accepts first improvement in the ILS and then a random descent heuristic is employed. A new solution is accepted if it is better than the current solution. Contents and size of the external memory are then updated. The external memory stores the elite solutions found that are allowed to update the pheromone trails. This guides the search efficiently and prevents constructing the same solutions. It is also used in constructing new solutions in successive iterations from the elite solutions rather than from scratch.

Due to small number of ants and a tiny amount of pheromone update, the hybrid Elitist-AS has limited diversification. However, it has a better capability to diversify rather than intensify the search. On the other hand, the elitism strategy may guarantee a degree of intensification. Therefore, we have specifically selected the ILS due to its capability of balancing exploration and exploitation via the perturbation and local search phases (Blum and Roli, 2008).

The intensification phase will be left out if there is no improvement to the predefined number of iterations, and the diversification phase will be started. In the diversification phase, the pheromone trail values will be reinitialized so that the search can be reinitiated. These steps will be repeated until the stopping criterion is fulfilled. The stopping criterion is fulfilled following the discovery of the maximum amount of iterations or a global solution.

Then in Step 5, for each nurse-shift and nurse-skill pairs, the pheromone is updated based on the persistence of the pheromone trail and evaporation. If shift and skill are not chosen by the ants, the associated pheromone trail decreases exponentially. The best solution found so far is denoted as global-best. This is achieved by adding to the shifts and skills of global-best a small quantity whenever the pheromones are updated (e.g. elitism strategy).

In the end, the diversification mechanism (Step 6), is employed after performing a predefined number of non-improvement iterations applied to the best solution so far. The mechanism periodically erases all pheromone trails if the search stagnates. We then re-initialize pheromone values once the ILS failed to improve all ant solutions. The whole processes of our algorithm are repeated until the stopping criterion is met, either the best solution is found or the number of iterations reached its limit (Step 7).

4. Experimental setup

The hybrid Elitist-AS, was used on more than 69 NRP datasets from INRC2010[1]. The suggestion is that, on each dataset, the hybrid algorithm is performed 25 times for 100,000 iterations as a stopping condition for each dataset. Intel Core i7 2.30 GHz processor, 8 GB RAM, and Java NetBeans IDE v 8.1 were the instruments used for the experiments. The parameters displayed in Table 1 were experimentally determined (*e.g.* size of external memory) and according to the literature (*e.g.* Elitism). For instance, in the Elitist-AS, relatively small population size is preferred (Glover et al., 2003).

The obtained *NRP* test results are exhibited in Table 2 and the best ones are in bold. The table also presents the standard deviation (*Std.*) for each dataset.

Comparison of results between this study's and those of the best known is highlighted in Table 2. Our hybrid method is also compared to the conventional Elitist-AS in Table 2. As evidenced in Table 2, the hybrid Elitist-AS proposed in this study appear to be competitive when compared to the best known methods. It is also clear that our hybrid Elitist-AS has outperformed the conventional Elitist-AS for all datasets. Additionally, the gap between the best known solution and our hybrid Elitist-AS is also shown. A zero gap means the attainment of the best known (or rather optimal) solutions. When compared to the best known solutions, the hybrid Elitist-AS proposed in this study could generate very good solutions. So far, the solutions generated by the hybrid Elitist-AS are optimal for 51 datasets from 69 datasets. Further, in terms of consistency for this study's hybrid method, it can be elucidated by the *Std*. Here, the *Std.* generated by the hybrid Elitist-AS is fairly small (e.g. sprint_early_03, sprint_early_08, sprint_early_09), denoting that our methodology is highly efficient and competitive in resolving the INRC2010. Therefore, the fulfilment of those criteria leads to the generality of the hybrid Elitist-AS over diverse datasets sizes.

5. Computational results and discussion

This section comprises the evaluation of the performance of our hybrid Elitist-AS over other conventional and hybrid methods as found in the literature. Thus, the aims of this paper are to: (i) evaluate the benefit of the integration of an external memory within the hybrid method proposed, and (ii) test the generality and consistency of the proposed hybrid Elitist-AS against the *NRP* and make comparison with those of others.

