Aggregate Formulations for Large-Scale Process Scheduling Problems

by

Stephen James Wilkinson

March 1996

A thesis submitted for the degree of Doctor
of Philosophy of the University of London and for the
Diploma of Membership of the Imperial College.

Department of Chemical Engineering and Chemical Technology,
Imperial College of Science, Technology and Medicine,
London, SW7 2BY, U.K.

LIBR. LONDIN. UNIV.

ABSTRACT

This thesis is concerned with efficient scheduling in production systems involving flexible multipurpose plants. Scheduling in such systems generally involves a complex set of decisions regarding the allocation of production resources over a time horizon in order to meet certain objectives such as the fulfillment of customer orders. These decisions interact with each other and must therefore be considered simultaneously if the best schedule is to be found. The best schedule is determined in relation to some desired criterion such as the maximisation of profit or the minimisation of costs.

Large-scale process scheduling problems often arise because of the need to model the system in detail. Problem size is further increased if there are decisions on widely differing time-scales (*i.e.* days *vs.* minutes) or if the problem involves many different resource types (materials, equipment items *etc.*). One class of large-scale problem which is given particular attention in this thesis is multisite scheduling involving several multipurpose plants and warehouses distributed over a wide geographical area. Both production and transportation issues need to be considered in this context.

In this thesis it is shown how many large-scale process scheduling problems can be represented in detail using a very general framework based on a discrete-time mathematical formulation. This *detailed* formulation, however, may be too large to be computationally tractable due to the factors discussed above. A general method for reducing problem size is therefore introduced in which a smaller *aggregate* formulation is derived directly from the detailed formulation. Because of its rigorous derivation, this aggregate formulation has certain useful properties which enable it to be a close approximation to the detailed formulation in many cases. It can therefore provide useful information relating to the solution of the original problem which was intractable in its detailed form. This is illustrated by several examples. The aggregate formulation has several potential applications which are also discussed in this thesis.

Acknowledgements

I would like to thank Dr. Costas Pantelides and Dr. Nilay Shah for their excellent supervision throughout my PhD work. I would also like thank Prof. Roger Sargent for accepting me into the Centre in the first place. I appreciate the patient efforts of Alexandre Cortier in helping me with the industrial case study in chapter 7, as well as the valuable work done by Andreas Dimitriadis in checking this thesis.

On the non-academic side of things, I would like to acknowledge the support of my parents and the rest of my family. Finally, many thanks to all the friends that I have made at Imperial College who have made my time in London such an enjoyable one.

Table of Contents

Abstract 2
Acknowledgements 3
Table of Contents 4
List of Figures 7
List of Tables 9

1. Introduction 10
1.1 Scheduling of Multipurpose Process Plants 10
1.2 The Benefits of Efficient Scheduling 11
1.3 Large-Scale Process Scheduling Problems 12
1.3.1 Multisite Production Systems 12
1.4 Outline of Thesis 15

2. Literature Review 17
2.1 Aggregation Techniques in Mathematical Programming Formulations 18
2.2 Hierarchical Production Planning 20
2.3 Multisite Planning 21
2.4 Process Scheduling Formulations 23
2.4.1 Aggregate Process Scheduling Formulations 25
2.5 Concluding Remarks 26

3. The RTN Framework for Process Scheduling 28
3.1 Resources and Tasks 28
3.2 Discrete-Time RTN Formulation 28
3.3 Example I 31
3.4 Example II 38
3.5 RTNs for Multisite Scheduling 41
3.5.1 Third Party Transportation and Quantity Discounts 42
3.5.2 Mixed Lorry Loads 43
3.5.3 Material Incompatibilities 45
3.5.4 Warehousing Considerations 46
3.6 Concluding Remarks 48

4. Aggregation Techniques for RTN Formulations 50

4.1 Basic Ideas 51
4.2 A General Approach to the Aggregation of Discrete-Time Formulations 52
4.3 Excess Resource Balance Aggregation 54
4.3.1 Zeroth Order Aggregation 54
4.3.2 Higher Order Aggregation 57
4.4 Aggregation of other Constraints 59
4.4.1 Excess Resource Capacity Constraints 59
4.4.2 Extent Non-Negativity Constraints 62
4.4.3 Operational Constraints 65
4.4.4 Higher Order Aggregate Excess Resource Capacity and Operational Constraints 69
4.5 Aggregate Objective Function 71
4.6 Constructing Aggregate Formulations 72
4.7 Concluding Remarks 73

5. Theoretical Properties of the Aggregate RTN Formulations 74
5.1 Property I: Strict Relaxation of Detailed Formulation 74
5.2 Property II: Reduction to Detailed Formulation for $h \rightarrow 1$ 76
5.3 Property III: Reduction to Detailed Formulation for $m \rightarrow h-1$ 79
5.3.1 Excess Resource Balances 79
5.3.2 Capacity and Operational Constraints 80
5.4 Property IV: Maximal Aggregation Order 84
5.5 Property V: Covering Property for Capacity and Operational Constraints 88
5.6 Illustrative Example 89
5.7 Improving the Accuracy of the Aggregate Formulation 93
5.7.1 Cut Constraints 93
5.7.2 Detailed Portions in the Time Horizon 93
5.8 Concluding Remarks 94

6. Discussion 96
6.1 Physical Interpretation of Zeroth and First Order Aggregation 96
6.2 Importance of the Linking Variables 100
6.3 The Effect of ATP Length and Overall Order on Problem Size 103
6.4 Comparison with the Aggregate Formulation of Bassett *et al.* (1994) 104
6.4.1 Treatment of Material Resources 105
6.4.2 Treatment of Processing Equipment Resources 107
6.4.3 Remarks 109
6.5 Applications of the Aggregate Formulation 110

6.5.1 A Decomposition Approach for Generating Detailed Schedules 111
6.5.2 A Rolling Horizon Approach for Generating Detailed Schedules 112
6.6 Concluding Remarks 113

7. Planning of Europe-Wide Multisite Production and Distribution: An Industrial Case Study 115
7.1 Problem Outline 116
7.1.1 Overview 116
7.1.2 Process Description 117
7.1.3 Distribution of Materials 119
7.2 RTN Formulation 120
7.2.1 Time Discretisation 120
7.2.2 Size of RTN 120
7.3 Solution Strategy 122
7.4 Comparative Example 123
7.4.1 RTN Description 124
7.4.2 Problem Specification 126
7.4.3 Results 127
7.5 Overview of Results for Full Multisite Case Study 133
7.5.1 Stage I: Aggregate Solution for Multisite System 133
7.5.2 Stage II: Detailed Production Schedules for Individual Plants 136
7.6 Concluding Remarks 138

8. Conclusions and Further Work 140
8.1 Contribution of this Thesis 140
8.2 Some Potential Directions for Further Work 141
8.2.1 Finding Detailed Schedules for Large-Scale Problems 142
8.2.2 Allowing for Uncertainty 143
8.2.3 Wider Applications of the Aggregation Approach 144

References 145

Appendices 150
A. Derivation of the General Order Aggregate Excess Resource Balance 150
B. Summary of the First and Second Order Aggregate RTN Formulations 154
C. A Matrix Derivation of the Aggregate Excess Resource Balance Constraints . . . 161
D. Modelling Tasks with Different Time-Scales 168
E. Cut Constraints to Account for Start-Up Effects 173

List of Figures

Figure 1.1: A multisite production and distribution system.

Figure 3.1: Uniform discretisation of the planning horizon.

Figure 3.2: State-Task Network of process for Example I.

Figure 3.3: RTN of process for Example I (omitting operator resource).

Figure 3.4: Gantt chart for optimal solution to Example I.

Figure 3.5: Utilisation of equipment and manpower for optimal solution to Example I.

Figure 3.6: Variation of stock levels for optimal solution to Example I.

Figure 3.7: State-Task Network of process for Example II.

Figure 3.8: Effect of the distillation task on the cooling water availability.

Figure 3.9: RTN representation of transportation options using three tasks.

Figure 3.10: RTN for a transportation task with mixed loads in fixed proportions.

Figure 3.11: RTN for a transportation task with mixed loads in variable proportions.

Figure 3.12: RTN representation of transportation of incompatible materials.

Figure 3.13: Illustration of sub-division of warehouse capacity.

Figure 4.1: Aggregation of the planning horizon.

Figure 4.2: Weighting functions for $p = 0$, 1 and 2.

Figure 4.3: Weighting functions for first order aggregate excess resource capacity constraint.

Figure 5.1: Solution spaces of the detailed and aggregate formulations.

Figure 5.2: Comparison of detailed and aggregate ($h = 8$, $m = 2$) solutions to Example I.

Figure 5.3: Variation of optimal objective value with the number of ATPs for Example I.

Figure 5.4: An aggregate planning horizon with detailed start-up and shut-down portions.

Figure 6.1: Variation of feed level for illustrative example.

Figure 6.2: The first order weighting function and the concept of resource-time.

Figure 6.3: Illustration of the function served by linking variables.

Figure 6.4: Illustration of a rolling horizon approach to generating detailed schedules.

Figure 7.1: Geographical scope of the case study (only countries directly involved are labelled).

Figure 7.2: Processing structure for some products made at the Italian plant.

Figure 7.3: RTN representation of change-overs on packing lines *A* and *B*.

Figure 7.4: Comparison of SKU tonnages for the detailed and aggregate solutions.

Figure 7.5: Deliveries to the warehouses in the producing countries as determined by the aggregate solution.

Figure 7.6: Deliveries to the warehouses in the non-producing countries as determined by the aggregate solution.

Figure 7.7: Histogram showing achievable production at each plant.

Figure 7.8: Utilisation of some equipment in the detailed solution for the UK plant.

Figure 8.1: Directions for further fundamental research on aggregate RTN formulations.

List of Tables

Table 3.1: Resource utilisation coefficients for the *Blending* task.

Table 3.2: Resource utilisation coefficients for the *Storage* task.

Table 3.3: Resource utilisation coefficients for the *Re-tooling 1 to 2* task.

Table 3.4: Resource utilisation coefficients for the reaction task.

Table 3.5: Resource utilisation coefficients for the distillation task.

Table 3.6: Haulage company price quote.

Table 3.7: RTN transport task parameters.

Table 5.1: Data for aggregate solutions to Example I (no margin of optimality used).

Table 5.2: Results for aggregate formulation with detailed constraints at start and end of horizon.

Table 7.1: Resources in the RTN for the multisite case study.

Table 7.2: Tasks in the RTN for the multisite case study.

Table 7.3: Details of the four task types in the comparative example.

Table 7.4: Initial availabilities and capacities for the equipment resources.

Table 7.5: Computational statistics for the detailed and aggregate solutions to the comparative example.

Table 7.6: Demands for each SKU and the actual production for the detailed and aggregate solutions.

Table 7.7: Continuous and discrete (in brackets) task extents for detailed and aggregate solutions.

Table 7.8: Deliveries from plants to warehouses in the aggregate solution.

Table 7.9: Computational statistics for the aggregate solution.

Table 7.10: Computational statistics for the detailed solutions to the sub-problems.

Chapter 1

Introduction

1.1 Scheduling of Multipurpose Process Plants

This thesis is concerned with efficient scheduling in multipurpose process plants. These differ from continuous process plants, such as those used for oil refining or bulk chemical production, in that they are generally highly *flexible*. They are often able to produce a wide range of different products as required using a variety of production routes. This flexibility is often gained by using batch processing, although some production activities may be continuous (*e.g.* packing of final product items). These characteristics make multipurpose plants particularly effective for the manufacture of product types with a large degree of inherent diversity and which are subject to time varying customer demands. Such products include paints, foods, consumer products and speciality chemicals.

An analogy can be drawn between the operation of a multipurpose plant and that of a kitchen in a restaurant where orders for meals have to be met during the course of an evening. In the case of the kitchen, there is a finite set of resources (chefs, saucepans, cookers, kitchen implements *etc.*) which are used to convert the set of available foodstuffs into the finished meals via a set of pre-specified tasks given in the recipes. Different meals are likely to have some ingredients and intermediates in common, as well as sharing similar equipment items in their preparation. In a similar way, multipurpose plants have the flexibility to produce a wide range of products in an order-driven environment. A set of common production resources (equipment items, manpower, utilities *etc.*) are used to convert the raw materials into the final products. The processing stages involved in the production of each final product are generally known but the specific equipment items which will actually carry out these tasks are not. This is how the flexibility of the system arises.

Given such flexibility, scheduling is concerned with finding the *best* (*e.g.* minimum cost, maximum profit) way of assigning the production resources to tasks over the time horizon of interest in order to meet the required production objectives (*e.g.* fulfill the customer demands). For many multipurpose plants, this represents a highly complex problem given the large number of decisions that an operating schedule involves.

1.2 The Benefits of Efficient Scheduling

The benefits of finding an optimal (or close to optimal) schedule are obviously determined by the criterion which one is trying to optimise. Identifying the benefits of finding the minimum cost way of doing something, for example, is not hard — the benefit is that it minimises costs! However, as well as being desirable on an operational level, efficient scheduling can also bring less immediate or quantifiable (but, in some cases, more significant) benefits in terms of a company's strategic policy. Some important implications of efficient scheduling are given below:

1. *Economies of scale.*

   In many cases, there are significant cost savings that can be derived from the different parts of a production system co-operating to meet certain goals or sharing commonly required resources. For example, an equipment item such as a packing line that is not fully utilised by one part of the production system may be used to pack products from another part. Quantity discounts for procurement of raw materials from suppliers or transportation of products to customers may also make closer co-ordination desirable. The scale of such co-operation may vary from closer harmonisation of different production lines in the same plant up to interactions between different plants on an international or even global basis.

2. *Customer satisfaction.*

   The timely fulfillment of orders is essential in a competitive environment where a customer can turn to another supplier if orders are late. In many cases, demand patterns are time dependent and unpredictable over the longer term. A plant, therefore, may schedule its activities on a short-term (daily or weekly) basis in order to be able to adapt to such changes. For periods of heavy demand, efficient schedules will enable the maximum number of orders to be met on time. Another important feature that customers look for is a short lead-time from making their order to receiving it. By using efficient scheduling, a company can offer faster service and therefore gain a significant competitive advantage.

3. *Product Diversification.*

   Companies often need to tailor their products to the customers' individual requirements to increase or maintain their market share. In the detergents sector, for example, the large producers offer a range of different brands, each having different variants, pack sizes *etc*. Such product diversification, therefore, is desirable in marketing terms but it

also introduces a higher degree of operational complexity in the manufacture of these products. A company with a capability for efficient scheduling of its production activities will be able to cope better in this respect.

1.3 Large-Scale Process Scheduling Problems

Process scheduling problems are inherently complex due to their combinatorial nature which stems from the large number of discrete decisions that are involved at each stage of the time horizon considered. There are, however, certain features in some problems which make them an order of magnitude larger than many of those commonly encountered. Clearly, if a problem involves lengthy process recipes involving many different types of equipment items, this will increase the size of the problem. However, even comparatively simple processing structures can give rise to large-scale problems if they involve highly flexible equipment items. Suppose an equipment item can perform a total of n different tasks and that change-over activities (*e.g.* cleaning) are required between each. There will then be a total of $n^2 - n$ possible change-over activities that can be performed. If each task is performed once in the horizon, there are a total of $n!$ possible task sequences. If there are many such equipment types, then a very large combinatorial optimisation problem will result.

Multisite production systems which involve multipurpose plants also give rise to large-scale scheduling problems. These are given particular attention in this thesis (see chapters 3 and 6) and are described in more detail below.

1.3.1 Multisite Production Systems

For the purpose of this thesis, a multisite system will be defined as a collection of geographically distinct sites at which material is either stored or converted to other materials. These sites co-operate with each other in order to meet certain required objectives.

The most obvious example of a co-ordinated multisite system is the network of plants and warehouses operated by a single manufacturing company. This may have arisen because the company has inherited existing sites after acquiring other operations, or it may be economically advantageous to have several production and storage facilities, each near to centres of demand. The geographical extent of such a system could range from a regional level (*e.g.* in the case of a small-scale food producer) up to a truly global level for a multi-national chemical company.

Multisite scheduling is a topic which is currently attracting a great deal of interest since increased co-ordination at an operational level can often lead to significant benefits. Flaherty (1986) studied five multi-nationals (two of which were chemical companies) and concluded that increased co-ordination should not only give significant reductions in costs, but also enhance their effectiveness in the market place. For organisational and other reasons, there may be some necessity for a degree of autonomous management at each site. On the other hand, the need for a co-ordinated response to diverse customer demands and the desire to minimise costs, imply that ideally the various plants and warehouses should be treated as parts of one large production system. Planning for this system would be carried out centrally, allowing full interaction between all the sites and consequently the most efficient utilisation of the available resources. For example, a product of one multipurpose plant may be used as one of the feedstocks of another, and it may sometimes be beneficial for plants operating similar processes to exchange intermediates. This is illustrated in figure 1.1. In this case, the multisite system consists of 3 plants and 7 warehouses serving customer demands and shows a set of possible routes along which materials may be transported. The plants procure the necessary raw materials from a set of suppliers (dashed lines) and then convert them into products which are transported to the warehouses (dotted lines). The plants may well operate very similar processes; consequently it could be desirable for them to share intermediates, or for a product from one plant to form the feedstock for another. Such exchanges correspond to the solid lines in the diagram.

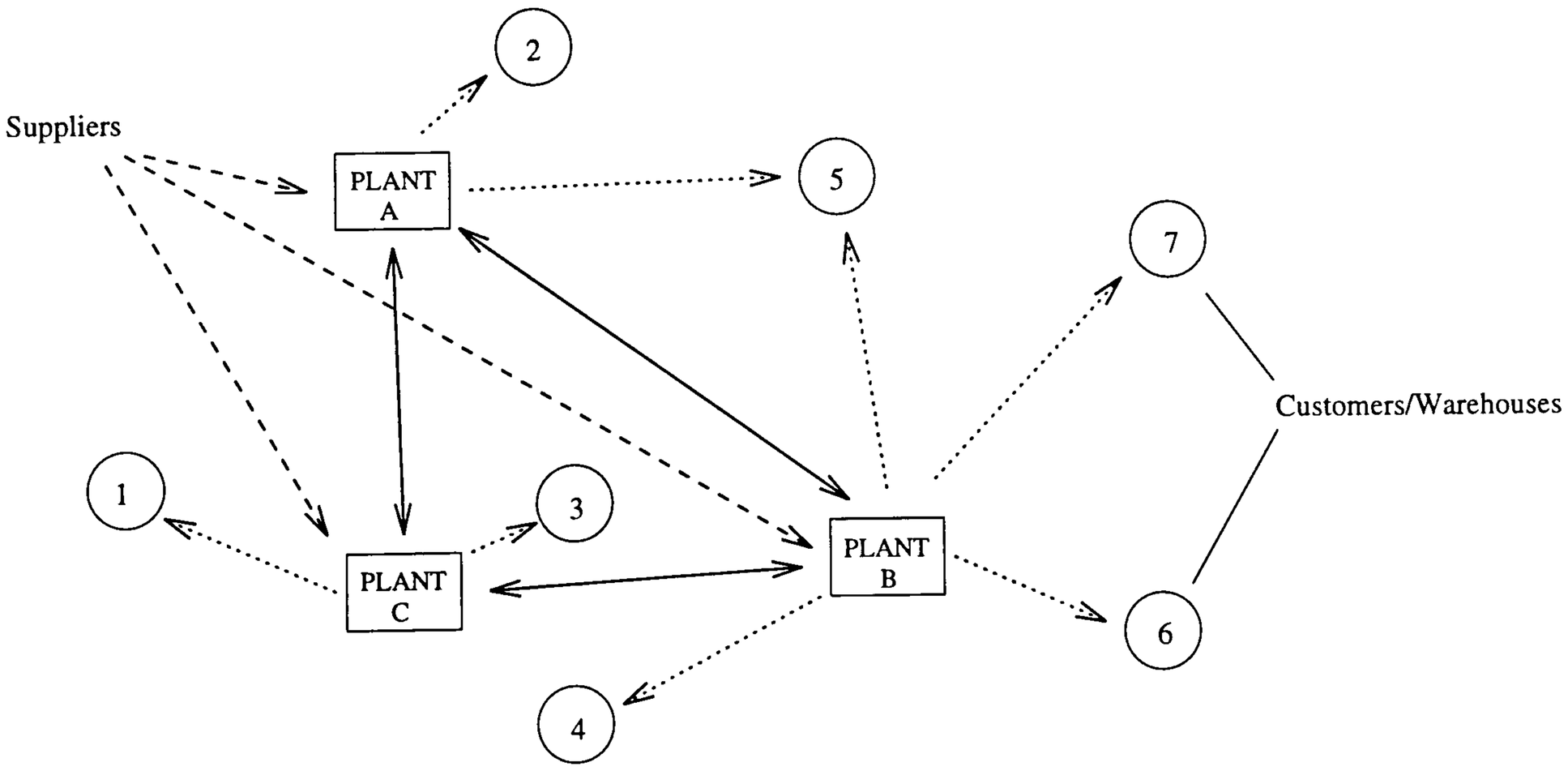


Figure 1.1: A multisite production and distribution system.

In addition to multisite systems operated by a single company, there are an increasing number of examples where separate companies have benefitted from co-ordinating their activities in agreements and partnerships. Large retailers, for example, may benefit from working closely with their suppliers in such areas as distribution, warehousing and demand forecasting. Norman and Ramirez (1993) describe the success of companies who have co-ordinated their activities with those of their suppliers and customers. Even directly competing companies may see an advantage in limited co-operation. An example of this is the direct buying and selling of stocks ('exchanges') between large oil companies which is sometimes carried out to reduce distribution costs. This may arise when a company needs to meet demand in an area which is far away from its own refinery but close to one of its competitors.

The phrase 'supply chain' is used to cover all the activities involved in satisfying customer demand, from procuring raw materials through to coverting them into products and distributing them to warehouses and demand centres. As well as costs associated with production such as labour, utility and raw material costs, those associated with distribution and storage can be significant proportions of the cost of supplying the product to the customer. Cooper (1987) studied several large food and drink companies and found that distribution costs were, on average, 9.4% of their total sales revenue. Many companies work with profit margins of less than 5% and consequently only a small reduction in costs can lead to a significant competitive advantage.

The activities with which these costs are associated interact with each other and must therefore be considered simultaneously. For example, maintaining higher stock levels has a buffering effect which can smooth the demand on the production facilities but causes increased stock-holding costs. Similarly, a warehouse can reduce the average amount of stock held (and hence its costs) by increasing the frequency of its re-order policy. This, however, may be at the expense of increased transportation costs.

In a highly competitive and constantly changing market environment, it is even more important to have a high degree of co-ordination between all the supply chain activities. It is widely recognised that there is a great deal of potential for reducing costs in many areas if more efficient scheduling methods can be found which consider the multisite system in its entirety. A very good indicator of the amount of inefficiency present in most multisite production and distribution systems is the total amount of stock held. Of course, holding large stocks can help decouple the different supply chain activities reducing the effect that one has on the other. There are, however, unit costs associated with holding stock which can be very significant. Edwin L. Artz, Chairman and Chief Executive of Procter and Gamble, summed up the difficulties and trade-offs involved in operating a multisite system in 1992:

> *"Swings in price create variability and massive inefficiency in the manufacturing and distribution system. To meet heavy short-term demand, manufacturers must build raw material inventory and design peaks and valleys in production schedules, which entail excessive warehousing costs, excessive shipping rates and new handling costs. As an industry we've estimated annual grocery sales in the United States of 300 billion U.S. dollars. Out of that, there's between 75 billion to 100 billion U.S. dollars of inventory, much of it unproductive, caught between the manufacturer, the manufacturer's supplier and the retailer — up to a third of total sales just trapped in the pipeline."*†

Excessive stock levels, therefore, incur a quantifiable cost that can be greatly reduced by increased co-ordination of the supply chain activites.

1.4 Outline of Thesis

The objective of this thesis is to develop a general method for obtaining approximate solutions to large-scale process scheduling problems. Chapter 2 contains a brief review of some work relating to aggregation in mathematical programming formulations. Since multisite systems are given particular attention in this thesis, a review of work in the logistics and supply chain planning area is also included. Finally, literature relating to detailed and aggregate formulations for large-scale process scheduling problems is discussed.

Chapter 3 reviews the Resource-Task Network (RTN) process representation of Pantelides (1994) which is a very general representation allowing a wide range of problems to be modelled. A discrete-time fomulation based on this representation is presented which involves very few constraint and variable types. Finally, some examples are presented to illustrate the general modelling capability of this formulation. It is this formulation that forms the basis of the aggregate formulation derived in chapter 4. It is shown how a family of aggregate formulations can be derived directly from the detailed RTN formulation by summing constraints and introducing aggregate variables.

The aggregate RTN formulation has some potentially useful mathematical properties that allow the trade-off between accuracy of solution and problem size to be exploited. These are discussed in chapter 5 with the aid of an example. Chapter 6 discusses some applications of the aggregate formulation, along with schemes for generating feasible detailed schedules

† Taken from Fuller *et al.* (1993).

from the aggregate solution. A comparison with a recently proposed formulation for multi-purpose plant scheduling (Bassett *et al.*, 1994) is also presented in this chapter.

The aggregate formulation is applied to an industrial case study in chapter 7. This is concerned with production and distribution scheduling for a multisite system of plants and warehouses that span Western Europe.

Finally, chapter 8 summarises the contributions of this thesis and indicates some directions for further research.

Chapter 2

Literature Review

When faced with a problem that cannot be solved when expressed at the desired level of detail, some simplification must be found to convert it into a form that *is* tractable and which still yields valuable information about the solution of the original problem. A perfectly natural way to proceed when faced with such a large problem is to find a way of breaking it up into smaller sub-problems which can then be solved separately. This method often works quite adequately when the critical interactions in the system can be determined, indicating which parts can satisfactorily be considered in isolation from the whole. In these cases, the decomposed solution may be very close (or even identical) to the solution of the original problem.

For problems that can be expressed in terms of a mathematical programming formulation, another approach is to aggregate constraints and/or variables to create, in effect, another problem which is small enough to be solved. Depending on the nature of the problem and the form of aggregation used, this solution may then provide a reasonable approximation to the fully detailed solution. This is the approach adopted in this thesis for large-scale process scheduling problems which are generally too complicated to be tackled by an intuitive decomposition method.

The complexity of process scheduling problems, in common with other decision-making problems, arises from their combinatorial nature as was illustrated briefly in section 1.3. Karp (1972) has shown that many such problems, the travelling salesman problem being a well-known example, are *NP-complete i.e.* they cannot be solved exactly by any algorithmic method of polynomial complexity. Scheduling in multiproduct plants (in which all products follow the same route through the plant) is another problem that falls into this category (Gabow, 1983). Multiproduct plants are a less general case of the multipurpose plants and therefore the general multipurpose plant scheduling problem considered in this thesis can also be regarded as being NP-complete. Finding an exact solution, therefore, may only be possible for small problems (Garey and Johnson, 1979), since it requires some kind of enumerative approach. Most published work therefore aims to use heuristic polynomial algorithms to find 'good' solutions (*i.e.* with objective values that can be shown to be tolerably close to that of the true optimal solution).

In this chapter we shall briefly review the literature on four areas that are relevant to the work in this thesis. Firstly, some previous work on formal aggregation techniques in

mathematical programming will be examined. Hierarchical production planning in which various aggregation and disaggregation techniques are utilised will also be considered. Also, since multisite planning problems are given particular emphasis throughout this thesis, some work in this area will also be reviewed. Finally, as the vast majority of the earlier research carried out in the first three areas is specific to assembly or machine shop manufacturing systems, work specific to the modelling and solution of *process* scheduling problems is considered. Multipurpose process plants are fundamentally different in nature from factories that manufacture items by machining and assembling various component parts. Optimal planning in such flexible plants therefore necessitates a different approach.

2.1 Aggregation Techniques in Mathematical Programming Formulations

Zipkin (1980a) considers aggregation of variables in a general linear programming formulation of the form:

$$z^* = \max cx \ , \ (Ax \leq b \ , \ x \geq 0) \tag{2.1}$$

where c is an n-dimensional vector; b is an m-dimensional vector; and A is an $m \times n$ matrix. Variable aggregation is achieved by partitioning the n-vector of variables x into a set of subvectors $\{x_k: k = 1, .., K\}$, each of which is then replaced by a single aggregate variable. The aggregate formulation is then written as:

$$\overline{z} = \max \overline{c}\overline{X} \ , \ (\overline{A}\overline{X} \leq b \ , \ \overline{X} \geq 0) \tag{2.2}$$

where $\overline{c}$ is a K-dimensional vector, each element of which ($\overline{c}_k$) is a weighted sum of a subset of the elements of the original vector c that correspond to those in the sub-vector x_k. Likewise the matrix $\overline{A}$ is an $m \times K$ matrix formed from weighted sums of the columns in the original matrix A corresponding to those in the sub-vector x_k. The same weightings are used in both cases. It is shown that, if a feasible solution to (2.2) is found, then the n-dimensional vector formed by disaggregating the aggregate variables (using the same weighting functions as were used to generate $\overline{c}$ and $\overline{A}$) is feasible with respect to the original problem (2.1).

Such an aggregation of variables places extra restrictions on the variables of (2.1). Even though (2.1) may have a feasible solution, therefore, it may not necessarily give rise to an aggregate problem (2.2) that is feasible. Error bounds on the accuracy of the optimal aggregate solution with respect to the original problem are derived assuming certain collective bounds on the variables in each partition and the dual of the aggregate solution. Optimal weightings are suggested to obtain a 'good' aggregate formulation (*i.e.* to minimise this

error).

In direct contrast to the aggregation described above, aggregation of constraints rather than variables gives rise to a formulation which is *less* constrained than the original problem. An error analysis similar to that introduced above can also be carried out when the variables and constraints in a general linear programming problem are aggregated simultaneously (Zipkin, 1980b).

Geoffrion (1977) considers a specific linear programming formulation for a procurement planning problem. A set of facilities is considered that procure a range of items from a set of suppliers in order to meet customer demands. No production activities are considered in this model (*i.e.* those that transform one product type into another). The objective is to minimise the overall procurement cost subject to meeting the demands. An aggregate formulation is produced by aggregating constraints of different item types. Aggregate variables are then introduced, corresponding to summations of variables for a subset of product types. In common with other approaches where different product types are aggregated, each aggregate variable represents a simple summation (*i.e.* weighting functions are all unity) of disjoint groups of product types. Error bounds on the accuracy of the aggregate formulation and necessary conditions for optimal grouping of product types are also derived.

Aggregation of constraints has been exploited in solution algorithms for integer programming problems. Much of this work is based on a theorem due to Matthews (1897) who considers two linear constraints:

$$\sum_{j=1}^{n} a_{1j}x_j = b_1 \qquad \& \qquad \sum_{j=1}^{n} a_{2j}x_j = b_2 \tag{2.3}$$

where $x_j \geq 0,\ j = 1,..,n$ are integer unknowns and the coefficients a_{ij} and b_i are positive integers. It is shown that the equation:

$$\sum_{j=1}^{n}(a_{1j}t_1 + a_{2j}t_2)x_j = b_1t_1 + b_2t_2 \tag{2.4}$$

has the same solutions as (2.3) if t_1 and t_2 are suitably chosen positive integers that are prime relative to each other. In most solution algorithms for integer programming problems that are based on this property, constraints are sequentially aggregated to eventually form a single constraint that has the same solutions as all the original ones. Bradley (1971) demonstrates how such a scheme can be used to convert any bounded, equality constrained integer programming problem into an equivalent knapsack problem with the same number of variables but only one constraint.

The papers discussed above serve to illustrate the wide body of work that has been done on aggregation in mathematical programming. Rogers (1991) gives a full review of aggregation and disaggregation methodology in optimisation.

The work described in this section is mainly concerned with solving abstract mathematical problems with little attention being paid to the application of these techniques to realistic examples such as the process scheduling problems addressed in this thesis. These often result in mixed integer linear programming (MILP) formulations and it is not obvious how aggregation of variables as proposed by Zipkin (1980a,b) for linear programming problems can be effectively used in this case. In addition, any techniques for obtaining error bounds that rely on the dual solution would not be appropiate for MILP problems. Where a real life problem has been addressed (Geoffrion, 1977), the formulation bears little relation to those required for process scheduling problems which are time-dependent rather than steady state in nature.

2.2 Hierarchical Production Planning

Hierarchical Production Planning (HPP) is a name connected to a fairly diverse body of work developed by several authors over the last two decades to tackle problems involving large-scale assembly or machine shop manufacturing systems. Such systems generally have a fixed 'bill of materials' where, for any intermediate item in the manufacturing process, it is known exactly which items will be involved in its production and which items will themselves require it for their own production. The basic idea of this approach is to identify a hierarchy of decisions on different levels which can be associated with aggregate groups of items which share certain characteristics.

Bitran and Hax (1977) identify three levels of product aggregation: *items* which are the end products delivered to the customers; *families* which are groups of items which have similar set ups (*i.e.* when a machine is set up for one item, it is set up for all the others in its family); and *product types* which are groups of items having similar production costs. For single stage manufacturing systems, Bitran *et al.* (1981) present a hierarchical approach which determines aggregate quantities (*e.g.* total production levels within each family) at one stage and then passes them on to another for disaggregation. This method has also been extended for systems involving two production stages (Bitran *et al.*, 1982).

Axsäter (1981) considers product aggregation in a model which is typical in this field and which is governed by the following equality constraint:

$$y_t = y_{t-1} + x_t - Ax_t - d_t^e \qquad \forall\ t \tag{2.5}$$

where $y_t = (y_{t1}, .., y_{tN})^T$ is a vector of inventory levels for all N products involved in the system in period t; x_t is the corresponding vector of product orders initiated in period t; and d_t^e is the external demand vector for the products. The $N \times N$ matrix $A = (a_{ij})$ $i, j = 1, .., N$ reflects the fixed interdependence of the items in the manufacturing process, each item of product j requiring a_{ij} items of product i for its manufacture. An important assumption implicit in (2.4) is that all orders are initiated and completed within the same period t. Aggregation into groups of products can be carried out to generate constraints of lower dimensionality. The author presents conditions for so-called 'perfect aggregation' when the aggregate constraint is fully consistent with the detailed constraint.

In a two-level hierarchical approach, Axsäter (1986) identifies constraints which, when applied to the aggregate level, will guarantee a feasible solution at the detailed level. Axsäter and Jonsson (1984) use simulation experiments for an industrial example in order to evaluate disaggregation procedures and methods for aggregating items.

Das and Sarin (1994) consider aggregate planning in integrated manufacturing systems. They use a master schedule in which capacity, time and products are aggregated to synchronize product activities with demand.

Although the work on HPP addresses realistic manufacturing problems, it is of limited significance for process scheduling problems involving flexible plants where a given product can be often produced by a variety of different production resources. In such a case, it is impossible to sum up the interdependencies of all resources involved in the problem by a single matrix A as in (2.5) above. In the same way, it is very often not meaningful to group resources into families in multipurpose plants.

2.3 Multisite Planning

When considering a multisite system of plants and warehouses operating on an international or even global basis, material transportation and storage activities need to be taken into account, as well as those associated with production at each plant. This section will present a brief survey of work that has considered the interaction between two or more of the following supply chain activities: distribution, inventory control and production. The work is highly varied, both in the nature of the problems that are examined, and in the methods of analysis. It should be stressed that, in all the work, the production models used were rather simplistic. The objective is to minimise distribution or inventory holding costs (or both) but production costs are not included. This is indicative of the fact that the production models are so simple. In particular they contain none of the detail needed to represent the many choices and degrees

of freedom that exist in flexible production facilities.

Anily and Federgruen (1993) do not consider production scheduling at all but give detailed consideration to interactions between the cost of distribution and inventory holding. They consider problems with up to 500 demand points and a central warehouse in a problem that may be faced by a large retail chain, for example. Travelling Salesman concepts are used to derive bounds on the minimal transportation and storage costs incurred when using individual vehicle subtours for the delivery of a single commodity.

Zuo *et al.* (1991) seek to minimise overall production and distribution costs for a case study of seed hybrid production in the United States. They use Mixed Integer Programming and consider the steady state allocation of seeds to processing facilities. This leads to the conclusion that it is not always optimal to process seeds in the nearest facility to where they were produced.

The highest level of detail for production used in the literature surveyed was that by Blumenfield *et al.* (1985). They consider different production runs, each producing a single product, with set up costs between each run. They extend the *Economic Order Concept* to generate an explicit expression for the total costs. The optimal production cycle time and shipment size are then found by calculus. Cohen and Lee (1988) decouple the supply chain into material control, production, and distribution submodels that are linked by constraints. The cost incurred in meeting customer demands is then minimised using heuristic methods.

Williams (1981) surveys 7 different heuristic techniques of varying sophistication for production-distribution scheduling in multi-echelon structures. Different path structures from raw materials to final customer demand points are considered: 'assembly' (each node having exactly one successor node); 'arborescent' (each node having exactly one predecessor node); and co-joined assembly-arborescent. Customers are considered to have steady state demands for the final products. It was found that no one heuristic worked better in all examples, but that some consistently gave better results than the methods for scheduling actually employed by companies.

Ishii *et al.* (1988) consider a dynamic market situation where a company makes one product that is reaching the end of its life (whose demand is decreasing linearly) and is being replaced by a new product (whose demand is increasing linearly). They develop a method to find base stock levels and leadtimes to keep the probability of stockouts below a required level whilst minimising the amount of unsaleable 'dead stock' when demand for the old product has dropped to zero.

Conway (1988) considers the interaction of production activities with work-in-process inventory. Generic production systems comprising sets of workstations in series were studied

by computer simulation in conditions of variable process times and unreliability of equipment. It was found that, not only was the size of the inventory buffers between each workstation important, but also their position in the production line.

Computer simulations of production-distribution systems have also been carried out. Dorairaj (1989) developed a multiperiod model of a multisite system with stock re-allocation decisions to be made each day. He uses an inventory cost model that includes back-ordering costs as well as carrying costs. The aim of his method is to minimise total system costs by finding suitable warehousing operating policies. Mohanty and Chandrasekhar (1983) simulate the case of a single factory supplying 3 regional warehouses. Some autonomy is allowed at each site to decide transportation policy. A feedback control model is developed to derive optimal operational parameters for the system. The production planning side of the model is concerned with workload balancing (hiring, firing and overtime costs are considered). Runs are carried out with stochastic demands and fixed customer service levels.

So-called 'lumpy' demand patterns on a plant are often encountered in practice and can cause major problems for the production planner. They arise when there is insufficient co-ordination of the individual parts of the system. A production facility and the final demand points are separated by links which may be warehouses, transshipment centres or other production facilities. Each link places a demand on the previous link (in order to satisfy the demands placed on it), with no regard for the scheduling of that previous link. Edgehill *et al.* (1988) examine the dynamics of the supply chain and how variations in demand at retail outlets can be amplified by the time they have been transmitted back to the production facility. They suggest feedforward as well as feedback control loops to smooth out the demand.

The papers referred to above are illustrative of a large body of work in the literature which addresses the efficient design and/or operation of large-scale distribution systems. Issues such as vehicle routing and optimal re-ordering policies are obviously important in any multisite system with distributed plants, warehouses and customers. However, the lack of attention to the details of production makes them unsuitable for systems involving multipurpose process plants which are the subject of this thesis. An approach is needed which simultaneously considers the allocation of production resources at each plant and the wider supply chain activities such as product distribution and storage.

2.4 Process Scheduling Formulations

Flexible multipurpose plants can accommodate the production of several different products in the same facility. They are often associated with batch processing although some of their

operations may be continuous in nature. In general, equipment items can be used for different types of task and each task can be performed in different equipment items of the same or different types. For example, several similar mixing vessels may be used to blend any one of number of intermediates. Hence, different products may be produced by following the same route through the plant and batches of the same product may have followed different routes. Examples of such plants include those that produce food/dairy products, paints, detergents, and pharmaceuticals.

Their inherent flexibility means that the scheduling of multipurpose plants involves a combinatorial problem of high dimensionality. Compared to the other activities in the multi-site system (*i.e.* those associated with distribution and storage), the production scheduling problem often involves the allocation of far more resource types (intermediates, equipment items, utilities *etc.*).

Much work has been carried out on the detailed scheduling of multipurpose plants. Reviews of the literature in this area have recently been published by Reklaitis (1992) and Rippin (1993). In deriving aggregate scheduling formulations that will be of wide applicability, we are primarily interested in general frameworks for the scheduling of multipurpose plants. The recent work in this area is reviewed below.

Kondili (1988) and Kondili *et al.* (1993) presented a mathematical formulation based on the *State-Task Network* (STN) process representation which allows a detailed description of the operation of multipurpose plants. A solution algorithm based on it has been shown to be effective for solving a variety of realistic short-term planning problems (Shah *et al.*, 1993). More recently, Pantelides (1994) presented a unified framework based on a *Resource-Task Network* (RTN) representation of the process in which all resources (equipment items, storage capacity, utilities *etc.*) are treated uniformly. This generic approach facilitates taking account of complicating production features such as sequence-dependent cleaning, as well as maintenance and wider supply chain activities, all in a unified and consistent manner.

Efficient solution strategies have been presented by authors who seek to increase the capability of discrete-time process scheduling formulations for tackling large-scale problems. Sahinidis and Grossmann (1991) present an approach in which the constraints in the STN formulation of Kondili *et al.* are disaggregated. Although this increases the number of variables and constraints in the MILP problem, it can give a reduction in the integrality gap (and hence the solution time) for certain classes of problem. Shah *et al.* (1993) demonstrate how reformulation of constraints of the STN formulation of Kondili *et al.* and pre-processing of problem information can significantly reduce solution times and enable previously intractable problems to be solved. Elkamel (1993) presents a variety of measures for developing efficient

solution strategies for discrete-time process scheduling formulations. These include model pre-processing to eliminate certain variables at the initial stage of the solution procedure; heuristics to decompose the problem into a set of sub-problems; and the use of strong valid inequalities which exploit the structure of the problem.

All of the work described above is based on formulations employing a discrete representation of time. More recently, work has been aimed at producing formulations based on a *continuous* time representation. Pinto and Grossmann (1994) describe one such formulation for cyclic scheduling of multiproduct plants comprising a sequence of stages interconnected by storage tanks. Zhang (1995) and Zhang and Sargent (1996) present a general formulation based on the RTN framework of Pantelides. They demonstrate how a mixed integer linear programming (MILP) formulation can be generated by linearising constraints. A similar general continuous-time formulation is presented by Schilling and Pantelides (1996). They propose a novel branch-and-bound algorithm which aims to address the computational problems caused by the large integrality gap inherent in this and other related formulations.

The papers above illustrate the very considerable recent progress in process scheduling methods that has enabled many problems to be solved that were considered intractable only a few years ago. There is, however, a limit to the extent to which the efficiency of these algorithms can be improved. The large-scale problems considered in this report often require some approximate solution method. The work described below attempts to use aggregate formulations to give a significant reduction in problem size whilst giving meaningful results.

2.4.1 Aggregate Process Scheduling Formulations

There are very few papers which present aggregate formulations that are specific to process scheduling problems. In the area of process synthesis rather than scheduling, Daichendt and Grossmann (1994) have developed a preliminary screening procedure by aggregating the variables in the mixed integer non-linear programming (MINLP) problem formulation to give a mixed integer linear programming (MILP) formulation. This is of lower dimension and provides a rigorous upper bound to the global optimal solution of the original MINLP problem.

Musier and Evans (1988) emphasise the cost of holding inventory and present an aggregate multi-period formulation which simultaneously minimises production and distribution costs over an intermediate planning period. They suggest a discretisation interval of the order of weeks or months. The formulation is a linear programming problem and ignores discrete decisions such as equipment set ups and change-overs. It is assumed that these decisions can be dealt with by a lower level scheduling formulation.

Liu and Sahinidis (1995) consider a long range MILP formulation for a system involving several chemical processes and in which capacity expansions or shutdowns are allowed during the course of the planning horizon. They present an aggregation-disaggregation heuristic for large-scale planning problems. An aggregate problem is first solved, in which time periods are aggregated into groups. This effectively restricts the times at which changes to the system can be implemented over the planning horizon. Then the detailed model can be solved as a linear programming problem with all the integer decision variables fixed at the values given by the aggregate solution. This yields a feasible but not necessarily optimal solution to the original problem. Finally, to demonstrate the effect of the discretisation interval on the integrality gap of the detailed formulation they present a series of examples. These have a total planning horizon of 12 years and discretisation intervals ranging from 1 to 12 years.

Aggregation over time is used by Bassett *et al.* (1994) to form a higher level planning problem based on a uniform time discretisation model. This forms part of decomposition strategy whereby production is allocated to different time zones, thus creating a set of detailed scheduling problems that can be solved independently to produce a global schedule. Their aggregate formulation bears some relation to the one derived in this thesis and is therefore discussed more fully in chapter 7.

2.5 Concluding Remarks

In this section we have reviewed four areas in the literature which are related to the work in this thesis. The work on formal aggregation in mathematical programming is relatively limited in the scope of the aggregation it considers despite the fact that very general original problems are considered. Much work has been done on the formal derivation of error bounds for the aggregate formulation and finding the best weightings to use. However, the accuracy of the aggregate formulations is limited since only single summations of disjoint groups of variables and/or constraints are considered.

A similar criticism could be applied to the work on hierarchical production planning which adopts a similar approach of single weighted summations. In addition there is an exclusive emphasis on assembly type manufacturing systems where techniques such as the aggregation of products into families can be meaningful. This work is not directly applicable to the case of scheduling in multipurpose plants and also uses a simplistic form of aggregation.

The work on scheduling of supply chain activites, whilst highlighting the trade-off between distribution and inventory costs, devotes no attention to problems requiring detailed production models. It is therefore difficult to see how multipurpose plants could be

incorporated into these methodologies, indicating that a different approach is required.

The aggregate process scheduling formulations presented in this thesis are based on a detailed discrete-time RTN formulation (Pantelides, 1994) which is discussed more fully in the next chapter. The RTN framework has two very important properties. Firstly, it is a very general representation which allows a wide range of different scheduling problems to be tackled. Secondly, as a result of its generality, formulations based on this framework are very simple, involving only a few different constraint types. This aids in the derivation of aggregate formulations *via* the mathematical manipulation of these constraints.

Chapter 3

The RTN Framework for Process Scheduling

This chapter outlines the Resource-Task Network (RTN) framework recently presented by Pantelides (1994). This is a unified framework for process scheduling which captures many of the complicating features involved in the operation of multi-purpose plants. We start by introducing the basic concepts of resources and tasks, and the way they can be used to derive very general, yet simple, mathematical formulations for process scheduling problems. Two examples are presented to illustrate these ideas. It will then be demonstrated how wider supply chain activities such as distribution and warehousing can also be modelled within the RTN framework.

3.1 Resources and Tasks

The key property of the RTN framework is its uniform description of all *resources* involved in the process. No conceptual distinction is made between, for example, raw materials, processing equipment items, utilities and transportation resources.

The other fundamental concept is that of the *task* which is an abstract operation that consumes and/or produces a set of *resource types* and is uniquely defined by that set. Resources are classified into the minimum number of resource types such that any part (or element) of a resource type is functionally equivalent to any other. Equipment items of different capacities, for example, are regarded as being in different resource types, even though they may be used for exactly the same transformations of material.

Functional equivalence within resource types depends on the level of detail demanded by the scheduling application. In a problem involving equipment cleaning considerations, for example, two equipment items may belong to the same resource type only if they are in the same condition (*e.g.* 'clean' or 'soiled'). In multisite scheduling, *location* is obviously a crucial feature which distinguishes different resource types. In other words, a material at a site A is considered to be in a different resource type to the same material at a site B.

3.2 Discrete-Time RTN Formulation

Based on the RTN process representation, different formulations can be derived based on the representation of time chosen. Zhang and Sargent (1996) present one based on a continuous

time representation. As well as determining when to perform each of the processing tasks, their algorithm also allows certain decisions that are intrinsic to each task (such as its duration) to be included in the optimisation process. Pantelides (1994) presents a formulation based on a uniform, discrete representation of time (see figure 3.1) which is the one that will be discussed for the remainder of this chapter. Unlike the formulation of Zhang and Sargent, it assumes that the process recipe is completely fixed. Despite this assumption, processing tasks can still be modelled in detail as the example given in section 3.4 demonstrates. Importantly, it ensures that the resulting mixed integer programming problems are *linear*, allowing relatively large-scale problems to be tackled. In addition, linear, discrete-time formulations can be used to generate *aggregate* formulations with useful properties. These are discussed in the next two chapters.

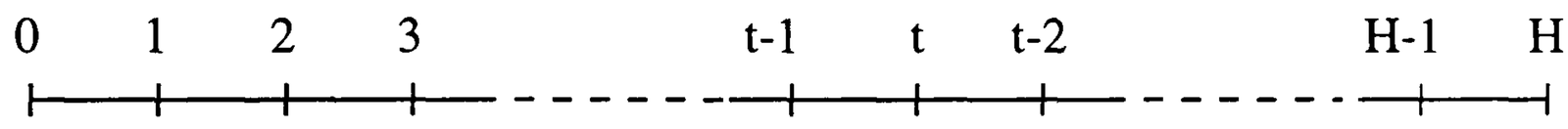


Figure 3.1: Uniform discretisation of the planning horizon.

The overall planning horizon is divided into H equal time intervals. The time discretisation is made fine enough such that all tasks can always be considered to start and end on a time interval boundary. Each task k has a fixed duration τ_k and the execution of task k starting at time t is characterised by a pair (N_{kt}, ξ_{kt}), where N_{kt} is an integer variable and ξ_{kt} a continuous one. This pair is called the *extent* of the task k starting at time t.

Resources are produced and consumed at discrete times during the execution of the task. The amount of resource type r produced at a time θ relative to the start of task k at time t is:

$$\mu_{kr\theta} N_{k,t-\theta} + \nu_{kr\theta} \xi_{k,t-\theta} \tag{3.1}$$

These coefficients allow tasks such as those with time-dependent manpower or utility requirements to be modelled, as well as those which involve a simple transformation from a feed to a product using an equipment item.

Changes to the resource utilisation can occur only at interval boundaries. The variable R_{rt} denotes the amount of *excess resource* r over time interval t (*i.e.* the amount that is not involved in the tasks that are active over this time interval). The change in the excess resource level for each resource type from one time interval to the next is given by *excess resource balance* constraints of the form:

$$R_{rt} = R_{r,t-1} + \sum_{k} \sum_{\theta=0}^{\tau_k} \left(\mu_{kr\theta} N_{k,t-\theta} + \nu_{kr\theta} \xi_{k,t-\theta} \right) + \Pi_{rt} \qquad \forall\ r, t \tag{3.2}$$

where Π_{rt} is the amount of resource r made available from external sources at a time t. The quantity R_{r0} corresponds to the amount of resource r available initially and is assumed to be given.

We also need to ensure that the amount of excess resource at any time is within certain levels. This can be written as follows:

$$0 \le R_{rt} \le R_r^{\max} \qquad \forall\ r, t \tag{3.3}$$

The upper bound $R_r^{\max}$ may, for instance, reflect limited storage capacity for an intermediate material in the process. In this case, the lower bound on R_{rt} is zero but it may be greater than this (*i.e.* $R_r^{\min} \le R_{rt}$) if a minimum inventory of material is required to guard against uncertainty in the process. This is similar to 'safety stocks' which are often maintained by warehouses (see section 3.5.4).

Individual tasks may be subject to operational constraints. These are normally expressed in terms of the task extent. For instance, the continuous extent of a batch processing task corresponds to the amount of material undergoing the task at any given time. This is constrained by the integer extent of the task which can be interpreted as the number of processing equipment items being used by the task at that time. The necessary capacity constraints can be written as:

$$V_{kr}^{\min} N_{kt} \le \xi_{kt} \le V_{kr}^{\max} N_{kt} \qquad \forall\ k, t, r \in PE_k \tag{3.4}$$

where PE_k is the set of processing equipment resource types used by task k, and $V_{kr}^{\min}$ and $V_{kr}^{\max}$ are respectively lower and upper limits on the corresponding batch-sizes. We shall see examples of more complex operational constraints later in this chapter (section 3.5.2).

Finally, the *objective function* will typically be a linear combination of task extent and excess resource level variables. In general, there will be fixed and also variable (*i.e.*batch-size dependent) costs associated with each processing task. There will also be costs associated with the use of utilities and possibly costs for holding inventories of process materials. Assuming that we wish to maximise the total value-added at the end of the planning horizon ($t = H$), we can write the following objective function:

$$\max \sum_{r \in MR} \left(C_r^F (R_{rH} - R_{r0}) - C_r^I \sum_{t=1}^{H} R_{rt} \right) - \sum_{k} \sum_{t=1}^{H} (C_k^F N_{kt} + C_k^V \xi_{kt}) - \sum_{t=1}^{H} C_{rt}^E \Pi_{rt} \tag{3.5}$$

where MR is the set of material resources in the problem and C_r^F is the end-of-horizon value

of each of these resources $r \in MR$. Each material resource also has an associated inventory holding cost C_r^I. It should be noted that, if the cost of holding stock can be assumed to be due only to lost interest on the capital it ties up, then C_r^I and C_r^F are related:

$$C_r^I = C_r^F \times \frac{IR}{100\%} \times \frac{\delta}{365 \times 24} \tag{3.6}$$

where IR is the annual percentage interest rate and δ is the length, in hours, of the time discretisation interval being used in the formulation. For intermediates, however, it may be difficult to establish any realistic sale value C_r^F, so we may wish to set it to zero for these resources. On the other hand, C_r^I should be set to a positive value in order to prevent too much stock of intermediates being held during the time horizon.

Returning to the objective function (3.5), C_k^F and C_k^V are the fixed and variable costs, respectively, for each task k. These costs may arise from the utility or manpower requirements of the tasks; the depreciation of equipment items; or from charges quoted by a third party operator (see section 3.5.1). Finally, the last term refers to the cost of receiving resources from external sources, or, if Π_{rt} is negative, it becomes the revenue generated by delivering resources to external sources.

To illustrate the basic RTN concepts and the formulation presented above, two examples are presented below.

3.3 Example I

Consider the simple process represented as a State-Task Network (Kondili *et al.*, 1993) in figure 3.2. Two ingredients are blended and the resulting product is packaged to give either a 1 *kg* or a 2 *kg* pack-size as the final products.

There are two identical batch blenders, each of capacity 5 tonnes, available for the blending task. This task requires one operator per blender used and has a duration of 2 hours. The unpacked product can be stored in a silo of capacity of 5 tonnes while awaiting packing. The packaging tasks are both carried out by a single continuous flexible packaging line which can perform only one task at any given time and has a capacity of 3000 packs per hour for both pack-sizes. A re-tooling task taking 3 hours and requiring an operator must be carried out whenever the packaging line is to switch from one task to the other.

