Combining Parallel Coordinates with Multi-Objective Evolutionary Algorithms in a Real-World Optimisation Problem

Neil Urquhart
Edinburgh Napier University
10 Colinton Road
Edinburgh, United Kingdom EH10 5DT
n.urquhart@napier.ac.uk

ABSTRACT

Optimisation problems based upon real-world instances often contain many objectives. Many existing Multi-Objective Evolutionary Algorithm techniques return a set of solutions from which the user must make a final selection; typically such a set of solutions may take the form of a non-dominated set. The size of such fronts, especially for larger numbers of objectives, can make it difficult for the user to make a selection of the final solution. This paper outlines an initial investigation into combining elements of Parallel Coordinate plots with multi-objective evolutionary algorithms to allow the user to specify solution areas of interest prior to executing the algorithm. The algorithm encourages the evolution of solutions in these areas through selection pressure. The user is presented with one solution from each area on a Parallel Coordinates plot allowing a simple, informed decision as to the solution to be chosen. This paper uses a Workforce Scheduling and Routing Problem (WSRP) to demonstrate the approach. The WSRP formulation used was previously cited in literature as a multi-objective problem, we formulate it as a 5 objective problem. Our initial results suggest that this approach has potential and is worth investigating further.

KEYWORDS

Evolutionary Algorithms, Transportation, Multi-Objective Optimisation, Real-World Problems

ACM Reference format:
Neil Urquhart. 2017. Combining Parallel Coordinates with Multi-Objective Evolutionary Algorithms in a Real-World Optimisation Problem. In *Proceedings of GECCO '17 Companion, Berlin, Germany, July 15-19, 2017,* 6 pages.
DOI: http://dx.doi.org/10.1145/3067695.3082485

1 INTRODUCTION AND MOTIVATION

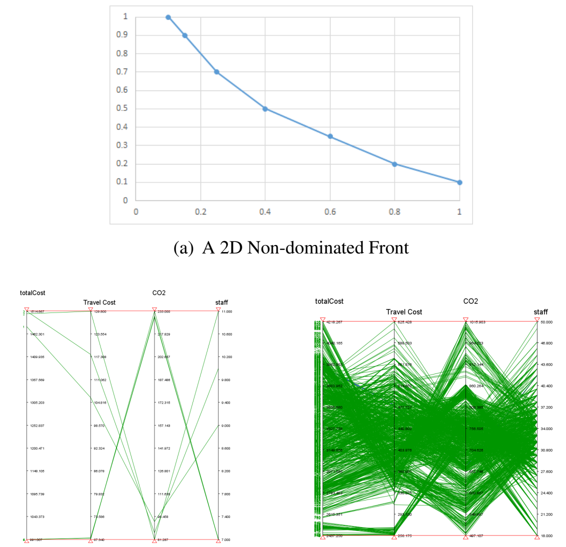
Many real-world optimisation problems are multi-dimensional, having two or more objectives to address. Multiple-Objective Evolutionary Algorithms (MOEAs) can handle multi-objectives using techniques such as Pareto dominance to compare and identify potentially useful solutions. MOEAs such as NSGA-II [3] have made use of Pareto dominance and distance techniques to create a non-dominated set of solutions which are presented to the user, leaving final choice up to user. Previous research by the authors has shown that Pareto Fronts may contain a large number of solutions, which makes the task of evaluating them and choosing a final solution difficult. When visualising bi-objective problems, they may be plotted in two dimensions (see figure 1(a)), tri-objective problems can be visualised using software tools that plot solutions using the three objectives as axis'. Many-objective problems can be visualised using Parallel Coordinates [11], which represent each objective as a vertical axis and each solution as a polyline, which intersects each axis at the appropriate values, see figure 1(b). When the size of the non-dominated front grows, the Parallel Coordinate plots become crowded and difficult to interpret (figure 1(c)). In addition to using Parallel Coordinates to visualise the output of an algorithm, this paper proposes that they can also be used to specify areas of interest to the user and then guide the search, rather than the user being presented with a potentially large Pareto set of solutions from which to choose a final solution. We explore these concepts using a multi-dimensional Workforce Scheduling and Routing Problem (WSRP).

2 PREVIOUS WORK

This paper formulates the WSRP as a multi-objective problem, with cost reduction, CO_2 reduction and car use reduction as the criterion, using the problem formulation and instances from [16, 17] with the addition of extra instances as described in section 3.2. For a comprehensive introduction to the WSRP and an overview of the latest developments, the reader is directed towards [2], [1] and [10]. A number of previous researchers have dealt with problems relating to the scheduling and routing of workforces; [15] deals with home care scheduling, [13] with security personnel scheduling and [8] with technician scheduling.

