University of Leeds

SCHOOL OF COMPUTER STUDIES

RESEARCH REPORT SERIES

Report 98.21

Installing an Urban Transport Scheduling System

by

Anthony Wren & Raymond S K Kwan

October 1998

Summary

Bus and driver scheduling methods developed by the authors and their associates form the basis of the leading British commercial scheduling system installed for many organisations throughout the 1980s. New research in the 1990s led to more advanced systems. This paper describes the transition from these research-based systems which had been demonstrated to operating companies through to the implementation of a fully operational system for an urban bus company. After outlining the bases of the scheduling methodology, the paper works through the stages of demonstrating and developing a system to meet the real needs of a client while under considerable pressure of time. It is shown that substantial savings were achieved, and that a good scheduling system can assist in the evaluation of alternative operating arrangements.

KEY WORDS: bus; bus driver; scheduling; implementation

1. Introduction

The purpose of this paper is to present an account of some of the practical difficulties encountered in implementing solutions to a common transport scheduling problem, and to show how these difficulties may be overcome so that genuine savings are achieved through a new scheduling system.

The authors are part of a team which has been working intensively in transport scheduling since 1961. The world's first operational rail locomotive scheduling system was installed by one of us for British Railways in 1963 [14], saving three locomotives on its first application. A bus scheduling system which became known as VAMPIRES was developed in the early 1970s [15], installed for Greater Manchester Transport in 1975, and further developed within the BUSMAN commercial package [17]. A major bus driver scheduling system, IMPACS, was installed in London Transport in 1984 [12], and is still used by London successor companies; it was subsequently incorporated in BUSMAN. Various methods for constructing rotating rosters of bus drivers have been developed for individual users, with a generic version included in BUSMAN.

Throughout the 1980s the team had worked in partnership with a commercial body to develop and extend BUSMAN and to instal it for some thirty to forty bus and light rail organisations. However, our original partners disposed of their interest in 1992, and our own involvement with BUSMAN ceased in 1993.

In 1990 we started to work with models for rail driver scheduling, while from 1994 a new driver scheduling system applicable to both rail and bus was developed [4,5,6], in parallel with an object-oriented bus scheduling system [8]. The new driver scheduling system was successfully applied to many rail problems over the period 1995-96, and subsequently demonstrated to bus companies.

In 1997 we entered into agreements with two new commercial partners for the exploitation of the new systems for bus and rail respectively, and the first contract for the installation of a complete scheduling system for a bus operating company was agreed towards the end of that year. This paper describes the transition from a set of research oriented scheduling algorithms to a fully operational scheduling system, discussing several of the practical problems met and overcome. The algorithms used have been presented elsewhere, and are only summarised here.

2. Background

Computer scheduling of public transport has been the subject of a series of seven international conferences since 1975 [1,2,3,9,11,13,16]. While there are several commercial systems whose methods have not been fully published, the most widely exposed have been the HASTUS system from Montreal [10] and the BUSMAN system already mentioned. The authors' newest work was described at the most recent of these conferences, including the bus scheduling system [8], together with driver scheduling presented from a number of viewpoints [4,6,7] and a study of practical combinations of vehicle and driver scheduling [18].

We now sketch the methods of the bus and driver scheduling systems used in the installation to be presented here before outlining the way in which they were to be brought together in a commercial system.

Bus scheduling may be defined as the allocation of vehicles to previously defined journeys. This requires the linking of individual journeys into *blocks* which include all the work done by a vehicle between leaving and returning to the garage. Some vehicles may work more than one block in a day, depending on the nature of the operational peaks. Usually the prime objective is to minimise the number of vehicles (which implicitly usually also minimises the demand for drivers). To this end, a good bus scheduling system should suggest to the user how vehicles might be saved by revising certain critical journeys.

A vehicle block normally consists of a series of journeys interspersed with linking activities, these being either waiting (layover) time at a terminus or movements between termini or to and from garages. Both these types of activity may contain slack time. Waiting time at a terminus usually includes a minimum layover time which is specified to allow the driver to make up for any late running, to carry out certain essential activities and to have a brief rest; typically this may be about ten percent of the duration of the previous journey. Further *slack* waiting time is often necessary in order to match arriving journeys with departing ones, although a good planning team will have chosen departure times which tend to minimise slack. (Sometimes this may be done with the help of the scheduling system.)

Movements between termini are usually made up of a minimum layover time as above, a minimum *dead running* time between the points involved (which may vary by time of day), and a slack as above. Slack time is not present in movements to and from the garage, and minimum layovers may not be required in such movements. Table 1 presents a typical bus block, although the last three columns would not normally be shown in working documents. These documents might however also include statistics of distance travelled, as well as summaries of the various components of the block.

