Scheduling Trains on Parallel Lines with Crossover Points

R. L. Burdett & E. Kozan

School of Mathematical Sciences, Queensland University of Technology, Australia

Available online: 29 Oct 2009
Scheduling Trains on Parallel Lines with Crossover Points

R. L. BURDETT¹ and E. KOZAN²

¹School of Mathematical Sciences, Queensland University of Technology, Australia
²School of Mathematical Sciences, Queensland University of Technology, Australia

In this article a sequencing approach is proposed for train scheduling on parallel lines separated by crossover points. The primary feature that is introduced is a modeling device called a compound buffer that is very powerful, particularly because it has widespread applicability. It may be used to maintain the correct occupancy levels of lines while allowing trains to pass through the crossover points without additional routing decisions. The compound buffer is a collection of machines that collectively acts as a traditional capacitated buffer. The machines that are part of the compound buffer, however, maintain their independence and are not treated differently in the scheduling process. To demonstrate the validity and effectiveness of this new approach, a variety of typical railway infrastructure were considered. Extensive numerical investigations show that the approach successfully and consistently creates train schedules of high quality. It also shows that compound buffers accurately portray the technical constraints of the real system.

Keywords  Job Shop; Train Scheduling; Decomposition; Metaheuristics

INTRODUCTION

Many railway systems around the world consist of serial corridors with parallel lines (tracks). The primary reason for having parallel lines is to allow a greater flow of traffic (particularly passenger trains) between two locations than is possible with a single line. The cost of building additional line(s) parallel to an existing line may also be considerably less than the cost of building a new line elsewhere.

Though bidirectional traffic can theoretically be handled on each line, unidirectional flow is usually imposed on each parallel track. This in theory allows a greater volume of traffic to be moved between two locations because traffic moving in opposite directions does not interfere with each other. Trains of different speeds and with different stopping profiles that travel in the same direction, however, still interact with each other and cause additional delays that reduce the traffic flow. At present the best way to reduce these delays is to use passing facilities such as sidings (passing loops) to allow trains to overtake each other. Very often, however, independent passing facilities are not constructed for each line because it is too expensive and requires too much additional space adjacent to the parallel lines. Similarly, when the number of parallel lines exceeds two, it is physically impossible for inner lines to have independent passing loops attached to them unless those lines are adjusted/separated. Therefore, interaction between trains traveling in opposite directions cannot be completely removed and must be tolerated.

A cheaper alternative to passing loops that is more commonly used is crossover points. Crossover points, which are the focus of this article, are small links between adjacent lines that allow an adjacent line to be used as a temporary passing lane. Trains must move onto an adjacent track that is unoccupied at that location in order to overtake other slower trains or to be overtaken themselves. A return to the original line via another crossover point is strictly necessary for this move to be successful. Consequently, crossover points usually come in twos. A corresponding pair of crossover points (without the intermediate horizontal tracks) looks something like “\ /” or “\ /” and not “\ ” or “\ /”. The second two in particular do not allow a train to return to the line it was originally on except via a reverse movement, which is not desirable or practical.

This research was funded by Queensland Rail and the Australian Research Council.

Address correspondence to E. Kozan, School of Mathematical Sciences, Queensland University of Technology, P.O. Box 2434, QLD 4001, Australia.
E-mail: e.kozan@qut.edu.au
device called a compound buffer is introduced and is particularly useful for accomplishing this. The compound buffer approach, however, is not limited to this scenario. There are other scenarios (i.e., layouts and networks) where it is also very effective if not absolutely necessary. The compound buffer approach is necessary because a conventional approach to modeling does not necessarily provide the desired level of functionality. For example, a single crossover point can be modeled conventionally as a single independent section. However, if each crossover point were modeled in this way, no passing would be possible between adjacent crossover points without additional routing decisions.

The compound buffer approach relies upon a suitable decomposition of the system, which then provides passing opportunities without additional routing decisions. In this article we identify how best to use the compound buffer modeling approach by considering some typical railway infrastructure. Our approach is generally to model as many passing possibilities that we can in one representation provided the complexity is not excessive. However, it is not guaranteed that all possibilities can be modeled simultaneously. It is up to the modeler (planner) to choose a representation that best fulfills his or her passing objectives. Similarly, it is up to the planner to decide upon the level of detail and accuracy that is necessary or similarly appropriate.

To prove the validity of compound buffer representations, direct deduction or contradiction is used. This amounts to finding cases that violate standard occupancy conditions. In other words, situations that allow physically impossible train movements to occur are sought. If no such conditions can be found, then the representation is assumed valid.

A particular benefit of the compound buffer approach is that it allows the current practice of imposing unidirectional traffic on each parallel line to be relaxed. A comparison of the relative merits of both policies and other alternative policies is possible without the need for the development of a simulation model.

This topic is addressed because it is a novel problem of widespread applicability and is of significant complexity. The compound buffer modeling and accompanying scheduling approach is an innovative solution to a difficult contemporary combinatorial optimization problem. It can be used as part of an intelligent transportation system to more efficiently utilize current railway systems and to help plan more efficient railway systems of the future. The format of this article is as follows. In the following section a previous train scheduling approach upon which this article is based is reviewed. In the next section the compound buffer approach is introduced and simple examples demonstrating how it can be used for modeling are explained. A previous limitation is resolved in the next section and full-sized case studies are then considered. Conclusions and further research directions are given last.

### Past Research

Train scheduling and rescheduling problems have had considerable attention from researchers over the last fifteen years. Examples include Carey and Lockwood (1992); Carey (1994a, 1994b); Cai and Goh (1994); Higgins, Kozan, and Ferreira (1996, 1997); Odijk (1996); Zwaneveld et al. (1996); Kroon, Romeijn, and Zwaneveld (1997); Brannlund et al. (1998); Cai, Goh, and Mees (1998); Chiang et al. (1998); Cordeau, Toth, and Vigo (1998); Sahin (1999); R.M.P. Goverde (1999); Adenso-Diaz, Gonzalez, and Gonzalez-Torre (1999); Lindner (2000); Zwaneveld, Kroon, and Van Hoesel (2001); Kroon and Peeters (2003); Billionnet (2003); Carey and Carville (2003); Dorfman and Medanic (2004); Dessouky et al. (2006); Stella et al. (2006); Carey and Crawford (2007); D’Ariano, Pacciarelli, and Pranzo (2007, 2008); R. Goverde (2007); Mazzarello and Ottaviani (2007); Rodriguez (2007); Tornquist and Persson (2007); and Burdett and Kozan (2008, 2009a, 2009b, 2009c). To our knowledge none of these papers considers how to explicitly and efficiently utilize crossover points, particularly without making additional routing decisions. Dessouky et al. (2006), however, have applied a branch and bound procedure to determine the optimal dispatching times for trains in more complex rail networks. Though effective on a case study, it is reported that effective heuristics are necessary for more realistic sized problems. Carey and Crawford (2007) also considered railway networks consisting of complex stations linked by multiple one-way lines in each direction and developed heuristic algorithms to assist in the task of finding and resolving conflicts in draft train schedules. Mazzarello and Ottaviani (2007) developed an advanced traffic management system for optimizing traffic movement in large railway networks. The system consists of a train scheduling and routing module (CDR) and a speed regulation module (SPG). The CDR system is based upon an alternative graph representation of train movements (similar in nature to that of Burdett and Kozan, 2008) and is used to generate a conflict-free schedule. Rodriguez (2007) considered the routing and scheduling of trains through a junction and proposed a constraint programming model.

