SpaceMAP - Applying Meta-Heuristics to Real World Space Allocation Problems in Academic Institutions

E.K. Burke[1], C. Beyrouthy[1], J.D. Landa Silva[1],
B. McCollum[2], and P. McMullan[2]

[1] Automated Scheduling, Optimisation and Planning Research Group
School of Computer Science and IT, University of Nottingham, UK
{ekb, cbb, jds}@cs.nott.ac.uk
[2] School of Computer Science
The Queen's University of Belfast, UK
{b.mccollum, p.p.mcmullan}@qub.ac.uk
[3] Realtime Solutions Limited
21 Stranmillis Road, Belfast, Northern Ireland
http://www.realtimesolutions-uk.com

1 Overview

Space allocation is an important issue in the university sector [1]. Frequently, the utilisation of office space is calculated as the product of the frequency of use of individual rooms and the associated occupancy at a particular time. The Higher Education Funding Council for England (HEFCE) reported a mean frequency value of 51% across UK institutions and a mean occupancy rate of 52%. This gives a mean utilisation rate of 26.5%. The main reason for this massive inefficiency in use of space is the poor allocation of space within the institutions. Therefore, an automated system that takes into consideration user defined constraints to calculate the best allocation of activities to rooms would help to improve enormously the efficiency of space utilisation. Here, activities are defined as any event that requires resources to support its delivery. Examples of activities include: staff, lectures, laboratories, research, tutorials, etc. Currently, due to the complexity of the issues involved and the associated intricate constraints, activities are allocated to rooms through manual processes that cannot reflect the most efficient manner of achieving this allocation. We aim to provide an automated system that replaces the manual process with one that is based on achieving the most efficient and effective allocation of activities to spaces, leading to considerable capital and operational cost savings for individual institutions.

2 Goals of SpaceMAP

SpaceMAP is an automated system, under development by Realtime Solutions Limited, which seeks to achieve the following:

1. For new build projects, allow a quick and efficient means of allocating activities to new space. This will be achieved by modelling scenarios, which will in the first instance match the space requirements with possible configurations of space. This should be achieved graphically allowing the user to modify the suggested configuration.
2. Ensure that as many interrelationships between office users are recorded and factored into calculating room allocations.
3. Provide intelligent interfaces, which enables information about users of a particular space to be recorded. In particular, this information should be graphically represented and intelligent algorithms developed which enable multiple solutions to be produced.
4. Ensure that necessary connections between staff and the facilities and resources in buildings are also factored into the room allocation process.
5. For existing buildings, allow an objective review of room allocations to provide a potentially improved allocation. This will be particularly powerful for modelling the impact of change on existing facilities - student numbers rising in one Department and falling in another for example. The algorithm will review and determine the most efficient and effective allocation of current activities to existing accommodation.
6. Remove inefficient practices in the allocation of space to existing activities including staff.
7. Highlight where activities have been allocated into spaces that are not appropriate.
8. Allow the user to try various scenarios in defining the best allocation. As 'best allocation' is user defined, it should be possible to work through many alternate room allocations according to the priorities that are driving the particular problem.

3 Heuristics in SpaceMAP

An investigation into the development of metaheuristic techniques to tackle the first issue listed above was presented in [2]. In that work, a number of heuristics were proposed and tested on a range of test instances. The heuristics proposed include constructive heuristics, neighbourhood exploration heuristics and several metaheuristic approaches. Now, we incorporate the heuristic methods proposed in [2] to the SpaceMAP system. Then, we apply SpaceMAP to various real-world scenarios and report on the results obtained and observations made.

Two initialisation heuristics are incorporated into SpaceMAP:

AllocateRnd-BestRnd. This is a partially random and partially greedy (also called peckish) heuristic. It first selects one unallocated activity a at random. Then, it explores a subset of the available rooms and allocates a to the best one.

AllocateCsrt-BestRnd. This heuristic is similar to the above one but it also takes into consideration how constrained the problem instance is when allocating activities to rooms. It first selects one unallocated activity a at random. If there are hard constraints associated to the activity a then one room is chosen to allocate a ensuring that these hard constraints are satisfied. If no hard constraints are associated to activity a then the allocation is made as in the heuristic AllocateRnd-BestRnd.

Local search is used to improve the initialised allocations. The three neighbourhood structures that have been incorporated into SpaceMAP are the following:

RelocateRnd-BestRnd. Selects an allocated activity a at random. Then, it explores a subset of the available rooms and relocates a to the best one.

SwapRnd-BestRnd. Selects an allocated activity a at random. Then, it explores a subset A of allocated activities and swaps the assigned rooms between a and the activity in A that produces the best improvement in the allocation.

InterchangeRnd-BestRnd. Selects a non-empty room r_1 at random. Then, it explores a subset R of non-empty rooms and interchanges the allocated activities between r_1 and the room r_2 in R that produces the best improvement in the allocation. The interchange consists on moving all activities from r_1 to r_2 and from r_2 to r_1.

4 A New Model for the Space Allocation Problem

The formulation and test instances described in [2] for the space allocation problem include many of the requirements and constraints that are encountered on real-world problems. However, the complexity and the intricate constrained nature of real-world scenarios make it difficult to model the space allocation problem accurately. Therefore, we also describe a new model based on graph theory that helps to better capture all the details of real-world space allocation problems [3]).

In the new model, the set of rooms is represented as a fully connected graph $G(V, E)$ in which V is the set of nodes representing rooms and E is the set of edges representing the physical distance between rooms. The set of activities is represented as a labelled spanning tree $R(V_{Ai}, E_{Ai})$ where V_{Ai} is the set of activities and E_{Ai} represents the logical connectedness between those nodes. Then we create a mapping between the spanning tree and the fully connected graph. Such a mapping corresponds to assigning activities to rooms and each assignment is labelled by its relative occupancy and frequency of use. In our model, a labeled minimum spanning tree is a spanning tree to which we add parameters (e.g. occupancy and frequency of use) to every node. We introduce the concept of triggered allocation which assigns an instance of $R(V_{Ai}, E_{Ai})$ to the nodes of $G(V, E)$. The assignment generates a new graph and two main candidate parameters F and O that express occupancy and basic frequency of utilization respectively. The model attempts to incorporate all constraints in two flexible graph structures and is has the flexibility for growth and expansion in order to incorporate time slots and associated timetabling parameters.

References

1. Burke E.K., Varley D.B.: Space allocation: an analysis of higher education requirements. The practice and theory of automated timetabling II: Selected papers from the 2nd international conference on the practice and theory of automated timetabling (PATAT 1997). Lecture notes in computer science, Vol. 1408, 20-33, Springer, (1998).
2. Landa Silva J.D.: Metaheuristics and multiobjective approaches for space allocation. PhD thesis. School of computer science and information technology. University of Nottingham, UK, (2003).
3. Beyrouthy C.: An enriched model based on graph theory for the real-world space allocation problem. Technical report (available on request), School of computer science and information technology, University of Nottingham, UK, (2004).