Previous approaches to Multi-Objective Evolutionary Algorithms (MOEAs) include SEAMO [18], SPEA2 [19] and NSGA-II [3]. SEAMO (Simple Evolutionary Algorithm for Multi-objective Optimisation) [18] creates a random population of solutions and records the best values so far for each objective. New solutions are created from two parents and a mutation is applied. If the new solution improves on the best-so-far for any of the objectives it replaces one of its parents within the population; if it dominates one of the parents, then it replaces that parent. Within each generation, each member is

GECCO '17 Companion, Berlin, Germany
© 2017 ACM. 978-1-4503-4939-0/17/07. . . $15.00
DOI: http://dx.doi.org/10.1145/3067695.3082485



(a) A 2D Non-dominated Front

(b) A small 4D Non-dominated Front (c) A large 4D Non-dominated Front

Figure 1: Examples of non-dominated fronts.

selected to become a parent, with the second parent being selected at random. SPEA2 (Strength Pareto Evolutionary Algorithm for Multi-objective Optimisation) was first introduced by [19]. SPEA2 maintains an archive of solutions in addition to the main population. The archive contains the non-dominated set of individuals from the main population. Within each generation, a pool of parents is selected by tournament selection from the population, and a replacement population of children is created by crossover and mutation. The Non-dominated Sorting Genetic Algorithm II (NSGA-II) [3] ranks solutions according to dominance, but also assigns a *crowding* distance to solutions to highlight areas within the front that are highly populated. This results in fronts that are more likely to have solutions distributed evenly between the extreme solutions.

Parallel Coordinates were developed by Inselberg (see [11] for a comprehensive guide to their use), to allow visualisation in many dimensions. Each dimension (problem objective) is represented by a vertical axis. In order to ensure that the axis are all at the same height, the objectives must be normalised. Each solution may then be plotted by means of a polyline that intersects each axis at the point on the scale that represents the solutions' value on that objective. By examining the patterns formed by the polylines between the objective axis it becomes possible to visualise the relationship between objectives. The ability of a parallel coordinate plot to represent multiple dimensions in a 2D space makes it a useful technique when discussing results in non-interactive media such as a printed page. Parallel coordinate plots can be made interactive the with the user being allowed to reorder axis in order to explore the relationships between different objectives. A technique known as *brushing* may be used to select areas of interest on one or more axis, and those polylines that intersect the brushed areas are then highlighted.

3 METHODOLOGY

3.1 A Mobile Workforce Scheduling and Routing Problem

As a testbed for our algorithm we utilise a Workforce Scheduling and Routing Problem (WSRP) which was originally described in [16] and [17]. This particular formulation uses real-world geographical and transport data, to calculate journey times, CO_2 values and associated financial costs. The problem formulation allows each worker to utilise public transport or car based travel with consequent effects on CO_2 and other objectives. Car journey paths and times are determined using the GraphHopper library [12] based on OpenStreetMap [7, 9] data, public transport times and costs are derived from the Transport for London (TfL) API [6]

The costs associated with each solution are calculated as follows:

- CO2 emission: for cars we adopt the WebTAG UK transport assessment mode, for public transport we use the emission factors published by TfL [5].
- Financial cost: we calculate staff costs based on [14] and transportation costs based on car travel costs in [4] and public transport costs based on TfL fare costs.
- Staff: the number of staff required within the solution.

3.2 Problem Instances

In this paper, twenty problem instances based around four geographical scenarios as follows:

- London (lon): 60 visits at randomly selected locations, within London, each visit being within a radius of approximately 16 miles of the agency headquarters.
- BigLondon (blon): 110 visits within a radius of approximately 23 miles of the agency headquarters.
- Offset: 110 visits set out as per BigLondon, but with the headquarters located on the outside of the visit cluster.
- Cluster: 60 visits grouped in six clusters. The clusters are randomly distributed around the headquarters.

For each scenario, we consider five problem instances:

- 1: all visits have a time window of 8 hrs. i.e. the visits can start any time during the working hours of the employees.
- 2, 4, 8: Each visit has a 4, 2 or 1 hr. time window allocated. The beginning of the time window is chosen randomly.
- Rnd: The duration of the time window of each visit is randomly chosen among 1, 2, 4, or 8 hrs.

3.3 Algorithm

The algorithm is based upon conventional EA principles and utilises the representation, crossover, mutation and fitness operators described in [16, 17]. Rather than produce a non-dominated front of solutions, we allow the user to specify expected performances in each of the objectives. Each objective may have a higher and lower performance, set to 0, 0.3, 0.7 and 1 (indicating low, medium or high) within the axis. Figure 2 shows six highlighted areas of interest. These were selected at random to represent potential user preferences. As the performance cannot be predicted these classifications are not based on absolute values (e.g. CO_2) but on the normalised values within the ranges found within the current population. The population is normalised, so that each objective within each solution is presented in the range 0..1, where a value of 1 represents the highest value for that objective within the population and 0 represents the lowest value for that objective within the population. The four objectives that are considered as Travel Cost, CO_2, number of staff and total working time (the sum of all the staff shift lengths in the solution). Time and travel cost were added to the original formulation in order to increase the dimensionality of the problem. In order to ensure that only one solution is presented for each of the six classes total financial cost is used as an additional objective in order to determine the "best" solution in each class. This ensures that only one solution is presented within each user specified criterion class.

The algorithm used is shown in Algorithm 1. The parameters used to obtain the results presented may be seen in table 1, the parameter values were established through past experience and empirical studies. Lines 1 to 4 setup the population with random solutions, which are evaluated, the results of each objective are normalised and each member of the population allocated a classification based on the user preferences specified. If the solution fits into one of the highlighted areas (see figure 2) then it is allocated that classification. Each classification is checked in the order that the user specified them in, if a solution fits into more than one classification (a user may potentially setup overlapping classifications) then it is allocated to the first that it fits. Solutions that fit none of the user-specified classifications are marked as unclassified. Thus the current population is classified according to the preferences set by the user.