The hypothesis of the effect of the external memory on the performance of the hybrid Elitist-AS is supported by the comparison between the hybrid Elitist-AS with several traditional and hybrid meta-heuristics with no external memory. For instance, a typical genetic algorithm is equipped with a pool, in particular, an implicit memory. Meaning that, it does not have diverse and high-quality solutions (Blum and Roli, 2008; Talbi, 2009).

Table 2 highlights the comparison of this study's results with those of similar methods. Following the INRC2010 rules, the computational times are: 10 s for sprint datasets, 10 min for medium long datasets, and 10 h for datasets. As can be viewed in the table, the '–' symbol denotes the no running of these problem datasets by the method.

As Table 2 is showing, the proposed hybrid Elitist-AS outperforms other population-based and hybrid methods (e.g., HH-GS, HABC, HS-HC and MODBCO) for nearly across all datasets. For instance, the proposed method obtained superior results (e.g., *long_late_04* = 221, *long_hidden_05* = 41) in comparison to other methods (e.g., HABC, HH-GS, HS-HC and MODBCO).

As shown by Table 2, an explicit memory was not used in any of the methodologies (e.g., IP, HABC and HS-HC). These methodologies are only capable in tackling just one specific problem; basically, the methodologies are a tailored for the solution of a specific problem. This contributes to the lack of effective maintenance of a balance between diversity and quality of the search. Also, a systematic selection strategy or a solution combination strategy is lacking in some of these methodologies (e.g. ATS and VDS-BP), and such dearth may be the cause of their poor outcomes in some datasets. Meanwhile, some utilize a portion of the memory for the storage of the best solution as can be seen in HABC and HH-GS. Somehow, no clear description of the memory's structure, contents and update strategy, was provided.

Table 2 also shows that for nearly all datasets, the proposed hybrid Elitist-AS shows superior performance than other methods that do not include elite pools (e.g., COS and IP-NS) or implicit memory (e.g. HH-GS, HS-HC and ATS). This means that a memory of diverse and high-quality solutions increases the effectiveness of population-based methods. Many results obtained by the proposed method are precisely similar to others (e.g. sprint_late_09 = 17), while in some cases, the proposed method generated worse results in comparison to those by the original approaches. Such observations could be linked to the behavior of the probabilistic of a population-based meta-heuristic. Also, the Elitist-AS is equipped

Table 1
Parameters settings employed by the hybrid Elitist-AS.

Parameter	Value
Population size	Number of ants = 100
Number of iterations	100,000
Number of non-improvement iterations	100
Pheromone initial values	0.01
Evaporation rate	0.25 ∈ [0,1]
Controlling ratio (exploration vs. exploitation)	0.95 ∈ [0,1]
Importance of constraints (penalty)	2.0
Number of employed neighborhood structures per solution	4
Initial external memory size	5
Local search routine	ILS
Search update	Use the best ant to update global pheromone

[1] https://www.kuleuven-kulak.be/nrpcompetition/instances/1

Table 2
The best results on the INRC2010 datasets in comparison to similar methods.