In the discrete time formulation, continuous tasks can be approximated as being batch tasks with a duration equal to the chosen time discretisation interval. In this case we shall use a time discretisation interval of one hour. This means that the packaging tasks are taken to be

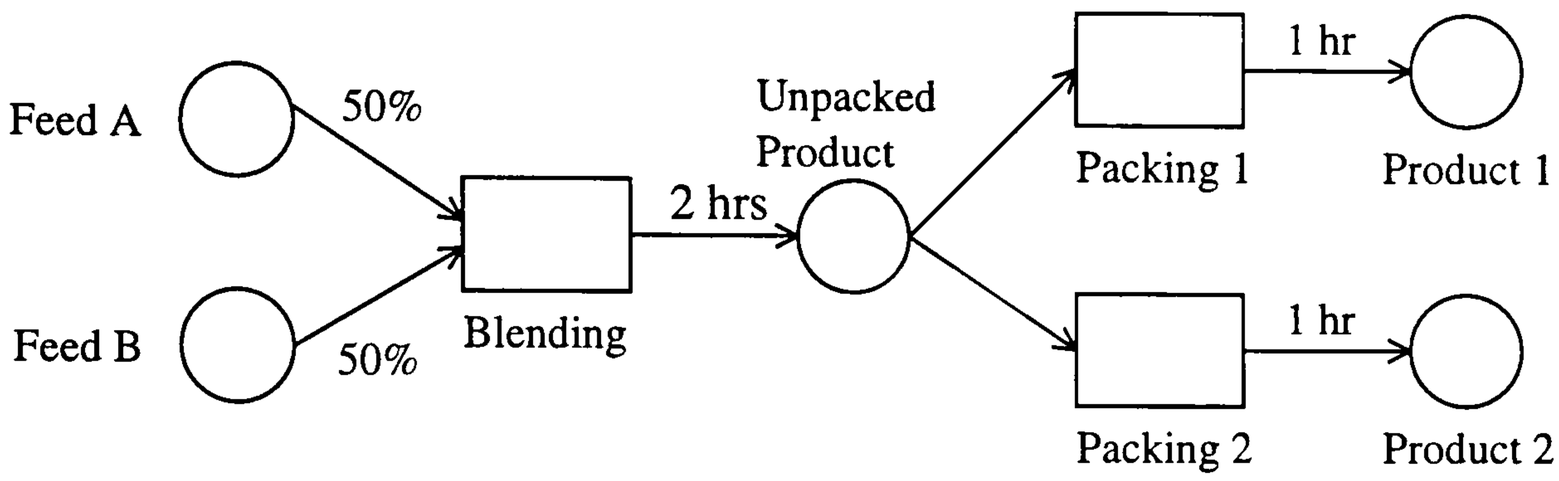


Figure 3.2: State-Task Network of process for Example I.

batch tasks with capacity 3000 packs and a duration of one hour. In a similar way, the storage of the unpacked product in the silo is modelled in the RTN formulation by a storage task of duration one hour. This implies that changes to the amount of material stored in the silo can occur only on an hourly basis.

The above process corresponds to the RTN shown in figure 3.3, with circles denoting resource types, and rectangles corresponding to tasks. Note that the operator resource which is required for the blending and re-tooling tasks has been omitted from this diagram for the sake of simplicity.

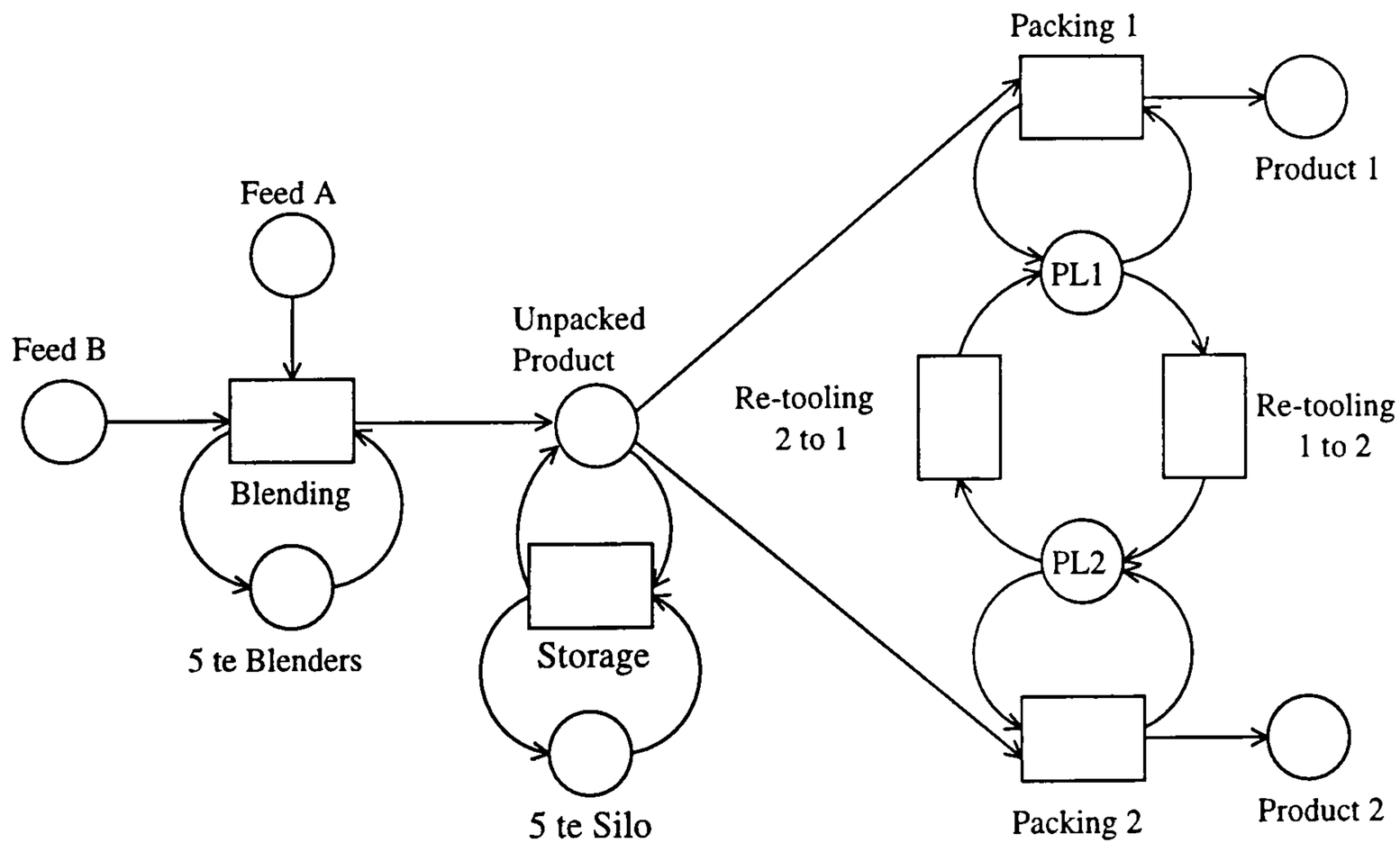


Figure 3.3: RTN of process for Example I (omitting operator resource).

The packing line is considered to be in one of two distinct resource types at any time while not performing a task. These are *PL*1 and *PL*2 which correspond to the line being tooled up to perform *Packing 1* or *Packing 2* respectively. Appropriate re-tooling tasks can transform one of these resource types to the other.

As examples of RTN tasks, consider the *Blending*, *Storage* and *Re-tooling 1 to 2* tasks whose resource utilisation coefficients are given in tables 3.1, 3.2 and 3.3 respectively (*cf.* the expression given by (3.1)).

In the blending task, the two feeds are consumed in equal proportions at the start of the task to produce the unpacked product at the end. This gives rise to the values of the coefficients $\nu_{kr\theta}$ shown in the first three rows of table 3.1. The equipment items (in this case, the blenders) are considered to be 'consumed' at the start of the task and 'produced' at the end, as are the operators. Again, this corresponds to the $\mu_{kr\theta}$ values shown in the last two rows of table 3.1. In the storage task, the equipment item (the silo in this case) is also used for the duration of the task (*i.e.* 'consumed' at the start and 'produced' at the end). So also is the material that is undergoing the task (the unpacked product). The re-tooling task is considered to 'consume' the Packing Line in one state *PL*1 and 'produce' it in another *PL*2 three hours later. It also uses one operator for its duration in the same way as the blending task.

Given the process recipe and the available equipment item resources outlined above, we can now define a scheduling problem. The additional information we need to give is:

- the length of the planning horizon;
- the initial and maximum allowable stock level for each material resource;
- the availability profile of any utilities required (manpower in this case);
- the initial state of those equipment items that can be in more than one possible state;
- the magnitude and timing of any demands/deliveries;
- cost data relating to tasks, material storage, utilities and end-of-horizon inventories.

We shall now consider an illustrative scheduling problem. The planning horizon is one day (24 hours) long. Initially there are 60 tonnes of both *Feed A* and *Feed B* with no unpacked or packed product in stock. There is unlimited storage capacity available for the feeds and packed products but the *Unpacked Product* can only be stored in the silo. We therefore consider that the *Unpacked Product* must be stored only by undergoing a storage task and that its excess resource level must always be zero. Two operators are available for the duration of the planning horizon and the packing line is initially tooled up to pack the 2 kg pack size. There is single demand of 20 tonnes for the 1 kg product (*i.e.* 20,000 packs) which

$k = Blending \ (\tau_k = 2)$						
	$\mu_{kr\theta}$			$\nu_{kr\theta}$		
θ:	0	1	2	0	1	2
$r = Feed\ A$	0.0	0.0	0.0	-0.5	0.0	0.0
$r = Feed\ B$	0.0	0.0	0.0	-0.5	0.0	0.0
$r = Unpacked\ Product$	0.0	0.0	0.0	0.0	0.0	1.0
$r = Blenders$	-1.0	0.0	1.0	0.0	0.0	0.0
$r = Operators$	-1.0	0.0	1.0	0.0	0.0	0.0

Table 3.1: Resource utilisation coefficients for the *Blending* task.

$k = Storage \ (\tau_k = 1)$				
	$\mu_{kr\theta}$		$\nu_{kr\theta}$	
θ:	0	1	0	1
$r = Unpacked\ Product$	0.0	0.0	-1.0	1.0
$r = Silo$	-1.0	1.0	0.0	0.0

Table 3.2: Resource utilisation coefficients for the *Storage* task.

$k = Re\text{-}tooling\ 1\ to\ 2 \ (\tau_k = 3)$								
	$\mu_{kr\theta}$				$\nu_{kr\theta}$			
θ:	0	1	2	3	0	1	2	3
$r = PL1$	-1.0	0.0	0.0	0.0	0.0	0.0	0.0	0.0
$r = PL2$	0.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0
$r = Operators$	-1.0	0.0	0.0	1.0	0.0	0.0	0.0	0.0

Table 3.3: Resource utilisation coefficients for the *Re-tooling 1 to 2* task.

occurs at $t = 18$ hours.

Due to the short length of the horizon, inventory costs are omitted. Also, there are considered to be no costs associated with any of the tasks. Each of the two feeds costs £100 per tonne and the packaging materials cost £200 per tonne of final product for both 1 kg and 2 kg

pack size. Finally, at the end of the horizon, the 1 kg and 2 kg packed products can be sold for £400 per tonne and £500 per tonne respectively. Since the external demand at $t = 18$ hours is for a fixed amount, its value is not included in the objective function. The objective is to maximise the total value of the stock held at the end of the planning horizon, comprising both products and feeds.

The optimal solution to this problem gives a profit of £20,100 with the end-of-horizon stock levels for the 1 kg and 2 kg products being 1.0 and 50.0 tonnes respectively†. The other contribution to the total value of stock held comes from the unused feeds whose stock levels are both reduced to 24.5 tonnes by the end of the planning horizon. Of the material that is made for stock, the 2 kg product is made in preference to the 1 kg product because, on a weight basis, it has a higher value and can also be produced at a higher rate. Despite this fact, 1.0 tonne of the 1 kg product is produced surplus to the demand of 20.0 tonnes. This is due to the fact that the 1 kg product can be produced by the packing line at the rate of 3000 packs (3.0 tonnes per hour) and the discretisation interval used for the problem is one hour. Given that all tasks must start and end on a time interval boundary, it can be seen that 7 hours of packing time are required to meet the 20.0 tonne demand. In this time, a total of $7 \times 3.0 = 21.0$ tonnes of 1 kg product can be produced which results in 1.0 tonne (which still has value, though not as much as the 2 kg product) left in stock at the end of the horizon after the demand has been met.

The optimal schedule that the optimal solution represents can be shown in the form of a Gantt chart as in figure 3.4. It shows which tasks are being performed by which equipment items at each time in the planning horizon. It is interesting to note how the *Blending* task is 'staggered' or out-of-phase for the two blenders, whenever their operations overlap. This is due to the limited intermediate storage — the 5 tonne silo can take only one full blending task batch-size (5 tonnes) at any one time.

Despite the fact that the two blenders have been disaggregated into *Blender (a)* and *Blender (b)* for the purposes of drawing the Gantt chart, it should be stressed that no distinction is made between them in the RTN formulation since they are functionally identical. The optimal solution merely gives the starting time and extent (discrete and continuous) of each blending task occurrence. This information has then been used to generate *a posteriori* one of several operating policies for the two blenders that can be directly implemented in practice. For instance, the schedule shown in figure 3.4 was generated using the rule that *Blender (a)* should perform as many of the tasks as possible, with *Blender (b)* taking over any extra tasks that *Blender (a)* could not perform. As an example, consider the first 6 hours of the schedule.

† The optimal solution to this problem and all others presented in this thesis was obtained using the SCICONIC MILP code (SD-SCICON, 1990).

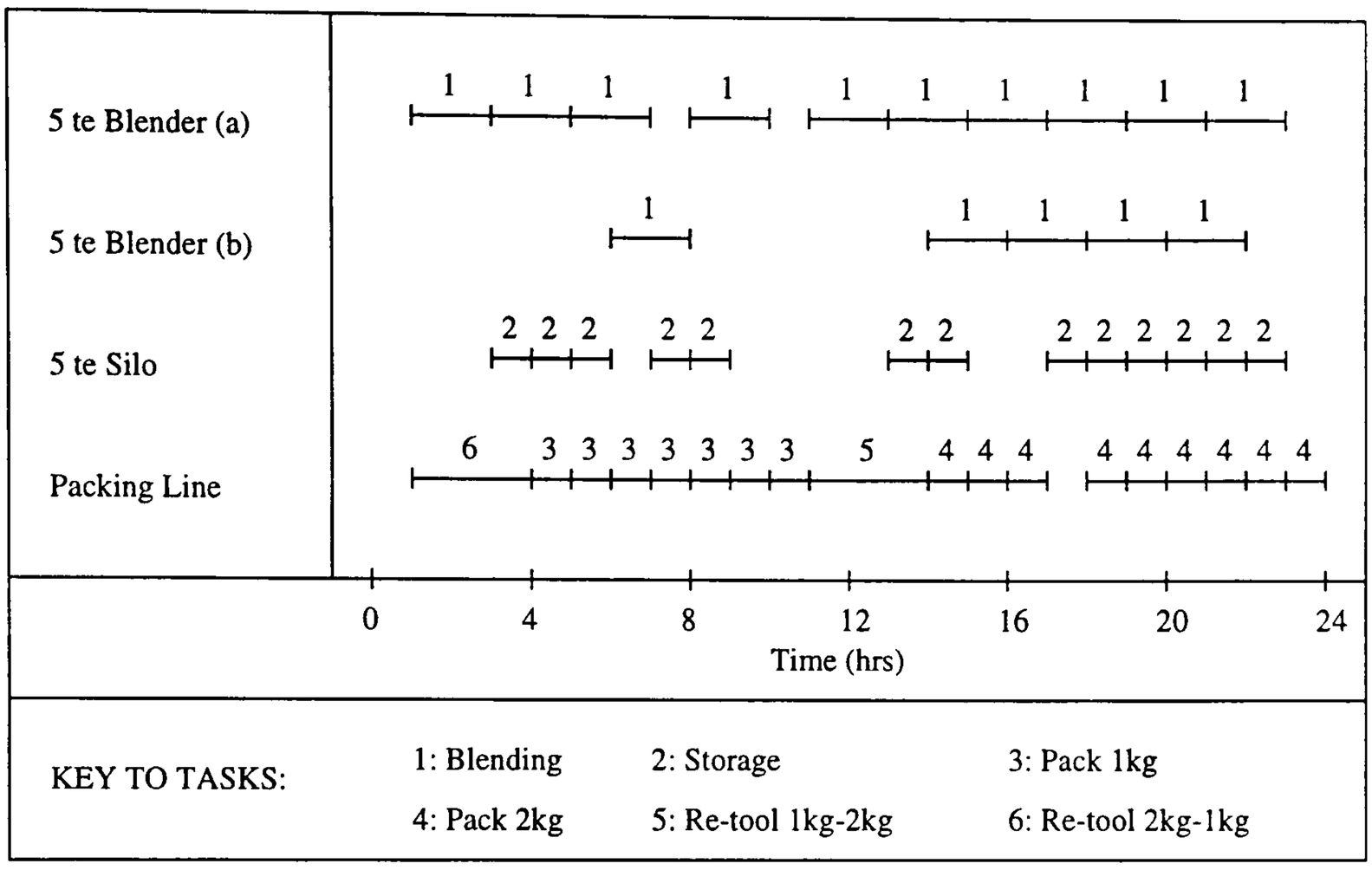


Figure 3.4: Gantt chart for optimal solution to Example I.

The RTN solution has blending tasks, each of discrete extent equal to one, starting at times 1, 3, 5 and 6. Since each task is two hours long, it can be seen that *Blender (b)* must perform the task starting at $t = 6$ if *Blender (a)* performs those starting at $t = 1$, $t = 3$ and $t = 5$.

The equipment and manpower utilisations are given in figure 3.5 with the maximum availability for each given by the dashed lines. The packing line is considered as being utilised when it is being re-tooled as well as when it is actually engaged in packing.

If figures 3.4 and 3.5 are considered together, it can be seen that the 'bottle-neck' resources or activities vary over the course of the planning horizon. In the first part of the horizon (up to $t = 9$), when the 1 kg product is being made, the packing line is fully utilised whereas the blenders are idle for some of the time. Here we could say that the packing line is the bottle-neck or rate determining resource. This is due to the fact that that the blenders can produce *unpacked product* at a higher rate than can be packed by the packing line. After the packing line has been re-tooled, however, and the 2 kg product is being made, the packing line can pack twice the rate of unpacked product (since it packs a constant number of actual packs per hour) and it is the blenders that become the bottle-neck with the packing line having an hour's idle time at $t = 17$. Such variable bottle-necks are common for problems with time dependent demands for different products.

Figure 3.6 gives the variation in stock levels for four of the material resources involved. The unpacked product is not included because its stock level is always zero due the

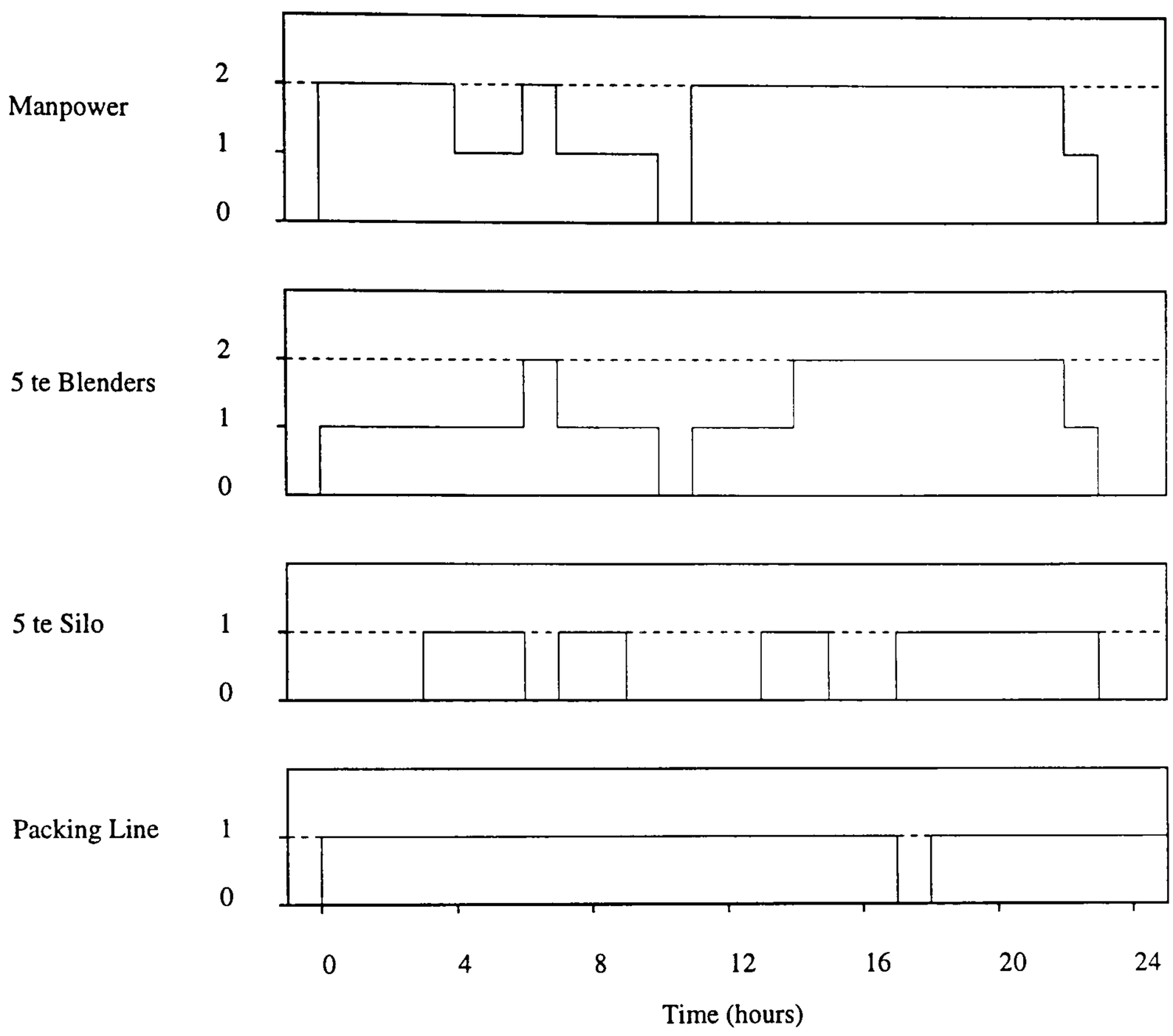


Figure 3.5: Utilisation of equipment and manpower for optimal solution to Example I.

silo storage task explained above. Notice that no product appears until $t = 5$. This is due to the fact that the RTN formulation adopts the convention that the time horizon starts at $t = 1$, at which time the packing line immediately undergoes a 3 hour re-tooling (or change-over) task in order to start packing the 1 kg product — it is initially tooled up to pack the 2 kg product. Packing therefore only starts at $t = 4$ resulting in the first appearance of product at $t = 5$. By $t = 11$, enough 1 kg product has been made to meet the demand and the packing line undergoes another 3 hour re-tooling task. This takes it back to the state where it can produce the more valuable 2 kg product which therefore begins to appear at $t = 15$ since packing of it starts at $t = 14$. Notice, also, how the stock level of the 1 kg product drops when the demand for it occurs at $t = 18$.

This example illustrates the general modelling properties of the RTN formulation in which all resources are treated uniformly allowing complicating features such as limited intermediate storage and equipment change-overs to be taken into account. It also demonstrates just how these features and others such as time dependent demands can have a significant impact on the form of the optimal solution to a scheduling problem. Even for such a small system, somewhat counter-intuitive results can arise, such as the fact that a total of two

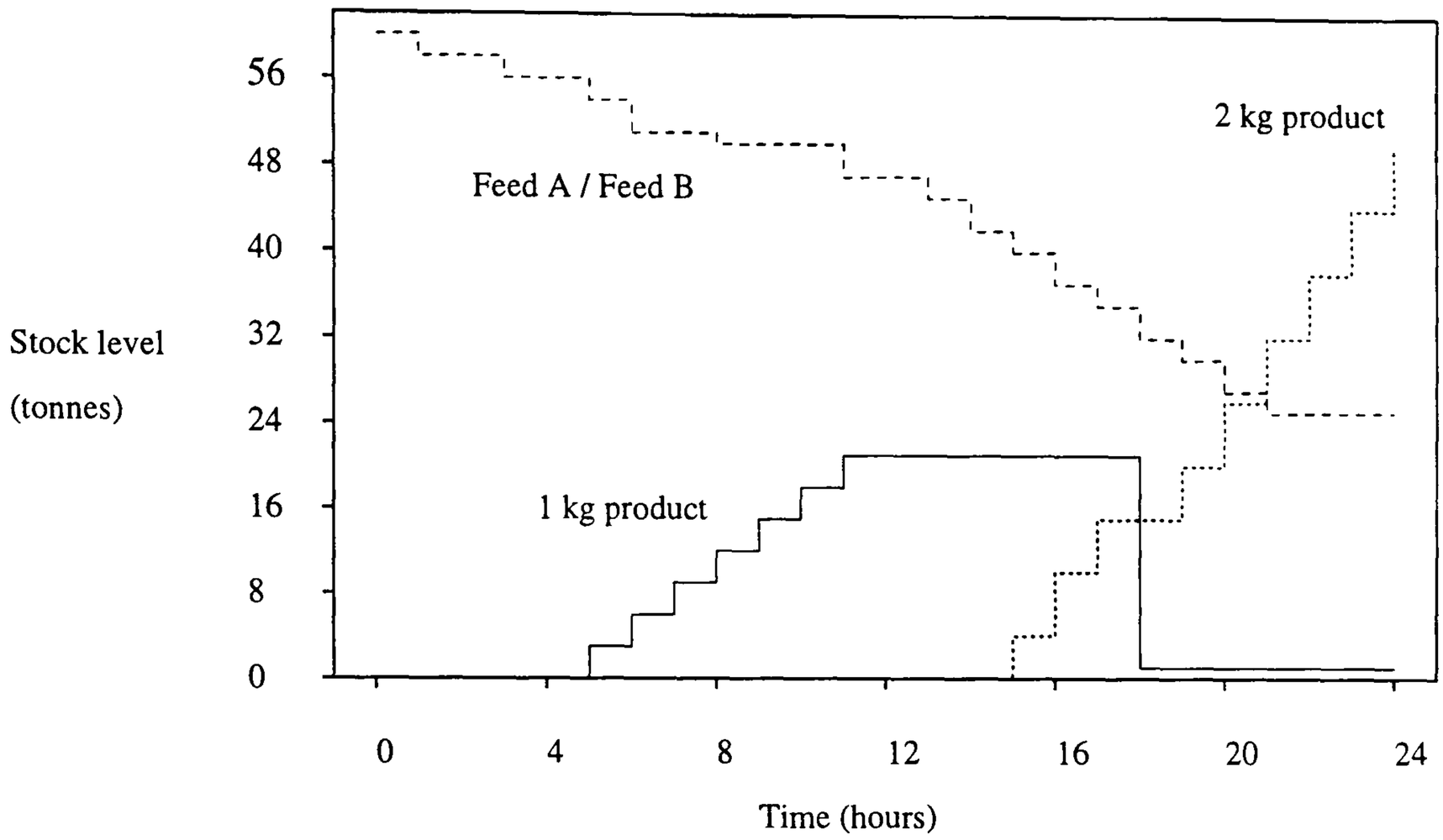


Figure 3.6: Variation of stock levels for optimal solution to Example I.

change-overs are carried out in the optimal solution rather than only one which might be initially supposed. This highlights the importance of detailed modelling for larger systems. Of course, this may result in a prohibitively large problem, which gives rise to the need for an aggregate formulation which can capture some of these important effects.

The RTN formulation can also be used to model more complicated chemical processing tasks as is illustrated by the next example.

3.4 Example II

Consider the process described by the State-Task Network shown in figure 3.7. This converts a feed A to a product $P1$. Initially there is a batch reaction step lasting 3 hours which starts with a 20% solution of A in an inert solvent and uses a 10 tonne glass-lined reactor. Steam is required at a rate of 0.05 kg/s per tonne of reactor contents for the first hour to bring the contents of the reactor up to operating temperature; after this, the temperature is maintained by the heat of reaction. As well as the primary reaction $A \rightarrow P1$, there is also a side reaction $A \rightarrow P2$ forming a less valuable product $P2$. There is a negligible amount of A remaining at the end of the reaction stage.

The reaction products then undergo a 2 hour batch distillation operation to separate the products and the solvent. This task takes place in a 10.0 tonne capacity batch distillation

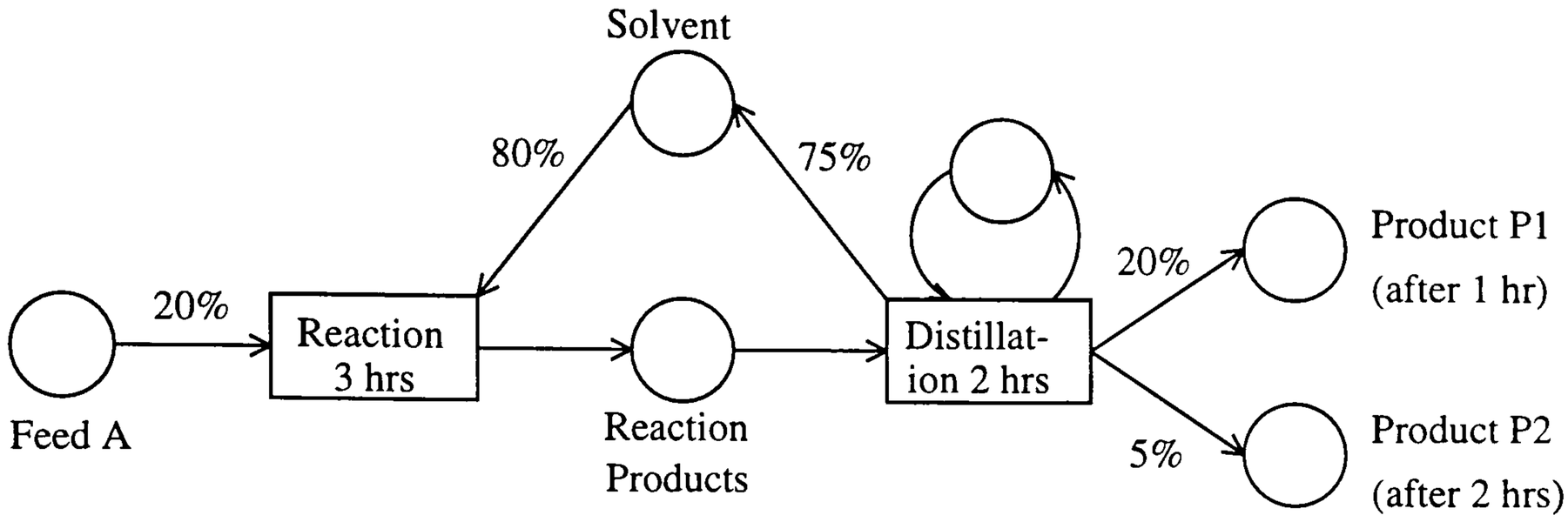


Figure 3.7: State-Task Network of process for Example II.

column. It requires 0.15 kg/s of steam for the reboiler and 2.0 kg/s of cooling water for the condenser, these values being independent of the initial volume of the contents of the vessel which has to be greater than a certain value for it to operate safely. This is to ensure that the entire heat exchange surface in the reboiler of the vessel is completely submerged for the duration of the distillation task. A first cut product which is rich in $P1$ (the most volatile component) is collected for the first hour of the distillation. Then the second product $P2$ is removed, leaving very nearly pure solvent which can be re-used in the reaction step. An operator is required for the duration of this activity, to supervise it and change the condensate collection vessel halfway through.

The RTN utilisation coefficients for the resources involved in the reaction and distillation tasks are given in tables 3.4 and 3.5.

Note that the utilities (steam, cooling water, manpower) are treated in the same way as the equipment items and the different types of material. The quantity of resource required for the task is taken from the 'excess amount' (corresponding to the current available supply allowing for the demands of other active tasks) at the start of the task and then released at the end (or, in some cases, part way through the task). By way of illustration, suppose that there is a total supply of 5.0 kg/s of cooling water available and that we consider a period when no other tasks making use of this facility are active. The distillation task is considered to 'borrow' 2.0 kg/s of the total supply for its 2 hour duration. This is illustrated in figure 3.8.

We can also represent the restrictions on the safe operation of the distillation vessel, by the following operational constraint (see equation (3.4)).

$$V_{kr}^{\min} N_{kt} \leq \xi_{kt} \leq V_{kr}^{\max} N_{kt} \qquad (k = \textit{Distillation},\ r = \textit{Distillation Vessel}) \quad \forall\ t$$

Here, $V_{kr}^{\min}$ is the minimum quantity (measured as a volume or weight as applicable) of liquid

	k = *Reaction*							
	$\mu_{kr\theta}$				$\nu_{kr\theta}$			
θ:	0	1	2	3	0	1	2	3
r = 10 *te Reactor*	-1.0	0.0	0.0	+1.0	0.0	0.0	0.0	0.0
r = *Feed A*	0.0	0.0	0.0	0.0	-0.20	0.0	0.0	0.0
r = *Solvent*	0.0	0.0	0.0	0.0	-0.80	0.0	0.0	0.0
r = *Reaction Products*	0.0	0.0	0.0	0.0	0.0	0.0	0.0	+1.0
r = *Steam*	0.0	0.0	0.0	0.0	-0.05	+0.05	0.0	0.0

Table 3.4: Resource utilisation coefficients for the reaction task.

	k = *Distillation*					
	$\mu_{kr\theta}$			$\nu_{kr\theta}$		
θ:	0	1	2	0	1	2
r = *Distillation Unit*	-1.0	0.0	+1.0	0.0	0.0	0.0
r = *Reaction Products*	0.0	0.0	0.0	-1.0	0.0	0.0
r = *Product P*1	0.0	0.0	0.0	0.0	+0.20	0.0
r = *Product P*2	0.0	0.0	0.0	0.0	0.0	+0.05
r = *Solvent*	0.0	0.0	0.0	0.0	0.0	+0.75
r = *Steam*	-0.15	0.0	+0.15	0.0	0.0	0.0
r = *Cooling Water*	-2.0	0.0	+2.0	0.0	0.0	0.0
r = *Manpower*	-1.0	0.0	+1.0	0.0	0.0	0.0

Table 3.5: Resource utilisation coefficients for the distillation task.

that should initially be in the distillation vessel to ensure its safe operation. The maximum capacity of the distillation vessel is given by $V_{kr}^{\max}$.

Having demonstrated how the RTN formulation can represent important details of chemical processing tasks, we shall now show it can be applied on a higher level for multisite scheduling problems.

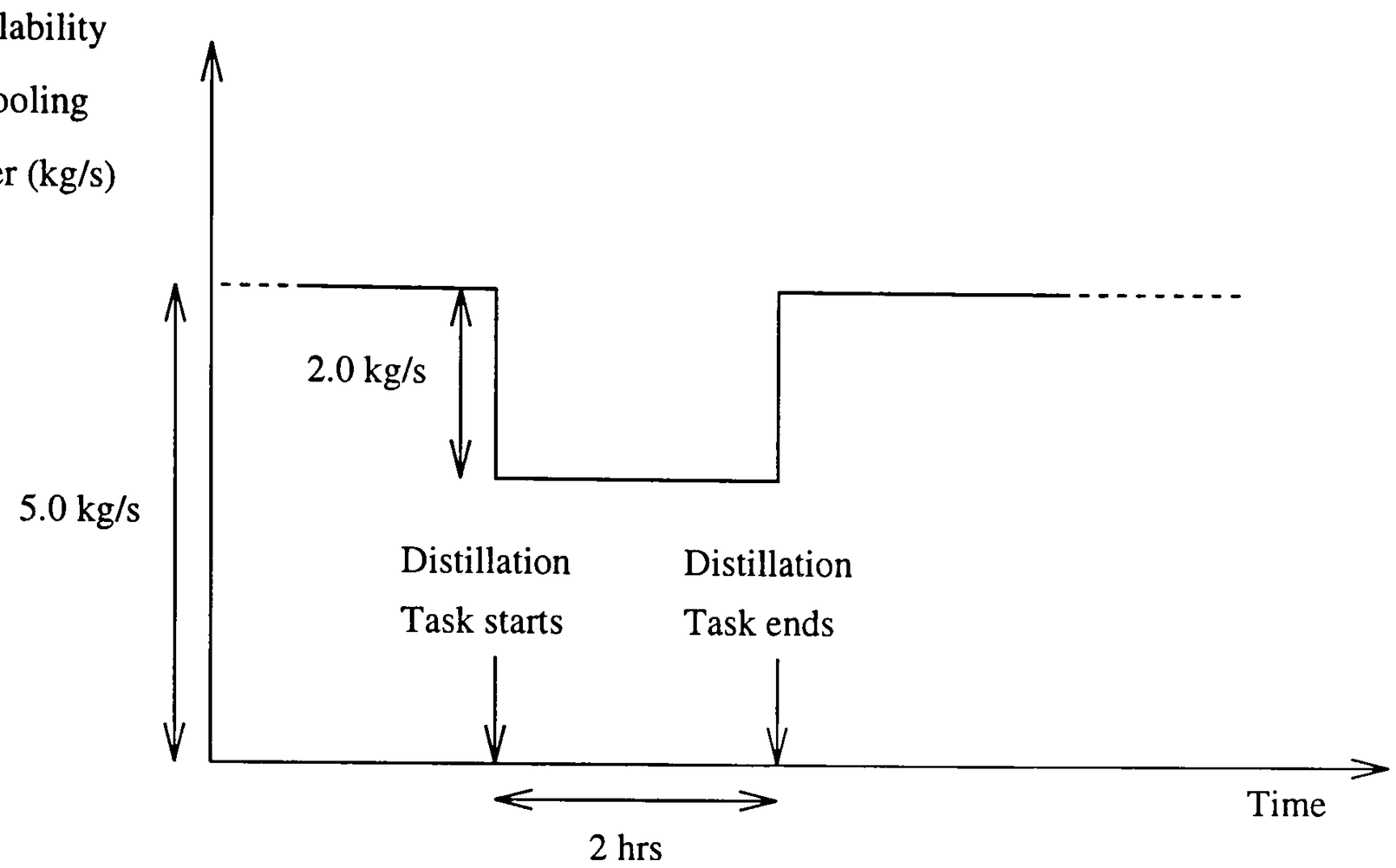


Figure 3.8: Effect of the distillation task on the cooling water availability.

3.5 RTNs for Multisite Scheduling

The examples given above demonstrate how the RTN formulation can model production activities in detail including such operational constraints as change-over tasks and limited utility levels, as well as the usual productive tasks. Formulations based on this framework have allowed several detailed industrial scheduling problems involving production at a single multipurpose plant to be effectively tackled.

The general nature of the RTN framework means that it can also be applied directly to multisite scheduling and the wider supply chain activities of distribution and warehousing. In this context, location is an important characteristic that distinguishes different resource types. The transportation of materials, therefore, can be modelled as tasks that transform materials at one location into materials in another location. If the task uses a transportation medium (*e.g.* a lorry), then it is considered as also transforming the lorry at the departing location (one resource type) into a lorry at the destination location (another resource type). The level of detail used in the RTN representation depends upon the specific features that require modelling. Some examples are given below.

3.5.1 Third Party Transportation and Quantity Discounts

It is often the case that producing companies contract out the transportation of materials to a third party. For many companies, this turns out to be a more economic alternative to maintaining their own fleet of vehicles. Also, raw material transportation to the company's sites is sometimes the responsibility of the suppliers. It is, therefore, unlikely that the company operating the multisite system will concern itself with the details of lorry movements in these cases. Despite this, even if a third party is carrying out the transportation, it may be still be necessary for the producing company to take into account features such as delivery lead-times and quantity discounts.

Unit Transportation Costs (£ per te)		
Standard		*Express*
< 100 *te*	≥ 100 *te*	
50.0	40.0	80.0

Table 3.6: Haulage company price quote.

Consider the following example in which a haulage company is charged with the transportation of a certain raw material from a supplier to one of the producing company's plants. Two types of delivery service with different lead-times are offered. In this case 'lead-time' refers to the maximum amount of time that the haulage company guarantees will elapse between the placing of an order and the receipt of the goods at the plant. On their standard delivery service (which has a delivery lead time of 24 hours), they offer a quantity discount for amounts over 100 tonnes. They also offer an express service which takes 12 hours but without any quantity discount. The minimum load that the haulage company will accept on both the standard and express services is 10.0 tonnes. These unit costs are summarised in table 3.6.

This cost structure can be represented in the RTN framework by introducing three tasks which have a common output (raw material at the plant) and are considered to affect this resource only. The tasks have no inputs since there is no need to include the raw material at the supplier in the RTN. They have continuous extents only and are characterised by a maximum and minimum extent (*i.e.* $V_k^{\min} \leq \xi_{kt} \leq V_k^{\max}$), a duration ($\tau_k$) and a variable cost ($C_k^V$; *cf.* equation (3.5)). This is shown in figure 3.9.

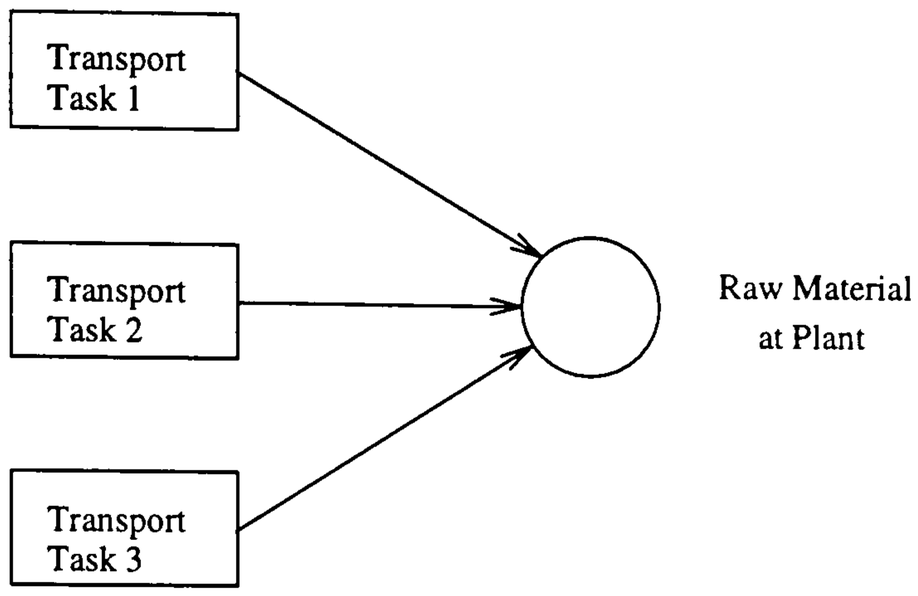


Figure 3.9: RTN representation of transportation options using three tasks.

The maximum and minimum extents, the durations and the unit costs for each task are given in table 3.7.

Transport Task	$V_k^{\min}$ (te)	$V_k^{\max}$ (te)	τ_k (hrs)	C_k^V (£/te)
1	10.0	100.0	24	50.0
2	100.0	∞	24	40.0
3	10.0	∞	12	80.0

Table 3.7: RTN transport task parameters.

Note that the maximum practical amount that the haulage company can transport is assumed to be so large that it is not constraining for the producing company who can effectively make their order as large as they want.

3.5.2 Mixed Lorry Loads

Suppose now that the producing company *does* own its own fleet of lorries, ships, *etc.* and therefore *is* directly concerned with the details of transportation. It will often be desirable to have mixed lorry loads (*i.e.* more than one material or product type in the same vehicle) — especially for problems with many product types and smaller delivery quantities. Situations in which each lorry load consists of a single material, or a number of materials in fixed proportions can be modelled directly within the RTN framework using the concepts discussed so far. In this case the resource utilisation coefficients correspond to the proportions in which the

various materials are carried. This is illustrated in figure 3.10.

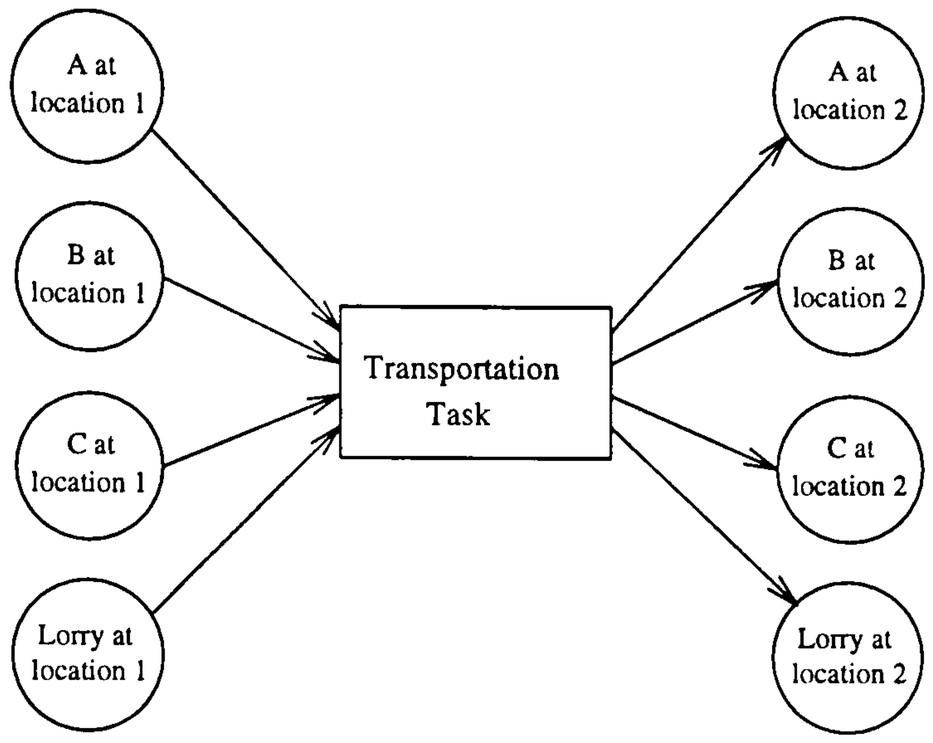


Figure 3.10: RTN for a transportation task with mixed loads in fixed proportions.

In practice, however, the proportions of the various materials being transported by a vehicle are not usually fixed but may be allowed to vary, the only constraint being the overall vehicle capacity itself. In such cases, the transportation task cannot be modelled in terms of just a single continuous extent variable ξ_{kt}, since we need to characterise the actual amounts of each material being transported. If $\xi_{kt}^{(m)}$ denotes the amount of material m carried by transportation task k starting at time t, we have the following operational constraint:

$$V_k^{\min} N_{kt} \leq \sum_m \sigma_m \xi_{kt}^{(m)} \leq V_k^{\max} N_{kt} \qquad \forall\ t \tag{3.7}$$

where σ_m represents the specific volume of material m; $V_k^{\min}$ is a lower bound on the total volume of the load; and $V_k^{\max}$ is the maximum volume capacity of the vehicle. We then use $\xi_{kt}^{(m)}$ in the resource balance equation (3.2) for the resources corresponding to material m at the two locations.

It is worth noting that the set of constraints (3.7) is, in fact, completely equivalent to replacing the composite transportation task of figure 3.10 by a set of simpler tasks k_m, each transporting a single material m, plus one more task k_L representing movement of the lorry. This situation is illustrated in figure 3.11. We then have the operational constraint,

$$V_k^{\min} N_{k_L t}^{(m)} \leq \sum_m \sigma_m \xi_{k_m t} \leq V_k^{\max} N_{k_L t}^{(m)} \qquad \forall\ t \tag{3.8}$$

which, of course, is mathematically identical to (3.7).

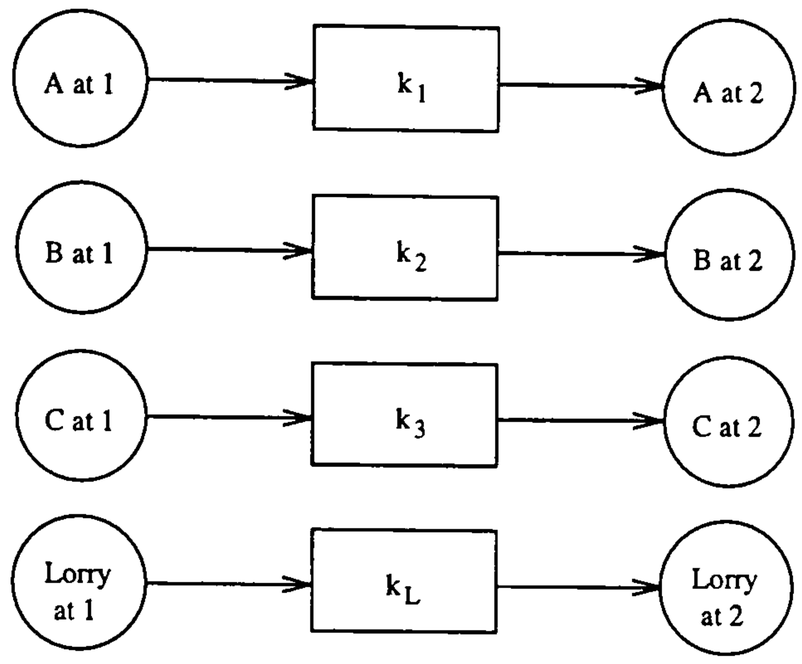


Figure 3.11: RTN for a transportation task with mixed loads in variable proportions.

As well as the overall capacity constraints (3.7), it may be the case that there is a separate lower limit $V_{km}^{\min}$ for each material being carried:

$$V_{km}^{\min} N_{kt}^{(m)} \leq \xi_{kt}^{(m)} \qquad \forall\ m, t \tag{3.9}$$

This lower limit might correspond to a single pallet of each product item, for example.

3.5.3 Material Incompatibilities

More complex features concerning the transportation of materials can also be modelled within the RTN framework with the introduction of further constraints or more transportation tasks. For instance, some materials may be incompatible as far as transportation or storage is concerned. Thus, they cannot be transported together in the same vehicle or stored in the same area of the warehouse. This is sometimes the case for products with packings whose geometries make it difficult to stack them together, or occasionally for certain foodstuffs, when one can corrupt the other (*e.g.* the ethylene emitted from apples can cause lettuce to deteriorate).

Consider a simple example concerning the transportation of three materials A, B, and C betweens two locations 1 and 2. A and C are mutually incompatible, but either can be transported with B. We assume that only one lorry type is used and that any other resources involved (manpower, for instance) are effectively unlimited and do not need to be included. We can model this situation using a total of eight resources and three transportation tasks as shown in figure 3.12.

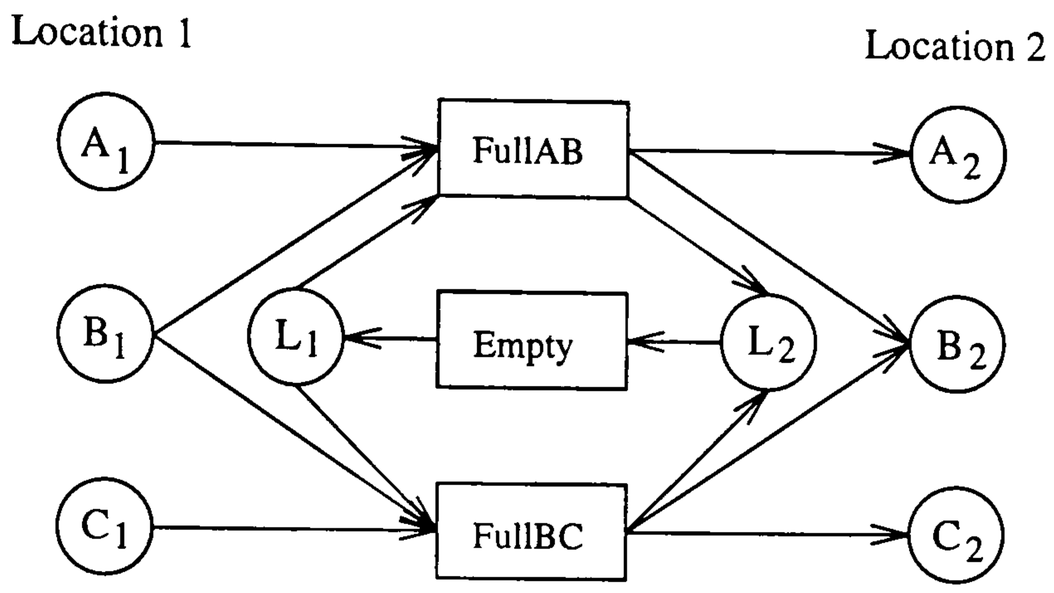


Figure 3.12: RTN representation of transportation of incompatible materials.

There are four physical resources involved, the three materials *A*, *B* and *C*, and also the lorry *L*. Since all these resources can occur at each of the two locations, this means that we have a total of *eight* RTN resources (location obviously being an important characteristic of a resource in this case). It can also be seen that, for this simple example, we have two possible mixed lorry load combinations: *A* with *B*, and *B* with *C*. We therefore need a transportation task to model each of these possibilities in the same way as has already been described in section 3.5.2. We also need a transportation task in which the empty lorry returns from location 1 to location 2 to pick up more material. This task will have a discrete extent N_{kt} but no continuous extent ξ_{kt} since no materials are being carried by the empty lorry.

3.5.4 Warehousing Considerations

In its simplest sense, a warehouse is a single building in which a certain set of materials are stored, its storage capacity being determined by its volume. If a warehouse w has capacity $V_w^{\max}$, then we can write the following constraint for the materials that can be stored within it:

$$\sum_{r \in S_w} \sigma_r R_{rt} \leq V_w^{\max} \qquad \forall\ w, t \tag{3.10}$$

where σ_r is the specific volume of resource r and S_w is the set of resources that are stored in warehouse w. Note that these sets S_w are disjoint, *i.e.* a resource r can be stored only in one warehouse w. Although there can obviously be stocks of the same *physical material* in different warehouses, these correspond to different *resources* depending on where they are (see discussion in section 3.1).

It is nearly always the case that a certain amount of 'safety stock' is kept corresponding to a minimum storage level which must be maintained for each material. This guards against

uncertainty in, for example, demand levels and re-stocking leadtimes, keeping the probability of a 'stock-out' to an acceptable level. We can enforce this by the following constraint:

$$R_{rt} \geq R_r^{\min} \qquad \forall\ r, t \tag{3.11}$$

where $R_r^{\min}$ is the safety stock level for resource r.

Equation (3.10) assumes that all the storage capacity of the warehouse can be used to store all the materials, in other words, it is entirely *flexible* as to which material is stored in which part of it. The converse to this situation is where the warehouse is partitioned into separate storage areas which are *dedicated* to the storage of one particular material. In this case we would write the following:

$$\sigma_r R_{rt} \leq V_{wr}^{\max} \qquad \forall\ w, r, t \tag{3.12}$$

where $V_{wr}^{\max}$ is the storage capacity available in warehouse w for resource r. We note that both (3.11) and (3.12) are special cases of the general capacity constraint (3.3).

In practice, a warehouse is likely to contain both dedicated storage areas and also flexible storage areas in which all or perhaps only a subset of the overall inventory set is stored; this may be due to packing or accessibility considerations. In this case, we can consider each storage area as a separate warehouse, writing constraints similar to (3.9), (3.10) and (3.11) for each one.

Further complications can arise if the operation of warehouses is considered in detail and if the costs associated with operating them (labour, administration, heating, lighting *etc.*) are to be modelled correctly. For example, it may be precisely because there are fixed costs associated with operating the storage area within a warehouse, that it is divided into distinct portions which are filled on a precedence basis. This is illustrated in figure 3.13.

