Periodic Radio Scheduling Example

Ville Koskinen, 48461N
Pauli Alanaatu, 48425R
Jenni Brunila, 48376C
Tuomo Pyhälä, 49560K
Vesa Timonen, 47494R

13th March 2003

Contents

1 Description of the Radio Network

Let us denote the set of *routers* by $\mathcal{M} = \{M_1, \ldots, M_N\}$ and the set of *packets* by $\mathcal{P} = \{p_1, \ldots, p_n\}$. Each packet p_i is being transmitted from the origin to the destination via certain sequence of routers $\mathcal{M}_i = \{M_{i_1}, \ldots, M_{i_k}\}$. Transmission of packet p_i from M_{i_j} to $M_{i_{j+1}}$ is called as *transmission operation* and it is denoted by O_{ij}. The durations of the operations are known in advance and they are denoted by T_{ij}.

A transmitting router interferes its neighbors and therefore we define the set of *reserved routers* $\mathcal{R}_{ij} \subseteq \mathcal{M}$ that contains the routers being reserved during the operation O_{ij}. Additionally, there are some sequence dependent setup times that need to be taken into account in the model. Namely, a routers' state transitions form "transmit" to "receive" or vice versa takes a certain setup time T_v. We denote by $M_{Tr}(O_{ij})$ and $M_{Re}(O_{ij})$ the routers that transmit and receive data during the operation O_{ij} respectively.

As a conclusion we can identify three types of constraints characterizing the set of feasible *schedules*:

- *The precedence constraints.* A packet must be transmitted via the routers in a fixed order $\mathcal{M}_i$.
- *The disjunctive constraints.* If two operations O_{ij} and $O_{i'j'}$ reserve same routers, i.e., $\mathcal{R}_{ij} \bigcap \mathcal{R}_{i'j'} \neq \emptyset$, they cannot be completed simultaneously. This implies that *either* O_{ij} precedes $O_{i'j'}$ *or* vice versa.
- *Setup constraints.*

Let us denote the time when packet p_i arrives to the network by T_i^0 and the time when it reaches the destination by T_i^f. Now the *total transmission time* of a packet can be defined as

$$D_i = T_i^f - T_i^0.$$

To offer high quality of service for each user of the network we minimize the greatest D_i, that is,

$$\min \max_{p_i} D_i. \tag{1}$$

For a broader view regarding the fairness in telecommunications, see, e.g., [9].

This formulation of the underlying problem is somewhat similar to the well known *job shop scheduling problem* (JSP) [3] if we interpret the packets as jobs consisting of operations and routers as machines completing the jobs. However, JSP in its basic form assumes that there are no setup times and performing an operation reserves only a single machine. Also, typically the completion time of the final job is minimized.

JSP model has been generalized in literature to describe real manufacturing systems. For instance, [4] have introduced how the sequence dependent setup times can be taken into account, [7] have generalized the model to include the reservation of multiple machines and in [8] it is shown how the objective function can be varied. We combine these ideas to create models for our purposes and to develop suitable solution methods for them.

2 Mixed Integer Linear Programming Model

Let us denote the *starting time* of operation O_{ij} by t_{ij} and assume that $T_{ij} = T_l$ for each O_{ij}.

1. The disjunctive constraints

$$t_{ij} + T_l \geq t_{i'j'} \text{ or } t_{i'j'} + T_l \geq t_{ij}, \forall \mathcal{R}_{ij} \bigcap \mathcal{R}_{i'j'} \neq \emptyset. \tag{2}$$

2. Setup constraints

$$t_{ij} + T_l + T_v \geq t_{i'j'} \text{ or } t_{i'j'} + T_l + T_v \geq t_{ij}, \forall M_{Tr}(O_{ij}) = M_{Re}(O_{i'j'}) \text{ or } M_{Tr}(O_{i'j'}) = M_{Re}(O_{ij}). \tag{3}$$

3. The precedence constraints

$$t_{i(j+1)} \geq t_{ij} + T_v + T_l \forall O_{ij}. \tag{4}$$

4. Packet's arrival to the network

$$t_{ij} > T_i^0. \tag{5}$$

If we introduce a variable

$$\varepsilon \geq t_{ij} - T_i^0, \tag{6}$$

our optimization problem is to $\min \varepsilon$ s.t. constraints (2)-(6).