Dataset	Hybrid Elitist-AS		CEAS	HABC	MODBCO	HS-HC	IP	ATS	IP-NS	HH-GS	VDS-BP	COS
	Best	*Std.*		(Awadallah and Bolaji, 2015)	(Rajeswari et al., 2017)	(Awadallah et al., 2017)	(Santos et al., 2016)	(Lü and Hao, 2012)	(Valouxis et al., 2012)	(Bilgin et al., 2012)	(Burke and Curtois, 2014)	(Nonobe, 2010)
sprint_early_01	57	1.9	70	**56**	**56**	58	**56**	**56**	**56**	57	**56**	**56**
sprint_early_02	59	1.1	71	**58**	59	64	**58**	**58**	**58**	59	**58**	**58**
sprint_early_03	**51**	0.7	66	**51**	**51**	59	**51**	**51**	**51**	**51**	**51**	**51**
sprint_early_04	**59**	0.9	75	**59**	**59**	67	**59**	**59**	**59**	60	**59**	**59**
sprint_early_05	**58**	0.3	68	**58**	**58**	63	**58**	**58**	**58**	**58**	**58**	**58**
sprint_early_06	54	0.3	67	54	**53**	58	54	54	54	54	54	54
sprint_early_07	**56**	0.6	70	**56**	**56**	61	**56**	**56**	**56**	**56**	**56**	**56**
sprint_early_08	**56**	0.5	68	**56**	**56**	58	**56**	**56**	**56**	**56**	**56**	**56**
sprint_early_09	**55**	0.9	70	**55**	**55**	61	**55**	**55**	**55**	**55**	**55**	**55**
sprint_early_10	**52**	0.5	67	**52**	**52**	58	**52**	**52**	**52**	**52**	**52**	**52**
sprint_hidden_01	**32**	1.6	66	**32**	**32**	46	**32**	**32**	33	-	-	-
sprint_hidden_02	**32**	1.3	64	**32**	**32**	44	**32**	**32**	**32**	-	-	-
sprint_hidden_03	**62**	2.2	96	**62**	**62**	78	**62**	**62**	**62**	-	-	-
sprint_hidden_04	**66**	0.9	99	**66**	**66**	78	**66**	**66**	67	-	-	-
sprint_hidden_05	**59**	0.8	91	**59**	**59**	69	**59**	**59**	**59**	-	-	-
sprint_hidden_06	**130**	3.6	249	**130**	134	169	**130**	**130**	134	-	-	-
sprint_hidden_07	**153**	4.1	309	**153**	**153**	187	**153**	**153**	**153**	-	-	-
sprint_hidden_08	**204**	1.5	359	**204**	**204**	240	**204**	**204**	209	-	-	-
sprint_hidden_09	**338**	3.1	468	**338**	**338**	372	**338**	**338**	**338**	-	-	-
sprint_hidden_10	**306**	2.1	530	**306**	**306**	322	**306**	**306**	**306**	-	-	-
sprint_hint_01	78	6.9	148	75	**73**	90	-	-	-	78	-	-
sprint_hint_02	47	3.4	89	46	**43**	56	-	-	-	47	-	-
sprint_hint_03	50	7.1	132	50	**49**	69	-	-	-	57	-	-
sprint_late_01	**37**	1.2	64	**37**	**37**	52	**37**	**37**	**37**	40	**37**	**37**
sprint_late_02	42	1.7	65	42	**41**	56	42	42	42	44	42	42
sprint_late_03	48	1	78	48	**45**	60	48	48	48	50	48	48
sprint_late_04	73	2.6	146	73	**71**	95	73	73	75	81	75	76
sprint_late_05	**44**	1	73	**44**	46	57	**44**	**44**	**44**	45	**44**	45
sprint_late_06	**42**	0.7	63	**42**	**42**	52	**42**	**42**	**42**	**42**	**42**	**42**
sprint_late_07	**42**	0.9	91	44	44	55	**42**	**42**	**42**	46	**42**	43
sprint_late_08	**17**	0	53	**17**	**17**	19	**17**	**17**	**17**	**17**	**17**	**17**
sprint_late_09	**17**	0	48	**17**	**17**	**17**	**17**	**17**	**17**	**17**	**17**	**17**
sprint_late_10	**43**	1.4	87	**43**	**43**	54	**43**	**43**	**43**	46	**43**	44
medium_early_01	241	1.8	383	245	245	280	**240**	**240**	**240**	242	244	241
medium_early_02	241	1.1	379	245	243	281	**240**	**240**	**240**	241	241	**240**
medium_early_03	**236**	1.1	370	242	239	287	**236**	**236**	**236**	238	238	**236**
medium_early_04	**237**	1.5	389	240	245	278	**237**	**237**	**237**	238	240	238
medium_early_05	**303**	1.9	448	308	310	330	**303**	**303**	**303**	304	308	304
medium_hidden_01	**111**	4.7	1050	155	143	410	**111**	117	130	-	-	-
medium_hidden_02	**220**	10.6	976	254	230	412	221	**220**	221	-	-	-
medium_hidden_03	**34**	2.8	298	54	53	182	**34**	35	36	-	-	-
medium_hidden_04	**78**	3.1	351	94	85	168	**78**	79	81	-	-	-
medium_hidden_05	**119**	6.6	966	177	182	520	**119**	**119**	122	-	-	-
medium_hint_01	42	7	270	48	42	64	-	-	-	**40**	-	-
medium_hint_02	**91**	9.2	713	94	**91**	133	-	-	-	**91**	-	-
medium_hint_03	140	11.5	1163	140	**135**	187	-	-	-	144	-	-
medium_late_01	161	8.7	748	174	176	234	**157**	164	158	163	187	176
medium_late_02	**18**	2.9	217	31	30	49	**18**	20	**18**	21	22	19
medium_late_03	**29**	2.6	252	38	35	59	**29**	30	**29**	32	46	30
medium_late_04	**35**	0.8	208	48	42	71	**35**	36	**35**	38	49	37
medium_late_05	**107**	7.9	795	134	129	272	**107**	117	**107**	122	161	125
long_early_01	198	4.4	414	197	**194**	339	197	197	197	197	198	197
long_early_02	220	5.9	466	229	228	399	**219**	222	**219**	220	223	**219**
long_early_03	**240**	0	421	**240**	**240**	349	**240**	**240**	**240**	**240**	242	**240**
long_early_04	**303**	0.9	511	**303**	**303**	411	**303**	**303**	**303**	**303**	305	**303**
long_early_05	**284**	1.3	483	**284**	**284**	383	**284**	**284**	**284**	**284**	286	**284**
long_hidden_01	**346**	6.4	2265	400	389	4466	**346**	**346**	363	-	-	-
long_hidden_02	**89**	3.9	550	117	108	1071	**89**	**89**	90	-	-	-
long_hidden_03	**38**	1.3	475	51	48	163	**38**	**38**	**38**	-	-	-
long_hidden_04	**22**	1.7	463	29	27	113	**22**	**22**	**22**	-	-	-
long_hidden_05	**41**	2.8	485	56	55	139	**41**	45	**41**	-	-	-
long_hint_01	40	2	483	42	40	126	-	-	-	**33**	-	-
long_hint_02	28	1.3	336	30	29	122	-	-	-	**17**	-	-
long_hint_03	**55**	1.4	1712	83	79	278	-	-	-	**55**	-	-
long_late_01	**235**	3.4	1941	257	249	588	**235**	237	**235**	241	286	**235**
long_late_02	**229**	1.4	2253	263	261	577	**229**	**229**	**229**	245	290	**229**
long_late_03	**220**	1.1	2007	262	259	567	**220**	222	**220**	233	290	**220**
long_late_04	**221**	2.9	2102	261	257	604	222	227	**221**	246	280	**221**
long_late_05	**83**	0.6	1552	102	92	329	**83**	**83**	**83**	87	110	**83**