A secondary objective is normally to minimise some combination of slack time and dead running time, typically in the ratio of one to two, although most bus operators would not define such a ratio unless they were using a computer program. The constraints of a bus scheduling system are usually the departure and arrival times of journeys to be scheduled, together with a table showing the minimum dead running times between points. Minimum layover times may be included in the dead running entries or may be added to the arrival times of the individual journeys. A good bus scheduling system will help the user to develop individual journeys, starting with a departure time, and building up point-to-point journey legs from route definitions. Thus the arrival times may be computed automatically, and the journey duration will depend on the journey times specified for individual legs, which may vary by time of day.

DEPART	Time	ARRIVE	Time	Route	Layover	Dead run	Slack
Garage	1547					4	
Station	1551	Moreton Cross	1623	6	4		3
Moreton Cross	1630	Station	1658	6	3		2
Station	1703	St Andrews	1722	18	3	13	1
High Street	1739	Ash Road	1809	25	3		4
Ash Road	1816	North End	1901	25	5		4
North End	1910	Ash Road	1950	25	4		1
Ash Road	1955	North End	2035	25	4		1
North End	2040	Ash Road	2120	25	4		1
Ash Road	2125	North End	2205	25	4		1
North End	2210	Ash Road	2250	25		11	
		Garage	2301				

Table 1. A typical bus working

The scheduling heuristic used in the present work is an extended version of that initially used for rail locomotive scheduling [14], and subsequently more formally defined as the VAMPIRES algorithm for bus scheduling [15]. The current version, known as BOOST, is presented in [8]. In its simplest form, this heuristic examines pairs of linking activities in a schedule and exchanges their end events if this would not lead to a worsening of the secondary objective. Clearly this cannot reduce the number of vehicles required, but the process starts by calculating a strict lower bound to the number of vehicles, which the user can override if appropriate. Using this as a target number of vehicles, the system constructs an arbitrary schedule, linking two journeys infeasibly if no feasible connection may be made without exceeding the target. Infeasible links are assigned a penalty equal to the time necessary to make the link (the minimum layover and dead running times), minus the time available between the arrival and departure times of the two connected journeys (which may be negative). The improvement heuristic then gives priority to exchanges which reduce the total penalty of the pair of links being considered.

Clearly, no guarantee of optimality can be given, but in many thousands of practical applications, no schedule has ever been found to be better than that produced by the heuristic, and the heuristic has never failed to produce a feasible schedule where one is known to exist with the target number of vehicles. If the heuristic does not reduce the total penalty to zero it is therefore fair to assume that no schedule can be obtained. By adjusting the times or other characteristics of journeys associated with the remaining infeasible links, the user may be able to make the schedule feasible, but otherwise the process may be repeated with a higher target.

Driver scheduling is the process of constructing shifts, each of which obeys often complex rules which determine its legality and its agreed desirability, and which together cover the entire contents of all the blocks (except possibly layover at points where there is security for unattended vehicles). Drivers may be changed at *relief opportunities*, occasions when the vehicle is at one of a number of designated changeover points. In bus operation there is usually one relief opportunity per journey, either at a terminal point, or when the bus passes near the garage or near a town centre. The portion of a block between successive relief opportunities may be called a *piece of work*.

A survey of computer methods for bus driver scheduling up to 1993 appears in [19]. The more recent TRACS II system is based on a set covering formulation in which a large number of potential shifts is first generated, then refined through a selection process before being presented to a specialised integer linear programming process which ensures that each piece of work in the bus schedule is covered by a shift..

TRACS II has been described in the context of rail driver scheduling [6,7], and is also used for scheduling bus drivers. It depends on two basic data files. One of these, the VEH file, presents information about the network being scheduled, namely the list of relief opportunities (point and time pairs) for each bus block, together with information about designated time allowances for travelling between points, signing on and off, etc.

The other, the LAB file, consists of parameters defining the labour scheduling rules appropriate to the area being considered. These include such things as the maximum time on duty, the maximum elapsed time without a break, the minimum meal break length., etc. A wide range of parameters has been chosen so that most features which are likely to vary between organisations can readily be defined. However, some users present rules which cannot easily be defined by generic parameters and have to be handled by specially written code. Such special rules can usually be coded very easily.

In most bus undertakings drivers are able to walk between relief points using a standard table of walking times, or are given standard allowances for travelling on frequent services. However, some bus operators, and most train operators, require drivers to travel as passengers on scheduled journeys to move from point to point, and the first process of TRACS II is to compile a list of all point-to-point travel opportunities at all times of day.

