Ilmenauer Beiträge zur Wirtschaftsinformatik

Herausgegeben von U. Bankhofer, V. Nissen
D. Stelzer und S. Straßburger

Maik Günther, Volker Nissen

Application of Particle Swarm Optimization to the British Telecom Workforce Scheduling Problem

Arbeitsbericht Nr. 2013-04, Dezember 2013

Technische Universität Ilmenau
Fakultät für Wirtschaftswissenschaften
Institut für Wirtschaftsinformatik

Autor: Maik Günther, Volker Nissen

Titel: Application of Particle Swarm Optimization to the British Telecom Workforce Scheduling Problem

Ilmenauer Beiträge zur Wirtschaftsinformatik Nr. 2013-04, Technische Universität Ilmenau, Dezember 2013

ISSN 1861-9223

ISBN 978-3-938940-49-5

URN urn:nbn:de:gbv:ilm1-2013200237

Anschrift: Technische Universität Ilmenau, Fakultät für Wirtschaftswissenschaften, Institut für Wirtschaftsinformatik, PF 100565, D-98684 Ilmenau.
http://www.tu-ilmenau.de/wid/forschung/ilmenauer-beitraege-zur-wirtschaftsinformatik/

Inhalt / Contents

A prior version of this paper appeared in the conference proceedings of PATAT 2012.

Full Citation:

Günther, M.; Nissen, V.: Application of Particle Swarm Optimization to the British Telecom Workforce Scheduling Problem. In: Kjenstad, D.; Riise, A.; Nordlander, T.E.; McCollum, B.; Burke, E. (eds.): Proceedings of the 9th Int. Conference on the Practice and Theory of Automated Timetabling, SINTEF, Trondheim, 2012, 242 – 256.

Zusammenfassung/Abstract:

This work adresses a practical problem that is of relevance in several industries, such as logistics, maintenance work, mobile health care and security services. When workers are deployed in field service, they must be allocated to the correct assignment and their routes should also be optimised as a part of that process. Data from a practical case of British Telecom has been used widely in the literature to test many different solution methods. We suggest a modification of particle swarm optimization (PSO) for this problem and compare the performance of the resulting hybrid approach to competing solution methods.

PSO produces better results than the currently best-known solution that was achieved using fast guided local search. Combined with our previous results on sub-daily staff scheduling in logistics this result underlines the potential of PSO to solve complex workforce scheduling problems. Moreover, there is a strong indication that hybridising a metaheuristic with a problem-specific repair heuristic is a useful approach of resolving the conflict between domain-specific characteristics of a real-world problem and the desire to employ a generic optimisation technique, at least in the domain of workforce management.

Schlüsselwörter/Key Words:

Combinatorial Optimization, Workforce Scheduling, Particle Swarm Optimization, Hybrid Metaheuristics

Application of Particle Swarm Optimization to the British Telecom Workforce Scheduling Problem

Maik Günther · Volker Nissen

Abstract This work adresses a practical problem that is of relevance in several industries, such as logistics, maintenance work, mobile health care and security services. When workers are deployed in field service, they must be allocated to the correct assignment and their routes should also be optimised as a part of that process. Data from a practical case of British Telecom has been used widely in the literature to test many different solution methods. We suggest a modification of particle swarm optimization (PSO) for this problem and compare the performance of the resulting hybrid approach to competing solution methods. PSO produces better results than the currently best-known solution that was achieved using fast guided local search. Combined with our previous results on sub-daily staff scheduling in logistics this result underlines the potential of PSO to solve complex workforce scheduling problems. Moreover, there is a strong indication that hybridising a metaheuristic with a problem-specific repair heuristic is a useful approach of resolving the conflict between domain-specific characteristics of a real-world problem and the desire to employ a generic optimisation technique, at least in the domain of workforce management.

Keywords Combinatorial Optimization · Workforce Scheduling · Particle Swarm Optimization · Hybrid Metaheuristics

1 Introduction

The combination of route and personnel scheduling arises in many different applications, for example for the assignment of technicians in field service, for the allocation of care workers,

M. Günther
Burghart 14a, 86920 Denklingen, Germany
Tel.: +49-08243-9930844
E-mail: maik.guenther@gmx.de

V. Nissen
Ilmenau University of Technology, Faculty of Economic Sciences
Department of Information Systems in Services
P.O. Box 100565, D-98684 Ilmenau, Germany
Tel.: +49-3677-694047
Fax: +49-3677-694219
E-mail: volker.nissen@tu-ilmenau.de

transportation companies or security services. Each industry has its own unique characteristics. Patients should be cared for by the same care worker as much as possible and in transportation companies basic routes are usually dictated. Complex planning demands especially arise when technicians are being assigned. Many different qualifications, timeslots for tasks, limited employee availability, individual performance figures and a complex street network are only some of the constraints which must be taken into account. For these reasons, this work looks into technician assignment at British Telecom (BT). In contrast, for instance, to the France Telecom problem used in the more recent ROADEF-challenge [9], the BT-problem assumes individual workers instead of temporary teams to work on a particular job. This makes the results more easily transferable to other branches of industry with similar characteristics.

For almost two decades, various methods have been tested using BT's data, including simulated annealing (SA), genetic algorithms (GA), constraint logic programming (CLP), local search (LS) fast local search (FLS) and fast guided local search (fast GLS). Because very good results have already been achieved using hybrid versions of particle swarm optimization (PSO) on similar scheduling problems [28,18], this solution approach will be investigated for use on the British Telecom problem.

The research goals we pursue are twofold. First, we aim for good solutions to a meaningful and complex practical application that is of significant economic value in diverse industries as mentioned above. Second, we want to contribute to the comparison of modern metaheuristics on practical problems of realistic size and complexity.

