On the Partitioning of Dynamic Scheduling Problems – Assigning Technicians to Areas *

Yossi Borenstein
yboren@essex.ac.uk

Nazaraf Shah
shahn@essex.ac.uk

Edward Tsang
edward@essex.ac.uk

Raphaël Dorne
raphael.dorne@BT.com

Abdullah Alsheddy
aalshe@essex.ac.uk

Christos Voudouris
chris.voudouris@BT.com

ABSTRACT

BT workforce scheduling problem considers technicians (with different skills) which are assigned to tasks which arrive (partially) dynamically during the day. In order to manage their workforce, BT divides the different regions into several areas. In the beginning of each day all the technicians in a region are assigned to one of these areas. During the day, tasks can only be allocated to technicians from the same area.In this paper we use a (1+1) EA in order to decide, once the area have been defined, which technicians to assign to which areas.

Categories and Subject Descriptors

H.4 [**I.2.8 Problem Solving, Control Methods, and Search**]: Heuristic methods

General Terms

Algorithms

1. PROBLEM DESCRIPTION

The region under investigation has 86 engineers which perform around 250 tasks every day. Some of the tasks (around 220) are scheduled in advance and some (around 50) arrive dynamically during the day. Each task requires a technician with a particular skill. We distinguish between ten different skills, abbreviated: $s_1, \cdots, s_{10}$. Each technician can have one or more skills. We assume that the number of skills (that technicians in the workforce have) follows a binomial-like distribution.

The time required to finish a task varies as a function of the associated skill. This is modeled as a triangle distribution (based on past data). In this paper we consider a simple scenario in which all the tasks can be modeled by the same triangle distribution: (60, 95, 180).

Some scheduled tasks can be performed throughout the day, other are associated with a time window (either morning or afternoon). Tasks which arrive dynamically must be performed within 3 hours.

We consider a 80 km^2 region. The density of tasks over this region is based on real geographical distribution of houses. Every morning the region is divided into several non-intersecting areas. Tasks which fall within a boundary of a particular area are considered to be in that area. We are looking at the problem of deciding how to assign technicians into these areas. The assignment should maximize the matching between the demand (number and type of tasks) of an area to the capability of the assigned technicians. We assume that technicians start the working day from home. For this reason the distance of a technician from the area should be considered as well.

Travel time (between any two coordinates) is measured as the Euclidean distance divided by speed. We do not consider dynamic travel times. Finally, we are particularly interested in highly constraint travel times and hence we assume that the speed is 5 km/h.

As technicians become available they are allocated according to a rule based system to a new task. The performance of the system is measured solely by the number of non-completed tasks.

A detailed simulation of BT's workforce allocation has been developed. Given any configuration of the problem (i.e., tasks, technicians and the division into areas) the simulation can give an estimate of the number of unallocated jobs in the end of the day. It is possible to use, in principle, the output of the simulation as a performance measure. However, the simulation is both very expensive and very noisy. In the following we develop algorithms which assign technicians to areas which are not based on the simulation. We use the simulation only to compare, in the end, the output of the different algorithms.

2. ALGORITHMS

The assignment problem can be divided into two subproblems. Firstly, as the number of tasks in each cluster differs, it is necessary to decide how many technicians to assign to each area. Secondly, one would like to assign technicians which can match the demand (needed skills). Naturally, everything is subject to constraints (distance of technicians from areas and different demands and skill mixes).

We compare the performance of three stochastic heuris-

*Borenstein, Shah, Tsang and Alsheddy are from the Dept. of Computing and Electronic Systems, University of Essex, U.K. Dorne and Voudouris are from the Intelligent Enterprise Technologies Groups, BTexact, U.K

GECCO'08, July 12–16, 2008, Atlanta, Georgia, USA.
ACM 978-1-60558-130-9/08/07.

tics. The first one is a simple random search (which is mainly used for comparison purposes). The other two are described in sections 2.1 and 2.2.

2.1 Randomized Heuristic

The randomized heuristic builds solutions in two steps. In the first, the number of technicians in each area is chosen (stochastically). In the second, technicians are assigned in a greedy manner to the different areas.