The system next builds a large number of potential shifts, each obeying all the rules as determined by the parameter list and by any special code. These are normally constructed in such a manner that every piece of bus work is covered in many different ways, and there is currently an upper limit of 100,000 in the number of potential shifts that may be carried forward to subsequent processes. Sometimes errors in the data or choice of inappropriate parameter values may result in some work having no potential shifts covering it, in which case the user is warned. In cases of error the data may be corrected. However, sometimes some rules are too wide ranging, causing too many potential shifts to be generated. The potential shifts may then be generated in a few separate runs, each of which covers a subrange of rule combinations with tighter parameter values. The shift sets may be merged later, in which case it is only necessary that the merged set covers all the work.

The set of generated shifts may now be refined, either so that later processes are speeded up, or so that several sets may be merged without exceeding the limit of 100,000. The refining process consists of several passes through the generated set in which shifts are ranked according to a combination of efficiency and likely importance. (In this context, an important shift is one which contains some work which is included in a relatively small number of other shifts.) Poorly ranked shifts are eliminated, provided that this does not leave any work uncovered. The ranks of the remaining shifts are adjusted before the next pass, to reflect the increased importance of those shifts, some of whose work was covered by the eliminated shifts. If several sets of potential shifts have been generated these are now merged. The final refining stage eliminates any shift which leaves a bus where no other shift takes over, or vice versa, unless this would result in some work having no covering shifts.

The remaining shifts (again currently up to 100,000) are presented to a specialised integer linear programming process [4]. This minimises a combination of total cost and number of resultant shifts, subject to each piece of bus work being covered by *at least* one shift, with limits on the numbers of shifts of particular types specified by the user. The stipulation of at least one shift, rather than exactly one shift, covering each piece of work is designed to cater for situations where less efficient shifts have been discarded in earlier processes. Sometimes the nature of the bus schedule is such that some inefficient shifts are necessary if pieces of work are to fit together exactly; these are generally shorter versions of potential larger shifts which would overlap other shifts in the computed solution. Management may choose which of the overlapping portions are discarded.

3. Implementation of system

By 1996 the research team had developed a new generation of scheduling software as above, but despite having extensive experience of tackling real problems and implementing solutions, we did not have the resources to move from research led methods to a viable commercial system. We therefore sought new commercial partners.

Throughout the period of development of the new methods we had maintained contacts with many bus and rail companies, demonstrating our work as it evolved. It was evident that there was a strong need for a new generation of scheduling software for each industry, and we were able to identify potential partners with links to the relevant industries.

Grampian Computers Ltd had developed software for the bus industry over fifteen years, covering Depot Operations, such as Duty Allocation, Engineering and Passenger/Mileage/Revenue Analysis, as well as more general software packages such as Depot Stores, Payroll, Personnel and Accounting. They had a wide client base, and were strongly recommended by several of our industrial contacts. We therefore approached that company towards the end of 1996, and after some months of planning how we might work together, we entered into a formal agreement on 1 October 1997. This provided, among other things, for Grampian to provide windows-based interfaces for the software, and for us to work together towards further development and exploitation. It would be necessary not only to provide user-friendly bus and driver scheduling systems, but also to link these so that the driver scheduling system could pick up details of relief opportunities from the vehicle blocks resulting from bus scheduling.

An opportunity to launch the fruits of the co-operation presented itself in the biennial UK national Coach and Bus Exhibition in early October 1997. Although this was earlier than would have best suited the development plans, it was felt that the opportunity should not be missed. An audio-visual presentation of the projected system, named Openbus, was therefore prepared, together with some working windows-based components which could be demonstrated along with the windows-based prototype version of BOOST, and the TRACS II bus and driver scheduling systems operating under MS-DOS and using flat data files. The demonstration of driver scheduling included data from a user of our earlier software for whom we had been able to show that significant improvements could be obtained from the new methods.

The lead-time between initial demonstration of a scheduling system to a potential client and the placing of an order is normally several months, if not years, and it was anticipated that the six months immediately following the launch would be devoted to the completion of an integrated working system, and to the promotion of the system to possible clients. However, one of the organisations attending the demonstration was already actively investigating scheduling systems, and required a new system which would produce working schedules according to a new operating scenario by the end of the year. After discussions with this organisation it was accepted that a prototype driver scheduling system could meet this target, and that the integration with the bus scheduling component in January 1998 would be acceptable. It would be necessary however to prove during November 1997 that the new system could produce efficient driver schedules matching current operations.