with an implicit memory which could be a problem. Thus, the hybrid version proposed in this study is equipped with an explicit memory (e.g. external memory). The external memory is structured well; it can effectively interact with the combination of solutions and the diversification generation methods to deliver an adaptive search update. As such, it may promise a convergence that is fairly quick towards high-quality (or optimal) solutions while the search diversity remains the same. As indicated, Elitist-AS meta-heuristic has an implicit memory for the storage of high-quality and diverse solutions. However, direct application of permutations and perturbations, for instance, the application of problem dependent/specific neighborhood structures, to good quality or diverse solutions for more improvements of quality, can be exhaustive. The transition between implicit and explicit solution representations during the search process is a challenging task. Somehow, the hybrid method proposed in this study demonstrated its effectiveness and consistency across all datasets.

The *Std.* and the $\Delta(\%)$ of the hybrid Elitist-AS show this method's outcomes on all datasets are constant and very close to the best known results from other hybrid meta-heuristics. The obtained outcomes in this study demonstrate the capacity of the hybrid Elitist-AS in generating good quality results on all datasets, not just covering a few.

This study shows that the hybrid Elitist-AS generated competitive results, or even optimal ones for certain datasets, as opposed to the best known results. As proven by the percentage deviation, the results of the hybrid meta-heuristic proposed are almost identical to the best known ones, as demonstrated by the consistency and generality of the hybrid Elitist-AS across all datasets. The hybridization of an explicit memory in the Elitist-AS may be the factor, as it diversifies the search through the exploration of different regions of the search space, or rather, through the escape of local optima, while high-quality solutions are preserved. On the whole, the result implies the significant impact of the hybridization of an explicit memory with the Elitist-AS on its performance in the solution of NRPs.

