19th International Congress on Modelling and Simulation, Perth, Australia, 12–16 December 2011
http://mssanz.org.au/modsim2011

Exploiting Symmetries in Logistics Distribution Planning

L. Marsh [a] **and D. Gossink** [a]

[a] *Defence Science and Technology Organisation, Edinburgh, South Australia, Australia*
E-Mail: Luke.Marsh@dsto.defence.gov.au

Abstract: Logistics distribution and movements planning involves developing an efficient plan to move all of the required supplies, equipment and personnel to desired locations in a timely manner. It is an inherently difficult problem to solve and given the complexity of the problem, scenarios of a realistic size can take a significant amount of time to compute.

A greedy distribution algorithm has been created for the task of developing transport schedules to satisfy the movement requirements. At each phase of the simulation the algorithm assigns the most capable transport to deliver the highest priority resources. Although it is observed that the distribution algorithm performs faster than exact algorithms based on integer programming or brute force, even greedy based approaches can have performance issues as the scenario size increases.

In this paper, a technique is examined which exploits redundancies garnered due to symmetries in the distribution model to improve performance of the distribution algorithm. A key redundancy in the model occurs when there are multiple similar transports at the same location. By performing one calculation for all transports of a similar type at the same location, it is suspected that computational savings could be made. Given that many of the distribution scenarios are complex, many of the transports are not always at the exact location as each other. One way to increase the effectiveness of the symmetry exploitation is to perform only one calculation for all transports of a similar type within a certain range. However, there is a side effect to increasing the range value, and that is it introduces an error in the planning calculation. Part of the analysis to be explored through simulation is what an effective range is, given that the larger the range the greater the computational savings but also the greater the error is introduced into the distribution plan.

Simulations were run testing the performance of the distribution algorithm with and without utilising the symmetries with varying range values to minimize the distribution model. It is shown empirically that by exploiting the symmetries in the distribution model, for the scenarios tested computation time could be reduced on average by approximately 24-45%, while the efficiency of the distribution plan is only reduced by approximately 0.3-18% due to introduced error.

Keywords: *Logistics distribution, movements planning, model minimisation*

1. INTRODUCTION

Logistics distribution and movements planning involves developing an efficient plan to move all of the required supplies, equipment and personnel to desired locations in a timely manner (Perugini, 2006). It is an inherently difficult problem to solve, and is suspected to be NP-Hard (Cook, 1971). Given the complexity of the problem, scenarios of a realistic size can take a significant amount of time to compute.

A greedy distribution algorithm (Cormen et al., 1990) has been created which develops a schedule for each transport in the scenario, to satisfy all of the required movements and complete the distribution. It aims to complete all movements as quickly as possible by greedily assigning the most capable transports to the highest priority items to move. An overview of the algorithm is outlined in Section 3. For this distribution problem, even greedy based approaches can have performance issues as the scenario size increases. It is suspected that improvements to the greedy distribution algorithm developed could be made.

A technique that exploits redundancies (Ravindran and Barto, 2001) which occur due to symmetries in the distribution model to improve performance of the distribution algorithm is examined in Section 4. Finally, Section 5 outlines the simulations which were run in order to test the performance of the distribution algorithm with and without utilizing the symmetries to minimize the distribution model.

2. OVERVIEW OF THE DISTRIBUTION PROBLEM

The distribution and movements planning domain which is the focus of this paper incorporates many transport platforms covering land, sea, air and rail with a goal of delivering supplies to a destination from a source location. The supplies, equipment and people can be sourced from many locations, and similarly need to be transported to possibly many locations. Immediately it can be seen that the solution space can expand rapidly, as the number of transports, routes, base locations and requirements of supply increases. Additionally there are business rules, constraints and complexities inherent in this distribution domain, which need to be satisfied when developing a plan.

The set of complexities inherent to the distribution domain include: multiple source locations; multiple destination locations; multiple route types (land, sea, air, rail); directed routes; multiple routes to destination, intermediary locations to destination; known as "transshipment"; multiple transport trips via 'drop and swap'; many types of transports, multiple supply types to be moved, transports can carry many types of supply at once; and many trips may be performed if needed to transport item from the source to destination.

The set of constraints inherent in this distribution problem include: The minimum runway length required for aircraft; minimum runway classification required for aircraft, known as "pavement classification number"; transport speed; transport range; transport personnel capacity; transport cargo capacity; and transport loading/unloading times. If in the scenario there is no route to the destination, or due to the constraints or specifications on the transports there are no possible options for distribution, then the scenario is deemed infeasible.