In this article the hybrid job-shop scheduling approach of Burdett and Kozan (2008, 2009a) is extended. In that approach trains and sections of rail, respectively, are synonymous with jobs and machines. An operation is regarded as the movement (traversal) of a job across a machine instead of as the processing of a job on a machine. A number of features that are not commonly accommodated by the machine scheduling perspective but are common in railways were incorporated. These included train length and dwell times, safety headways, blocking conditions, acceleration and deceleration, passing loops, and routing flexibility. The result of the adaptations was an innovative activity on node disjunctive graph model for train scheduling. The activity on node (AON) graph representation differed from
Ignoring buffer occupancy violations allows significantly higher quality solutions to be constructed more quickly by the constructive algorithm although the solutions may sometimes be partially infeasible. The application of metaheuristics may be (but not exclusively) viewed as the primary mechanism in which to resolve the buffer occupancy violations (i.e., infeasibilities) in the starting solution.

- Movement through only the feasible part of the search space can limit the solution quality. Sometimes there are very few if any “conventional” moves that can be made without causing BOVs. Similarly, all of the feasible moves that can be made may result in greatly inferior solutions.

- Solution quality is inferior on problems that contain adjacent sections not separated by passing loops. This is because “compound” moves are necessary. Many of the current unitary moves must be restricted because precedence impossibilities result and consequently solutions that are closer to the optimal cannot always be reached.

- Solution quality may be inferior when unnecessary overtaking is allowed. For example, two trains traveling in the same direction may be forced to overtake each other multiple times. This is undesirable and inappropriate in a “real” schedule. In normal circumstances a train should only overtake another train once and it should not itself be overtaken by the same train. Unnecessary overtaking is usually resolved automatically by metaheuristics but this is not an exact process.

- The properties upon which critical path operators have been applied previously do not hold for train scheduling problems. Critical path operators may therefore be inferior and insufficient because the movement of noncritical operations associated with capacitated buffers can affect critical operations.

MODELING USING COMPOUND BUFFERS

This section demonstrates how parallel sections of linked rail can be modeled using capacitated buffers and compound buffers. A number of examples are considered and were chosen because they exhibit generic aspects that appear commonly in real railway systems. It should be firstly noted, however, that if parallel tracks do not interact in any way, then compound buffers and the strategies of this section are unnecessary. It should also be noted that the capacitated buffer approach converts machine routing (or precedence) networks like the one in Figure 1(a) to equivalent networks like the one shown in Figure 1(b). Round circles represent sequenced machines and rectangles capacitated buffers. Links between nodes represent feasible paths between machines. Parallel sections of rail can be modeled as capacitated buffers as long as they are of the same or comparable length.
The travel time on parallel sections should also be of the same magnitude but this may not be true in reality. For example, in passing loops a train will need to slow down on the divergent path (i.e., the loop), whereas it does not need to on the main line. Fortunately, this aspect does not need to be explicitly modeled in the capacitated buffer approach. In a solution a train automatically uses the main line unless it is in conflict with another train. When two trains use the capacitated buffer at a particular time, one train is made to stop while the other moves through freely onto the next section. The slower speed on the divergent path is indirectly enforced and is absorbed into the waiting time of the train. The sequences on adjacent sections dictate which train should move through freely and which train should not.

**Single Crossover Point Example**

Now let us consider the most basic segment of parallel lines as shown in Figure 2(a). This figure, which is not shown to scale, is a line diagram and signifies the track topology from above. For this example it is assumed that sections S2 and S5 are less than 1 km, sections S7 and S8 are in the order of 10 m, and sections S1, S3, S4, and S6 are of any length. The usual machine layout associated with this is shown in Figure 2(b) as an undirected graph where each node represents a machine. Each machine is associated with a section of rail that can only hold a single train at any one time. It is assumed that signals are placed in such a way that a collision cannot occur between a train on section S4 and one traveling on section S7 that is transitioning between the top and bottom lines. This is similarly true of the right-hand side between sections S6 and S8.

In order to model the system accurately, each of the eight sections must usually be sequenced. This requires much effort, specifically when alternative routing is allowed. For example, a train traveling on section 1 may traverse sections (S7, S5, S8) or S2 to get to section S3. The same train could even swap tracks altogether and continue from section S6 instead.

Various simplification strategies may be proposed. The approach advocated in this article is shown in Figure 3 and involves compound buffers. It should be firstly noted that section S7 and S8 are deemed to be part of section S5 in order to achieve these results.

This method reduces the number of sequenced machines and does not require any job to have a flexible route. Consequently, there are no routing flexibility complications and no additional associated computational burdens. Furthermore, trains traveling on the top line may use the second line for overtaking and passing. Figure 3(a) shows how each line is separated conceptually. This decomposition results in a passing profile. Figures 3(b) and 3(c) are two alternative networks that correspond to Figure 2(a). The square nodes represent capacitated buffers and the associated capacity (i.e., the maximum occupancy level) is written in the right-hand corner. The round nodes represent standard sequenced machines that have an occupancy of one train at a time. It should be noted that specific train travel between S1 and S6 and S4 and S3 is through sections S5 and S7 (i.e., machine M5) and not the S2 + S5 buffer section (machine M2). Machine M2 is for travel across the top line only and to facilitate passing on the top line. In Figure 3(b) two trains cannot be on section S5 at the same time because M5 is a sequenced machine and only allows one train at a time. Similarly, two trains in M2 means that one train is on S2 and one is on S5. The full list of valid train routes and associated paths is shown in Table 1.
annealing was applied several times with different parameters. Solutions were improved using simulated annealing. Simulated permit and forbid BOV options, respectively. Following this the constructive algorithm was applied for each case twice using the approximations proposed in Burdett and Kozan (2009a). The makespan was computed as 67.71 min using the lower bound sections S1 and S4 are 2 km long, sections S3 and S6 are 5 km, 60, 80, 100, 120, and 160 km/h, respectively. For the layout, paths are given in Table 2.

To ensure that machine M2 and M5 together do not exceed the overall capacity of two trains at any one time, the concept of a compound buffer is required. A compound buffer is a list of machines that are “virtually” merged and it is identified by the dashed rectangle in Figures 3(b) and 3(c). The capacity of the compound buffer is also written in the right-hand corner. The compound buffer is treated as a standard buffer with respect to determining buffer occupancy violations, however the compound buffer may include both standard machines and other buffer machines. For example, in Figure 3(b) section S5 may alternatively be defined as a standard sequenced machine. The compound buffer provides a link between the two lines that have been decomposed.

Whenever the objective function is to be evaluated and solution feasibility is to be determined, the standard buffers and the compound buffers are inspected for occupancy violations. In order to accomplish this, intervals of section occupancy are determined and train numbers (i.e., occupants) are counted. Compound buffers should be inspected secondly if at all. For example, if a BOV is found on a standard buffer, then looking for further violations may not be required if identification of infeasibility is only required as opposed to determining the level of infeasibility.