The hybrid Elitist-AS is applicable to other areas with only minimal changes, that is, only the constructive heuristic and neighborhood structures are to be changed. Usually, the use of a methodology to different datasets of the same problem needs substantial amount of modification, for instance, modification on the parameters or structures of the algorithm. The generality of the proposed hybrid Elitist-AS across different *NRP* datasets has been proven in this work and it is hope that its usage could be expanded to other domains including clustering and big data.

The proposed hybrid algorithm was compared to other identical algorithms with respect to solution quality rather than computational time. It should be noted that the usage of different computer resources causes great difficulty in making comparison. Thus, the amount of iterations became the termination criteria; it is ascribed to the adaptive memory usage (*e.g.* 20 min) in the proposed hybrid algorithm. This results in the execution time that is within the range similar to those found in the literature.

Further, organizing the pheromone trail updates of the ant system requires elitism. Here, only the best ant in each iteration shows the capacity in updating the pheromone trail. These mechanisms have proven their considerable effectiveness in directing the search in the ant system, particularly, the search process of the ant system. Some authors in question disregarded the diversification mechanism in their ant colony system on purpose. According to them, this mechanism is useless in the application of the problem's specific knowledge (heuristic information) presented by local and global pheromone trail updates.

6. Conclusion

This study is motivated by the need to understand how much an external memory is vital for the performance of an Elitist-AS. Hence, this study demonstrated the generalization, and consistency of the hybrid Elitist-AS on the NRP. For this purpose, the impact of an external memory on the general performance were investigated and tested. The hybrid Elitist-AS employs an external memory comprising a set of solutions that are diverse and of high-quality. This memory structure assists in the preservation of balance between diversity and quality of the search. For instance, escape from local optima, *i.e.* the minima or maxima based on the formulation of the problem, is attainable by the use of new solutions production from those diverse ones in the external memory. Thus, it is possible for the search to be diversified for new potential domains. Also, the search can be combined toward better quality solutions by having it focused around the solutions of good quality from the external memory.

As demonstrated by the experimental results, the hybrid algorithm could produce sound outcomes with respect to quality and computational time. Also, the method's performance is generalized well across different datasets using the exact parameter settings. Further, the analysis and discussion on the computational complexity is also evidence of the efficiency of this method across a range of datasets.

This study contributes to the applicable domain as particularized below:

- The hybrid Elitist-AS is equipped with external memory and conducts heuristic perturbations. What this study is demonstrating is that: strengths of different search algorithms can be combined into one hybrid methodology.
- The use of the external memory has generated results that are not only consistent but also generalized across different datasets. Also, high-quality solutions which are either competitive or better than other similar methods, has been generated.
- The hybrid Elitist-AS can be simply executed on different datasets. The proposed method has low level of complexity and size.

The hybrid Elitist-AS has obtained outstanding and optimal results. It outperformed other population-based methods for most datasets, especially those with an implicit memory. Using an explicit memory (e.g. external memory) particularly offers diverse of high-quality solutions from which the proposed hybrid Elitist-AS can initiate its search process to obtain better solutions. Additionally, the external memory makes available a way for implementing cooperation and attaining swifter convergence. It is suggested that the effectiveness of the hybrid Elitist-AS for other COPs as well as the 2nd international nurse rostering competition (INRC-II) is tested.

Appendix A.

See Fig. 2.