For this logistics distribution problem the most relevant area in the literature refers to the Vehicle Routing Problem (VRP) (Toth and Vigo, 2002). The VRP can be seen as a generalised Travelling Salesman Problem (TSP) when certain constraints in the VRP are relaxed (Kulkarni and Bhave, 1985). The VRP covers many logistics distribution activities, and has many variants (Toth and Vigo, 2002). It is a commonly studied problem in combinatorial optimisation given its complexity as an NP-Hard problem (Toth and Vigo, 2002). There is a plethora of work done applying different techniques to solve the VRP. Laporte (1991) provides a comprehensive review of various works done applying exact and heuristic techniques to the VRP, and Yeun et al. (2008) provides an excellent review of the more recent work done applying various algorithms to the VRP. Some other more relevant recent work includes: Geiger (2010) who outlines a number of fast approximation heuristics to solve the VRP, and Ma et al (2010) who specifies two fast greedy algorithms for solving the VRP.

While the VRP is the closest problem in the literature to the logistics distribution problem used as the topic of this paper, it has some significant differences. The first difference is that the goal of this distribution problem is to make the deliveries as quickly as possible, as opposed to trying to minimise the overall cost as in the VRP. In this distribution problem transports are not required to finish at the starting location between each trip, and can have different capacities and distance costs. There are also many different types of resources that need to be transported, each with different dimensions and requirements. Also transports can carry many types of resources at once, and customers can also be serviced by more than one transport. Finally, in this distribution problem there is a requirement to move large amounts of resources; this entails many trips being conducted, as opposed to the VRP where a customer node only needs to be visited once.

Since the VRP has too many significant differences to the logistics distribution problem outlined in this work, as stated above, exact formulations and solutions as stated in the literature cannot be directly used. Therefore a customised greedy algorithm was developed to find solutions to this specific problem. While the scope of this paper is not to fully describe the greedy distribution algorithm developed, a brief description of it is outlined below.

3. GREEDY DISTRIBUTION ALGORITHM

The goal of the greedy distribution algorithm is to develop a schedule for each transport in the scenario, referred to as a distribution plan, to satisfy all of the required movements. The distribution plan is developed over a number of phases. In each phase, the algorithm goes through each required movement (deployment, sustainment, re-deployment, administrative move, etc.) which are sorted by priority (starting at the highest priority movement), and through each item within the movement (food, water, etc), and assigns the best transports for the movement based on the potential ‘flow rate’ of each transport. The *flow rate* is calculated as in (1) below.

$$flowrate = \frac{\min(h,r)}{t} \tag{1}$$

The *flow rate* is calculated as the minimum of how much of the item the transport can hold *h* and how much of the item is required to be moved *r*, divided by the time *t* it takes the current transport to pick up the item and deliver it. The higher the *flow rate*, the more capable the transport is, to deliver the item. If a transport cannot make the movement due to a distribution constraint, a *flow rate* of 0 is applied. Note that *t* is calculated by the distance to travel divided by the speed of the transport. The distance to travel is calculated using A*, which is a heuristic-based best-first search (Nilsson, 1980). Given that there are potentially many routes from a source to a destination node, A* is used to find an efficient route and therefore distance through the distribution network. This calculation also takes into account many of the constraints mentioned in Section 2.

For each item in each movement the above calculation is performed for each transport. The flow rates for all of the transports are sorted, and the transport(s) with the highest *flow rate(s)* are assigned to move the current item. If there are insufficient transports in the scenario to move all of the items entirely, the remaining items will be assigned to transports in subsequent phases of the distribution plan. The idea of the algorithm is that it greedily assigns for the current phase the transport(s) which can move the items the fastest. As the movements are sorted via priority, the highest priority items get the most capable transport(s) assigned.

Once all movements have been addressed in the current phase or all transports have been assigned, the simulation is forwarded to the point in time to when a transport finishes its schedule first before other transports for the current phase (signalling that there are free transports in the scenario to assign to new movements), the algorithm is re-run and a new phase is calculated as above (ignoring transports which still have movements to complete, and acknowledging what has already been moved and assigned in previous phases). The algorithm is repeated until all movements have been satisfied, or the remaining movements cannot be satisfied due to constraints.