The main generational loop is contained in lines 5-23 and it will loop until the specified number of solutions have been created and evaluated. The selectParent() function selects a member of the population at random, but with a bias towards individuals with a classification. If after 10 attempts an individual has not been selected with a classification then an unclassified individual is used. The second parent is selected in a similar manner, but the bias is towards selecting a parent with the same classification as the first parent. The new child solution is created randomly by cloning one parent or by recombination from both parents, as determined by the crossover pressure.

After a child population has been created, each member of the child population is added to the main population if the solution does not already exist in the main population. An individual is selected to be replaced (line 17) using the getRIP() function. The getRIP() function selects a solution at random, but with a bias towards unclassified solutions; finally, the child solution replaces the RIP solution if it has a lower financial cost than the solution that it is replacing.

After each of the children have been considered, the objectives of each member of the population are normalised and the classification of each member is determined. If a solution is added to the population with a new lowest value in an objective, then the normalised values for that objective will change. This could potentially lead to solutions changing classification as the evolution progresses. After the set number of evaluations have taken place the individual with the lowest financial cost for each class is printed (line 24).

Algorithm 1: runGA()

```
   // Initialise Population
1  setup_population();
2  evaluate_population();
3  normalise_population();
4  classify_population();
5  while evals<MAX_EVALS do
       // Execute until a set number of
          solutions have been evaluated
6      for ch =0; c <CHILDREN; ch++ do
           // Create children, either crossover
              from two parents or clone from
              one.  Add a mutation to every
              child.
7          parent1 = selectParent();
8          parent2 = selectParent(parent1.classification);
9          if rnd() <XO_PRESSURE then
10             child = new Individual(parent1,parent2);
11         else
12             child = parent1.clone();
13         mutate(child);
14         evaluate(child);
15         children.add(child);
       // Add children to the main population,
          replacing weaker members
16     if unique(child,population) then
17         rip = getRIP(population);
18         if rip.finCost() >child.finCost() then
19             population.remove(rip);
20             population.add(child);
21     normalse_population();
22     classify_population();
23 for class = 0 to MAX_CLASSIFICATIONS do
24     printBest(class);
```

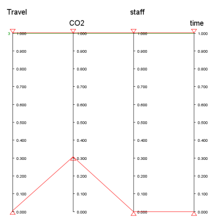
Table 1: Algorithm Parameters

Parameter	Value
Population Size	500
Children	20
Crossover Pressure	0.8
Maximum Evaluations	1,000,000
Maximum Classifications	6 (see fig 2)

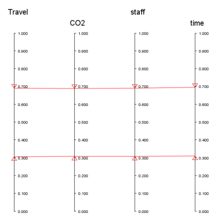
4 RESULTS

4.1 Comparison With Previous Results

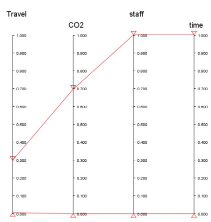
For each problem instance results were produced using the Portfolio approach described in [16]. The portfolio approach used a group of solvers based upon differing configurations of NSGA-II and SPEA algorithms. Each solver produces a front, which is then combined into a single front of non-dominated solutions. Due to the stochastic nature of MOEAs, each solver is executed 10 times and the results combined into a single front. A summary of the results obtained using the portfolio are presented in table 2, the values for the three objectives are the average value across all solutions within the front. Note the large sizes of the fronts produced, in many cases they are too large to allow easy selection of a final solution by the end-user,



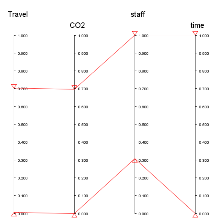
(a) Class 1



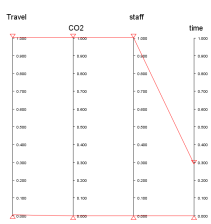
(b) Class 2



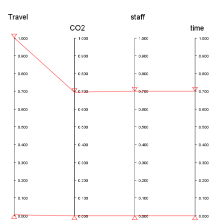
(c) Class 3



(d) Class4



(e) Class5



(f) Class6

Figure 2: The exemplar solution classifications used within this paper.

samples of the Parallel Coordinate plots produced may be see in figure 3.

A summary of the results obtained may be seen in table 4; the values shown are the averages obtained over 10 runs. As the total cost, *CO2* and number of staff are also objectives within the benchmark, the % of their benchmark values are shown (values $<$100% indicate an improvement and $>$100 % indicate a worse value). The results are grouped by the results classes specified by the user (see figure 2) the final result chosen being the result with the lowest financial cost. If we consider results of class 1 we note that the average financial cost is reduced by up to 20 %. We also note that CO_2 does not show a reduction for any of the problem instances, but reference to figure 2(a) shows that the user has not specified low CO_2 values in problems of this class. Problems of class 2 (see figure 2(b)) are specified to exist within the medium range (0.3-0.7) for each objective. Table 4 shows that for 11 instances no results could be found. When the user is specifying the classes there will always be a risk that they will specify criteria, which cannot be satisfied. For the problem instances where class 2 solutions could be found they represent considerable improvements in terms of financial cost, with a number also showing an improvement in CO_2.