First, section 2 gives a description of the application problem, also discussing the appropriate representation of the problem for PSO as well as the complexity. Then we highlight related work from the literature in section 3. The PSO approach is outlined in section 4. An experimental assessment of PSO and a comparison with prior solution methods is done in section 5. The paper concludes with a short summary of main results and some avenues for further research.

2 The Application Problem

2.1 Problem Description

The problem discussed here comes from British Telecom and consists of a planning scenario in which 118 technicians are to handle 250 spatially separated jobs in one day (actual data can be found in [4]). The working time models – i.e., starting and ending times – of the technicians are given for the day to be planned and may not be changed during planning.

A total of 250 jobs exist $J = \{1,\ldots,J\}$. Each job j consists of a five-element tuple: job number, map coordinate x, map coordinate y, duration and job type. The x- and y-coordinates can be used to calculate the travel costs ct (interpreted as error points of the respective solution) for the paths traveled as follows:

$$ct((x_1,y_1),(x_2,y_2)) = \begin{cases} \frac{\frac{1}{2}*\Delta_x+\Delta_y}{8}, & \text{if } \Delta_x > \Delta_y \\ \frac{\frac{1}{2}*\Delta_y+\Delta_x}{8}, & \text{else} \end{cases} \tag{1}$$

The duration d_j of job j can be between 10 and 513 minutes. This value does not represent the actual job time but rather the time required by the average qualified technician $E = \{1,\ldots,E\}$. The actual job duration time rd_j is highly dependent on the experience level

r_e of the technician. Formula 2 shows the calculation of that factor. A task can only be carried out by one technician alone, eliminating the possibility of accelerating job time through cooperation. Neither is it allowed to change technicians during the fullfilment of a job.

Job type refers to the time of day when it is to be carried out. These requirements are hard constraints, meaning they must be fulfilled. Three different job types are distinguished:

- Morning: The job must begin before 12:00.
- Afternoon: The job must begin after 12:00.
- No preference: No requirement has been given regarding starting time.

Each of the 118 technicians is deployed by contract for 8 hours. Starting time is either 8:00 or 8:30 with corresponding ending times of 16:00 or 16:30. A technician does not necessarily have to be either traveling or completing a job during the 8-hour shift. It is possible for a technician not to be assigned any jobs during the alotted time. A tuple for a technician consists of five elements: working time start, working time end, experience level r_e and the x- and y-coordinates of his or her service center.

With the help of a technician's experience level r_e and the average duration d_j of a job the actual job time rd_j of job j can be calculated. For example, assuming a technician with experience level 8 (values below ten signify above average experience) and an average job duration of 20 minutes, the actual completion time rd_j is 16 minutes.

$$rd_j = d_j * \frac{r_e}{10} \tag{2}$$

Technicians begin their routes at their respective service centers at the start of their working day and must reach the centers again after completing the last job while within their total working time. A total of 11 service centers exist. Figure 1 shows the positions of the jobs and the service centers. The numbers next to the service centers indicate the number of technicians assigned to each of them.

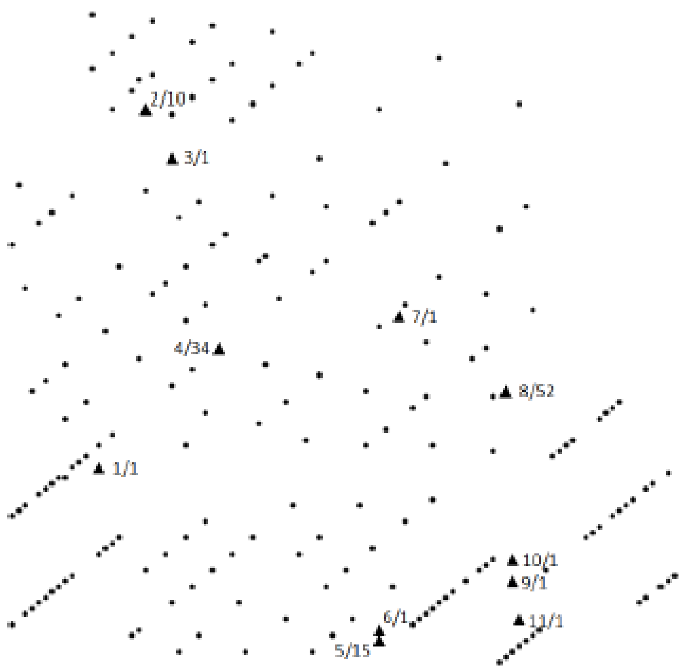

Fig. 1 Job and service center position [23, 14] (numbers: service center ID/number of assigned technicians)

Table 1 gives the number of technicians E_i for each service center and the total available capacity of technicians, which is determined using daily working time in minutes.

Table 1 Service Center Capacity [23, 18]

Service center ID	1, 3, 6, 7, 9, 10, 11	2	4	5	8
Number of technicians E_i	1	10	34	15	52
Capacity in min.	480	4.800	16.320	7.200	24.960

Qualifications represent hard constraints. Jobs may not be assigned to unqualified technicians. There exists for each job a set of employees who are eligible for that job. The number can vary between a single qualified technician up to 107 technicians.