Given that the distribution algorithm calculates the *flow rate* mentioned above for each item for each required movement and for each transport over all of the calculated phases in the scenario, this results in a lot of computation. The exact computational complexity of the greedy distribution algorithm is not in the scope of this paper, however it can be foreseen that as the number of movements, items and transports increases the amount of computation increases. Prior testing of the distribution algorithm in large real world scenarios has required minutes of calculation on a PC with an Intel i7 2.80 GHz, 4 GB of RAM, whereas exact algorithms using integer programming can take days to compute much smaller scenarios (Marsh, 2008). The requirements of this work are to do broad logistics feasibility analysis in an extremely time pressured environment; therefore exact optimal plans are not a requirement, although the speed of computation is. For user-acceptance of this work, it is required that the computation time required by this distribution algorithm be less than ten minutes for large real world plans, but the faster the better. Given that integer programming techniques can take days to produce an optimal solution, the user is willing to accept sub-optimality in the solutions in order to get a timely result. Determining the difference in quality between the optimal solution and solutions generated with this greedy distribution algorithm is not in the scope of this paper. Although it is observed that this greedy distribution algorithm performs faster than optimal algorithms (Marsh, 2008), improvements could still be made.

4. SYMMETRY EXPLOITATION

A key redundancy in the model occurs when there are multiple similar transports at the same location. The distribution algorithm in this case would compute individual *flow rates* for all of these similar transports for all items and requirements. However it could be advantageous to compute the flow rate only once for transports of a similar type (i.e. they have the same general attributes of speed, range, and capacity) and location.

Given that many of the scenarios are complex, many of the transports are not always at the exact location of each other. One way to increase the effectiveness of the symmetry exploitation is to perform only one *flow rate* calculation for all transport of a similar type within a certain range *R*. This concept is depicted in Figure 1, where there are 5 nodes (1-5) and two transports (T_1-T_2), where in this example the range between T_1 and T_2 is 5 units. If T_1 and T_2 where similar transports and *R* was specified to be 5, then only one *flow rate* calculation would be performed for both transports, thus saving computational processing.

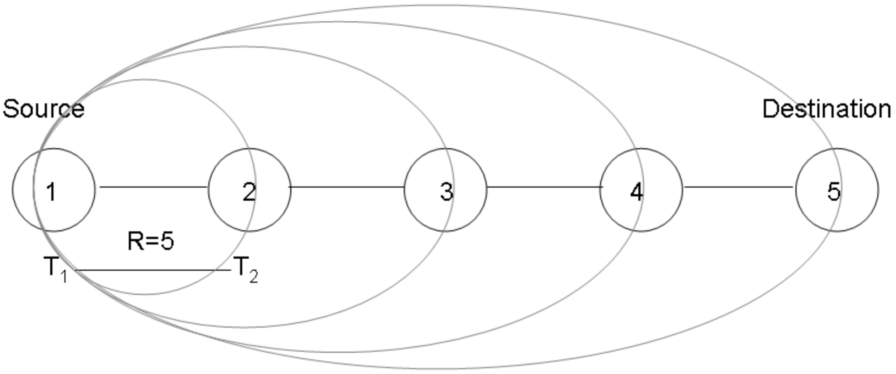

Figure 1. Example of a distribution scenario highlighting the possible symmetry ranges.

There is a side effect to increasing the range value in that it introduces an error ε in the planning calculation. For example, in Figure 1, the distance between each node is 5 units. Transport 1 (T_1) is at the source node (1) and takes 20 units to travel to the destination node (5). Transport 2 (T_2) is at node 2 and takes 25 units: 5 units to travel to the source node first and then 20 units to travel to the destination node. If *R* is set to 5, one planning calculation is performed, covering both transports. Taking the average location of both transports (the mid-point between nodes 1 and 2) in consideration in the planning calculation gives a final cost of 22.5 for one leg for both transports. If T_1 is chosen randomly to perform the trip, an actual trip cost of 20 is made, and $\varepsilon = 0$, as the most efficient trip would have taken 20. If T_2 is chosen randomly to perform the trip, an actual trip cost of 25 is made, and $\varepsilon = 5$, as the most efficient trip would have taken 20.

Presumably the larger the radius and the more transports included in the planning calculation, the less computation *c* required, but the larger the possible error ε is introduced. The question to be explored is: how much computational savings can be made using this symmetry exploitation technique, and how much efficiency is lost in the overall distribution plan due to introduced error? It will be up to the clients of this work to determine whether the proportion of computational savings versus inefficiency introduced in the distribution plan is acceptable for this technique to be employed.