We calculate for each area (under simplified conditions) the expected number of technicians required to perform all the jobs (denoted $f_{required}(a)$). In each step we choose an area a with a probability proportional to $f_{required}(a)$. Given an area and a set of technicians which has not been assigned, the next technician is chosen deterministically according to the expected number of tasks it can do in this area (considering skills and distance constraints).

2.2 Multiobjective $(\mu + 1)$ EA

Given an assignment of technicians to regions, it is possible to measure some properties of the assignment which are likely to be correlated with the performance. We define four measurable properties of a particular assignment. In the following, we note in brackets whether we want to maximize or minimize them:

- $[f_{util}]$ (maximize): the approximated number of tasks a technician can do (considering available time, task duration etc.).
- $[f_{Delta}]$ (maximize): the number of tasks (on average) that a technician can perform considering the "competition" factor (i.e., the number of technicians available for each task)
- $[f_{dyn}]$ (maximize) : the fraction of of skills the technicians in a particular area cover
- $[f_{cost}]$ (minimize) : the impact of partitioning a problem on the number of tasks a technician has the skill to perform

Ideally, for any particular clustering we can find a solution which optimizes all the different features. Unfortunately, this is difficult to be found in practice.

Furthermore, it is difficult to find a (linear) combination of these features which can reliably account for what each represents independently. For example, to maximize the skill mixes in an area, one should (possibly) compromise on technicians which are located far away from the area to which they are assigned. The travel time required to reach the area should affect negatively the value of f_{util}.

We decided to follow a different approach. We consider the problem as a multiobjective problem. Recall that in a multiobjective optimization the output of the algorithm is not a single solution but, rather, a set of non-dominated solutions (i.e., none of the solutions in the set is strictly better nor, worse than another solution). In our case, by better we mean, none of the solutions have strictly better features than another solution.

We defined two operators: a *swap* operator and a *move* operator. The swap operator chooses uniformly at random two areas. It selects from each area a technician with the maximum potential utility for the other area and swap them. This can be described as follows:

- Select $a_1, a_2 \in A$ uniformly at random
- Let $r_1 = \arg\max r \in a_1 \;\; f_{util}(a_2, r)$
- Let $r_2 = \arg\max r \in a_2 \;\; f_{util}(a_1, r)$
- Assign r_1 to a_2. Assign r_2 to a_1.

where a_i denotes an area in the set of all possible areas A, r denotes a resource (i.e., a technician) and $f_{util}(a_i, r)$ is the approximated number of tasks that r can perform when assigned to area a_i.

The move operator selects uniformly at random a technician and assigns it to *another* area which maximizes its utility:

- Select $r \in R$ uniformly at random. Assume $r \in a'$.
- Let $a^* = \;\arg\max a \neq a' \in A \;\; f_{util}(a, r)$
- Assign r to a^*.

Finally, in order to focus the search in a particular region of the pareto front, we made some preliminary experiments in which we calculated the correlation between the different features and the actual cost as measured by running the simulation. We found out that f_{util} gives the highest correlation. In order to make the search focused around the most promising feature we implemented the following heuristic. Every time the size of the pareto front exceeded 100 solutions, we chose the 50 solutions with the highest f_{util} and discarded the other solutions. The algorithm can be described as follows:

Initialize: sample 100 solutions using the Stochastic Heuristic algorithm to generate an initial Pareto set.
while NumberOfEvaluations < MAX **do**
  Chooses a solution uniformly at random from the pareto set
  Apply the *swap* operator
  Conditionally add the new solution to the pareto set
  Apply the *move* operator
  Conditionally add the new solution to the pareto set
  if the pareto set > 100, delete 50 solutions with the smallest f_{util} value
end while

3. CONCLUSIONS

We compared the performance of the three algorithm in 78 trials. In each trial, the three heuristics generated 50 solutions. The solutions generated by the $(\mu + 1)$ EA were the pareto set obtained after 10,000 fitness evaluations. The cost of each solution was measured as the average output of 20 simulation's runs.

We measured the number of times that an algorithm both gave the best solution (in each trial) and was shown to significantly (i.e., $p < 0.05$) outperform the others. Random search always gave significantly worse solutions. The randomized heuristic significantly performed better in 15% of the trials. Finally, the EA gave significantly better results in 50% of the trials.

We strongly believe that the multiobjective approach which was used to solve this problem can be used with other problems in which the actual cost function is noisy and expensive (as long as some measurable features of the function can be identified).