Due to the aforementioned constraints and restrictions (job duration, travel times, job type, working times etc.) it is possible for some jobs to remain unfulfilled. As part of the objective function, the volume of incomplete work is to be minimized. For this, an error value f is added to the average job duration d_j of job j, with the value of f set to 60 for this problem, according to the literature. A binary variable s_j indicates whether the job will be completed ($s_j = 0$) or not ($s_j = 1$). The error value f simply increases the weight of the unassigned jobs in the objective function. Also, travel time should be kept as small as possible, since it is not a value-producing activity and utilizes resources. The objective function that minimizes the total number of error points can then be calculated as follows:

$$c = \sum_{e=1}^{E} ct_e + \sum_{j=1}^{J} (d_j + f) * s_j \tag{3}$$

2.2 Classification of the Problem Space

Ernst et al. [13] provide a comprehensive overview of problems and solution methods for personnel scheduling and rostering from more than 700 analysed sources between the years 1954 and 2004. Moreover, they classify the different problems they encountered in the literature. According to this classification, the British Telecom problem can be viewed as a problem of “task-based demand” because demand arises from the jobs to be completed. These tasks have an earliest start time, a duration and a latest end time. Also, the employees are assigned to shifts before job allocation is done so that absent employees are known. Therefore, the British Telecom problem also belongs to the group “task assignment”.

In addition to the classification of scheduling problems according to Ernst et al., the present problem can also be classified within the context of the traveling salesman problem. Azarmi and Abdul-Hameed [1] place it in the class of multi-time-constraint traveling salesman problems (multi-TCTSP). Technicians must travel to a series of locations in the shortest order and return to their respective starting positions while adhering to their working time allotment. The problem then becomes a multi-TCTSP because multiple technicians are available and each technician route is a TCTSP. Also, there are restrictions with respect to job time constraints. For this reason, the problem becomes a multi-TCTSPTW (TW = time windows). Furthermore, the addition of qualifications turns the problem into a mutli-SDTCTSPTW (SD = Site Dependent). Finally, the existence of multiple service centers brings about the full classification as a multi-MDSDTCTSPTW (MD = Multi-Depot).

2.3 Problem Representation

In order to use the various solution methods, the problem must be represented in an appropriate way. For this, Tsang et al. [36] use a permutation of all jobs. The permutation is transformed into an assignment plan using the objective function. However, with this method, some regions of the solution space are excluded from the start. This means that some shorter paths may not necessarily be found. Therefore, this method is not used in the present work even though it would reduce the number of plausibility checks and correction mechanisms in the utilized solution method.

In the present work, each technician is assigned his or her own permutation of jobs to be completed. The permutation can possibly contain all 250 jobs, which however will not occur in practice. This forms a two-dimensional matrix, in which the rows represent the technicians and the columns stand for series of jobs. Each matrix element contains a job number and the job order in each permutation determines when the job is to be completed, while travel and completion times as well as restrictions on start times (where applicable) are accounted for. Matrix elements without a job receive a uniform dummy value. Each job must be assigned to exactly one technician. If a technician is allocated more jobs than he or she can complete, the objective function marks the excess jobs as incomplete.

2.4 Complexity

With respect to assignment planning, Garey and Johnson showed in 1979 [15] that even the simplest forms of staff scheduling are NP complete. Bartholdi [3] proved the NP completeness of personnel scheduling with shift cycles, in which the employees are available with interruptions. In 1982, Tien and Kamiyama [32] showed that practical personnel scheduling problems are more complex than the traveling salesman problem (TSP), which is already NP complete by itself. From an experimental point of view, the works of Easton and Rossin [12] as well as Brusco and Jacobs [5] suggest that general personnel assignment planning problems are difficult to optimize while Cooper and Kingston [8] demonstrated that they are even NP complete. Finally, Kragelund and Kabel [24, 12–15] proved that the general employee timetabling problem is NP hard.

The NP-hard British Telecom problem encompasses 118 employees and 250 jobs, which results in a two-dimensional matrix with 29,500 elements, of which 250 (the actual jobs) do not contain the dummy value. The complexity of the problem space is J^S, where J is the total number of jobs and S the number of jobs for which the average technician is qualified. This yields 250^{27} combinations (approx. 10^{56}) [38].

3 Related Work

Many solution methods have been tested in the past for solving the British Telecom problem. However, the problem space was sometimes modified, such that not all approaches can be compared. This is true, for instance, for the work by Kokkoras et al. [23], [30]. They generate an agent for each service center and solve the sub-problems using CLP from Yang [38]. All solution methods which have been tested on the original version of the British Telecom problem are listed in table 2. The results are also shown in section 5.

The first publication on the British Telecom problem was done by Baker in 1993 [2]. He uses SA, which was used at that time in the British Telecom's Software Work Manager.

He represents the problem space as a set of routes, in which each technician is assigned one route, which can also be empty. Jobs which cannot be completed are allocated to a dummy technician. Four different actions are available for the generation of a move.

In the same year, Muller et al. presented their work [26] using distributed working GAs. Each of the GAs can have its own method to transform a chromosome into an assignment plan. Additionally, they can differ with respect to their behavior. Shared memory exists in which the respective best chromosome is saved. This provides access to chromosomes generated from different approaches.

Two CLP prototypes were presented by Azarmi and Abdul-Hameed in 1995 [1]. The first is tour generation (CLP TG), in which modeling of the problem space is done by assigning each job to a technician and excess jobs are assigned to a dummy employee. The second method involves implemented compact generation (CLP CG), which allows parallel processing. The main difference between this method and CLP TG is that a technician's route is immediately resequenced into an optimum order as soon as a new job is assigned.

A year later, Yang published his CLP approach with and without forward checking [38]. He also tested three variantions for ordering during allocation of jobs to technicians.