If we wish to model fixed as well as variable costs associated with the operation of each storage area, we can introduce the idea of a storage task which is similar to the storage task used in the example of section 3.3, *i.e.* an operation which is considered to consume the resource being stored and then produce the same resource one time interval later. Using an analogous approach to that used for transportation of mixed lorry loads above, we can write the following operational constraints for the storage task in each storage area:

$$V_k^{\max} N_{kt} \leq \sum_m \sigma_m \xi_{kt}^{(m)} \leq V_k^{\max} N_{kt} \qquad \forall\ k, t \tag{3.13}$$

where σ_m represents the specific volume of material m, and $V_k^{\max}$ is the maximum volume capacity of the storage area. Here, $\xi_{kt}^{(m)}$ is effectively the amount of material m stored in the

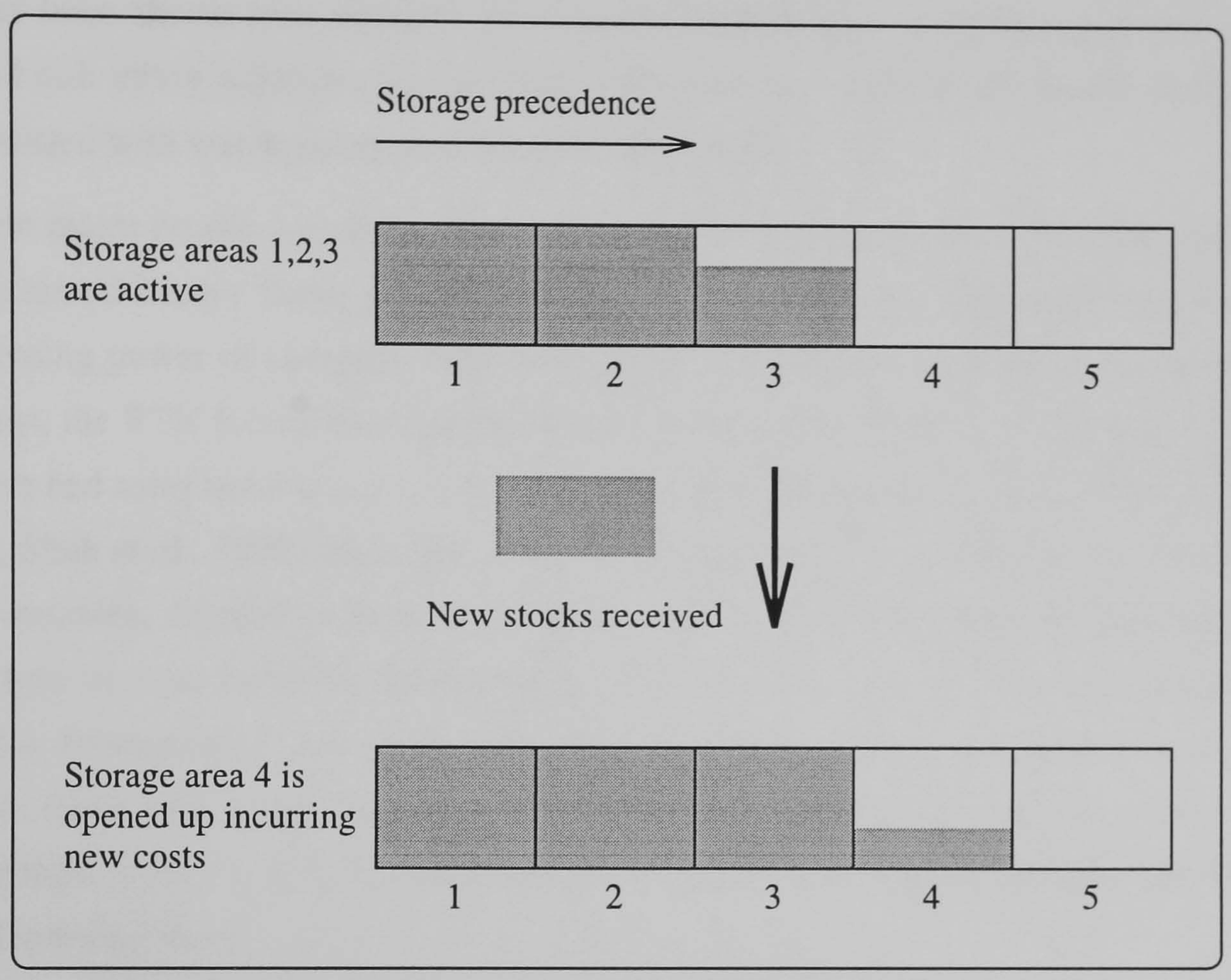


Figure 3.13: Illustration of sub-division of warehouse capacity.

storage area represented by storage task k over time interval t. The discrete extent N_{kt} is, in this case, a binary variable indicating whether or not this storage area is in use over time interval t. Therefore, constraint (3.13) is simply an overall capacity constraint on storage tasks. The introduction of N_{kt} allows any fixed costs associated with the operation of each storage area to be included in the objective function.

A safety stock limitation on the total amount of material m held in the warehouse may be imposed *via* the constraint:

$$V_m^{\min} \leq \sum_{k \in SA} \xi_{kt}^{(m)} \qquad \forall\ m, t \tag{3.14}$$

where SA is the set of storage tasks describing the operation of different storage areas in the warehouse.

3.6 Concluding Remarks

This chapter has demonstrated that the RTN framework, with its unified and general approach, provides a way to model most of the activites and resources involved in a

production system involving flexible multipurpose plants to a high level of detail. In particular, it has been shown how complex production considerations such as equipment changeovers and task utility requirements can be modelled, as well as the wider supply chain activities associated with warehousing and material distribution.

One major practical problem with such a detailed approach, however, is the size of the resulting mixed integer linear programming (MILP) problems for large-scale systems. With the increasing power of computer hardware and the introduction of more and more efficient algorithms, the RTN formulation presented here and the related STN formulation of Kondili *et al.* have had some notable success in solving a variety of industrial sized problems (see, for instance, Shah *et al.*, 1993). However, while these typically involve hundreds or thousands of integer variables, detailed models of the large-scale systems considered in this thesis may involve tens or even hundreds of thousands of integer variables, which often renders them intractable. Examples of such problems include multisite systems, such as that described in chapter 6; those with a very large number of different product types and equipment changeovers; or those with a very long overall planning horizon in relation to the time discretisation required to model them.

There is, therefore, a need for a method that can reduce problem size whilst retaining as many as possible of the important advantages achieved by modelling the system in detail using the RTN framework. In the next chapter, we consider aggregate RTN formulations that seek to address this issue.

BIBL. LONDIN.

Chapter 4

Aggregation Techniques for RTN Formulations

The RTN formulation presented in the previous chapter is typical of multiperiod MILP scheduling formulations in which the planning horizon is discretised into equally spaced time intervals. A set of discrete and continuous decision variables describe the resource levels and the activities being performed in each time interval. An objective function is optimised (*e.g.* profit is maximised) subject to certain constraints such as resource balances and capacity constraints. By adopting a uniform representation of all available resources, the RTN framework allows complicating production features such as sequence dependent cleaning, as well as maintenance and wider supply chain activities to be considered as part of the scheduling problem.

The above approach certainly permits the solution of many detailed industrial scheduling problems that were previously considered intractable. However, in some cases, the resulting formulations may be too large to be solvable using currently available solution algorithms and codes. We note that problem size increases with the number of resource types and tasks in the process, and also with the length of the planning horizon in relation to the time discretisation interval. Multisite problems, in particular, may have planning horizons of several months and may involve several complex plants with processing steps on a time-scale of less than one hour. Such problems would result in MILP formulations involving tens of thousands of integer variables and would probably be beyond the scope of current branch-and-bound algorithms. On the other hand, the central plan for the multisite system needs to take account of the individual plant capacities but not their detailed schedules. Hence there is a need for a method that involves fewer integer variables but that still captures enough of the detail required to produce an optimal or near-optimal solution. The *aggregate* formulations described in this chapter provide one such method.

In this chapter, we introduce the basic time aggregation concept. We then outline a general approach to the aggregation of discrete time formulations introducing the idea of an *aggregation operator*. This is used to derive a family of aggregate scheduling formulations of increasing accuracy and complexity which are based on the RTN formulation of Pantelides (1994) presented in the previous chapter. As expected, these aggregate formulations represent an approximation to the fully detailed model. Our general approach, however, allows the trade-off between problem size and accuracy of solution to be fully exploited. In addition, the aggregate formulations inherit the advantages of the RTN representation, automatically taking

account of all available resources and all tasks including production activities, equipment cleaning and change-overs, and material transportation and storage.

4.1 Basic Ideas

Generating an aggregated formulation involves replacing groups of related variables from the detailed formulation with a set of aggregated variables in order to reduce the size of the problem to be solved. In the approach adopted in this thesis, this is achieved by *temporal aggregation* which involves splitting the planning horizon (of length H) into smaller portions or *Aggregated Time Periods* (ATPs). Each ATP is described by aggregated variables, each of which is equivalent to a weighted sum of the corresponding detailed variables over the time intervals spanned by that ATP. This is represented pictorially in figure 4.1 below.

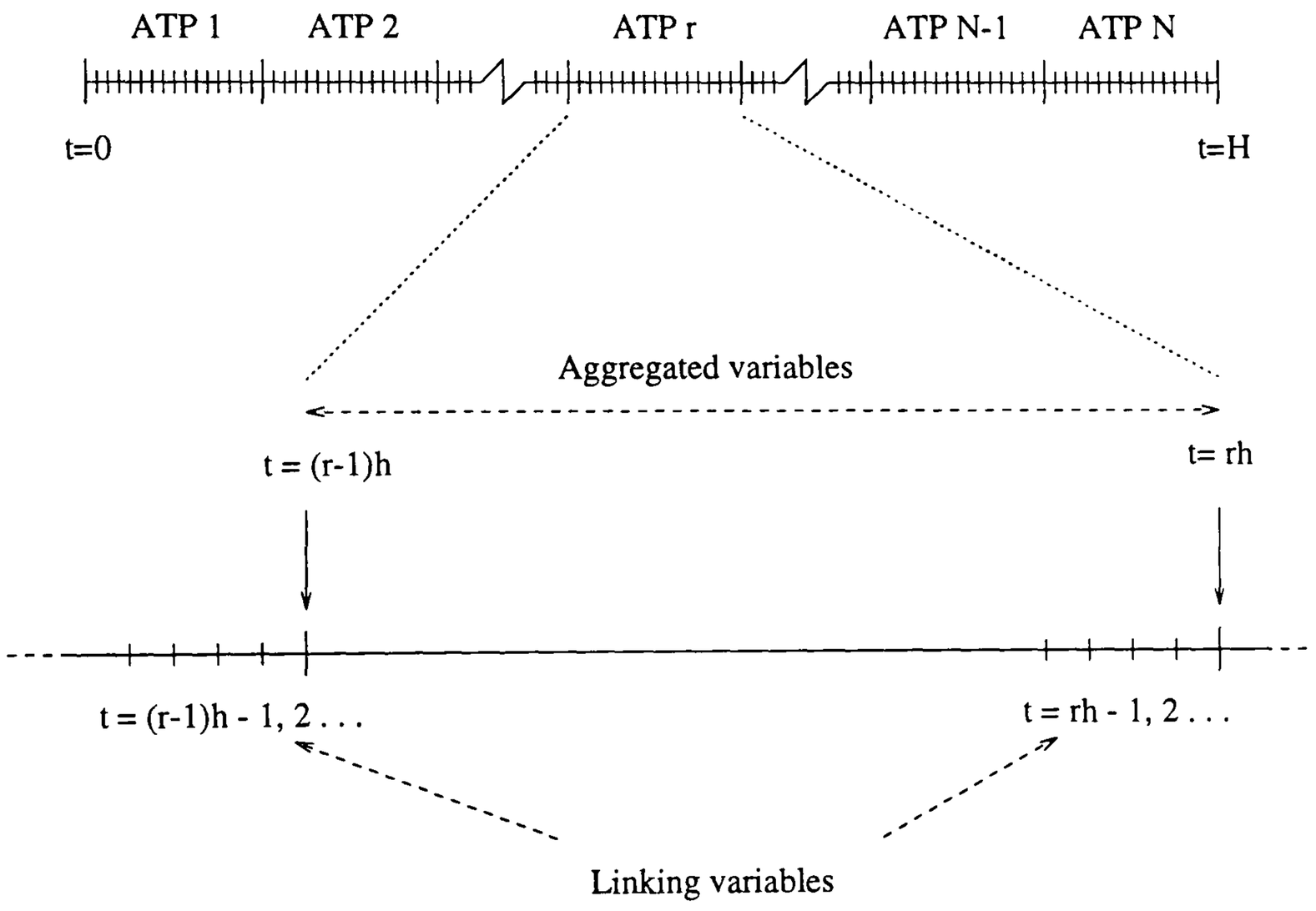


Figure 4.1: Aggregation of the planning horizon.

The excess resource levels at the beginning of one ATP are related to those at the beginning of the next ATP by an aggregated excess resource balance which is derived from the detailed RTN excess resource balance. The balance involves aggregated variables representing, for example, the total number of times that a certain task is started within the ATP. The total change in a material level from the start of one ATP to the next is related to the aggregated batch-size in that ATP of all the tasks that involve it.

In most cases, some tasks that start near the end of one ATP will continue into the next one. The extent variables describing these tasks are termed *linking variables* and play an important role in our aggregated formulations.

Other constraints from the detailed RTN formulation (*e.g.* the excess resource capacity constraints (3.2)) can also be summed over time in the same way to give constraints that involve the same aggregated and linking variables.

In the next section we shall formalise the ideas outlined above, showing how the aggregate and linking variables arise naturally from mathematical manipulations performed on the detailed RTN formulation.

4.2 A General Approach to the Aggregation of Discrete-Time Formulations

For any discrete time formulation, consider the general inequality constraint†:

$$C_t(\,.\,) \leq 0 \quad (t = 1, 2, .., H) \tag{4.1}$$

This constraint holds for each time t in the planning horizon of length H. Suppose that we can divide the planning horizon into n equal sections or aggregated time periods (ATPs), each of h time intervals (*i.e.* $nh = H$). We can then sum the constraints within each section obtaining:

$$\sum_{t'=t-h+1}^{t} C_{t'}(\,.\,) \leq 0 \quad (t = h, 2h, .., H) \tag{4.2}$$

In addition to the simple summation carried out above, we can also carry out a time weighted summation within each ATP. To do this, we define a non-negative time-dependent weighting function $f(t-t')$:

$$f(t-t') \geq 0 \quad (t' = t-h+1, t-h+2, .., t) \tag{4.3}$$

The fact that f is function of $t-t'$ (rather than simply t) ensures that it has the same variation over each ATP. The time weighted summation of (4.1) can then be written as:

$$\sum_{t'=t-h+1}^{t} f(t-t')\, C_{t'}(\,.\,) \leq 0 \quad (t = h, 2h, .., H) \tag{4.4}$$

If the original constraint (4.1) is *linear*, we can replace *some* of the original variables in the aggregated constraint by aggregate variables which are time weighted sums of the original

† Equality constraints can be regarded as special cases of inequality constraints for the purposes of the discussion in this section.

variables. In general, however, we cannot replace *all* the original variables by their aggregated counterparts. We must retain some of them explicitly in the aggregated constraint. These are the linking variables discussed in the previous section.

We shall now introduce a specific family of weighting functions:

$$f^p(t-t') \equiv (t-t'+1)^p \qquad (p = 0, 1, .., m) \tag{4.5}$$

It can be seen that these functions have the non-negativity property over the range given by (4.3) above. Their variation over each ATP interval is illustrated in figure 4.2 for $p = 0$, 1 and 2:

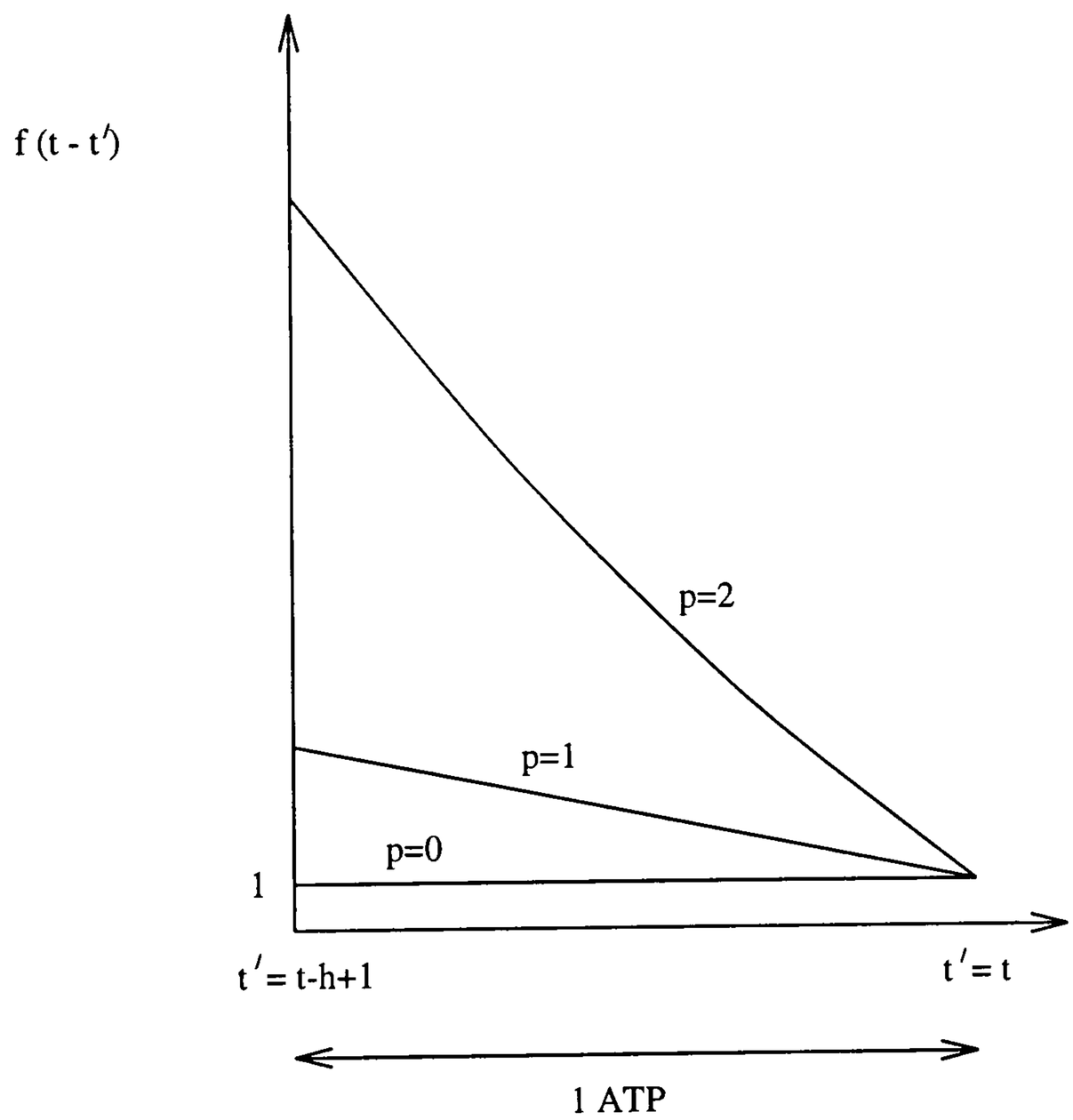


Figure 4.2: Weighting functions for $p = 0$, 1 and 2.

The use of the weighting functions given by (4.5) leads to a family of aggregated constraints of the form:

$$\sum_{t'=t-h+1}^{t} f^p(t-t')\, C_{t'}(\,.\,) \;\leq\; 0 \qquad (p = 0, 1, .., m \;\;,\;\; t = h, 2h, .., H) \tag{4.6}$$

It is helpful to define a general p^{th} order *aggregation operator* $A^p[\,]$ as below:

$$A^p[C_t(\,.\,)] \equiv \sum_{t'=t-h+1}^{t} (t-t'+1)^p C_{t'}(\,.\,) \qquad (p=0,\,1,\,..,\,m) \tag{4.7}$$

Application of such operators to the constraints of the detailed formulation then results in the aggregate formulation. It can be seen that, for $p=0$, the weighting function given above becomes unity over the whole ATP. Hence, the application of the *zeroth* order aggregation operator is equivalent to a simple summation.

The aggregate *variables* mentioned earlier can then be defined as:

$$\tilde{X}_t^{(p)} \equiv A^p[X_t] = \sum_{t'=t-h+1}^{t} (t-t'+1)^p X_{t'} \qquad (p=0,\,1,\,..,\,m) \tag{4.8}$$

In the next section we apply this and related operators to the constraints of the detailed RTN formulation discussed in the previous chapter. This enables us to generate a family of *aggregate* RTN formulations.

4.3 Excess Resource Balance Aggregation

4.3.1 Zeroth Order Aggregation

Consider the general RTN excess resource balance constraint given below:

$$R_{rt} = R_{r,t-1} + \sum_k \sum_{\theta=0}^{\tau_k} \left(\mu_{kr\theta} N_{k,t-\theta} + \nu_{kr\theta} \xi_{k,t-\theta} \right) + \Pi_{rt} \qquad \forall\ r,\, t \tag{3.2}$$

We now sum each side of the above equality from a time t back to a time $t-h+1$. This is equivalent to applying the zeroth order (*i.e.* $p=0$) aggregation operator, and leads to:

$$\sum_{t'=t-h+1}^{t} R_{rt'} = \sum_{t'=t-h+1}^{t} R_{r,t'-1} + \sum_{t'=t-h+1}^{t} \sum_k \sum_{\theta=0}^{\tau_k} \left(\mu_{kr\theta} N_{k,t'-\theta} + \nu_{kr\theta} \xi_{k,t'-\theta} \right) + \sum_{t'=t-h+1}^{t} \Pi_{rt'}$$

which can be simplified to:

$$R_{rt} = R_{r,t-h} + \sum_k \left(\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \sum_{t'=t-h+1}^{t} N_{k,t'-\theta} + \sum_{\theta=0}^{\tau_k} \nu_{kr\theta} \sum_{t'=t-h+1}^{t} \xi_{k,t'-\theta} \right) + \sum_{t'=t-h+1}^{t} \Pi_{rt'} \tag{4.9}$$

Consider now the quantity:

$$\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \sum_{t'=t-h+1}^{t} N_{k,t'-\theta}$$

appearing on the right hand side of (4.9). This can be written as:

$$\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \sum_{t'=t-h+1}^{t} N_{k,t'-\theta} = \sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \sum_{t'=t-h+1-\theta}^{t-\theta} N_{kt'}$$

$$= \sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \left(\sum_{t'=t-h+1-\theta}^{t-h} N_{kt'} + \sum_{t'=t-h+1}^{t} N_{kt'} - \sum_{t'=t+1-\theta}^{t} N_{kt'} \right)$$

$$= \sum_{\theta=1}^{\tau_k} \mu_{kr\theta} \sum_{t'=t-h+1-\theta}^{t-h} N_{kt'} + \sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \sum_{t'=t-h+1}^{t} N_{kt'} - \sum_{\theta=1}^{\tau_k} \mu_{kr\theta} \sum_{t'=t+1-\theta}^{t} N_{kt'}$$

Note that the lower limit of the outer summation in the first and third terms above has been changed from $\theta = 0$ to $\theta = 1$ since the corresponding inner summations vanish for $\theta = 0$. By shifting the indices of the inner summation of the first and third terms, the above can be written as:

$$\sum_{\theta=1}^{\tau_k} \mu_{kr\theta} \sum_{\theta'=1}^{\theta} \left(N_{k,t-h+1-\theta'} - N_{k,t+1-\theta'} \right) + \left(\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \right) \sum_{t'=t-h+1}^{t} N_{kt'}$$

Finally, the order of the double summation in the first term above can be reversed to yield the following relation for the quantity appearing on the right hand side of (4.9):

$$\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \sum_{t'=t-h+1}^{t} N_{k,t'-\theta} = \sum_{\theta'=1}^{\tau_k} \left(N_{k,t-h+1-\theta'} - N_{k,t+1-\theta'} \right) \sum_{\theta=\theta'}^{\tau_k} \mu_{kr\theta} + \left(\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \right) \sum_{t'=t-h+1}^{t} N_{kt'}$$

Returning to (4.9), we can see that a similar set of manipulations can be applied to the term:

$$\sum_{\theta=0}^{\tau_k} \nu_{kr\theta} \sum_{t'=t-h+1}^{t} \xi_{k,t'-\theta}$$

that also appears on the right hand side of (4.9). Hence we can write (4.9) as follows:

$$R_{rt} = R_{r,t-h} + \sum_k \left(\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \right) \sum_{t'=t-h+1}^{t} N_{kt'} + \sum_k \sum_{\theta'=1}^{\tau_k} \left(N_{k,t-h+1-\theta'} - N_{k,t+1-\theta'} \right) \sum_{\theta=\theta'}^{\tau_k} \mu_{kr\theta} +$$

$$\sum_k \left(\sum_{\theta=0}^{\tau_k} \nu_{kr\theta} \right) \sum_{t'=t-h+1}^{t} \xi_{kt'} + \sum_k \sum_{\theta'=1}^{\tau_k} \left(\xi_{k,t-h+1-\theta'} - \xi_{k,t+1-\theta'} \right) \sum_{\theta=\theta'}^{\tau_k} \nu_{kr\theta} + \sum_{t'=t-h+1}^{t} \Pi_{rt'} \quad (4.10)$$

We note that the above involves the following *aggregate variables* which were defined by (4.8), each of which replaces several variables from the detailed formulation:

$$\tilde{N}_{kt}^{(0)} \equiv \sum_{t'=t-h+1}^{t} N_{kt'} \quad (4.11a)$$

$$\tilde{\xi}_{kt}^{(0)} \equiv \sum_{t'=t-h+1}^{t} \xi_{kt'} \tag{4.11b}$$

For simplicity of representation, we also define the following *aggregated coefficients* which are calculated from the original RTN coefficients:

$$\tilde{\mu}_{kr\theta}^{(0)} \equiv \sum_{\theta'=\theta}^{\tau_k} \mu_{kr\theta'} \tag{4.12a}$$

$$\tilde{\nu}_{kr\theta}^{(0)} \equiv \sum_{\theta'=\theta}^{\tau_k} \nu_{kr\theta'} \tag{4.12b}$$

and also the following *aggregated demand parameter:*

$$\tilde{\Pi}_{rt}^{(0)} \equiv \sum_{t'=t-h+1}^{t} \Pi_{rt'} \tag{4.13}$$

We can then write (4.10) as follows:

$$\begin{aligned} R_{rt} \;=\; R_{r,t-h} \;+\; \sum_{k}\left(\tilde{\mu}_{kr0}^{(0)}\tilde{N}_{kt}^{(0)} \;+\; \tilde{\nu}_{kr0}^{(0)}\tilde{\xi}_{kt}^{(0)}\right) \\ +\; \sum_{k}\sum_{\theta=1}^{\tau_k}\tilde{\mu}_{kr\theta}^{(0)}\left(N_{k,t-h+1-\theta} - N_{k,t+1-\theta}\right) \;+\; \sum_{k}\sum_{\theta=1}^{\tau_k}\tilde{\nu}_{kr\theta}^{(0)}\left(\xi_{k,t-h+1-\theta} - \xi_{k,t+1-\theta}\right) \\ +\; \tilde{\Pi}_{rt}^{(0)} \qquad\qquad \forall\; r,\, t = h,\, 2h,\, \ldots,\, H \end{aligned} \tag{4.14}$$

which represents an aggregated excess resource balance constraint. As well as the aggregate variables $\tilde{N}_{kt}^{(0)}$ and $\tilde{\xi}_{kt}^{(0)}$, this constraint also contains some variables that have been retained from the detailed formulation. These are:

$$N_{k,t+1-\theta}\;,\;\xi_{k,t+1-\theta} \qquad \forall\; k,\, t = h,\, 2h,\, \ldots,\, H,\, \theta = 1,\, \ldots,\, \tau_k \tag{4.15a}$$

$$R_{rt} \qquad\qquad \forall\; r,\, t = h,\, 2h,\, \ldots,\, H \tag{4.15b}$$

The task extent variables (4.15a) represent tasks k that start less than one task duration τ_k from the end of each ATP and therefore finish in the next ATP. The excess resource variables (4.15b) correspond to the resource levels on the ATP boundaries, where the resource level at the end of one ATP is equal to the resource level just before the start of the next ATP. These are therefore called *linking* variables since they account for interactions (or 'links') between neighbouring ATPs.

4.3.2 Higher Order Aggregation

We now consider applying the aggregation operator for general $p > 0$ to (3.2). Using a similar set of manipulations as for the zeroth order aggregation above we get the following aggregated constraint. This is derived in detail in Appendix A.

$$\sum_{t'=t-h+1}^{t} \left\{ (t-t'+1)^p - (t-t')^p \right\} R_{rt'} \; - \; h^p R_{r,t-h} \; = \; \sum_k \left(\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \right) \sum_{t'=t-h+1}^{t} (t-t'+1-\theta)^p N_{kt'}$$

$$+ \; \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (h+\theta'-\theta)^p \; \mu_{kr\theta} \; - \; \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (\theta'-\theta)^p \; \mu_{kr\theta}$$

$$+ \; \sum_k \left(\sum_{\theta=0}^{\tau_k} \nu_{kr\theta} \right) \sum_{t'=t-h+1}^{t} (t-t'+1-\theta)^p \xi_{kt'}$$

$$+ \; \sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (h+\theta'-\theta)^p \; \nu_{kr\theta} \; - \; \sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (\theta'-\theta)^p \; \nu_{kr\theta}$$

$$+ \; \sum_{t'=t-h+1}^{t} (t-t'+1)^p \Pi_{rt'} \tag{4.16}$$

It is also shown in Appendix A that, as expected, (4.16) reduces to (4.14) for $p = 0$. Furthermore, for any positive integral value of p, (4.16) can be expressed in terms of aggregate variables up to order p and the *same* set of linking variables ((4.15a) and (4.15b)) that were used in the zeroth order aggregate constraint. While the derivation of the explicit form of (4.16) is instructive in that it shows where the linking and aggregate variables arise from, a more concise and general form can be derived if the constraints of the detailed RTN formulation are considered in matrix form (see Appendix C).

To illustrate how we can represent (4.16) in terms of aggregate variables, consider the case of $p = 1$. For the sake of simplicity, we shall consider only the discrete N_{kt} variables as exactly the same type of terms occur for the continuous ξ_{kt} variables. The following is obtained:

$$\sum_{t'=t-h+1}^{t} \Big\{(t-t'+1)-(t-t')\Big\} R_{rt'} \;-\; hR_{r,t-h} \;=\; \sum_{k}\left(\sum_{\theta=0}^{\tau_k}\mu_{kr\theta}\right)\sum_{t'=t-h+1}^{t}(t-t'+1-\theta)N_{kt'}$$

$$+\;\sum_{k}\sum_{\theta'=1}^{\tau_k} N_{k,t-h+1-\theta'}\sum_{\theta=\theta'}^{\tau_k}(h+\theta'-\theta)\;\mu_{kr\theta}\;-\;\sum_{k}\sum_{\theta'=1}^{\tau_k} N_{k,t+1-\theta'}\sum_{\theta=\theta'}^{\tau_k}(\theta'-\theta)\;\mu_{kr\theta}$$

$$+\;\sum_{t'=t-h+1}^{t}(t-t'+1)\Pi_{rt'}$$

By simplifying the first term on the left hand side, and splitting the first term on the right hand side into two separate summations, we obtain:

$$\sum_{t'=t-h+1}^{t} R_{rt'} \;-\; hR_{r,t-h} \;=\; \sum_{k}\left(\sum_{\theta=0}^{\tau_k}\mu_{kr\theta}\right)\sum_{t'=t-h+1}^{t}(t-t'+1)N_{kt'} \;-\; \sum_{k}\left(\sum_{\theta=0}^{\tau_k}\theta\;\mu_{kr\theta}\right)\sum_{t'=t-h+1}^{t} N_{kt'}$$

$$+\;\sum_{k}\sum_{\theta'=1}^{\tau_k} N_{k,t-h+1-\theta'}\sum_{\theta=\theta'}^{\tau_k}(h+\theta'-\theta)\;\mu_{kr\theta}\;-\;\sum_{k}\sum_{\theta'=1}^{\tau_k} N_{k,t+1-\theta'}\sum_{\theta=\theta'}^{\tau_k}(\theta'-\theta)\;\mu_{kr\theta}$$

$$+\;\sum_{t'=t-h+1}^{t}(t-t'+1)\Pi_{rt'}$$

Recognising the aggregate variables and parameters defined by (4.11a,b) and (4.12a,b), and also re-introducing the continuous task extent variables, we can write this first order aggregate constraint as follows:

$$\tilde{R}_{rt}^{(0)} \;-\; hR_{r,t-h} \;=\; \sum_{k}\tilde{\mu}_{kr0}^{(0)}\tilde{N}_{kt}^{(1)} \;-\; \sum_{k}\tilde{\mu}_{kr1}^{(1)}\tilde{N}_{kt}^{(0)} \;+\; \sum_{k}\tilde{\nu}_{kr0}^{(0)}\tilde{\xi}_{kt}^{(1)} \;-\; \sum_{k}\tilde{\nu}_{kr1}^{(1)}\tilde{\xi}_{kt}^{(0)} \;+$$

$$\sum_{k}\sum_{\theta'=1}^{\tau_k} N_{k,t-h+1-\theta'}\sum_{\theta=\theta'}^{\tau_k}(h+\theta'-\theta)\;\mu_{kr\theta}\;-\;\sum_{k}\sum_{\theta'=1}^{\tau_k} N_{k,t+1-\theta'}\sum_{\theta=\theta'}^{\tau_k}(\theta'-\theta)\;\mu_{kr\theta}\;+$$

$$\sum_{k}\sum_{\theta'=1}^{\tau_k}\xi_{k,t-h+1-\theta'}\sum_{\theta=\theta'}^{\tau_k}(h+\theta'-\theta)\;\nu_{kr\theta}\;-\;\sum_{k}\sum_{\theta'=1}^{\tau_k}\xi_{k,t+1-\theta'}\sum_{\theta=\theta'}^{\tau_k}(\theta'-\theta)\;\nu_{kr\theta}\;+\;\tilde{\Pi}_{rt}^{(1)} \qquad (4.17)$$

where the newly introduced aggregate variables $\tilde{N}_{kt}^{(1)}$, $\tilde{\xi}_{kt}^{(1)}$ and $\tilde{R}_{rt}^{(0)}$ are defined by:

$$\tilde{N}_{kt}^{(1)} \;\equiv\; \sum_{t'=t-h+1}^{t}(t-t'+1)N_{kt'} \qquad (4.18a)$$

$$\tilde{\xi}_{kt}^{(1)} \equiv \sum_{t'=t-h+1}^{t} (t-t'+1)\xi_{kt'} \tag{4.18b}$$

$$\tilde{R}_{rt}^{(0)} \equiv \sum_{t'=t-h+1}^{t} R_{rt'} \tag{4.19}$$

New aggregate parameters $\tilde{\mu}_{kr1}^{(1)}$, $\tilde{\nu}_{kr1}^{(1)}$ and $\tilde{\Pi}_{rt}^{(1)}$ have also been introduced. These conform to the following general definitions:

$$\tilde{\mu}_{kr\theta}^{(p)} \equiv \sum_{\theta'=\theta}^{\tau_k} (\theta')^p \mu_{kr\theta'} \tag{4.20a}$$

$$\tilde{\nu}_{kr\theta}^{(p)} \equiv \sum_{\theta'=\theta}^{\tau_k} (\theta')^p \nu_{kr\theta'} \tag{4.20b}$$

$$\tilde{\Pi}_{rt}^{(p)} \equiv \sum_{t'=t-h+1}^{t} (t-t'+1)^p \Pi_{rt'} \tag{4.21}$$

Higher order aggregated excess resource balances require the introduction of more aggregate variable types. Both the first and second order aggregated resource balances are given in Appendix B. In general, for a p^{th} order aggregated balance, we will need the aggregate variables $\tilde{N}_{kt}^{(i)}$, $\tilde{\xi}_{kt}^{(i)}$, $i = 0, \ldots, p$ and $\tilde{R}_{rt}^{(i)}$, $i = 0, \ldots, p-1$.

4.4 Aggregation of other Constraints

The operational and capacity constraints of the detailed RTN formulation ((3.3) and (3.4)) are simpler in form than the excess resource balance and also differ in that they are inequalities rather than equalities. This second feature results in some more degrees of freedom when deriving their aggregated counterparts as will be demonstrated below. A major consideration in this context is the tightness of the resulting aggregate inequality constraints. In general, we shall endeavour to derive the latter by aggregating the minimum possible number of the original constraints that still allows us to express the aggregate formulation in terms of the aggregate variables as defined by (4.8) and the also the linking variables (4.15).

4.4.1 Excess Resource Capacity Constraints

The excess resource capacity constraints (3.3) of the detailed RTN formulation still apply to the linking excess resource variables:

$$0 \leq R_{rt} \leq R_r^{\max} \qquad \forall\ r,\ t = h, 2h, \ldots, H \tag{4.22}$$

Also, by summing (3.3) from a time $t - h + 1$ to a time t (zeroth order aggregation) we get the following:

$$0 \leq \sum_{t'=t-h+1}^{t} R_{rt'} \leq \sum_{t'=t-h+1}^{t} R_r^{\max} \qquad \forall\ r,\ t = h, 2h, \ldots, H$$

which can be written in terms of the zeroth order aggregated excess resource variables as:

$$0 \leq \tilde{R}_{rt}^{(0)} \leq hR_r^{\max} \qquad \forall\ r,\ t = h, 2h, \ldots, H \tag{4.23}$$

In fact, we can exploit the presence of the linking variables (R_{rt}: $t = h, 2h, \ldots, H$) in the aggregated formulation to obtain a different constraint. This can be achieved by summing (3.3) from $t - h + 1$ to $t - 1$ (instead of t), which results in the following:

$$0 \leq \tilde{R}_{rt}^{(0)} - R_{rt} \leq (h-1)R_r^{\max} \qquad \forall\ r,\ t = h, 2h, \ldots, H \tag{4.24}$$

In this case we are summing over a smaller interval $[t - h + 1, \ldots, t - 1]$ than that given in the general definition of constraint aggregation (4.5) and therefore (4.24) is tighter than (4.23). The aggregation is equivalent to applying the alternative weighting function:

$$f^{(0)}(t - t') \equiv \begin{cases} 1 & t' = t - h + 1, \ldots, t - 1 \\ 0 & t' = t \end{cases} \tag{4.25}$$

to the excess resource capacity constraints. Although the weighting functions used for these inequality constraints are different to that used in section 4.3 for the excess resource balances, it should be emphasised that they give rise to the *same* aggregate variables and linking variables as were introduced in section 4.3.

Constraint (4.24) is the *zeroth order* aggregated excess resource capacity constraint. It contains the linking variables R_{rt} as well as the aggregated variables $\tilde{R}_{rt}^{(0)}$ found in the first order aggregated excess resource balance. The second order aggregate excess resource balance contains $\tilde{R}_{rt}^{(0)}$ and $\tilde{R}_{rt}^{(1)}$ variables as well as the linking variables. To derive the tightest possible excess resource *capacity* constraints that can be written in terms of these variables, we use the following weighting functions†:

$$f_1^{(1)}(t - t') \equiv \begin{cases} (t - t') - 1 & t' = t - h + 1, \ldots, t - 1 \\ 0 & t' = t \end{cases} \tag{4.26a}$$

† The subscripts used for these functions are explained later in section 4.4.3.

$$f_0^{(1)}(t-t') \equiv \begin{cases} h-1-(t-t') & t' = t-h+1, \ldots, t-1 \\ 0 & t' = t \end{cases} \tag{4.26b}$$

The variation of these weighting functions over each ATP is given in figure 4.3.

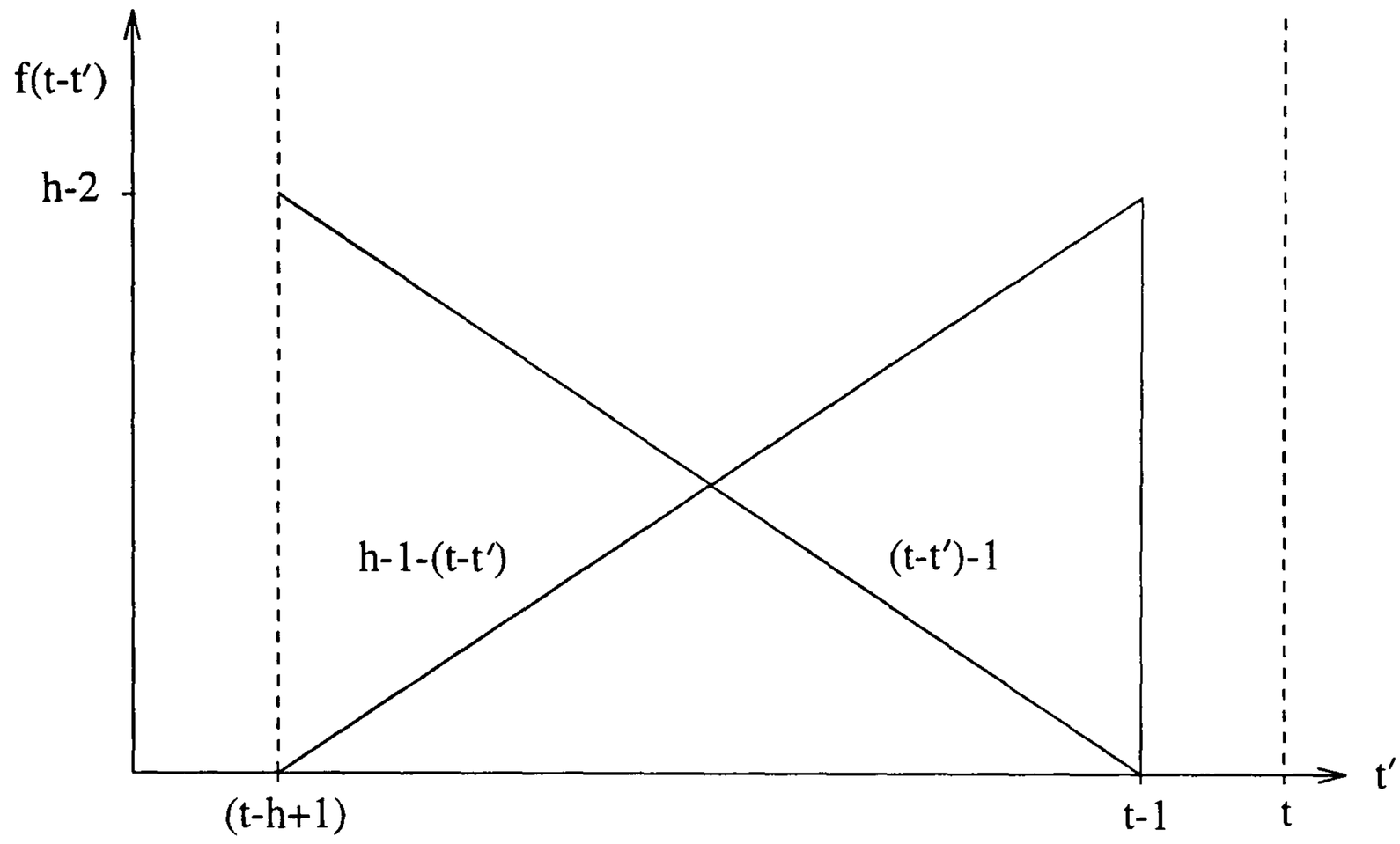


Figure 4.3: Weighting functions for first order aggregate excess resource capacity constraint.

Using the first weighting function (4.26a) we get the following:

$$0 \leq \sum_{t'=t-h+1}^{t} f_1^{(1)}(t-t')\, R_{rt'} \leq \sum_{t'=t-h+1}^{t} f_1^{(1)}(t-t')\, R_r^{\max} \qquad \forall\ r,\ t = h, 2h, \ldots, H$$

This can be written as:

$$0 \leq \sum_{t'=t-h+1}^{t} (t-t'-1)\, R_{rt'} + R_{rt} \leq R_r^{\max} \sum_{t'=t-h+1}^{t-1} (t-t'-1)$$

then as:

$$0 \leq \sum_{t'=t-h+1}^{t} (t-t'+1)\, R_{rt'} - 2 \sum_{t'=t-h+1}^{t} R_{rt'} + R_{rt} \leq \frac{1}{2}(h-1)(h-2)R_r^{\max}$$

before being written in terms of the aggregate and linking variables as required:

$$0 \leq \tilde{R}_{rt}^{(1)} - 2\tilde{R}_{rt}^{(0)} + R_{rt} \leq \frac{1}{2}(h-1)(h-2)R_r^{\max} \qquad \forall\ r,\ t = h, 2h, \ldots, H \tag{4.27a}$$

The same can be done for the second weighting function (4.26b):

$$0 \leq \sum_{t'=t-h+1}^{t} f_0^{(1)}(t-t')\, R_{rt'} \leq \sum_{t'=t-h+1}^{t} f_0^{(1)}(t-t')\, R_r^{\max} \qquad \forall\ r,\ t = h,\ 2h,\ldots\ H$$

This can be written as:

$$0 \leq \sum_{t'=t-h+1}^{t} (h-1-t+t')\, R_{rt'} - (h-1)R_{rt} \leq R_r^{\max} \sum_{t'=t-h+1}^{t-1} (h-1-t+t')$$

then as:

$$0 \leq h \sum_{t'=t-h+1}^{t} R_{rt'} - \sum_{t'=t-h+1}^{t} (t-t'+1)\, R_{rt'} - (h-1)R_{rt} \leq \frac{1}{2}(h-1)(h-2)R_r^{\max}$$

Again, we can write it in terms of aggregate and linking variables:

$$0 \leq h\tilde{R}_{rt}^{(0)} - \tilde{R}_{rt}^{(1)} - (h-1)R_{rt} \leq \frac{1}{2}(h-1)(h-2)R_r^{\max} \qquad \forall\ r,\ t = h,\ 2h,..,\ H \quad (4.27b)$$

The first order excess resource capacity constraints therefore consist of constraints (4.27a) and (4.27b).

4.4.2 Extent Non-Negativity Constraints

Consider the non-negativity constraint on the continuous extent of each task:

$$\xi_{kt} \geq 0 \qquad \forall\ k, t \quad (4.28)$$

This is implicit in the detailed formulation, all variables having a lower bound of zero, and was not written explicitly in chapter 3. Its aggregated counterpart, however, *does* need to be explicitly included in the aggregate formulation as will become apparent below.

As in the previous section, we shall derive an aggregated constraint by summing over time in such a way as to produce the tightest set of constraints from the aggregate and linking variables introduced in section 4.3. The excess resource levels have a single linking variable per ATP (R_{rt}, $t = h, 2h,..,\ H$), whereas the continuous task extents of a task k have τ_k linking variables per ATP ($\xi_{k,t+1-\theta}$, $t = h, 2h,..,\ H$, $\theta = 1,..,\ \tau_k$). We can exploit the presence of these linking variables to generate the zeroth order aggregated extent non-negativity constraints by using the following weighting function:

$$g_k^{(0)}(t-t') \equiv \begin{cases} 1 & t' = t-h+1,\ldots,t-\tau_k \\ 0 & t' = t-\tau_k+1,\ldots,t \end{cases} \quad (4.29)$$

The aggregate constraint formed by summing (4.28) using this weighting function is then,

$$\sum_{t'=t-h+1}^{t} g_k^{(0)}(t-t')\xi_{kt'} \geq 0 \qquad \forall\ k, t = h, 2h, \ldots, H$$

which can be written in terms of the zeroth order aggregate and linking variables as:

$$\tilde{\xi}_{kt}^{(0)} - \sum_{\theta=1}^{\tau_k} \xi_{k,t+1-\theta} \geq 0 \qquad \forall\ k, t = h, 2h, \ldots, H \tag{4.30}$$

Clearly, an aggregate constraint of the same form can also be written for the discrete task extent variables which are also implicitly non-negative in the detailed formulation:

$$\tilde{N}_{kt}^{(0)} - \sum_{\theta=1}^{\tau_k} N_{k,t+1-\theta} \geq 0 \qquad \forall\ k, t = h, 2h, \ldots, H \tag{4.31}$$

We shall now derive the first order aggregate version of (4.28). This means that it can contain only those aggregate continuous task extent variables that occur in the first order aggregate excess resource balance (4.17), namely $\tilde{\xi}_{kt}^{(0)}$ and $\tilde{\xi}_{kt}^{(1)}$. Again, in order to derive the tightest possible constraints using these variables, we shall use the following weighting functions:

$$g_{k1}^{(1)}(t-t') = \begin{cases} (t-t'+1)-(\tau_k+1) & t' = t-h+1, \ldots, t-\tau_k \\ 0 & t' = t-\tau_k+1, \ldots, t \end{cases} \tag{4.32a}$$

$$g_{k0}^{(1)}(t-t') = \begin{cases} h-(t-t'+1) & t' = t-h+1, \ldots, t-\tau_k \\ 0 & t' = t-\tau_k+1, \ldots, t \end{cases} \tag{4.32b}$$

Summing (4.28) over each ATP using the weighting functions (4.32a) and 4.32b), we obtain:

$$\sum_{t'=t-h+1}^{t} g_{k1}^{(1)}(t-t')\xi_{kt'} \geq 0 \qquad \forall\ k, t = h, 2h, \ldots, H$$

and

$$\sum_{t'=t-h+1}^{t} g_{k0}^{(1)}(t-t')\xi_{kt'} \geq 0 \qquad \forall\ k, t = h, 2h, \ldots, H$$

which can be written as:

$$\sum_{t'=t-h+1}^{t} [(t-t'+1)-(\tau_k+1)]\xi_{kt'}$$

$$- \sum_{t'=t-\tau_k+1}^{t} [(t-t'+1)-(\tau_k+1)]\xi_{kt'} \geq 0 \qquad \forall\ k,\ t=h,\ 2h,\ldots,\ H$$

and

$$\sum_{t'=t-h+1}^{t} [h-(t-t'+1)]\xi_{kt'} - \sum_{t'=t-\tau_k+1}^{t} [h-(t-t'+1)]\xi_{kt'} \geq 0 \qquad \forall\ k,\ t=h,\ 2h,\ldots,\ H$$

The two constraints above can be written as:

$$\sum_{t'=t-h+1}^{t} (t-t'+1)\xi_{kt'} - (\tau_k+1) \sum_{t'=t-h+1}^{t} \xi_{kt'}$$

$$- \sum_{t'=t-\tau_k+1}^{t} [(t-t'+1)-(\tau_k+1)]\xi_{kt'} \geq 0 \qquad \forall\ k,\ t=h,\ 2h,\ldots,\ H$$

and

$$h \sum_{t'=t-h+1}^{t} \xi_{kt'} - \sum_{t'=t-h+1}^{t} (t-t'+1)\xi_{kt'} - \sum_{t'=t-\tau_k+1}^{t} [h-(t-t'+1)]\xi_{kt'} \geq 0$$

$$\forall\ k,\ t=h,\ 2h,\ldots,\ H$$

Finally, recognising the aggregate variables (and using the substitution $\theta = t-t'+1$ in the last summation of each left hand side) the two constraints above can be written as:

$$\tilde{\xi}_{kt}^{(1)} - (\tau_k+1)\tilde{\xi}_{kt}^{(0)} + \sum_{\theta=1}^{\tau_k} (\tau_k+1-\theta)\xi_{k,t+1-\theta} \geq 0 \qquad \forall\ k,\ t=h,\ 2h,\ldots,\ H \quad (4.33a)$$

$$h\,\tilde{\xi}_{kt}^{(0)} - \tilde{\xi}_{kt}^{(1)} - \sum_{\theta=1}^{\tau_k} (h-\theta)\xi_{k,t+1-\theta} \geq 0 \qquad \forall\ k,\ t=h,\ 2h,\ldots,\ H \quad (4.33b)$$

Constraints (4.33a) and (4.33b), therefore, represent the first order aggregated non-negativity constraint for the continuous task extent variables. Again, we can write constraints of the same form for the discrete task extent variables:

$$\tilde{N}_{kt}^{(1)} - (\tau_k+1)\tilde{N}_{kt}^{(0)} + \sum_{\theta=1}^{\tau_k} (\tau_k+1-\theta)N_{k,t+1-\theta} \geq 0 \qquad \forall\ k,\ t=h,\ 2h,\ldots,\ H \quad (4.34a)$$

$$h\,\tilde{N}_{kt}^{(0)} - \tilde{N}_{kt}^{(1)} - \sum_{\theta=1}^{\tau_k}(h-\theta)N_{k,t+1-\theta} \geq 0 \qquad \forall\ k,\ t = h, 2h, \ldots, H \tag{4.34b}$$

In fact, as will be shown in the next section, the introduction of aggregated operational constraints means that only (4.33a,b) *or* (4.34a,b) need be included explicitly in the aggregate formulation.

Finally, it should be noted that the implicit constraint that all linking variables must be non-negative:

$$\xi_{k,t+1-\theta} \geq 0 \qquad \forall\ k,\ t = h, 2h, \ldots, H,\ \theta = 1, \ldots, \tau_k$$

does not need to be included since all variables are assumed to have a lower bound of zero anyway.

4.4.3 Operational Constraints

Here we consider the processing equipment capacity constraint (3.4). Of course, this applies directly to the linking task extent variables of the aggregated formulation:

$$V_{kr}^{\min} N_{k,t+1-\theta} \leq \xi_{k,t+1-\theta} \leq V_{kr}^{\max} N_{k,t+1-\theta} \qquad \forall\ k,\ t = h, \ldots, H,\ \theta = 1, \ldots, \tau_k,\ r \in PE_k \tag{4.35}$$

Exactly the same aggregation procedure as in the previous section can be carried out on this constraint to produce its zeroth order aggregated counterpart which is given below:

$$V_{kr}^{\min}\left(\tilde{N}_{kt}^{(0)} - \sum_{\theta=1}^{\tau_k} N_{k,t+1-\theta}\right) \leq \tilde{\xi}_{kt}^{(0)} - \sum_{\theta=1}^{\tau_k} \xi_{k,t+1-\theta} \leq V_{kr}^{\max}\left(\tilde{N}_{kt}^{(0)} - \sum_{\theta=1}^{\tau_k} N_{k,t+1-\theta}\right)$$

$$\forall\ k,\ t = h, 2h, \ldots, H,\ r \in PE_k$$

The first order aggregated operational contraints can also be derived in the same way as the first order aggregated non-negativity constraints in section 4.4.2. They are given below:

$$V_{kr}^{\min}\left(\tilde{N}_{kt}^{(1)} - (\tau_k+1)\tilde{N}_{kt}^{(0)} + \sum_{\theta=1}^{\tau_k}(\tau_k+1-\theta)N_{k,t+1-\theta}\right) \leq$$

$$\tilde{\xi}_{kt}^{(1)} - (\tau_k+1)\tilde{\xi}_{kt}^{(0)} + \sum_{\theta=1}^{\tau_k}(\tau_k+1-\theta)\xi_{k,t+1-\theta}$$

$$\leq V_{kr}^{\max}\left(\tilde{N}_{kt}^{(1)} - (\tau_k+1)\tilde{N}_{kt}^{(0)} + \sum_{\theta=1}^{\tau_k}(\tau_k+1-\theta)N_{k,t+1-\theta}\right)$$

$$\forall\ k, t = h, 2h, .., H, r \in PE_k \qquad (4.36a)$$

and

$$V_{kr}^{\min}\left(h\tilde{N}_{kt}^{(0)} - \tilde{N}_{kt}^{(1)} - \sum_{\theta=1}^{\tau_k}(h-\theta)N_{k,t+1-\theta}\right) \leq h\,\tilde{\xi}_{kt}^{(0)} - \tilde{\xi}_{kt}^{(1)} - \sum_{\theta=1}^{\tau_k}(h-\theta)\xi_{k,t+1-\theta}$$

$$\leq V_{kr}^{\max}\left(h\tilde{N}_{kt}^{(0)} - \tilde{N}_{kt}^{(1)} - \sum_{\theta=1}^{\tau_k}(h-\theta)N_{k,t+1-\theta}\right) \forall\ k, t = h, 2h, .., H, r \in PE_k \qquad (4.36b)$$

As mentioned in the previous section, there is no need to include an aggregate non-negativity constraint for the *discrete* task extent variables N_{kt} since (4.36a) and (4.36b) (combined with the aggregate non-negativity constraints for the *continuous* task extent variables (4.33a) and (4.33b)) ensure that this condition holds anyway provided that $V_{kr}^{\max} > 0$.