4. Reading Buses

The Borough of Reading is situated on the River Thames, about forty miles west of London. The population of the Greater Reading Urban Area is approximately 250,000. Following the deregulation of the British bus industry in 1986, the municipal bus undertaking was reconstituted as a private company, but remained in local authority ownership. At that time it had about 120 buses, all on urban operations. In 1992 the company bought part of another company, thus acquiring a set of rural services based on Reading, and a smaller set of rural services around Newbury, twenty miles to the west. This acquisition added about sixty buses to the fleet. All the Reading based services were subsequently operated from the former municipal depot, but in two separate groups, urban and rural, while a smaller depot at Newbury was retained. The Reading based services all either terminated in or passed through the town centre or the railway station. The Newbury services all operated into and out of a bus station.

By the middle of 1997, the company was running 215 vehicles in all, in seven main separately scheduled groups, namely “big buses” and “midibuses” for each of Reading urban, Reading rural and Newbury services, and coaches on a Reading to London service. Although there were several types of big bus, necessitating separate vehicle schedules in some cases, they were largely interchangeable as far as drivers were concerned. There were about 375 drivers employed, working on rotating rotas with around 270 shifts required on a normal weekday. The peak output of about 170 buses in each of the morning and afternoon peak periods was serviced by approximately 100 early shifts, 70 split shifts covering both peaks with a long break in the middle of the day, and 100 middle or late shifts covering the afternoon peak together with respectively the midday or late evening period.

The Reading bus depot had evolved from an old tramway depot, originally built around 1901. It was about to be redeveloped as part of a shopping complex, with a new depot due to open elsewhere in the town in February 1998. The old depot was near the town centre, and drivers on most bus routes could be relieved very close to the depot. The new depot was less suitably placed and was expected to lead to higher operating costs, with most driver reliefs taking place at the station, or as buses moved into the depot. It had been agreed with the drivers’ union that all shifts would start and finish at the depot, where meals would also be taken, and that eleven minutes would be given where necessary to walk between the station and the depot. As there would initially be no change in the maximum duration of shifts, it had been estimated that, after allowing for increased walking time, the transfer to the new depot would result in something over fifteen minutes driving time being lost from the average shift.

The company had decided to purchase a computer scheduling system, in order initially to facilitate the construction of schedules for the new depot, but also so that later they could experiment more readily with alternative forms of labour agreement. At the time of the October exhibition they were already using some software (for accounts, payroll and stores management) from Grampian Computers, but were in an advanced stage of negotiation with another supplier of scheduling systems. However, their knowledge of the authors’ previous work in the design of scheduling systems coupled with their confidence in Grampian's ability to provide links to other systems, the quality of inputs and outputs, and projected cost savings persuaded them to investigate the embryonic Openbus system. A presentation was made to the Company on 4 November, and within a week agreement had been reached for the purchase of the system, subject to an early demonstration of its capabilities.

5. The first steps

The Openbus system, when completed, would start from the definition of bus routes with point-to-point journey times (which could vary by time of day). This, together with specification of dead running times between selected points, would comprise the basic network data. The description of each bus route would include a definition of the appropriate driver relief point. Departure times would then be defined for each direction of travel on each route, by specifying the time of the first journey, the service interval, the time of the last journey on that service interval, a new service interval where appropriate, and so on through the day. The system would then generate full details of each journey, and these would be presented to the bus scheduling algorithm outlined in Section 2.

Following the computation of the vehicle schedule, relevant data for driver scheduling would be extracted automatically. For each vehicle, a set of driver relief opportunities would be prepared, consisting of the time the vehicle left the depot, and a list of the times and places of subsequent opportunities, followed by the return time to the depot. This is the basic component of the VEH file already mentioned; the remainder of that file would be presented to the system through a new windows-based interface. The components of the other major data file, the LAB file, would also be presented through a windows interface.

The vehicle and driver scheduling algorithms would be driven also through a windows system.

When the contract with Reading Buses was signed in early November, some work had already been carried out in developing the interfaces for the new system, using the Progress fourth generation language. There was a requirement that the efficiency of the driver scheduling method should be demonstrated by the end of November, comparing existing schedules for the old depot with trial schedules produced for that depot by the computer. It was accepted however that this demonstration should be carried out using the existing vehicle schedules and that the research version of the driver scheduling algorithms could be used.

Once the demonstration had been carried out to the satisfaction of the bus company, full schedules for the new depot would have to be prepared by Christmas. Priority would be given to customising the driver scheduling system for any specific requirements of Reading Buses, and it was accepted that the fully integrated system might not be available for this work. In the first instance, therefore, new bus schedules would not be produced by computer. Instead, existing schedules would be amended manually to provide for extra running time to and from the new depot, without altering the sequences of journeys for any vehicle; the relief opportunities would be computed manually, showing the times the buses on the existing schedule passed the new relief points. These manual relief opportunities would be typed into appropriate VEH files to provide data for the existing TRACS II driver scheduling system.