Tsang and Voudouris utilized LS for the British Telecom problem in 1997 [36]. In order to represent the problem space, they use a permutation of the jobs to be completed (see section 2.3). LS proceeds by exchanging two jobs within the permutation if this improves solution quality. Starting with LS, Tsang and Voudouris integrate FLS and fast GLS together. FLS has the goal of accelerating the optimization method. But this has the consequence that good results could remain out of consideration. There exists for the position of each job in the permutation an activation bit, through which it is determined whether a job remains in consideration or not. For GLS, the objective function is expanded to take into account additional error points due to other rules, so that the search can escape from local optima and can extend into other regions. In summary, fast GLS has yielded the best results up to now. For a long time, no improvements on the BT problem have been published, as apparently it is difficult to surpass the results produced by fast GLS.

Table 2 Solution Methods and Authors for the (original) British Telecom Problem

Method	Author(s)	Year	Source
SA	Baker	1993	[2]
GA	Muller, Magill, Prosser and Smith	1993	[26]
CLP	Azarmi and Abdul-Hameed	1995	[1]
CLP	Yang	1996	[38]
LS	Tsang and Voudouris	1997	[36]
FLS	Tsang and Voudouris	1997	[36]
Fast GLS	Tsang and Voudouris	1997	[36]

Several modifications of the BT problem occur in the literature. Kliem and Anderson [22], for instance, expanded the original problem by investigating correct team formation. Relationships between success of a project and the personalities of the team members were analysed to be able to choose the person with the right personality for a certain project.

Naveh et al. [27] focus on matching the right employee with the appropriate job by considering different individual characteristics. They use constraint programming, which is especially suited to this problem.

During the ROADEF 2007 competition [9] various solution methods were analysed using data sets from France Telecom. These vary between 5 and 150 employees and from 5 and 800 jobs. Qualifications as well as job completion order and priority are taken into account. Additionally, the number of vehicles is limited. In contrast to the British Telecom problem, in which employees always work alone, here teams are created which exist for multiple days. In the ROADEF competion computing time was a limiting factor. Therefore, constructive methods were used quite often which were highly varied with respect to their functionality, also explaining the standings. At times, the problem space was significantly fragmented in order to reduce complexity.

In 2005 and 2008, Tsang et al. presented an agent-based approach utilizing the RECONET protocol for solving the British Telecom problem using dynamic changes [33], [34]. Experiments were only done on randomly generated problem spaces. Whereas Kokkoras et al. [23], [30] only generated one agent for each service center, agents will additionally be generated for regions with jobs. Also, there is a superior agent (manager), who controls the other agent types.

In 2008, Tsang et al. [35] extended the multi-agent system by adding aspects to support employee self-determination. However, their work only describes an idea – no experiments were carried out.

We now give a short overview of particle swarm optimization to complete the picture of related work. PSO is a population-based metaheuristic based on the concept of swarm intelligence. The basic principles of PSO were developed by Kennedy and Eberhart among others [19,20] for the solution space of the form $\mathbb{R}^d$. Swarm members are assumed to be massless, collision-free particles that search for optima with the aid of a fitness function within a solution space. In this process each single particle together with its position embodies a solution to the problem [37]. While looking for the optimum, a particle does not simply orient itself using its own experience but also using the experience of its neighbours [14]. This means that the particles exchange information, which can then positively influence the development of the population in the social system as a whole [29].

In PSO each particle knows its current position in the solution space, its personal best position (pBest) and the best position of its global (gBest) or local (lBest) neighbourhood. The basic PSO procedure is given in algorithm 1.

Algorithm 1 Basic PSO Procedure

```
initialize the swarm
calculate the fitness of initial particles
determine pBest for each particle and gBest (or lBest)
repeat
   for i = 1 to number of particles do
      calculate new position
      calculate fitness
      new pBest and new gBest (or lBest)?
   end for
until termination criterion reached
```

Modifications of standard real-valued PSO exist for binary variables, where the speed of a particle is used as the probability for the change of the binary value [20]. This approach, however, has several limitations and was changed from binary to decimal variables in [37]. A combinatorial PSO-variant was developed for sequence planning tasks in [31]. In 2006 Chu et al. [7] adapted PSO for exam scheduling. They changed PSO in such a way that velocity is no longer calculated in order to determine the new position of a particle. Instead, the new position of a particle in each iteration results from the exchange of two allocations in one particle as well as from copying an allocation from pBest or gBest into the new particle position. Brodersen and Schumann [6] build upon this approach and use it for university schedule generation.

In the following section a new solution approach for the original BT problem based on an adaptation of particle swarm optimization is described. The choice of PSO as a metaheuristic approach to solve this problem was motivated by the very good performance of PSO on a roughly similar staff scheduling problem from logistics [18].

4 Particle Swarm Optimization for the British Telecom Problem

4.1 Outline of Particle Swarm Optimization Approach

The British Telecom problem is a combinatorial problem space in which integers are used, which means PSO must be modified in an appropriate way. We have previously expanded the above mentioned approaches of Chu et al. [7] and Brodersen and Schumann [6] by adding probabilities of different actions and this expanded method has been successfully applied to personnel assignment in logistics [28], [17], [18]. The PSO algorithm modified for the British Telecom problem is based on that previous work and will be explained in detail below. First, however, we discuss methods to avoid premature convergence on a local suboptimum as this is a major obstacle in finding good solutions.

In connection with premature convergence, choosing an appropriate neighbourhood topology is important. The topologies used most often in the original form of PSO are the gBest and lBest topologies. In gBest the swarm members are connected in such a way that each particle is a neighbour of every other particle. This means that each particle immediately knows the best global value found up to that point. All particles are included in the position calculation of gBest. If the global optimum is not located near enough to the best particle, it can be difficult for the swarm to search other parts of the solution space, possibly converging instead to a local optimum [25].

