

Anthony Wren \*

## PRÍSPEVOK INFORMAČNÝCH TECHNOLOGIÍ K EFEKTÍVNEMU RIADENIU DOPRAVY

### THE CONTRIBUTIONS TO EFFICIENT TRANSPORT MANAGEMENT

*Stredobodom dopravných systémov je tvorba časopriestorových rozvrhov. Počítačové systémy pre tvorbu rozvrhov sú inštalované v mnohých dopravných organizáciách a úspory sú dobre dokumentované. Môžu to byť: priame úspory vychádzajúce z efektívneho plánovania, nepriame úspory vychádzajúce z možnosti preskúmať úspešné alternatívne scenáre, vyvolané úspory, ktoré vzniknú tým, že počítač upozorní na to, kde by úprava časových rozvrhov alebo prevádzkových pravidiel mohla zvýšiť výkonnosť. Na druhej strane použitie počítača pre tvorbu rozvrhov nedáva samo osebe záruku, že rozvrh bude dobrý. Jestvuje veľa atraktívne prezentovaných systémov, ktoré často veľmi rýchlo produkujú biedne rozvrhy. Takýchto sa treba vyvarovať a je dôležité, aby ľudia, ktorí sú zainteresovaní vo voľbe systémov, vedeli oceniť rozdiely medzi dobrým a zlým algoritmom. Potvrďujeme to svojimi vlastnými skúsenosťami s úspešným využívaním informačných technológií pre plánovanie a poukajeme na ich niektoré úskalia.*

*Transport systems revolve around their schedules. Computer scheduling systems are now installed in very many transport organisations, and savings are well documented. These may be: direct savings from more efficient schedules; indirect savings because of the ability provided to investigate speedily alternative scenarios; induced savings where the computer has pointed out how adjustments to timetables or operating rules would increase efficiency. However, the use of an IT system for scheduling does not, in itself, guarantee a good schedule. There are many attractively presented systems that produce poor schedules, often quite quickly. These should be avoided, and it is important that people involved in the choice of systems should appreciate the distinction between good and bad algorithms. We report from our own experience on successful uses of IT for scheduling and show some of the pitfalls.*

#### 1. Introduction

Although scheduling is central to transport operation, management has often failed to appreciate the scope for increased efficiency through better scheduling practices. Schedulers in many transport organisations are relatively poorly paid and have little formal training in either in conventional scheduling processes or in IT skills. Often scheduling staffs are nearing retirement age, and little is being done to plan their replacement.

The skills required for scheduling are very similar to those required for computer programming, and young people with these skills are generally attracted towards IT, leaving transport organisations in a vulnerable position. Yet where good computer systems have been integrated with the scheduling function, the job satisfaction -even of older staff- has increased, and potential recruits have been excited by the challenge of getting the most out of both the scheduling system and the transport resources.

The author has been engaged in the development of computer processes for scheduling trains, buses, trucks and their drivers since 1960. These processes have yielded:

- *direct savings*, where computer generated schedules have typically saved five percent of operating costs, and often more;
- *indirect savings*, where the system has enabled the planner speedily to produce schedules for many possible operating scenarios, and to choose the scenario which provides the best compromise between service levels, operating cost, and staff satisfaction;
- *induced savings*, where the computer has highlighted those elements of the planned operation which would inhibit the production of an efficient schedule, allowing the user to adjust departure times or other factors where practicable, and then to produce a new schedule matching the adjusted parameters.

Within the limits of this paper the author cannot detail the many successful applications of scheduling systems with which he has been involved, let alone other people's applications. Some of the author's own experience is highlighted here, and reference is given to other sources. Those seeking further information are referred to the proceedings of the seven published international conferences on Computer-Aided Scheduling of Public Transport (e.g., [1]). An eighth conference was held in Berlin in June 2000. Much of the work of the author and his team is summarised in [2].

\* Anthony Wren

Emeritus Professor of Scheduling and Constraint Management, University of Leeds, UK

## 2. Train and Bus Scheduling

In its simpler forms, the problem of scheduling rail locomotives is logically equivalent to scheduling buses. A set of planned journeys is pre-specified by the train company, and can be summarised in a list showing, for each journey, the starting and finishing points and the relevant times. Vehicles are then assigned to these journeys in such a way as to minimise the number of vehicles used and within that number, to minimise the amount of running empty (known as dead running) between journeys to or from the depot. It is necessary to give computer information about the times required for these dead journeys; good software can intelligently construct a full table of such times from skeletal information.