In order to test that the two representations (configurations) are valid and to identify which, if either, is superior, an example consisting of forty trains was investigated. The trains and their paths are given in Table 2.

Of the five trains traversing each path, the train speeds are 60, 80, 100, 120, and 160 km/h, respectively. For the layout, sections S1 and S4 are 2 km long, sections S3 and S6 are 5 km, and sections S2 and S5 are 0.5 km long.

In order to judge solution quality, the lower bound for the makespan was computed as 67.71 min using the lower bound approximations proposed in Burdett and Kozan (2009a). The constructive algorithm was applied for each case twice using the permit and forbid BOV options, respectively. Following this the solutions were improved using simulated annealing. Simulated annealing was applied several times with different parameters.

The purpose of the multiple runs was to be sure that the best solution obtained from the starting solution was consistent and could not be improved upon easily. The results are shown in Table 3. The metaheuristic took a couple of minutes on a 3.2-Ghz Pentium 4 computer, whereas the constructive algorithm took less than a second.

The solutions were analyzed visually and seem to be valid for this basic parallel line segment. The following observations were made from the numerical investigations:

- Layout 1 requires compound moves. In particular, machines M4, M5, and M6 are not separated by buffers and hence a change in position of one operation of a job may require a simultaneous change in position of other operations of that job.
- When BOVs are permitted, the number of occupancy violations is large and most are associated with the compound buffers.
- The second representation is better for the constructive algorithm but not for the metaheuristic. The first layout is slightly better because additional sequencing decisions can be made and more accurate timings can be achieved. More precisely, in the second layout the timings are not always sufficient because entry to buffer machines is automatic.
- Removing existing BOVs associated with compound buffers is quite difficult when using metaheuristics. Consequently, the permit BOV option is less appealing for this type of problem.
- Schedules associated with layout 1 are valid for layout 2 and vice versa as long as operations timings are maintained. If not, the disjunctive graph may alter the timings and cause BOVs.

**Table 1** Valid train routes and paths.

<table>
<thead>
<tr>
<th>Route</th>
<th>Path</th>
<th>Route</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>(A, C)</td>
<td>M1, M2, M3</td>
<td>(C, A)</td>
<td>M3, M2, M1</td>
</tr>
<tr>
<td>(A, D)</td>
<td>M1, M5, M6</td>
<td>(D, A)</td>
<td>M6, M5, M1</td>
</tr>
<tr>
<td>(B, D)</td>
<td>M4, M5, M6</td>
<td>(D, B)</td>
<td>M6, M5, M4</td>
</tr>
<tr>
<td>(B, C)</td>
<td>M4, M5, M3</td>
<td>(C, B)</td>
<td>M3, M5, M4</td>
</tr>
</tbody>
</table>

**Table 2** Selected trains and paths for example.

<table>
<thead>
<tr>
<th>Trains</th>
<th>Path</th>
<th>Trains</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>1—5</td>
<td>AB</td>
<td>21—25</td>
<td>AD</td>
</tr>
<tr>
<td>6—10</td>
<td>BA</td>
<td>26—30</td>
<td>DA</td>
</tr>
<tr>
<td>11—15</td>
<td>BD</td>
<td>31—35</td>
<td>BC</td>
</tr>
<tr>
<td>16—20</td>
<td>DB</td>
<td>36—40</td>
<td>CB</td>
</tr>
</tbody>
</table>

**Table 3** Constructive and SA algorithm results (in min).

<table>
<thead>
<tr>
<th>Option</th>
<th>Layout</th>
<th>CA Makespan (# BOV)</th>
<th>SA Makespan (# BOV)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forbid BOV 1</td>
<td>134.06</td>
<td>87.92</td>
<td></td>
</tr>
<tr>
<td>Permit BOV 1</td>
<td>102.08</td>
<td>83.00</td>
<td></td>
</tr>
<tr>
<td>Forbid BOV 2</td>
<td>120.53</td>
<td>92.87</td>
<td></td>
</tr>
<tr>
<td>Permit BOV 2</td>
<td>77.63</td>
<td>93.94</td>
<td></td>
</tr>
</tbody>
</table>

**Overlapping Passing Possibility Example**

This section considers another typical parallel line example and was chosen because it demonstrates the possibility of overlapping passing facility. The layout is shown in Figure 4(a). This scenario typically occurs when an existing single line is partially duplicated. Passing issues, however, arise due to the length of the duplicated segment and the one train per section safety protocol. More precisely, there is a complete lack of passing opportunities. In order to allow passing between the original line and the duplicated segment, crossover points are introduced. With crossover points there are five routes between A and B. This scenario is not very different to example 1 in the preceding section. For example, the precedence network in Figure 2(b) is almost valid for this example. All that is needed is two
additional nodes, one placed at the front and one placed at the back. In order to model this scenario without making explicit routing decisions, the decomposition in Figure 4(b) may be adopted.

This profile of passing allows two compound buffer representations to be created and these are shown in Figure 5.

The representations in Figure 5 capture the most important passing possibilities. Other representations could be used, for example, if some of the passing possibilities are not to be taken. It should be noted that Figure 4(b) is used as a guide and is not completely necessary for determining the representations in Figure 5. For example, using Figure 1 and the traditional representation of the system the same result can be obtained.

In Figure 5(a) the three crossing loops modeled as capacitated buffers are assumed to be separate. In reality they are linked because they have a common section. For example, sections S9 and S10 are part of three different crossing loops. Consequently, this representation may allow two trains to be on these links at the same time. It is not known whether a conflict will occur if the first representation is used. It is hypothesized that this is unlikely to occur in a solution and if it does, then the position of signals would disallow it in practice. The first representation, however, may be deemed more acceptable if sections S9 and S10 are absorbed into section S6, just like S7 and S8 were absorbed into section S5 in the first example. The representation in Figure 5(b) is alternatively proposed and tries to correct this issue by adding in two additional dummy machine nodes to force occupancy of one train only. The main disadvantage of this approach is that it stops two trains, albeit on different tracks being opposite S9 and S8 at the same time.

In order to test that the two representations are valid and to identify which, if either, is superior, an example consisting of twenty trains was investigated. The trains and their paths are given in Table 4.

Of the five trains traversing each path, the train speeds are 60, 80, 100, 120, and 160 km/h, respectively. For the layout, sections S1 and S8 are 5 km long, sections S2–S7 are 20 km, and sections S9 and S10 are 0.01 km long. The lower bound for layout 1 is 154.16 and 149.12 for layout 2. The results are shown in Table 5.

The following observations were made from the numerical investigations:

- The layouts give slightly different lower bounds.
- The simple interchange (shift) perturbation operator was able to improve the solution more easily than the better compound interchange perturbation operator.
- The best solutions given by the two alternative layouts are not significantly different.

In conclusion, the layout that should be used is a matter of preference. However, layout 1 is simpler and did result in the best solution of 317.57. In comparison to example 1, the throughput is inferior and this is attributable to sections S1 and S8 limiting the flow of trains into the parallel segment.