Table 2: Benchmark Results. These results are averaged across the front created using the portfolio.

Problem	totalcost	co2	Staff	Front Size
blon-1	2533.780651	458.44	15	76
blon-2	2729.327875	501.85	19	91
blon-4	2721.989397	584.49	21	126
blon-8	2630.28358	616.28	22	178
blon-rnd	2761.534234	533.21	19	145
cluster-1	1192.113533	197.47	7	57
cluster-2	1302.806602	240.61	9	100
cluster-4	1356.736786	276.71	11	131
cluster-8	1402.91447	288.35	12	100
cluster-rnd	1275.663808	246.64	9	131
lon-1	1028.970327	143.85	6	98
lon-2	1166.20092	149.31	8	136
lon-4	1248.726196	175.73	11	195
lon-8	1244.144208	227.65	13	173
lon-rnd	1135.828493	183.35	9	152
offset-1	2475.971286	675.50	14	53
offset-2	2857.62541	782.65	19	126
offset-4	2937.527205	901.60	22	122
offset-8	2874.870618	982.35	21	101
offset-rnd	2664.177344	785.44	17	51

Class 3 solutions require minimum travel costs and CO_2 costs in the medium and minimum categories (see figure 2(c)). Once more we note a reduction in the average total cost, an increase in CO_2 and a reduction in staff in some instances. When considering class 4 solutions we typically note a reduction in overall cost from the benchmark and a reduction in staff for all but 3 instances. Within the class 5 solution instances found, we note the decrease in total cost, along with a decrease in CO_2 for 8 instances and a reduction in staff in 17 instances.

The only class of solution where an increase in total cost was noted were those solutions of class 6, where significant increases in total cost, CO_2 and staff were all noted.

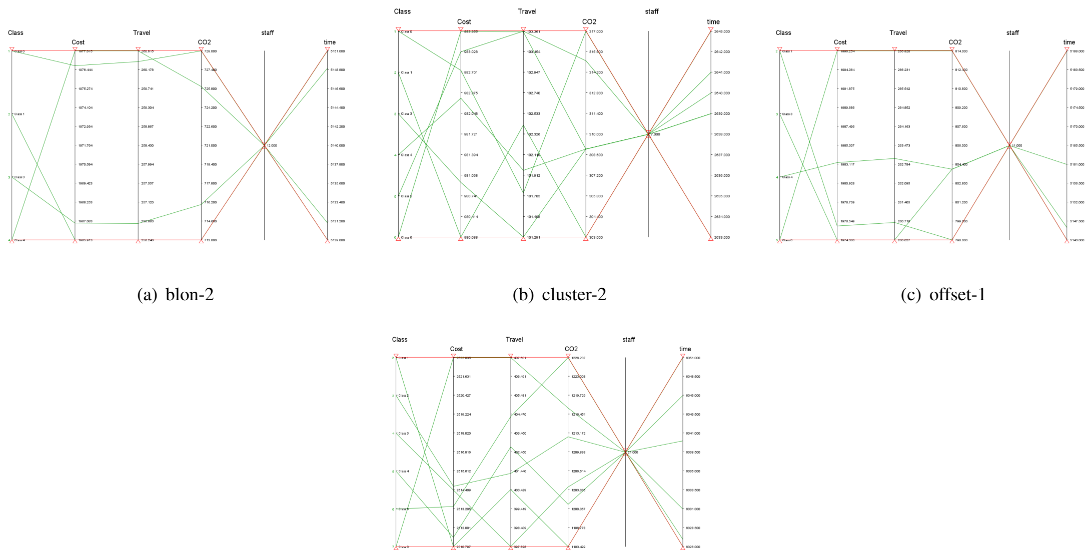
4.2 Distribution of Solution Classes

As we have defined 6 possible solution classes of interest to the user it is useful to note how membership of these classes is distributed across the 500 members of the population. Table 3 shows the average membership of each class across the 10 runs of each problem instance. We note that as the user is free to specify the desired objectives in any given class there is no guarantee that the algorithm will generate any solutions that fall into that class. The only class for which difficulties were experienced in generating solutions for was class 2.

There may also exist some solutions which do not fit into any specific classes, once again this is dependent on the choices made by the user when setting up the classes. We note from the results that for every instance we have some members of the population who cannot be classified, which on average range from 7.6 % to 21.6 %.

5 CONCLUSIONS AND FURTHER WORK

This paper presents an initial investigation into combining the principles of Parallel Coordinate plots with a basic MOEA. One of the major difficulties of the Pareto-based MOEAs used in previous work was their tendency to produce large fronts, thus the user is left with the task of selecting the final solution. Even visualising such fronts using Parallel Coordinates does not necessarily ease the problem as the resulting plot can be difficult to interpret due to the number of intersecting polylines contained. We propose a solution that allows the user to specify which areas within each coordinate interest them; each of these areas defines a solution class (see figure 2). Only one



(a) blon-2 (b) cluster-2 (c) offset-1

(d) offset-8

Figure 3: Examples of the Parallel Coordinate plots produced by the algorithm, with one solution selected in each class.