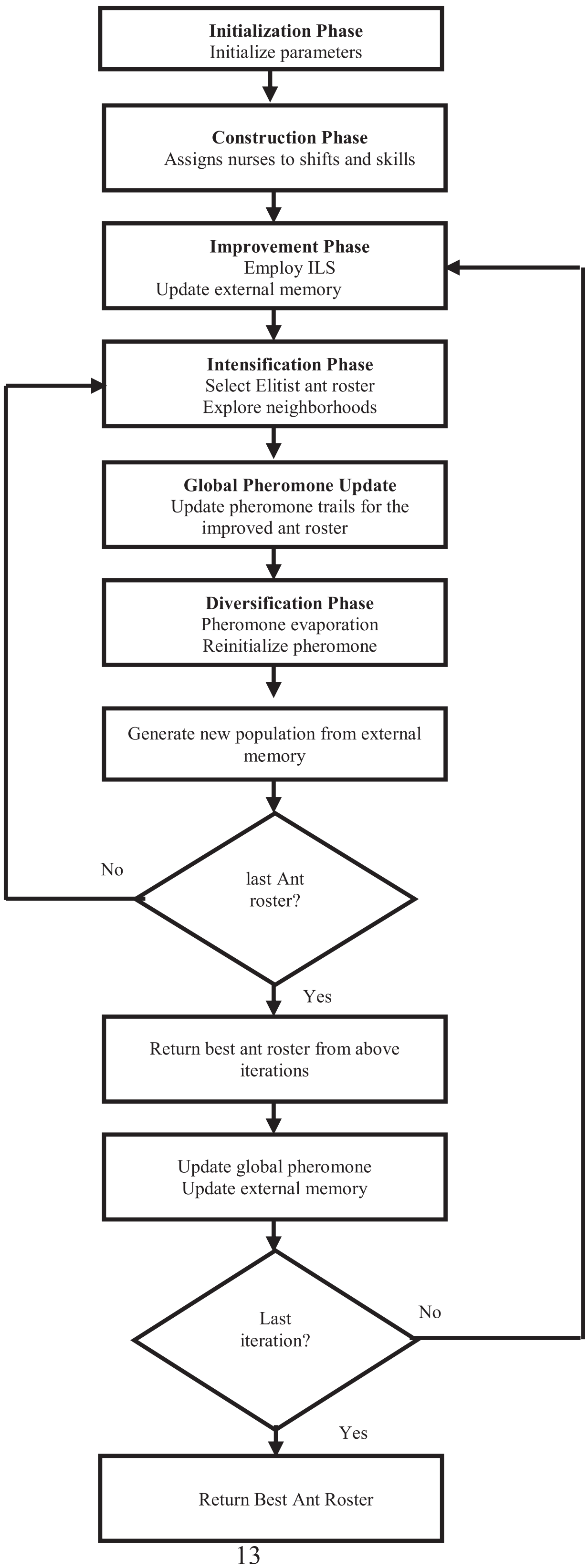

Fig. 2. A flowchart of the hybrid Elitist-AS.

References

Alsmadi, M., 2016. Facial recognition under expression variations. Int. Arab J. Inf. Technol. 13 (1A), 133–141.

Alsmadi, M., Omar, K., Almarashdeh, I., 2012. Fish Classification: Fish Classification Using Memetic Algorithms with Back Propagation Classifier. LAP Lambert Academic Publication.

Alsmadi, M., Omar, K., Noah, S., Almarashdeh, I., 2011. A hybrid memetic algorithm with back-propagation classifier for fish classification based on robust features extraction from PLGF and shape measurements. Inf. Technol. J. 10 (5), 944–954.

Alsmadi, M.K., 2017a. An efficient similarity measure for content based image retrieval using memetic algorithm. Egyptian J. Basic Appl. Sci. 4 (2), 112–122.

Alsmadi, M.K., 2017b. Forecasting River Flow in the USA Using a Hybrid Metaheuristic Algorithm with Back-Propagation Algorithm. Scientific J. King Faisal Univ. (Basic Appl. Sci.) 18 (1), 13–24.

Alsmadi, M.K., 2018. Query-sensitive similarity measure for content-based image retrieval using meta-heuristic algorithm. J. King Saud Univ.- Comput. Inf. Sci. 30 (3), 373–381.

Awadallah, M.A, Bolaji, A.L.a, Al-Betar, M.A., 2015. A hybrid artificial bee colony for a nurse rostering problem. Appl. Soft Comput., 35, 726–739.

Awadallah, M.A., Al-Betar, M.A., Khader, A.T., Bolaji, A.L.a., Alkoffash, M., 2017. Hybridization of harmony search with hill climbing for highly constrained nurse rostering problem. Neural Comput. Appl., 28(3), 463–482.

Badawi, U.A., Alsmadi, M.K.S., 2013. A Hybrid Memetic Algorithm (Genetic Algorithm and Great Deluge Local Search) With Back-Propagation Classifier for Fish Recognition. Int. J. Comput. Sci. Issues 10 (2), 348–356.

Badawi, U.A., Alsmadi, M.K., 2014. A general fish classification methodology using meta-heuristic algorithm with back propagation classifier. J. Theor. Appl. Inf. Technol. 66 (3), 803–812.

Bartholdi III, J.J., 1981. A guaranteed-accuracy round-off algorithm for cyclic scheduling and set covering. Oper. Res. 29 (3), 501–510.