Avoiding such convergence to a sub-optimum is one of the goals of the lBest topology, in which a particle is only connected to its immediate neighbours. The parameter k represents the number of neighbours of a particle. With k=2 the topology is a circle (or ring). Increasing k to particle count minus 1 yields a gBest topology. In an lBest topology, each particle only possesses information about itself and its neighbours. The swarm converges slower than with gBest but also has a higher chance of finding the global optimum [20].

Another neighbourhood form is the wheel. There exists a central particle which is connected to all other particles. These particles only have that central particle as its neighbour and are isolated from all others. This arrangement prevents a new best solution from being immediately distributed throughout the swarm.

Results on neighbourhood topologies and premature convergence are ambiguous in the literature, as is further discussed in [17]. Therefore, in the experimental section, all topologies outlined here are tested and compared.

Another option for preventing premature convergence for the British Telecom problem is to outfit each particle with the capability of looking ahead to its new position (forward checking) and to decide whether that position is potentially worthwhile. This is sensible because the particles in the British Telecom problem could possible penetrate into regions that would lead to increasing deteriation of the solution. After several iterations the particles

Fig. 2 Neighbourhood topologies gBest, wheel, circle (k=2) and lBest (k=4) [25].

will have been changed so much that they can no longer escape from such a local optimum. The particles no longer communicate just with each other. They can also "see". There is still a small probability for the particles to accept worsening solutions. This prevents the particle from becoming trapped in a local optimum, unable to move in the solution space.

Our version of PSO that was adapted for the British Telecom problem is shown in algorithm 2. In particular, the calculation of the new particle position has been modified compared to standard PSO. Velocity is no longer required. This means that the constriction factor and the inertia weight can be omitted. The same applies to dimension overrun. One may argue that the resulting method should no longer be called PSO. However, inertia weight and constriction factor were also not present in the original version of standard PSO, but later added since this helped to improve results. Moreover, in our view, the basic properties of PSO are swarm intelligence and a combination of individual and social behavior. These remain intact in our combinatorial variant of PSO.

The new particle position is calculated in line 7 of algorithm 2. The calculation occurs within a loop with the index w. This is necessary because the two-dimensional matrix of a particle cannot be systematically processed in our approach to determine the new particle position, yet several changes are to be carried out in each iteration. Prior tests were used to heuristically determine a value for w of 300. In practice, however, 300 changes are never applied to a particle in one iteration. Some changes, for instance, cannot be carried out because of missing qualifications. Moreover, it might occur during copying from pBest or gBest that no actual change is made to the particle position because the copied element is already located at the specified position. Some changes may be rejected by the particle due to an unacceptable deterioraton of the solution.

Algorithm 2 Modified PSO

```
1:  initialize the swarm using constructive heuristic of algorithm 1
2:  calculate the fitness of the particle
3:  determine pBest and gBest (or lBest)
4:  repeat
5:    for i = 1 to number of particles do
6:      for w = 1 to 300 do
7:        calculate the new particle position with the help of 6 actions
8:      end for
9:      repair the particle using repair heuristic
10:     calculate fitness
11:     new pBest and new gBest (or lBest)?
12:   end for
13: until termination criterion reached
```

There are now 6 actions used to determine the new position. The probability of occurrence was heuristically determined using prior tests. The actions are:

1. 0.05%: Exchange two job assignments (without rejection): Two jobs are randomly chosen and the currently assigned technicians are identified. Then, the technicians exchange job assignments, with the new job placed at the same spot in the respective technician's permutation as the old job. Qualifications are taken into account during this action by possibly repeating the choice. This step may not be rejected by the particle even if it leads to worse fitness of the solution.
2. 24.95%: Exchange two job assignments (with possible rejection): The procedure is analogous to the above action, the difference being that the particle does not carry it out if fitness would worsen.
3. 0.25%: Move (without rejection): One technician is randomly chosen and the last job in his permutation is moved to the end of the permutation of another qualified technician, if available. If no other qualified technician can be found, nothing occurs. The particle may not reject the action even if fitness suffers.
4. 14.75%: Move (with possible rejection): The process is analogous to action 3, the difference being that the particle may reject the action if it would worsen fitness.
5. 20%: Insert a value from pBest: Choose a random technician from pBest and a random job within that technician's permutation. Insert that job into the new particle position at the same location as in pBest. If another job is already located at that position, postpone the rest of the jobs by one slot in order to make room. If the job is isolated within the permutation, it is shifted until it borders an occupied slot. This action may be rejected by the particle if it deteriorates fitness.
6. 40%: Insert a value from gBest: This action is analogous to action 5 but gBest (or lBest in other topologies) is used instead of pBest.

Qualifications are especially critical in the British Telecom problem and compliance represents a hard constraint. Therefore, a solution is only valid without qualification errors. In order to remove qualification errors that do occasionally arise, they are repaired using a heuristic approach. This repair heuristic searches for violations and assigns an incorrectly allocated job to a randomly determined qualified employee, where the job is inserted in an appropriate time slot. If a gap has arisen in the job sequence for the unqualified technician it is closed by shifting jobs so that the solution becomes valid.

4.2 Initialization of PSO

An initial solution for PSO is created by applying a constructive method. More specifically, a solution is constructed that respects the BT problem's hard constraints, such as availability of technicians and required qualifications. In addition, the allocation of assignments to technicians is based on capacity still available and the distance of a job from the service center. The exact distances from one job to the next job cannot be applied to the calculation of the remaining capacity because the optimum job order is not yet known. Capacity is therefore determined approximately using the regional allocation of jobs to service centers. The initialization procedure is shown in algorithm 3.