**Adjacent Crossover Point Example**

This example demonstrates how to model adjacent crossover points and emphasizes the effect of section length on the machine network representation. In the first parallel line example the distance between crossover points was assumed to be sufficient to hold a train and allow passing. In reality this is not always true. For example, if the length of section S5 in Figure 2 is not long enough, then trains will backflow onto adjacent sequenced sections (i.e., section S1 or S3) and the headways there will enforce a one train limit within the crossover point segment. Consequently, two trains both traveling on the top line will not be able to pass each other on this segment. In situations like this the purpose of crossover points appears to be strictly for trains to cross from one line to another and not for trains using the
same line to necessarily pass each other using this infrastructure. In this event two adjacent crossover segments may be used for passing trains. Two examples are shown in Figure 6.

The track lengths are measured between intersection points. In Figure 6(a) trains on the top line may pass each other between sections S1 and S5 by traversing either path (S2, S3, S4) or (S7, S8, S9). Trains on the second line may pass each other between S7 and S9 by traversing either path S3 or S8. Similarly in Figure 6(b) trains on the top line may pass each other between S1 and S4 by traversing either path (S2, S3) or (S7, S8). Trains on the second line may pass each other between S7 and S10 by traversing path (S3, S4) or (S8, S9). For the scenarios shown in Figure 6, compound buffer approaches are more difficult to define and there are more alternatives to choose from if sections S7 and S9 are of sufficient length for trains to pass, then two possibilities are shown in Figure 7.

If sections S7 and S9 are small, the representations shown in Figure 7 do not allow trains to pass each other using sections S3 and S8. In order to do so, the entire middle section can be treated as one capacitated buffer of capacity two. Hence machines M1, M5, M6, and M10 would all connect to this buffer. This is quite a drastic simplification that could be quite appropriate and does not require compound buffers.

In practice, sections S1 and S6 may also be sandwiched between another segment containing crossover points. This is similarly true for sections S5 and S10. Therefore, the representations displayed could be inappropriate. It is important to note yet again that some individual interpretation and selection are required. For example, the user must decide where trains may or may not be allowed to pass each other. This choice dictates how and when to use buffers and compound buffers. Similarly, the track lengths dictate the modeling approach to a great extent. It should also be mentioned that system complexity can be so great that all passing opportunities cannot be simultaneously modeled and a trade-off must be made.

### Table 4

<table>
<thead>
<tr>
<th>Trains</th>
<th>Path</th>
<th>Trains</th>
<th>Path</th>
</tr>
</thead>
<tbody>
<tr>
<td>1–5</td>
<td>AB top line; i.e. (A1, B1)</td>
<td>11–15</td>
<td>BA top line; i.e. (B1, A1)</td>
</tr>
<tr>
<td>6–10</td>
<td>AB bottom line; i.e. (A2, B2)</td>
<td>16–20</td>
<td>BA bottom line; i.e. (B2, A2)</td>
</tr>
</tbody>
</table>

### Table 5

<table>
<thead>
<tr>
<th>Option</th>
<th>Layout</th>
<th>CA Makespan (# BOV)</th>
<th>SA Makespan (# BOV)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forbid BOV 1</td>
<td>623.93</td>
<td>327.68</td>
<td></td>
</tr>
<tr>
<td>Permit BOV 1</td>
<td>184.91 (69 BOV)</td>
<td>317.57 (0 BOV)</td>
<td></td>
</tr>
<tr>
<td>Forbid BOV 2</td>
<td>731.39</td>
<td>322.07</td>
<td></td>
</tr>
<tr>
<td>Permit BOV 2</td>
<td>195.48 (90 BOV)</td>
<td>316 (1 BOV)</td>
<td></td>
</tr>
</tbody>
</table>

### Merging Parallel Lines Example

This section considers the railway network shown in Figure 8(a). This scenario typically occurs when several parallel lines are successively reduced in number. Section S5 allows trains traveling on S2 from B to avoid those trains merging from A on section S1. This example is quite straightforward to model with buffers and compound buffers and the result is shown in Figure 8(b). Consequently, there are no competing representations to be tested.

A compound buffer approach is necessary because sections S6 and S7 constitute a passing loop for traffic moving between B and D but not otherwise. The compound buffer allows passing options for this traffic while maintaining traffic flow from A and C. Trains originating from A and C should be specifically routed through S6 and S7 and must therefore use machines M4 and M6. Trains traveling between B and D may be specifically routed through S6 and S7 or this decision may be made by the buffer machine M5.

For safety purposes to ensure that one train only can be around the vicinity of S4 and S5, M1, M2, and M3 could be attached to a dummy machine, which is attached to M4, M5, and M6. This level of detail is assumed to be unnecessary due to S4 and S5 being quite small.

### A Station Example

In this section the track topology shown in Figure 9(a) is considered. This situation commonly occurs at railway stations. There are two parallel lines and platforms occur on either side of the top line. A third line has been provided opposite the station to allow passing (overtaking). Links S8 and S10 often occur at distances of approximately 1 km from the station in practice.

![Figure 6](image-url)

_layouts with adjacent crossover point segments._

intelligent transportation systems vol. 13 no. 4 2009
In other situations links S2 and S5 may not exist and trains may travel directly between S8 and S3 or S3 and S10. Two compound buffer representations can be created for the layout shown in Figure 9(a), and they are shown in Figures 9(b) and 9(c).

The representation in Figure 9(b) is very similar to the single crossover point example considered in section 2.1 and can be obtained by decomposing the top and bottom lines. However, the top line is also a passing loop and hence M2 is a capacitated buffer. The two buffers M2 and M5 could be combined to simplify the representation; however, this would restrict the possibility of three trains being present in the middle area. Therefore, they are kept separate and a compound buffer of level three is created instead. In this representation S2 and S5 are assumed to be quite small and are assumed to take little time to traverse. Therefore, the possibility of three trains being scheduled over S2 and S8 or S5 and S10 at any time is minimal. In order to strictly enforce that this does not occur or to deal with examples where S2 and S5 are much longer and can hold an entire train, the representation in Figure 9(c) may be used. In Figure 9(c) S2 and S5 are modeled by their own machines M2 and M4, respectively. Three overlapping compound buffers ensure correct occupancy possibilities.

**Embedded Junction Example**

This section considers the railway network shown in Figure 10 that typically occurs when a junction is embedded within one side of a passing loop or duplicated segment. This scenario occurs quite often in Australian railways and is quite difficult to model comprehensively. In fact the complexity is quite deceiving.

A conventional approach where the entire middle section (i.e., S2, S3, S5, S6, and S8) is represented as a buffer of capacity two is not valid though at first glance seems adequate. The reason for this is that a train traveling from A to C could overtake another traveling from A to C. A train traveling from C to A could also pass one traveling from A to C. A similar effect occurs between B and C. A compound buffer approach is therefore required and is specifically designed to get around this problem. Several compound buffer configurations seem to be valid for this example and they are shown in Figure 11.