Bilgin, B., Demeester, P., Misir, M., Vancroonenburg, W., Berghe, G.V., 2012. One hyper-heuristic approach to two timetabling problems in health care. J. Heuristics 18 (3), 401–434.

Blum, C., Roli, A., 2008. Hybrid metaheuristics: an introduction. Hybrid Metaheuristics. Springer, pp. 1–30.

Brucker, P., Qu, R., Burke, E., 2011. Personnel scheduling: models and complexity. Eur. J. Oper. Res. 210 (3), 467–473.

Burke, E.K., Curtois, T., 2014. New approaches to nurse rostering benchmark instances. Eur. J. Oper. Res. 237 (1), 71–81.

Burke, E., De Causmaecker, P., Petrovic, S., Berghe, G.V., 2003. Variable neighborhood search for nurse rostering problems. Metaheuristics: Computer Decision-Making. Springer, pp. 153–172.

Črepinšek, M., Liu, S.-H., Mernik, M., 2013. Exploration and exploitation in evolutionary algorithms: a survey. ACM Comput. Surveys (Csur) 45 (3), 35.

De Causmaecker, P., Berghe, G.V., 2011. A categorisation of nurse rostering problems. J. Sched. 14 (1), 3–16.

Dorigo, M., Stützle, T., 2010. Ant colony optimization: overview and recent advances. Handbook of Metaheuristics. Springer, pp. 227–263.

Dorigo, M., Maniezzo, V., Colorni, A., 1991. Positive feedback as a search strategy.

Glover, F., Laguna, M., Martí, R., 2003. Scatter search. Advances in Evolutionary Computing. Springer, pp. 519–537.

Hadwan, M., Ayob, M., Sabar, N.R., Qu, R., 2013. A harmony search algorithm for nurse rostering problems. Inf. Sci. 233, 126–140.

Jaradat, G.M., Ayob, M., 2010. An elitist-ant system for solving the post-enrolment course timetabling problem. Database Theory and Application, Bio-science and Bio-technology. Springer, pp. 167–176.

Jaradat, G., Ayob, M., Almarashdeh, I., 2016. The effect of elite pool in hybrid population-based meta-heuristics for solving combinatorial optimization problems. Appl. Soft Comput. 44, 45–56.

Lü, Z., Hao, J.-K., 2012. Adaptive neighborhood search for nurse rostering. Eur. J. Oper. Res. 218 (3), 865–876.

Millar, H.H., Kiragu, M., 1998. Cyclic and non-cyclic scheduling of 12 h shift nurses by network programming. Eur. J. Oper. Res. 104 (3), 582–592.

Nonobe, K., 2010. INRC2010: An approach using a general constraint optimization solver. The First International Nurse Rostering Competition (INRC 2010).

Osman, I.H., Kelly, J.P.,1996. Meta-heuristics: an overview. Meta-heuristics. Springer, pp. 1-21.

Özcan, E., 2005. Memetic algorithms for nurse rostering. In: International Symposium on Computer and Information Sciences, LNCS 3733, pp. 482–492.

Özcan, E., 2007. Memes, self-generation and nurse rostering. In: International Conference on the Practice and Theory of Automated Timetabling 2006, LCNS 3867, pp. 85–104.

Rajeswari, M., Amudhavel, J., Pothula, S., Dhavachelvan, P., 2017. Directed Bee Colony Optimization Algorithm to Solve the Nurse Rostering Problem. Comput. Intell. Neurosci. 2017.

Resende, M.G., Ribeiro, C.C., Glover, F., Martí, R., 2010. Scatter search and path-relinking: Fundamentals, advances, and applications. Handbook of Metaheuristics. Springer, pp. 87–107.

Santos, H.G., Toffolo, T.A., Gomes, R.A., Ribas, S., 2016. Integer programming techniques for the nurse rostering problem. Ann. Oper. Res. 239 (1), 225–251.

Talbi, E.-G., 2009. Metaheuristics: from design to implementation. p. 74.

Valouxis, C., Gogos, C., Goulas, G., Alefragis, P., Housos, E., 2012. A systematic two phase approach for the nurse rostering problem. Eur. J. Oper. Res. 219 (2), 425–433.