The constraints (2) and (3), which are tricky *or* constraints, can be written in can be written in linear form by introducing integer variables. Hence, the problem can be rewritten as a mixed integer linear programming MILP-formulation. If we assume that we have a certain radio network that consists of six routers receiving and transmitting one packet, we get an MILP problem which has 204 variables and 424 inequalities. Hence, we have a problem that is unrealistic large to be solved by using traditional MILP-optimization methods, such as branch and bound method.

3 Disjunctive Graph Representation

The schedules can be presented as a *graph*, see figure 2 below representing transmission of two packets as depicted in 1. The graph presents each transmission operation O_{ij} as a *node* that are aligned such that each row corresponds to a packet. You can imagine that each node contains a label that shows the set of reserved routers $\mathcal{R}_{ij}$ and the durations T_{ij} explicitly. The durations are called the *weights* of the nodes.

The *arcs* present the order in which the operations are performed, i.e., an *directed* arc $(O_{ij}, O_{i'j'})$ indicates that the operation O_{ij} is completed before $O_{i'j'}$. Hence, the solid arcs correspond to the precedence constraints. The slashed arcs present the disjunctive constraints and they are referred to as *disjunctive* arcs. Initially, the disjunctive arcs are undirected and basically our problem is to choose their directions optimally.

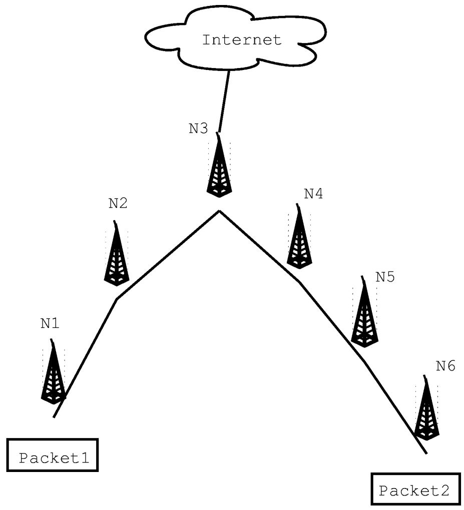

Figure 1: The radio network

The setup times can also be included in this graph representation by setting an additional weight T_v for an arc $(O_{ij}, O_{i'j'})$ if a setup is needed between operations O_{ij} and $O_{i'j'}$.

Let us now assume that we have one schedule, i.e., we have all the disjunctive arcs directed, as shown in figure 2. Consider a *path* $P = \{O_{i_1j_1}, \ldots, O_{i_nj_n}\}$ in the graph and define the *length of a path* as the sum of the weights of the nodes and arcs on the path. Hence, the length of a path can be interpreted as a lower bound of the time lag between the operations $O_{i_1j_1}$ and $O_{i_nj_n}$. This is because the directed arcs give the completion order of the operations.

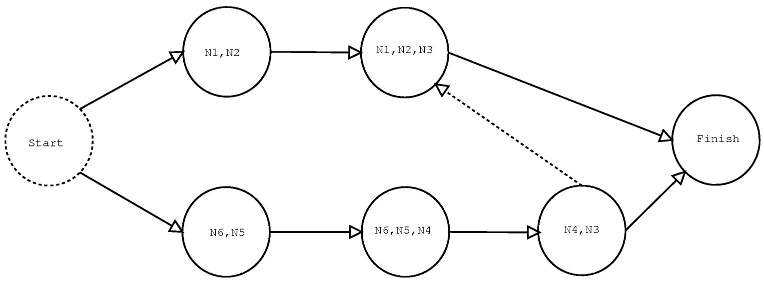

Figure 2: An example of a schedule

This representation allows us to determine the total transmission time for each packet p_i. Choose the operations O_{i1} and $O_{ij_{\max}}$ that is the last operation for packet p_i. Now we can calculate a *critical path* for p_i, that is the longest path from O_{i1} to $O_{ij_{\max}}$. Its length equals the total transmission time of the packet and hence we are able to calculate the transport delays of the packets for the schedule.

Note that the schedule described by a directed graph is feasible provided that the graph is *acyclic*, i.e., there are no cycles in the graph. This result can be shown easily by assuming the controversy. Suppose that there is a cycle $\{O_{ij}, O_{i'j'}, \ldots, O_{ij}\}$, which implies that the operation O_{ij} needs to be completed before O_{ij}.