The author first met this problem in 1960 when he was asked to undertake research into the possible use of computers in rail locomotive scheduling, in the particular context of a set of local freight trains operating between London docks (east of London) and several goods yards linked to the main lines to the north and west. Although it was in principle necessary to ensure that paths through the tracks were available for any locomotives running dead through the system, the operating staff agreed that it was always possible to find a path for a locomotive within a few minutes of any desired time, and that a deviation of a few minutes was unimportant, given the flexibility of freight train operation.

The existing manually prepared schedule had been built up over many years, and alterations to journey requirements had been met by a series of patches. It was therefore believed that there was considerable scope for savings. A simple heuristic was developed, and the first solution obtained a saving of one locomotive out of every 15. However, the principal advantage of the heuristic was that it could produce a solution for any given number of locomotives; if the number was too small for a feasible solution, then the computer pinpointed those journeys whose times would need to be changed in order to make the solution operable. In this case, changes to the times of only two trains allowed a full solution with only 12 locomotives, saving 28 percent of the dead running time. This solution was put into operation in 1963, and this is believed to have been the first implementation anywhere of a computer-produced rail schedule. Subsequently, the system was used to schedule passenger train units in many parts of the United Kingdom.

Nowadays, rail operation in the UK is planned with the availability of traction units in mind, so this type of scheduling system is no longer necessary. However, the use of such a system is still valuable where problems of assigning units to pre-planned journeys exist.

The same process, known as the VAMPIRES algorithm, has been used since 1971 for bus operation, and has saved many vehicles for different companies. It is particularly valuable where there are irregular operations, some journeys starting where no journey has recently finished, for example, journeys that collect children from schools. During the 1970s it saved over 100 buses for 12 companies, with an average saving of nearly 10 percent, after taking account of the computer's suggestions for retiming journeys. Versions of

VAMPIRES, which scheduled several depots and several different vehicle types, simultaneously were developed through the 1970s, but were not often used. More recently, the VAMPIRES algorithm has become the standard scheduling process for many operators. *Direct* savings are no longer regularly expected, because its earlier use has eliminated inefficient operations, but it has become a valuable tool in assessing alternative patterns of operation, allowing advantage to be taken of possible new route structures, thus yielding *indirect* savings. Its power of showing how vehicles can be saved by minor retimings and other adjustments to journeys still leads to *induced* savings. The VAMPIRES algorithm is now incorporated in the BOOST bus-scheduling module, part of the Schedules Office comprehensive scheduling system. A recent application is described in [3].

In the late 1970s it was suggested to the author that VAMPIRES might be complemented by a simpler system, which assisted in developing timetables and schedules for individual bus routes or groups of related routes. The TASC system [4] was intended to be much faster than VAMPIRES. It gave priority to linking arrivals with departures from the same terminus. Where there was no such departure it originally returned the relevant bus to the depot, but subsequently users asked that it should be extended to consider running buses dead to nearby points. Eventually, the extensions requested by users made the system so complex that it was not significantly faster than VAMPIRES, and it did not consider the whole range of dead running possibilities used by VAMPIRES. The scheduling component of TASC was therefore abandoned in favour of a simplified version of VAMPIRES.

It may be worth remarking that there are now many bus-scheduling systems that link arrivals and departures at individual points, leaving unmatched events to be treated manually. Such systems can be very fast, but do not provide the savings, which can be made by systems such as BOOST, which consider the full range of possible empty vehicle movements.

## 3. Driver scheduling

Bus and train driver scheduling problems are among the most difficult managerial problems to tackle using a computer. They belong to a class of problems known by mathematicians as NP-hard. This means that their complexity grows exponentially with the problem size. It is known that no method of solving an NP-hard problem to optimality will ever be found, except for the most trivially small problems.

However, many system suppliers regularly claim that their systems will produce optimal schedules. A supplier who says this is either ignorant of the mathematical nature of driver scheduling or is lying or both. Often their systems use very crude heuristics, which initially produce a few efficient drivers' shifts, but cannot piece together all the elements of vehicle work to form an overall good schedule.