5. SIMULATION

5.1. Preliminary Analysis

Before the effect of the radius on computation and error through simulation is explored, a preliminary question to determine the usefulness of the symmetry breaking technique would be "how many of the same type of transport does there need to be for symmetry breaking to pay-off?" It is obvious that for the symmetry breaking to have any effect, there needs to be at least two transports of the same type, with more than two preferred. In Figure 1 the notion of an increasing radius *R* was presented, where a constraint of transports being at the same location could be relaxed. Rather than saying that they are in the same locus of *R*, they can be treated as the same transport for symmetry breaking purposes. It is this relaxation that carries an intrinsic penalty of error for the calculations. This was described by example before hand where for equally likely cases the error is *R/2*. Naturally as *R* increases there are two direct effects: firstly, the likelihood of the number of same transports encountered increases; and secondly, the error also increases. A related question might be "under some assumptions can *R* be determined such that there is a satisfactory likelihood of encountering the same transport type?" Concomitant with the acceptance of *R* would be that it is within an accepted error range. The first step to addressing the last question is to make the simplifying assumption that the available transports are uniformly distributed over a maximum distance of *L*. It is easily shown that under this assumption the probability that the transport is within a radius *R* is:

$$\alpha = R / L \tag{2}$$

If there are *n* independent identically distributed transports the probability that there are *k* transports within the radius *R* can be derived from the binomial distribution and is:

$$\binom{n}{k}\alpha^k (1-\alpha)^{n-k} \tag{3}$$

It is trivial to show that the expected mean error for a given *R* is $\alpha/2$. Now it remains to determine the probability that there are a minimum of *k* transports within the radius *R*. This is achieved by subtracting from 1 the probability that there are less than *k* transports with the radius *R*. i.e.:

$$\text{Prob(at least k, n, } \alpha) = 1 - \left(\sum_{n=0}^{k-1} \binom{n}{k} \alpha^k (1-\alpha)^{n-k} \right) \tag{4}$$

Equation (4) can now be utilised to specify three variables and solve for the fourth. In the table, *n*, *k* and an acceptable probability are specified and solved for α.

Table 1. Derived α = R/L for parameters k, n and given probability.

Min # Desired Transports, k	**Probability**	**Transports, n**				
		5	**10**	**20**	**30**	**40**
2	**0.5**	0.31	0.16	0.08	0.055	0.041
	0.65	0.39	0.208	0.107	0.072	0.055
	0.8	0.49	0.27	0.142	0.096	0.073
3	**0.5**	0.5	0.258	0.131	0.088	0.066
	0.65	0.58	0.313	0.162	0.109	0.082
	0.8	0.67	0.38	0.202	0.137	0.104
4	**0.5**	0.686	0.35	0.18	0.12	0.091
	0.65	0.758	0.413	0.219	0.145	0.109
	0.8	0.831	0.483	0.258	0.176	0.134

The results from the Table 1 are not surprising. The more transports *n* there are in the scenario, the less the radius has to be to ensure at least two transports are within the specified radius. From the above simplified analysis a specific deduction would be a bold statement; however there are some indicators that the total number of transports needs to be relatively large, in the order of 10 as a minimum. For smaller numbers a derived α needs to be quite large, inferring a significant potential error cost.

5.2. Simulation and Results

From the previous section, it is observed that to enable the symmetry breaking technique – i.e. to have two or more similar transports within the radius and without the radius being excessively large (so as to not introduce too much error) – it seems there needs to be roughly ten or more transports in the scenario. Based on this analysis a number of simulations were run with varying numbers of transports, ranges and resources to be moved in order to determine how much computational savings can be made using this symmetry exploitation technique, and how much efficiency is lost in the distribution plan due to introduced error.

Sets of 1,000 simulation trials were run, for statistical significance, over a network of 10 nodes with 20 and 40 transports placed at random locations, and with ranges of 0, 10, 20, 30, 40, 50 and 60 units. These simulations were run over two scenarios, one scenario contained 2500 units of to be delivered, and the second scenario contained 5000 units of resources to be delivered. These resources are to be moved from the source node to the destination node, thus requiring the transports to do multiple trips to deliver the resources. A range value of 0 indicates that no symmetry breaking is used, and is the baseline to compare against the symmetry breaking technique. In the scenario the maximum distance between any two nodes is approximately 60 units, while the average distance is approximately 26 units. A range of 60 units encompasses the whole distribution network. Two types of transports were used in the scenario, a plane and a ship. The plane can travel ten times faster than the ship but can only carry one unit of a resource. The ship can only travel a tenth of the speed of the plane but can carry ten units of a resource. In each of the trials there is an equal ratio of planes to ships. In each of the trials one node is randomly set to be the destination node (i.e. where the resources are required to be moved), and one node is randomly set to be the source node (i.e. where the resources are initially held).

