Flexible dynamic scheduling in public transport: environmental benefits of a more efficient system

A. Marqués, M. Torregrosa, A. Camarena, K. Darby-Dowman, J. Little, S. Moody
Eletronic Traffic, S.A., Valencia, Spain
Department of Mathematics and Statistics, Brunel University, UK

Abstract

SUPERBUS is a 30 months R&D project funded by the European Union under the ESPRIT programme. The project to develop a Dynamic Scheduling System for public transport (PT) started in November 1993 with an European multidisciplinary consortium involving PT operators. The aim of a conventional PT scheduling system is to produce a schedule consisting of timetables and the associated allocations of buses and crews given a set of available resources (buses, crews) and the desired level of service (bus frequencies). The SUPERBUS system will offer additional features of flexibility and integration.

1. Introduction

Urban transport is not only an important economic and social issue but also an increasing source of pollution. In order to avoid excessive environmental damage it is necessary to promote a modal shift from private to public transport (PT). This can only be done if PT is attractive and efficient.

Flexibility is needed to cope with evolving requirements for PT operators. Integration with Automatic Vehicle Location (AVL) systems offers major improvements in the ability to respond to incidents and unexpected events that disrupt a planned schedule. The system can carry out a dynamic or reactive rescheduling process so that the situation may be assessed and actions recommended in real time. The aim is to minimise disruption by returning to an orderly level of service as quickly as possible.

The underlying technology in the SUPERBUS system is a combination of advanced optimisation tools, constraints logic programming and the object oriented paradigm.

2. Initial Planning & Scheduling

The initial bus & crew scheduling module of the Superbus system provides one of the main modules within the application and is accessed through its user interface. This
provides the facility to specify the inputs graphically. The menu items and buttons provide the interface for the user to initiate the processes comprising bus and crew scheduling.

The initial bus & crew scheduling module provides the scheduler with the tools to create a timetable, allocate a set of buses to it, allocate crews to be responsible for the buses and finally create the duty roster for the crews. The problem is defined in terms of a number of specified urban transport characteristics such as, service levels and journey patterns for different periods of the day and types of day. The problem, due to its complexity, is not usually solved as a whole, but broken down into inter-related sub-problems, each with its own characteristic method of solution. These sub-problems are timetabling, bus scheduling, crew scheduling and rostering and are usually solved in a sequential manner.

The initial input for timetabling includes a set of lines with required services levels and journey patterns, from which a timetable is created. The timetabling process should also take into account any synchronisation within or between lines which will take place. This input is available for every different day type (e.g. weekday, Saturday, Sunday, Public Holiday) which requires a different timetable. The timetabling process is carried out using specification techniques from Constraint Logic Programming (CLP). This involves dividing the whole problem into a series of independent sub-problems and then deriving optimal transitions between different time periods of the day. These sub-problems are solved using constraints and combinatorial search techniques. The final timetable is created by placing the pre-computed sections, obtained from the solution of the sub problems, in sequence. From this, sets of vehicle blocks are constructed. Vehicle blocks represent pieces of work one vehicle does between leaving and arriving back at a garage or parking point.

This set of vehicle blocks are then linked together during the bus scheduling process to form a complete work program for a bus on a day. The bus scheduling process allocates (logical) vehicles to one or more vehicle blocks. The maximum number of overlapping vehicle blocks provides an upper limit on the number of vehicles required to cover the timetable. However, more vehicles may be required. For example, if certain lines (and hence vehicle blocks) require particular types of vehicle (e.g. single, double-deckers, mini, articulated) due to physical restrictions on the route (such as low bridges) or due to the estimated demand for that particular time of day then perhaps more buses will be needed. Once these logical vehicles have been allocated to vehicle blocks, then actual buses can be allocated on a daily basis by the depot manager (bus rostering).

The resulting bus schedules provide the basic input to the crew scheduling part of the system and represent the tasks to be covered by the crews i.e. the vehicle blocks represent the times at which a crew must be responsible for a bus. The crew scheduling process involves the generation of a set of feasible duties from the tasks to be covered within the constraints such as legal shift patterns, recommended practices etc. These feasible duties contain information on periods of driving, driver transfers, clocking on & off, cash accounting and breaks. Mathematical Programming (MP) techniques are used to select the necessary duties to cover the set of required tasks at minimum cost. The model used here is a set covering model with under and over cover penalties which provides more flexibility in the types of schedules obtained. Crew rostering is performed by ‘fairly’ distributing these crew duties amongst a set of real crews.
Throughout the analysis and design phase of the project an Object Oriented approach following the Rumbaugh methodology has provided a framework for the whole system and an OO database will be used to support the system. Combining these advanced technologies of CLP, MP and the OO approach enables more complex problems to be handled. For example synchronization is considered resulting in smoother connections between services and therefore more attractive service for the customer. This approach also enables more flexibility for the evolving needs of the public transport operators.