Higher order aggregate excess resource capacity and operational constraints can be developed in a similar manner to the first order constraints. Here we shall consider only the derivation of the second order non-negativity constraint for the continuous task extent variables. As for the first order constraints, the second order aggregated operational constraints are derived by performing exactly the same set of operations.

In this case we can use the aggregate variables $\tilde{\xi}_{kt}^{(0)}$, $\tilde{\xi}_{kt}^{(1)}$ and $\tilde{\xi}_{kt}^{(2)}$ along with the usual linking variables $\xi_{k,t+1-\theta}$, $\theta = 1, .., \tau_k$ to generate the aggregate non-negativity constraints. We use the following three weighting functions:

$$g_{k2}^{(2)}(t-t') \equiv \begin{cases} [(t-t'+1)-(\tau_k+1)][(t-t'+1)-(\tau_k+2)] & t'=t-h+1,\ldots,t-\tau_k \\ 0 & t'=t-\tau_k+1,\ldots,t \end{cases} \quad (4.37a)$$

$$g_{k1}^{(2)}(t-t') \equiv \begin{cases} [h-(t-t'+1)][(t-t'+1)-(\tau_k+1)] & t'=t-h+1,\ldots,t-\tau_k \\ 0 & t'=t-\tau_k+1,\ldots,t \end{cases} \quad (4.37b)$$

$$g_{k0}^{(2)}(t-t') \equiv \begin{cases} [h-(t-t'+1)][h-1-(t-t'+1)] & t'=t-h+1,\ldots,t-\tau_k \\ 0 & t'=t-\tau_k+1,\ldots,t \end{cases} \quad (4.37c)$$

Summing (4.28) over each ATP using the weighting functions (4.37a), (4.37b) and (4.37c), we obtain:

$$\sum_{t'=t-h+1}^{t} g_{k2}^{(2)}(t-t')\xi_{kt'} \geq 0 \qquad \forall\ k, t=h,..,H \quad (4.38a)$$

$$\sum_{t'=t-h+1}^{t} g_{k1}^{(2)}(t-t')\xi_{kt'} \geq 0 \qquad \forall\ k, t=h,..,H \quad (4.38b)$$

$$\sum_{t'=t-h+1}^{t} g_{k0}^{(2)}(t-t')\xi_{kt'} \geq 0 \qquad \forall\ k, t=h,..,H \quad (4.38c)$$

which can be written as:

$$\sum_{t'=t-h+1}^{t} [(t-t'+1)-(\tau_k+1)][(t-t'+1)-(\tau_k+2)]\xi_{kt'} -$$

$$\sum_{t'=t-\tau_k+1}^{t} [(t-t'+1)-(\tau_k+1)][(t-t'+1)-(\tau_k+2)]\xi_{kt'} \geq 0 \quad (4.39a)$$

$$\sum_{t'=t-h+1}^{t} [h-(t-t'+1)][(t-t'+1)-(\tau_k+1)]\xi_{kt'} -$$

$$\sum_{t'=t-\tau_k+1}^{t} [h-(t-t'+1)][(t-t'+1)-(\tau_k+1)]\xi_{kt'} \geq 0 \quad (4.39b)$$

$$\sum_{t'=t-h+1}^{t} [h-(t-t'+1)][h-1-(t-t'+1)]\xi_{kt'} \; -$$

$$\sum_{t'=t-\tau_k+1}^{t} [h-(t-t'+1)][h-1-(t-t'+1)]\xi_{kt'} \; \geq \; 0 \qquad (4.39c)$$

and then as:

$$\sum_{t'=t-h+1}^{t} (t-t'+1)^2\xi_{kt'} \; - \; (2\tau_k+3) \sum_{t'=t-h+1}^{t} (t-t'+1)\xi_{kt'} \; + \; (\tau_k+1)(\tau_k+2) \sum_{t'=t-h+1}^{t} \xi_{kt'}$$

$$- \sum_{t'=t-\tau_k+1}^{t} [(t-t'+1)-(\tau_k+1)][(t-t'+1)-(\tau_k+2)]\xi_{kt'} \; \geq \; 0 \qquad (4.40a)$$

$$- \sum_{t'=t-h+1}^{t} (t-t'+1)^2\xi_{kt'} \; + \; (h+\tau_k+1) \sum_{t'=t-h+1}^{t} (t-t'+1)\xi_{kt'} \; - \; h(\tau_k+1) \sum_{t'=t-h+1}^{t} \xi_{kt'}$$

$$- \sum_{t'=t-\tau_k+1}^{t} [h-(t-t'+1)][(t-t'+1)-(\tau_k+1)]\xi_{kt'} \; \geq \; 0 \qquad (4.40b)$$

$$\sum_{t'=t-h+1}^{t} (t-t'+1)^2\xi_{kt'} \; - \; (2h-1) \sum_{t'=t-h+1}^{t} (t-t'+1)\xi_{kt'} \; + \; h(h-1) \sum_{t'=t-h+1}^{t} \xi_{kt'}$$

$$- \sum_{t'=t-\tau_k+1}^{t} [h-(t-t'+1)][h-1-(t-t'+1)]\xi_{kt'} \; \geq \; 0 \qquad (4.40c)$$

Finally, recognising the aggregate variables in (4.40a,b,c) and using the substitution $\theta = t - t' + 1$ in the last summation on the left hand side of each inequality, we obtain:

$$\xi_{kt}^{(2)} - (2\tau_k+3)\xi_{kt}^{(1)} + (\tau_k+1)(\tau_k+2)\xi_{kt}^{(0)} - \sum_{\theta=1}^{\tau_k}[(\tau_k+1)-\theta][(\tau_k+2)-\theta]\xi_{k,t+1-\theta} \; \geq 0 \qquad (4.41a)$$

$$- \; \xi_{kt}^{(2)} \; + \; (h+\tau_k+1)\xi_{kt}^{(1)} \; - \; h(\tau_k+1)\xi_{kt}^{(0)} \; + \; \sum_{\theta=1}^{\tau_k}[h-\theta][(\tau_k+1)-\theta]\xi_{k,t+1-\theta} \; \geq \; 0 \qquad (4.41b)$$

$$\xi_{kt}^{(2)} - (2h-1)\xi_{kt}^{(1)} + h(h-1)\xi_{kt}^{(0)} - \sum_{\theta=1}^{\tau_k}(h-\theta)(h-1-\theta)\xi_{k,t+1-\theta} \geq 0 \qquad (4.41c)$$

$$\forall\ k,\ t = h, 2h, \ldots, H$$

The corresponding operational constraints follow directly from the above constraints in the same way as was demonstrated for the first order aggregated constraints. In the next section we present a general method for generating aggregate excess resource capacity and operational constraints for orders greater than 2.

4.4.4 Higher Order Aggregate Excess Resource Capacity and Operational Constraints

We can generalise the approach adopted in sections 4.4.1, 4.4.2 and 4.4.3 as follows. To derive the m^{th} order aggregate excess resource capacity constraints $m > 0$, we use the following set of weighting functions which are given as functions of $(t - t' + 1)$ rather than $(t - t')$ so as to conform with those used for the excess resource balances:

$$f_y^{(m)}(t-t'+1) \equiv \left\{ \begin{array}{ll} \prod_{v=1}^{y}\left[(t-t'+1)-1-v\right]\prod_{w=1}^{m-y}\left[h+1-(t-t'+1)-w\right] & t' = t-h+1, \ldots, t-1 \\ 0 & t' = t \end{array} \right\}$$

$$y = 0, \ldots, m > 0 \qquad (4.42)$$

For the zeroth order aggregated excess resource capacity constraint, we use the weighting function given by (4.25) which we shall repeat here for completeness:

$$f^{(0)}(t-t'+1) \equiv \begin{cases} 1 & t' = t-h+1, \ldots, t-1 \\ 0 & t' = t \end{cases} \qquad (4.25)$$

Similarly, to derive m^{th} order aggregate operational constraints, we use the following weighting functions (for $m > 0$):

$$g_{ky}^{(m)}(t-t'+1) \equiv \left\{ \begin{array}{ll} \prod_{v=1}^{y}\left[(t-t'+1)-\tau_k-v\right]\prod_{w=1}^{m-y}\left[h+1-(t-t'+1)-w\right] & t' = t-h+1, \ldots, t-\tau_k \\ 0 & t' = t-\tau_k+1, \ldots, t \end{array} \right\}$$

$$y = 0, \ldots, m > 0 \qquad (4.43)$$

To derive the zeroth order operational constraints, we use weighting function (4.29) which

was introduced earlier:

$$g_k^{(0)}(t-t'+1) \equiv \begin{cases} 1 & t' = t-h+1,\ldots,t-\tau_k \\ 0 & t' = t-\tau_k+1,\ldots,t \end{cases} \tag{4.29}$$

After summing over each ATP ($t' = t-h+1,\ldots,t$) using each of the weighting functions from the desired set, we can then write the constraint in terms of the corresponding aggregate variables as shown in sections 4.4.1, 4.4.2 and 4.4.3. It is these constraints, together with the linking variables constraints (4.22) and (4.35), that are then included in the aggregate formulation.

Although it can easily be verified that the general weighting function sets (4.42) and (4.43) given above correctly reduce to the specific sets given in sections 4.4.1, 4.4.2 and 4.4.3, it is not immediately obvious why they are of the precise form that they are. This is expained more fully in the next chapter.

It should be noted that each set of m^{th} order aggregate constraints 'covers' all lower order constraint sets. In other words, all constraint sets of order lower than m can be derived from the m^{th} order constraint set. This has implications when constructing an aggregate formulation (see section 4.6). We can demonstrate this property for the second order aggregate non-negativity constraint set (4.41a,b,c). If (4.41a) is added to (4.41b), we get the following:

$$(-2\tau_k-3+h+\tau_k+1)\bar{\xi}_{kt}^{(1)} + (\tau_k+1)(\tau_k+2-h)\bar{\xi}_{kt}^{(0)} + \sum_{\theta=1}^{\tau_k}(\tau_k+1-\theta)(-\tau_k-2+\theta+h-\theta)\xi_{k,t+1-\theta} \geq 0$$

Simplifying the terms in brackets we get:

$$(h-\tau_k-2)\bar{\xi}_{kt}^{(1)} - (\tau_k+1)(h-\tau_k-2)\bar{\xi}_{kt}^{(0)} + \sum_{\theta=1}^{\tau_k}(\tau_k+1-\theta)(h-\tau_k-2)\xi_{k,t+1-\theta} \geq 0$$

It can be recognised that this is simply (4.33a), the first of the two first order constraints, multiplied by the factor $(h-\tau_k-2)$. Similarly, if (4.41c) is added to (4.41b), we get the following:

$$(-2h+1+h+\tau_k+1)\xi^{(1)}_{kt} \;+\; h(h-1-\tau_k-1)\xi^{(0)}_{kt}$$

$$+\;\sum_{\theta=1}^{\tau_k}(h-\theta)(-h+1+\theta+\tau_k+1-\theta)\xi_{k,t+1-\theta} \;\geq\; 0$$

Again, simplifying the terms in brackets we get:

$$-(h-\tau_k-2)\xi^{(1)}_{kt} \;+\; h(h-\tau_k-2)\xi^{(0)}_{kt} \;-\; \sum_{\theta=1}^{\tau_k}(h-\theta)(h-\tau_k-2)\xi_{k,t+1-\theta}$$

It can be recognised that this is simply (4.33b), the other first order constraint, multiplied by the same factor $(h-\tau_k-2)$. We have shown, therefore, that the second order extent non-negativity constraints cover the first order extent non-negativity constraints. This covering property is proved for the general case in section 5.5 of the next chapter.

4.5 Aggregate Objective Function

As mentioned in the previous chapter, the objective function depends on the type of problem being solved. We shall consider the detailed objective function (3.5) from the previous chapter which represents nett profit over the planning horizon taking into account utility costs, storage costs, and fixed and variable costs associated with tasks:

$$\max \sum_{r\in MR}\left(C^F_r(R_{rH}-R_{r0}) \;-\; C^I_r\sum_{t=1}^{H}R_{rt}\right) - \sum_k\sum_{t=1}^{H}(C^F_k N_{kt}+C^V_k\xi_{kt}) \;-\; \sum_{t=1}^{H}C^E_{rt}\Pi_{rt} \qquad (3.5)$$

To form the aggregate form of this objective function, all terms involving a summation of variables over time are written in terms of the zeroth order aggregate variables to give the following:

$$\max \sum_{r\in MR}\left(C^F_r(R_{rH}-R_{r0}) \;-\; C^I_r\sum_{t=h,2h,..,H}\tilde{R}^{(0)}_{rt}\right) - \sum_k\sum_{t=h,2h,..,H}(C^F_k\tilde{N}^{(0)}_{kt}+C^V_k\tilde{\xi}^{(0)}_{kt})$$

$$-\sum_{t=h,2h,..,H}C^E_{rt}\tilde{\Pi}^{(0)}_{rt} \qquad (4.44)$$

Since it contains only zeroth order variables, this aggregate objective function is independent of the overall order of the formulation.

4.6 Constructing Aggregate Formulations

We have shown how aggregate constraints of different orders can be derived from the constraints present in the detailed formulation. It is then necessary to use these constraints to construct aggregate formulations.

The m^{th} order aggregate *formulation* involves the following variables:

- the linking excess resource variables

$$R_{rt} \quad \forall\ r,\ t = h,\ 2h,\ldots,\ H$$

and the linking task extent variables

$$N_{k,t+1-\theta}\ ,\quad \xi_{k,t+1-\theta} \quad \forall\ k,\ t = h,\ 2h,\ldots,\ H,\ \theta = 1,\ldots,\ \tau_k$$

- the aggregate excess resource variables

$$\tilde{R}_{rt}^{(p)} \quad \forall\ r,\ t = h,\ 2h,\ldots,\ H,\ p = 0,\ldots,\ m-1$$

defined by:

$$\tilde{R}_{rt}^{(p)} \equiv \sum_{t'=t-h+1}^{t} (t-t'+1)^p R_{rt'}$$

- the aggregate task extent variables

$$\tilde{N}_{kt}^{(p)}\ ,\quad \tilde{\xi}_{kt}^{(p)} \quad \forall\ k,\ t = h,\ 2h,\ldots,\ H,\ p = 0,\ldots,\ m$$

defined by:

$$\tilde{N}_{kt}^{(p)} \equiv \sum_{t'=t-h+1}^{t} (t-t'+1)^p N_{kt'} \qquad ; \qquad \tilde{\xi}_{kt}^{(p)} \equiv \sum_{t'=t-h+1}^{t} (t-t'+1)^p \xi_{kt'}$$

The m^{th} order aggregate formulation involves the aggregate objective function (4.44) and the following constraints:

- the set of aggregated excess resource balances of order $p = 0,\ 1,\ldots,\ m$;
- the aggregated excess resource capacity constraints of order $m-1$;

- the aggregated non-negativity and operational constraints of order m.

It should be noted that all the aggregated excess resource balances from order 0 to m are included in the m^{th} order formulation. This is due to the form of (4.16). A constraint of order p contains *all* the variables contained by any lower order constraint. 'Missing out' constraints of any orders less than m is not, therefore, desirable since it may well decrease the accuracy of the approximation and will not result in any savings in terms of the number of variables.

For the excess resource capacity and operational constraints, on the other hand, only the highest order aggregation is included. This is due to the covering property mentioned in section 4.4.4.

Note, also, that the aggregate excess resource capacity constraints used are of order $m-1$ rather than m since the constraints in the m^{th} order aggregated formulation do not include the variables $\tilde{R}_{rt}^{(m)}$.

The complete first and second order aggregate formulations are listed in Appendix B.

4.7 Concluding Remarks

This chapter has shown how a family of aggregate formulations can be constructed from the detailed RTN process scheduling formulation using a set of rigorous mathematical manipulations. The derivations raise a number of interesting questions, such as:

(a) What is the relation between the aggregate formulations and the detailed one?

and,

(b) What is the relation between aggregate formulations of increasing order?

The discussion in the next chapter will seek to answer these and other related questions. By examining the detailed properties of the aggregate formulation, the next chapter will also provide some *a posteriori* justification regarding the form of the mathematical manipulations used in this chapter.

Chapter 5

Theoretical Properties of the Aggregate RTN Formulations

The last chapter described a general aggregation technique involving the summation of the detailed RTN constraints using a time-dependent weighting function to form constraints that could be written in terms of aggregate variables. It was also shown how these aggregate constraints could be used to construct an aggregate RTN formulation. In this chapter, some important mathematical properties of this aggregate formulation are demonstrated. These will provide some justification for the form of aggregation used in the last chapter and also serve to answer the two questions posed at the end of it.

5.1 Property I: Strict Relaxation of Detailed Formulation

All the aggregated RTN constraints have been derived from the detailed RTN formulation by performing valid mathematical manipulations on the equality and inequality constraints. If we consider a scheduling problem in which the objective is to maximise profit, therefore, the optimal aggregate objective value will always provide a valid upper bound on the optimal objective value of the solution to the detailed formulation of the same problem. Hence the aggregate formulation can be used confidently in any application in which a valid upper bound is required. It will be shown in the example presented in section 5.6 that this bound can be quite a tight one.

To illustrate the relationship between the aggregate and detailed formulations we can consider the aggregate formulation not in terms of the aggregate variables, but in terms of the original detailed variables that they replace. We can then compare the solution space of the aggregate formulation with that of the detailed formulation since both contain the same set of variables. Since we have summed constraints, the solution space (also referred to as the feasible region) defined by the aggregate constraints must contain the solution space defined by the detailed constraints, as illustrated in figure 5.1.

The objective functions of the two formulations (when both are written in terms of the detailed variable set) are identical but the *optimal* objective value of the aggregate formulation may be higher since the aggregate solution space is larger. For the remainder of this thesis, mention will be made of the 'accuracy' of the aggregate formulation with respect to the detailed formulation which can be taken to be a measure of the similarity of their respective feasible regions. One obvious way to quantify this is by comparing the optimal objective

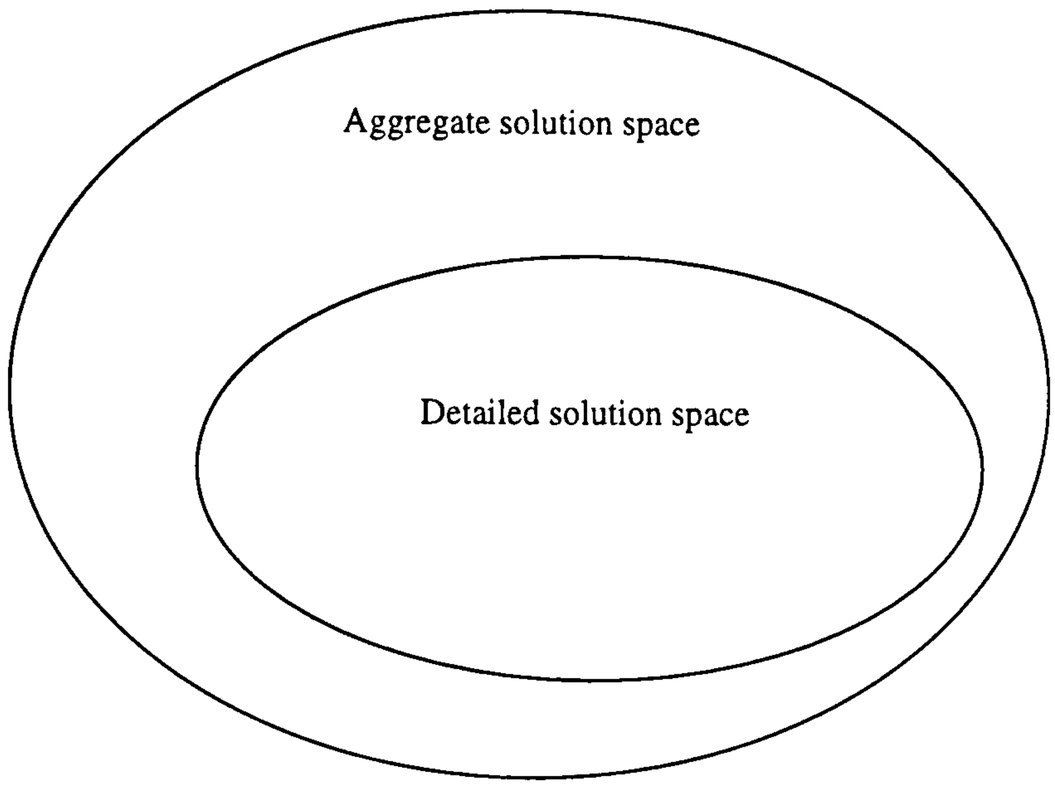


Figure 5.1: Solution spaces of the detailed and aggregate formulations.

values of each formulation. This, however, may be somewhat misleading since, even if the optimal objective values are the same, it does not mean that the aggregate formulation is completely 'accurate'. In other words, the aggregate feasible region is not necessarily identical to the detailed feasible region.

We can apply a similar argument when considering the relationship between aggregate formulations of different orders. The accuracy of the $(m+1)^{th}$ order aggregate formulation of a certain problem must be greater than or equal to that of the m^{th} order formulation. This is due to the fact that the m^{th} order formulation contains all the aggregated excess resource balance constraints from zeroth to m^{th} order, whereas the $(m+1)^{th}$ order formulation contains all these constraints as well as the $(m+1)^{th}$ order constraint. As far as the aggregated capacity and operational constraints are concerned, the $(m+1)^{th}$ order constraints cover the m^{th} order constraints (see section 5.6 later in this chapter).

In general, having more constraints in an aggregate formulation does not necessarily mean that its accuracy must increase. For instance, it would stay the same if the higher order constraints are redundant and do not tighten the approximation represented by the aggregate formulation. This, however, is *not* the case. In fact, it is shown later (in section 5.3) that the aggregate formulation becomes exact (*i.e.* completely equivalent with the detailed formulation) for sufficiently high orders. We can therefore generate a whole family of aggregate formulations of increasing accuracy and complexity.

For many applications, low order aggregate formulations turn out to provide a sufficiently good approximation to the detailed formulation. However, the lowest (zeroth) order

formulation is of no use for any RTNs in which some resources are consumed and then later produced (*i.e.* 'borrowed') by any tasks. This is explained in more detail in section 6.1. Many practical RTNs, including the ones considered in this thesis, *do* contain such resources (equipment items, renewable utilities *etc.*). The zeroth order formulation, therefore, is only very narrowly applicable and shall not be discussed further in this thesis. On the other hand, first and second order aggregated formulations *are* generally applicable and often provide a good approximation to the detailed RTN formulation.

5.2 Property II: Reduction to Detailed Formulation for h→1

It is interesting to note that, when the ATP length $h = 1$, an aggregate formulation of any order is equivalent to the original detailed formulation from which it is derived. This is an important property — the accuracy of the aggregate formulation should increase with decreasing ATP length since this increases the number of constraints. Eventually, when the ATP length is reduced to a single time interval, the aggregate formulation becomes exact (*i.e.* equivalent to the detailed formulation. As demonstrated below, this behaviour is closely related to the incorporation of the linking variables within the formulation.

Firstly, consider the general p^{th} order aggregate excess resource balance presented in section 4.3.2 and reproduced below:

$$\sum_{t'=t-h+1}^{t}\left\{(t-t'+1)^p-(t-t')^p\right\}R_{rt'} - h^p R_{r,t-h} = \sum_k\left(\sum_{\theta=0}^{\tau_k}\mu_{kr\theta}\right)\sum_{t'=t-h+1}^{t}(t-t'+1-\theta)^p N_{kt'}$$

$$+\sum_k\sum_{\theta'=1}^{\tau_k}N_{k,t-h+1-\theta'}\sum_{\theta=\theta'}^{\tau_k}(h+\theta'-\theta)^p\,\mu_{kr\theta} - \sum_k\sum_{\theta'=1}^{\tau_k}N_{k,t+1-\theta'}\sum_{\theta=\theta'}^{\tau_k}(\theta'-\theta)^p\,\mu_{kr\theta}$$

$$+\sum_k\left(\sum_{\theta=0}^{\tau_k}\nu_{kr\theta}\right)\sum_{t'=t-h+1}^{t}(t-t'+1-\theta)^p\xi_{kt'} +$$

$$\sum_k\sum_{\theta'=1}^{\tau_k}\xi_{k,t-h+1-\theta'}\sum_{\theta=\theta'}^{\tau_k}(h+\theta'-\theta)^p\,\nu_{kr\theta} - \sum_k\sum_{\theta'=1}^{\tau_k}\xi_{k,t+1-\theta'}\sum_{\theta=\theta'}^{\tau_k}(\theta'-\theta)^p\,\nu_{kr\theta}$$

$$+\sum_{t'=t-h+1}^{t}(t-t'+1)^p\Pi_{rt'} \qquad (4.16)$$

Consider first the left hand side of (4.16). For $h = 1$, the sum reduces to a single term with $t' = t$. Hence the last term on the right hand side becomes Π_{rt} and the left hand side becomes:

$$\left(1^p - 0^p\right)R_{rt} \;-\; 1^p R_{r,t-1} \;=\; R_{rt} - R_{r,t-1}$$

Now consider the three discrete extent terms on the right hand side of (4.16). For $h = 1$, the first and second of these terms become:

$$\sum_k \left(\sum_{\theta=0}^{\tau_k} \mu_{kr\theta}\right)(1-\theta)^p N_{kt} \;+\; \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t-\theta'} \sum_{\theta=\theta'}^{\tau_k} (1+\theta'-\theta)^p \;\mu_{kr\theta} \qquad (5.1)$$

Now consider the third term. If we make the substitution $\theta'' = \theta' - 1$, it can be written as follows:

$$\sum_k \sum_{\theta''=0}^{\tau_k - 1} N_{k,t-\theta''} \sum_{\theta=\theta''+1}^{\tau_k} (1+\theta''-\theta)^p \;\mu_{kr\theta}$$

$$= \sum_k \sum_{\theta''=0}^{\tau_k - 1} N_{k,t-\theta''} \left[\sum_{\theta=\theta''}^{\tau_k} (1+\theta''-\theta)^p \;\mu_{kr\theta} \;-\; (1)^p \;\mu_{kr\theta''}\right]$$

$$= \sum_k \sum_{\theta''=1}^{\tau_k} N_{k,t-\theta''} \sum_{\theta=\theta''}^{\tau_k} (1+\theta''-\theta)^p \;\mu_{kr\theta} \;+\; \sum_k N_{kt} \sum_{\theta=0}^{\tau_k} (1-\theta)^p \;\mu_{kr\theta}$$

$$- \sum_k N_{k,t-\tau_k} (1)^p \;\mu_{kr\tau_k} \;-\; \sum_k \sum_{\theta''=0}^{\tau_k - 1} N_{k,t-\theta''} (1)^p \;\mu_{kr\theta''}$$

$$= \sum_k \sum_{\theta''=1}^{\tau_k} N_{k,t-\theta''} \sum_{\theta=\theta''}^{\tau_k} (1+\theta''-\theta)^p \;\mu_{kr\theta} \;+\; \sum_k N_{kt} \sum_{\theta=0}^{\tau_k} (1-\theta)^p \;\mu_{kr\theta}$$

$$- \sum_k N_{k,t-\tau_k} \mu_{kr\tau_k} \;-\; \sum_k \sum_{\theta''=0}^{\tau_k - 1} N_{k,t-\theta''} \mu_{kr\theta''} \qquad (5.2)$$

Overall, then, the first three terms on the right hand side of (4.16) are equivalent to expression (5.1) minus (5.2):

$$\sum_k \left(\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \right) (1-\theta)^p N_{kt} \; + \; \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t-\theta'} \sum_{\theta=\theta'}^{\tau_k} (1+\theta'-\theta)^p \; \mu_{kr\theta}$$

$$- \; \sum_k \sum_{\theta''=1}^{\tau_k} N_{k,t-\theta''} \sum_{\theta=\theta''}^{\tau_k} (1+\theta''-\theta)^p \; \mu_{kr\theta} \; - \; \sum_k N_{kt} \sum_{\theta=0}^{\tau_k} (1-\theta)^p \; \mu_{kr\theta}$$

$$+ \; \sum_k N_{k,t-\tau_k} \mu_{kr\tau_k} \; + \; \sum_k \sum_{\theta''=0}^{\tau_k - 1} N_{k,t-\theta''} \mu_{kr\theta''}$$

We note that the first four terms of the above expression immediately cancel out, while the last two can be combined to give:

$$\sum_k \sum_{\theta=0}^{\tau_k} N_{k,t-\theta} \mu_{kr\theta}$$

Equivalent manipulations can be applied to the fourth, fifth and sixth terms on the right hand side of (4.16) involving the continuous extent variables. Hence, we have shown that (4.16) reduces to the following for $h = 1$:

$$R_{rt} - R_{r,t-1} \; = \; \sum_k \sum_{\theta=0}^{\tau_k} N_{k,t-\theta} \mu_{kr\theta} \; + \; \sum_k \sum_{\theta=0}^{\tau_k} \xi_{k,t-\theta} \nu_{kr\theta} \; + \; \Pi_{rt}$$

This can readily be recognised as the detailed excess resource balance (3.2).

Turning now to the capacity and operational constraints, it can be seen that the generalised weighting functions (4.25), (4.42), (4.29) and (4.43) of section 4.4.4 reduce to zero for $h = 1$ and consequently only the linking variable constraints remain:

$$0 \; \leq \; R_{rt} \; \leq \; R_r^{\max} \qquad \forall \; r, \; t = h, \ldots, H \tag{4.22}$$

$$V_{kr}^{\min} \; N_{k,t+1-\theta} \; \leq \; \xi_{k,t+1-\theta} \; \leq \; V_{kr}^{\max} \; N_{k,t+1-\theta} \qquad \forall \; k, t = h, \ldots, H, \theta = 1, \ldots, \tau_k, \; r \in PE_k \tag{4.35}$$

For $h = 1$, these apply for $t = 1, 2, \ldots, H$. Thus, (4.22) becomes equivalent to the detailed excess resource capacity constraint (3.3). In addition (4.35) becomes equivalent to the detailed operational constraints (3.4) although there is some redundancy if they are written for $\theta = 1, \ldots, \tau_k$ rather than just for $\theta = 1$ as necessary.

In conclusion, then, the aggregate RTN formulation tends to the detailed RTN formulation as $h \rightarrow 1$. In a sense, this is to be expected given the rigorous nature of its derivation. A different kind of limiting behaviour is discussed in the next section.

5.3 Property III: Reduction to Detailed Formulation for m→h – 1

In this section it shall be shown that an aggregate RTN formulation of order $m = h - 1$ is equivalent to the detailed RTN formulation. To do this, we shall first write the detailed formulation in matrix form. The aggregation procedure described in the previous chapter can then be formalised as a matrix transformation which maps one constraint space to another.

We shall first consider the excess resource balances (equality constraints) before dealing with the operational and capacity constraints which are inequalities and are treated differently.

5.3.1 Excess Resource Balances

Consider any equality constraint in a general discrete-time scheduling formulation (*e.g* the excess resource balance of the detailed RTN formulation). In general, this constraint will apply for all times in the planning horizon of length H and can be written in vector notation as:

$$\mathbf{c} = \mathbf{0} \tag{5.3}$$

where:

$$\mathbf{c} = \begin{bmatrix} C_1(\,.\,) \\ C_2(\,.\,) \\ . \\ C_t(\,.\,) \\ . \\ C_H(\,.\,) \end{bmatrix}$$

and **0** is a null vector, also having H elements. We now aggregate the constraints in the way described in section 4.2, using an ATP of length h and aggregation operators of order $p = 0, \ldots, m$. We write the following aggregate constraints:

$$\sum_{t'=t-h+1}^{t} f^p(t-t')\, C_{t'}(\,.\,) \;=\; 0 \qquad (p = 0, 1, \ldots, m \;\;,\;\; t = h, 2h, \ldots, H \equiv nh) \tag{4.5}$$

Again we shall use the weighting functions introduced in section 4.2:

$$f^p(t-t') \;\equiv\; (t-t'+1)^p \qquad (p = 0, 1, \ldots, m) \tag{4.6}$$

We can then verify that the aggregated constraints can be written in matrix form as:

$$\mathbf{A}^{(m,h)}\mathbf{c} = \mathbf{0} \tag{5.4}$$

In (5.4), **0** is a null vector with $n(m+1)$ elements and $\mathbf{A}^{(m,h)}$ is an $n \times n$ block matrix given by:

$$\mathbf{A}^{(m,h)} = \begin{bmatrix} \mathbf{S}^{(m,h)} & \mathbf{0} & . & \mathbf{0} \\ \mathbf{0} & \mathbf{S}^{(m,h)} & . & \mathbf{0} \\ . & . & . & . \\ \mathbf{0} & \mathbf{0} & . & \mathbf{S}^{(m,h)} \end{bmatrix} \tag{5.5}$$

Each block of $\mathbf{A}^{(m,h)}$ is itself a $(m+1) \times h$ matrix. In particular, $\mathbf{S}^{(m,h)}$ is given by:

$$\mathbf{S}^{(m,h)} = \begin{bmatrix} 1 & 1 & . & 1 & . & 1 \\ h & (h-1) & . & j & . & 1 \\ . & . & . & . & . & . \\ h^p & (h-1)^p & . & j^p & . & 1 \\ . & . & . & . & . & . \\ h^m & (h-1)^m & . & j^m & . & 1 \end{bmatrix} \tag{5.6}$$

In (5.5), **0** is a $(m+1) \times h$ null matrix.

We note that $\mathbf{S}^{(m,h)}$ is generally rectangular $(m+1 < h)$. However, when $m = h-1$, this matrix is square and furthermore nonsingular†. In this case, $\mathbf{A}^{(m,h)}$ is also square and non-singular since it is square block diagonal with each diagonal block corresponding to the non-singular $\mathbf{S}^{(m,h)}$. Hence, when $m = h-1$, the inverse of $\mathbf{A}^{(m,h)}$ exists and therefore the original constraints (5.5) may be recovered from the aggregated ones (5.6) by pre-multiplying the latter by $\mathbf{A}^{(m,h)^{-1}}$.

It is interesting to note that Property II (see section 5.2) can also be proved by realising that $\mathbf{A}^{(m,h)}$ reduces to the identity matrix $\mathbf{I}^{H \times H}$ when $h = 1$.

5.3.2 Capacity and Operational Constraints

The detailed capacity and operational constraints (3.3) and (3.4) differ from the excess resource balances in that they are inequalities rather than equalities. As a consequence of this difference, the aggregate operational constraints are formed by using different weighting function sets. These were defined in section 4.4.4 and are recalled below:

---

† In fact, this matrix is a special case of the Vandermonde matrix named after the French mathematician A.T. Vandermonde (1735-1796). He derived a general expression for its determinant showing that it is always non-zero.

$$f_y^{(m)}(t-t'+1) \equiv \left\{ \begin{array}{ll} \prod_{v=1}^{y}\left[(t-t'+1)-1-v\right]\prod_{w=1}^{m-y}\left[h+1-(t-t'+1)-w\right] & t'=t-h+1,\ldots,t-1 \\ 0 & t'=t \end{array} \right\}$$

$$y=0,..,m>0 \qquad (4.42)$$

$$g_{ky}^{(m)}(t-t'+1) \equiv \left\{ \begin{array}{ll} \prod_{v=1}^{y}\left[(t-t'+1)-\tau_k-v\right]\prod_{w=1}^{m-y}\left[h+1-(t-t'+1)-w\right] & t'=t-h+1,..,t-\tau_k \\ 0 & t'=t-\tau_k+1,..,t \end{array} \right\}$$

$$y=0,..,m>0 \qquad (4.43)$$

Like those used for the excess resource balances, these weighting function sets are polynomials in $(t-t'+1)$ and are summed from $t'=t-h+1$ to t (*i.e.* $(t-t'+1)=h$ to $(t-t'+1)=1$). They ensure that the tightest possible aggregate constraints are derived using only the variables used in the aggregate excess resource balance. Furthermore, they ensure that the aggregate operational constraints become exact if a sufficiently high order is used in a similar way to the aggregate excess resource balances.

The operational constraints also differ in that each constraint $C_t(\ .\ )$ only involves one variable at that time t whereas the excess resource balances also involve variables at previous times $t-1$, $t-2$ *etc.* The result of this is that there are no constraints linking variables in different ATPs. For the following argument, therefore, we need only consider a general ATP.

As before, we represent the detailed variables by vectors but we only need to consider those spanned by a single ATP. We define the following:

$$\mathbf{r}_{rt} \equiv \begin{bmatrix} R_{r,t-h+1} \\ . \\ R_{rt} \end{bmatrix} \qquad \forall\ r,\ t=h,\ 2h,..,\ H \qquad (5.7)$$

$$\mathbf{x}_{kt} \equiv \begin{bmatrix} \xi_{k,t-h+1} \\ . \\ \xi_{kt} \end{bmatrix} \qquad \forall\ k,\ t=h,\ 2h,..,\ H \qquad (5.8)$$

$$\mathbf{n}_{kt} \equiv \begin{bmatrix} N_{k,t-h+1} \\ . \\ N_{kt} \end{bmatrix} \qquad \forall\ k,\ t=h,\ 2h,..,\ H \qquad (5.9)$$

We also introduce the $(m+1)\times h$ matrices $\mathbf{F}^{(m,h)}$ and $\mathbf{G}_k^{(m,h)}$, the $(i,j)^{th}$ elements of which can be defined in terms of the weighting functions (4.42) and (4.43):

$$[\mathbf{F}^{(m,h)}]_{ij} \equiv f_{i-1}^{(m)}(j) \qquad i=1,..,m+1\ ,\ j=1,..,h \tag{5.10}$$

$$[\mathbf{G}_k^{(m,h)}]_{ij} \equiv g_{k,i-1}^{(m)}(j) \qquad i=1,..,m+1\ ,\ j=1,..,h \tag{5.11}$$

$\mathbf{F}^{(m,h)}$ then operates on the detailed excess resource capacity constraints (3.3) to form the aggregate constraints:

$$\mathbf{0} \le \mathbf{F}^{(m,h)}\,\mathbf{r}_{rt} \le \mathbf{F}^{(m,h)}\,\mathbf{R}_{\mathbf{r}}^{\max} \tag{5.12}$$

while $\mathbf{G}_k^{(m,h)}$ is applied to the detailed implicit non-negativity constraint (4.28) and also the operational constraint (3.4) leading to:

$$\mathbf{0} \le \mathbf{G}_k^{(m,h)}\,\mathbf{x}_{kt} \tag{5.13}$$

$$V_k^{\min}\,\mathbf{G}_k^{(m,h)}\,\mathbf{n}_{kt} \le \mathbf{G}_k^{(m,h)}\,\mathbf{x}_{kt} \le V_k^{\max}\,\mathbf{G}_k^{(m,h)}\,\mathbf{n}_{kt} \tag{5.14}$$

As well as these aggregate capacity and operational constraints, the aggregate formulation also contains separate capacity and operational constraints for each of the linking variables. We shall now show that the operational aggregated and linking constraints become equivalent to the detailed operational constraints when $m = h - \tau_k - 1$.

The elements of $\mathbf{G}_k^{(m,h)}$ are always zero for the times in the ATP corresponding to the linking variables: $t' = t+1-\theta\ ,\ \theta = 1,..,\tau_k$. This can be seen from the weighting functions which are forced to be zero for these times. Otherwise, the elements are given by a vector of $m+1$ polynomials which are evaluated at times $t' = t-h+1\ ,\ ..\ ,t-\tau_k$. This is best illustrated by an example. Consider a single ATP of length $h = 8$ time intervals and a task k of duration $\tau_k = 3$ time intervals. Suppose that the order of aggregation m is set at $m = h - \tau_k - 1 = 4$. Then, using (5.11) and the weighting functions (4.43), we can now write the 5×8 matrix $\mathbf{G}_k^{(m,h)}$ as follows:

$$\mathbf{G}_k^{(m,h)} = \begin{bmatrix} 0 & 0 & 0 & \{(8-q)(7-q)(6-q)(5-q)\ ,\ q=4,..,8\} \\ 0 & 0 & 0 & \{(q-4)(8-q)(7-q)(6-q)\ ,\ q=4,..,8\} \\ 0 & 0 & 0 & \{(q-4)(q-5)(8-q)(7-q)\ ,\ q=4,..,8\} \\ 0 & 0 & 0 & \{(q-4)(q-5)(q-6)(8-q)\ ,\ q=4,..,8\} \\ 0 & 0 & 0 & \{(q-4)(q-5)(q-6)(q-7)\ ,\ q=4,..,8\} \end{bmatrix}$$

Here q has been used to represent the quantity $(t-t'+1)$ for simplicity. It can be seen that

the polynomial in q in each row has roots at every value of q for which it is evaluated except one, at which it always takes a positive value. A different root is left out of the polynomial for each row which means that we get the following form for $\mathbf{G}_k^{(m,h)}$:

$$\mathbf{G}_k^{(m,h)} = \begin{bmatrix} 0 & 0 & 0 & 24 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 24 \end{bmatrix}$$

In this case, the aggregated non-negativity constraint, for example, can therefore be written as:

$$\begin{bmatrix} 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \\ 0 \end{bmatrix} \leq \begin{bmatrix} 0 & 0 & 0 & 24 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 6 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 4 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 6 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 24 \end{bmatrix} \begin{bmatrix} \xi_{k8} \\ \xi_{k7} \\ \xi_{k6} \\ \xi_{k5} \\ \xi_{k4} \\ \xi_{k3} \\ \xi_{k2} \\ \xi_{k1} \end{bmatrix}$$

The above is equivalent to the following constraint:

$$0 \leq \xi_{kt'} \qquad t' = 1, .., 5 \tag{5.15}$$

This enforces non-negativity of the ξ_{kt} variables for each time in the ATP except for those times at which linking variables are defined. But each of these linking variables is subject to an implicit non-negativity constraint anyway:

$$0 \leq \xi_{k,t+1-\theta} \qquad \theta = 1, .., 3$$

or:

$$0 \leq \xi_{kt'} \qquad t' = 6, .., 8 \tag{5.16}$$

For this example, therefore, we have shown that the aggregated constraint (5.15) and the linking variable constraint (5.16) together enforce non-negativity at all times in the ATP and are therefore equivalent to the detailed non-negativity constraint:

$$0 \leq \xi_{kt'} \qquad t' = 1, .., 8$$

If the weighting functions (4.42) and (4.43) are examined in detail, it can be seen that this result is generally true — *i.e.* for $m = h - \tau_k - 1$, $\mathbf{G}_k^{(m,h)}$ has h columns and $(m + 1) = h - \tau_k$

rows and reduces to a form which can be partitioned into two distinct matrices:

$$\mathbf{G}_k^{(h-\tau_k-1,h)} = \left[\begin{array}{c|c} \mathbf{0}^{(h-\tau_k)\times\tau_k} & \mathbf{D}^{(h-\tau_k)\times(h-\tau_k)} \end{array} \right] \tag{5.17}$$

where $\mathbf{0}^{(h-\tau_k)\times\tau_k}$ is a null matrix and $\mathbf{D}^{(h-\tau_k)\times(h-\tau_k)}$ is a diagonal matrix whose non-zero elements are all positive.

In such a case, the aggregated extent non-negativity constraint (5.13) becomes equivalent to:

$$0 \leq \xi_{kt'} \qquad t' = 1,..,t-\tau_k \tag{5.18}$$

which, combined with the implicit non-negativity constraints on each linking variable:

$$0 \leq \xi_{kt'} \qquad t' = t-\tau_k+1,..,h$$

is obviously equivalent to the original detailed non-negativity constraints which apply for each time in the ATP:

$$0 \leq \xi_{kt'} \qquad t' = 1,..,h$$

Exactly the same can be shown for the operational constraints (5.13).

If we now turn our attention to the aggregated excess resource capacity constraints (5.11) it can be verified that, when combined with the linking constraints (4.25), these become equivalent to the original detailed constraints for $m = h-2$. This is to be expected since the weighting function (4.42) is, in fact, a special case of (4.43) with $\tau_k = 1$. For completeness, therefore, we can write the following:

$$\mathbf{F}^{(h-2,h)} = \left[\begin{array}{c|c} \mathbf{0}^{(h-1)\times 1} & \mathbf{D}^{(h-1)\times(h-1)} \end{array} \right] \tag{5.19}$$

where $\mathbf{0}^{(h-1)\times 1}$ is a null vector and $\mathbf{D}^{(h-1)\times(h-1)}$ is a diagonal matrix whose non-zero elements are all positive.

5.4 Property IV: Maximal Aggregation Order

It was shown in section 5.3.1 that the set of aggregated excess resource balance constraints of orders $p = 0,..,h-1$ (where h is the ATP length) is exactly equivalent to the detailed constraints. This corresponds to the matrix $\mathbf{A}^{(m,h)}$ becoming square and non-singular. Any aggregated constraints of order greater than $h-1$ would correspond to $\mathbf{A}^{(m,h)}$ having more rows

than columns and must be redundant with respect to this set. Hence we can say that $h-1$ is the maximal order for the aggregated excess resource balance constraints.

For the operational and capacity constraints, the situation is slightly different. Looking again at (5.18), it is natural to ask what happens to $\mathbf{G}_k^{(m,h)}$ for $m > h - \tau_k - 1$. The answer is that *all* the elements of the matrix become zero (the polynomials have roots at all values of $t - t' + 1$ at which they are evaluated). So we can write:

$$m > h - \tau_k - 1 \quad \Rightarrow \quad \mathbf{G}_k^{(m,h)} = \mathbf{0}^{(m+1)\times h} \tag{5.20}$$

where $\mathbf{0}^{(m+1)\times h}$ is a null matrix. Consequently, any aggregate constraints formed from such a matrix are of no use at all. This indicates that we should place a limit on the overall order of any formulation of the type defined in section 4.6 given by:

$$m^{\max} = h - \tau^{\max} - 1 \tag{5.21}$$

where $\tau^{\max}$ is the maximum duration of any task in the RTN under consideration:

$$\tau^{\max} \equiv \max_k \tau_k \tag{5.22}$$

It is conceivable that there may be a few particularly long tasks in the RTN which make the maximum order $m^{\max}$ given by (5.21) overconstraining for the shorter tasks. If it is desired to use a formulation with an overall order m^+ which is higher than $m^{\max}$, then we need to consider each task individually. We can introduce an *effective task order* m_k^{eff} for each task k given by:

$$m_k^{eff} = \min\left(m^+ \ , \ (h - \tau_k - 1) \right) \qquad \forall \ k \tag{5.23}$$

For each task k, therefore, the following aggregated and linking variables will appear in the operational constraints:

$$\tilde{N}_{kt}^{(i)} \qquad (i = 0, \ldots, m_k^{eff}, \ t = h, \ldots, H)$$

$$N_{k,t+1-\theta} \qquad (\theta = 1, \ldots, \tau_k, \ t = h, \ldots, H)$$

This then implies that, for the longer tasks k which have $m_k^{eff} \leq m^+$, the operational constraints will be exact for the reasons given above — it can be seen that there are h variables per ATP for these tasks as for the original detailed formulation.

For these tasks, the aggregate excess resource balances of order $p > h - \tau_k - 1$ can be expressed completely in terms of the aggregate extent variables of order $p \leq h - \tau_k - 1$. This is clear since there are a total of h aggregate and linking extent variables per ATP for these tasks. Any aggregate extent variable of order $p > h - \tau_k - 1$, therefore, can be uniquely expressed as a linear combination of the $h - \tau_k$ lower order aggregate extent variables and the τ_k linking extent variables. This can be demonstrated very simply in matrix notation as follows.

As well as the detailed task discrete extent vector $\mathbf{n}_{kt}$ which is defined by (5.9) in section 5.3.2, we can also define the following two vectors of discrete extent variables for each ATP:

$$\hat{\mathbf{n}}_{kt} = \begin{bmatrix} \tilde{N}_{kt}^{(0)} \\ . \\ \tilde{N}_{kt}^{(m_k^{eff})} \\ N_{k,t+1-\tau_k} \\ . \\ N_{kt} \end{bmatrix} \tag{5.24}$$

$$\tilde{\mathbf{n}}_{kt} = \begin{bmatrix} \tilde{N}_{kt}^{(0)} \\ . \\ \tilde{N}_{kt}^{(m^+)} \end{bmatrix} \tag{5.25}$$

Note that the following argument also applies to the continuous task extent variables. It can be seen that $\hat{\mathbf{n}}_{kt}$ is made up of aggregate and linking variables, and, for a task k that has $m_k^{eff} \leq m^+ = h - \tau_k - 1$, it has h elements. For the same task k, $\tilde{\mathbf{n}}_{kt}$ is a vector containing only aggregate variables that has $m^+ + 1$ elements. The aggregate excess resource balance constraints must be written in terms of $\hat{\mathbf{n}}_{kt}$ and not all the aggregate variables in $\tilde{\mathbf{n}}_{kt}$ plus the linking variables as suggested by the recipe for constructing the m^{th} order formulation given in section 4.6 of the last chapter. For formulations with orders $m^+ > m^{\max}$, therefore, we can still use this recipe provided we substitute, for any task k with duration $\tau_k > h - 1 - m^+$, each aggregate task extent variables of order $p > h - \tau_k - 1$ by a linear combination of lower order aggregate extent variables and the linking extent variables.

We can relate $\mathbf{n}_{kt}$ and $\hat{\mathbf{n}}_{kt}$ as follows:

$$\begin{bmatrix} \tilde{N}_{kt}^{(0)} \\ . \\ \tilde{N}_{kt}^{(m_k^{eff})} \\ N_{k,t+1-\tau_k} \\ . \\ N_{kt} \end{bmatrix} = \begin{bmatrix} 1 & 1 & . & 1 & . & 1 \\ . & . & . & . & . & 1 \\ h^{m_k^{eff}} & (h-1)^{m_k^{eff}} & . & (\tau_k)^{m_k^{eff}} & . & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & . & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} N_{k,t-h+1} \\ N_{k,t-h+2} \\ . \\ N_{k,t+1-\tau_k} \\ . \\ N_{kt} \end{bmatrix}$$

or as:

$$\hat{\mathbf{n}}_{kt} = \hat{\mathbf{S}}_k^{(m_k^{eff},h)} \mathbf{n}_{kt} \tag{5.26}$$

where:

$$\hat{\mathbf{S}}_k^{(m_k^{eff},h)} = \begin{bmatrix} 1 & 1 & . & 1 & . & 1 \\ . & . & . & . & . & 1 \\ h^{m_k^{eff}} & (h-1)^{m_k^{eff}} & . & (\tau_k)^{m_k^{eff}} & . & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & . & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \tag{5.27}$$

We can also relate $\mathbf{n}_{kt}$ and $\tilde{\mathbf{n}}_{kt}$ as follows:

$$\begin{bmatrix} \tilde{N}_{kt}^{(0)} \\ . \\ \tilde{N}_{kt}^{(m^+)} \end{bmatrix} = \begin{bmatrix} 1 & 1 & . & 1 \\ . & . & . & 1 \\ h^{m^+} & (h-1)^{m^+} & . & 1 \end{bmatrix} \begin{bmatrix} N_{k,t-h+1} \\ N_{k,t-h+2} \\ . \\ N_{kt} \end{bmatrix}$$

which can also be written in matrix notation as:

$$\tilde{\mathbf{n}}_{kt} = \mathbf{S}^{(m^+,h)} \mathbf{n}_{\mathbf{kt}} \tag{5.28}$$

Note that $\mathbf{S}^{(m^+,h)}$ was defined earlier by (5.6) in section 5.3.1. From (5.26) and (5.28), therefore, we can relate $\tilde{\mathbf{n}}_{kt}$ and $\hat{\mathbf{n}}_{kt}$ as follows:

$$\tilde{\mathbf{n}}_{kt} = \mathbf{S}^{(m^+,h)} \quad \hat{\mathbf{S}}_k^{(m_k^{eff},h)^{-1}} \quad \hat{\mathbf{n}}_{kt} \tag{5.29}$$

In the aggregate excess resource balances, therefore, we can substitute the aggregate variables with order $p > m_k^{eff}$ using (5.26).

5.5 Property V: Covering Property for Capacity and Operational Constraints

It shall now be shown that the m^{th} order weighting function set (4.43) (used to generate the aggregate operational constraints) can also be used to derive all lower order weighting functions sets. A specific case of this was demonstrated in section 4.4.4 of the previous chapter. Consider the non-zero portion of a general weighting function from the set for a given task k:

$$g_{ky}^{(m)}(t-t'+1) \equiv \prod_{v=1}^{y}\Big[(t-t'+1)-\tau_k-v\Big]\prod_{w=1}^{m-y}\Big[h+1-(t-t'+1)-w\Big]$$

$$(t'=t-h+1,\ldots,t-\tau_k) \qquad (5.30)$$

Assuming that $y>0$, we can also write the corresponding weighting function for $y-1$:

$$g_{k,y-1}^{(m)}(t-t'+1) = \prod_{v=1}^{y-1}\Big[(t-t'+1)-\tau_k-v\Big]\prod_{w=1}^{m-y+1}\Big[h+1-(t-t'+1)-w\Big]$$

$$(t'=t-h+1,\ldots,t-\tau_k) \qquad (5.31)$$

Adding (5.30) to (5.31) and taking out the common factors results in:

$$g_{ky}^{(m)}(t-t'+1)+g_{k,y-1}^{(m)}(t-t'+1)=$$

$$\prod_{v=1}^{y-1}\Big[(t-t'+1)-\tau_k-v\Big]\prod_{w=1}^{m-y}\Big[h+1-(t-t'+1)-w\Big]\times$$

$$\Big\{\Big[(t-t'+1)-\tau_k-y\Big]+\Big[h+1-(t-t'+1)-(m-y+1)\Big]\Big\}$$

$$=\prod_{v=1}^{y-1}\Big[(t-t'+1)-\tau_k-v\Big]\prod_{w=1}^{m-y}\Big[h+1-(t-t'+1)-w\Big]\{h-m-\tau_k\}$$

Finally, recognising the form of the product term on the right hand side, and omitting the function arguments for the sake of simplicity, this can be written:

$$g_{ky}^{(m)} + g_{k,y-1}^{(m)} = (h-m-\tau_k)\, g_{k,y-1}^{(m-1)} \quad , \quad y=1,\ldots,m \qquad (5.32)$$

We note that $g_{k,y}^{(m-1)}$, $y=0,\ldots,m-1$ are the weighting functions used for constructing the aggregate constraints of order $m-1$. Therefore, provided that $h-m-\tau_k>0$, (5.32) shows that any aggregate inequality constraints of order $m-1$ are directly implied from the

corresponding constraints of order m and are therefore redundant.

By applying the above reasoning in a recursive manner, we deduce that only the highest order aggregate inequality constraints need to be included in the aggregate formulation.

As remarked in section 5.3.2, the resource capacity weighting function $f_y^{(m)}$ can be viewed as a special case of $g_{ky}^{(m)}$ with $\tau_k = 1$. Hence, an entirely analogous conclusion holds for the aggregate resource capacity constraints.

5.6 Illustrative Example

We shall now return to Example I presented in section 3.3 and solve it using the aggregate RTN formulation. Different solutions are generated by employing aggregate formulations of differing ATP lengths (h) and orders (m).

The results presented in table 5.1 demonstrate how the two aggregate parameters, ATP length (h) and order (m) can be used to vary the size and accuracy of the aggregate formulation. For comparison purposes, results for the fully detailed (exact) optimal solution, which was described in detail in section 3.3, are also given in the last column of table 5.1. As the ATP length is decreased or the aggregation order is increased, the number of integer variables increases and (for the higher values of h, at least) the objective function decreases to become closer to the detailed value.