Table 3: Distribution of Result Classes

	Unclassified	1	2	3	4	5	6
blon-1	14.54%	15.76%	3.32%	15.18%	46.42%	1.38%	3.40%
blon-2	16.26%	17.30%	0.00%	12.54%	51.60%	0.90%	1.40%
blon-4	21.60%	22.38%	9.38%	18.38%	19.12%	6.00%	3.14%
blon-8	9.30%	24.26%	0.00%	14.18%	38.44%	11.96%	1.86%
blon-rnd	9.72%	29.52%	22.74%	6.76%	14.22%	10.56%	6.48%
lon-1	11.86%	20.22%	0.00%	10.38%	56.18%	0.10%	1.26%
lon-2	8.58%	29.92%	0.00%	20.18%	37.00%	3.10%	1.22%
lon-4	15.74%	16.80%	0.00%	14.72%	42.56%	6.28%	3.90%
lon-8	20.80%	18.82%	26.06%	9.08%	10.56%	6.78%	7.90%
lon-rnd	10.52%	22.80%	6.64%	11.04%	42.00%	0.58%	6.42%
cluster-1	12.30%	14.00%	0.00%	14.96%	50.60%	3.70%	4.44%
cluster-2	9.66%	36.06%	0.00%	15.94%	32.14%	2.26%	3.94%
cluster-4	9.40%	23.56%	0.00%	12.10%	40.86%	6.38%	7.70%
cluster-8	7.58%	30.54%	9.96%	5.38%	23.00%	18.38%	5.16%
cluster-rnd	8.86%	33.22%	0.00%	13.76%	35.08%	4.82%	4.26%
offset-1	10.40%	19.48%	0.00%	11.26%	52.36%	1.04%	5.46%
offset-2	17.72%	21.22%	4.16%	6.06%	41.52%	1.26%	8.06%
offset-4	13.24%	22.72%	4.56%	11.76%	37.64%	6.42%	3.66%
offset-8	19.32%	18.30%	9.88%	7.18%	24.02%	14.38%	6.92%
offset-rnd	8.46%	26.50%	0.00%	13.56%	41.86%	1.92%	7.70%

solution within each class is presented to the user, thus the number of solutions presented to the user is equal to the number of solution classes. Of the six exemplar solutions classes presented, we note that in only one case (class 2) is the algorithm unable to find solutions. It may be significant that class 2 is the only classification which does not require a normalised value of 0 in at least one objective. When comparing to the benchmark solutions, we note that the new algorithm consistently finds solutions with a lower overall cost and in some cases lower $C0_2$ and staff numbers. Although this initial version of the algorithm does not show large improvements, it finds enough improved solutions to suggest that the principle of using a MOEA combined with the principles of parallel coordinates is a useful concept.

Further development includes the possibilities of using an alternative means to select the final solution in each class. Parallel coordinates allow a useful means of the user specifying the sorts of solution that they are interested in, and the ability of this algorithm to find solutions in 6 classes, suggests that directing the search towards particular areas of the parallel coordinates chart is a viable methodology. Rather than have the user specify several areas of interest prior to running the algorithm, it may be feasible to allow the user to identify areas of interest at runtime, by interacting with a parallel coordinates plot of some or all of the current population whilst the algorithm is running. The use of financial cost in order to select only one solution for presentation, needs further consideration; approaches such as using the lowest average score over all criteria, or allowing multiple solutions per class (assuming that they are sufficiently different) require investigation.

REFERENCES

[1] Kris Braekers, Richard F. Hartl, Sophie N. Parragh, and Fabien Tricoire. 2016. A bi-objective home care scheduling problem: Analyzing the trade-off between costs and client inconvenience. *European Journal of Operational Research* 248, 2 (2016), 428 – 443. DOI: http://dx.doi.org/10.1016/j.ejor.2015.07.028

[2] J. Arturo Castillo-Salazar, Dario Landa-Silva, and Rong Qu. 2016. Workforce scheduling and routing problems: literature survey and computational study. *Annals of Operations Research* 239, 1 (2016), 39–67. DOI: http://dx.doi.org/10.1007/s10479-014-1687-2

[3] Kalyanmoy Deb, Amrit Pratap, Sameer Agarwal, and T. Meyarivan. 2000. A Fast Elitist Multi-Objective Genetic Algorithm: NSGA-II. *IEEE Transactions on Evolutionary Computation* 6 (2000), 182–197.

[4] National Health Service Employers. 2016. Millage Allowances. (2016). http://www.nhsemployers.org/your-workforce/pay-and-reward/nhs-terms-and-conditions/nhs-terms-and-conditions-of-service-handbook/mileage-allowances.

[5] Transport for London. 2009. *Travel in London: Key trends and develpments*. Technical Report. Transport for London.

[6] Transport for London. 2015. Transport for London Unified API. (2015). https://api.tfl.gov.uk/

[7] OpenStreetMap Foundation. 2014. http://www.openstreetmap.org. (2014). http://www.openstreetmap.org

[8] Nissen V Gunther M. 2012. Application of particle swarm optimization to the british telecom workforce scheduling problem. *Proceedings of the 9th International Conference on the Practice and Theory of Automated Timetabling (PATAT*

Table 4: A summary of results obtained, by solution class, averaged over 10 runs. The % columns represent the difference from the benchmark results in table 2 (<100% equates to an improvement, >100 % indicating a worse result). Note that for class 2 there exist a number of problem instances where no solutions in the final population were allocated to the class.