The prototype TRACS II driver scheduling system would be installed on a Reading Buses PC while the early demonstrations were taking place, so that Reading staff could get experience in running the programs for themselves even before they were installed as part of the new

Openbus system. Other system components would be handed over, even in prototype form, as they became available.

Although the driver scheduling system had been written with an extensive parameter list which could reflect most operating conditions of any organisation, it almost always happens that a new client has some specific needs that cannot be quantified by means of the existing parameters. Reading was no exception to this. For example, the system as written assumed a certain maximum time, uniform over the whole system, that a driver could be on duty without a break. In Reading this time varied according to the principal route being followed by the bus. Further, there was a rule that any shift had to contain a mixture of bus routes, allowable mixtures consisting of quite complex combinations. While these, and other specific features, could in principle be accommodated, it was agreed that they need not be followed in the demonstration to take place during November.

The early work was divided among the team according to the urgency of the various tasks to be followed. Our Grampian colleagues concentrated on preparing windows-based interfaces, including a system to extract relief opportunities from the bus schedule and to create the flat VEH file required by the driver scheduling component. One of the present authors improved the bus scheduling system and liaised with Grampian on specification of data formats, while the other worked with Reading Buses staff on the manual preparation of files for the first demonstration runs of TRACS II. Other University colleagues adapted the TRACS II code so that it could ultimately meet all the specific Reading requirements.

As data for the initial computer runs had to be prepared manually, it was agreed with Reading Buses that it would be sufficient to demonstrate that TRACS II could produce good driver schedules for the most complex of the various schedule groups operating from the Reading depot, and that this could be restricted to weekday operations. In total there were just over two hundred shifts in Reading on a weekday, of which the largest group of 141 shifts operated the urban big bus schedules.

The Reading Buses planning team consisted of two people, only one of whom had computer skills but had not previously used an automatic scheduling system. However, within two days they had grasped the way in which data had to be collated and presented to the computer, and had prepared with us the necessary input files for a run of TRACS II. The first computer run was carried out on 19 November, only nine days after agreement for the project had been reached.

The resultant schedule used considerably fewer shifts than the existing operating schedule, but contained a few shifts which violated some of the conditions specific to Reading which had not yet been coded. On the other hand, the system had not yet been able to take advantage of some circumstances in which rules could be relaxed. After careful analysis of the schedule produced, Reading Buses concluded that after allowing for the above variations, it had been satisfactorily demonstrated that the TRACS II schedules were significantly more efficient than the existing ones. Authority was therefore given to proceed to a full implementation.

The above computer run had drawn attention to some further company-specific rules that had not been fully appreciated in the data preparation or in the parallel revision of code being carried out in the University. The next priority was therefore to complete the recoding including these rules, and to revise the data so that a stricter comparison could be made. This work was carried out over the next three weeks, and a computer schedule was then produced for the old depot which strictly followed all the necessary conditions. The computer schedule used 136 shifts at a cost of 67786 minutes of paid time, compared to 141 shifts and 69714 minutes in the manual schedule. This saving of 32 daily paid hours arising from only a part of the total operation was judged sufficient on its own to justify the system costs.

In parallel with the above comparative runs, data had been prepared for the computation of all driver schedules for the new Reading depot. As already foreseen, the bus schedules were adjusted manually to take account of the new depot, as the integration of the bus and driver scheduling components of Openbus was not yet complete. Although it had originally been anticipated that the integrated system would not be ready in time to prepare schedules for the new depot, it was by this time realised that a projected delay in depot construction might provide the opportunity in January to complete the whole scheduling process by computer. However, by completing the first set of schedules by Christmas as originally stipulated, it would be possible to review the schedules thoroughly and to make adjustments in the parameters in time for full implementation.

Work proceeded through early December to prepare data and undertake the necessary computer runs. By 23 December, schedules had been produced for twelve data sets, namely urban and rural big bus and midibus for each of weekday, Saturday and Sunday. Table 2 compares the computer-produced schedules for urban operations from the new depot with the manual schedules from the old depot, the cost being expressed in hours and minutes of paid time. It will be seen that despite a reduction in the numbers of shifts required, the total cost of the TRACS II shifts for the new depot is higher, as expected, with a total weekly paid time of 8479 hours compared with 8366. For this reason, the mean time per shift is presented; had this not been acceptable it would have been possible to constrain the computer to create more shifts with lower average costs. It should be recalled that it had been estimated that the new depot would result in about 15 minutes less productive time per shift, or an increase of about 255 hours paid time based on (855 + 165) = 1020 shifts. This could have brought the total paid time to 8621 hours had TRACS II not been used.