Consider the production data in the first two columns of table 5.1 representing the amount of both products produced in the aggregate solutions with $h = 24$, $m = 1$ and $h = 24$, $m = 2$. In both cases, 1.0 tonne and 84.0 tonnes of the 1 kg and 2 kg products respectively are produced. These values could be arrived at by using a calculation based upon a naive consideration of the maximum production rates of the equipment involved in the problem. This calculation, which takes no account of the manpower or intermediate storage limitations or the timing of the demand, is presented below:

1. A change-over is required since packing line is initially tooled up to produce the 2 kg product and there is demand for the 1 kg product. Hence, total available packing time is the horizon length less the down-time due to the change-over (= 24 – 3 = 21 hours).

2. The demand for the 1 kg product is 20.0 tonnes, while the maximum packing rate for this product is 3.0 tonnes per hour. The packing line will therefore need to spend a total of 7 hours producing the 1 kg product. A total of $7 \times 3.0 = 21.0$ tonnes of this product will therefore be produced, leaving 1.0 tonne in storage at the end of the planning

Aggregation Parameters									
No. of ATPs	1		2		3		4		24
ATP length, h	24		12		8		6		1
Order of formulation, m	1	2	1	2	1	2	1	2	-
Computational Statistics									
Optimal Objective (£)	26,900	26,900	24,500	23,300	24,020	23,300	23,300	23,300	20,100
Fully Relaxed Objective (£)	29,524	29,524	29,524	29,524	29,524	28,754	27,600	27,600	24,872
No. nodes examined	17	27	97	762	632	458	232	315	115
CPU secs. (SUN SPARC 10)	1.1	2.0	3.8	55.4	37.3	59.2	14.9	60.6	21.9
No. Integer Variables	23	29	46	58	69	87	92	116	144
Total No. Variables	66	88	132	176	198	264	264	352	528
No. Constraints	71	103	141	205	211	307	281	409	649
Selected Solution Data									
Final Storage of 2kg Product	84.0	84.0	72.0	66.0	70.0	66.0	66.0	66.0	50.0
Final Storage of 1kg Product	1.0	1.0	1.0	1.0	4.0	1.0	1.0	1.0	1.0
Number of change-overs	1	1	1	2	1	2	2	2	2

Table 5.1: Data for aggregate solutions to Example I (no margin of optimality used).

horizon after the demand has been met.

3. The packing line then has $21 - 7 = 14$ hours left to produce the more profitable 2 kg product. Since the maximum packing rate is 6.0 tonnes per hour, a total of $14 \times 6.0 = 84.0$ tonnes of the 2 kg product can be produced by the end of the horizon.

Since, in the optimal *detailed* solution to this problem, only 50.0 tonnes of the 2 kg product can be made, a simplistic calculation such as that presented above results in a considerable over-estimate of the production capacity of the system (34.0 tonnes of the 2 kg product). This

is a problem that the aggregate solutions with shorter ATP lengths address to some extent, reducing the over-estimate to 16.0 tonnes in some cases (see table 5.1). This reduction of 18.0 tonnes in the production over-estimation is consistent with the inclusion of an extra change-over. This results in an additional 3 hours down-time for the packing line.

Figure 5.2 compares some of the output values from one of the aggregate solutions (ATP length $h = 8$, and order $m = 2$) indicated in table 5.1 with the corresponding values from the detailed solution to the same problem, presented in section 3.2.

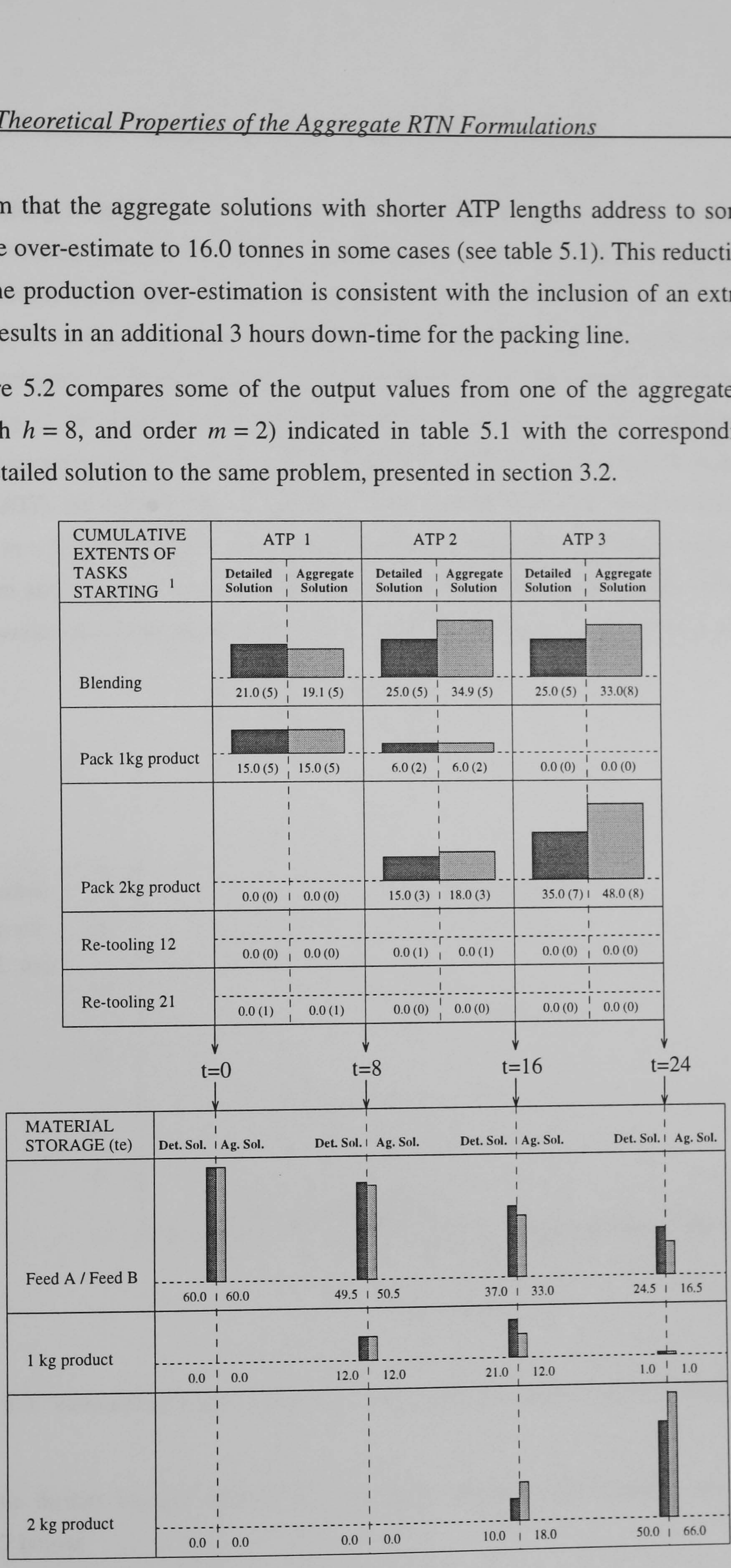


Figure 5.2: Comparison of detailed and aggregate ($h = 8$, $m = 2$) solutions to Example I.

1 For each task occurrence, figure 5.2 gives the continuous extent followed by the discrete extent in brackets.

Returning to table 5.1, it can be seen that the production capacity estimated by the aggregate solution with $h = 12$ and $m = 2$ is the same as that estimated by the $h = 6$, $m = 2$ solution. Hence, the doubling in formulation size from 58 to 116 integer variables appears not to affect the accuracy of the approximation in this case. In fact, at first sight, it might appear that an 'asymptote' of 66.0 tonnes of the 2 kg product exists. This would represent a limit on the accuracy of the aggregate formulation for this example and would, therefore, contradict the properties proved in sections 5.2 and 5.3. However, this is not the case as can be verified if smaller ATPs are used. For $h = 4$ and $m = 2$, the optimal objective value is £22,630 and for $h = 3$ and $m = 2$, the aggregate formulation becomes completely equivalent with the detailed formulation and therefore has the same objective of £20,100. For the second order formulations, the variation of optimal objective value with the number of ATPs used is shown in figure 5.3.

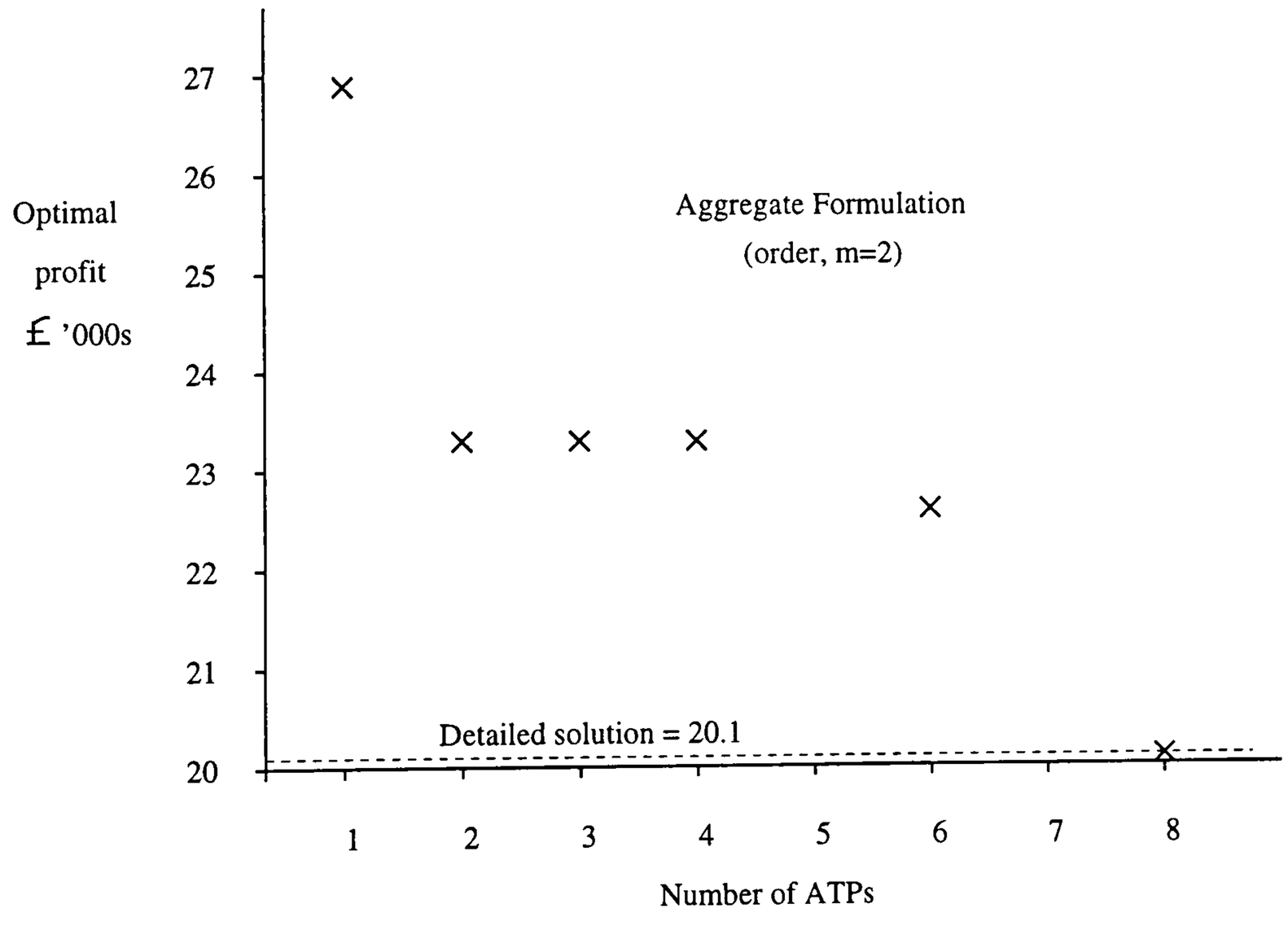


Figure 5.3: Variation of optimal objective value with the number of ATPs for Example I.

Some further ways of making the aggregate solution more accurate are discussed in section 5.7 below.

5.7 Improving the Accuracy of the Aggregate Formulation

5.7.1 Cut Constraints

It may often be possible to introduce extra constraints into the aggregate formulation which improve its accuracy in relation to the detailed formulation. Such constraints can often be derived for a specific problem using 'engineering intuition'. It must be ensured that they are redundant with respect to the detailed constraints even though they become active in the aggregate formulation and help tighten it.

One important feature that the aggregate formulation often misses can be termed the 'start-up time' for the production system. For a multi-stage process, with sufficiently low initial levels of intermediates, there will be a time delay between the start of the time horizon and the earliest time at which products can be produced. If there are a large number of serial processing stages, then such start-up times can be quite long. Some start-up constraints for the RTN formulation are derived in Appendix E.

5.7.2 Detailed Portions in the Time Horizon

Another way of increasing accuracy is to use an aggregate formulation in which some critical portions of the time horizon are governed by the fully detailed constraints. The 'splicing in' of detailed portions into the planning horizon is made possible by the presence of the linking variables. The resulting formulation will still provide a rigorous upper bound on the true optimal solution. For example, we can include a detailed portion at the start of the time horizon to help reduce the start-up error discussed above. We can also reduce any 'shut-down' error by including a detailed portion at the end of the time horizon. Such a planning horizon is illustrated in figure 5.4.

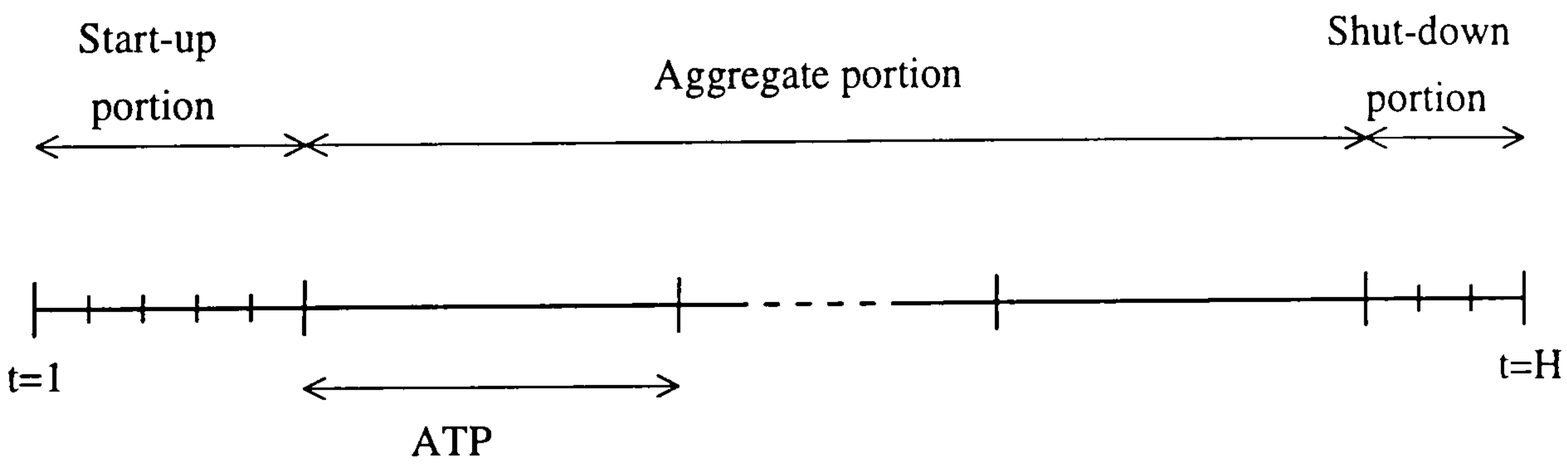


Figure 5.4: An aggregate planning horizon with detailed start-up and shut-down portions.

Table 5.2 gives the results for two different cases when this approach is applied to the small

making-packing example.

	Case 1	Case 2
Length of start-up portion	3 hrs	5 hrs
Length of shut-down portion	3 hrs	3 hrs
Length of aggregate portion	1×18 hrs	2×8 hrs
Optimal objective	£22,100	£20,400
Fully relaxed objective	£26,089	£25,985
Final storage of 1kg product (tonnes)	1.0	4.0
Final storage of 2kg product (tonnes)	60.0	50.0
Number of integer variables	59	106
CPU secs. (SUN SPARC 10)	3.1	30.6
Number of nodes examined	38	205

Table 5.2: Results for aggregate formulation with detailed constraints at start and end of horizon.

It can be seen that Case 1 gives a 10% over-estimate of the objective value while having only 41.0% of the total number of integer variables in the detailed formulation. Case 2 gives a 1.5% over-estimate at the expense of using 73.6% of the integer variables used by the detailed formulation. The fact that both these sets of results compare very favourably with those given in table 5.1 for uniformly aggregated horizons suggests that there may be some merit in this approach.

5.8 Concluding Remarks

This chapter has demonstrated some mathematical properties of the aggregate formulation. Perhaps the most important of these are the following:

- the aggregate formulation is a strict relaxation of the detailed formulation;
- the aggregate formulation reduces to the detailed formulation for $h \rightarrow 1$;
- the aggregate formulation reduces to the detailed formulation for $m \rightarrow h - 1$.

The strict relaxation property is significant in that it guarantees that the aggregate formulation derived from a feasible detailed problem will *never* be infeasible. It also ensures that

the aggregate formulation *always* provides an upper bound on the solution of the detailed one.

It should be noted that the two limiting aggregate formulations, with $h = 1$ and $m = h - 1$ respectively, have exactly the same number of variables as the detailed formulation and would consequently never be used in practice. However, the fact that an exact limit does exist provides some justification for using higher orders. Of course, it should be emphasised that the theoretical analysis in this chapter has not established any results regarding the *manner* in which accuracy increases as the order is increased or the ATP length is decreased. However, the example of section 5.6 does provide some practical evidence on this issue (*cf.* figure 5.3).

Previous aggregate process scheduling formulations (*e.g* Bassett *et al.*, 1994) have often failed to take account of important features such as limited intermediate storage. For our aggregate formulations, however, the fact that an exact limit exists means that such features *are* represented in some way by the higher order constraints. In this way, the trade-off between problem size reduction and accuracy of solution can be exploited.

Chapter 6

Discussion

In chapter 4 we presented a family of aggregate formulations based on the RTN concept and in chapter 5 we analysed their mathematical properties. In this chapter, we discuss a number of issues relating to these aggregate formulations. In particular, section 6.1 attempts to offer a physical interpretation to some of the mathematical definitions and manipulations used in earlier chapters. Section 6.2 considers the importance of the linking variables while section 6.3 is concerned with the size of the aggregate formulations as a function of their order and the length of the ATP. Section 6.4 presents a detailed comparison between our aggregate formulations and that recently proposed by Bassett *et al.* (1994). Finally, the use of aggregate formulations in practical planning and scheduling problems is discussed in section 6.5.

6.1 Physical Interpretation of Zeroth and First Order Aggregation

In chapter 4, the idea of a time-weighted summation of the detailed constraints of the RTN formulation was introduced. This somewhat abstract concept combining resources, tasks and time is a generalisation of some ideas that have a more obvious physical interpretation. These ideas can be related to the zeroth and first order aggregated excess resource balances.

The zeroth order aggregated excess resource balance relates the change in resource level across an ATP to the combined extents of all tasks that start within that ATP:

$$R_{rt} = R_{r,t-h} + \sum_k \tilde{\mu}^{(0)}_{kr0} \tilde{N}^{(0)}_{kt} + \sum_k \tilde{v}^{(0)}_{kr0} \tilde{\xi}^{(0)}_{kt} + \sum_k \sum_{\theta=1}^{\tau_k} \tilde{\mu}^{(0)}_{kr\theta} \left(N_{k,t-h+1-\theta} - N_{k,t+1-\theta} \right) +$$

$$\sum_k \sum_{\theta=1}^{\tau_k} \tilde{v}^{(0)}_{kr\theta} \left(\xi_{k,t-h+1-\theta} - \xi_{k,t+1-\theta} \right) + \tilde{\Pi}^{(0)}_{rt} \qquad \forall\ r,\ t = h, 2h, \ldots, H \qquad (4.14)$$

Consider a resource (*e.g.* a raw material) that is only *consumed* by the tasks under consideration in the RTN. In any detailed schedule, its excess level will, in general, decrease monotonically with time. As an example, consider a feed that can be consumed by only one reaction task. This is the only feed that is consumed by this reaction task and it is all consumed at the start of the task, *i.e.*:

$$\nu_{kr\theta} = \begin{cases} -1 & \theta = 0 \\ 0 & \theta = 1, \ldots, \tau_k \end{cases}$$

Suppose that, in a particular detailed schedule corresponding to the solution of a certain problem, this task starts three times during the period of time spanned by an ATP in an aggregated formulation of the same problem. There are also no demands or deliveries during this time. If there are initially 20 tonnes of feed at the start of the ATP, and if the tasks have continuous extents equal to 5.0, 3.0 and 4.0 tonnes respectively, the variation of the excess level of this resource will be as illustrated in figure 6.1.

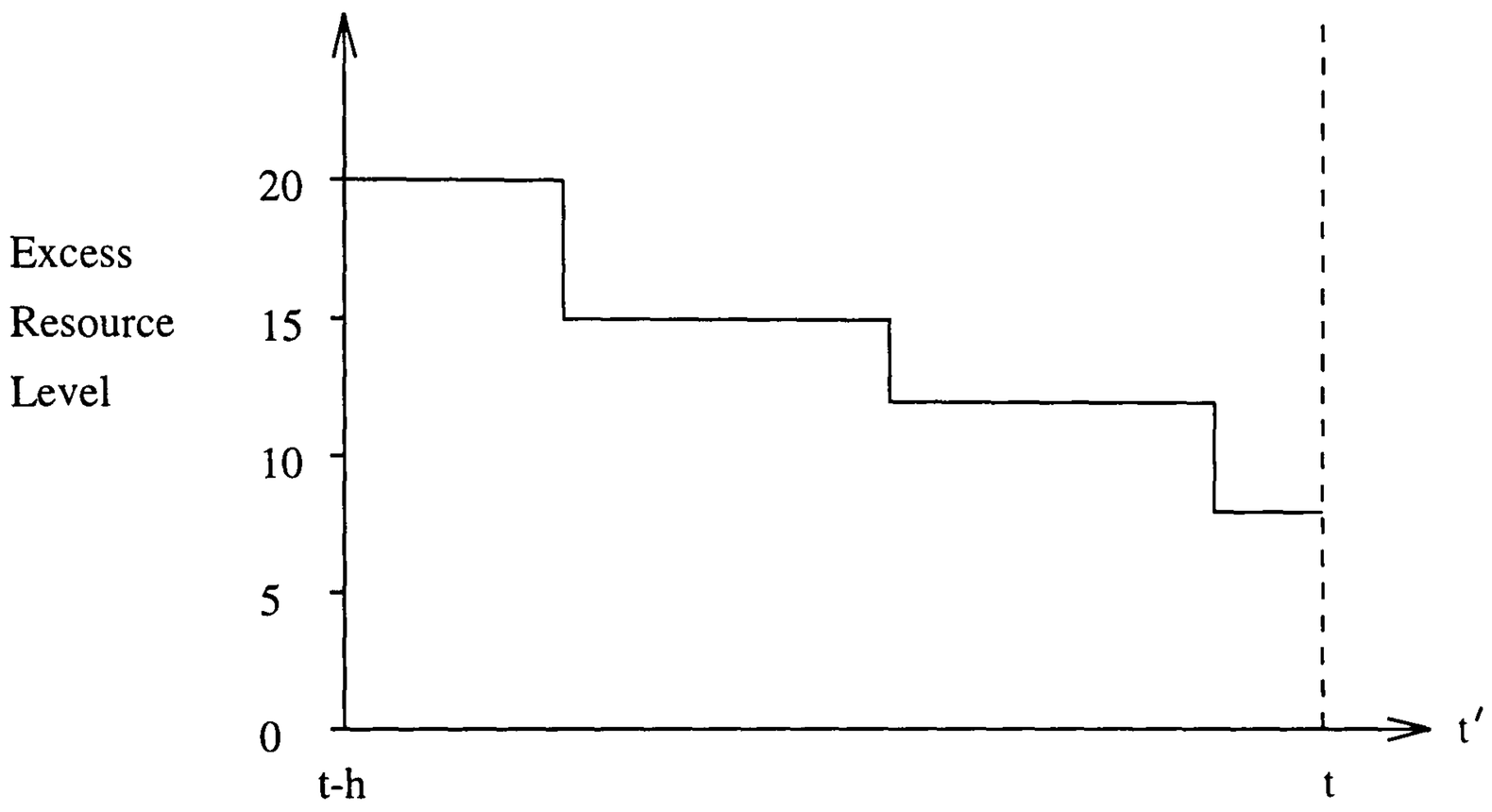


Figure 6.1: Variation of feed level for illustrative example.

In the detailed schedule, it is clear that the excess level of the feed at the end of the ATP is equal to $20.0 - 5.0 - 3.0 - 4.0 = 8.0$ tonnes. This is reflected in the zeroth order aggregate excess resource balance for the feed which (with the continuous extent coefficients $\nu_{kr\theta}$ given above and the discrete extent coefficients $\mu_{kr\theta}$ all being zero) can be written simply as:

$$R_{rt} = R_{r,t-h} - \xi_{kt}^{(0)} \tag{6.1}$$

The combined extent of all tasks starting within the ATP is:

$$\xi_{kt}^{(0)} = \sum_{t'=t-h+1}^{t} \xi_{kt'} = 5.0 + 3.0 + 4.0 = 12.0$$

Since $R_{r,t-h} = 20.0$, we therefore obtain $R_{rt} = 8.0$ from (6.1) as expected.

In a similar way, the zeroth order aggregate excess resource balance accurately represents the change in excess levels for resources that are only *produced* by tasks (*e.g.* final products). However, it becomes less accurate for resources that can be produced by some tasks but consumed by others (*e.g.* intermediate materials) and meaningless for resources that

are consumed and produced in equal quantities by the *same* task (*e.g.* equipment items).

To illustrate this point, consider a piece of equipment that is used only for a task k. For this task, the equipment resource has discrete coefficients $\mu_{kr\theta}$ as follows:

$$\mu_{kr\theta} = \begin{cases} -1 & \theta = 0 \\ 0 & \theta = 1, .., \tau_k - 1 \\ +1 & \theta = \tau_k \end{cases}$$

with all continuous task extent coefficients $\nu_{kr\theta}$ being equal to zero. Therefore:

$$\tilde{\mu}_{kr0}^{(0)} = \sum_{\theta=0}^{\tau_k} \mu_{kr\theta} = 0$$

Since $\tilde{\mu}_{kr0}$ is the multiplier of the variable $\tilde{N}_{kt}^{(0)}$ in the zeroth order aggregate excess resource balance for this resource, the aggregate formulation effectively does not impose any constraint on $\tilde{N}_{kt}^{(0)}$. This means that, as far as this resource is concerned, the task k can start an infinite number of times. This is obviously incorrect — each occurrence of the task has a finite duration which must place a limit on the number of times that it can start within an ATP.

In this case, we must consider the concept of *resource-time* and this leads to the physical interpretation of the *first* order aggregate excess resource balance. Consider figure 6.2 which relates the first order weighting function to the area under a plot of excess resource level *vs.* time for a detailed schedule. Suppose that the two increases in resource level shown in figure 6.2 are due to a certain task that is producing it. The change in resource level for the earlier task occurrence persists longer than that for the later one. Hence it gives rise to a proportionately larger effect on the overall resource-time for that resource in that ATP. This is reflected in the first order weighting function which is linear in time and therefore weights tasks proportionately to their starting time within each ATP.

Consider the first order aggregate excess resource balance:

$$\tilde{R}_{rt}^{(0)} - hR_{r,t-h} = \sum_k \tilde{\mu}_{kr0}^{(0)} \tilde{N}_{kt}^{(1)} - \sum_k \tilde{\mu}_{kr0}^{(1)} \tilde{N}_{kt}^{(0)} + \sum_k \tilde{\nu}_{kr0}^{(0)} \tilde{\xi}_{kt}^{(1)} - \sum_k \tilde{\nu}_{kr0}^{(1)} \tilde{\xi}_{kt}^{(0)} +$$

$$\sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (h + \theta' - \theta)\, \mu_{kr\theta} - \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (\theta' - \theta)\, \mu_{kr\theta} +$$

$$\sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (h + \theta' - \theta)\, \nu_{kr\theta} - \sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (\theta' - \theta)\, \nu_{kr\theta} + \tilde{\Pi}_{rt}^{(1)} \quad (4.17)$$

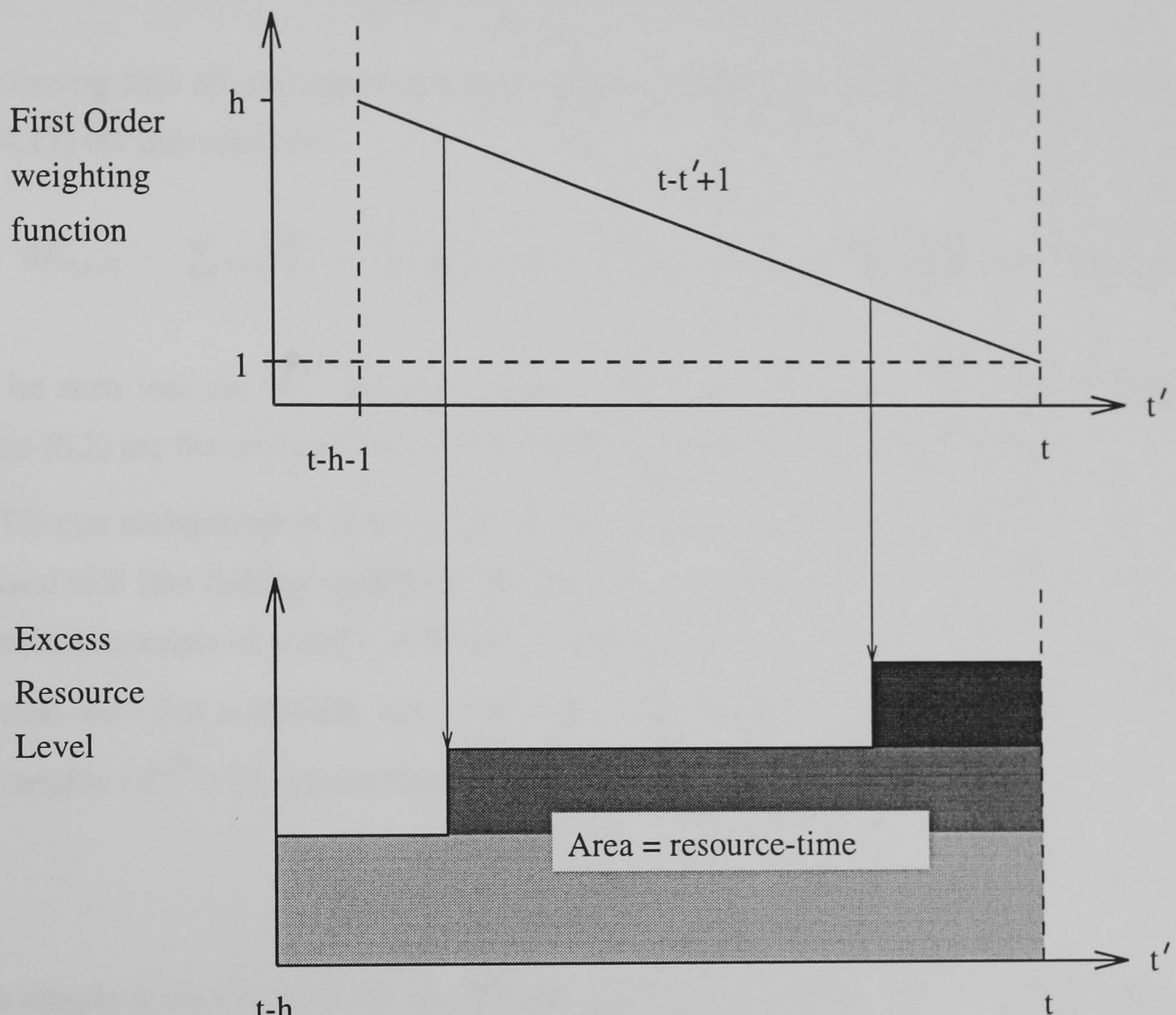


Figure 6.2: The first order weighting function and the concept of resource-time.

Consider again the equipment resource mentioned above for which the zeroth order aggregate excess resource balance alone proved insufficient. As before:

$$\tilde{\mu}_{kr0}^{(0)} \equiv \sum_{\theta=0}^{\tau_k} \mu_{kr\theta} = 0$$

but we now have another aggregate parameter:

$$\tilde{\mu}_{kr0}^{(1)} \equiv \sum_{\theta=0}^{\tau_k} \theta\, \mu_{kr\theta} = \tau_k$$

For this example, we can also simplify the coefficients on the linking variables:

$$\sum_{\theta=\theta'}^{\tau_k} (h + \theta' - \theta)\ \mu_{kr\theta} = (h + \theta' - \tau_k)$$

$$\sum_{\theta=\theta'}^{\tau_k} (\theta' - \theta)\ \mu_{kr\theta} \;=\; (\theta' - \tau_k)$$

Remembering that all the continuous task extent coefficients $\nu_{kr\theta}$ are zero, we can now re-write (4.17) for this resource:

$$\tilde{R}_{rt}^{(0)} \;=\; hR_{r,t-h} \;-\; \sum_k \tau_k \tilde{N}_{kt}^{(0)} \;+\; \sum_k \sum_{\theta'=1}^{\tau_k} (h + \theta' - \tau_k) N_{k,t-h+1-\theta'} \;+\; \sum_k \sum_{\theta'=1}^{\tau_k} (\tau_k - \theta') N_{k,t+1-\theta'} \qquad (6.2)$$

It can be seen that the $\tilde{N}_{kt}^{(0)}$ variable is now constrained because all the other variables in equation (6.2) are themselves bounded by capacity and operational constraints.

We can understand (6.2) better if we simplify it by ignoring the last two terms on the right hand side (the linking variables). We shall also consider the case where the whole planning horizon consists of a single ATP ($t = h = H$) and where the resource involved is a single equipment item that is initially idle at the start of the horizon ($R_{r0} = 1$). Regarding $\tilde{R}_{rt}^{(0)}$ as a slack variable ($\tilde{R}_{rt}^{(0)} \geq 0$), we can then write (6.2) in inequality form as:

$$\sum_k \tau_k \tilde{N}_{kt}^{(0)} \;\leq\; H$$

This is simply a statement of the fact that the ATP length and the task duration place a limit on the number of task instances that can start within it.

Although higher order constraints are not readily amenable to the same sort of direct physical interpretation, they represent a mathematical extension of the principles considered here. Their inclusion increases the accuracy of the aggregate formulation as was shown in chapter 5.

6.2 Importance of the Linking Variables

One interesting feature of the aggregate formulation presented in this thesis is the inclusion of *some* of the variables from the detailed formulation. At first sight, it may be thought that these linking variables, although they arise naturally in the derivation of the aggregated formulation, are an unnecessary detail. They are likely to account for the majority of the variables in the aggregate formulation and yet only model the 'end effects' for each ATP. Such effects may be very small for tasks with short durations. Many scheduling problems, however, involve tasks with a considerable spread of time-scales, with the longer tasks having durations that are a significant fraction of the ATP length. In this case, the linking variables assume greater importance since they allow tasks to span the boundaries between ATPs. This

will be illustrated below.

Consider again the zeroth order aggregated excess resource balance:

$$R_{rt} = R_{r,t-h} + \sum_k \tilde{\mu}^{(0)}_{kr0} \tilde{N}^{(0)}_{kt} + \sum_k \tilde{\nu}^{(0)}_{kr0} \tilde{\xi}^{(0)}_{kt} + \sum_k \sum_{\theta=1}^{\tau_k} \tilde{\mu}^{(0)}_{kr\theta} \left(N_{k,t-h+1-\theta} - N_{k,t+1-\theta} \right) +$$

$$\sum_k \sum_{\theta=1}^{\tau_k} \tilde{\nu}^{(0)}_{kr\theta} \left(\xi_{k,t-h+1-\theta} - \xi_{k,t+1-\theta} \right) + \tilde{\Pi}^{(0)}_{rt} \qquad \forall\ r, t = h, \ldots, H \qquad (4.14)$$

Consider a material resource r that is produced at the end of a task k and only by that task. The task extent coefficients are as follows:

$$\mu_{kr\theta} = 0 \qquad \theta = 0, \ldots, \tau_k$$

$$\nu_{kr\theta} = \begin{cases} 0 & \theta = 0, \ldots, \tau_k - 1 \\ +1 & \theta = \tau_k \end{cases}$$

The discrete extent aggregate coefficients are therefore all zero and the continuous extent aggregate coefficients are as follows:

$$\tilde{\nu}^{(0)}_{kr\theta} = \sum_{\theta'=\theta}^{\tau_k} \nu_{kr\theta'} = 1 \qquad \theta = 0, \ldots, \tau_k$$

Assuming no external deliveries or demands ($\tilde{\Pi}^{(0)}_{rt} = 0$), (4.14) can be simplified to:

$$R_{rt} = R_{r,t-h} + \tilde{\xi}^{(0)}_{kt} + \sum_{\theta=1}^{\tau_k} \left(\xi_{k,t-h+1-\theta} - \xi_{k,t+1-\theta} \right) \qquad \forall\ r, t = h, \ldots, H \qquad (6.3)$$

Suppose that an aggregate formulation of ATP length $h = 24$ hours is being used, based on a detailed discretisation interval of 1 hour, and that the task has a duration of 8 hours. Figure 6.3 illustrates the function served by the linking variables for this case.

Remembering that the aggregate variable $\tilde{\xi}^{(0)}_{kt}$ represents the combined continuous extent of a task k *starting* within the ATP, the linking variables account for the fact that not all tasks that start within the ATP will actually finish within it: some will carry over into the next one. These will therefore not contribute to the increase in resource level across the ATP under consideration. This is accounted for by subtracting the linking variables $\xi_{k,t+1-\theta}$, $\theta = 1, \ldots, \tau_k$ in (6.3). Similarly, there are a number of possible task occurrences initiated in the previous ATP that finish within this ATP and therefore these represent an extra contribution to the increase in resource level across this ATP. This is accounted for by adding

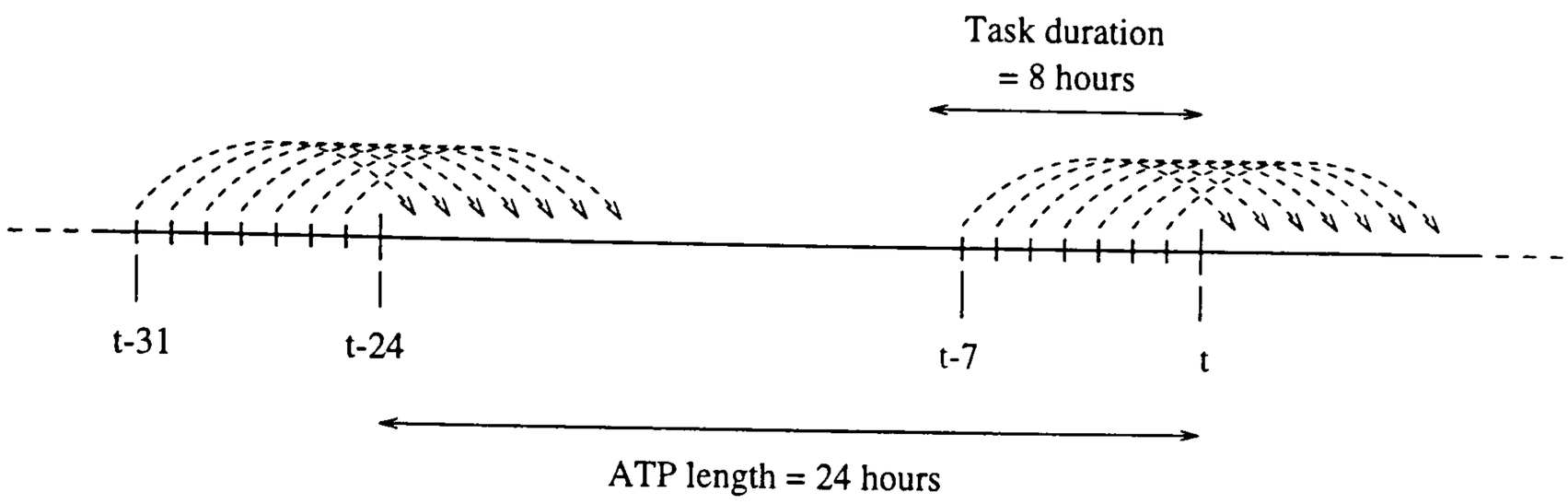


Figure 6.3: Illustration of the function served by linking variables.

the linking variables $\xi_{k,t-h+1-\theta}$, $\theta = 1, .., \tau_k$ in (6.3) above.

The linking variables which result from aggregation of the excess resource balance can also be used to help tighten the capacity and operational constraints as was demonstrated in section 4.4.

In a mathematical sense, linking variables also ensure consistency, *i.e.* that the aggregated formulation tends to the detailed formulation as $h \rightarrow 1$. This property was demonstrated in section 5.2. It is interesting to note that, in general, the accuracy of aggregated formulations that do *not* have this property will not improve as h is decreased: in fact, it will start to deteriorate as h becomes comparable to the length of some of the tasks in the process.

The fact that longer tasks give rise to more linking variables means that they may increase the number of integer variables (and hence the solution time) to unacceptable levels. Multisite scheduling problems will often involve tasks that operate on widely different time-scales. A transportation task, for example, may take 24 hours or more whereas a blending task may be only 30 minutes in duration. Applying aggregation to a detailed formulation using a 30 minute discretisation interval would introduce 48 linking variables for each transportation task. This can be prevented if the longer tasks are constrained to start on fewer occasions (every 4 hours, say) within the ATP without significantly compromising the optimality of the resulting solution. This can be done quite simply by forcing to zero all linking variables not corresponding to the allowed times, thereby removing them from the formulation. In effect, we are using a different discretisation level d_k for each task k depending on the minimum time-scale on which it affects the levels of the common resources. As shown in Appendix B, aggregated RTN formulations employing multiple time discretisation scales can be derived in a very similar manner to the uniformly discretised formulations of the previous chapter.

6.3 The Effect of ATP Length and Overall Order on Problem Size

Since the computational effort required for solving MILP problems of a given structure is usually related to the number of integer variables, we shall use this as a guide to the reduction in problem size that can be achieved by using the aggregate formulation. Only the single time-scale formulation will be considered here although using multiple time-scales will result in even fewer integer variables.

The number of integer variables for a given problem depends on the number of ATPs into which the total planning horizon is divided. Consider a RTN that has a total of q different tasks. We can define an *average task duration* as follows:

$$\overline{\tau} \equiv \frac{\sum_{k=1}^{q} \tau_k}{q} \tag{6.4}$$

A general m^{th} order aggregated formulation will have $(m+1)q$ aggregated integer variables and $q\overline{\tau}$ linking integer variables per ATP. So, for an ATP length of h time intervals we can write the following:

$$\frac{no.\ integer\ variables\ in\ aggregated\ formulation}{no.\ integer\ variables\ in\ detailed\ formulation} = \frac{\overline{\tau} + m + 1}{h} \tag{6.5}$$

Note that it is the *average* task duration that is important in terms of the total number of integer variables. An RTN may have a few tasks that have durations that are a significant fraction of the ATP length. If, however, it has many more short tasks, then the average value will still be low.

Increasing the value of h reduces the problem size but results in the aggregated formulation becoming less constrained and therefore causes it to over-estimate the objective function by a larger amount. Conversely, increasing m will increase the accuracy of the aggregated formulation at the expense of increased problem size. In many cases it has been found that the optimal objective function is a weak function of h. This suggests that h can be made quite large in comparison to the total length of the planning horizon H — for some problems, splitting the latter into only two ATPs or even treating it as a single ATP still produces satisfactory results. We note that ATP boundaries do not have to be constrained to coincide with discrete product demands: as already explained, the aggregated formulation naturally takes some account of the *timing* of demands as well as their magnitude through its use of the aggregated demand parameters.

Overall these considerations provide some encouragement that the approach described in this thesis will provide a means of solving previously intractable problems.

6.4 Comparison with the Aggregate Formulation of Bassett *et al.* (1994)

The subject of aggregate formulations for process scheduling problems has also recently been addressed by Bassett *et al.* (1994) whose aggregate formulation is summarised below†:

$$\sum_{i \in I_j^{tasks}} p_{ij} y_{ij\tau} \leq H_\tau \qquad \forall\ j, \tau \tag{B4.1}$$

$$A_{s\tau} = A_{s,\tau-1} + \sum_{i \in I_{sp}^{tasks}} \sum_{j \in I_i^{equip}} f_{si}^{p} B_{ij\tau} - \sum_{i \in I_{sc}^{tasks}} \sum_{j \in I_i^{equip}} f_{si}^{c} B_{ij\tau} - q_{s\tau}^{s} + q_{s\tau}^{b} \qquad \forall\ s, \tau \tag{B4.2}$$

$$B_{ij\tau} \leq m_{ij} y_{ij\tau} \qquad \forall\ i, j \in I_i^{equip}, \tau \tag{B4.3}$$

$$q_{s\tau}^{b} \leq Q_{s\tau}^{b} \qquad \forall\ s, \tau \tag{B4.4}$$

$$A_{s\tau} \leq A_{s\tau}^{\max} \qquad \forall\ s, \tau \tag{B4.5}$$

$$\max\ NPV = \sum_{\tau} \sum_{s} v_{s\tau}^{s} q_{s\tau}^{s} - \sum_{\tau} \sum_{i} \sum_{j} C_{ij\tau}^{ov} B_{ij\tau} - \sum_{\tau} \sum_{i} \sum_{j} C_{ij\tau}^{of} y_{ij\tau} - \sum_{\tau} \sum_{s} v_{s\tau}^{b} q_{s\tau}^{b} \tag{B4.6}$$

In the Bassett *et al.* formulation, the τ subscripts denote 'planning periods' which are similar to the ATPs in the aggregate formulation presented in this thesis. However, unlike our formulation which considers all production resource (equipment items, material, utilities *etc.*) in a unified manner, their formulation considers only material and processing equipment resources. The two are treated in different ways and are denoted by the subscripts s and j respectively. In addition, the Bassett *et al.* formulation considers a task (denoted by subscript i) as being independent of the equipment item j in which it is being performed. Hence, task occurrences are represented by 3 subscripts — i, j and τ — which denote the task type, the equipment item that it is performed in, and the planning period in which it took place, respectively. Each task i consumes and produces a subset of the materials s in the process. Overall, this is consistent with the State-Task Network process representation (Kondili *et al.*, 1993).

---

† The notation (B*x.y*) refers to constraint *(x.y)* in the paper by Bassett *et al.* (1994). Their original notation is used in this discussion.

We shall now consider each of the constraints (B4.1) to (B4.5) and the objective function (B4.6) in turn. Constraint (B4.1) is a horizon constraint placing a limitation on the integer variable $y_{ij\tau}$ — the number of batches of task i processed by unit j in period τ. Here, p_{ij} is the processing time of task i when performed in unit j and H_τ is the number of time discretisation intervals in period τ. Constraint (B4.2) is the material balance constraint which relates the change in each material stored over each period to the batch-sizes of the tasks starting within it. The variable $A_{s\tau}$ is the amount of material s in storage at the end of period τ; $B_{ij\tau}$ is the cumulative batch-size for task i carried out on unit j over period τ; and $q^b_{s\tau}$ and $q^s_{s\tau}$ are respectively the quantities of material s bought and sold externally in period τ. For each task i, the parameters f^p_{si} and f^c_{si} are the fractions of material s that it respectively produces and consumes.

Constraint (B4.3) is an operational constraint which places an upper bound on the cumulative batch-size for task i carried out on unit j over period τ. The parameter m_{ij} is the maximum capacity of unit j when performing task i. Constraints (B4.4) and (B4.5) impose upper bounds on the amount of material s that can be acquired over period τ and the amount of material s in storage at the end of period τ, respectively. Finally, the objective function represents the maximisation of nett present value over the planning horizon of interest. The material s sold or bought in period τ has a unit selling value $v^s_{s\tau}$ and a unit cost of purchase $v^b_{s\tau}$. There are also fixed and variable costs associated with performing the tasks; these are denoted by $C^{of}_{ij\tau}$ and $C^{ov}_{ij\tau}$ respectively.

The Bassett *et al.* formulation outlined above is, to a certain extent, comparable to the first order formulation presented in this thesis. One notable difference is that Bassett *et al.* (1994) do not include any linking variables in their formulation. We now proceed to compare the two formulations in detail.

6.4.1 Treatment of Material Resources

Consider the zeroth order aggregated excess resource balance (4.14) which was derived in chapter 4. Typically, the amounts of a material resource r consumed and/or produced by a task k will be directly proportional to the continuous extent ξ_{kt} (*i.e.* the batch-size) of this task. Consequently, the resource utilisation coefficients $\mu_{kr\theta}$ for the *discrete* extent of the task will all be zero. Hence, in this case, we can simplify (4.14) to:

$$R_{rt} = R_{r,t-h} + \sum_k \tilde{\nu}_{kr0}^{(0)} \tilde{\xi}_{kt}^{(0)} + \sum_k \sum_{\theta=1}^{\tau_k} \tilde{\nu}_{kr\theta}^{(0)} \left(\xi_{k,t-h+1-\theta} - \xi_{k,t+1-\theta} \right) + \tilde{\Pi}_{rt}^{(0)}$$

$$\forall \; r \in materials, \; t, \; h \qquad (6.6)$$

The material balance (B4.2) in the planning formulation of Bassett *et al.* (1994) is just an approximation of (6.6) obtained by ignoring the linking variables and consequently dropping the third (double summation) term on the right hand side:

$$R_{rt} = R_{r,t-h} + \sum_k \tilde{\nu}_{kr0}^{(0)} \tilde{\xi}_{kt}^{(0)} + \tilde{\Pi}_{rt}^{(0)} \qquad \forall \; r \in materials, \; t, \; h \qquad (6.6')$$

To see the equivalence of (B4.2) and (6.6′), we identify:

$$R_{rt} \equiv A_{s\tau} \; ; \qquad (6.7a)$$

since these quantities have exactly the same meaning, *i.e.* the amount of excess resource r (≡ material s in inventory) at the end of discretisation interval t (≡ planning period τ). Similarly,

$$R_{r,t-h} \equiv A_{s,\tau-1} \; ; \qquad (6.7b)$$

since each ATP (≡ planning period) covers h discretisation intervals.

We also note that a task k in the RTN formulation corresponds to a specific combination of task i and equipment item j in the Bassett *et al.* formulation. Therefore the single summation on the right hand side of (6.6′) covers *all* tasks under consideration, in the same way as the two double summations on the right hand side of (B4.2) which can be combined as:

$$+ \sum_{i \in I_{sp}^{tasks} \cup I_{sc}^{tasks}} \sum_{j \in I_i^{equip}} \left(f_{si}^p - f_{si}^c \right) B_{ij\tau}$$

Now, by definition (*cf.* (4.11b)), $\tilde{\xi}_{kt}^{(0)}$ is the cumulative batch-size of task k during the ATP ending at t, and is therefore equivalent to $B_{ij\tau}$. Also, from (4.12b), $\tilde{\nu}_{kr0}^{(0)}$ is the coefficient of proportionality between the amount of material resource r produced by task k and the cumulative batch-size $\tilde{\xi}_{kt}^{(0)}$ of this task. Therefore:

$$\tilde{\nu}_{kr0}^{(0)} \equiv f_{si}^p - f_{si}^c$$

Consequently, we can identify:

$$\sum_k \tilde{\nu}^{(0)}_{kr0}\tilde{\xi}^{(0)}_{kt} \equiv \sum_{i\in I^{tasks}_{sp}} \sum_{j\in I^{equip}_i} f^p_{si} B_{ij\tau} - \sum_{i\in I^{tasks}_{sc}} \sum_{j\in I^{equip}_i} f^c_{si} B_{ij\tau} \qquad (6.^{-}c)$$

Finally, by definition (*cf.* equation (4.13)), $\tilde{\Pi}^{(0)}_{kt}$ is the net amount of resource r received from external sources over the ATP finishing at time t. We therefore identify:

$$\tilde{\Pi}^{(0)}_{kt} \equiv q^b_{s\tau} - q^s_{s\tau} \qquad (6.7d)$$

The overall effect of (6.7a-d) is that Bassett *et al.*'s constraint (B4.2) is entirely equivalent to the simplified constraint (6.6´).

6.4.2 Treatment of Processing Equipment Resources

Assuming that a piece of processing equipment is required for the entire duration of each processing task for which it is suitable, we have $\mu_{kr0} = -1$ and $\mu_{kr\tau_k} = +1$, with all other coefficients $\mu_{kr\theta}$ and $\nu_{kr\theta}$ being zero. Hence the corresponding aggregated coefficients (as defined by (4.12a) and (4.12b)) are as follows:

$$\tilde{\mu}^{(0)}_{kr0} = 0 \;\; ; \quad \tilde{\mu}^{(0)}_{kr\theta} = 1,\ \theta > 0 \;\; ; \quad \tilde{\nu}^{(0)}_{kr\theta} = 0, \forall\, \theta$$

Under these conditions and assuming that equipment availability is constant over the planning horizon, we can write (4.14) as:

$$R_{rt} = R_{r,t-h} + \sum_k \sum_{\theta=1}^{\tau_k} \left(N_{k,t-h+1-\theta} - N_{k,t+1-\theta} \right) \qquad \forall\ r \in equipment,\ t,\ h \qquad (6.8)$$

We now let $h = t$ and, assuming that tasks can only start at $t \geq 1$, we get:

$$R_{rt} + \sum_k \sum_{\theta=1}^{\tau_k} N_{k,t+1-\theta} = R_{r0} \qquad \forall\ r \in equipment,\ t \qquad (6.9)$$

Now consider the first order aggregated excess resource balance (4.17) from chapter 4. By collecting terms that are pre-multiplied by h, we can rewrite this as:

$$\tilde{R}^{(0)}_{rt} - h\left(R_{r,t-h} + \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} \mu_{kr\theta} \right) = \sum_k \tilde{\mu}^{(0)}_{kr0} \tilde{N}^{(1)}_{kt} - \sum_k \left(\sum_{\theta=1}^{\tau_k} \theta\, \mu_{kr\theta} \right) \tilde{N}^{(0)}_{kt} +$$

$$\sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (\theta' - \theta)\, \mu_{kr\theta} - \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (\theta' - \theta)\, \mu_{kr\theta}$$

The above can be simplified further by noting that, for equipment resources, $\sum_{\theta=1}^{\tau_k} \theta\ \mu_{kr\theta} = \tau_k$ and $\sum_{\theta=\theta'}^{\tau_k} (\theta' - \theta)\ \mu_{kr\theta} = (\theta' - \tau_k) \quad \forall\ \theta' > 0$. This leads to:

$$\tilde{R}_{rt}^{(0)} - h\left(R_{r,t-h} + \sum_k \sum_{\theta=1}^{\tau_k} N_{k,t-h+1-\theta}\right) = -\sum_k \tau_k\ \tilde{N}_{kt}^{(0)} +$$

$$\sum_k \sum_{\theta=1}^{\tau_k} N_{k,t-h+1-\theta}(\theta - \tau_k) - \sum_k \sum_{\theta=1}^{\tau_k} N_{k,t+1-\theta}(\theta - \tau_k)$$

which, by virtue of (6.9), is equivalent to:

$$\sum_k \tau_k\ \tilde{N}_{kt}^{(0)} + \sum_k \sum_{\theta=1}^{\tau_k - 1} (\tau_k - \theta)\left(N_{k,t-h+1-\theta} - N_{k,t+1-\theta}\right) = hR_{r0} - \tilde{R}_{rt}^{(0)} \tag{6.10}$$

We note that $\tilde{R}_{rt}^{(0)}$ is a non-negative variable which, by its definition (*cf.* equation (4.19)), can be interpreted as the total idle time for equipment of type r over the ATP finishing at time t. Also, R_{r0} is the number of available equipment items of type r. In view of the non-negativity of $\tilde{R}_{rt}^{(0)}$, we can write (6.10) as:

$$\sum_k \tau_k\ \tilde{N}_{kt}^{(0)} + \sum_k \sum_{\theta=1}^{\tau_k - 1} (\tau_k - \theta)\left(N_{k,t-h+1-\theta} - N_{k,t+1-\theta}\right) \leq hR_{r0} \tag{6.11}$$

Now, *if* we ignore the linking variables *and* assume that $R_{r0} = 1$ (*i.e.* that each equipment item forms a distinct resource type), (6.11) simplifies to:

$$\sum_k \tau_k\ \tilde{N}_{kt}^{(0)} \leq h \tag{6.12}$$

We note that, by definition (*cf.* (4.11a)), $\tilde{N}_{kt}^{(0)}$ is the number of batches of task k undertaken† during the ATP finishing at time t and is therefore equivalent to variable $y_{ij\tau}$ in Bassett *et al.*'s formulation. Also, we identify $h \equiv H_\tau$ and $\tau_k \equiv p_{ij}$. Hence, (6.12) is, in fact, identical to (B4.1).