Class 1

Problem	Total Cost	%	CO2	%	Staff	%	Time	Travel Cost
blon-1	1,884.71	74.4%	678.10	147.91%	10.90	72.67%	4,954.90	233.08
blon-2	1,973.57	72.3%	742.86	148.02%	12.80	67.37%	5,140.70	260.00
blon-4	1,998.26	73.4%	799.15	136.73%	17.60	83.81%	5,200.20	264.86
blon-8	2,546.37	96.8%	1,227.62	199.20%	22.70	103.18%	6,411.10	409.34
blon-rnd	2,085.38	75.5%	816.92	153.21%	16.70	87.89%	5,392.90	287.74
cluster-1	945.14	79.3%	284.15	143.89%	6.10	87.14%	2,565.00	90.14
cluster-2	1,011.85	77.7%	337.23	140.16%	7.70	85.56%	2,704.00	110.52
cluster-4	1,080.06	79.6%	414.32	149.73%	11.40	103.64%	2,855.90	128.10
cluster-8	1,067.17	76.1%	449.09	155.74%	14.90	124.17%	2,835.40	122.04
cluster-rnd	1,088.30	85.3%	406.01	164.61%	9.60	106.67%	2,873.20	130.56
lon-1	794.70	77.2%	225.00	156.41%	5.00	83.33%	2,204.90	59.74
lon-2	835.26	71.6%	255.40	171.05%	6.30	78.75%	2,291.00	71.60
lon-4	879.30	70.4%	310.00	176.41%	9.40	85.45%	2,386.90	83.67
lon-8	936.04	75.2%	378.70	166.35%	14.60	112.31%	2,514.60	97.84
lon-rnd	894.78	78.8%	307.70	167.82%	8.80	97.78%	2,417.50	88.95
offset-1	2,068.26	83.5%	848.05	125.54%	11.60	82.86%	5,343.70	287.03
offset-2	2,207.25	77.2%	972.41	124.25%	14.20	74.74%	5,642.20	326.52
offset-4	2,388.52	81.3%	1,139.61	126.40%	19.90	90.45%	6,043.40	374.05
offset-8	2,552.67	88.8%	1,235.16	125.74%	22.20	105.71%	6,412.90	415.04
offset-rnd	2,318.57	87.0%	1,030.71	131.23%	15.10	88.82%	5,902.30	351.14

Class 2

Problem	Total Cost	%	CO2	%	Staff	%	Time	Travel Cost
blon-1	188.10	7.4%	69.40	15.14%	1.10	7.33%	494.30	23.33
blon-2								
blon-4	399.52	14.7%	156.73	26.81%	3.40	16.19%	1,040.30	52.76
blon-8								
blon-rnd	1,672.17	60.6%	665.12	124.74%	13.40	70.53%	4,320.20	232.10
cluster-1								
cluster-2								
cluster-4								
cluster-8	641.92	45.8%	284.48	98.66%	9.50	79.17%	1,705.40	73.45
cluster-rnd								
lon-1								
lon-2								
lon-4								
lon-8	655.39	52.7%	271.20	119.13%	10.70	82.31%	1,760.40	68.59
lon-rnd	180.64	15.9%	65.60	35.78%	2.00	22.22%	487.00	18.31
offset-1								
offset-2	220.82	7.7%	98.30	12.56%	1.40	7.37%	564.00	32.82
offset-4	239.47	8.2%	113.70	12.61%	2.00	9.09%	605.60	37.60
offset-8	503.10	17.5%	241.89	24.62%	4.30	20.48%	1,264.70	81.53
offset-rnd								

Class 3

Problem	Total Cost	%	CO2	%	Staff	%	Time	Travel Cost
blon-1	1,884.69	74.4%	688.42	150.17%	10.90	72.67%	4,955.50	232.86
blon-2	1,974.46	72.3%	750.05	149.46%	12.80	67.37%	5,142.90	260.16
blon-4	1,997.84	73.4%	809.95	138.57%	17.70	84.29%	5,198.40	265.04
blon-8	2,546.14	96.8%	1,243.51	201.78%	22.80	103.64%	6,414.30	408.04
blon-rnd	2,089.43	75.7%	826.44	154.99%	16.90	88.95%	5,411.60	285.57
cluster-1	942.78	79.1%	291.26	147.50%	6.10	87.14%	2,558.40	89.98
cluster-2	906.84	69.6%	303.82	126.27%	6.80	75.56%	2,424.60	98.64
cluster-4	1,080.19	79.6%	417.52	150.89%	11.40	103.64%	2,856.30	128.09
cluster-8	959.36	68.4%	413.39	143.36%	13.90	115.83%	2,549.90	109.40
cluster-rnd	874.65	68.6%	330.51	134.01%	7.90	87.78%	2,307.50	105.48
lon-1	794.80	77.2%	228.90	159.12%	5.00	83.33%	2,205.00	59.80
lon-2	835.02	71.6%	258.20	172.93%	6.30	78.75%	2,290.50	71.52
lon-4	878.68	70.4%	314.20	178.80%	9.40	85.45%	2,385.20	83.61
lon-8	935.27	75.2%	385.50	169.34%	15.20	116.92%	2,512.40	97.80
lon-rnd	894.38	78.7%	311.60	169.95%	8.80	97.78%	2,417.10	88.68
offset-1	2,071.11	83.6%	857.54	126.95%	11.60	82.86%	5,353.50	286.61
offset-2	2,213.74	77.5%	983.64	125.68%	14.40	75.79%	5,662.20	326.34
offset-4	2,392.14	81.4%	1,149.64	127.51%	19.80	90.00%	6,056.10	373.44
offset-8	2,555.86	88.9%	1,250.60	127.31%	22.30	106.19%	6,424.40	414.40
offset-rnd	2,097.63	78.7%	943.28	120.10%	13.70	80.59%	5,338.40	318.16