5 Results and Discussion

The choice of an appropriate termination criterion for a solution method strongly influences result quality and required CPU time. The experiments discussed here uniformly used the

Algorithm 3 Initialization

```
set the capacity Cap_e of technician e to 0
repeat
  choose a random job j
  write all technicians qualified for job j in list Quali
  calculate the travel time ct_e for each technician from his or her service
  center i to job j
  while Quali is not empty do
    choose the technician e with the shortest distance ct_e to job j
    remove technician e from Quali
    calculate the completion time rd_j by technician e
    if Cap_e + rd_j < 8 hours then
      add job j to the end of the permutation for technician e
      remove all technicians from Quali
      update Cap_e
    end if
  end while
  if job cannot be assigned then
    choose a technician e who is qualified for job j
    add job j to the end of the permutation for technician e
    update Cap_e
  end if
until all jobs have been assigned
```

number of objective function evaluations (set to 20 million according to pre-tests) as the termination criterion. This offers excellent comparability of the solution methods. Moreover, it will allow for fair comparisons with results of others in the future, as CPU-time is less meaningful in the light of ever increasing computing power. All results using PSO are based on 30 independent runs. They were performed on a PC with an Intel Core Quad 4x1.66 GHz with 4 GB of RAM. When comparing our results to the literature, we will focus on solution quality not speed. CPU-time would be hard to compare, as hardware is always very different. In some cases, CPU-requirements were not even published. More importantly, computing time is not a significant limiting factor in the present problem. It can be assumed that authors who previously worked on the BT-problem terminated their runs when no further improvement appeared possible in reasonable time.

Interpretation of competing solution methods from the literature as introduced in section 3 will only be done briefly here. The respective publications can be referred to for more comprehensive information. Only the mimimum value (best solution found) is known for these methods, which reduces comparability. An indication of mean values and the number of replications could not be found. For GA, GA + R (R=repair heuristic) and SA, CPU time is not available. In general, it can be noted that agents and CLP are significantly faster than metaheuristic methods, such as GA, SA and PSO. Table 3 shows the results of PSO with different neighbourhood topologies as well as the methods listed in section 3.

The first set of results was published for SA. Here, a value of 21,050 error points was achieved. In comparison, the results of distributed GA with repair (22,570) and without repair (23,790) are significantly worse. The repair heuristic, however, does improve results for the GA. Azarmi and Abdul-Hameed tested the two CLP variants CLP-CG + R (21,292) and CLP-TG + R (22,241), both using a repair heuristic. Using these solution approaches, more jobs were able to be assigned than in GA and GA + R because the repair in the former method focuses on reducing the number of incomplete jobs. For GA and GA + R on the other hand, the focus of the repair is on the reduction of total error points.

Table 3 Results for the original BT problem (BT_Mod-250-118). Results that improve the previous best known solution (generated by FLS) are bold. New best solution is set bold and underlined. If solutions are repaired w.r.t hard constraints this is indicated with +R. If a solution method uses forward checking of potential new solutions this is indicated with FC. Results for PSO are based on 30 independent runs each.

Method	Error			Travel costs	Error open jobs	Number open jobs	CPU time in sec.
	⌀	minimum	std. dev.				
SA [2]	-	21,050	-	4,390.0	16,660.0	56.0	-
GA [26]	-	23,790	-	-	-	67.0	-
GA + R [26]	-	22,570	-	-	-	54.0	-
CLP TG + R [1]	-	22,241	-	5,269.0	16,972.0	48.0	600
CLP CG + R [1]	-	21,292	-	4,902.0	16,390.0	53.0	600
CLP SD + FC [38]	-	20,981	-	4,716.0	16,220.0	54.0	97
LS [36]	-	20,788	-	4,604.0	16,184.0	50.0	20,056
FLS [36]	-	20,732	-	4,608.0	16,124.0	49.0	1,242
Fast GLS [36]	-	20,433	-	4,707.0	15,726.0	48.0	9,183
PSO (10) gBest + R	20,585.5	**20,371**	164.8	4,221.5	16,363.9	55.9	10,306
PSO (200) gBest + R	21,184.1	20,958	112.2	4,374.4	16,809.7	60.2	10,607
PSO (10) Wheel + R	20,637.7	**20,340**	162.2	4,169.5	16,468.2	56.6	10,668
PSO (10) Circle + R	20,505.4	**20,273**	112.8	4,185.0	16,320.4	55.0	10,572
PSO (10) lBest + R	20,435.9	<u>**20,193**</u>	130.9	4,168.0	16,267.8	54.5	10,615

Yang tested various CLP approaches, the best variant of which is shown in table 3 – the variant with foward-checking (FC) and the selection heuristic same direction first (SD) for technician order (CLP SD + FC). There, 20,981 error points were achieved, which was the best value up to that point. Using significantly more CPU time, Tsang and Voudouris were able to produce even better results. They implemented LS, FLS and fast GLS, which all produced better results than previous solution methods. Fast GLS yielded the best error point value of 20,433 which essentially marked the final achievement on this practical problem so far.

Using PSO, a new best solution with an objective function value of 20,193 was found. Comparing different swarm sizes, it became evident that a gBest topology with a swarm of 10 particles performs significantly better than a larger swarm with 200 particles. This result is consistent with our previous results with a similar modified PSO algorithm for a logistics problem [28], [17] [18]. The swarm requires a high number of iterations to get from the initial solution to a very good solution. Using the number of fitness calculations as termination criterion means that significantly more iterations can be performed on a small swarm size than on a large one. The advantage of a large swarm that more knowledge is available on the solution space in each iteration is apparently less important.