---

† The word 'undertaken' is used somewhat loosely here. To be more precise, $\tilde{N}_{kt}^{(0)}$ is the number of batches of task k *starting* in the ATP finishing at time t. This is not necessarily the same as the number of batches that are *completed* in this ATP (see discussion of section 6.2). Bassett *et al.* define the variable $y_{ij\tau}$ as 'the number of batches of task i on unit j at τ' which, along with the absence of any linking variables in their formulation, highlights the fact that they do not make the distinction that has just been discussed.

6.4.3 Remarks

We can summarise the conclusions of section (6.4.1) and (6.4.3) as follows:

- Bassett *et al.*'s (1994) material balance constraint (B4.2) is an approximation to our zeroth order aggregate resource balance constraint (4.14) formed by omitting the linking variables.
- Bassett *et al.*'s (1994) equipment allocation constraint (B4.1) is an approximation to our zeroth and first order aggregate resource balance constraints (4.14) and (4.17) formed by omitting the linking variables.
- Constraint (B4.1) treats each processing equipment item as a distinct resource. This corresponds to $R_{r0} = 1$ in our formulations. In contrast, the latter treat all identical equipment items as a *single* resource. This will be more efficient in problems involving large numbers of such identical items ($R_{r0} \gg 1$), which is often the case in practice. For example, in the industrial case study presented in the next chapter, the intermediate storage tasks at each plant are performed by sets of 5 or more identical silos. Each of these silo sets are considered as the same resource in the RTN formulation of this problem. Since the RTN involves a different storage task for each of the 18 intermediates in the process, significantly more integer variables would be required if each silo were treated as a different resource.

The importance of including linking variables in aggregate formulations has already been examined in section 6.2. However, another important difference between the two formulations is that the first order aggregate formulation presented in this thesis uses *both* the zeroth order aggregated excess resource balance *and* the first order aggregated excess resource balance for *all* resource types — including materials.

The inclusion of the first order constraint for materials is important for a variety of reasons. First, it introduces the first order aggregated demand parameter $\tilde{\Pi}_{rt}^{(1)}$ in the formulation, which allows the latter to take account of the *timing* as well as the magnitude of demands within each ATP. Consider a single point demand of fixed size for a resource r occurring within an ATP that ends at time t. The zeroth order aggregated demand parameter $\tilde{\Pi}_{rt}^{(0)}$ has the same value irrespective of when the demand occurs within the ATP. The first order aggregated demand parameter, on the other hand, will have a value of $h\tilde{\Pi}_{rt}^{(0)}$ if the demand occurs at the start of the ATP ($t' = t - h + 1$), decreasing to $\tilde{\Pi}_{rt}^{(0)}$ if it occurs at the end ($t' = t$). This allows relatively long ATPs to be used without losing all information as to the precise timing of demands occurring within them.

A second advantage of including the first order aggregated excess resource balance for materials is that it introduces the $\tilde{R}_{rt}^{(0)}$ variables in the formulation. This allows the aggregated objective function to take some account of material inventory costs which may be significant, especially for long-term planning problems. The $\tilde{R}_{rt}^{(0)}$ variables are also useful for problems that are constrained by limited intermediate storage. The first order aggregated excess resource capacity constraints (4.27a) and (4.27b) constrain not only the resource levels at the start and end of each ATP (as done by constraint (B4.5)), but also the time integral of the resource profile over the entire ATP.

Finally, it should be emphasised that the formulation of Bassett *et al.* (1994) considers only material and equipment item resources. The aggregate RTN formulation, however, with its general definitions of resources and tasks, allows other resources, such as utilities and transportation devices to be considered. It also allows complicating features such as sequence dependent cleaning to be taken into account in a direct and general fashion.

6.5 Applications of the Aggregate Formulation

It will be seen in the industrial case study presented in the next chapter that the aggregate formulation can give a tight upper bound on the production capacity of a flexible plant and its ability to meet demands imposed upon it. Thus the aggregated model can, in itself, be very useful as a decision support tool for assessing the production capability of the system or identifying resource bottle-necks without having to solve the detailed scheduling problem.

One obvious application of the aggregate formulation is for performing rapid feasibility checks in problems driven by customer demands as a prelude to attempting the solution of detailed scheduling problems. If the aggregate problem is found to be infeasible, then so must be the detailed problem since the former represents a relaxation of the latter.

Another possible use of the aggregate formulation may be in solving detailed scheduling problems that have prohibitively long solution times because of a large integrality gap. In such cases, the aggregate formulation may provide a much tighter upper bound on the optimal value of the objective function than that obtained by simply solving the fully relaxed detailed problem. Such an improved upper bound may significantly speed up the solution of the detailed formulation. For instance, it may allow the branch-and-bound search to be terminated early once an integer feasible solution with an objective function value sufficiently close to this upper bound is obtained.

For central planning in a multisite system, the aggregate formulation for the whole system can be solved to provide a set of 'production targets' for each ATP which take into

account interactions between all the sites. These will consist of amounts of material in storage at each site at the end of each ATP and amounts of material to be transported between sites in each ATP. These targets are then useful to the individual sites when they schedule their activities individually.

The ultimate aim of most scheduling procedures is to generate a fully detailed schedule *i.e.* one that involves a fully implementable sequence of operations that is relevant to the day-to-day operation of the multipurpose plant involved.

In general, there is no rigorous method of deriving an optimal, or even feasible, detailed solution from the corresponding aggregate solution, although specific problems may have particular characteristics which can be exploited to this end. Since the aggregate formulation will always tend to over-estimate production capacity and the ability to meet demands, it is important to build a degree of flexibility into the original problem formulation. Fortunately, this is often the case in practice anyway. For the industrial case study presented in the next chapter, for example, all the demands are regarded as being 'soft' (*i.e.* as being desirable to meet rather than being hard constraints that must be satisified). This conforms to the actual operating policy for the system and means that the problem of infeasibilities at the detailed scheduling stage does not arise. It is assumed that any demands not met in full in one week will be given higher priority in subsequent weeks.

When a fully detailed schedule is being sought, therefore, the aggregate solution should be used in a 'target setting' mode rather than to lay down hard constraints on the amount of each product or intermediate that should have been produced by a certain time in the horizon. Two approaches for generating detailed schedules in this way are outlined below.

6.5.1 A Decomposition Approach for Generating Detailed Schedules

One approach to generating detailed solutions from aggregate ones is independently to solve the ATP sub-problems that the aggregate formulation generates. The aggregate solution gives values of the aggregate and linking variables for each ATP and also the amount of material produced in each ATP. The problem, therefore, can be conveniently decomposed into a set of sub-problems, one for each ATP. Hence the detailed formulation could be solved for each ATP with the objective of minimising the difference between the actual production and the production 'target' indicated by the aggregate solution.

In some cases, this approach may be unsatisfactory since there is no scope for interaction between the ATP sub-problems. In particular, no use is made of the linking variables, which arise naturally in the derivation of the aggregate formulation and which represent these

interactions.

Clearly, if one ATP cannot meet its targets, then it may give rise to infeasibilities in subsequent ATPs in which demands cannot be met. This then dictates the use of some intuitive rules to shift production between ATPs. Alternatively, an iterative scheme could be developed in which the aggregate formulation is modified before being re-solved in the search for a feasible solution.

6.5.2 A Rolling Horizon Approach for Generating Detailed Schedules

An alternative approach, involves the repeated solution of the problem as many times as there are ATPs in the planning horizon. Initially, the problem is solved with full detail in the first ATP (*i.e.* the detailed constraints and variables) but with subsequent ATPs described by aggregated variables and constraints. As was demonstrated in section 5.7.2 of the previous chapter, the linking variables enable detailed portions to be included in the planning horizon in this way.

Once the optimal solution to the first problem is found, the detailed variables in the first ATP are fixed to these values. The problem is then solved again for the rest of the planning horizon. This time the original first ATP is not included in the formulation while the second ATP uses the detailed constraints and variables, and the remaining ATPs use aggregate descriptions as before. This procedure is repeated until the entire planning horizon has been scheduled in detail. This scheme is outlined in figure 6.4 for a week long scheduling problem containing ATPs of 24 hours.

It can be seen that when the first ATP is being scheduled in detail, information about the actions that are needed in the future is contained in the aggregated variables of the subsequent ATPs. This kind of approach may also be useful in systems where there is considerable uncertainty in demands occurring in the latter portion of the planning horizon. In this case, the detailed constraints can be used for short-term decisions with the longer-term (and hence usually more uncertain) information being represented on an aggregate basis.

As has been stated earlier, the ATP length can be made quite large while still keeping the aggregate solution as a tight upper bound to the detailed solution. Hence, for the scheme suggested above, the aggregated portion of the planning horizon need consist only of a few ATPs. Of course, it is still possible for infeasibilities to occur if there is significant overestimation of production capacity in the aggregated portion of the planning horizon. For this reason, a procedure that allows the detailed schedule to be modified as required may be necessary. For example, existing tasks could be increased in extent, or entirely new tasks could

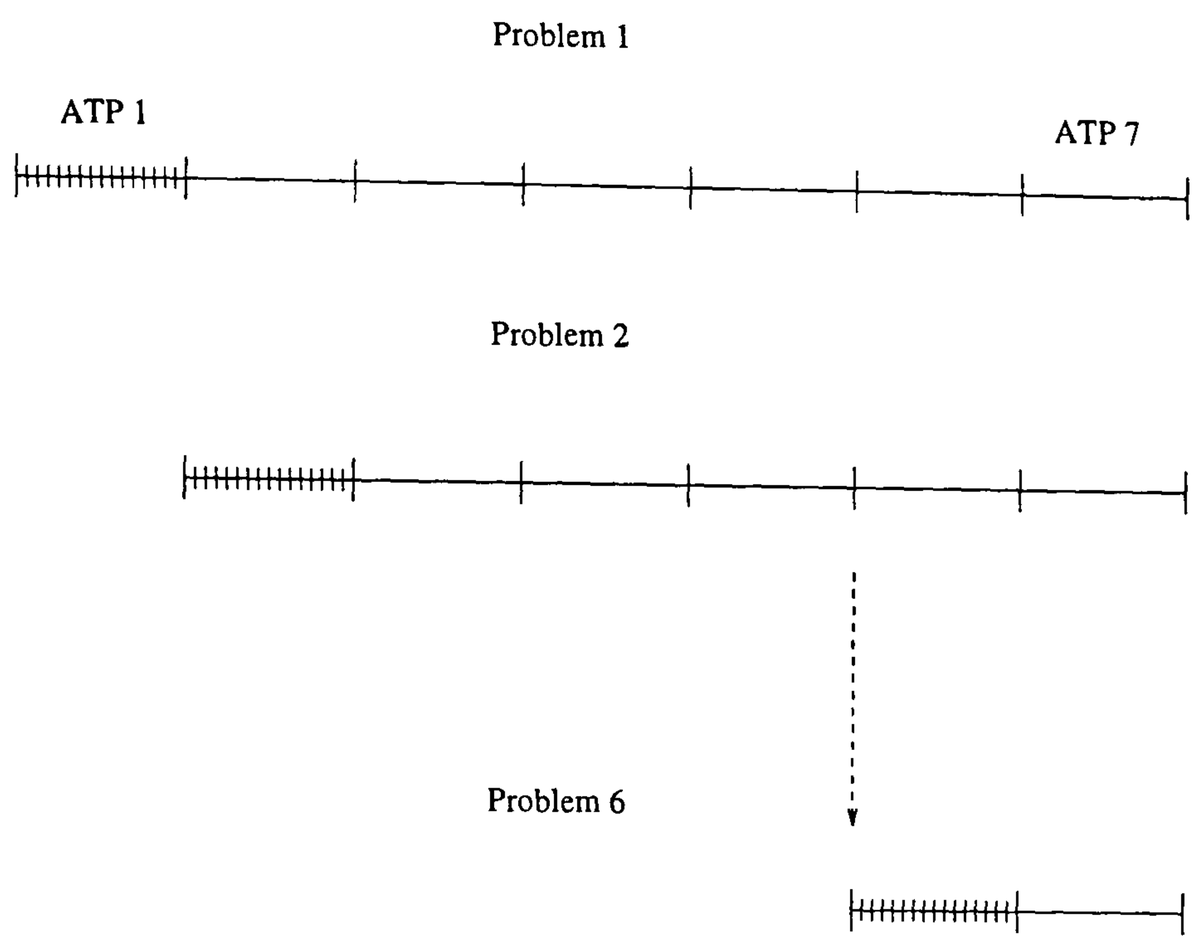


Figure 6.4: Illustration of a rolling horizon approach to generating detailed schedules.

be initiated in gaps in the existing detailed schedule.

6.6 Concluding Remarks

This chapter has considered some diverse and yet inter-related topics concerning the aggregate formulation presented in chapter 4. The physical interpretation of the zeroth and first order constraints gives a logical grounding for the concept of aggregation using time-weighted summations of constraints.

In section 6.2 it was shown that, although the linking variables increase the size of the aggregate formulation, they are needed to model interactions between adjacent ATPs. The reduction in problem size that is achieved by using an aggregate formulation of ATP length h and order m was quantified in section 6.3.

The concepts discussed in section 6.1 were reinforced in section 6.4 which compares an existing aggregate process scheduling formulation (Bassett *et al.*, 1994) with our first order aggregate formulation. The Bassett *et al.* formulation is based on certain 'intuitively obvious' constraints on material and processing equipment resources. For problems involving these

two types of resource only, it is shown to be equivalent to part of our first order aggregate formulation, once certain approximations (most notably, the omission of the linking variables) are introduced. However, it is not obvious how such an intuitive approach to aggregation could be extended to cover different types of resource (*e.g.* utilities, manpower, transportation devices, storage equipment) or to take account of complications such as sequence-dependent cleaning. Nor is it clear whether or how one could derive higher order aggregations in this manner. The advantages of deriving aggregations by the application of rigorous mathematical manipulations to the general RTN scheduling formulations are evident in this context.

Finally, in section 6.5, possible strategies for generating detailed schedules by using the aggregate formulation were discussed. This represents an important area for further work and is discussed in more detail in section 8.2. The next chapter demonstrates how the aggregate formulation can be applied to an industrial multisite case study. The aggregate solution is used to generate detailed schedules in a simple two stage procedure which gives reasonable results.

Chapter 7

Planning of Europe-Wide Multisite Production and Distribution: An Industrial Case Study

Aggregate formulations based on the Resource-Task Network (RTN) concept were derived in chapter 4, and their properties were analysed in chapters 5 and 6. In this chapter, we consider a real industrial application of the aggregate RTN formulations derived in this thesis. This involves multisite production and distribution planning on a Europe-wide basis.

As discussed in section 1.3.1, efficient scheduling in multisite production systems is becoming increasingly important as companies seek closer co-ordination of their operational activities on an international or even global basis. The problem takes on an added significance in the context of the European single market, where the opportunities and potential rewards of co-ordinated international operations are as numerous as anywhere (Collins and Schmenner, 1995). It was demonstrated in chapter 3 that multisite production and distribution systems can be modelled in terms of RTNs. Examples were given to illustrate how complex features *within* multipurpose plants could be modelled in a manner that was entirely consistent with *external* activites associated with storing and transporting materials.

In section 7.1 below, we shall give an outline of the multisite production system. For reasons of confidentiality, full details of the system and the final schedule which we obtain for it cannot be disclosed. Therefore, before the full solution is discussed, a smaller problem is presented in a fully reproducible form. Although the data for the processing structure of the smaller problem has been disguised, it has the same essential features as for the plants in the full multisite system. In addition, this problem is sufficiently small for the detailed scheduling formulation to be tractable. This allows a direct comparison against which to measure the performance of the aggregate formulation. Finally, in section 7.3, the main features of the solution to the full multisite case study are presented.

7.1 Problem Outline

7.1.1 Overview

The industrial case study considered in this chapter involves three production facilities which are to supply a large portfolio of products to a set of 15 warehouses in order to meet given customer demands. Figure 7.1 indicates the geographical scope of the problem. The plants (shown as shaded rectangles) are situated in Spain, the United Kingdom and Italy. Each country served by the plants, including those in which a plant is sited, has a single warehouse (shown as a shaded circle) which is subject to demands from the retailers in that country.

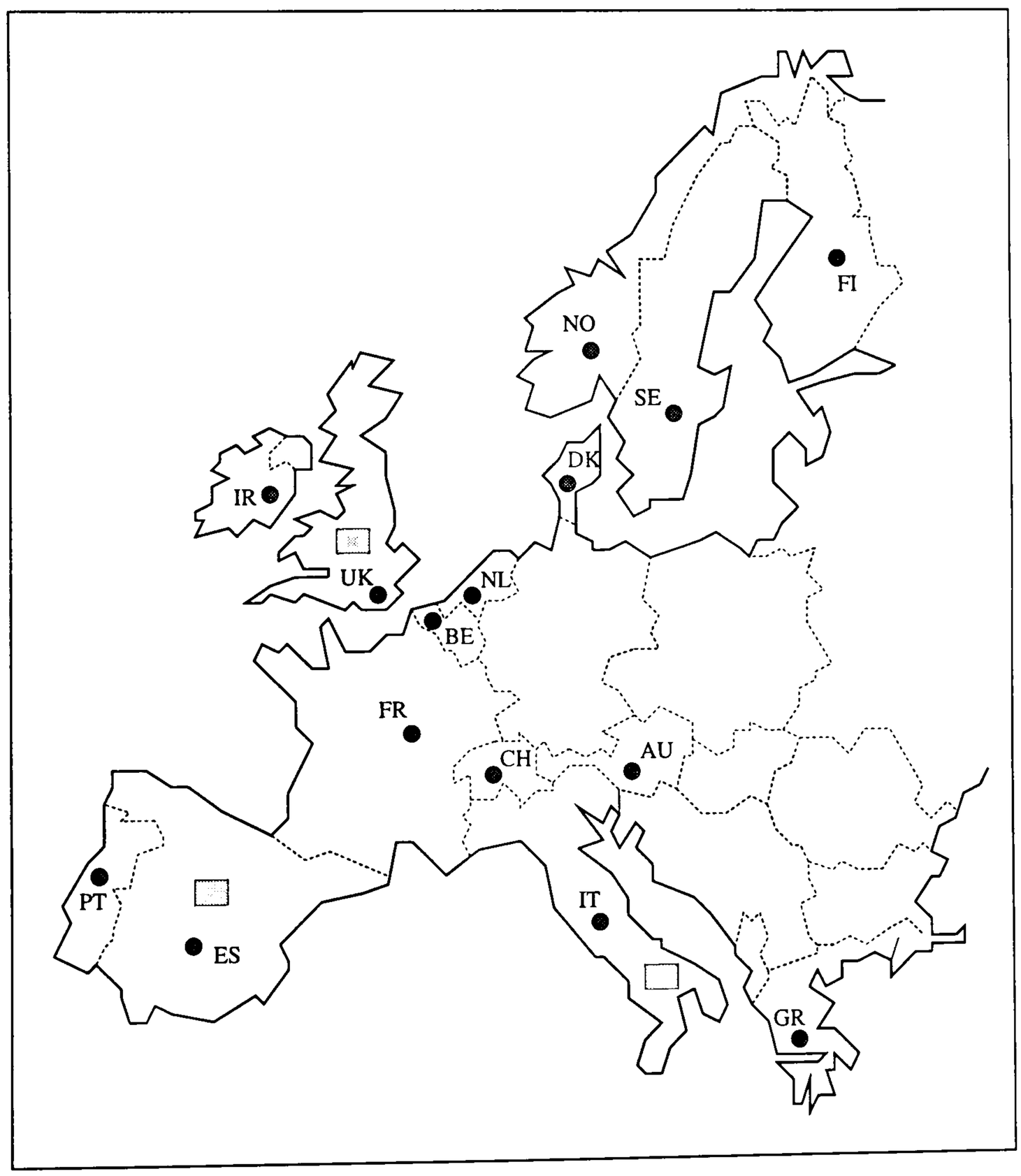


Figure 7.1: Geographical scope of the case study (only countries directly involved are labelled).

A planning horizon of one week is considered in line with the weekly replenishment policy of the operating company. This corresponds to 140 hours of production *i.e.* 5 working

days, each of three 8 hour shifts, plus a maximum of 20 hours overtime at the weekend. All deliveries from the plants to the warehouses must be sent by the end of the planning horizon. Not all the demands can be met but, in practice, this does not usually result in unfulfilled orders since each warehouse maintains safety stocks in most products which can be replenished in a subsequent week if they become depleted. The objective, therefore, is to maximise the overall profit over the planning horizon with revenue generated for the deliveries and costs associated with the production, transportation and storage tasks. This is equivalent to meeting as much of the customer demand as possible while minimising the costs associated with all the activities required to meet them.

A key property of this problem is that each plant is capable of making a large proportion of the different types of products demanded by the warehouses. Hence, there is a large degree of flexibility as to which plant can supply which product to each warehouse over the time horizon of interest. Again, efficient scheduling can lead to significant improvements in productivity, allowing costs to be kept to a minimum and the maximum number of demands to be met.

7.1.2 Process Description

All products are made at each of the plants using a two stage blending-packing processing structure. A base can be blended with different additives to produce one of a number of different variants which is then packed and labelled. The blending operations are batch tasks, each with a duration of two hours whilst the packing tasks are continuous in nature.

The complexity arises from the different variants that can be blended, the different pack-sizes in which these variants can be packed, and the different labels (corresponding to different countries) that appear on the individual packs. This results in a total of 114 different retailed product types (commonly known as 'stock keeping units' or SKUs) and is typical in the fast moving consumer goods sector which is heavily driven by marketing considerations. The large degree of diversity within the product range is designed to target particular customer preferences and hence maintain or increase market share in a highly competitive environment. A consequence of this is that the production activities become very complicated, and efficient scheduling is therefore highly desirable. A portion of the processing structure of the Italian plant showing some of the making and packing tasks which it can carry out is shown in figure 7.2.

Flexible as well as dedicated packing lines are used in all the plants. For example, all the packing tasks in the process illustrated in figure 7.2 are carried out by just two packing lines, A and B, which can pack into pack types $A1$, $A2$ and $B1$, $B2$, $B3$ respectively. The

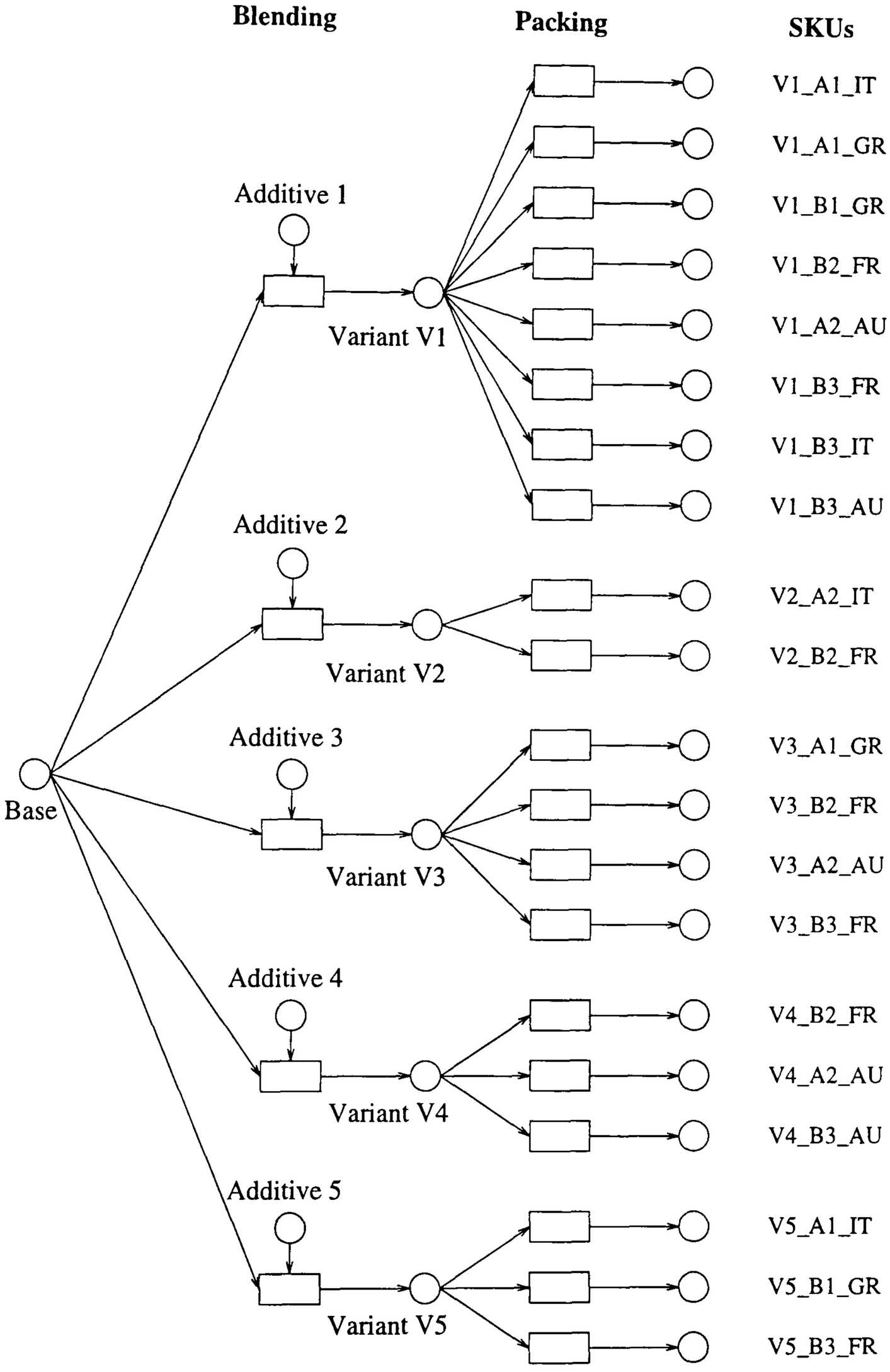


Figure 7.2: Processing structure for some products made at the Italian plant.

change-over times for the re-tooling of the flexible packing lines vary from 2 to 20 hours resulting in a significant down-time for the packing lines while they are being carried out.

Limited intermediate storage is another important constraining feature in the problem. Each plant has a fixed number of multipurpose storage silos available for storing the unpacked variants that comprise the intermediate materials between the making and packing

tasks. The UK plant, for example, can produce a total of 18 variant types but has only five silos, each of which can be used to store only one variant at any one time.

Manpower is a limited resource at all three plants and is therefore considered in the formulation of the problem. One operator is required to supervise each blending task for its two hour duration and two workers are required for the duration of each packing line change-over task. All other utility type resources such as steam, electricity, cooling water *etc.* are in plentiful supply at all the plants and do not impose any restrictions on their process operations. They are not therefore modelled in the problem formulation.

The costs of utilities (*e.g.* electricity) required for the blending and packing tasks *are* included in the formulation by means of a fixed and variable cost associated with each task. These 'aggregate' costs also include the cost of any other materials required by the task and which are not modelled explicitly in the problem formulation for the system. For the blending tasks, these include the additives that are added to the base while for the packing tasks these are the packing materials in which the final SKUs are packed. Another important component of the variable costs involved in the blending operations is a cost associated with the effluent produced from cleaning the blenders. In this case, the effluent costs are the actual costs associated with this task. However, they could equally represent an index which reflects the environmental impact of the effluent produced. Modelling these costs means that any differentials in local costs for each plant, for example those associated with labour, feed chemicals and additives, packing and labelling materials, effluent treatment, utilities *etc.*, can be included in the overall objective function in the problem formulation. Such differentials in local costs are indeed significant in this problem as is discussed in section 7.5.2.

7.1.3 Distribution of Materials

Transportation of raw materials to the production facilities is the responsibility of local suppliers and is not considered in this problem, all three plants having a large amount of storage capacity for the feed chemicals and packing materials that they use. In addition, allowing the possibility of transferring intermediates (in this case the variants) between plants is not considered necessary. This is because the blending tasks rarely constitute the bottle-neck in this problem and any differential in the cost of producing intermediates at the plants is far outweighed by the cost of transporting intermediates between them.

All products are transported from the plants to the warehouses on a third party basis by a variety of different haulage companies who quote a cost per tonne for each different pack type on each route. Detailed issues regarding the allocation of transportation resources are not considered in the formulation of this problem since the operating company is not directly

involved in the transportation activities. Transportation tasks are therefore considered to have a variable extent only with no capacity limitations and an associated unit cost. Costs are generally dependent on the distance travelled but are greater if a sea crossing is involved (*e.g.* from the UK plant to the Irish warehouse). Lead-times for these tasks vary from a few hours up to a day or more.

7.2 RTN Formulation

7.2.1 Time Discretisation

An important consideration in the formulation of this problem is the choice of time discretisation. This should be made small enough so as to take account of all the important features of the problem. In order to include all the important packing line change-overs in the case study problem, a detailed RTN formulation with a time discretisation interval of 2 hours is used. This is also the largest interval that can be used if the batch blending tasks (of duration 2 hours) are to be adequately modelled. The continuous packing task are then effectively treated as batch tasks with a maximum batch-size equal to twice their hourly production rate. Similarly, this discretisation interval implies that, if intermediates are stored between the blending and packing tasks, they can be stored only for durations that are multiples of 2 hours. Both these assumptions are considered as reasonable in the context of this large-scale problem.

It should be noted that there are other tasks in the process that operate on much shorter time-scales having durations of less than 15 minutes. These include the loading of a packing line with new labels (though obviously keeping the pack-size the same), or the cleaning tasks for the blenders. Such tasks are therefore ignored in the problem formulation for the entire multisite system in which a time discretisation of 2 hours is employed. As well as using aggregation to reduce the size of the detailed formulation in a formal mathematical sense, therefore, some intuitive aggregate assumptions are also used in developing the detailed formulation in the first place.†.

7.2.2 Size of RTN

The number of material and equipment resources at each site in the multisite system are given in table 7.1. For the manufacturing plants, the material resources comprise both variants and

† A discussion of aggregation in terms of modelling of flexible process systems is given by Liu and Sanhinidis (1995).

SKUs, whereas for the warehouses they are made up of SKUs only. The total number of equipment resources at each plant is made up of blenders, silos and packing lines. Each plant also has a manpower resource as discussed in section 7.1.2. The large number of resources (453) involved in the RTN formulation is due to the fact that the same physical material at different locations counts as two different resources.

Site	Number of material resources	Number of equipment resources	Number of manpower resources	Total
UK	129	15	1	145
Spain	96	11	1	108
Italy	113	12	1	126
France	12	0	0	12
Sweden	6	0	0	6
Ireland	13	0	0	13
Netherlands	8	0	0	8
Greece	10	0	0	10
Denmark	4	0	0	4
Finland	4	0	0	4
Portugal	4	0	0	4
Belgium	3	0	0	4
Switzerland	3	0	0	3
Norway	4	0	0	4
Austria	3	0	0	3
TOTAL	412	38	3	**453**

Table 7.1: Resources in the RTN for the multisite case study.

All the tasks in the RTN are associated with one of the three plants. Each plant has four types of processing tasks and there are also transportation tasks from the plant to the warehouses. The total number of tasks for each plant is given in table 7.2. It can be seen that there are many more transportation tasks from each plant than there are warehouses in the system. The reason for this is that SKUs with different pack types have different unit transportation costs on the same route (essentially governed by the tonnage of each pack type that haulage company can carry in one lorry load). Hence, different transportation tasks have to be introduced for each pack type on each route.

It should also be noted that, in table 7.2, the distinction is drawn between the tasks that have a discrete extent (in this case all the processing tasks because they involve equipment items) and those that do not (the transportation tasks). This has been done to give an

Site	Tasks with Discrete & Continuous Extent Variables					Transportation
	Blending	Storage	Packing	Change-over†	Total	(continuous extent only)
UK	18	36	110	18	182	54
Spain	18	18	77	14	127	51
Italy	18	18	94	12	142	63
TOTAL	54	72	281	44	**451**	168

Table 7.2: Tasks in the RTN for the multisite case study.

indication of the size of the detailed formulation of the multisite problem in terms of the number of integer variables that it contains. The planning horizon is 140 hours long and the discretisation interval is 2 hours. We therefore have 70 intervals in the detailed time horizon. There are a total of 451 tasks with discrete extents (from table 7.2), each of which is modelled by a single integer variable for every interval in the detailed time horizon. The detailed formulation of the case study problem therefore contains $70 \times 451 = 31,570$ integer variables. This is completely intractable if tackled by currently available MILP algorithms.

7.3 Solution Strategy

In order to generate a solution to the case study problem, we adopt a simple two stage approach. In the first stage, a first order aggregate formulation of the problem is solved in which the entire time horizon of 140 hours is aggregated into a single ATP. This is reasonable since all the demands have deadlines at the end of the time horizon. Even so, the aggregate formulation is still quite large (over 1,500) variables and finding an integer feasible solution‡ takes a considerable amount of computational effort (see table 7.9 in section 7.5.1).

The amounts of each SKU produced in the aggregate solution for each plant are then used in the second stage of the process. The full problem is decomposed into three detailed sub-problems, one for each plant. Each plant is given a soft demand for each SKU which is of

† In fact, the change-over tasks do not need to be modelled using continuous extent variables. However, the latter are included in order to implicitly enforce the non-negativity conditions (4.34a,b) on the discrete extent variables which are not included in the formulation. This is achieved *via* the analogous constraints on the continuous extent variables (4.33a,b) which *are* explicitly included in the formulation (see discussion in section 4.4.2). Strictly speaking therefore, a more efficient formulation would enforce (4.34a,b) directly for each change-over task, thereby removing the need for the inclusion of the continuous variables.

‡ The solutions to all problems presented in this thesis were obtained using the SCICONIC MILP code (SD-SCICON, 1990).

the same magnitude as the amount produced by that plant in the aggregate solution.

These sub-problems are several times smaller than the detailed multisite formulation since each plant is subject to demands for only a subset of the total number of SKUs that it is capable of producing. Consequently, the detailed formulation for each plant is based on much smaller RTNs than the overall aggregate formulation. Each plant RTN contains only those tasks necessary to make the SKUs which are in the target demands set by the aggregate solution.

Also, certain approximations were made with regard to the change-over tasks. In principle, a packing line that is capable of packing into n different pack types may undergo $n^2 - n$ different types of change-over. In a graph theoretic sense, if we consider the packing line states as the nodes on a graph, and the change-over tasks as directed arcs between them, then there should be complete connectivity, reflecting the fact that the packing line can be switched between any pair of states. However, in an effort to reduce problem size, we do not allow such complete connectivity in the detailed sub-problems. As long as the directed graph of states and change-overs is still *fully* connected (*i.e.* it is possible to reach all states from any other) such approximations often do not significantly compromise optimality (Papageorgiou, 1994).

Together, the solutions formed by solving the detailed sub-problems constitute a feasible solution to the overall multisite case study problem. As mentioned earlier, feasibility is ensured by having soft demands. The degree to which this feasible solution is sub-optimal can be judged, to some extent, by the proportion of the targets given by the aggregate solution that is met by the solutions to the detailed sub-problems. The higher the proportion of targets met, the better the solution is likely to be. In the highly unlikely event of 100% of the targets being met *and* the sum of the objective values of the sub-problems being equal to the aggregate objective value, a guaranteed optimal solution would have been obtained since the aggregate formulation is a strict relaxation of the corresponding detailed formulation. This is not the case for the problem considered in this chapter as is discussed in section 7.5.2. Before discussing the main features of these results, however, we shall present a smaller problem, the detailed and aggregate formulations of which can be directly compared.

7.4 Comparative Example

For this example, we shall consider a single site problem involving only the part of the process shown in figure 7.2. This corresponds to a problem in which the Italian plant attempts to meet demands for the 20 products shown in this figure; no transportation is considered in this

problem. The data given in this example have been altered in order to preserve confidentiality.

7.4.1 RTN Description

There are four types of task for this problem: blending, storage of the intermediates, packing and change-over of the packing lines. The batch blending tasks require supervision by a single operator for their entire duration of two hours. The five intermediate variants $V1$ to $V5$ can be stored in one of three identical multipurpose silos before being packed by the packing lines which operate on a continuous basis. Finally, there are change-over tasks which re-tool the packing lines in order to switch them from packing one pack type to another. All these change-over tasks take one shift (8 hours) to complete.

This process is modelled using an RTN formulation with a time discretisation interval of two hours. In order to do this, we model the continuous packing tasks as batch tasks of duration two hours with a capacity equal to twice the hourly rate of the packing line which peforms the task. We also model the storage of intermediates as batch tasks with a duration of two hours. Hence, the silos can store the variants only for multiples of two hours.

The durations of each task type, their manpower requirements, and the equipment in which they can be performed are given in table 7.3.

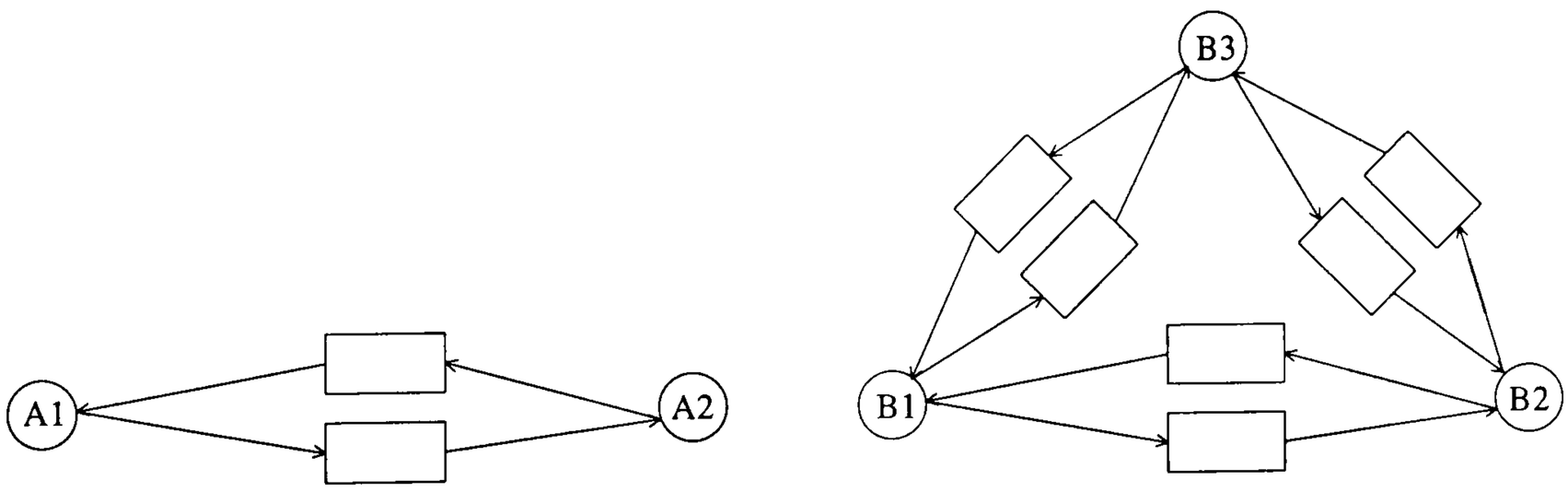
Task type	Duration	Manpower requirements	Suitable equipment
Blend any variant	2 hrs	1	*Blender*
Store any variant	2 hrs	-	*Silo*
Pack into type A1	2 hrs	-	*Packing Line A*
Pack into type A2	2 hrs	-	*Packing Line A*
Pack into type B1	2 hrs	-	*Packing Line B*
Pack into type B2	2 hrs	-	*Packing Line B*
Pack into type B3	2 hrs	-	*Packing Line B*
Change-overs	8 hrs	2	-

Table 7.3: Details of the four task types in the comparative example.

Within each task type given in table 7.3, there are 5 blending tasks and 5 storage tasks, one for each variant $V1 - V5$. There are also 20 packing tasks, one for each SKU involved. Finally, there are 8 change-over tasks, 2 for *Packing Line A* and 6 for *Packing Line B*. Hence, there are a total of 38 tasks in the RTN for this example.

The problem involves a total of 26 material resources: the base, 5 variants and 20 final products (SKUs). In figure 7.2, the SKUs have been given codenames: *Vx_pt_cd*, where *Vx* is the variant from which they were produced, *pt* is the pack type in which they are packed; and *cd* is their labelling which reflects their country of destination. The additive chemicals are not considered in the RTN because they are only used in very small quantities. Similarly, the packaging materials are not considered since sufficient supplies are always available. At the start of the time horizon, 2,000 tonnes of base are in storage. The initial storage of all other material resources is zero.

For the equipment resources, *Packing Line B* can be in one of 3 possible states to pack into one of the 3 pack types that it can produce. We shall call these states: *Packing Line B1*, *Packing Line B2* and *Packing Line B3*, corresponding to the pack types that they produce. Similarly, *Packing Line A* can be in one of 2 possible states since it can produce 2 different pack types. This is illustrated in figure 7.3.



Packing Line A

Packing Line B

Figure 7.3: RTN representation of change-overs on packing lines *A* and *B*.

The 5 silos are identical both in their capacity and in their task suitability and are therefore considered in the RTN as a single equipment resource. Hence, there are 7 equipment resources: the 5 packing line states, the blender and the silos. The initial availabilities of the equipment resources and their maximum capacities are given in table 7.4.

Finally, the operators required for the blending and change-over tasks are provided by the manpower resource. A total of 4 operators are available. There are therefore a total of 34

resource types in the RTN for this example.

Equipment	Initial availability	Capacity (te)
Blender	1	25.0
Silo	5	80.0
Packing line A1	0	9.0
Packing line A2	1	13.0
Packing line B1	0	5.0
Packing line B2	0	7.5
Packing line B3	1	10.0

Table 7.4: Initial availabilities and capacities for the equipment resources.

The maximum capacity of the equipment resources governs the maximum extent $V_k^{\max}$ for any task k which it can perform. It should also be noted that $V_k^{\min}$ is zero for all tasks considered in this problem. This means that any amount of material, however small, can be blended or stored. The continuous packing line can only operate at a fixed rate. A task extent which is less than $V_k^{\max}$ for the packing tasks, therefore, corresponds to the packing line lying idle for some of the 2 hour discretisation interval over which the task is occurring. This operational possibility is therefore modelled by setting $V_k^{\min}$ equal to zero for the packing tasks as well as for the blending and storing tasks.

7.4.2 Problem Specification

The scheduling objective is to maximise profit over the time horizon considered which is 6 days (144 hours) *i.e.* 72 time discretisation intervals. The demands for all 20 SKUs are soft. There are no costs associated with any of the tasks or the holding of stock. It should also be noted that no material is delivered to external clients or received from external sources during the horizon ($\Pi_{rt} = 0 \ \forall r, t$). In this case, the detailed and aggregate objective functions (3.5) and (4.44) (see section 4.5) both have the same simplified form:

$$\max \sum_{r \in MR} C_r^F (R_{rH} - R_{r0}) \tag{7.1}$$

Each tonne of each SKU in storage at the end of the time horizon (up to the demand tonnage) is assumed to contribute £100 to the profit ($C_r^F = 100.0 \ \forall r \in SKUs$). The base raw material has an end-of-horizon value of £10 per tonne ($C_{Base}^F = 10.0$) whereas the variants have no

end-of-horizon value and are therefore omitted from the objective function. Finally, there are no SKUs in storage at the start of the time horizon ($R_{r0} = 0.0 \ \forall r \in SKUs$). The objective function (7.1) therefore becomes:

$$\max \ 10 \left(R_{Base,H} - R_{Base,0} \right) + 100 \sum_{r \in SKUs} R_{rH} \tag{7.2}$$

Of course, the initial amount of base is constant in this example ($R_{Base,0} = 2000.0$), and can therefore be excluded from the objective function leaving:

$$\max \ 10 \ R_{Base,H} + 100 \sum_{r \in SKUs} R_{rH} \tag{7.3}$$

The requirement that no more of each SKU r should be produced than the demand for it, Δ_r, is enforced simply by the constraints:

$$R_{rH} \leq \Delta_r \qquad \forall \ r \in SKUs$$

which can be seen as a special case of the resource capacity of the detailed formulation (3.3), and also the linking resource capacity constraints of the aggregate formulation (4.22).

7.4.3 Results

The problem outlined above was solved using the detailed RTN formulation and also a first order aggregate formulation based on splitting the time horizon ($H = 72$ discretisation intervals) into 3 ATPs, each of length 48 hours ($h = 24$ discretisation intervals). The computational statistics for both are given in table 7.5, while the amount of each SKU produced in both solutions is given in table 7.6 along with the demands. For sets of results, there are no variants in stock at the end of the time horizon. This is to be expected since they have no value themselves but their production depletes the base which does have value. For these solutions, therefore, the depletion of base in storage is due to the production of SKUs only. Therefore:

$$R_{Base,H} = \left(R_{Base,0} - \sum_{r \in SKUs} R_{rH} \right) = \left(2000 - \sum_{r \in SKUs} R_{rH} \right)$$

Therefore, using the above in (7.3), the profits given in table 7.5 are related to the total production given in table 7.6 by:

$$profit = 10\left(2000 - \sum_{r \in SKUs} R_{rH}\right) + 100 \sum_{r \in SKUs} R_{rH}$$

$$= 20000 + 90 \sum_{r \in SKUs} R_{rH}$$

The similarity of the production patterns for each solution is also demonstrated graphically in figure 7.4.

COMPUTATIONAL STATISTICS	Detailed Formulation	Aggregate Formulation $h = 24,\ m = 1$
Best integer feasible objective value obtained (profit in £)	101,630	107,750
Fully relaxed objective value (£)	110,731	111,909
Number of integer variables	2,736	414
CPU seconds (SUN SPARC 10)	25,279†	86
Number of nodes examined	834†	293

† Up to second integer feasible solution during branch-and-bound search.

Table 7.5: Computational statistics for the detailed and aggregate solutions to the comparative example.

The aggregate solution was obtained using a margin of optimality of 2%. The detailed solution represents the second integer feasible solution found by the MILP solver and is not necessarily optimal: the search for a better solution proved to be fruitless even after more than 60 hours of CPU time. In fact, there is evidence for some sub-optimality in the detailed solution when the occurrence of change-over tasks is examined. This is discussed later.

Looking at table 7.5, the problem size reduction factor (in terms of integer variables) achieved by the aggregate formulation is:

$$\frac{414}{2736} = 0.1513$$

The same figure can also be arrived at using equation (6.5) and (6.4) (see section 6.3) which is recalled below:

$$\frac{no.\ integer\ variables\ in\ aggregated\ formulation}{no.\ integer\ variables\ in\ detailed\ formulation} = \frac{\overline{\tau} + m + 1}{h} \qquad (6.5)$$

where:

SKU	Soft Demand (tonnes)	Production (tonnes)		Difference Aggr. – Det. (tonnes)
		Detailed Solution	Aggregate Solution	
V1_A1_IT	116.0	116.0	116.0	0.0
V1_A1_GR	9.0	9.0	9.0	0.0
V1_B1_GR	16.0	0.0	0.0	0.0
V1_B2_FR	81.0	0.0	30.0	30.0
V1_A2_AU	40.0	40.0	39.0	-1.0
V1_B3_FR	7.0	7.0	7.0	0.0
V1_B3_IT	202.0	200.0	200.0	0.0
V1_B3_AU	24.0	10.0	20.0	10.0
V1_A2_IT	22.0	22.0	22.0	0.0
V2_B2_FR	35.0	0.0	0.0	0.0
V3_A1_GR	8.0	8.0	8.0	0.0
V3_B2_FR	47.0	0.0	45.0	45.0
V3_A2_AU	25.0	25.0	25.0	0.0
V3_B3_FR	98.0	95.0	90.0	-5.0
V4_B2_FR	90.0	0.0	0.0	0.0
V4_A2_AU	31.0	31.0	31.0	0.0
V4_B3_AU	184.0	180.0	180.0	0.0
V5_A1_IT	89.0	89.0	81.0	-8.0
V5_B1_GR	140.0	5.0	0.0	-5.0
V5_B3_FR	72.0	70.0	72.0	2.0
TOTAL	1,336.0	907.0	975.0	68.0

Table 7.6: Demands for each SKU and the actual production for the detailed and aggregate solutions.

$$\overline{\tau} = \frac{\sum_{k=1}^{q} \tau_k}{q} \tag{6.4}$$

For this RTN, there are a total of $q = 38$ tasks, 30 of which have a duration of one discretisation interval, with the remaining eight (the change-over tasks) having a duration of four time intervals. Using (6.4), therefore, we can calculate the average task duration as:

$$\overline{\tau} = \frac{(30 \times 1) + (8 \times 4)}{38} = 1.6316$$

Hence, from (6.5) we get:

$$\frac{\textit{no. integer variables in aggregate formulation}}{\textit{no. integer variables in detailed formulation}} = \frac{1.6316 + 1 + 1}{24} = 0.1513$$

as above.

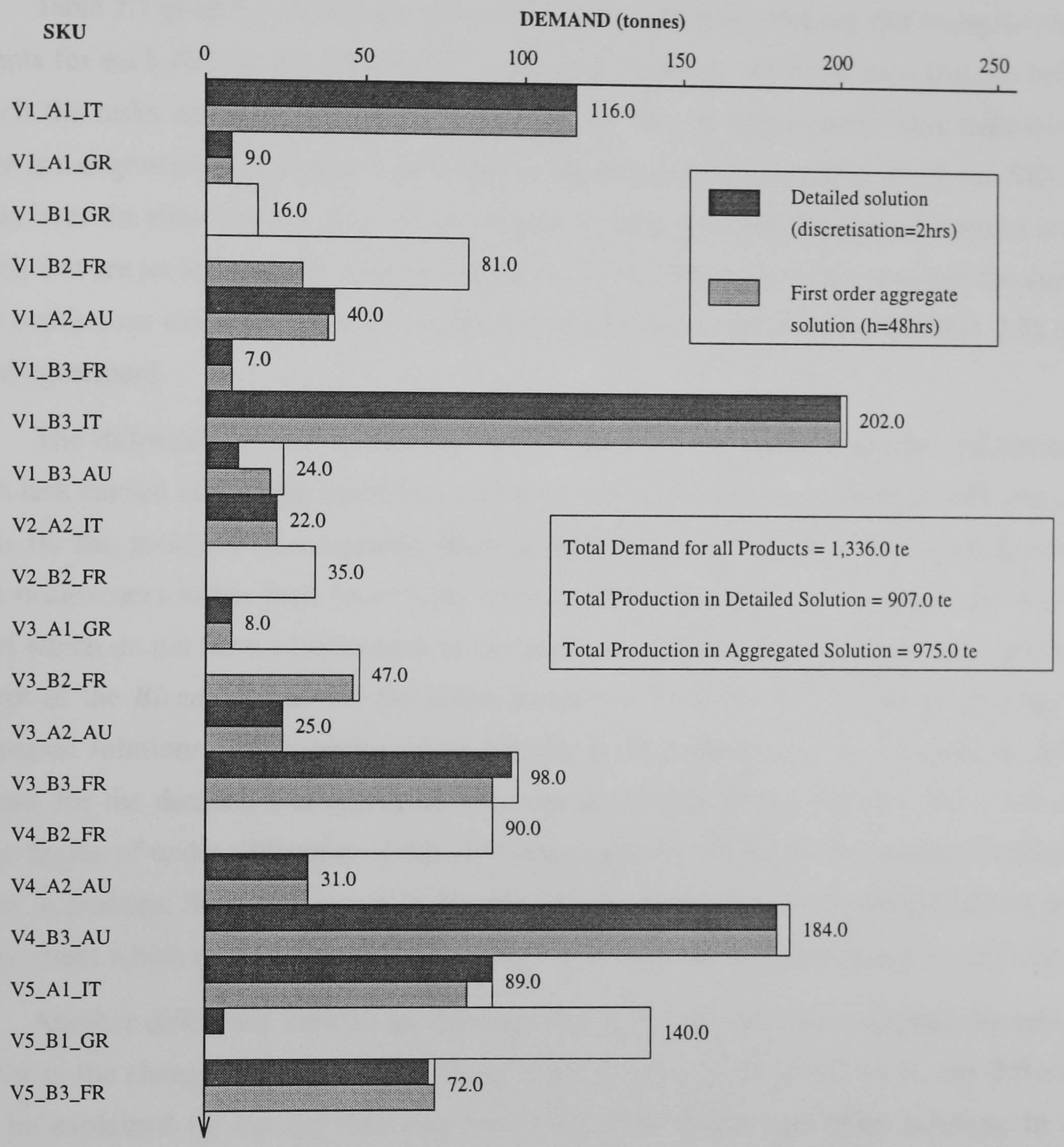


Figure 7.4: Comparison of SKU tonnages for the detailed and aggregate solutions.

It can also be seen from table 7.5 that the reduced number of integer variables in the aggregate formulation translates into a much reduced amount of CPU time required to obtain the solution.

Considering the total tonnage of SKUs produced, table 7.6 demonstrates that the aggregate solution gives a 7.5% over-estimate of the overall production capacity of the plant as represented by the detailed solution. In addition, the similarity of the patterns of production is indicated by figure 7.4. Because the demands cannot be met in full, the SKUs that are produced tend to be those that can be made at the highest rates (*i.e.* tonnes per hour). These SKUs can contribute most to the overall profit per unit time since all SKUs have the same

end-of-horizon value per tonne.

Table 7.7 gives a comparison of the cumulative blending, packing and change-over task extents for each ATP in the detailed and aggregate solutions. It can be seen that the order in which the tasks are performed differs between the two in many cases. This indicates that there is a degree of degeneracy in the problem regarding the sequence in which the SKUs are made over the time horizon and is to be expected since it is only the end-of-horizon storage levels that are included in the objective function. Despite this, it can be seen that the cumulative continuous extents for the *entire* horizon (given in the last column of table 7.7) are in good agreement.