Initial bus & crew scheduling provides a prescribed schedule which is used at the start of each day as the planned programme of tasks to be carried out. Deviations from this prescribed schedule are monitored by the automatic regulation module of the system. Incidents are reported as soon as they are detected together with guidance to the possible corrective action. Consequently the user is able to take action earlier on without waiting until this incident accumulates into a more severe incident. Therefore many of the problems that arise will be less disruptive to the transport system providing a more efficient and effective service. The major environmental impact however results from the automatic regulation and rescheduling components of the SuperbuS system.

3. Real-Time Control

The real-time control system of SuperbuS complements the planning and scheduling tools presented in section 2. Real-time functionalities such as incident detection and corrective actions are undertaken by each of its two modules: Incident Detection and Automatic Regulation.

The incidents are processed in the Incident Detection module. The consequences on the schedule, the scope and the relation to other incidents are identified. An overall severity evaluation is presented to the operator for validation or cancellation. If the operator decides to cancel the incident it is stored in a historical database. If the operator decides to tackle the incident, then it must be solved by the Automatic Regulation module. This module determines whether the incident can be solved by the Tactical Solver ("soft" incident) or should be attempted at the On-line Rescheduling ("hard" incident). In the first case, a list of tactical solutions will be offered to the operator through the MMI for validation or cancellation. In the second case, the real-time system will request an adjustment of the initial schedule by the Re-scheduling module. The operator may also decide to reschedule if the list of tactical solutions offered is not satisfactory.

The Incident Detection and Automatic Regulation modules of the real-time system provide important tools for sparing the operator from repetitive control actions. The decision-making tasks are simplified but still under the operator responsibility. Within the SuperbuS application, the Incident Detection module represents the interface between the real world and the regulation functions, whereas the Automatic Regulation module alleviates the On-line rescheduling of processing small but probably frequent tasks.

3.1. Automatic Incident Detection

The extent of the automatic incident detection process depends on whether the public transport operator (PTO) has an automatic location system for the buses or not, and whether it is connected to a traffic control system. A monitoring system can provide
SuperbuS automatically with information concerning the position and load of the vehicles. A traffic control system can give information regarding traffic conditions and road work. Comparing these data with the state expected form the planned schedule, the Incident Detection module can detect position and load deviations of buses and predict running time changes due to traffic load changes. Generally, the Incident Detection module will receive incident information manually through the MMI. PTO experience shows that important incidents occur unpredictably and are transmitted to the operator by radio or other means (telephone, police, etc.).

Four different incident categories have been defined according to their causes. Any incident is caused either by the buses, the traffic conditions, the drivers or the load:

- **B** **Bus:** the incident is caused by or affects to a bus or set of buses
- **T** **Traffic:** the incident is caused by the traffic conditions
- **D** **Driver:** the incident is caused by or affects to one or several drivers
- **L** **Load:** the incident is caused by or affects to the balance between the offer conditions and the actual demand conditions

Within SuperbuS system there are three methods for detecting or input incidents:

- **Automatic Detection:** Comparing this real-time information (such as buses positions, demand or road traffic status) with the one foreseen in the actual schedule, it is possible to detect discrepancies. These discrepancies are considered incidents that need to be managed.

- **Human Detection:** All the incidents that can not be detected in an automatic way by the system and that therefore must be introduced in the system manually by the human operator.

- **Incident Prediction:** With incident prediction we mean the ability of detecting incidents that will probably occur in the future if the things evolve as expected. SuperbuS will be able to detect these kind of situations because every time a new incident or event is introduced all the causal consequences of this event are calculated according to the information present in the system.

When the incident has been introduced in the system the following step consists in searching the possible consequences produced by the incident. This means to check the logical or causal connections between the disruption caused by the incident and the rest of the schedule for the day. We identify each consequence with one Consequence object, that is associated to the original incident.