	Old depot, manual			**New depot, TRACS II**		
	Shifts	**Cost**	**Mean**	**Shifts**	**Cost**	**Mean**
Urban Big Bus, Mon-Fri	**141**	**1161:54**	**8:14**	**138**	**1189:26**	**8:37**
Urban Big Bus, Saturday	**118**	**988:31**	**8:23**	**111**	**980:25**	**8:50**
Urban Big Bus, Sunday	**32**	**263:04**	**8:13**	**29**	**245:30**	**8:28**
Weekly total	**855**	**7061:05**	**8:16**	**830**	**7173:05**	**8:38**
Urban Midibus, Mon-Fri	**27**	**216:53**	**8:02**	**25**	**215:23**	**8:37**
Urban Midibus, Saturday	**25**	**181:05**	**7:15**	**24**	**186:45**	**7:47**
Urban Midibus, Sunday	**5**	**39:04**	**7:49**	**5**	**42:20**	**8:28**
Weekly total	**165**	**1304:34**	**7:54**	**158**	**1306:00**	**8:16**
Overall	**1020**	**8365:39**	**8:12**	**988**	**8479:05**	**8:34**

Table 2. Manual schedule at old depot compared with the first computer schedule for new depot. Cost is in paid hours and minutes.

Priority for the next stage of work was given to the Reading urban services, which were those most affected by the change of depot. In the meantime, the bus schedling component was being improved in several respects. Specifically, provision was made for the user to specify desired linking of arrivals on one set of routes to departures on the same or on another set of routes, provided that this did not cost more than a stipulated additional waiting time. The system was also extended to provide automatic calculation of distances on dead runs (empty bus working between points) as well as on scheduled journeys, so that full mileage statistics could be provided.

6. Full development

By the end of 1997, driver schedules had been produced as above for the new Reading depot. (There was no urgency to reschedule the smaller depot at Newbury, as there were to be no immediate alterations.) The first few months of 1998 saw the building of the new depot fall further behind schedule. This provided an opportunity to complete the windows-based data input and the integration of vehicle and driver scheduling. It also allowed the new Openbus system to be fine tuned to provide more user-friendly facilities, and to be extended to give additional features. For example, the bus scheduling component was altered to allow the user to express preferences for constraining buses to remain on certain groups of related routes throughout the day if this did not incur costs greater than a certain user specified amount.

The first full runs of the integrated system were carried out during February 1998, yielding new vehicle and driver schedules acceptable to management. It is established practice in the bus industry that new schedules are presented to drivers' representatives and that they may be adjusted after negotiations to meet any objections. The trades union had been alerted at the outset of the project to the fact that a computer would be used to produce schedules in future. The benefits of computer use had been explained to the representatives, principally that this would enable potential new working conditions to be explored through the speedy production of schedules meeting any proposed alterations.

For some time the union had been pressing for a number of concessions, for example longer meal breaks, restrictions on the amount of continuous driving time on particular routes. The representatives were disappointed that the schedules now being presented did not meet any of these, and complained that they were not seeing the promised benefits of computer scheduling. Management explained that it had been necessary to proceed speedily to get schedules produced for the new depot and that it had always been the intention that these first computer schedules would follow the old rules. Once the new depot was in operation it would be possible to use the computer to experiment with a whole range of possible new working scenarios.

However, the delay in the depot's opening now allowed some of the union's wishes to be tested by computer, and a number of alterations to the agreements were made as a result. The wide range of possible scheduling parameters available through Openbus enabled experiments with potential new conditions to be carried out without any alterations to the programs, and entirely by the staff of the bus company on their own.

Originally it had been anticipated that following the expected move to the new depot in January, some overdue alterations to the bus network would be made, leading to a complete revision of schedules in April. With the delay to the depot opening, it was now possible to carry out these alterations with the help of Openbus so that they could be implemented at the same time as the move to the new depot. The vehicle and driver schedules incorporating revised working conditions and the alterations to the urban bus network were completed in time for the opening of the new depot on 20 April 1998.

Although these schedules were produced by Openbus, with adequate summary information, the system did not yet produce all output documents in the form and with the full details normally used by the drivers. These documents were prepared partly by hand and partly by linking to other existing systems. Final union negotiations took place using these documents, which were altered by hand to meet specific objections.

Following the implementation on 20 April, discussions were held with the union to determine whether it was possible to lessen the amount of manual adjustment for the next service change, due on 17 August. It was agreed that the original TRACS II form of output contained all the information needed for schedule negotiation, and the following procedure was agreed for the future.

Openbus would first produce the bus schedule. This would be adjusted interactively using Openbus to reflect some conditions that could not be specified by parameter. The driver schedule would then be produced by Openbus in summary form, and negotiated with the union. Agreed alterations might then be made either by adjusting parameters and producing a new schedule, or by carrying out minor amendments using Openbus's interactive facilities. By this time Openbus had been extended to produce full details of bus running schedules in standard Reading format (one full A4 page per vehicle), and to give links to other systems to produce timetables and other forms of output.