The difference in the cumulative discrete extents (*i.e.* the total number of batches of each task carried out) is due to the fact that there are no fixed costs associated with any of the tasks for this problem. Consequently, there is no penalty for having several partially utilised task occurrences rather than fewer fully utilised ones. Consider, for example, the blending tasks which do not form a bottle-neck in this problem. In table 7.7, the cumulative continuous extent of the *Blend V2* task for the entire horizon is 22.0 tonnes for both the detailed and aggregate solutions. The capacity of the blender is 25.0 tonnes and the cumulative discrete extents for the detailed and aggregate solutions are 5 and 13 respectively. This indicates a large degree of under-utilisation of the 25.0 tonne capacity of the blender, which is unlikely to occur in practice. Such differences in discrete extents, therefore, can be attributed to a degeneracy effect which could be removed by assigning a fixed penalty cost to task occurrences.

Another difference betwen the detailed and aggregate solutions indicated by table 7.7 relates to the change-over tasks. For change-overs involving *Packing Line A*, any differences can be explained by the fact that this line is not fully utilised in either solution. In other words, it has more than enough time to meet all the demands for the SKUs of pack type *A*. Any idle time that it has can therefore be used to undergo non-essential change-overs without decreasing the objective value. *Packing Line B*, on the other hand, *is* heavily utilised in both solutions — in fact, several demands for SKUs of pack type *B* are not met. The extra change-over tasks for this line in the detailed solution are therefore likely to be due to its sub-optimality.

Finally, mention should be made of the silo and manpower resources. The detailed solution to this problem suggests that both of them are initially available in quantities that are sufficient not to cause them to limit the occurrences of the tasks for which they are required. All three silos are simultaneously utilised on just three occasions over the time horizon. For the manpower resource, there is always at least one operator surplus to requirements throughout the time horizon.

Task	Soln.	ATP 1 ($t = 1 \rightarrow 24$)		ATP 2 ($t = 25 \rightarrow 48$)		ATP 3 ($t = 49 \rightarrow 72$)		Total	
Blend V1	Det.	80.0	(8)	242.0	(10)	60.0	(4)	382.0	(22)
	Agg.	306.6	(13)	74.1	(3)	40.3	(3)	421.0	(19)
Blend V2	Det.	0.0	(0)	22.0	(5)	0.0	(0)	22.0	(5)
	Agg.	0.0	(0)	22.0	(13)	0.0	(0)	22.0	(13)
Blend V3	Det.	95.0	(5)	0.0	(0)	33.0	(6)	128.0	(11)
	Agg.	24.5	(1)	70.0	(3)	73.5	(3)	168.0	(7)
Blend V4	Det.	110.0	(7)	60.0	(5)	41.0	(5)	211.0	(17)
	Agg.	46.0	(2)	75.0	(4)	90.0	(4)	211.0	(10)
Blend V4	Det.	50.0	(4)	32.0	(2)	82.0	(7)	164.0	(13)
	Agg.	72.0	(3)	9.0	(1)	72.0	(4)	153.0	(8)
Pack V1_A1_IT	Det.	55.0	(13)	61.0	(11)	0.0	(0)	116.0	(24)
	Agg.	116.0	(13)	0.0	(0)	0.0	(0)	116.0	(13)
Pack V1_A1_GR	Det.	0.0	(0)	9.0	(3)	0.0	(0)	9.0	(3)
	Agg.	8.7	(1)	0.0	(0)	0.3	(1)	9.0	(2)
Pack V1_B2_FR	Det.	0.0	(0)	0.0	(0)	0.0	(0)	0.0	(0)
	Agg.	0.0	(0)	0.0	(0)	30.0	(4)	30.0	(4)
Pack V1_A2_AU	Det.	0.0	(0)	0.0	(0)	40.0	(4)	40.0	(4)
	Agg.	26.0	(2)	13.0	(1)	0.0	(0)	39.0	(3)
Pack V1_B3_FR	Det.	0.0	(0)	7.0	(1)	0.0	(0)	7.0	(1)
	Agg.	7.0	(1)	0.0	(0)	0.0	(0)	7.0	(1)
Pack V1_B3_IT	Det.	0.0	(0)	120.0	(12)	80.0	(8)	200.0	(20)
	Agg.	110.0	(11)	80.0	(8)	10.0	(1)	200.0	(20)
Pack V1_B3_AU	Det.	0.0	(0)	0.0	(0)	10.0	(1)	10.0	(1)
	Agg.	10.0	(1)	10.0	(1)	0.0	(0)	20.0	(2)
Pack V1_A2_IT	Det.	0.0	(0)	22.0	(4)	0.0	(0)	22.0	(4)
	Agg.	0.0	(0)	22.0	(12)	0.0	(0)	22.0	(12)
Pack V3_A1_GR	Det.	0.0	(0)	0.0	(0)	8.0	(1)	8.0	(1)
	Agg.	4.5	(1)	0.0	(0)	3.5	(1)	8.0	(2)
Pack V3_B2_FR	Det.	0.0	(0)	0.0	(0)	0.0	(0)	0.0	(0)
	Agg.	0.0	(0)	0.0	(0)	45.0	(6)	45.0	(6)
Pack V3_A2_AU	Det.	0.0	(0)	0.0	(0)	25.0	(2)	25.0	(2)
	Agg.	0.0	(0)	0.0	(0)	25.0	(2)	25.0	(2)
Pack V3_B3_FR	Det.	65.0	(7)	30.0	(3)	0.0	(0)	95.0	(10)
	Agg.	10.0	(1)	80.0	(8)	0.0	(0)	90.0	(9)
Pack V4_A2_AU	Det.	0.0	(0)	0.0	(0)	31.0	(3)	31.0	(3)
	Agg.	26.0	(3)	5.0	(1)	0.0	(0)	31.0	(4)
Pack V4_B3_AU	Det.	110.0	(11)	60.0	(6)	10.0	(1)	180.0	(18)
	Agg.	20.0	(2)	70.0	(7)	90.0	(9)	180.0	(18)
Pack V5_A1_IT	Det.	0.0	(0)	12.0	(2)	77.0	(9)	89.0	(11)
	Agg.	0.0	(0)	9.0	(1)	72.0	(8)	81.0	(9)
Pack V5_B1_GR	Det.	0.0	(0)	0.0	(0)	5.0	(1)	5.0	(1)
	Agg.	0.0	(0)	0.0	(0)	0.0	(0)	0.0	(0)
Pack V5_B3_FR	Det.	50.0	(5)	20.0	(2)	0.0	(0)	70.0	(7)
	Agg.	72.0	(8)	0.0	(0)	0.0	(0)	72.0	(8)
Change B3_B2	Det.	-	(0)	-	(0)	-	(1)	-	(1)
	Agg.	-	(0)	-	(0)	-	(1)	-	(1)
Change B3_B1	Det.	-	(0)	-	(0)	-	(1)	-	(1)
	Agg.	-	(0)	-	(0)	-	(0)	-	(0)
Change B2_B3	Det.	-	(0)	-	(0)	-	(1)	-	(1)
	Agg.	-	(0)	-	(0)	-	(0)	-	(0)
Change B1_B3	Det.	-	(0)	-	(0)	-	(1)	-	(1)
	Agg.	-	(0)	-	(0)	-	(0)	-	(0)
Change A2_A1	Det.	-	(1)	-	(0)	-	(1)	-	(2)
	Agg.	-	(1)	-	(1)	-	(2)	-	(4)
Change A1_A2	Det.	-	(0)	-	(1)	-	(1)	-	(2)
	Agg.	-	(0)	-	(2)	-	(1)	-	(3)

Table 7.7: Continuous and discrete (in brackets) task extents for detailed and aggregate solutions.

7.5 Overview of Results for Full Multisite Case Study

A feasible detailed solution for the case study problem described in section 7.1 is derived by first solving an aggregate formulation and then using this solution to decompose the problem into three detailed sub-problems as described in section 7.3. As for the comparative example, an RTN formulation with a time discretisation interval of 2 hours is used in order to include all the important packing line change-overs. Again, this means that the continuous processing tasks (blending and packing) are effectively treated as batch tasks with a maximum batch-size equal to twice their hourly production rate and intermediates can be stored only for durations that are multiples of 2 hours.

7.5.1 Stage I: Aggregate Solution for Multisite System

Based on the detailed RTN formulation, we derive an aggregate formulation with a single ATP spanning the entire 140 hour planning horizon. The aggregate solution for the multisite system determines the deliveries from the plants to the warehouses and the equipment utilisation at each plant. Table 7.8 gives the total tonnage of the deliveries from the plants to each warehouse in this solution. The total number of different product types comprising each delivery is also given. These results are also represented in graphical form in figures 7.5 and 7.6.

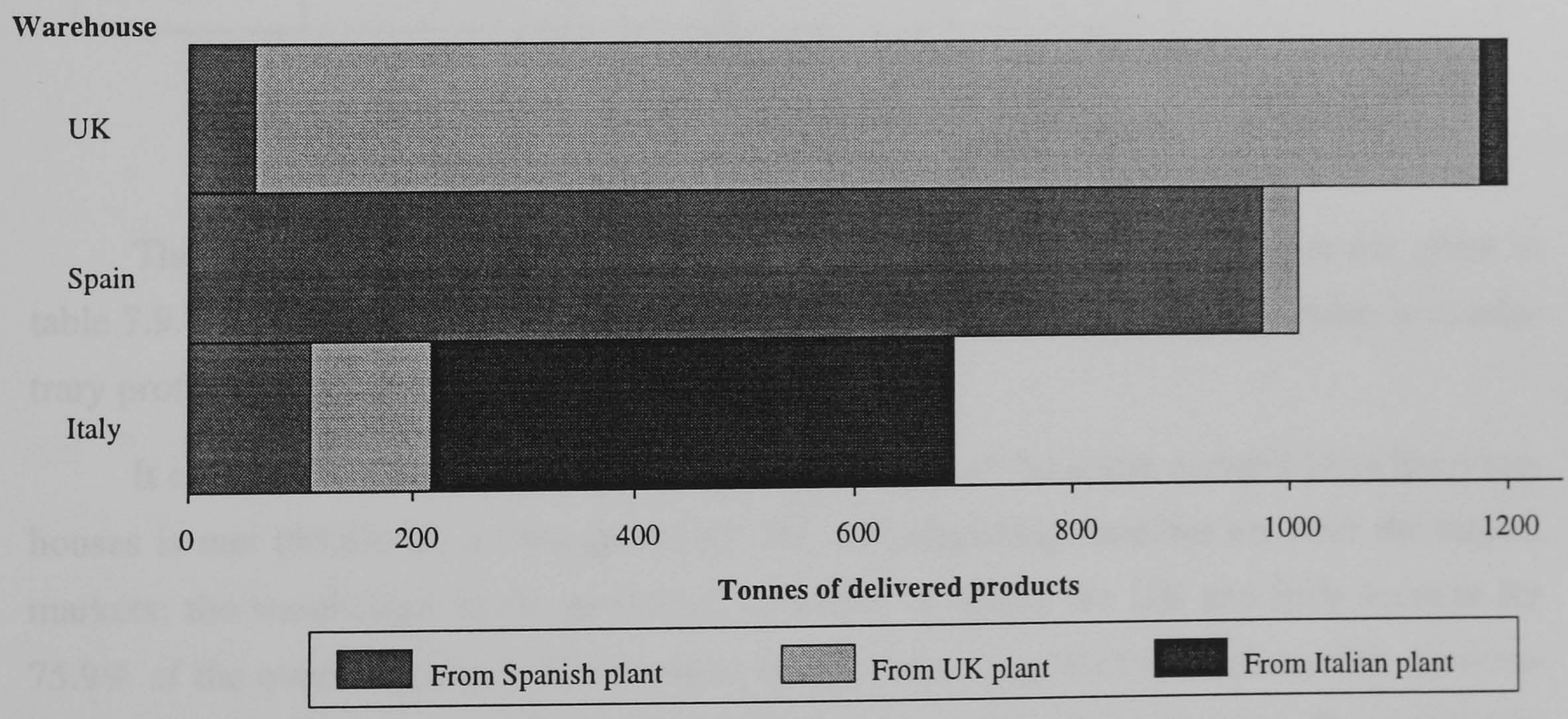


Figure 7.5: Deliveries to the warehouses in the producing countries as determined by the aggregate solution.

Warehouse	Demand		Deliveries from plants			Unfulfilled demand
	Tonnage	No. Products	Spain	UK	Italy	
UK	1199.0	22	60.0	1118.0	21.0	0.0
Spain	1011.0	10	980.0	31.0	0.0	0.0
Italy	692.0	8	104.8	113.0	474.2	0.0
France	246.0	12	215.0	31.0	0.0	0.0
Sweden	155.0	6	38.0	108.6	0.0	8.4
Ireland	123.0	13	0.0	120.5	0.0	2.5
Netherlands	89.0	8	42.0	34.1	0.0	12.9
Greece	70.0	10	0.0	0.0	64.0	6.0
Denmark	54.0	4	49.0	5.0	0.0	0.0
Finland	42.0	4	0.0	26.0	0.0	16.0
Portugal	38.0	4	38.0	0.0	0.0	0.0
Belgium	36.0	3	18.6	14.0	3.4	0.0
Switzerland	32.0	3	32.0	0.0	0.0	0.0
Norway	24.0	4	0.0	24.0	0.0	0.0
Austria	12.0	3	9.0	0.0	3.0	0.0
Totals	3823.0	114	1586.4	1625.2	565.6	45.8

Table 7.8: Deliveries from plants to warehouses in the aggregate solution.

The computational statistics for the solution of the aggregate formulation are given in table 7.9. This solution is optimal to within a margin of 5% and the objective value is in arbitrary profit units.

It can be seen that, for the aggregate solution, almost the entire demand from the warehouses is met (98.8% on a tonnage basis). The manufacturing facilities are near the largest markets: the warehouses in the producing countries of Spain, the UK and Italy account for 75.9% of the overall demand. The demand in most other countries is much smaller by comparison — these are the growth markets which the company is keen to target. Transportation costs make up a significant fraction of the total and this is reflected by the fact that most (but not all) of the demands in Spain, the UK and Italy are met by the corresponding local plant. Where this is not the case is due to the fact that it can often be cheaper to concentrate production of certain types of lower volume SKUs at a single plant, rather than try to produce everything as near as possible to where it is consumed.

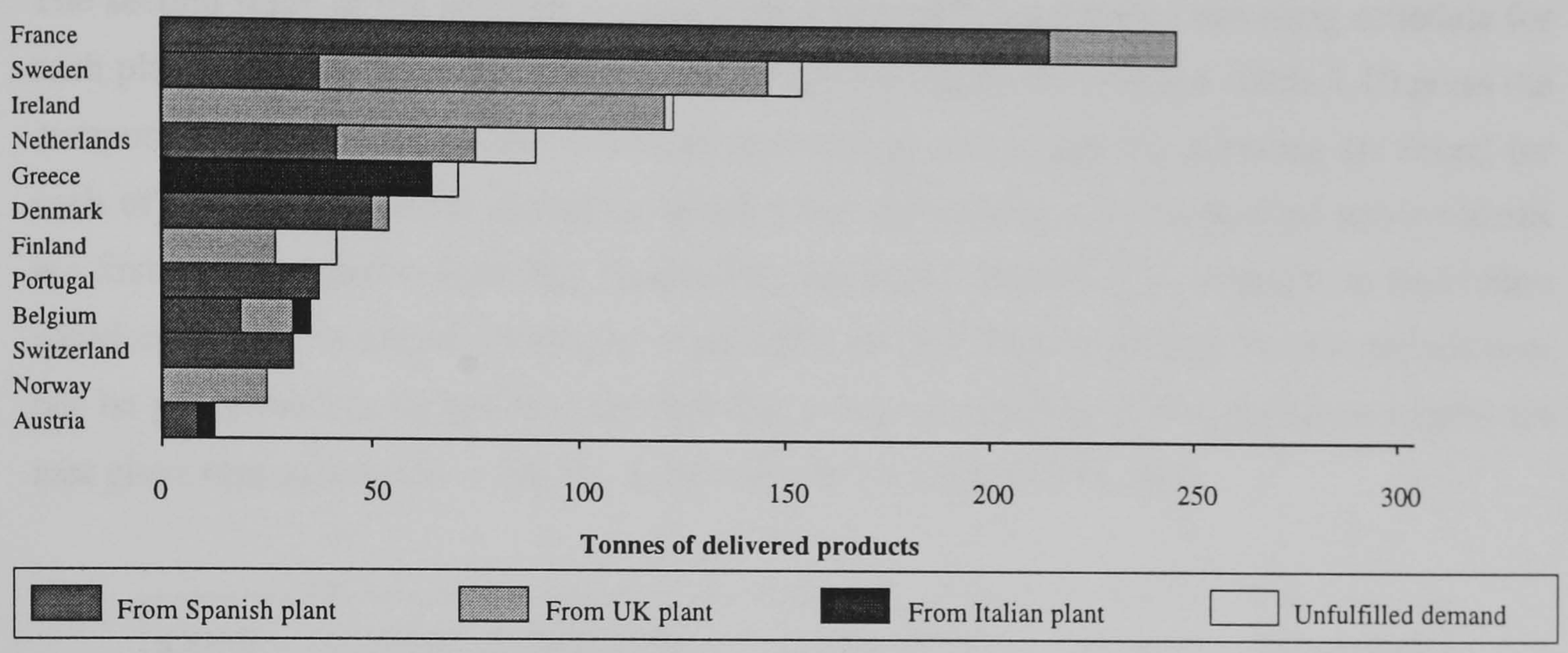


Figure 7.6: Deliveries to the warehouses in the non-producing countries as determined by the aggregate solution.

Best integer feasible objective obtained:	2,483
Fully relaxed objective function:	2,713
Number of integer variables:	1,586
CPU seconds (SUN SPARC 10):	5,570
Number of nodes examined:	40,125

Table 7.9: Computational statistics for the aggregate solution.

The unfulfilled demands are mainly products that can be made only on two particular lines at the UK plant which is the most heavily utilised one due to its low raw material and labour costs. In the aggregate solution, neither of these lines (which we shall call P and Q) are fully utilised, which indicates that the limited intermediate storage (five 30 tonne silos for the 14 variants made at the plant) may well be the bottle-neck resource during some parts of the planning horizon. Another reason for demand not being met is due to company policy regarding the countries which each plant can serve. For example, some of the unfulfilled Finnish demand could have been met from the Spanish plant but this was ruled out as a possibility from the start.

7.5.2 Stage II: Detailed Production Schedules for Individual Plants

The second stage of the solution approach aims to establish a detailed operating schedule for each plant using the production targets defined by the aggregate solution. Table 7.10 gives the computational statistics for the three sub-problems in which detailed solutions are found for each of the three plants. It should be stressed that the solutions to the detailed sub-problems are first integer feasible solutions obtained by the solver. In each case, attempts to find better solutions were terminated after many more hours of CPU time. Although these solutions cannot be guaranteed to be optimal, the fact that a large proportion of the aggregate targets are met gives reason to believe that the degree of sub-optimality is not great.

COMPUTATIONAL STATISTICS	Spanish plant	UK plant	Italian plant
Best integer feasible objective obtained:	886	725	361
Fully relaxed objective function:	1,182	1,115	413
Number of integer variables:	4,690	6,440	2,940
CPU seconds (SUN SPARC 10):	6,719	9,538	380
Number of nodes examined:	2,216	3,188	505

Table 7.10: Computational statistics for the detailed solutions to the sub-problems.

Figure 7.7 shows the actual production that can be achieved by each plant in response to the aggregate production targets for each of the 110 products that were produced in the aggregate solution.

The Spanish and Italian plants respectively meet 97.1% and 99.2% of their aggregate targets on total tonnage basis. The fact that the UK plant is more heavily utilised in the aggregate solution means that its targets are slightly more over optimistic but it still meets 90.8% of these target demands in the detailed schedule.

It can be seen that the demands that are not met at the UK plant tend to be the lower volume demands. The reason for this is that all the deliveries were given the same value per tonne in the objective function for this problem. It is therefore more beneficial to meet 100% of the larger orders which tend to be for nearby countries (UK, Republic of Ireland *etc.*) and which, therefore, have cheaper delivery costs. Any production shortfall occurs in the orders for more distant countries (*e.g.* those in Scandinavia); these also tend to be the lower volume orders. In practice, it is likely to be highly undesirable to fail to produce certain low volume products altogether. One possible remedy would be to constrain the detailed solution to

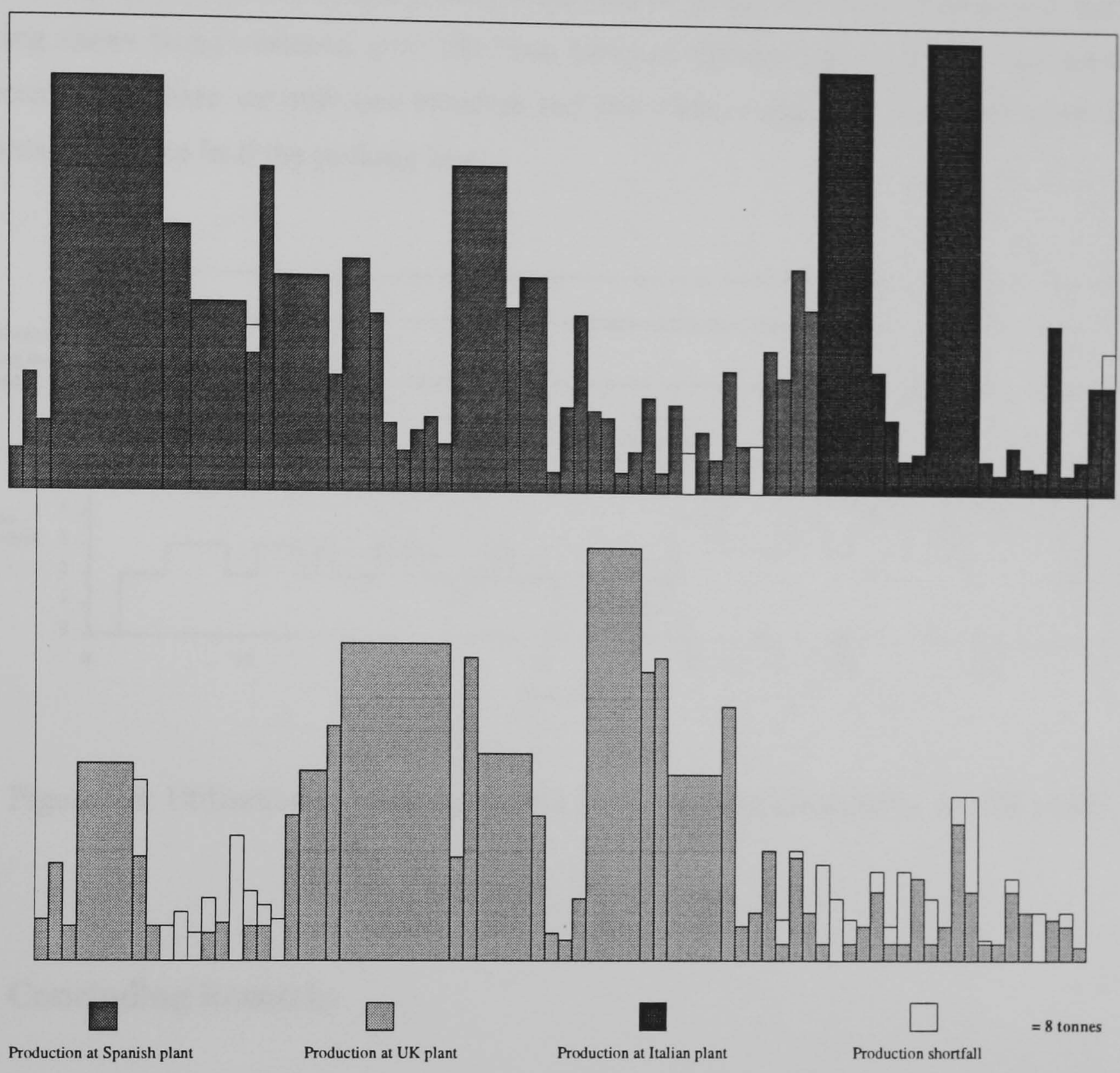


Figure 7.7: Histogram showing achievable production at each plant.

produce a certain fraction (say 80%) of each target product.

It is worth mentioning that the weekly demands used for this case study were considerably higher than average. In practice, any depletions in warehouse safety stocks due to delivery shortfalls in this week could be eliminated in future weeks when demands are likely to be lower.

For the UK plant, figure 7.8 gives the number of silos being used to store material at any time as determined by the detailed solution. The utilisation of packing lines P and Q referred to earlier is also shown. Limited intermediate storage for the variants is an important bottle-neck in this plant — the silo utilisation profile hits the maximum level of five on several occasions over the planning horizon. This accounts for the fact that lines P and Q do not always run at their maximum rate and also need a large number of change-overs. Line Q, for

example, only packs three different pack-sizes and can therefore be in one of three packing states at any one time. However, it undergoes a total of six change-overs with two of the three packing states being revisited over the time horizon. Complications of this sort are to be expected when there are only two blenders and five silos available to make and store all the variants that are to feed the packing lines.

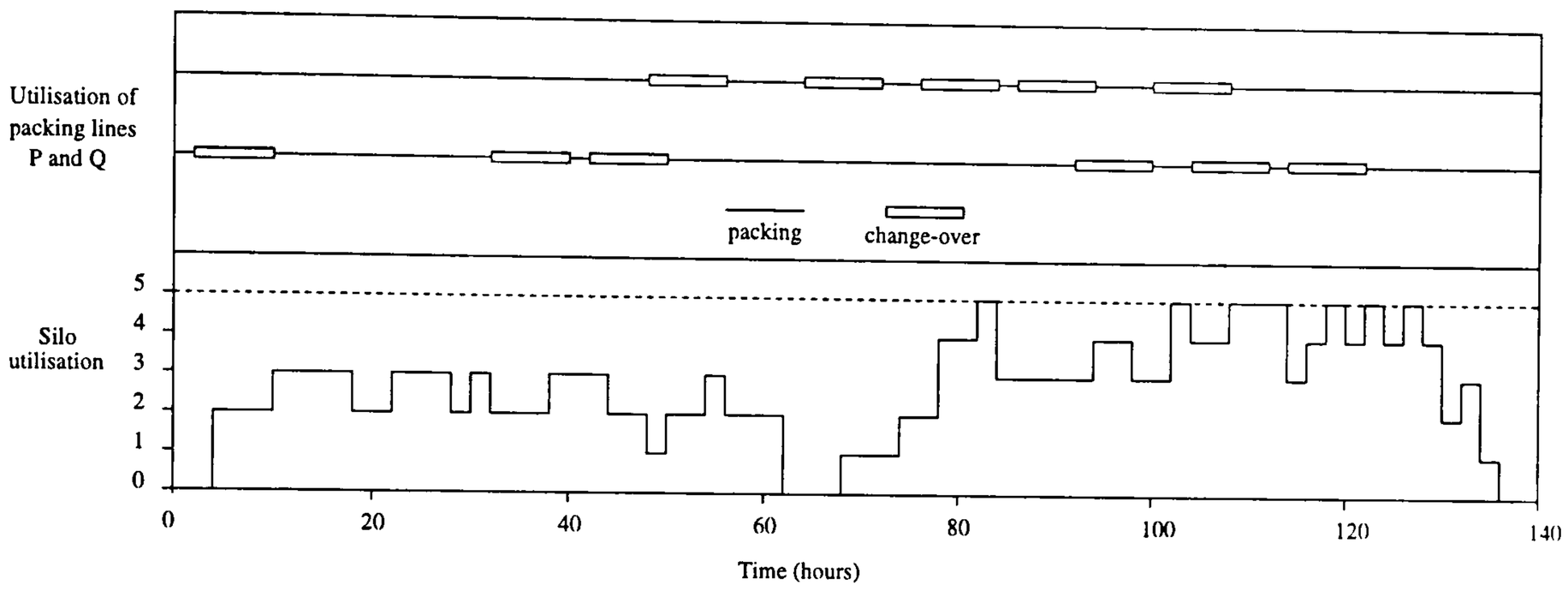


Figure 7.8: Utilisation of some equipment in the detailed solution for the UK plant.

7.6 Concluding Remarks

This chapter has demonstrated how the aggregate RTN formulation can be applied to a large industrial case study involving several multipurpose plants and warehouses. The system needs to be considered in its entirety for a schedule that takes account of all important interactions to be found. In addition, the plants require detailed production models. This gives rise to a problem that is probably an order of magnitude larger than the capability of current computing resources. Despite this, practically useful results are obtained using a solution strategy based on the aggregate RTN formulation.

The Europe-wide system is modelled in detail using the RTN framework which allows operational complexities (packing line change-overs, limited intermediate storage *etc.*) to be adequately represented in the problem. This allows a realistic objective function to be developed that represents the detailed costs of all parts of the production distribution network.

In the two-stage solution approach, the aggregate solution was used to set production targets for each plant. This allowed the problem to be decomposed into three smaller subproblems, each involving a single plant for which detailed schedules were derived with the objective of meeting as many of the targets as possible. The comparative example

demonstrated that the aggregate formulation gives a good approximation to the true production capacity of the flexible plants involved in this case study. This, combined with the fact that a large proportion of the aggregate targets are actually met in the full multisite problem, suggests that a reasonable solution has indeed been found.

The fact that the demands were regarded as being soft allowed a fairly simple solution strategy to be successful in this case. Other problems, such as those involving hard constraints, may require more sophisticated approaches such as those discussed in section 6.5 of the last chapter.

Finally, it should be pointed out that, even given the reduction in problem size achieved, the aggregate problem and the detailed sub-problems were on the limit of the MILP software and computer hardware used. This gives an indication of the size of the original problem.

Chapter 8

Conclusions

This thesis has presented aggregate Resource-Task Network formulations for large-scale process scheduling problems with particular attention being paid to multisite systems. The key contributions of this work are summarised below. We then discuss some potential directions for future research.

8.1 Contribution of this Thesis

It is of particular significance that the aggregate formulations presented in this thesis are based on a very general process scheduling representation, namely that of Resource-Task Networks (RTNs). This allows a large number of the complicating features associated with multipurpose plant operation to be taken into account in a unified manner. These include sequence-dependent equipment change-overs, task utility requirements and limited intermediate storage. Until recently, process scheduling formulations have tended to ignore these very important features, or else they have contained complicated sets of constraints which have been tailored to specific problem types.

The RTN framework also allows wider supply chain activites to be considered in as much detail as required. For problems involving warehouses, flexible and dedicated storage areas can be modelled, as well as the fixed and variable costs associated with their operation. Distribution can also be modelled on the scale of individual vehicle loads. Important details, such as mixed loads with material incompatibilities, can also easily be modelled within the general RTN framework. Finally, on a higher level, perhaps where distribution has been contracted out to a third party, several transportation options involving different lead-times, unit costs and allowable quantities can be modelled to allow the optimal choice to be made.

Overall then, this thesis has proposed aggregate formulations of very wide applicability by basing them on the RTN concept.

The second contribution of this work is that it employs a rigorous methodology for deriving the aggregate process scheduling formulation from the detailed one. This contrasts with the small body of earlier work in this area that has used what are essentially reasonable guesses and intuitive reasoning to decide the form of the aggregate formulation. This means that the optimal solution of the aggregate formulations described in this thesis can be used

with confidence, providing rigorous upper bounds (in the case of maximisation) on the optimal values of the corresponding detailed formulations.

Finally, in contrast with other work in process scheduling, and also in the much wider field of general aggregation techniques, the aggregate formulations presented in this thesis can be made arbitrarily accurate in relation to the detailed formulation from which they are derived. This allows the trade-off between accuracy and problem size to be exploited fully. This property arises from the fact that parametrisation of the weighting functions is used to give aggregated constraints and variables of varying orders. While physical meaning can be attached to the zeroth and first order order aggregate variables, the higher order variables and constraints appear to have no obvious physical interpretation but they *do* increase the accuracy of the formulation. In the past, aggregate formulations have tended to aggregate groups of variables or constraints once only. The more abstract mathematical work has been concerned with finding optimal weightings (*e.g.* Zipkin, 1980a), whereas much work in production planning has used simple summations of groups of variables into larger entities (*e.g.* Bitran and Hax, 1977). So the formulations of this thesis have both intuitive lower order variables to enable useful results on real problems to be obtained *and* abstract higher order variables which improve the accuracy of the approximation.

It is hoped that, as in the industrial case study presented in chapter 7, aggregate formulations will give a close approximation to the true production capacity of systems involving multipurpose process plants. This is useful in itself as a decision support tool for production planning in large-scale systems. The aggregate solution can be also be used to set production targets and decompose the problem into a set of smaller sub-problems which can be solved in detail. Finding more sophisticated techniques for generating detailed schedules from the aggregate solution is one important area for further research which is discussed below.

8.2 Some Potential Directions for Further Work

The fundamental work presented in this thesis points to several interesting and important areas for future research. Three such areas are outlined below. The first relates to the use of the aggregate RTN formulation to generate good (or even optimal) solutions to the detailed RTN formulation. The second is concerned with the introduction of uncertainty into process scheduling problems. The third is the application of the aggregation techniques developed in this thesis to other types of discrete-time MILP formulations.

8.2.1 Finding detailed schedules for large-scale problems

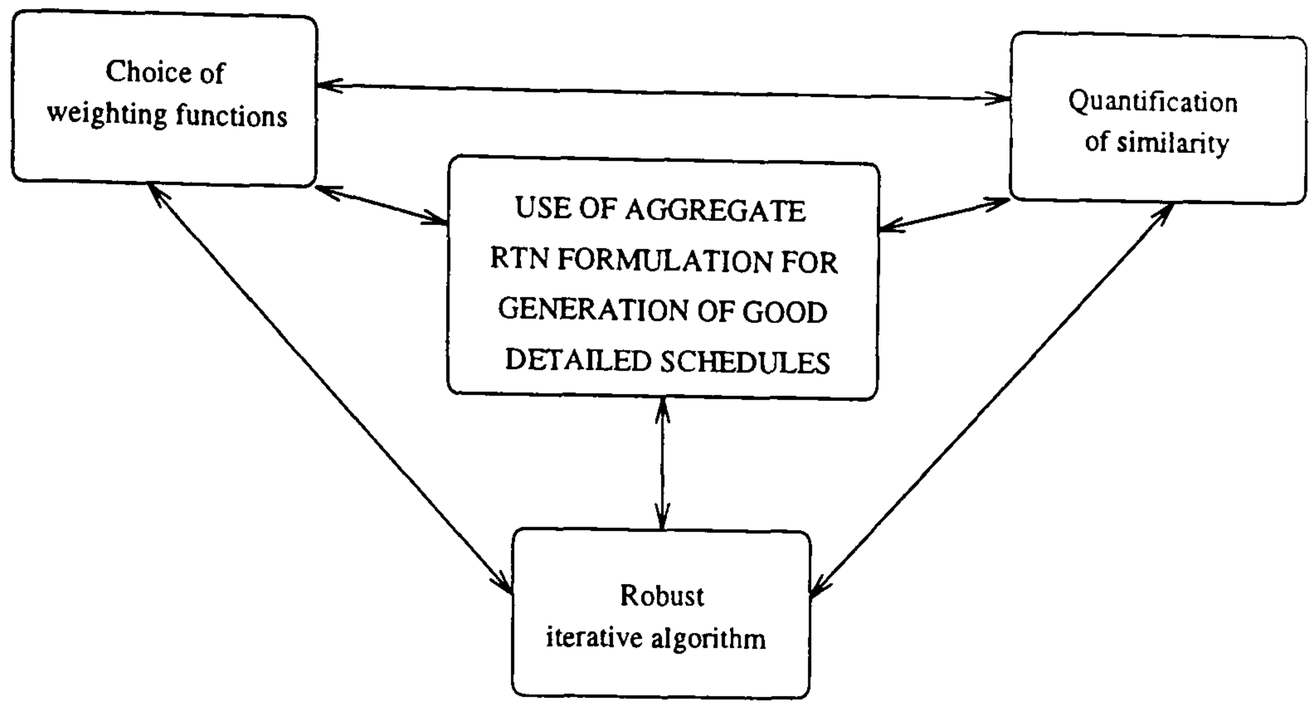


Figure 8.1: Directions for further fundamental research on aggregate RTN formulations.

As illustrated in figure 8.1, the question:

How do we use the aggregate formulation to generate optimal solutions to the original detailed problem?

leads to three inter-related areas for further research. These can summarised by the questions:

- What is the most accurate form of aggregation (weighting functions, ATP length *etc.*) to use?
- How do we best quantify this accuracy?
- Can we develop a robust iterative algorithm to generate good, feasible schedules?

The answer to the last question above is dependent on the quality of the approximation represented by the aggregate formulation. This, in turn, is likely to depend on the type of weighting functions used in the aggregation. In this thesis, simple polynomial basis functions were used. It may be possible, however, to use results from the theory of approximation of functions where alternative sets of basis functions (such as the Chebyshev polynomials) can be proved to give better accuracy than others, in general.

The choice of weighting functions also governs the nature of the aggregate variables that arise in the aggregate formulation. For the aggregation used in this thesis, we use zeroth order aggregation representing a simple summation. This generates variables with an obvious physical interpretation *e.g.* the total number of times a particular task starts. In addition, the time horizon was also split up into a number of distinct ATPs. Both these features are important in finding a robust algorithm for generating feasible detailed schedules. They allow the detailed problem to be decomposed into a number of smaller sub-problems, each

characterised by a set of meaningful quantities.

The search for 'better' weighting functions leads to another area for further research concerning the establishment of good measures for the accuracy of the aggregate formulation with respect to the detailed formulation. In this thesis, we have primarily relied on comparisons of the optimal objective functions of the two formulations. This approach appears to be reasonable, but it is a rather crude quantification of the *degree of similarity* between the optimal aggregate and detailed solutions. For example, the solutions to two aggregate formulations employing different ATP lengths may have identical optimal objective values, but that with the shorter ATP length may well be 'more similar' to the detailed solution in terms of when certain tasks are executed in the time horizon. As well as comparing optimal objective values, therefore, more sophisticated measures could be investigated (perhaps involving vector norms of the differences between the variables) which may be more pertinent in the quest for more accurate aggregations.

The general solution strategy for generating detailed schedules is another topic worthy of more study. Two possibilities (a decomposition and a rolling horizon approach) were briefly discussed in section 6.5.

8.2.2 Allowing for Uncertainty

The general process scheduling problem addressed in this thesis has been deterministic in nature. It has taken no account of uncertainty in such important parameters as the timing and magnitude of demands, raw material prices, equipment reliability *etc.*. In practice, especially for longer range planning problems, such uncertainties are likely to have a very significant effect. Two ways of dealing with uncertainty are discussed below, along with the contribution that aggregate formulations could make.

A well known way of dealing with uncertainty in mathematical programming formulations is to represent it by a set of likely scenarios and then attempt to find a robust schedule that will perform well, which ever of these possible outcomes should arise (see, for instance, Mulvey *et al.*, 1995). A potential problem with this approach is the large number of scenarios that might be required in order to adequately represent the 'uncertainty space'. However, in the context of the aggregate scheduling formulations presented in this thesis, it is interesting to note that the aggregate demand parameters $\tilde{\Pi}_{rt}^{(0)}$, $\tilde{\Pi}_{rt}^{(1)}$ *etc.* may provide an alternative and much more compact way of representing variability in the distribution of demands over each ATP. This is analogous to the characterisation of complex probability distributions by moments of various orders. It means that we have a useful method of representing the

potentially very large number of detailed demand scenarios that are required to represent future uncertainty by a smaller, more manageable number of aggregate demand scenarios.

Another way of taking into account uncertainty is to schedule in detail only for a short period of time into the future but with reference to estimates of longer-term eventualities which can be updated as new information arrives. One possible implementation of this scheme could involve a method similar to the rolling horizon approach outlined in section 6.5.2. For example, one could have a fixed long-term planning horizon of one month but only schedule in detail for the first week of this period. In other words, detailed constraints govern the first (or present) week, with aggregate constraints for the 3 future weeks. Each week, both monthly and weekly horizons are 'rolled on' and existing demand patterns could be revised as new orders are obtained or new market information arrives.

8.2.3 Wider Applications of the Aggregation Approach

For the detailed representation of large-scale process scheduling problems, the RTN formulation has several important advantages which were discussed in chapter 3. The general aggregation method, however, can be applied to any discrete-time MILP formulation containing inequality/equality constraints. Using this method on other detailed scheduling formulations (including those, perhaps, that have been specificly tailored to the particular problem they are addressing) is therefore another possible area for further research.

References

Anily, S. and A. Federgruen, "2-Echelon Distribution Systems with Vehicle-Routing Costs and Central Inventories", *Opns. Res.*, **41**, 37-47 (1993).

Axsäter, S., "Aggregation of Product Data for Hierarchical Production Planning", *Opns. Res.*, **29**, 744-756 (1981).

Axsäter, S.and H. Jonsson, "Aggregation and Disaggregation in Hierarchical Production Planning", *Eur. J. Opnl. Res.*, **17**, 338-350 (1984).

Axsäter, S., "On the Feasibility of Aggregate Production Plans", *Opns. Res.*, **34**, 796-800 (1986).

Bassett M.H, F.J. Doyle III, G.K. Kudva, J.F. Pekny, G.V. Reklaitis, S. Subrahmanyam, M.G. Zentner and D.L. Miller, "Perspectives on Model-Based Integration of Process Operations" *Proc. Fifth Int. Symp. on Process Systems Eng.*, Kyongju, Korea (1994).

Bitran, G.R. and A.C. Hax, "On the Design of Hierarchical Production Planning Systems", *Dec. Sci.*, **8**, 28-55 (1977).

Bitran, G.R., E.A. Haas and A.C. Hax, "Hierarchical Production Planning: A Single Stage System", *Opns. Res.*, **29**, 717-743 (1981).

Bitran, G.R., E.A. Haas and A.C. Hax, "Hierarchical Production Planning: A Two-Stage System", *Opns. Res.*, **30**, 232-251 (1982).

Blumenfield, D.E., Burns, L.D., Diltz, J.D., Daganzo, C.F., "Analysing Trade-Offs Between Transportation, Inventory and Production Costs on Freight Networks", *Transportation Research Part B — Methodological,* **19**, 361-380 (1985).

Bradley, G.H., "Transformation of Integer Programming Problems to Knapsack Problems", *Discrete Math.*, **1**, 29-45 (1971).

Cohen, M.A. and H.L. Lee, "Strategic Analysis of Integrated Production-Distribution Systems — Models and Methods", *Opns. Res.*, **36**, 216-218 (1988).

Collins, R.S. and R.W. Schmenner, "Taking Manufacturing Advantage of Europe's Single Market", *European Management Journal*, **13** 257-268 (1995).

Conway, R., W. Maxwell and J.O. McClain., "The Role of Work In Process Inventory in Serial Production Lines", *Opns. Res.*, **36**, 229-241 (1988).

Cooper, J., "Physical Distribution in the U.K.", *International Journal of Physical Distribution and Materials Management,* **17**, 3-47 (1987).

Daichendt, M.M. and I.E. Grossmann, "Preliminary Screening Procedure for the MINLP Synthesis of Process Systems — I. Aggregation and decomposition techniques", *Comput. chem. Engng.*, **18**, 663-677 (1994).

Das, S.K. and S.C. Sarin, "An Integrated Approach to Solving the Master Aggregate Scheduling Problem", *Int. J. Production Economics*, **34**, 164-178 (1994).

Dorairaj, P., "A Simulation Model for a Production-Distribution System", *International Journal of Physical Distribution & Materials Management,* **13**, 51-67 (1988).

Edgehill, J., Olsmats, C., Towill, D., "Industrial Case-Study on the Dynamics of a Close-Coupled Production-Distribution System", *International Journal of Production Research,* **19**, 1681-1693 (1988).

Elkamel, A., "Scheduling of Process Operations Using Mathematical Programming Techniques: Towards a Prototype Decision Support System", Ph.D thesis, Purdue University, USA. (1993).

Flaherty, M.T. Coordinating International Manufacturing and Technology. *Competition in Global Industries*, M. Porter Ed., Havard Business School Press, 83-109 (1986).

Fuller, J.B., J. O'Conor and R. Rawlinson, "Tailored Logistics: The Next Advantage", *Harvard Business Review,* **71**, 87-98 (1993).

Gabow, H.N., "On the Design and Analysis of Efficient Algorithms for Deterministic Scheduling", *Proc. Second Intl. Conf. on Foundations of Computer-Aided Proc. Des.*, Snowmass, Colorado (1983).

Garey, M.R. and D.S. Johnson, "Computers and Intractability: A Guide to the Theory of NP-Completeness", W.H. Freeman and Co. (1979).

Geoffrion, A.M., "A Priori Error Bounds for Procurement Commodity Aggregation in Logistics Planning Models", *Naval Res. Logist. Quart.*, **24**, 201-212 (1977).

Ishii, K., Takahashi, K., Muramatsu, R., "Integrated Production, Inventory and Distribution Systems", *International Journal of Production Research,* **26**, 473-482 (1988).

Karp, R.M., "Reducability among Combinatorial Problems", in R.E. Miller and J.W. Thatcher (Eds.), Complexity of Computer Computations, Plenum Press (1972).

Kondili, E., "Optimal Scheduling of Batch Chemical Processes", Ph.D thesis, University of London, U.K. (1988).

Kondili, E., C.C. Pantelides and R.W.H. Sargent, "A General Algorithm for Short-Term Scheduling of Batch Operations. Part I — Mathematical Formulation", *Comput. chem. Engng.*, **17**, 211-227 (1993).

Liu, M.L. and Sahinidis N.V., "Computational Trends and Effects of Approximations in an MILP Model for Process Planning", *Ind. Eng. Chem. Res.*, **34**, 1662-1673 (1995).

Matthews, G.B., "On the Partition of Numbers", In *Proceedings of the London Mathematical Society*, **28**, 486-490 (1897).

Mohanty, R.P. and V. Chandrasekhar, "A Computer Simulation Study for a Production-Distribution System", *International Journal of Physical Distribution & Materials Management,* **13**, 51-67 (1983).

Mulvey, J.M., R.J. Vanderbei and S.A. Zenios, "Robust Optimisation of Large-Scale Systems", *Opns. Res.*, **43**, 264-281 (1995).

Musier, R.F.H. and L.B. Evans, "Optimal Multi-Period Production and Inventory Policies for Multi-Stage Chemical Process Plants", Paper presented at AIChE Annual Meeting, Washington D.C., USA (1988).

Norman, R. and R. Ramirez, "From Value Chain to Value Constellation: Designing Interactive Strategy", *Harvard Business Review,* **71** 65-77 (1993).

Papageorgiou, L.G., "Optimal Scheduling of Multipurpose Batch/Semi-Continuous Plants", Ph.D thesis, University of London, U.K. (1994).

Pantelides, C.C., "Unified Frameworks for Optimal Process Planning and Scheduling", *Proc. Second Conf. on Foundations of Computer-Aided Operations*, Rippin D.W.T and J. Hale eds., CACHE Publications, 253-274 (1994).

Pinto J., and I.E. Grossmann, "Optimal Cyclic Scheduling of Multistage Continuous Multi-product Plants", *Comput. chem. Engng.*, **18**, 797-816 (1994).

Reklaitis, G.V., "Overview on Scheduling and Planning of Process Operations", *Proc. NATO Advanced Study Institute on Batch Processing Systems Engineering*, Antalya, Turkey, (1992).

Rippin, D.W.T., "Batch Process Systems Engineering: a Retrospective and Prospective Review", *Comput. chem. Engng.*, **17**, S1-S13 (1993).

Rogers, D.F., R.D. Plant, R.T. Wong and J.R. Evans, "Aggregation and Disaggregation Techniques and Methodology in Optimization", *Opns. Res.*, **39**, 553-582 (1991).

SD-SCICON, "*SCICONIC vs. 2.11 — User Manual*" (1990).

Sahinidis, N.V. and I.E. Grossmann, "Reformulation of Multi-Period MILP Models for Planning and Scheduling of Chemical Processes", *Comput. chem. Engng.*, **15**, 255-272 (1991).

Schilling, G. and C.C. Pantelides, "A Simple Continuous-Time Process Scheduling Formulation and a Novel Solution Algorithm", Paper presented at ESCAPE'6 Conference, Rhodes, Greece (1996).

Shah, N., C.C. Pantelides and R.W.H. Sargent, "A General Algorithm for Short-Term Scheduling of Batch Operations. Part II — Computational issues", *Comput. chem. Engng.*, **17**, 229-244 (1993).

Williams, J.F., "Heuristic Techniques for Simultaneous Scheduling of Production and Distribution in Multi-Echelon Structures", *Management Science,* **27**, 336-352 (1981).

Zhang, X., "Algorithms for Optimal Process Scheduling Using Nonlinear Models", Ph.D thesis, University of London, U.K. (1995).

Zhang, X., and R.W.H. Sargent, "The Optimal Operation of Mixed Production Facilities — A General Formulation and Some Approaches for the Solution", submitted to *Comput. chem. Engng.* (1996).

Zipkin, P.H., "Bounds on the Effect of Aggregating Variables in Linear Programs", *Opns. Res.*, **28**, 403-418 (1980a).

Zipkin, P.H., "Bounds for Row Aggregation in Linear Programming", *Opns. Res.*, **28**, 913-916 (1980b).

Zuo, M.J., W. Kuo and K.L. McRoberts, "Application of Mathematical Programming to a Large-Scale Agricultural Production and Distribution System", *Journal of the Operational Research Society,* **42**, 639-648 (1991).

Appendix A

Derivation of the General Order Aggregate Excess Resource Balance

Consider applying the Aggregation Operator to the excess resource balance (3.1) for a general $p > 0$. This results in the following:

$$\sum_{t'=t-h+1}^{t} (t-t'+1)^p R_{rt'} = \sum_{t'=t-h+1}^{t} (t-t'+1)^p R_{r,t'-1} +$$

$$\sum_{t'=t-h+1}^{t} (t-t'+1)^p \sum_{k} \sum_{\theta=0}^{\tau_k} \left(\mu_{kr\theta} N_{k,t'-\theta} + \nu_{kr\theta} \xi_{k,t'-\theta} \right)$$

$$+ \sum_{t'=t-h+1}^{t} (t-t'+1)^p \Pi_{rt'} \qquad \text{(A.1)}$$

To simplify the above expression, we proceed in a manner similar to that used for the zeroth order aggregation, re-writing it as:

$$\sum_{t'=t-h+1}^{t} (t-t'+1)^p R_{rt'} - \sum_{t'=t-h+1}^{t} (t-t'+1)^p R_{r,t'-1} =$$

$$\sum_{k} \left(\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \sum_{t'=t-h+1}^{t} (t-t'+1)^p N_{k,t'-\theta} + \sum_{\theta=0}^{\tau_k} \nu_{kr\theta} \sum_{t'=t-h+1}^{t} (t-t'+1)^p \xi_{k,t'-\theta} \right)$$

$$+ \sum_{t'=t-h+1}^{t} (t-t'+1)^p \Pi_{rt'} \qquad \text{(A.2)}$$

The second term on the left hand side of (A.2) simplifies as follows:

$$\sum_{t'=t-h+1}^{t} (t-t'+1)^p R_{r,t'-1} = \sum_{t'=t-h}^{t-1} (t-t')^p R_{rt'} = \sum_{t'=t-h}^{t} (t-t')^p R_{rt'} =$$

$$\sum_{t'=t-h+1}^{t} (t-t')^p R_{rt'} + h^p R_{r,t-h}$$

and therefore the left hand side of (A.3) can be written as:

$$\sum_{t'=t-h+1}^{t} \left((t-t'+1)^p R_{rt'} - (t-t')^p R_{rt'} \right) - h^p R_{r,t-h}$$

Consider now the quantity:

$$\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \sum_{t'=t-h+1}^{t} (t-t'+1)^p N_{k,t'-\theta}$$

which can be written as:

$$\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \sum_{t'=t-h+1-\theta}^{t-\theta} (t-t'+1-\theta)^p N_{kt'}$$

or,

$$\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \left(\sum_{t'=t-h+1-\theta}^{t-h} (t-t'+1-\theta)^p N_{kt'} + \sum_{t'=t-h+1}^{t} (t-t'+1-\theta)^p N_{kt'} - \sum_{t'=t+1-\theta}^{t} (t-t'+1-\theta)^p N_{kt'} \right)$$

$$= \sum_{\theta=1}^{\tau_k} \mu_{kr\theta} \sum_{t'=t-h+1-\theta}^{t-h} (t-t'+1-\theta)^p N_{kt'} + \sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \sum_{t'=t-h+1}^{t} (t-t'+1-\theta)^p N_{kt'}$$

$$- \sum_{\theta=1}^{\tau_k} \mu_{kr\theta} \sum_{t'=t+1-\theta}^{t} (t-t'+1-\theta)^p N_{kt'}$$

Note that the lower limit of the outer summation in the first and third terms above has been changed from $\theta = 0$ to $\theta = 1$ since the inner summation vanishes for $\theta = 0$. The inner summation of the first and third terms above can be re-written using the substitutions $\theta' = t - t' + 1 - h$ and $\theta' = t - t' + 1$ respectively to give:

$$\sum_{\theta=1}^{\tau_k} \mu_{kr\theta} \sum_{\theta'=1}^{\theta} (h+\theta'-\theta)^p \; N_{k,t-h+1-\theta'} + \left(\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \right) \sum_{t'=t-h+1}^{t} (t-t'+1-\theta)^p N_{kt'}$$

$$- \sum_{\theta=1}^{\tau_k} \mu_{kr\theta} \sum_{\theta'=1}^{\theta} (\theta'-\theta)^p \; N_{k,t+1-\theta'}$$

Finally, the order of the double summation in the first term above can be reversed to yield:

$$\sum_{\theta'=1}^{\tau_k} N_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (h+\theta'-\theta)^p \; \mu_{kr\theta} + \left(\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \right) \sum_{t'=t-h+1}^{t} (t-t'+1-\theta)^p N_{kt'}$$

$$- \sum_{\theta'=1}^{\tau_k} N_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (\theta'-\theta)^p \; \mu_{kr\theta}$$

Returning to (A.2) we can see that a similar set of manipulations can be applied to the term below:

$$\sum_{\theta=0}^{\tau_k} \nu_{kr\theta} \sum_{t'=t-h+1}^{t} \xi_{k,t'-\theta}$$

Using this and the simplified form of the left hand side of (A.2), we obtain the following:

$$\begin{aligned}
\sum_{t'=t-h+1}^{t} \Big((t-t'+1)^p - (t-t')^p\Big) R_{rt'} \; - \; h^p R_{r,t-h} \; &= \; \sum_k \left(\sum_{\theta=0}^{\tau_k} \mu_{kr\theta}\right) \sum_{t'=t-h+1}^{t} (t-t'+1-\theta)^p N_{kt'} \\
&+ \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (h+\theta'-\theta)^p \; \mu_{kr\theta} \; - \; \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (\theta'-\theta)^p \; \mu_{kr\theta} \\
&+ \sum_k \left(\sum_{\theta=0}^{\tau_k} \nu_{kr\theta}\right) \sum_{t'=t-h+1}^{t} (t-t'+1-\theta)^p \xi_{kt'} \; + \; \sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (h+\theta'-\theta)^p \; \nu_{kr\theta} \\
&- \sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (\theta'-\theta)^p \; \nu_{kr\theta} \; + \; \sum_{t'=t-h+1}^{t} (t-t'+1)^p \; \Pi_{rt'}
\end{aligned} \tag{A.3}$$

Equation (A.3) is the general order aggregated excess resource balance. It can be readily shown that it correctly reduces to the zeroth order aggregated excess resource balance (4.14) for $p = 0$. If we note that the zeroth power of any non-zero number is unity and that $0^0 = 0$, we can write (A.3) for $p = 0$ as follows:

$$\begin{aligned}
R_{rt} - R_{r,t-h} \; &= \; \sum_k \left(\sum_{\theta=0}^{\tau_k} \mu_{kr\theta}\right) \sum_{t'=t-h+1}^{t} N_{kt'} \; - \; \sum_k \sum_{\theta=1}^{\tau_k} \mu_{kr\theta} N_{k,t+1-\theta} \\
&+ \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} \mu_{kr\theta} \; - \; \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t+1-\theta'} \sum_{\theta=\theta'+1}^{\tau_k} \mu_{kr\theta} \; + \\
&\sum_k \left(\sum_{\theta=0}^{\tau_k} \nu_{kr\theta}\right) \sum_{t'=t-h+1}^{t} \xi_{kt'} \; - \; \sum_k \sum_{\theta=1}^{\tau_k} \nu_{kr\theta} \xi_{k,t+1-\theta} \; + \; \sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} \nu_{kr\theta} \\
&- \sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t+1-\theta'} \sum_{\theta=\theta'+1}^{\tau_k} \nu_{kr\theta} \; + \; \sum_{t'=t-h+1}^{t} \Pi_{rt'}
\end{aligned}$$

If terms are cancelled from the above equation, this simplifies to the following:

$$R_{rt} - R_{r,t-h} =$$

$$\sum_k \left(\sum_{\theta=0}^{\tau_k} \mu_{kr\theta} \right) \sum_{t'=t-h+1}^{t} N_{kt'} + \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} \mu_{kr\theta} - \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} \mu_{kr\theta} +$$

$$\sum_k \left(\sum_{\theta=0}^{\tau_k} \nu_{kr\theta} \right) \sum_{t'=t-h+1}^{t} \xi_{kt'} + \sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} \nu_{kr\theta} - \sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} \nu_{kr\theta}$$

$$+ \sum_{t'=t-h+1}^{t} \Pi_{rt'} \qquad \forall\ r, t = h, 2h, \ldots, H$$

This can be recognised as the zeroth order balance (4.10) written in terms of the variables from the detailed RTN formulation. Note that we have chosen to enforce it for $t = h, 2h, \ldots, H$.