Once the scenario has been initialised, the greedy distribution algorithm as described in Section 3 was run in order to task the transports to complete the distribution of the resources. The average time taken to complete the distribution (in simulation time steps) and the computational time (in milliseconds) to complete the scenario was recorded for each trial. A simulation time step is defined such that a transport travelling at a speed of 10 units over a distance of 100 units would take 10 time steps. The scenario was designed to be feasible, such that in each trial it was possible for the transports to deliver all of the resources, thus completing a successful distribution. Figure 2 shows the average time taken to complete the distribution (in time steps) over 1,000 trials for 20 and 40 transports placed at random locations, and with range values of 0, 10, 20, 30, 40, 50 and 60 units, over two scenarios containing 2500 and 5000 units of resources to deliver. Immediately it can be seen that with more transports in the scenario, the faster the distribution is completed, as expected. However, it can also be seen that at range values of 10 or more the average simulation time increases. This indicates an increase in the error that is introduced, reducing the efficiency of the distribution plan. The spread of error across all of the range levels for scenarios with 20 transports on average was 0.3 - 5%, where for scenarios with 40 transports on average the error across the ranges was 4-18%. This gives a broad indication that with more transports more error is introduced. Additionally as the range value increases so does the variance, this occurs because as mentioned in section 4 after one calculation is done for a set of transports, a transport is chosen randomly to perform the movement. Therefore given that the transports can be at different locations, the error may not always be the same, and an error may not always be introduced.

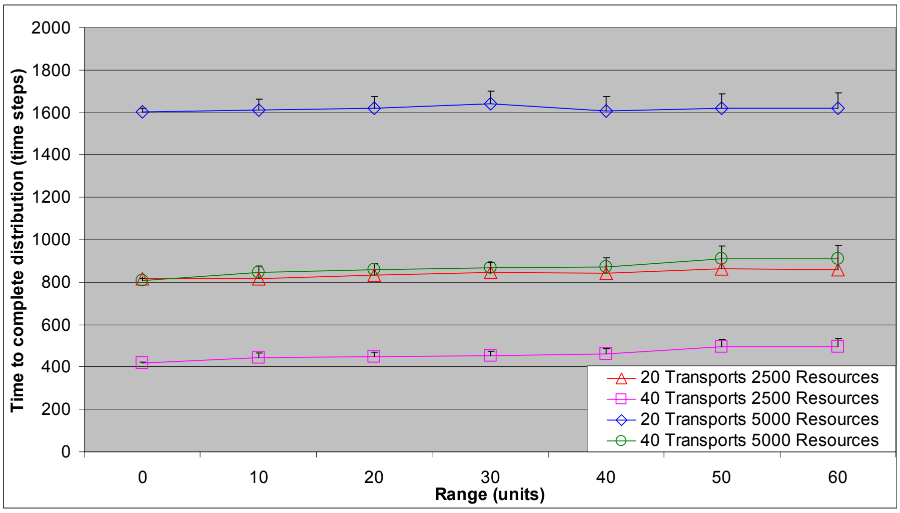

Figure 2. Average time to complete distribution (in time steps) taken over 1,000 trials.

Figure 3 shows the average computation time taken (in milliseconds) over 1,000 trials for 20 and 40 transports placed at random locations, and with range values of 0, 10, 20, 30, 40, 50 and 60 units, over two scenarios containing 2500 and 5000 units of resources to deliver. From the graph it can be seen that a range values of 10-40 units produces the most significant reduction in computation time, for 20 transports on average approximately 34-49%, and for 40 transports 24-44% reduction in computation across all ranges. Greater computational savings were made with scenarios with 5000 resources to be delivered. A range of 40 units produced the most savings in computation time, and for this distribution network only increased the distribution time by a maximum of ~10% across all scenarios.

Multiple sets of simulations were run on random distribution networks adjusting the range values according to the average and maximum distances between nodes, results were not displayed due to space limitations, however the same trend of results occurred. Simulations were also run testing scenarios with 250, 1000, 10000 and 20000 units of resource to be delivered. Results weren’t included due to space limitations and figure clarity; however it was found that computational gains of up to ~6%, ~25%, ~55% and ~59% were made respectively.