A real data network typically consists of 10 to 50 routers and there are hundreds of packets being

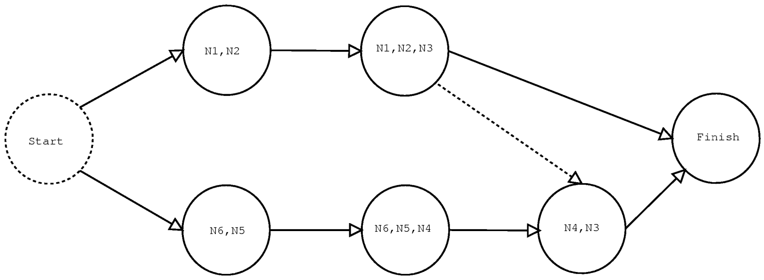

Figure 3: An example of an improved schedule

transmitted. This problem is so a large that we aim to use heuristic *local search* procedures to solve it. It has been shown in many studies that *taboo search* is the most suitable search procedure for this type of problems, see, e.g., [1], and hence we take it as the basis of our solution methodology. The basic idea in taboo search is to define a *neighborhood* for each schedule and iteratively search locally for the best schedule in this neighborhood and thus gradually reach the optimum. The figures 2 and 3, where a disjunctive arc is redericted, give a simple illustration on how a schedule can be locally improved.

The taboo search involves longest path subproblems when evaluating the objective function $\max D_i$. They can be solved by transforming the problem as the shortest path problem by switching the signs of the weights of the arcs and nodes. Hence, it is possible to apply Bellmann-Ford algorithm [2] that is implemented, e.g., by the Goblin library.

There are many computational examples on applying taboo search in job shop scheduling. Nevertheless, those problems involve only tens of machines and tens of jobs. Especially in [7] there are examples on job shop scheduling problem with reservation of multiple resources; those problems involve 10 resources and 30 jobs each consisting of 5 to 10 operations and reserving 1 to 3 resources. Evidently, this size of problem is too small for our purposes but it offers a pessimistic lower bound estimation for the size of problem that can be solved close to exact optimium.

4 Further Research

This graph model can be, in fact, interpreted as a project scheduling problem [6] besides the generalized JSP. So far, we do not have familiarized ourselves with the project scheduling literature and hence we first solve the directed graph by presenting the methods suggested in JSP literature.

Another interesting pattern in this problem is that if we assume constant periodic non-bursty traffic in the network the optimal schedule will be periodic too. Periodic solutions are described in literature, see, e.g., [5]. Nevertheless, those models seem to be even more difficult to solve than the LP-formulation given in Chapter 2. Hence, we do not have addressed this aspect seriously so far. This could be an interesting area for future research as well if there are enough resources available at the end of the project.

The project is in schedule. The literature review is completed and the implementation of the taboo search can be readily initiated. TP takes the responsibility of the implementation with VT.

VK responds for the further development of the model. PA and JB grasp with the literature too but later, they can take part in the implementation too if necessary.

The identified risks remain unchanged: $\mathcal{NP}$-completeness is the major concern.

References

[1] Edward Anderson, Celia Glass, and Chris Potts. Machine scheduling. In *Aarts, E. and J.K. Lenstra (eds.). Local Search in Combinatorial Optimization.* John Wiley & Sons, 1997.

[2] Dimitri Bertsekas. *Network Optimization: Continuous and Discrete Models.* Athena Scientific, 1998.

[3] Dimitri Bertsimas and John Tsitsiklis. *Introduction to Linear Optimization.* Athena Scientific, 1997.

[4] In-Chan Choi and Osman Korkmaz. Job shop scheduling with sequence-dependent setup times. In *Proceedings of INFORMS Conference on Information Systems & Technology*, Washington D.C., USA, May 1996.

[5] Philippe Chrétienne, Edward Coffmann, Jan Lenstra, and Zhen Liu. *Scheduling Theory and its Applications.* John Wiley & Sons, 1997.

[6] Salah Elmaghraby. *Activity Networks Project Planning and Control by Network Models.* John Wiley & Sons, 1977.

[7] Yazid Mati, Rezg Nidhal, and Xiaolan Xieaolan. A taboo search approach for deadlock-free scheduling of automated manufacturing systems. *Journal of Intelligent Manufacturing*, 12(5):535–552, November 2001.

[8] Cintia Scrich, Vinícius Armentano, and Manuel Laguna. Tardiness minimization in a flexible job shop: a tabu search approach. Journal of Intelligent Manufacturing, To Appear.

[9] Vesa Timonen. Static fairness criteria in telecommunications. Special assignment, Networking Laboratory, Helsinki University of Technology, 2002.