Appendix B

Summary of the First and Second Order Aggregate RTN Formulations

B.1 First Order Formulation

The constraints and objective function of the first order aggregate RTN formulation were derived in chapter 4 and are summarised below:

(i) *Zeroth order aggregate excess resource balance*

$$R_{rt} = R_{r,t-h} + \sum_k \left(\tilde{\mu}_{kr0}^{(0)} \tilde{N}_{kt}^{(0)} + \tilde{\nu}_{kr0}^{(0)} \tilde{\xi}_{kt}^{(0)} \right)$$

$$+ \sum_k \sum_{\theta=1}^{\tau_k} \tilde{\mu}_{kr\theta}^{(0)} \left(N_{k,t-h+1-\theta} - N_{k,t+1-\theta} \right) + \sum_k \sum_{\theta=1}^{\tau_k} \tilde{\nu}_{kr\theta}^{(0)} \left(\xi_{k,t-h+1-\theta} - \xi_{k,t+1-\theta} \right)$$

$$+ \tilde{\Pi}_{rt}^{(0)} \qquad \forall\ r,\ t = h,\ 2h,\ \ldots,\ H \quad (4.14)$$

(ii) *First order aggregate excess resource balance*

$$\tilde{R}_{rt}^{(0)} - hR_{r,t-h} = \sum_k \tilde{\mu}_{kr0}^{(0)} \tilde{N}_{kt}^{(1)} - \sum_k \tilde{\mu}_{kr1}^{(1)} \tilde{N}_{kt}^{(0)} + \sum_k \tilde{\nu}_{kr0}^{(0)} \tilde{\xi}_{kt}^{(1)} - \sum_k \tilde{\nu}_{kr1}^{(1)} \tilde{\xi}_{kt}^{(0)} +$$

$$\sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (h + \theta' - \theta)\ \mu_{kr\theta} - \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (\theta' - \theta)\ \mu_{kr\theta} +$$

$$\sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (h + \theta' - \theta)\ \nu_{kr\theta} - \sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (\theta' - \theta)\ \nu_{kr\theta}$$

$$+ \tilde{\Pi}_{rt}^{(1)} \qquad \forall\ r,\ t = h,\ 2h,\ \ldots,\ H \quad (4.17)$$

(iii) *Linking variable excess resource capacity constraints*

$$0 \leq R_{rt} \leq R_r^{\max} \qquad \forall\ r,\ t = h,\ 2h,\ldots,\ H \tag{4.22}$$

(iv) *Zeroth order aggregate resource capacity constraints*

$$0 \leq \tilde{R}_{rt}^{(0)} - R_{rt} \leq (h-1)R_r^{\max} \qquad \forall\ r,\ t = h,\ 2h,\ldots,\ H \tag{4.24}$$

(v) *First order continuous extent non-negativity constraints*†

$$\tilde{\xi}_{kt}^{(1)} - (\tau_k + 1)\tilde{\xi}_{kt}^{(0)} + \sum_{\theta=1}^{\tau_k}(\tau_k + 1 - \theta)\xi_{k,t+1-\theta} \geq 0 \qquad \forall\ k,\ t = h,\ 2h,\ldots,\ H \tag{4.33a}$$

$$h\,\tilde{\xi}_{kt}^{(0)} - \tilde{\xi}_{kt}^{(1)} - \sum_{\theta=1}^{\tau_k}(h-\theta)\xi_{k,t+1-\theta} \geq 0 \qquad \forall\ k,\ t = h,\ 2h,\ldots,\ H \tag{4.33b}$$

(vi) *Linking variable operational constraints*

$$V_{kr}^{\min} N_{k,t+1-\theta} \leq \xi_{k,t+1-\theta} \leq V_{kr}^{\max} N_{k,t+1-\theta}$$

$$\forall\ k,\ t = h,\ 2h,\ldots,\ H,\ \theta = 1,\ldots,\tau_k,\ r \in PE_k \tag{4.35}$$

† It is assumed that $V_{kr}^{\max} > 0\ \forall\, k, r$ and consequently that the operational constraints (4.36a,b) make the inclusion of discrete extent non-negativity constraints unnecessary. If this is not the case, it will be necessary to include discrete extent non-negativity constraints of exactly the same form as the continuous constraints given by (4.33a,b).

(vii) *First order aggregate operational constraints*

$$V_{kr}^{\min}\left(\tilde{N}_{kt}^{(1)} - (\tau_k+1)\tilde{N}_{kt}^{(0)} + \sum_{\theta=1}^{\tau_k}(\tau_k+1-\theta)N_{k,t+1-\theta}\right) \leq$$

$$\tilde{\xi}_{kt}^{(1)} - (\tau_k+1)\tilde{\xi}_{kt}^{(0)} + \sum_{\theta=1}^{\tau_k}(\tau_k+1-\theta)\xi_{k,t+1-\theta} \leq$$

$$V_{kr}^{\max}\left(\tilde{N}_{kt}^{(1)} - (\tau_k+1)\tilde{N}_{kt}^{(0)} + \sum_{\theta=1}^{\tau_k}(\tau_k+1-\theta)N_{k,t+1-\theta}\right)$$

$$\forall\ k,\ t = h,\ 2h,\ldots,\ H,\ r \in PE_k \qquad (4.36a)$$

$$V_{kr}^{\min}\left(h\tilde{N}_{kt}^{(0)} - \tilde{N}_{kt}^{(1)} - \sum_{\theta=1}^{\tau_k}(h-\theta)N_{k,t+1-\theta}\right) \leq h\,\tilde{\xi}_{kt}^{(0)} - \tilde{\xi}_{kt}^{(1)} - \sum_{\theta=1}^{\tau_k}(h-\theta)\xi_{k,t+1-\theta}$$

$$\leq V_{kr}^{\max}\left(h\tilde{N}_{kt}^{(0)} - \tilde{N}_{kt}^{(1)} - \sum_{\theta=1}^{\tau_k}(h-\theta)N_{k,t+1-\theta}\right)$$

$$\forall\ k,\ t = h,\ 2h,\ldots,\ H,\ r \in PE_k \qquad (4.36b)$$

(viii) *Aggregate objective function*

$$\max \sum_{r\in MR}\left(C_r^F(R_{rH} - R_{r0}) - C_r^I \sum_{t=h,2h,..,H}\tilde{R}_{rt}^{(0)}\right) - \sum_k \sum_{t=h,2h,..,H}(C_k^F\tilde{N}_{kt}^{(0)} + C_k^V\tilde{\xi}_{kt}^{(0)})$$

$$- \sum_{t=h,2h,..,H} C_{rt}^E\tilde{\Pi}_{rt}^{(0)} \qquad (4.44)$$

B.2 Second Order Formulation

The second order excess resource balances formed by expressing the general order constraint (4.17) in terms of the appropriate aggregate variables of order $p = 0, 1, 2$ as defined by (4.8). The aggregate parameters are as defined by (4.20a,b) and (4.21). The aggregate resource capacity constraints are formed by the use the general weighting functions given by (4.42) with $m = 1$, followed by substitution with the same aggregate variables present in the second order aggregate resource balance. The extent non-negativity and operational constraints are formed by using the general weighting functions given by (4.43) for $m = 2$, again follwed by substitution with the same aggregate variables present in the second order aggregate resource balance. The linking variable constraints (4.22) and (4.35), and also the objective function (4.44) are the same as for the first order formulation.

(i) *Zeroth order aggregate excess resource balance*

$$R_{rt} = R_{r,t-h} + \sum_k \left(\tilde{\mu}_{kr0}^{(0)} \tilde{N}_{kt}^{(0)} + \tilde{\nu}_{kr0}^{(0)} \tilde{\xi}_{kt}^{(0)} \right)$$

$$+ \sum_k \sum_{\theta=1}^{\tau_k} \tilde{\mu}_{kr\theta}^{(0)} \left(N_{k,t-h+1-\theta} - N_{k,t+1-\theta} \right) + \sum_k \sum_{\theta=1}^{\tau_k} \tilde{\nu}_{kr\theta}^{(0)} \left(\xi_{k,t-h+1-\theta} - \xi_{k,t+1-\theta} \right)$$

$$+ \tilde{\Pi}_{rt}^{(0)} \qquad \forall \; r, t = h, 2h, \ldots, H \quad (4.14)$$

(ii) *First order aggregate excess resource balance*

$$\tilde{R}_{rt}^{(0)} - hR_{r,t-h} = \sum_k \tilde{\mu}_{kr0}^{(0)} \tilde{N}_{kt}^{(1)} - \sum_k \tilde{\mu}_{kr1}^{(1)} \tilde{N}_{kt}^{(0)} + \sum_k \tilde{\nu}_{kr0}^{(0)} \tilde{\xi}_{kt}^{(1)} - \sum_k \tilde{\nu}_{kr1}^{(1)} \tilde{\xi}_{kt}^{(0)} +$$

$$\sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (h + \theta' - \theta) \, \mu_{kr\theta} - \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (\theta' - \theta) \, \mu_{kr\theta} +$$

$$\sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (h + \theta' - \theta) \, \nu_{kr\theta} - \sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (\theta' - \theta) \, \nu_{kr\theta}$$

$$+ \tilde{\Pi}_{rt}^{(1)} \qquad \forall \; r, t = h, 2h, \ldots, H \quad (4.17)$$

(iii) *Second order aggregate excess resource balance*

$$2\tilde{R}_{rt}^{(1)} - \tilde{R}_{rt}^{(0)} - h^2 R_{r,t-h} = \sum_k \tilde{\mu}_{kr0}^{(0)} \tilde{N}_{kt}^{(2)} - 2\sum_k \tilde{\mu}_{kr1}^{(1)} \tilde{N}_{kt}^{(1)} + \sum_k \tilde{\mu}_{kr1}^{(2)} \tilde{N}_{kt}^{(0)}$$

$$+ \sum \tilde{\nu}_{kr0}^{(0)} \tilde{\xi}_{kt}^{(2)} - 2\sum \tilde{\nu}_{kr1}^{(1)} \tilde{\xi}_{kt}^{(1)} + \sum \tilde{\nu}_{kr1}^{(2)} \tilde{\xi}_{kt}^{(0)}$$

$$+ \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (h+\theta'-\theta)^2 \mu_{kr\theta} - \sum_k \sum_{\theta'=1}^{\tau_k} N_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (\theta'-\theta)^2 \mu_{kr\theta}$$

$$+ \sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t-h+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (h+\theta'-\theta)^2 \nu_{kr\theta} - \sum_k \sum_{\theta'=1}^{\tau_k} \xi_{k,t+1-\theta'} \sum_{\theta=\theta'}^{\tau_k} (\theta'-\theta)^2 \nu_{kr\theta}$$

$$+ \tilde{\Pi}_{rt}^{(2)} \qquad \forall\ r,\ t = h,\ 2h,\ldots,\ H \qquad \text{(B.1)}$$

(iv) *Linking variable excess resource capacity constraints*

$$0 \le R_{rt} \le R_r^{\max} \qquad \forall\ r,\ t = h,\ 2h,\ldots,\ H \qquad (4.22)$$

(v) *First order aggregate resource capacity constraints*

$$0 \le \tilde{R}_{rt}^{(1)} - 2\tilde{R}_{rt}^{(0)} + R_{rt} \le \frac{1}{2}(h-1)(h-2)R_r^{\max} \qquad \forall\ r,\ t = h,\ 2h,\ldots,\ H \qquad (4.27a)$$

$$0 \le h\tilde{R}_{rt}^{(0)} - \tilde{R}_{rt}^{(1)} - (h-1)R_{rt} \le \frac{1}{2}(h-1)(h-2)R_r^{\max} \qquad \forall\ r,\ t = h,\ 2h,\ldots,\ H \qquad (4.27b)$$

(vi) *Second order continuous extent non-negativity constraints*†

$$\tilde{\xi}_{kt}^{(2)} - (2\tau_k + 3)\tilde{\xi}_{kt}^{(1)} + (\tau_k + 1)(\tau_k + 2)\tilde{\xi}_{kt}^{(0)} - \sum_{\theta=1}^{\tau_k}[(\tau_k + 1) - \theta][(\tau_k + 2) - \theta]\xi_{k,t+1-\theta} \geq 0 \quad (4.41a)$$

$$- \tilde{\xi}_{kt}^{(2)} + (h + \tau_k + 1)\tilde{\xi}_{kt}^{(1)} - h(\tau_k + 1)\tilde{\xi}_{kt}^{(0)} + \sum_{\theta=1}^{\tau_k}[h - \theta][(\tau_k + 1) - \theta]\xi_{k,t+1-\theta} \geq 0 \quad (4.41b)$$

$$\tilde{\xi}_{kt}^{(2)} - (2h - 1)\tilde{\xi}_{kt}^{(1)} + h(h - 1)\tilde{\xi}_{kt}^{(0)} - \sum_{\theta=1}^{\tau_k}(h - \theta)(h - 1 - \theta)\xi_{k,t+1-\theta} \geq 0 \quad (4.41c)$$

$$\forall\ k,\ t = h,\ 2h,\ \ldots,\ H$$

(vii) *Linking variable operational constraints*

$$V_{kr}^{\min} N_{k,t+1-\theta} \leq \xi_{k,t+1-\theta} \leq V_{kr}^{\max} N_{k,t+1-\theta}$$

$$\forall\ k,\ t = h,\ 2h,\ \ldots,\ H,\ \theta = 1,\ \ldots,\ \tau_k,\ r \in PE_k \quad (4.35)$$

(viii) *Second order aggregate operational constraints*

$$V_{kr}^{\min}\left(\tilde{N}_{kt}^{(2)} - (2\tau_k + 3)\tilde{N}_{kt}^{(1)} + (\tau_k + 1)(\tau_k + 2)\tilde{N}_{kt}^{(0)} - \sum_{\theta=1}^{\tau_k}[(\tau_k + 1) - \theta][(\tau_k + 2) - \theta]N_{k,t+1-\theta}\right) \leq$$

$$\tilde{\xi}_{kt}^{(2)} - (2\tau_k + 3)\tilde{\xi}_{kt}^{(1)} + (\tau_k + 1)(\tau_k + 2)\tilde{\xi}_{kt}^{(0)} - \sum_{\theta=1}^{\tau_k}[(\tau_k + 1) - \theta][(\tau_k + 2) - \theta]\xi_{k,t+1-\theta} \leq$$

$$V_{kr}^{\max}\left(\tilde{N}_{kt}^{(2)} - (2\tau_k + 3)\tilde{N}_{kt}^{(1)} + (\tau_k + 1)(\tau_k + 2)\tilde{N}_{kt}^{(0)} - \sum_{\theta=1}^{\tau_k}[(\tau_k + 1) - \theta][(\tau_k + 2) - \theta]N_{k,t+1-\theta}\right)$$

$$\forall\ k,\ t = h,\ 2h,\ \ldots,\ H \quad (B.2a)$$

---

† See footnote on page 155.

$$V_{kr}^{\min}\left(- \tilde{N}_{kt}^{(2)} + (h+\tau_k+1)\tilde{N}_{kt}^{(1)} - h(\tau_k+1)\tilde{N}_{kt}^{(0)} + \sum_{\theta=1}^{\tau_k}[h-\theta][(\tau_k+1)-\theta]N_{k,t+1-\theta}\right) \leq$$

$$- \tilde{\xi}_{kt}^{(2)} + (h+\tau_k+1)\tilde{\xi}_{kt}^{(1)} - h(\tau_k+1)\tilde{\xi}_{kt}^{(0)} + \sum_{\theta=1}^{\tau_k}[h-\theta][(\tau_k+1)-\theta]\xi_{k,t+1-\theta} \leq$$

$$V_{kr}^{\max}\left(- \tilde{N}_{kt}^{(2)} + (h+\tau_k+1)\tilde{N}_{kt}^{(1)} - h(\tau_k+1)\tilde{N}_{kt}^{(0)} + \sum_{\theta=1}^{\tau_k}[h-\theta][(\tau_k+1)-\theta]N_{k,t+1-\theta}\right)$$

$$\forall\ k,\ t = h,\ 2h,\ldots,\ H \qquad \text{(B.2b)}$$

$$V_{kr}^{\min}\left(\tilde{N}_{kt}^{(2)} - (2h-1)\tilde{N}_{kt}^{(1)} + h(h-1)\tilde{N}_{kt}^{(0)} - \sum_{\theta=1}^{\tau_k}(h-\theta)(h-1-\theta)N_{k,t+1-\theta}\right) \leq$$

$$\tilde{\xi}_{kt}^{(2)} - (2h-1)\tilde{\xi}_{kt}^{(1)} + h(h-1)\tilde{\xi}_{kt}^{(0)} - \sum_{\theta=1}^{\tau_k}(h-\theta)(h-1-\theta)\xi_{k,t+1-\theta} \leq$$

$$V_{kr}^{\max}\left(\tilde{N}_{kt}^{(2)} - (2h-1)\tilde{N}_{kt}^{(1)} + h(h-1)\tilde{N}_{kt}^{(0)} - \sum_{\theta=1}^{\tau_k}(h-\theta)(h-1-\theta)N_{k,t+1-\theta}\right)$$

$$\forall\ k,\ t = h,\ 2h,\ldots,\ H \qquad \text{(B.2c)}$$

(ix) *Aggregate objective function*

$$\max \sum_{r\in MR}\left(C_r^F(R_{rH} - R_{r0}) - C_r^I \sum_{t=h,2h,..,H}\tilde{R}_{rt}^{(0)}\right) - \sum_{k}\sum_{t=h,2h,..,H}(C_k^F\tilde{N}_{kt}^{(0)} + C_k^V\tilde{\xi}_{kt}^{(0)})$$

$$- \sum_{t=h,2h,..,H} C_{rt}^E\tilde{\Pi}_{rt}^{(0)} \qquad (4.44)$$

Appendix C

A Matrix Derivation of the Aggregate Excess Resource Balance Constraints

In this Appendix, some of the ideas introduced in chapter 5 will be used to develop a general matrix form for aggregate resource balance constraints of an arbitrary order m. This provides the generalisation of the derivation given in Appendix A and is valid for any suitably chosen time-dependent weighting functions (*i.e.* any form of $\mathbf{A}^{(m,h)}$).

Consider the matrix form of the detailed resource balance constraint (3.2):

$$\begin{bmatrix} 1 & 0 & 0 & . & 0 & 0 \\ -1 & 1 & 0 & . & 0 & 0 \\ 0 & -1 & 1 & . & 0 & 0 \\ . & . & . & . & . & . \\ . & . & . & . & . & . \\ 0 & 0 & 0 & . & -1 & 1 \end{bmatrix} \begin{bmatrix} R_{r1} \\ R_{r2} \\ R_{r3} \\ . \\ R_{r,H-1} \\ R_{rH} \end{bmatrix} = \sum_k \begin{bmatrix} \mu_{kr0} & 0 & . & 0 \\ \mu_{kr1} & \mu_{kr0} & . & 0 \\ . & . & . & . \\ \mu_{kr\tau_k} & \mu_{kr\tau_k-1} & . & . \\ 0 & \mu_{kr\tau_k} & . & . \\ . & . & . & . \\ 0 & 0 & . & \mu_{kr0} \end{bmatrix} \begin{bmatrix} N_{k1} \\ N_{k2} \\ N_{k3} \\ . \\ N_{k,H-1} \\ N_{kH} \end{bmatrix}$$

$$+ \sum_k \begin{bmatrix} \nu_{kr0} & 0 & . & 0 \\ \nu_{kr1} & \nu_{kr0} & . & 0 \\ . & . & . & . \\ \nu_{kr\tau_k} & \nu_{kr\tau_k-1} & . & . \\ 0 & \nu_{kr\tau_k} & . & . \\ . & . & . & . \\ 0 & 0 & . & \nu_{kr0} \end{bmatrix} \begin{bmatrix} \xi_{k1} \\ \xi_{k2} \\ \xi_{k3} \\ . \\ \xi_{k,H-1} \\ \xi_{kH} \end{bmatrix} + \begin{bmatrix} R_{r0} + \Pi_{r1} \\ \Pi_{r2} \\ \Pi_{r3} \\ . \\ \Pi_{r,H-1} \\ \Pi_{rH} \end{bmatrix} \qquad \forall\, r \qquad \text{(C.1)}$$

We write this as:

$$\mathbf{B}\mathbf{r}_r = \sum_k \mathbf{M}_k \mathbf{n}_k + \sum_k \mathbf{U}_k \mathbf{x}_k + \mathbf{P}_r \qquad \text{(C.2)}$$

while the aggregate constraint set is given by:

$$\mathbf{A}^{(m,h)}\mathbf{B}\mathbf{r}_r = \sum_k \mathbf{A}^{(m,h)}\, \mathbf{M}_k \mathbf{n}_k + \sum_k \mathbf{A}^{(m,h)}\, \mathbf{U}_k \mathbf{x}_k + \mathbf{A}^{(m,h)}\mathbf{P}_r \qquad \text{(C.3)}$$

Here $\mathbf{A}^{(m,h)}$ is as defined in (5.5) and (5.6) of chapter 5. In (C.3), consider the term $\mathbf{A}^{(m,h)}\, \mathbf{M}_k \mathbf{n}_k$ for a task k. In the aggregate formulation, this expression is actually written in terms of aggregate and linking variables. We can represent this as follows:

$$\mathbf{A}^{(m,h)}\, \mathbf{M}_k \mathbf{n}_k \;=\; \hat{\mathbf{M}}_k \hat{\mathbf{n}}_k \qquad \text{(C.4)}$$

where $\hat{\mathbf{n}}_k$ is a vector made up of aggregate and linking variables for each ATP:

$$\hat{\mathbf{n}}_k = \begin{bmatrix} \hat{\mathbf{n}}_{k,h} \\ \hat{\mathbf{n}}_{k,2h} \\ . \\ \hat{\mathbf{n}}_{k,nh} \end{bmatrix} \in \mathbb{R}^{n(m+1+\tau_k)\times 1} \tag{C.5}$$

where:

$$\hat{\mathbf{n}}_{k,lh} \equiv \begin{bmatrix} \tilde{N}_{k,lh}^{(0)} \\ \tilde{N}_{k,lh}^{(1)} \\ . \\ \tilde{N}_{k,lh}^{(m)} \\ N_{k,lh-\tau_k+1} \\ N_{k,lh-\tau_k+2} \\ . \\ N_{k,lh} \end{bmatrix} \in \mathbb{R}^{(m+1+\tau_k)\times 1} \tag{C.6}$$

We can relate $\hat{\mathbf{n}}_k$ to the detailed task extent vector $\mathbf{n}_k$:

$$\hat{\mathbf{n}}_k = \hat{\mathbf{A}}_k^{(m,h)} \mathbf{n}_k \tag{C.7}$$

where, as defined earlier:

$$\mathbf{n}_k = \begin{bmatrix} N_{k1} \\ N_{k2} \\ . \\ N_{kH} \end{bmatrix} \in \mathbb{R}^{H\times 1}$$

and where the matrix $\hat{\mathbf{A}}_k^{(m,h)}$ has the following form:

$$\hat{\mathbf{A}}_k^{(m,h)} = \begin{bmatrix} \hat{\mathbf{S}}_k^{(m,h)} & \mathbf{0} & . & \mathbf{0} \\ \mathbf{0} & \hat{\mathbf{S}}_k^{(m,h)} & . & \mathbf{0} \\ . & . & . & . \\ \mathbf{0} & \mathbf{0} & . & \hat{\mathbf{S}}_k^{(m,h)} \end{bmatrix} \in \mathbb{R}^{n(m+1+\tau_k)\times H} \tag{C.8}$$

and the sub-matrix $\hat{\mathbf{S}}_k^{(m,h)}$ is as defined earlier in section 5.4:

$$\hat{\mathbf{S}}_k^{(m,h)} = \begin{bmatrix} 1 & 1 & . & 1 & . & 1 \\ . & . & . & . & . & 1 \\ h^m & (h-1)^m & . & (\tau_k)^m & . & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 \\ 0 & 0 & 0 & 0 & . & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} \mathbf{S}^{(m,h)} \\ \mathbf{0}^{\tau_k \times (h-\tau_k)} \; \mathbf{I}^{\tau_k \times \tau_k} \end{bmatrix} \quad \in \mathbb{R}^{(m+1+\tau_k)\times h} \tag{C.9}$$

Substituting for $\hat{\mathbf{n}}_k$ from (C.7) into (C.4) we get:

$$\mathbf{A}^{(m,h)} \mathbf{M}_k = \hat{\mathbf{M}}_k \hat{\mathbf{A}}_k^{(m,h)} \tag{C.10}$$

This gives us an expression for the matrix $\hat{\mathbf{M}}_k$ in terms of the known matrices $\mathbf{A}^{(m,h)}$, $\mathbf{M}_k$ and $\hat{\mathbf{A}}_k^{(m,h)}$. More specifically, we can write:

$$\hat{\mathbf{M}}_k = \mathbf{A}^{(m,h)} \mathbf{M}_k [\hat{\mathbf{A}}_k^{(m,h)}]^R \quad \in \mathbb{R}^{n(m+1)\times n(m+1+\tau_k)} \tag{C.11}$$

where $[\hat{\mathbf{A}}_k^{(m,h)}]^R$ is a right inverse of $\hat{\mathbf{A}}_k^{(m,h)}$ having the following form:

$$[\hat{\mathbf{A}}_k^{(m,h)}]^R = \begin{bmatrix} [\hat{\mathbf{S}}_k^{(m,h)}]^R & \mathbf{0} & . & \mathbf{0} \\ \mathbf{0} & [\hat{\mathbf{S}}_k^{(m,h)}]^R & . & \mathbf{0} \\ . & . & . & . \\ \mathbf{0} & \mathbf{0} & . & [\hat{\mathbf{S}}_k^{(m,h)}]^R \end{bmatrix} \quad \in \mathbb{R}^{H\times n(m+1+\tau_k)} \tag{C.12}$$

and the sub-matrix $[\hat{\mathbf{S}}_k^{(m,h)}]^R$ is a right inverse of $\hat{\mathbf{S}}_k^{(m,h)}$. Both $[\hat{\mathbf{S}}_k^{(m,h)}]^R$ and $\mathbf{0}$ are of dimension $\mathbb{R}^{h\times(m+1+\tau_k)}$. Hence (C.11) gives us the coefficients of the discrete extent variables in the aggregate excess resource balances.

Returning to (C.3), and considering the other aggregate terms, it can be seen that the term involving the continuous extent variables $\mathbf{A}^{(m,h)} \mathbf{U}_k \mathbf{x}_k$ is entirely equivalent to that involving the discrete extent variables $\mathbf{A}^{(m,h)} \mathbf{M}_k \mathbf{N}_k$. We can therefore immediately deduce that:

$$\mathbf{A}^{(m,h)} \mathbf{U}_k \mathbf{x}_k = \hat{\mathbf{U}}_k \hat{\mathbf{x}}_k \tag{C.13}$$

where $\hat{\mathbf{x}}_k$ is defined similarly to $\hat{\mathbf{n}}_k$ in (C.5) and $\hat{\mathbf{U}}_k$ is given by (*cf.* equation (C.11)):

$$\hat{\mathbf{U}}_k = \mathbf{A}^{(m,h)} \mathbf{U}_k [\hat{\mathbf{A}}_k^{(m,h)}]^R \tag{C.14}$$

We also note that matrix **B** in (C.2) can be viewed as a special case of $\mathbf{M}_k$ with $\tau_k = 1$, $\mu_{kr0} = +1$ and $\mu_{kr1} = -1$. We can therefore write analogous expressions which specify the aggregate resource level coefficients:

$$\mathbf{A}^{(m,h)}\,\mathbf{B}\mathbf{r}_r \;=\; \hat{\mathbf{B}}\hat{\mathbf{r}}_r \tag{C.15}$$

where $\hat{\mathbf{r}}_r$ is defined similarly to $\hat{\mathbf{n}}_k$ in (C.5) and $\hat{\mathbf{B}}$ is given by (*cf.* equation (C.11)):

$$\hat{\mathbf{B}} \;=\; \mathbf{A}^{(m,h)}\,\mathbf{B}\,[\hat{\mathbf{A}}_k^{(m,h)}]^R \tag{C.16}$$

Finally, the last term in (C.3), $\mathbf{A}^{(m,h)}\mathbf{P}_r$ needs no further simplification and represents a vector of aggregate demand parameters.

As an illustration of the general argument given above, consider the term $\mathbf{A}^{(m,h)}\,\mathbf{M}_k\mathbf{n}_k$ for a task k of duration $\tau_k = 1$ which has $\mu_{kr0} = -1$ and $\mu_{kr1} = +1$ for the resource r being considered in this example. This resource could therefore be a piece of equipment which is 'borrowed' for the duration of the task. Also, we shall use the aggregation described in section 5.3 with a planning horizon $H = 9$, ATP length $h = 3$ and order $m = 1$. We can write:

$$\mathbf{A}^{(m,h)}\,\mathbf{M}_k\mathbf{n}_k \;=$$

$$\begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 & 2 & 1 \end{bmatrix} \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 & -1 \end{bmatrix} \begin{bmatrix} N_{k1} \\ N_{k2} \\ N_{k3} \\ N_{k4} \\ N_{k5} \\ N_{k6} \\ N_{k7} \\ N_{k8} \\ N_{k9} \end{bmatrix}$$

For this example, the vector $\hat{\mathbf{n}}_k$ is given by:

$$\hat{\mathbf{n}}_k = \begin{bmatrix} \tilde{N}_{k3}^{(0)} \\ \tilde{N}_{k3}^{(1)} \\ N_{k3} \\ \tilde{N}_{k6}^{(0)} \\ \tilde{N}_{k6}^{(1)} \\ N_{k6} \\ \tilde{N}_{k9}^{(0)} \\ \tilde{N}_{k9}^{(1)} \\ N_{k9} \end{bmatrix} = \begin{bmatrix} 1 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & 2 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & 2 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 & 2 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix} \begin{bmatrix} N_{k1} \\ N_{k2} \\ N_{k3} \\ N_{k4} \\ N_{k5} \\ N_{k6} \\ N_{k7} \\ N_{k8} \\ N_{k9} \end{bmatrix}$$

In this case, $\hat{\mathbf{n}}_k$ happens to have the same dimension (H) as $\mathbf{n}_k$. This is simply because, in this case, we have chosen a sufficiently high order ($m = 1 = h - \tau_k - 1$). Using (C.5), we can write:

$$\hat{\mathbf{n}}_k \in \mathbb{R}^{n(m+1+\tau_k)\times 1} = \mathbb{R}^{nh\times 1} = \mathbb{R}^{H\times 1}$$

Continuing with the example, we recall (C.11):

$$\hat{\mathbf{M}}_k = \mathbf{A}^{(m,h)}\, \mathbf{M}_k\, [\hat{\mathbf{A}}_k^{(m,h)}]^R \tag{C.11}$$

The product of the first two matrices is:

$$\mathbf{A}^{(m,h)}\, \mathbf{M}_k = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 3 & -1 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 3 & -1 & -1 & -1 \end{bmatrix}$$

For this example, $\hat{\mathbf{S}}_k^{(m,h)}$ (and consequently $\hat{\mathbf{A}}_k^{(m,h)}$) is square since $m + 1 + \tau_k = 3 = h$. This means that, in this example, the right inverse of $[\hat{\mathbf{S}}_k^{(m,h)}]^R$ is actually a true inverse, $[\hat{\mathbf{S}}_k^{(m,h)}]^{-1}$ which is:

$$[\hat{\mathbf{S}}_k^{(m,h)}]^{-1} = \begin{bmatrix} -2 & 1 & 1 \\ 3 & -1 & -2 \\ 0 & 0 & 1 \end{bmatrix}$$

We can therefore write:

$$[\hat{\mathbf{A}}_k^{(m,h)}]^R = \begin{bmatrix} -2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & -1 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & -1 & -2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 & -1 & -2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

Finally, we can calculate $\hat{\mathbf{M}}_k$ as the following product:

$$\hat{\mathbf{M}}_k = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & -1 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 3 & -1 & -1 & -1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 3 & -1 & -1 & -1 \end{bmatrix} \begin{bmatrix} -2 & 1 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 3 & -1 & -2 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & -2 & 1 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 3 & -1 & -2 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & -2 & 1 & 1 \\ 0 & 0 & 0 & 0 & 0 & 0 & 3 & -1 & -2 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \end{bmatrix}$$

$$= \begin{bmatrix} 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 3 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 3 & -1 & 0 & 0 \end{bmatrix}$$

This matrix, therefore, gives the coefficients of the aggregate and linking variables in the zeroth and first order aggregate excess resource balances for this particular task. It pre-multiplies the vector $\hat{\mathbf{n}}_k$:

$$\hat{\mathbf{M}}_k\hat{\mathbf{n}}_k = \begin{bmatrix} 0 & 0 & -1 & 0 & 0 & 0 & 0 & 0 & 0 \\ -1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & -1 & 0 & 0 & 0 \\ 0 & 0 & 3 & -1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 1 & 0 & 0 & -1 \\ 0 & 0 & 0 & 0 & 0 & 3 & -1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \tilde{N}_{k3}^{(0)} \\ \tilde{N}_{k3}^{(1)} \\ N_{k3} \\ \tilde{N}_{k6}^{(0)} \\ \tilde{N}_{k6}^{(1)} \\ N_{k6} \\ \tilde{N}_{k9}^{(0)} \\ \tilde{N}_{k9}^{(1)} \\ N_{k9} \end{bmatrix}$$

Rembering that $\mathbf{B}$ is a special case of $\mathbf{M}_k$ with $\tau_k = 0$, $\mu_{kr0} = +1$ and $\mu_{kr1} = -1$, it can be seen that $\mathbf{B} = -\hat{\mathbf{M}}_k$ in this case. We therefore write the resource level terms in the ($h = 3$, $m = 1$ aggregate formulation as:

$$\mathbf{A}^{(m,h)}\,\mathbf{Br}_r = \hat{\mathbf{B}}\hat{\mathbf{r}}_r = \begin{bmatrix} 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 & 0 \\ 1 & 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & -1 & 0 & 0 & 1 & 0 & 0 & 0 \\ 0 & 0 & -3 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & -1 & 0 & 0 & 1 \\ 0 & 0 & 0 & 0 & 0 & -3 & 1 & 0 & 0 \end{bmatrix} \begin{bmatrix} \tilde{R}_{r3}^{(0)} \\ \tilde{R}_{r3}^{(1)} \\ R_{r3} \\ \tilde{R}_{r6}^{(0)} \\ \tilde{R}_{r6}^{(1)} \\ R_{r6} \\ \tilde{R}_{r9}^{(0)} \\ \tilde{R}_{r9}^{(1)} \\ R_{r9} \end{bmatrix}$$

It can be confirmed that are the coefficients for the aggregate resource level variables in the zeroth and first order aggregate balances given by (4.14) and (4.17) respectively, (with $h = 3$) as expected.

Appendix D

Modelling Tasks with Different Time-Scales

Consider a multi-discretised detailed RTN formulation. Each task has an associated discretisation interval d_k and we define the minimum discretisation level: $\min_k d_k = 1$. A task k, therefore, can start only at $t = 0, d_k, 2d_k, etc$. Each task has an associated duration τ'_k which is the total number of discretisation intervals (of length d_k) that it lasts for. Note that the absolute duration of the task is therefore $\tau'_k d_k$. We can write the excess resource balance as follows:

$$R_{rt} = R_{r,t-1} + \sum_{k \in S_t} \sum_{\theta=0}^{\tau'_k} \left(\mu_{kr\theta} N_{k,t-\theta d_k} + \nu_{kr\theta} \xi_{k,t-\theta d_k} \right) + \Pi_{rt} \qquad \forall\ r, t \qquad \text{(D.1)}$$

Here S_t is the set of tasks that can effect excess resource levels at time t — *i.e.* $k \in S_t \rightarrow mod(t, d_k) = 0$ — and θ is the number of time discretisations relative to the start of the task.

We now apply the p^{th} order aggregation operator to (D.1) in the usual way making sure that we sum from a time t for which S_t is full (*i.e.* contains all the tasks) back to a time $t - h + 1$ where h is such that S_{t-h} is also full. We can write:

$$\sum_{t'=t-h+1}^{t} (t - t' + 1)^p R_{rt} = \sum_{t'=t-h+1}^{t} (t - t' + 1)^p R_{r,t-1} +$$

$$\sum_{t'=t-h+1}^{t} (t - t' + 1)^p \sum_{k \in S_{t'}} \sum_{\theta=0}^{\tau'_k} \left(\mu_{kr\theta} N_{k,t-\theta d_k} + \nu_{kr\theta} \xi_{k,t-\theta d_k} \right) + \sum_{t'=t-h+1}^{t} \Pi_{rt} \qquad \text{(D.2)}$$

We now define n_k as being the number of discretisation intervals d_k that occur in one ATP $= \dfrac{h}{d_k}$. As an example, consider three tasks $k = 1, 2, 3$.

$d_1 = 1 \qquad \tau_1 = 2$

$d_2 = 3 \qquad \tau_2 = 3$

$d_3 = 5 \qquad \tau_3 = 2$

The discretisation grids for each task is represented in figure D.1. Suppose that h is chosen to be = 30. We then have $n_1 = 30$, $n_2 = 10$, $n_3 = 6$.

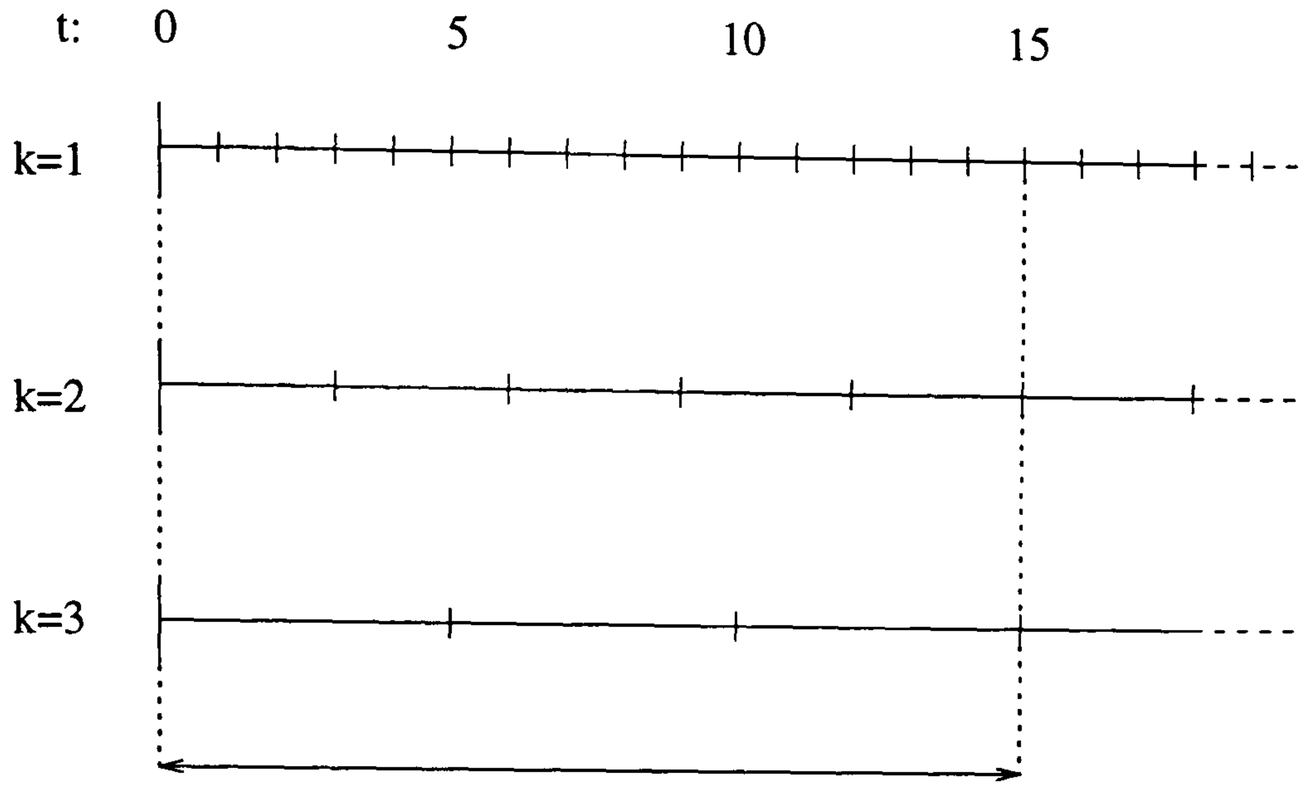


Figure D.1: Discretisation grids for the tasks in the example.

In order to simplify (D.2) we need to re-write it in a slightly different way. Consider the discrete task extent variables on the right hand side:

$$\sum_{t'=t-h+1}^{t}(t-t'+1)^p \sum_{k\in S_{t'}} \sum_{\theta=0}^{\tau'_k} \mu_{kr\theta} N_{k,t-\theta d_k}$$

Bearing in mind the above argument, it can be seen that this expression can be written in the following way:

$$\sum_{k} \sum_{\theta=0}^{\tau'_k} \mu_{kr\theta} \sum_{i=0}^{n_k-1} (id_k+1)^p N_{k,t-\theta d_k-id_k}$$

This can now be simplified in a way analogous to that described in Appendix A (in this case we shall omit the first summation sign over k for simplicity). Firstly we write it as:

$$\sum_{\theta=0}^{\tau'_k} \mu_{kr\theta} \sum_{i=\theta}^{n_k-1+\theta} [(i-\theta)d_k+1]^p N_{k,t-id_k} =$$

$$\sum_{\theta=0}^{\tau'_k} \mu_{kr\theta} \left(-\sum_{i=0}^{\theta-1} [(i-\theta)d_k+1]^p N_{k,t-id_k} + \sum_{i=0}^{n_k-1} [(i-\theta)d_k+1]^p N_{k,t-id_k} + \sum_{i=n_k}^{n_k-1+\theta} [(i-\theta)d_k+1]^p N_{k,t-id_k} \right)$$

$$= -\sum_{\theta=1}^{\tau'_k} \mu_{kr\theta} \sum_{i=0}^{\theta-1} [(i-\theta)d_k + 1]^p N_{k,t-id_k} + \sum_{\theta=0}^{\tau'_k} \mu_{kr\theta} \sum_{i=0}^{n_k-1} [(i-\theta)d_k + 1]^p N_{k,t-id_k}$$

$$+ \sum_{\theta=1}^{\tau'_k} \mu_{kr\theta} \sum_{i=n_k}^{n_k-1+\theta} [(i-\theta)d_k + 1]^p N_{k,t-id_k}$$

Note that the lower limit of the outer summation in the first and third terms above has been changed from $\theta = 0$ to $\theta = 1$ since the inner summation vanishes for $\theta = 0$.

$$= \sum_{\theta=0}^{\tau'_k} \mu_{kr\theta} \sum_{i=0}^{n_k-1} [(i-\theta)d_k + 1]^p N_{k,t-id_k} +$$

$$\sum_{\theta=1}^{\tau'_k} \mu_{kr\theta} \sum_{i=0}^{\theta-1} \left([(i-\theta)d_k + 1]^p N_{k,t-id_k} + [(i+n_k-\theta)d_k + 1]^p N_{k,t-(i+n_k)d_k} \right)$$

Finally, the order of the double summation in the first term above can be reversed to yield:

$$\sum_k \sum_{\theta=0}^{\tau'_k} \mu_{kr\theta} \sum_{i=0}^{n_k-1} (id_k + 1)^p N_{k,t-\theta d_k-id_k} = \sum_k \left(\sum_{\theta=0}^{\tau'_k} \mu_{kr\theta} \right) \sum_{i=0}^{n_k-1} [(i-\theta)d_k + 1]^p N_{k,t-id_k}$$

$$+ \sum_k \sum_{i=0}^{\tau'_k-1} \sum_{\theta=i+1}^{\tau'_k} \mu_{kr\theta} \left([(i-\theta)d_k + 1]^p N_{k,t-id_k} + [(i+n_k-\theta)d_k + 1]^p N_{k,t-(i+n_k)d_k} \right)$$

Returning to (D.2) we can see that a similar set of manipulations can be applied to the term below:

$$\sum_{\theta=0}^{\tau'_k} \nu_{kr\theta} \sum_{i=1}^{n_k} \xi_{k,t-(n_k+i-\theta)d_k}$$

Hence we can write (D.2) as follows:

$$\sum_{t'=t-h+1}^{t}\left((t-t'+1)^p-(t-t')^p\right)R_{rt'} - h^p R_{r,t-h} = \sum_k\left(\sum_{\theta=0}^{\tau'_k}\mu_{kr\theta}\right)\sum_{i=0}^{n_k-1}[(i-\theta)d_k+1]^p N_{k,t-id_k}$$

$$+\sum_k\sum_{i=0}^{\tau'_k-1}\sum_{\theta=i+1}^{\tau'_k}\mu_{kr\theta}\left([(i-\theta)d_k+1]^p N_{k,t-id_k} + [(i+n_k-\theta)d_k+1]^p N_{k,t-(i+n_k)d_k}\right)$$

$$+\sum_k\left(\sum_{\theta=0}^{\tau'_k}\nu_{kr\theta}\right)\sum_{i=0}^{n_k-1}[(i-\theta)d_k+1]^p \xi_{k,t-id_k}$$

$$+\sum_k\sum_{i=0}^{\tau'_k-1}\sum_{\theta=i+1}^{\tau'_k}\nu_{kr\theta}\left([(i-\theta)d_k+1]^p \xi_{k,t-id_k} + [(i+n_k-\theta)d_k+1]^p \xi_{k,t-(i+n_k)d_k}\right)$$

$$+\sum_{t'=t-h+1}^{t}(t-t'+1)^p\Pi_{rt'} \qquad \forall\ r,\ t=h,\ 2h,\ldots,\ H \tag{D.3}$$

We can define the following aggregated variables:

$$\tilde{N}_{kt}^{(p)} = \sum_{i=0}^{n_k-1}(id_k+1)^p N_{k,t-id_k} \qquad \& \qquad \tilde{\xi}_{kt}^{(p)} = \sum_{i=0}^{n_k-1}(id_k+1)^p \xi_{k,t-id_k}$$

It can be seen that these definitions are generalisations of the previous ones and reduce to them if $d_k = 1\ \ \forall\, k$. We can also write the general aggregation operator $A^p[\ ]$ for the multi-discretised RTN formulation as follows:

$$A^p\left[\sum_k X_{kt}\right] = \sum_k\left\{\sum_{i=0}^{n_k-1}(id_k+1)^p X_{k,t-id_k}\right\}$$

Multi-discretised aggregated resource capacity and operational constraints can also be produced in a very similar way to their singly discretised counterparts. As an example, consider the aggregation of the non-negativity continuous extent constraint from the detailed formulation. If we consider a multi-discretised horizon, we write this as:

$$\xi_{kt} \geq 0 \qquad \forall\ t = 0,\ d_k,\ 2d_k,\ldots,\ H$$

We can now sum this from a time $t-h+d_k$ $(=t-[n_k-1]d_k)$ to a time $t-\tau'_k d_k$ multiplied by any function of time that is always positive in this interval. To generate the first order constraint, for example, we can choose the following:

$$\sum_{i=\tau'_k}^{n_k-1} (i-\tau'_k) d_k \xi_{k,t-id_k} \geq 0$$

After similar simplifications to those performed in section 4.4.2, we obtain the following leads to the following aggregated constraint:

$$\sum_{i=0}^{n_k-1} (id_k+1)\xi_{k,t-id_k} - (\tau'_k d_k+1) \sum_{i=0}^{n_k-1} \xi_{k,t-id_k} + \sum_{\theta=1}^{\tau'_k} (\tau'_k+1-\theta) d_k \xi_{k,t-(\theta-1)d_k} \geq 0$$

The first two terms above can be recognised as the zeroth and first order aggregated variables respectively. It can also be verified that the above constraint reduces to (4.33a) for $d_k = 1$ as expected.

Appendix E

Constraints to Account for Start-Up Effects

The aggregate formulation may give a larger over-estimate of production capacity for RTNs which have 'start-up' times that are a significant fraction of the overall time horizon. For a given RTN, we can define the start-up time of a product as the minimum time, as measured from the start of the time horizon, in which an amount of this product can be produced. This depends upon both the characteristics of the RTN recipe and the initial storage of any intermediates. For example, the STN in figure E.1 represents a serial processing network in which a single feed is converted to a single product *via* a set of intermediates (*Int 1* to *Int 4*).

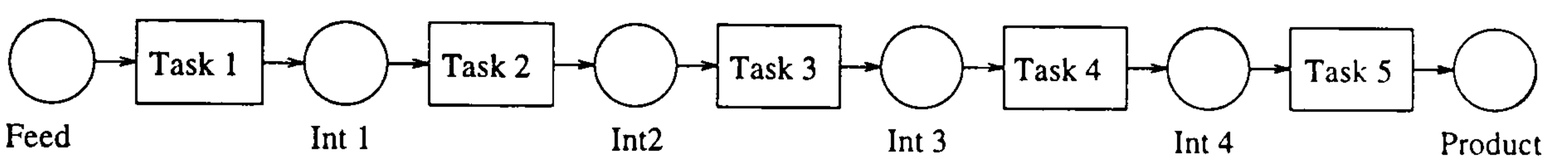


Figure E.1: STN to illustrate concept of start-up time.

If, in figure E.1, the initial storage of all intermediates is zero, it can be seen that the start-up time for this product is equal to the sum of all the task durations. However, if, for example, there is an initial storage of intermediate *Int 3*, then the start-up time for the product will be equal to the durations of tasks 4 and 5 only.

A useful concept to consider is the *maximum amount of a resource that can have been made available by a time t* — $A_{rt}^{\max}$. We can write down an expression for this quantity which can be calculated *a priori* from the initial excess resource levels and other given parameters:

$$A_{rt}^{\max} = \max_{k \in P_r \,,\, r' \in I_k \,,\, \theta = 1,..,\tau_k} \min \left\{ \frac{\sum_{\theta'=0}^{\theta} \nu_{kr\theta'}}{-\sum_{\theta'=0}^{\theta} \nu_{kr'\theta'}} A_{r',t-\theta}^{\max} \;\;,\;\; A_{r,t-\theta} + N_k^{\max} V_k^{\max} \sum_{\theta'=0}^{\theta} \nu_{kr\theta'} \right\}$$

$$+ \; R_{r0} \; + \sum_{t'=1}^{t} \Pi_{rt} \qquad \forall \; r \in materials, t \tag{E.1}$$

Here, P_r is the set of all tasks that produce resource r; I_k is the set of all resources that are consumed by a task k; and $N_k^{\max}$ is the maximum extent of task k (*i.e.* the initial number

of identical equipment items that can perform it). We shall now re-write (E.1) for the special case where tasks consume resources only at the start ($\theta = 0$) and produce them only at the end ($\theta = \tau_k$). This leads to the following:

$$A_{rt}^{\max} = \max_{k \in P_r \ , \ r' \in I_k} \min \left\{ \frac{\nu_{kr\tau_k}}{-\nu_{kr'0}} A_{r',t-\tau_k}^{\max} \ , \ A_{r,t-\tau_k} + V_k^{\max} \nu_{kr\tau_k} \right\}$$

$$+ R_{r0} + \sum_{t'=1}^{t} \Pi_{rt} \qquad \forall \ r \in materials, t \tag{E.2}$$

The two terms inside the minimisation brackets relate to the fact that $A_{rt}^{\max}$ is limited by either the maximum amount of feeds available to produce r (the first term) or the equipment capacity of the task that produces r. Once all the $A_{rt}^{\max}$ variables have been calculated, the following constraints can be included:

$$A_{rt}^{\max} + \sum_{t'=1}^{t} \sum_{k \in C_r} \sum_{\theta=0}^{\tau_k} \nu_{kr\theta} \xi_{k,t'-\theta} \geq 0 \tag{E.3}$$

Here C_r is the set of all tasks that consume resource r. The above constraint says that the amount of r that is consumed by a time t cannot exceed the maximum amount that can have been made available by that time. The constraint can be re-written more conveniently as follows:

$$A_{rt}^{\max} + \sum_{k \in C_r} \left(\sum_{\theta=0}^{\tau_k} \nu_{kr\theta} \right) \sum_{t'=1}^{t} \xi_{kt'} - \sum_{k \in C_r} \sum_{\theta=1}^{\tau_k} \left(\sum_{\theta'=\theta}^{\tau_k} \nu_{kr\theta'} \right) \xi_{k,t+1-\theta} \geq 0 \tag{E.4}$$

In terms of the aggregated formulation then, the procedure is to calculate $A_{rt}^{\max} \ \forall \ r, t = h, .., \lambda h$ (where λ is big enough to cover start-up effects — 1 or 2 may well be enough). Then constraints as given below can be added to those already in the aggregate formulation:

$$A_{rt}^{\max} + \sum_{k \in C_r} \tilde{\nu}_{kr0}^{(0)} \sum_{t'=h,\ 2h,..,t} \tilde{\xi}_{kt'}^{(0)} - \sum_{k \in C_r} \sum_{\theta=1}^{\tau_k} \tilde{\nu}_{kr\theta}^{(0)} \xi_{k,t+1-\theta} \geq 0$$

$$\forall \ r \in materials, \ t = h, .., \lambda h \tag{E.5}$$

where $\tilde{\xi}_{kt}^{(0)}$ and $\tilde{\nu}_{kr\theta}^{(0)}$ are as defined in section 4.3 by (4.11b) and (4.12a) respectively.

The *a priori* calculation of the $A_{rt}^{\max}$ parameters is straightforward for sequential RTNs since $A_{r',t-\tau_k}^{\max}$ (on the right hand side of (E.2)) will always have been already determined. In

general, however, this will not always be the case and (E.2) will give rise to a set of equations which will have to be solved to find the unknowns — namely the A_{rt}^{max} parameters.

The discussion above illustrates how some intuition can be applied to scheduling problems modelled in the RTN framework in order to develop valid constraints that can help to tighten the aggregate RTN formulation.

L.
LOUGH.
UNIV.