SuperbuS system will be able to manage several incidents at the same time. For example, it is possible that while the solver is searching for solutions for an incident a new incident is input to the system. As we can have in our system several incidents at the same time evolving independently, we need to establish mechanisms for relating the incidents that are dependent, and avoid to treat incidents that are related as if they were independent (this could lead to incoherent or contradictory system answers). In order to avoid this the real-time system checks if the incoming incidents are related or not. If the incidents are related, they will be grouped in a single unit and in the future will be treated as a unit.

The analysis of the incident finalizes calculating the severity of the incident and all their related consequences. The severity can be calculated according to different criteria that can be selected by the operator, such as punctuality, regularity, disruption respect the original schedule, etc. This severity will be used by the operator as a help at the moment of deciding if the incident must be passed to the Automatic Regulation module for searching for solutions, or if it must be discarded from the system.
3.2. Automatic Regulation

The approximation taken to make real-time control is based on two methods for solving different kind of problems:

- The **Tactical Solver**: is able to propose standard solutions to standard problems well identified by experienced operators.
- The **On-line Rescheduling**: general procedure for solving big problems not identified by the Tactical Solver (see point 4.2).

The control system will be able to provide automatic solutions to any incident problem presented in real time. Additionally, in order to make the system more reliable and to supervise and validate the proposed solutions, the human intervention is also possible for three aspects:

- To guide the solution search process (deciding strategies, limiting scope, etc.).
- To amend the automatic solutions.
- To validate the solutions.

When an incident group has to be solved the first place where it will look for solutions is in the tactical solver. When the problem is presented, the tactical solver will check if the incident and consequences match one (or more) of the “incident patterns” defined. Each pattern is also associated to a standard solution that can vary depending of certain parameters, available resources, etc. Incident patterns must be defined taking into account not only the original incident but also its consequences. Examples of incident patterns are: general delay in one line, bus gap, bus bundling, etc.

Together with each solution proposed by the system and independently if it is a tactical solution or a rescheduling, it will also be provided a cost-benefit evaluation of the solution. If the controller decides to apply the solution, this will imply to send the corresponding orders to the buses and drivers, and to update the internal data structures to reflect the new situation.

4. Rescheduling

Rescheduling within the SuperbuS application provides tools to the scheduler and network controller to carry out some form of scheduling after an initial timetable, bus schedule and crew schedule have been produced. For the scheduler, the tools allow schedules to be produced, based to some extent on an existing schedule. This allows the scheduler not only to amend an existing schedule, but also to answer ‘what-if’ questions. For the controller, the rescheduling tools allow different responses to real time incidents on the network to be evaluated and solutions proposed based on the current schedule.

The two types of rescheduling are often called off-line and on-line respectively because of the circumstances under which they take place.

For clarity, the relationship between the initial schedule and current schedule is that the initial schedule is seen as the programme of tasks at the beginning of the day. The current schedule is a reflection of the partially completed set of tasks at a point during the day. The Schedule is defined as being made up of a timetable, bus schedule and crew schedule.
4.1. Off-Line Rescheduling

Off-line rescheduling provides the scheduler with the opportunity to construct a new timetable or crew schedule using, if desired, an already completed one. A relevant scenario for its use is the requirement for rescheduling because of a future ‘incident’ or problem on the network. The objective is to produce a ‘new’ schedule trying to minimise the amount of disruption to the passenger timetable and crew schedule, while at the same time ensuring that the service level remains acceptable. Certainly, if incidents are handled inappropriately or not at all, they can lead to major economic and environmental costs. For example, not taking into account increasing running times can result in a decreasing frequency of buses, the passenger timetable unexpectedly becoming invalid, fuller buses, buses becoming ‘bunched’ together, and reduced layover times affecting crew morale.

Given that there are many common examples of incidents requiring rescheduling, it must be viewed as an essential part of fleet control; making a major contribution to the performance of the company in its service to the community.

The Rescheduling module in SuperbuS provides the scheduler with the tools to reduce the overall effect that a predicted incident might have on the running of the urban transport network.

Four techniques of rescheduling are identified which can be employed for all types of incidents requiring rescheduling. They are:

1. **Initial Scheduling** means to essentially re-run the timetabling, bus or crew scheduling algorithms either locally on a line, or globally on a set of lines. If buses were scheduled with no interlining, then it is likely that only one line would need to be bus scheduled if changes to the underlying line service timetable were made. Consequently, the amount of rescheduling of the crew would also depend on the level of crew interlining. If both bus and crew scheduling had been done on a per line basis then only those parts of the schedule relating to the line(s) which had been changed should be rescheduled. This same principle is true if sets of lines were bus or crew scheduled together i.e. re-schedule only those parts directly affected. Obviously, if there had been complete interlining of crews and buses, then complete re-initial scheduling would be required under this type of rescheduling.