Other recent systems, particularly for train driver scheduling, use constraint programming to develop schedules. It is claimed

that this approach is very fast. Constraint programming can quickly evaluate individual shifts, and can complete a crude schedule. However, it is unlikely that a constraint programming approach will ever satisfactorily solve complex bus or train driver scheduling problems because of the inherent complexity of an NP-hard problem. Nevertheless, suppliers of such systems continue to persuade glibble managements that they should spend large sums of money on these products.

The author knows of two British railway companies that decided some years ago to spending several million pounds on new driver scheduling systems. In at least one case, and probably in both cases, the chosen system was based on constraint programming. Each supplier had promised that the system would meet all the company's scheduling needs, and that it would be operational within six months. Neither company made any attempt to evaluate the quality of the algorithms offered by the suppliers.

The schedulers of one of the companies had taken part in the author's own research in the summer of 1995. Schedules had been produced by our fully automatic scheduling system, which saved two drivers' shifts in a group of 169 existing shifts. The schedules manager had accepted these, and he and his staff were keen to adopt our system. Higher management in the company decided to issue an invitation to tender, and an international firm of consultants was employed to draw up a systems specification. A number of organisations submitted proposals addressing the specification, and the schedules staff favoured a consortium including the author.

We were probably the only proposers who had experience of driver scheduling. We stated that we could provide a system capable of producing good driver schedules within six months, but that another year would be required to meet all the requirements of the specification. (Most of these were irrelevant to scheduling as practiced within the company.) The contract was awarded early 1996 to the firm, which had drawn up the specification. It is understood that the main reason for their preferment was that they had claimed that they could meet the company's full requirements for an automatic system within six months.

Four years later it is acknowledged that the system will never be fully automatic. It will be an interactive tool to assist the schedulers in their decision-making. It is still incapable of working on some of the lines operated by the company. The costs have exceeded the budget by several million pounds.

The other train-operating organisation issued an invitation to tender, having been approached in the first instance by an international consultancy. The author was unsuccessful in his attempts to obtain details of the tender, and the contract was awarded in 1996 to that consultancy. Subsequently, the author met a representative of the consultancy, who explained that they would use constraint programming to produce schedules very quickly. He claimed that this was far superior to the mathematical programming methods successfully used by the author and his colleagues. Although an international expert on constraint programming who

also had experience of driver scheduling refuted this claim, the consultancy persisted in this approach. Four years later, the system has not yet been accepted.

By contrast, the United Kingdom's largest bus group has recently decided to purchase a scheduling system for all 26 of its operating companies. Each company had its own set of scheduling conditions. Before making a decision, proposers were invited to take part in extensive tests. These were based on the schedules of one particular operating company, but several different scenarios were presented, representing the most complex sets of conditions from the whole group. Five proposers indicated an interest in the tests, as a result of which three were eliminated. The remaining two completed further tests. The final choice was made partly on the quality of the results, and partly on the perceived ability of the proposers to meet any future requirements. It is believed that only the successful bidder was prepared to discuss the nature of the underlying algorithms. Installation of the system started earlier this year, and significant operating savings have already been demonstrated. The selection process has been described by the group's schedules development manager [5].

Another system, which has been acquired by train companies in more than one country, is based on a process that at first, appears to search through all possible combinations of schedule portions. However, our knowledge of problem complexity tells us that this is impossible in any but a very small problem instance. In fact, this system starts by assembling some apparently good shifts, using the most efficient combinations of pieces of work. However, as the system proceeds, it becomes more and more difficult to form good shifts, and the overall quality of automatically produced schedules is poor. However, this same system working in an interactive mode has proved to be a useful tool for the scheduler whom does not have access to a good automatic system, in assisting the evaluation of possibilities as the schedule is built up.

Recently the author has learned that a major European rail operator is considering a system offered by a well-known international transport organisation. The system supplier is believed to claim that a large schedule will be produced in ten seconds. It is possible that this can be done. It is even possible that the schedule produced in ten seconds will be better than that currently operated by the railway. However, this is likely to be the result of an inefficient and poorly integrated existing schedule. It is very unlikely that a schedule produced by a system in ten seconds will be as good a schedule produced by a system that uses state-of-the-art scheduling software.

#### 4. Successful driver scheduling

In the previous section some problems in the selection of a driver-scheduling system have been outlined. therefore, one may ask: What distinguishes a good driver-scheduling system?

Recall that a driver-scheduling is NP-hard. No system can therefore guarantee to produce an optimal solution to a large problem.