The second representation (Figure 11(b)) demonstrates the possibility of overlapping compound buffers that has not occurred in previous examples. In order to obtain these representations we firstly observed that only one train may be around the vicinity of sections S3, S5, and S6. Because these sections are roughly of the same size and a train can only traverse one of them in its journey, they can be treated as a single sequenced machine. Furthermore, because they are so small, a train can never be fully contained there and they may be completely ignored or integrated into S4 in order to reduce computational effort. For many situations sections S2 and S7 will be the same size and, though capable of holding a train together, cannot do so individually. S2 and S7 cannot be treated as one machine whose length is equivalent to that of S8 because this would cause a length inconsistency error for trains routed between (A, C) and (B, C). For example, the actual traversed distance is half that of the machine’s actual length. Similarly, a machine whose length is the same as S2 and S7 cannot be used because a length inconsistency would occur for trains traveling between (A, B); i.e., traversed distance is twice as much. Therefore, S2 and S7 must be kept separate.

In order to utilize the passing loop independently of the junction when traveling between (A, B) a capacitated buffer (i.e., M2) is introduced in Figure 11(a). A compound buffer is also introduced to ensure that only two trains may be in the vicinity of the middle area. For example, the compound buffer stops a train from using the top of the passing loop when another train is approaching section 1 or 9 from section 4. It should be noted that trains traveling between (A, C) and (B, C) should be routed
through (M1, M4, M6) and (M3, M5, M6), respectively. Though it is possible to route trains through (M1, M4, M5, M3) when traveling between (A, B) the alternative routing (M1, M2, M3) is recommended. The likelihood that a train is on S2 and S7 at the same time is not great and can only occur if one train is going to A and one is going to B.

The compound buffer representation in Figure 11(a) assumes that the passing loop is of normal size. However, if the passing loop were instead a dedicated parallel section, this representation may be inappropriate because valid three-train occupancy possibilities may be restricted. For example, if sections S2 and S7 are long enough, then a train could be on section S2 and one could be on section S7 at the same time quite reasonably. Furthermore, it would also be quite beneficial to route trains through S2 and S7 because S8 is not divided using signals and hence can only maintain one train at a time. The compound buffer approach shown in Figure 11(b) is proposed for this scenario.

The main difference between Figures 11(a) and 11(b) is the number of trains allowed in the middle area, two in (a) and three in (b). Increasing the capacity of the compound buffer in Figure 10(a) to three was inadequate because two trains could be present on S2 or S7, which is infeasible. The exact conditions for this infeasibility to occur is two A to B trains in M2 and one B to A train in M5. In Figure 11(b) the answer is to split section S8 into two separate parts, each of which is modeled by a capacitated buffer. Additional compound buffers are introduced to maintain one train on each parallel track or similarly a two-train limit on each side of the junction. The fourth compound buffer maintains a three-train limit for the entire middle area. It should be noted that splitting section S8 into two buffers does not allow two trains on S8.

This representation is superior to Figure 11(a); however, it does not quite model all possible traffic flows. In particular, one three-train possibility is modeled but a second three-train possibility is not. The three-train possibility that is modeled occurs when two trains traveling between (A, B) are in the middle area and a third train traveling between (C, B) or (C, A) is able to "sneak by" on either section S2 or S7. However, the situation where three trains traveling from A to B or vice versa are in the middle area (i.e., one on S2, S7, and S8) is restricted. Furthermore, there is no link between M2 and M4 in Figure 11(b) as there was for Figure 11(a). If trains may be routed specifically through M2 (S2) and M4 (S7), then this representation may allow a collision. Consequently, all traffic moving between A and B must be routed through M3 and M5. It appears that a representation without capacitated buffers and specific routing is required in order to allow all three-train occupancy scenarios.

This example introduces a quirk of the compound buffer approach that can be quite nasty. In all previously considered representations compound buffers contained several machines in parallel. This meant that one path from several had to be taken and those paths consisted of only one operation. In the representation in Figure 11(b) there is a compound buffer that includes M3 and M5 in series. What this means is that a train crossing between one half of section S8 to the other will be present on two machines for a short period of time and will be counted twice toward the occupancy of the compound buffer. Therefore, a second train traveling between (A, B) will not be allowed to use the middle area and no passing possibilities will be allowed. This is a serious limitation and without some type of correction this representation may be unacceptable for certain planned traffic flows. There are two choices that may be taken here. The first is to find an alternative compound buffer representation. The second is to make a correction to compound buffer occupancy determination procedures. A correction seems warranted only if this phenomenon occurs greatly; i.e., is not a one off. So far we have not found any other situations though. We have also come to the conclusion that a scenario in which the middle section is quite long is highly unlikely. Therefore, we assume the middle area is a passing loop of normal length. In summary, this quirk is noted but current procedures are left as they are until further evidence becomes available.

![Figure 10](image-url)
In order to test the relative merits of the representations an example consisting of thirty trains was investigated. The trains and their paths are shown in Table 6.

Of the five trains traversing each path, the train speeds are 60, 80, 100, 120, and 160 km/h, respectively. For the original layout sections S1, S4, and S8 are 5 km long; section S8 is 1 km; and sections S2 and S7 are 0.5 km long. The lower bounds for each layout are 67.15 and 66.89 and the results are shown in Table 7.

In conclusion, there is only a slight difference between the best solutions of the two representations. The difference can be explained by the imprecision of the metaheuristic. This result is expected because the crossing loop is small. A longer section S8 is now considered and should show a larger and significant difference in solution quality between the two representations. For example, if section S8 is 10 km, and sections S2 and S7 are 5 km long, the lower bound for layout 1 is 69.61 and 66.89 for layout 2. The results are shown in Table 8. The second representation is now considerably better; i.e., on the order of half an hour better. This was expected and it is good that the compound buffer approach captures this phenomena. Considering that the representation does not allow some feasible traffic flows, this result is also very good. This demonstrates that simplified networks are often capable of providing good quality solutions.

**BUFFER REPRESENTATION FAILURES**

A sequencing approach that allows discrete decisions to be made has been taken for train scheduling. As a consequence, the size of the search space is significantly reduced and superior solutions can be obtained more easily. A primary component of the sequencing approach is the usage of capacitated buffer and compound buffers to represent passing loops and other infrastructure that would normally incur additional routing decisions. The buffer approach for representing crossing loops and other passing facilities works subject to the following conditions:

- Condition A: Entry to buffer machines is automatic.
- Condition B: All buffer occupancy violations (BOV) can be resolved by changing the position of one or more operations in the machine sequences. In other words, there is at least one feasible insertion point (position) for every operation.

Condition A stops a number of valid solutions from being obtained and is equivalent to a nondelay (i.e., forbid unforced idle time) scheduling policy. Unfortunately, some of these removed solutions may be necessary in order to satisfy condition B and to achieve schedule feasibility in some scenarios. That is, the buffer representation may fail. This is defined as a buffer representation failure (BRF). The failure of the buffer representation in particular is normally associated with blocks of unidirectional traffic on unbalanced networks. The term unbalanced refers to the situation where two adjacent sections, albeit separated by a passing loop, are of significantly different length. Put another way, one machine significantly dominates another adjacent machine in terms of processing requirements.

It should be noted that a BRF that causes a BOV is valid from a sequencing point of view, unlike other BOVs. Generally buffer occupancy violations result because infeasible sequencing decisions have been made and no amount of added operation delay...
Resolution of a BRF may cause additional BOV and BRF. Resolution of a BRF has several effects and these are as follows:

- Resolution of a BRF may cause additional BOV and BRF.
- Resolution of a BRF may resolve another BRF.
- Resolution of BRF does not affect the machine sequences.