A further major service change was to take place on 17 August, and it was agreed that this would be carried out fully as above using Openbus, which had now been extended by Grampian Computers to provide further desirable features. The scheduling activity was carried out as planned up to the point where the driver schedules were presented to the unions.

The schedules as produced conformed to all the agreed working conditions. However, in minimising the number of shifts required, the computer had produced some shifts which, while legal, were of a type never used before. For example, meal breaks on early shifts had previously not started until after the peak, some time after 0900. The computer formed several shifts with meals starting around 0800, as this enabled the drivers concerned to relief other drivers about an hour later, and so to build up a chain of mutual reliefs; this feature enabled single new shifts starting around 0800 each to provide cover for a series of meal breaks, avoiding the creation of additional shifts which did not contribute to the morning peak cover. Although these shifts with meals starting around 0800 were perfectly legal, they might have been forbidden if anyone had previously thought that they were likely to be used.

The union rejected the new schedule, but by this time it was too late properly to assess the consequences of any proposed revisions to the rules, and it was eventually agreed that the schedule could be implemented on a temporary basis after some modifications. It would then be necessary to negotiate new rules so that satisfactory schedules could be implemented on a permanent basis as soon as possible. Although these modifications might have been carried out by rerunning Openbus with new parameters, an injury to a critical member of the team meant that only one person trained in Openbus was available, and that person had had to assume much of the work of his injured colleague. The schedules were therefore adjusted by inspection and edited using the interactive facilities of Openbus, which then produced full output documentation. It should be pointed out that the computer system had produced an efficient schedule which then formed the framework for the final schedule implemented on 17 August; without this computer-produced framework, the efficiency gains presented in the next section would not have been achievable.

The above experience led to some important decisions affecting future work. First, it was necessary to agree with the union a revised set of rules which could be presented to Openbus and would cover all eventualities. Openbus would be used to determine the consequences on cost and on schedule structure of any rule change. Once a new set of rules had been agreed, the union would have to accept that normally any schedule which adhered to these rules would be put into place, subject only to minor adjustments if practicable, to meet particular concerns.

The revised scheduling procedure would be first to compute and negotiate the vehicle schedule with the union before proceeding to driver scheduling, any changes being made either by running the vehicle scheduling component again with different parameters or by using Openbus to edit the schedule interactively. The driver schedule would then be compiled by Openbus according to the agreed rules; it should not be necessary at this stage to produce the large quantity of output documents needed for operational purposes. Any required amendments would be carried out either interactively or by agreeing parameter changes and rerunning the system. Finally, full documentation would be produced.

At the time of writing, negotiations are in hand to determine new agreed driver scheduling rules as explained above. The rural bus services are due to be revised in November 1998, and these schedules will be produced by computer.

7. Conclusions

The elapsed time from the initial agreement for the system acquisition to the operation of computer-produced schedules in the new depot was just over four months. During this time, research-led vehicle and driver scheduling programs had been integrated into a full commercial system, user staff had been trained and become adept at using the system, and schedules had been produced for a new depot according to a range of potential new operating agreements as well as the existing one. This had been accomplished in a far tighter time scale than has normally been available for the installation of other commercial systems required in the first instance simply to emulate existing conditions at an existing depot.

That the above time scale was achievable was due to several main factors:

- The research team had very wide experience in developing and implementing scheduling systems for bus companies;
- The BOOST and TRACS II systems on which the work was based employed state of the art scheduling and computing methods;
- Grampian Computers was speedily able to design and build new user-friendly interfaces;
- Both the University and the Grampian teams were able speedily to alter the system to meet new and developing user needs;
- Reading Buses staff had an enthusiasm for the project and quickly learned how to make the best use of the system, understanding and accepting its strengths and limitations.

It had been estimated by Reading Buses that the move to the new depot would have resulted in an increase of £145,000 in annual driver costs. This was based on a theoretical exercise of adding the increased walking time to the existing shifts, even though some of them would then have become illegal. In practice, the schedules produced by Openbus for the new depot cost only £10,000 per annum more than the previous schedules for the old depot. Openbus therefore gave a net saving of approximately £135,000 per annum.

8. Acknowledgements

The work reported here was made possible by the efforts of many people. We are grateful to the staff of Grampian Computers Ltd. with whom we were able to establish an excellent working relationship, and who achieved the very considerable result of implementing an excellent set of interfaces and other relevant tools in a very short time. They made the system appear seamless to the user, so that anyone with a knowledge of what they wanted to achieve could achieve it. They deserve credit for excellence in responding quickly to user problems and for changing functionality where necessary.