Based on this insight, all further experiments for the wheel, lBest (k=4) and circle topologies were performed with 10 particles. With respect to the minimum objective function

Table 4 t-test results (one-tailed) for pairwise comparison of neighbourhood topologies in PSO

H_1	T	df	signi-ficance H_0	95% confidence interval of differences lower	upper
PSO (10) lBest + R < PSO (10) Circle + R	-2,205	58	0.016	-122,254	-16,812
PSO (10) lBest + R < PSO (10) gBest + R	-3,893	58	< 0.001	-213,832	-85,368
PSO (10) lBest + R < PSO (10) Wheel + R	-5,303	58	< 0.001	-265,415	-138,185

value, the lBest neighbourhood performs best, followed by circle, wheel and gBest. Considering the mean values, lBest again significantly outperforms the other topologies, followed by circle, gBest and wheel. Results of respective t-tests can be found in table 4. It can therefore be worthwhile not to immediately distribute information to all particles. Using smaller neighbourhoods helps to avoid premature convergence to a local optimum on this problem. This result concurs with many statements found in the literature and also with the basic idea behind neighbourhood topologies [21], [10]. However, it contradicts experience gained using a similarly modified PSO [17] on a different combinatorial problem. There, gBest almost always yielded the best results because good solutions were very rare in the solution space. If these good solutions were not almost immediately passed on to all particles, there was a danger of them being lost. The British Telecom problem does not seem to suffer from that effect in any great amount.

With respect to CPU time, PSO requires roughly 3 hours for one run. Since CPU time is not a limiting factor in this type of application, the CPU requirements of PSO can be regarded sufficient, considering the final solution quality produced.

6 Conclusions and Outlook

The British Telecom problem is derived from a practical situation and has been intensively analysed in the literature for almost two decades. In addition to the previously tested methods, this work assessed an adapted hybrid form of particle swarm optimization that integrates an initialization and a repair heuristic. Moreover, various neighbourhood topologies and swarm sizes of PSO were tested.

PSO with a small population size of 10 particles produced better results than the previous best known solution, independent of the neighbourhood topology. Among the neighbourhoods, lBest performed best, which is in line with the finding of others in the literature. Taken together with the results presented for a logistics problem elsewhere [18], there is a strong indication that hybridising a metaheuristic with a problem-specific repair heuristic is a useful approach of resolving the conflict between domain-specific characteristics of a real-world problem and the desire to employ a generic optimisation technique, at least in the domain of workforce management. Moreover, it seems to pay off to use available knowledge in the initialization phase of PSO in order to respect the hard problem constraints right from the start instead of having more diversity using random initialization.

Even the best schedules contain technicians who are not assigned to any jobs. This is due to the focus on the reduction of travel costs and unfinished jobs in this particular optimization problem. However, it is worth mentioning that a great economic potential also lies in the reduction of employee idleness.

The BT-problem only considers coordinates of the service centers and job location positions. Related real-world applications might require to use actual distances instead. Another interesting extension would be to view the BT-problem as a multiobjective optimization problem where total travel distance and the number of unassigned jobs are simultaneously optimized. This would require different solution approaches that can effectively search for the Pareto front.

In the current planning process technicians are allocated a shift model in a previous step. Only then does the planning problem discussed here begin. The merging of both planning phases would obviously render a lot of further potential for improvement.

References

1. Azarmi N., AbdulHameed W., Workforce scheduling with constraint logic programming, In: BT Technology Journal, 13(1):81–94, 1995
2. Baker S., Applying simulated annealing to the workforce management problem, In: ISR, British Telecom Laboratories, Martlesham Heath, Ipswich, 1993
3. Bartholdi J. J., A Guaranteed-Accuracy Round-off Algorithm for Cyclic Scheduling and Set Covering, In: Operations Research, 29(3):501–510, 1981
4. British Telecom Laboratories, 250-118 data set of WMS problem, http://dces.essex.ac.uk/CSP/WFS, last visited: 08.01.2010
5. Brusco M. J., Jacobs L. W., A Simulated Annealing Approach to the Solution of Flexible Labour Scheduling Problems, In: Journal of the Operational Research Society, 44:1191–1200, 1993
6. Brodersen O. B., Eignung schwarmintelligenter Verfahren fr die betriebliche Entscheidungsunterstützung, Cuvillier, 2008
7. Chu S.-C., Chen Y.-T., Ho J.-H., Timetable Scheduling Using Particle Swarm Optimization, Proceedings of the First International Conference on Innovative Computing, Information and Control (ICICIC 2006), 3:324–327, Beijing, 2006
8. Cooper T. B., Kingston J. H., The Complexity of Timetable Construction Problems, In: Burke E. K., Ross P. (eds.), Practice and Theory of Automated Timetabling, First International Conference (PATAT 1995), Vol. 1153, LNCS, 283–295, Edinburgh, U.K., Selected Papers, Berlin et al.: Springer, 1996
9. Cung V.-D., Briant O,Challenge ROADEF 2007. Technicians and Interventions Scheduling for Telecommunications; http://www.g-scop.inpg.fr/ ChallengeROADEF2007
10. Czogalla J., Fink A., Particle Swarm Topologies for Resource Constrained Project Scheduling, In: Nature Inspired Cooperative Strategies for Optimization (NICSO 2008), 236:61–73, Berlin: Springer, 2009
11. Dutot P.-F., Laugier A., Bustos A.-M., Technicians and Interterventions scheduling for Telecommunications, 2006
12. Easton F. F., Rossin D. F., Equivalent Alternate Solutions for the Tour Scheduling Problem, In: Decision Sciences, 22:985–1007, 1991
13. Ernst A. T. et al., An Annotated Bibliography of Personnel Scheduling and Rostering, In: Annals of Operations Research, 127:21–144, 2004
14. Fukuyama, Y. (2003). Fundamentals of Particle Swarm Optimization Techniques. In Lee, K.Y., El-Sharkawi, M.A. (Eds.). Modern Heuristic Optimization Techniques with Applications to Power Systems (pp. 24–51). Wiley-IEEE Press.
15. Garey M. R., Johnson D. S., Computers and Intractability. A Guide to the Theory of NP-Completeness, In: Series of Books in the Mathematical Sciences. Freeman, 1979
16. Günther, M. (2011). Hochflexibles Workforce Management: Herausforderungen und Lösungsverfahren, Dissertation (in German), TU Ilmenau 2011
17. Günther M., Nissen V., A Comparison of Neighbourhood Topologies for Staff Scheduling With Particle Swarm Optimisation, In: Mertsching B., Hund M., Zaheer A. (eds.): KI 2009: Advances in Artificial Intelligence, LNCS 5803:185–192, Berlin: Springer, 2009
18. Günther M., Nissen V., A Comparison of Three Heuristics on a Practical Case of Sub-Daily Staff Scheduling. In: Annals of Operations Research, 201:DOI 10.1007/s10479-012-1259-2, 2012
19. Kennedy J., Eberhart R. C., Particle Swarm Optimization, Proceedings of the IEEE International Conference on Neural Networks, 1942–1948, Perth, Australia, IEEE Service Center, 1995
20. Kennedy J., Eberhart R. C., Shi Y., Swarm Intelligence, San Francisco: Kaufmann, 2001
21. Kennedy J., Mendes R., Population structure and particle swarm performance, In: Proceedings of the IEEE Congress on Evolutionary Computation (CEC 2002), 2:1671–1676, Honolulu, USA, 2002
22. Kliem R. L., Anderson H. B., Styles and Their Impact on Project Management Results, In: Project Management Journal, 27(1):41–50, 1996