Potential users should therefore be suspicious of anyone who claims optimality. Likewise, no purely mathematical method will solve the problems. All present successful driver-scheduling systems combine mathematical processes with heuristics, which cut down the size of the search space. Sometimes system suppliers boast of their mathematical systems while suppressing the role of the heuristics, but this is unfortunate, as the quality of the heuristics and the way in which they are used determine the quality of the ultimate result. It is even possible that at some future date a successful system may be based entirely on metaheuristics.

If the author were an impartial advisor to a company choosing a bus or train driver-scheduling system, he would recommend close examination of only three existing systems: one from North America, and two from Europe. All use mathematical processes combined with a greater or lesser extent of heuristics. The process of choosing a system should include comprehensive tests involving the most complex scenarios likely to be met. The selectors should insist on learning something about the underlying algorithms, and if necessary, should employ an operational research analyst to evaluate them.

It is important that the developers of any driver-scheduling system should themselves have a good understanding of the operation of the relevant transport mode and should be aware not only of the possible algorithms to be used, but also, of the finer details of schedule operations. The selectors should therefore satisfy themselves of the competence of the individuals involved in the development and maintenance of the system.

The driver-scheduling systems (IMPACS and TRACS II) developed by the author and his team are in use in many transport organisations and are outlined in a parallel paper (Wren, 2000). They start by generating a very large number of potential drivers' shifts, which between them cover all the vehicle work many times over. In principle there would usually be many billion different potential shifts, so heuristics are used to restrict the generation to perhaps 100,000 potential shifts in TRACS II. A mathematical programming process then selects a near-optimal subset of these shifts, which between them provide a complete schedule.

IMPACS was first installed for London Transport Buses in 1984, and it is still used to produce all driver schedules for most of the current London bus operators. It was subsequently installed for many other bus companies, and some rail operators. It has now been superseded by TRACS II, which is being used by an increasing number of bus operators. In every case significant savings have been achieved when compared with manually produced schedules.

An important use of a driver-scheduling system is in the evaluation of alternative operating scenarios. In preparation for the privatisation of British Rail, the whole organisation was divided into 25 Train Operating Companies (TOCs). Many of these TOCs decided to adopt more flexible scheduling practices than were allowed by the former national scheduling rules. However, it was not clear how much any proposed alteration would actually cost or save. About half of the TOCs used TRACS II in the period 1995-6. The system was used first to produce schedules based on the existing timetable and using the current scheduling rules. In each case, TRACS II produced acceptable schedules, which were considerably cheaper to operate than the existing ones. A number of different possible operating rules were then provided to TRACS II, and the results were compared with the schedules produced according to the current rules. In this way, each TOC was able to agree to a new set of rules to suit its own operating conditions.

## 5. Conclusion

Computers processes can play a valuable role in ensuring that transport is based on efficient schedules. They can produce schedules for direct operation, saving considerable sums, and can also quickly show the costs of alternative operating scenarios. However, use of a computer in itself does not guarantee an efficient schedule. There are poor computer methods, often marketed by major international organisations. A potential purchaser of a system should carefully investigate the quality of algorithms used and should ensure that the potential supplier has a real knowledge both of transport scheduling and of suitable algorithms.

## References

- [1] WILSON, N. H.M. (ed.): *Computer-Aided Transit Scheduling*, Springer-Verlag, 1999.
- [2] WREN, A.: *Heuristics ancient and modern; transport scheduling through the ages*. J. Heuristics, vol. 4, pp. 87-100, 1998
- [3] WREN, A. and KWAN, R. S. K.: *Installing an urban transport scheduling system*. J. Scheduling, vol. 2, pp. 3-17, 1999.
- [4] SMITH, B. M. and WREN, A.: *VAMPIRES and TASC: two successfully applied bus scheduling programs*. In Wren, A. (ed.), *Computer scheduling of public transport*, pp. 97-124, North-Holland, 1981.
- [5] MEILTON, M.: *Selecting and implementing a computer aided scheduling system for a large bus company*. Presented at CASPT-2000, the 8th International Conference on Computer Aided Transit Scheduling, Berlin, 2000.
- [6] WREN, A.: *Mathematics, heuristics and metaheuristics in public transport driver scheduling*. Presented at MI 2000, Conference on Management and Informatics in the New Millenium, Žilina, 2000.