Chapter 6

Conclusion

6.1 Contributions

For this thesis, we studied the berth allocation problem (BAP) in a container transshipment port to be solved as a two-staged problem: a partitioning subproblem and a packing subproblem. We solved the BAP partitioning subproblem as a generalisation of the multi-way graph partitioning problem and extended our BAP-GP algorithm to a BAP partitioning framework with pre- and post-processor heuristics. Prior to this thesis, there is no known graph partitioning literature for the BAP. BAP-GP iteratively improves upon an initial solution until it reaches a local optimal. We studied different methods of generating initial solutions and found that methods which actively cluster vessels based on their transshipment volumes give good solutions. BAP-GP runs very fast and experimental results show that it performs very well for small data sets. For larger data sets, much better solutions are obtained if more time is invested in trying more different starting configurations, and the time advantage of BAP-GP gives us the freedom to do so. We also studied the feasibility issues of assigning vessels in a busy time zone. We modelled this as an instance of the multiple knapsack problem and tried to resolve it with a simple pairwise exchange heuristic. However, this heuristic is not effective and there is still much room for improvement in this aspect.

When we set out to study the BAP, very little research literature was available on it and real world information/data was hard to come by. So we had to put in extra effort to collect, verify, filter and consolidate BAP information and data from other sources. Through this process, we have written several documents on the BAP which will be helpful to others who carry on this research work ([RAS96], [RAS99]).

We designed and implemented a BAP planning system — the BAPS. The strength of the BAPS lies in its repertoire of BAP algorithms (which is still growing). It offers the user a choice of many algorithms, for which he may mix and match to tailor to different user requirements — long term planning, short term planning, trade-off between running time and solution quality. In addition, the BAPS contains subsystem extensions to the main solver engine to assist in data generation, experimentation and result analysis. This is not only useful in a research environment, but it could provide the layman planner with the capability to perform port planning by generating what-if scenarios and evaluating them.

6.2 Lessons Learnt

In doing this thesis, we learnt lessons in the areas of research and system development. In the area of research, we learnt about graph partitioning and knapsack problems. Both problems are NP-hard in nature and researchers have devised many techniques to overcome them. For VLSI in particular, where the problem sizes are very large, many innovative and ingenious methods have been developed that produces quality solutions and yet have good running times. We applied the knowledge we learnt about both problems to solve a real world problem — the BAP. Modelling and solving the BAP involved a good balance of mathematical formulation, data structure design and algorithm development.

In the area of system development, the BAPS was built through a collaboration with several other M.Sc. students, and it is a system of substantial size. To roll out such a system, proper project planning and coordination is important. The interfaces for the various subsystems must be well-planned and designed in order for them to work together flawlessly. We had to pick up sufficient knowledge of the different programming languages, operating system environments and hardware platforms in order to engineer a cross-platform BAPS that is easily extendible due to its object oriented nature.

6.3 Future Enhancements

Here we highlight some possible improvements and extensions to our BAP project. We classify these into short and long term extensions to provide some direction to those who carry on our research work.

6.3.1 Short Term Extensions

These extensions are relatively simple to implement for which we were unable to do so in the time frame for this thesis.

- Modifying BAP-GP to implement the 2-Phase FM idea from [WeCh90]. This extension has good potential to improve the quality of the partitioning solution. What is required is:
  1. A data structure to allow vessels to be clustered into a "hyper-vessel".
  2. An abstraction of BAP-GP to work on these hyper-vessels.
- Currently the post-processor heuristic is not effective in resolving infeasible solutions as it is very strict: it only swaps a pair of vessels if both sections have room to allow it, and this is certainly futile in a busy time zone. An alternative might be to relax the heuristic by swapping the vessels even if doing so overshoots the capacity of a section. This may free up enough space to make a good move later, which may allow all vessels to be assigned.

6.3.2 Long Term Extensions

These extensions require some additional study and research before we can implement them. Some of them may not have obvious solutions.

- Investigate different gain functions — of interest is a recent paper in VLSI partitioning which uses a gain function based on probability theory to incorporate look-ahead intelligence ([DuDe96]). The results look very promising. After implementing several gain functions, we can compare them and study the effectiveness and relevance of each.

- In-depth study into the BAP-Part infeasibility problem, for which we did not have time to do so. We should improve the performance of the post-processor heuristic by having it swap groups of vessels rather than just pairs of vessels. On the longer term, we could look into developing approximation algorithms for assigning the vessels in busy time zones or perhaps try to establish some lower bounds to drive the solution.