**BRF Correction**

The number of possible BRFs may be very large and, unlike the identification of multiple overtaking conflicts (see Burdett and Kozan 2009b), it is inappropriate to try and pre-identify them. Multiple overtaking conflicts (MOC) occur when two trains traveling in the same direction successively pass each other. The congestion that is caused by this action is very detrimental in a schedule. Therefore, BRFs should only be identified in a current schedule. It should also be noted that the BOV permit scheduling option must be firstly selected; otherwise, the solution will be deemed infeasible and no changes will be allowed.

BRF inspection and correction may be ongoing or left as a last phase in a general solution strategy. If it is left until last, then the computational burden will be reduced. The only downside with this approach is that additional BOV may be caused, thus resulting in an infeasible schedule. The likelihood of BRF occurrence may be used to influence this decision or a trial-and-error approach. For example, if the mix of trains is mostly unidirectional and machine dominance occurs, then BRF occurrence is highly likely and identification and removal should be ongoing. Alternatively, if the mix of trains is roughly equal in both directions and machine dominance does not occur, then BRF inspection and removal can be ignored with relative ease.

A variety of different mechanisms can be used to ensure that no buffer occupancy violations cause buffer representation failures. Correct scheduling may firstly be achieved using a mathematical programming model instead of using the longest path algorithm. In theory this approach would easily allow unforced idle time to be chosen. This approach, however, is not used as a first resort because the computational burden of solving a mathematical programming model is typically greater than other strategies. Furthermore, it has been observed that a mathematical programming model could be quite impractical because the capacitated buffer representation must be converted back to the original representation. The primary reason for this is that buffer occupancy is not easily or efficiently modeled using analytical equations. Track usage associated with capacitated buffers must also be predetermined or solved for within the model.

Other options include artificial release times and additional disjunctive arcs. Once operations have been scheduled the artificial release times or additional disjunctive arcs (DJAs) can be removed so that further modifications can be made to the schedule. Upon closer examination, the introduction of artificial temporary release times was found to be less suitable. The main problem is that resolving a BRF may invalidate the release times added to resolve a previous BRF. Inserting additional disjunctive arcs is more generic and robust and always results in correct timings. In our approach these additional DJAs are removed as soon as operations are correctly scheduled. It is theoretically possible to leave the artificial disjunctive arcs in the disjunctive graph after evaluation if sufficient tracking information is available. This would remove the necessity to re-identify the same BRF over and over again for example when the schedule is perturbed by metaheuristics.

Formally a buffer representation failure object is a three-tuple $BRF = (MOR, arc, added)$. The first member is a reference to a machine occupancy record (MOR) that signifies the BOV. The second member is a disjunctive arc that corrects the BRF when added. The third member is a Boolean flag that signifies whether the arc has been added. A machine occupancy record is a two-tuple that consists of an interval of occupancy plus a list of operation occupants; i.e., $MOR = ([\alpha, \beta], \lambda)$.

From a top-level perspective the correction of BRF is via an iterative algorithm that performs successive identification, correction, and evaluation steps until there are no more BRFs present in the schedule. Algorithms 1–4 in the Appendix demonstrate the finer details associated with BRB handling. At each step (iteration) a single BRF is corrected but others may also be created or resolved. In the BRF identification algorithm the artificial arc is connected between the operation that leaves the buffer first and the operation that enters last provided that they are not the same operation. More precisely the successor operation of the first to leave operation is connected to the last to enter operation. The arc weight is $-p_{from} - dwell_{from} + lag_{from}$ and when subtracted from the longest path value of a train gives the exit time from the buffer. This arc has the effect of delaying (blocking) the last operations entry until the right time. The BRF identification condition utilizes the direction of travel attribute of train operations. Therefore, the predecessor and successor

<table>
<thead>
<tr>
<th>Option</th>
<th>Layout</th>
<th>CA Makespan (# BOV)</th>
<th>SA Makespan (# BOV)</th>
</tr>
</thead>
<tbody>
<tr>
<td>Forbid BOV 1</td>
<td>1</td>
<td>168.23</td>
<td>149.15</td>
</tr>
<tr>
<td>Permit BOV 1</td>
<td>1</td>
<td>121.88 (46 BOV)</td>
<td>136.03 (0 BOV)</td>
</tr>
<tr>
<td>Forbid BOV 2</td>
<td>2</td>
<td>123.76</td>
<td>107.05</td>
</tr>
<tr>
<td>Permit BOV 2</td>
<td>2</td>
<td>107.1 (100 BOV)</td>
<td>105.65 (0 BOV)</td>
</tr>
</tbody>
</table>

**Table 8** Constructive algorithm and simulated annealing results.

<table>
<thead>
<tr>
<th>Mix</th>
<th>Path</th>
<th>Trains (60, 80, 100, 120, and 160 signify train speeds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(A1, B1)</td>
<td>10 (2 × 60 + 2 × 80 + 2 × 100 + 2 × 120 + 2 × 160)</td>
</tr>
<tr>
<td>2</td>
<td>(A1, B1)</td>
<td>168.23</td>
</tr>
<tr>
<td></td>
<td>(B1, A1)</td>
<td>5 (1 × 60 + 1 × 80 + 1 × 100 + 1 × 120 + 1 × 160)</td>
</tr>
<tr>
<td></td>
<td>(A2, B2)</td>
<td>5 (1 × 60 + 1 × 80 + 1 × 100 + 1 × 120 + 1 × 160)</td>
</tr>
<tr>
<td></td>
<td>(B2, A2)</td>
<td>5 (1 × 60 + 1 × 80 + 1 × 100 + 1 × 120 + 1 × 160)</td>
</tr>
</tbody>
</table>

**Table 9** Train mixes for example 1.
machine of each operation in the BOV do not necessarily have to be the same as occur in some networks. For basic job shop scheduling problems, direction of travel information is not available because it has no physical meaning. Therefore, a different BRF condition is required. An alternative condition is that all operations of the BOV have the same predecessor machine or all operations have the same successor machine.

An Example

Consider ten trains traveling at 100 km/h across a serial segment of track with three sections that are separated by two standard passing loops. The passing loops are 1 km long and the section lengths, respectively, are 5, 50, and 7 km. The original schedule for this situation using the capacitated buffer modeling approach is shown in a line chart in Figure 12(a). In this chart the vertical axis represents the distance traveled by trains and also shows the demarcation between sections. The horizontal axis is the elapsed time and the train trajectories are the sloping lines. Horizontal train trajectories signify that a train is stationary. From this figure we can see that the 50-km middle section dominates the adjacent sections greatly, thus causing a large number of buffer overflows when unidirectional traffic is scheduled as early as possible. In Figure 12(a), thirteen BRFs are caused. By applying the BRF correction approach as described above, the solution can be made feasible and this schedule is shown in Figure 12(b). The schedule shown in Figure 12(b) can be obtained by appropriately delaying the exit time of the last seven trains from the first section. Therefore, all of the BRFs are resolved without changing the original sequence of train movements.