23. Kokkoras F., D-WMS: A Distributed Version of a CLP-based Workforce Management System, Computer Science Department, University of Bristol, UK, 1996
24. Kragelund L., Kabel T., Employee Timetabling. An Empirical Study of Solving Real Life Multi-Objective Combinatorial Optimisation Problems by means of Constraint-Based Heuristic Search Methods, Master's Thesis in Computer Science, Department of Computer Science, University of Aarhus, 1998
25. Mendes R., Population Topologies and Their Influence in Particle Swarm Performance, PhD Thesis, Departamento de Informtica, Escola de Engenharia, Universidade do Minho, 2004
26. Muller C., Magill E. H., Prosser P., Smith D. G., Distributed genetic algorithms for resource allocation, In: Dorn J., Fröschl, K. (eds.), Scheduling of production processes, 70–78, Ellis Horwood, 1993
27. Naveh Y. et al. Workforce optimization: identification and assignment of professional workers using constraint programming, In: IBM Journal of Research and Development archive, 51(3):263–279, 2007
28. Nissen, V., Günther, M. (2009). Staff Scheduling with Particle Swarm Optimization and Evolution Strategies. In Cotta, C., Cowling, P. (Eds.). Proceedings of EvoCOP 2009 (pp. 228–239). LNCS 5482, Berlin: Springer.
29. Parsopoulos, K.E., Vrahatis, M.N. (2002). Recent Approaches to Global Optimization Problems through Particle Swarm Optimization. Natural Computing, 1, 235–306.
30. Sakellariou I., Kokkoras F., Vlahavas I., Applying a Distributed CLP Platform to a Workforce Management Problem, In: Proceedings of the 12th Conference on Intelligent Systems Application to Power Systems (ISAP 2003), Lemnos, Griechenland, 2003
31. Tasgetiren, M.F., Sevkli, M., Liang, Y.C., Gencyilmaz G. (2004). Particle Swarm Optimization Algorithm for Single Machine total Weighted Tardiness Problem. In Proceedings of the Congress on Evolutionary Computation (CEC) 2004 (pp. 1412–1419). IEEE.
32. Tien J. M., Kamiyama A., Manpower Scheduling Algorithms, In: SIAM Review, 24(3):275–287, 1982
33. Tsang E. P. K. et al., Retractable Contract Network for Distributed Scheduling, In: Proceedings of the 2nd Multidisciplinary International Conference on Scheduling: Theory & Applications (MISTA 2005), 485–500, New York, USA, 2005
34. Tsang E. P. K. et al., Retractable contract network for empowerment in workforce scheduling, In: Special Issue on Negotiation Mechanisms, Multiagent and Grid Systems, 4(1):25–44, 2008
35. Tsang E. P. K. et al., Multi-Agent Systems for Staff Empowerment, In: Voudouris C., Owusu G., Dorne R., Lesaint D. (eds.), Service Chain Management. Technology Innovation for the Service Business, 236–274, Springer, 2008
36. Tsang, E. P. K., Voudouris C., Fast Local Search and Guided Local Search and Their Application to British Telecom's Workforce Scheduling Problem, In: Operations Research Letters, 29(3):119–127, 1997
37. Veeramachaneni, K. (2003). Optimization Using Particle Swarm with Near Neighbor Interactions. In Proceedings of the Genetic and Evolutionary Computation Conference (GECCO) 2003 (pp. 110–121). LNCS 2723, Berlin: Springer.
38. Yang R., Solving a Workforce Management Problem with Constraint Programming, In: Proceedings of the 2nd International Conference on the Practical Application of Constraint Technology (PACT 1996), 373–387, London, UK, 1996