Class 4

Problem	Total Cost	%	CO2	%	Staff	%	Time	Travel Cost
blon-1	1,891.20	74.6%	692.60	151.08%	10.80	72.00%	4,964.70	236.30
blon-2	1,980.15	72.6%	753.59	150.16%	12.80	67.37%	5,153.10	262.45
blon-4	1,409.71	51.8%	564.95	96.66%	12.30	58.57%	3,662.40	188.91
blon-8	2,546.57	96.8%	1,247.75	202.46%	22.70	103.18%	6,406.20	411.17
blon-rnd	1,678.96	60.8%	664.99	124.71%	12.90	67.89%	4,339.10	232.59
cluster-1	945.90	79.3%	292.98	148.37%	6.10	87.14%	2,564.30	91.14
cluster-2	908.10	69.7%	304.72	126.65%	6.70	74.44%	2,426.40	99.30
cluster-4	973.21	71.7%	375.32	135.64%	10.10	91.82%	2,571.40	116.08
cluster-8	960.74	68.5%	407.49	141.32%	13.30	110.83%	2,552.30	109.98
cluster-rnd	1,090.08	85.5%	410.71	166.52%	9.60	106.67%	2,876.30	131.31
lon-1	797.08	77.5%	230.30	160.10%	5.00	83.33%	2,209.50	60.58
lon-2	836.56	71.7%	259.00	173.47%	6.30	78.75%	2,293.00	72.23
lon-4	879.41	70.4%	314.20	178.80%	9.40	85.45%	2,386.20	84.01
lon-8	843.71	67.8%	345.10	151.59%	12.70	97.69%	2,264.80	88.78
lon-rnd	896.50	78.9%	312.10	170.22%	8.60	95.56%	2,420.70	89.60
offset-1	2,074.07	83.8%	859.95	127.31%	11.60	82.86%	5,355.50	288.90
offset-2	2,213.89	77.5%	983.61	125.68%	14.30	75.26%	5,655.10	328.85
offset-4	2,391.38	81.4%	1,152.56	127.84%	19.80	90.00%	6,046.10	376.01
offset-8	2,301.39	80.1%	1,127.98	114.82%	20.10	95.71%	5,776.10	376.03
offset-rnd	2,320.41	87.1%	1,047.64	133.38%	15.10	88.82%	5,900.20	353.67

Class 5

Problem	Total Cost	%	CO2	%	Staff	%	Time	Travel Cost
blon-1	1,146.14	45.2%	427.36	93.22%	6.80	45.33%	3,002.70	145.24
blon-2	789.38	28.9%	294.75	58.73%	5.20	27.37%	2,055.10	104.34
blon-4	1,601.82	58.8%	649.56	111.13%	13.70	65.24%	4,165.80	213.22
blon-8	2,548.37	96.9%	1,259.37	204.35%	22.90	104.09%	6,405.20	413.30
blon-rnd	2,088.78	75.6%	836.78	156.93%	17.00	89.47%	5,395.70	290.21
cluster-1	570.53	47.9%	181.61	91.97%	3.70	52.86%	1,543.70	55.96
cluster-2	908.05	69.7%	308.92	128.39%	6.80	75.56%	2,426.10	99.35
cluster-4	972.18	71.7%	383.62	138.64%	10.30	93.64%	2,565.90	116.88
cluster-8	962.50	68.6%	413.19	143.29%	14.20	118.33%	2,555.80	110.57
cluster-rnd	659.39	51.7%	242.13	98.17%	5.80	64.44%	1,740.10	79.36
lon-1	159.95	15.5%	49.10	34.13%	1.00	16.67%	443.10	12.25
lon-2	752.54	64.5%	236.62	158.47%	5.60	70.00%	2,062.50	65.04
lon-4	792.40	63.5%	286.80	163.20%	8.60	78.18%	2,149.70	75.84
lon-8	845.06	67.9%	351.00	154.18%	13.80	106.15%	2,267.20	89.32
lon-rnd	442.91	39.0%	156.00	85.08%	4.40	48.89%	1,199.00	43.25
offset-1	1,245.88	50.3%	523.14	77.44%	7.00	50.00%	3,216.90	173.58
offset-2	1,102.23	38.6%	490.14	62.63%	7.20	37.89%	2,818.20	162.83
offset-4	2,157.86	73.5%	1,057.35	117.27%	18.30	83.18%	5,446.80	342.26
offset-8	2,305.67	80.2%	1,138.63	115.91%	20.20	96.19%	5,783.20	377.94
offset-rnd	1,856.49	69.7%	844.02	107.46%	12.10	71.18%	4,718.40	283.69