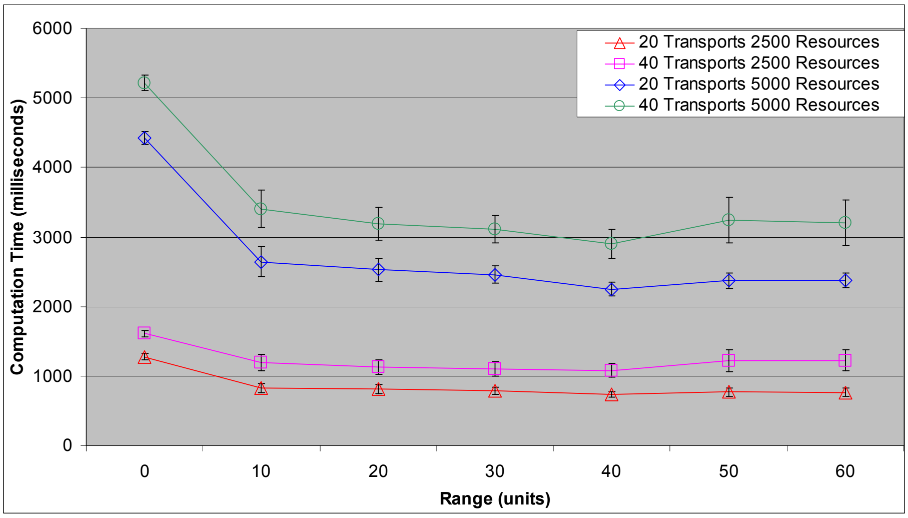

Figure 3. Average computation time (in milliseconds) taken over 1,000 trials.

6. CONCLUSION

In this paper the logistics distribution and movements planning problem was discussed. A greedy distribution algorithm designed to create transport schedules which satisfy all of the movement requirements was outlined. The concept and technique of exploiting symmetries in the distribution model was introduced. Experiments were run to test the hypothesis that exploiting redundancies caused by symmetries in the distribution model could save computation time in the algorithm. For the scenarios tested, the experiments indicated that across all ranges, computation time could be reduced on average by approximately 24-49%, while the efficiency of the distribution plan is only reduced by approximately 0.3-18%. These results signify that potentially significant computational savings could be made when solving larger distribution problems using the symmetry breaking technique.

REFERENCES

Cook, S. (1971), The complexity of theorem-proving procedures, In proceedings of Third Annual ACM Symposium on Theory of Computing, Ohio, USA, May 3-5.

Cormen, T. H., Leiserson, C. E., Rivest, R. L., and Stein, C. (1990), *Introduction to Algorithms*, p. 329, MIT Press and McGraw-Hill, USA.

Geiger, M. J. (2010), Fast approximation heuristics for multi-objective vehicle routing problems, Applications of Evolutionary Computation, 6025, 441-450.

Kulkarni, R. V. and Bhave, P. R. (1985), Integer programming formulations of vehicle routing problems, European Journal of Operational Research, 20(1), 58-67.

Laporte, G. (1991), The vehicle routing problem: An overview of exact and approximate algorithms, European Journal of Operational Research, 59, 345-358.

Ma, W. M., Dong, D. D., and Wang, K. (2010), Competitive analysis for the on-line vehicle routing problem, 4th International Conference on New Trends in Information Science and Service Science, Gyeongju, Korea, May 11-13.

Marsh, L. B. (2008), A study on optimisation techniques for logistics distribution. Masters thesis, The University of Adelaide, Adelaide.

Nilsson, N. J. (1980), *Principles of Artificial Intelligence*, pp. 72-88, Tioga Publishing Company.

Perugini, D. (2006), Agents for logistics: A provisional agreement approach. Ph.D. thesis, The University of Melbourne, Melbourne.

Ravindran, B. and Barto, A. G. (2001), Symmetries and model minimization in markov decision processes, Technical Report, University of Massachusetts, MA, USA.

Toth, P. and Vigo, D. (2002), *The Vehicle Routing Problem*, S.I.A.M., Philadelphia, USA.

Yeun, L. C., Ismail, W. R., Omar, K., and Zirour, M. (2008), Vehicle routing problem: models and solutions, Journal of Quality Measurement and Analysis, 4(1), 205-218.