FULL-SIZE CASE STUDIES

In this section a number of complete scenarios are considered. These scenarios contain basic elements that are often repeated and consequently the compound buffer representations are not shown. These scenarios were chosen because they are typical of systems found in practice. They are used to test the sequencing techniques (i.e., both constructive and metaheuristics) in a more general sense. How hard these types of problems are to solve and what quality of solution is obtainable is identified here. The sectional running times used in the experiments are based upon some chosen common train speeds and hence they are theoretical times. The lengths of each section in the track layouts were chosen arbitrarily but they are of reasonable and realistic lengths.

For each layout two track usage protocols (variations) were considered. The first considers predominantly unidirectional flow on each line and the second considers predominantly bidirectional flow. It should be noted that in the three-track unidirectional-only problem there is a choice of whether to have two up lines and one down line or two down lines and one up line. In order to not have to make this decision, the middle line was allowed bidirectional traffic. It should also be noted that a third track usage protocol is possible but is not easily enforced. This special case occurs when bidirectional traffic is allowed on each line but only in terms of blocks of unidirectional traffic. In other words, the schedule associated with each line should consist of two unidirectional sub-schedules appended together. The details of the considered scenario are now reported.

Examples

- **Example 1**: We consider two parallel lines (between location A and B) connected by four corresponding and alternating pairs of crossover point. From left to right the lengths of the five resulting sections are 2.25, 7.65, 9.38, 5.46, and 5.77 km, respectively. The distance between the crossover points is 1 km and this is assumed for all the examples. The train mixes given in Table 9.

- **Example 2**: We consider two parallel lines (between location A and B) connected by nine corresponding and alternating pairs of crossover point. From left to right the lengths of the ten resulting sections are 17.87, 6.68, 13.78, 17.78, 10.43, 2.39, 6.8, 10.97, 14.41, and 2.28 km. The train mix in example 1 is again used.

- **Example 3**: We consider three parallel tracks (between A and B) connected by four corresponding and alternating pairs of crossover points between the first and second line and the second and third line, respectively. In other words, each

<table>
<thead>
<tr>
<th>Mix</th>
<th>Path</th>
<th>Trains (60, 80, 100, 120, and 160 signify train speeds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(A, B)</td>
<td>(10 (2 \times 60 + 2 \times 80 + 2 \times 100 + 2 \times 120 + 2 \times 160))</td>
</tr>
<tr>
<td></td>
<td>(D, C)</td>
<td>(10 (2 \times 60 + 2 \times 80 + 2 \times 100 + 2 \times 120 + 2 \times 160))</td>
</tr>
<tr>
<td></td>
<td>(B, D)</td>
<td>(5(1 \times 60 + 1 \times 80 + 1 \times 100 + 1 \times 120 + 1 \times 160))</td>
</tr>
<tr>
<td></td>
<td>(D, B)</td>
<td>(5(1 \times 60 + 1 \times 80 + 1 \times 100 + 1 \times 120 + 1 \times 160))</td>
</tr>
<tr>
<td>2</td>
<td>(A, D)</td>
<td>(5(1 \times 60 + 1 \times 80 + 1 \times 100 + 1 \times 120 + 1 \times 160))</td>
</tr>
<tr>
<td></td>
<td>(D, A)</td>
<td>(5(1 \times 60 + 1 \times 80 + 1 \times 100 + 1 \times 120 + 1 \times 160))</td>
</tr>
<tr>
<td></td>
<td>(B, D)</td>
<td>(5(1 \times 60 + 1 \times 80 + 1 \times 100 + 1 \times 120 + 1 \times 160))</td>
</tr>
<tr>
<td></td>
<td>(D, B)</td>
<td>(5(1 \times 60 + 1 \times 80 + 1 \times 100 + 1 \times 120 + 1 \times 160))</td>
</tr>
</tbody>
</table>

Table 10: Train mixes for Example 3.

<table>
<thead>
<tr>
<th>Mix</th>
<th>Path</th>
<th>Trains (60, 80, 100, 120, and 160 signify train speeds)</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>(A, B)</td>
<td>(A1, B1) 10 (2 \times 60 + 2 \times 80 + 2 \times 100 + 2 \times 120 + 2 \times 160)</td>
</tr>
<tr>
<td></td>
<td>(B, D)</td>
<td>(A1, B1) 5(1 \times 60 + 1 \times 80 + 1 \times 100 + 1 \times 120 + 1 \times 160)</td>
</tr>
<tr>
<td></td>
<td>(A, B)</td>
<td>(A2, B2) 5(1 \times 60 + 1 \times 80 + 1 \times 100 + 1 \times 120 + 1 \times 160)</td>
</tr>
<tr>
<td></td>
<td>(B, D)</td>
<td>(A2, B2) 5(1 \times 60 + 1 \times 80 + 1 \times 100 + 1 \times 120 + 1 \times 160)</td>
</tr>
<tr>
<td>2</td>
<td>(A, B)</td>
<td>(A3, B3) 5(1 \times 60 + 1 \times 80 + 1 \times 100 + 1 \times 120 + 1 \times 160)</td>
</tr>
<tr>
<td></td>
<td>(B, D)</td>
<td>(A3, B3) 5(1 \times 60 + 1 \times 80 + 1 \times 100 + 1 \times 120 + 1 \times 160)</td>
</tr>
</tbody>
</table>

Table 11: Train mixes for Example 5.
Figure 12  Original schedule (line chart) containing BRF and corrected schedule containing none.
crossover point connects two lines and therefore there are eight pairs of crossover points in total. From left to right the lengths of the five resulting sections are 10.5, 13.5, 9, 7, and 6 km. The train mixes are as given in Table 11.  
- **Example 4:** This example is similar to Example 3. The only difference is that the crossover points connect all three lines. In other words, a crossover point starts on the first track and finishes on the third or vice versa. The train mix in example 3 is again used.  
- **Example 5:** This example considers three parallel lines that merge into one track as was addressed earlier in this article. There are, however, two corresponding pairs of alternating crossover points prior to the merge. From left to right the section lengths are 2, 7, 10, and 5 km, respectively. The train mixes are as given in Table 11.

### Results

The results of the numerical investigations are summarized in Table 12. The final obtained solutions are very good quality and this conclusion is substantiated by the lower bound values and a visual analysis of the solutions. The BOV permit solutions obtained by the constructive algorithm also provide a good indication of where the best solution lies and previous numerical investigations have substantiated this conjecture.

Table 12 makes evident a number of things worthy of mention. For example, the requirement for compound moves in many of the examples means that the solutions cannot be improved easily or by much. The schedules obtained for the bidirectional traffic mixes do not utilize the system as well as schedules that only allow purely unidirectional traffic; however, they are still of good quality. In other words, the considered railway infrastructure can be best utilized by unidirectional traffic. The observed relative difference was between 15 and 30 percent. The real difference, though, may be smaller or larger and can only be determined by solving the scheduling problems to optimality.