Class 6

Problem	Total Cost	%	CO2	%	Staff	%	Time	Travel Cost
blon-1	7,616.65	300.6%	2,747.68	599.35%	45.00	300.00%	19,965.00	961.65
blon-2	12,007.45	439.9%	4,593.65	915.34%	78.00	410.53%	31,179.00	1,614.45
blon-4	8,074.85	296.7%	3,282.58	561.61%	75.00	357.14%	20,944.00	1,093.52
blon-8	20,466.62	778.1%	10,039.37	1629.03%	182.00	827.27%	51,368.00	3,343.95
blon-rnd	18,924.07	685.3%	7,524.69	1411.21%	151.00	794.74%	48,824.00	2,649.41
cluster-1	8,550.29	717.2%	2,638.38	1336.09%	55.00	785.71%	23,144.00	835.62
cluster-2	7,080.63	543.5%	2,359.55	980.65%	52.00	577.78%	18,900.00	780.63
cluster-4	10,847.14	799.5%	4,196.19	1516.46%	113.00	1027.27%	28,621.00	1,306.81
cluster-8	9,644.35	687.5%	4,186.79	1451.98%	141.00	1175.00%	25,590.00	1,114.35
cluster-rnd	9,835.28	771.0%	3,661.05	1484.37%	84.00	933.33%	25,937.00	1,189.62
lon-1	7,214.67	701.2%	2,087.00	1450.82%	45.00	750.00%	19,971.00	557.67
lon-2	6,672.64	572.2%	2,092.20	1401.25%	50.00	625.00%	18,297.00	573.64
lon-4	8,809.38	705.5%	3,143.00	1788.54%	94.00	854.55%	23,889.00	846.38
lon-8	8,439.90	678.4%	3,476.00	1526.91%	135.00	1038.46%	22,643.00	892.23
lon-rnd	8,120.00	714.9%	2,809.98	1532.58%	78.00	866.67%	21,889.00	823.66
offset-1	16,707.22	674.8%	6,877.82	1018.18%	94.00	671.43%	43,086.00	2,345.22
offset-2	22,257.22	778.9%	9,873.80	1261.59%	143.00	752.63%	56,791.00	3,326.89
offset-4	21,545.44	733.5%	10,363.91	1149.50%	178.00	809.09%	54,393.00	3,414.44
offset-8	23,070.60	802.5%	11,342.25	1154.60%	201.00	957.14%	57,828.00	3,794.60
offset-rnd	21,036.74	789.6%	9,491.84	1208.47%	138.00	811.76%	53,407.00	3,234.40

2012), Son, Norway (2012).

[9] M. Haklay and P. Weber. 2008. OpenStreetMap: User-Generated Street Maps. *Pervasive Computing, IEEE* 7, 4 (Oct 2008), 12–18. DOI: http://dx.doi.org/10.1109/MPRV.2008.80

[10] Gerhard Hiermann, Matthias Prandtstetter, Andrea Rendl, Jakob Puchinger, and Günther R. Raidl. 2015. Metaheuristics for solving a multimodal home-healthcare scheduling problem. *Central European Journal of Operations Research* 23, 1 (2015), 89–113. DOI: http://dx.doi.org/10.1007/s10100-013-0305-8

[11] A. Inselberg. 2009. *Parallel Coordinates. Visual MultidimensionalGeometry and its applications*. Springer.

[12] P Karich. 2014. GraphHopper. https://graphhopper.com/. (2014). https://graphhopper.com/

[13] Smet P. Verbeeck K. Misir, M. and G. Vanden Berghe. 2011. Security personnel routing and rostering: a hyper- heuristic approach. *Proceedings of the 3rd International Conference on Applied Operational Research, ICAOR11* (2011).

[14] Payscale.com. 2016. Social Worker Salary (UK). (2016). http://www.payscale.com/research/UK/Job=Social_Worker/Salary

[15] Matias Sevel Rasmussen, Tor Justesen, Anders Dohn, and Jesper Larsen. 2010. *The Home Care Crew Scheduling Problem:: Preference-Based Visit Clustering and Temporal Dependencies*. DTU Management.

[16] Neil Urquhart and Emma Hart. 2015. Creating Optimised Employee Travel Plans.. In *Proceedings of the The 11th edition of the International Conference on Evolutionary and Deterministic Methods for Design, Optimization and Control with Applications to Industrial and Societal Problems*.

[17] Neil B. Urquhart, Emma Hart, and Alistair Judson. 2015. Multi-Modal Employee Routing with Time Windows in an Urban Environment. In *Proceedings of the Companion Publication of the 2015 Annual Conference on Genetic and Evolutionary Computation (GECCO Companion '15)*. ACM, New York, NY, USA, 1503–1504. DOI: http://dx.doi.org/10.1145/2739482.2764649

[18] C.L. Valenzuela. 2002. A simple evolutionary algorithm for multi-objective optimization (SEAMO). In *Evolutionary Computation, 2002. CEC '02. Proceedings of the 2002 Congress on*, Vol. 1. 717–722. DOI: http://dx.doi.org/10.1109/CEC.2002.1007014

[19] Eckart Zitzler, Marco Laumanns, and Lothar Thiele. 2002. SPEA2: Improving the Strength Pareto Evolutionary Algorithm for Multiobjective Optimization. In *Evolutionary Methods for Design, Optimisation, and Control*. CIMNE, Barcelona, Spain, 95–100.