James Carney, Graham Horn and Les Andrew of Reading Buses Ltd. are thanked for having the foresight to see the potential of our work before it was fully implemented and for having faith in its development. Graham Horn and John Wood are thanked for the speed with which they absorbed the ability to make best use of the system, and we are grateful to the former for reading a draft of the paper, verifying and correcting the figures quoted, and for making many useful suggestions.

Our colleagues Ann Kwan and Margaret Parker were responsible for developing TRACS II and for providing the amendments necessary for Reading. Les Proll and Sarah Fores developed a new integer linear programming process which forms the core of the scheduling algorithm. Barbara Smith laid the foundations by developing and installing our first successful driver scheduling system with one of the current authors in the early 1980s.

The Engineering and Physical Sciences Research Council provided the research grants under which the vehicle and driver systems were designed and developed.

References

1. J.R. Daduna, I. Branco and J.M.P. Paixao (eds.), *Computer-aided transit scheduling*, Springer, Berlin, 1995.

2. J.R. Daduna and A. Wren (eds.), *Computer-aided transit scheduling*, Springer-Verlag, Berlin, 1988.

3. M. Desrochers and J-M. Rousseau (eds.), *Computer-aided transit scheduling*, Springer-Verlag, Berlin, 1992.

4. S. Fores, L.G. Proll and A.Wren, 'An improved ILP system for driver scheduling', to appear in N.H.M. Wilson (ed.), *Computer-Aided Scheduling of Public Transport*, Springer, Berlin, 1999.

5. A.S.K.Kwan, R.S.K. Kwan, M.E. Parker and A. Wren, 'Producing train driver shifts by computer', in J. Allen, C.A.Brebbia, R.J. Hill, G. Sciutto and S. Sone (eds.). *Computers in Railways V, Vol.1: Railway Systems Management*, Southampton: Computational Mechanics Publications, 359-370 (1996).

6. A.S.K.Kwan, R.S.K Kwan, M.E. Parker and A. Wren, 'Producing train driver schedules under different operating strategies', to appear in N.H.M. Wilson (ed.), *Computer-Aided Scheduling of Public Transport*, Springer, Berlin, 1999.

7. A.S.K. Kwan, R.S.K. Kwan and A. Wren, 'Driver scheduling using genetic algorithms with embedded combinatorial traits', to appear in N.H.M. Wilson (ed.), *Computer-Aided Scheduling of Public Transport*, Springer, Berlin, 1999.

8. R.S.K. Kwan and M.A. Rahin, 'Object oriented bus vehicle scheduling - the BOOST system', to appear in N.H.M. Wilson (ed.), *Computer-Aided Scheduling of Public Transport*, Springer, Berlin, 1999.

9. J-M. Rousseau (ed.), *Computer scheduling of public transport - 2*, North-Holland, Amsterdam, 1985.

10. J.-M. Rousseau and J.-Y. Blais, 'HASTUS: an interactive system for buses and crew scheduling', in J-M. Rousseau (ed.), *Computer scheduling of public transport - 2*, North-Holland, Amsterdam, 45-60 (1985).

11. Preprints of Workshop on 'Automatic techniques for scheduling of vehicle operators for urban public transportation services', Chicago, 1975

12. B.M. Smith and A. Wren, 'A bus crew scheduling system using a set covering formulation', *Transpn.Res.* 22A, 97-108 (1988).

13. N.H.M. Wilson (ed.), *Computer-aided transit scheduling*, Springer, Berlin, 1999.

14. K. Wolfenden and A. Wren, 'Locomotive scheduling by computer', in *Proc. British Joint Computer Conference*, IEE Conference publication 19, 31-37 (1966).

15. A. Wren, 'Bus scheduling; an interactive computer method', *Transportation Planning and Technology*, 1, 115-122 (1972).

16. A. Wren (ed.), *Computer scheduling of public transport*, North-Holland, Amsterdam, 1981.

17. A. Wren and M. Chamberlain, 'The development of Micro-BUSMAN: scheduling on micro-computers', in J.R. Daduna and A. Wren (eds.) *Computer-aided transit scheduling*, Springer-Verlag, Berlin 160-174 (1988).

18. A. Wren and N.D.F. Gualda, 'Integrated scheduling of buses and drivers', to appear in N.H.M. Wilson (ed.), *Computer-Aided Scheduling of Public Transport*, Springer, Berlin, 1999.

19 A. Wren and J.-M. Rousseau, 'Bus driver scheduling - an overview', in J.R. Daduna, I. Branco and J.M.P. Paixao (eds.), *Computer-aided transit scheduling*, Springer, Berlin, 173-187 (1995).