When scheduling bidirectional traffic, there was always a way of inserting operations with the CA. When the traffic was predominantly unidirectional, however, the CA was unable to obtain a full solution that was feasible for four of the five examples using the forbid BOV option. This is at complete odds with its usual exhibited performance. It is concluded that more accurate timings are required in order to satisfy the compound buffer occupancy conditions and the current nondelay scheduling policy hinders this. The difficulties in obtaining a feasible solution are layout dependent when traffic flow is unidirectional. In other words, the lengths of the sections, the compound buffer representation, and the dominance relationships affect the complexity and the success of the scheduling strategies.

It should be noted that BRF correction on the constructive algorithm solution, for example, 1 (mix 1), removes all but six BOV. The solution can be further reduced to 78.88 min by SA but four BOVs still remain after BRF correction. The BOVs, however, can be removed by adding delays manually. BRF correction on example 5 for mix 1 removes all 15 BOV. The one-off application of BRF corrective measures is deemed quite successful, though further effort is warranted to find a more general strategy for resolving BRF and some normal BOV.

The permit BOV scheduling option resulted in far fewer BOV than has been observed previously in standard problems that do not have parallel lines and compound buffers. Furthermore, they appear to be more easily resolved. This is indicated by the fact that high numbers of BOVs were obtained by the constructive algorithm but zero remained after the application of simulated annealing in all but two occasions.

It is interesting to note that the lower bound values for the unidirectional traffic problems are slightly higher than the bidirectional traffic problem though the reverse relationship occurs with respect to schedule quality after the application of the metaheuristic.

### Conclusions

An innovative modeling device that has been called a compound buffer was introduced in this article. The compound buffers are integrated into an innovative hybrid job-shop sequencing approach for train scheduling. The purpose of
compound buffers is to model more complicated railway tracks and networks such as parallel lines with crossover points. A particular benefit of the compound buffer approach is that movement between lines is made possible without the need to make additional routing decisions. Similarly, compound buffers allow passing to occur that would not otherwise be possible. Our approach can also be used to analyze different parallel track topology and operating protocols. It is a superior alternative to conventional simulation strategies because near optimal train movements are determined, whereas train movements are dictated by preestablished myopic rules in simulation.

The case studies presented show that different representations involving compound buffers can sometimes be devised for particular situations. This may be seen as an advantage or a disadvantage depending on one’s point of view. For example, it is advantageous in one sense because it demonstrates the flexibility of the approach and its ability to model very complex railway features. On the other hand, this may be disadvantageous because the most efficient and safe way to model a given scenario is not entirely obvious. This last issue, however, is expected to be resolved in time as experience using this approach is gained and quirks of the approach are identified. Similarly, a library of scenarios and correct modeling strategies will be recorded. It should also be noted that in many situations there will only be one compound buffer approach that is viable and the above issue will not arise.

In order to utilize compound buffers the system must be decomposed and then linked correctly. The suggested decomposition strategy is to model each parallel line individually, taking into account passing possibilities facilitated by connected lines. Compound buffers are then used to link these separate representations. It is the linking stage that is most difficult. For example, some passing possibilities may need to be removed in order to synchronize the representations for adjacent tracks. Similarly, some representations may need to be simplified to reduce the complexity of the overall representation.

The safety issues associated with the use of compound buffers are a continuing source of research. It is not completely known how accurate and safe a schedule is when the system is a modeled using compound buffer; i.e., will trains collide at crossover points, junctions, and other intersection points. It should, however, be kept in mind that it is difficult to guarantee that any schedule is completely valid regardless of the technique and representation used to create it difficult environments such as railways. Furthermore, it is almost impossible to carry out a schedule exactly without change in the real world even if it could be identified as being completely feasible. In practice, small variations creep into the sectional running times and change the timings in the schedule.

The numerical investigations have demonstrated that parallel lines can be utilized effectively when bidirectional traffic is allowed, though unidirectional traffic is still superior. The difference in the utilization level is somewhere between 15 and 30 percent. The numerical investigations have also demonstrated that the nondelay scheduling policy may need to be relaxed at times in order to maintain schedule feasibility or to obtain better solutions. A more general approach for adding delays is required and is a source of continuing research.

REFERENCES


APPENDIX A

The following procedures perform buffer representation failure correction.
• Algorithm 1: RemoveBRF() // Iterative algorithm to coordinate the removal of all buffer representation failures. Note: BRFs are not resolved chronologically but could be via an alternative approach

Begin
Define set of BRF, namely brfs;
IdentifyBRF(); // Cal procedure to identify all BRF
while(|brfs| > 0) begin
CorrectBRF(); // Add arcs to remove current BRF
Evaluate(); // Re-evaluate the schedule
IdentifyBRF(); // Determine new list of BRFs
end
DisposeBRFObjects(); // Remove arcs and start again
End

• Algorithm 2: DisposeBRFObjects() // Procedure to destroy BRF objects and remove added arcs

Begin
for(k = 1, .., |brfs|) do begin
if (brfs_k.arcIn) then
The arc has been added
Remove the arc from the disjunctive graph;
delte brfs_k; // Delete the object
end
End

• Algorithm 3: CorrectBRF() // Algorithm to resolve all “new” BRFs

Begin
for(k = 1, .., |brfs|) do begin
if (~brfs_k.arcIn) then
The BRF is new as the arc has not been added
brfs_k.arcIn = true; //Reset the flag
Add brfs_k.arc to the disjunctive graph;
end
End

• Algorithm 4: IdentifyBRF (int counter)

Begin  
// Procedure to coordinate the identification of buffer representation failures.
// BRF Condition: Each operation of the BOV has the same direction of travel on the buffer machine
// Assumption: Predecessor and successor machines are not buffers
counter = 0; // Reset counter for the number of newly found BRF
for(k = 1, .., |BOV|) do begin  // For each BOV check BRF conditions
isBRF = true; // Assume the BOV causes a BRF unless shown otherwise
i = 1; // Start at the first operation
while(i ≤ MOR_k, nOcc and isBRF) begin
opn = MOR_k, occ ;
if (i = 1) begin

intelligence transportation systems vol. 13 no. 4 2009
\[ \text{dot} = \text{opn.dot}; \] //Set direction of travel parameter
\[ \text{first} = \text{last} = \text{opn}; \] //Set operation parameters
\[ \text{end} \]
\[ \text{else begin} \]
\[ \text{isBRF} = (\text{opn.dot} = \text{dot}); \] //Check for common direction of travel
\[ \text{if} (\text{opn.entry} > \text{last.entry}) \text{last} = \text{opn}; \] //Find operation that entered buffer last
\[ \text{if} (\text{opn.exit} < \text{first.exit}) \text{first} = \text{opn}; \] //Find operation that leaves buffer first
\[ \text{end} \]
\[ i + +; \] //Go to the next operation
\[ \text{end} \]
\[ \text{if} (\text{isBRF}) \]// Create a BRF object
\[ \text{begin} \]
\[ \text{from} = \text{first.next}(); \] //The successor operation of first
\[ \text{weight} = -\text{from.pt} - \text{from.dwell} + \text{from.lag}; \] //Define the arc weight
\[ \text{Addbrf} = (\text{MOR}_k, (\text{from}, \text{last}, \text{weight}), \text{false}) \] to the set of BRF.
\[ \text{counter} + +; \] //Increment the number of newly found BRF
\[ \text{end} \]
\[ \text{end} \]
\[ \text{End} \]