2. **Fix and Schedule** means to use some part of the initial timetable or crew schedule as the basis for rescheduling. This can be viewed as freezing part of the schedule and rescheduling using it as a basis. There are two aspects which can be frozen, the timetable and the crew duties. With timetabling a part of the day may be fixed. Since timetabling is done line by line, the rescheduling is similar to the initial timetabling, but with additional synchronizations from the fixed parts of the timetable. Fixing certain duties effectively removes those columns from the allocation matrix in the crew selection/optimisation process. It will also remove other potential duties which also cover those tasks now ‘covered’. It then remains to use existing generated duties or generate new ones and use these on the reduced optimisation problem.

3. **Editing** allows the scheduler to easily change a timetable or schedule graphically. It is often the case that the scheduler can accept, in a limited number of circumstances, certain constraints to be broken or knows of local improvements. These cases are exceptional and are not expected to be predictable from the software.

4. **Local Scheduling** is concerned with small changes to the schedules made at an individual vehicle journey level. For example, to compensate for extra driving time required due to some ‘incident’, it may be possible to use some of the flexibility in the
timetable or crew schedule. This flexibility in timetabling lies in the difference between actual and minimum layovers. It is also present during pauses or waiting times when a crew is responsible for the bus, but it is not actually making any physical progress. For the crew schedule, the flexibility is gained by using time allocated for non-driving tasks.

4.2. On-Line Rescheduling

On-line rescheduling is intended to be a general purpose solving tool for problems arisen in real-time. This kind of rescheduling can of course be minimised, but never removed, by providing robust initial schedules with adequate ‘float’. For example, during the layover or waiting time activities, time can be ‘gained’ in the event of disruption. Therefore a schedule can be produced which needs little controller alteration and still provides the required level of service. This however may use too much resource to achieve a schedule with enough ‘float’. Conversely, if layover times are low, greater utilisation of the vehicles is made, but delays to the published timetable are more likely and the quality of the service goes down. The scheduler in practice accepts a certain level of risk within his schedule to accommodate the conflicting goals of minimising resources against providing the expected level of service. Rescheduling in itself can be quite costly in terms of overtime, extra journeys as well as driver morale and customer satisfaction.

The process of on-line rescheduling may appear to be similar to initial scheduling except for four important differences which transform the problem. They are,

1. The time available to perform the rescheduling is very short, of the order of minutes rather than hours.
2. The multi-objective nature of the problem with different objectives such as reduction of disruption, keeping service levels within a temporary acceptable range and minimising cost of crews and vehicles.
3. The schedule has been implemented up to some point in time and only the remainder of the schedule has to be reconsidered in relation to the current status.
4. One of the assumptions on which the original schedule was built is no longer valid. The schedule may need to be altered for the future, as well as resolving the incident at the moment. e.g. durations across some timing links may have changed for the remainder of the day.

The rescheduling process consists of several stages. The operator has control in each of them: he has to decide on the different options, limit the ranges of the parameters, adapt the constraints, etc. The main steps of the process are:

1) Graphical presentation of the related incidents and their consequences.
2) Selection of the solving strategy.
3) Limiting the scope of the possible solution (i.e., defining which part of the schedule can be changed, and which one must remain unaltered).
4) Constraint relaxation or reinforcement. Possibility of tuning constraint parameters to cope with the situation.
5) Start the solution search.
6) When a solution (or a set of them) is found it is presented to the operator together with its cost-benefit analysis. The user has the possibility of graphically amend the solution.
5. Conclusions

The use of the system described in this paper can help to create a more efficient PT system that provides greater customer satisfaction. The result can be one in which there is a reduction in private transport in favour of PT and a corresponding reduction in congestion.

Also, the application of the optimization tools provided with SuperbuS for planification of the public transport services can result in a decreasing of the number of buses needed to cope with a given demand.

Both aspects provide that the potential environmental benefits are significant in most large cities.

References

[1] SuperbuS Consortium: SuperbuS Project Deliverables; R+D ESPRIT Project 8742
[3] Bankovic, Radovan: Méthodes modernes de détermination des besoins et d’établissement des horaires dans les services de transport public; Université de Belgrade (Yugoslavie)