BUS PRIORITY IN SCOOT

by G T Bowen

Prepared for: Driver Information Traffic Management Division, DOT
Project: Bus priority in UTC systems (UG17)

Copyright Transport Research Laboratory 1997. All rights reserved.
The information contained herein is the property of the Transport Research Laboratory. This report has been produced by the Transport Research Laboratory under a contract placed by the Department of Transport. Any views expressed in it are not necessarily those of the Department. Whilst every effort has been made to ensure that the matter presented in this report is relevant, accurate and up-to-date at the time of publication, the Transport Research Laboratory cannot accept any liability for any error or omission.

First Published 1997
ISSN 0968-4107
<table>
<thead>
<tr>
<th>CONTENTS</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>Executive Summary</td>
<td>1</td>
</tr>
<tr>
<td>Abstract</td>
<td>3</td>
</tr>
<tr>
<td>1. Introduction</td>
<td>3</td>
</tr>
<tr>
<td>2. Description of the bus priority system in SCOOT</td>
<td>3</td>
</tr>
<tr>
<td>2.1 General</td>
<td>3</td>
</tr>
<tr>
<td>2.2 Bus detection</td>
<td>4</td>
</tr>
<tr>
<td>2.2.1 Transponders</td>
<td>4</td>
</tr>
<tr>
<td>2.2.2 Automatic Vehicle Location</td>
<td>4</td>
</tr>
<tr>
<td>2.3 The queue model</td>
<td>4</td>
</tr>
<tr>
<td>2.3.1 Queueing by buses</td>
<td>4</td>
</tr>
<tr>
<td>2.3.2 Queueing by other vehicles</td>
<td>4</td>
</tr>
<tr>
<td>2.3.3 Link saturated green length</td>
<td>4</td>
</tr>
<tr>
<td>2.3.4 The link extension saturated green length</td>
<td>5</td>
</tr>
<tr>
<td>2.3.5 Node required cycle length</td>
<td>5</td>
</tr>
<tr>
<td>2.4 Bus optimisation</td>
<td>5</td>
</tr>
<tr>
<td>2.4.1 Overview</td>
<td>6</td>
</tr>
<tr>
<td>2.4.2 Calculation of time to clear stopline</td>
<td>6</td>
</tr>
<tr>
<td>2.4.3 Calculation of bus queueing time</td>
<td>6</td>
</tr>
<tr>
<td>2.4.4 Extensions</td>
<td>7</td>
</tr>
<tr>
<td>2.4.5 Recalls</td>
<td>7</td>
</tr>
<tr>
<td>2.4.6 Recovery</td>
<td>8</td>
</tr>
<tr>
<td>3. Implementation</td>
<td>11</td>
</tr>
<tr>
<td>3.1 Simulation</td>
<td>11</td>
</tr>
<tr>
<td>3.2 Installations</td>
<td>12</td>
</tr>
<tr>
<td>3.2.1 London</td>
<td>12</td>
</tr>
<tr>
<td>3.2.2 Southampton</td>
<td>12</td>
</tr>
<tr>
<td>3.3 Field trial</td>
<td>12</td>
</tr>
<tr>
<td>4. Conclusions</td>
<td>13</td>
</tr>
<tr>
<td>5. References</td>
<td>14</td>
</tr>
</tbody>
</table>
EXECUTIVE SUMMARY

A version of SCOOT has been developed which incorporates bus priority. The logic of bus priority is contained within the SCOOT kernel which has been modified by TRL. The bus priority optimiser implements green-time extensions and priority recalls; the decision to make an extension or recall and the limits imposed on them is decided with reference to the degree of saturation or spare capacity of the network, and so makes full use of SCOOT capabilities. Buses are modelled and account is taken of queues which may delay buses.

Simulation tests were performed at TRL using the STEP simulation, which was enhanced to model buses and bus priority. The simulation was run for a range of operating conditions, with congestion levels, bus demands, bus conflicts and target saturation levels being varied and different combinations of extensions and recalls being allowed. The simulation showed savings in bus passenger delay of typically 20% to 30%, with greater levels being possible at lightly trafficked junctions. In one simulated network, extensions were shown to produce savings of about 24% in bus passenger delay with the least disruption to other traffic (no significant increase in car passenger delay); recalls produced further savings in bus passenger delay of about 8% but were more disruptive to the other traffic (about 4% increase in car passenger delay). The simulation produced recommendations for the SCOOT system to be tested during the field trials, covering the target saturation levels and recovery logic to be used.

The SCOOT bus priority system was installed in London by TCSU. There are about 5000 London buses fitted with transponders. Special bus detectors are located about 70 metres upstream of the junction, depending on the location of bus stops. Microprocessor-based traffic signal controllers contain bus detection equipment, and new logic to perform local green extensions for bus priority has been included. The central computer contains the new SCOOT kernel together with software supporting the bus priority database and parameter values, and software to communicate with the bus priority facilities in the controllers.

The UTC system in Southampton contains the same SCOOT kernel as in London, but surrounding software has been written to interface between SCOOT and the Bus-Tracker AVL system supplied by GEC, which was installed as part of the ROMANSE project to supply passenger information at bus stops. Nine junctions in Southampton have been configured to allow bus priority within SCOOT using central extensions and recalls. Local extensions are not used in Southampton so modifications to the traffic signal controllers were not required. Local extensions would also have required some means of transmitting bus information to the controller.

A field trial has been conducted in Camden Town, London; Camden Town is a relatively congested network and bus delay savings averaged 5 seconds per junction (22%). The maximum delay saving to buses was achieved using local extensions and central recalls. Delay to other traffic was small with local extensions at all traffic levels, but recalls at higher traffic levels caused disbenefit to other traffic averaging 5 seconds per vehicle.
BUS PRIORITY IN SCOOT

ABSTRACT
A version of SCOOT has been developed which incorporates bus priority. The logic of bus priority is contained within the SCOOT kernel which has been modified by TRL. The bus priority optimiser implements green-time extensions and priority recalls; the decision to make an extension or recall and the limits imposed on them is decided with reference to the degree of saturation or spare capacity of the network, and so makes full use of SCOOT capabilities. Buses are modelled and account is taken of queues which may delay buses.

Simulation tests showed savings in bus passenger delay of typically 20% to 30%, with greater levels being possible at lightly trafficked junctions. In one simulated network, extensions were shown to produce savings of about 24% in bus passenger delay with the least disruption to other traffic (no significant increase in car passenger delay); recalls produced further savings of about 8% in bus passenger delay but were more disruptive to the other traffic (about 4% increase in car passenger delay). The simulation produced recommendations for the SCOOT system to be tested during the field trials, covering the target saturation levels and recovery logic to be used.

The SCOOT bus priority system has been installed in London and Southampton. In London, buses have been fitted with transponders and special bus detectors are installed in the road at ten junctions in Camden Town. The central computer contains the new SCOOT kernel, and logic to perform local green extensions for bus priority has been included in microprocessor controllers. Southampton uses the same SCOOT kernel, but connected to a GEC BusTracker AVL system to provide bus detection. Nine junctions in Southampton have been configured to allow bus priority within SCOOT.

A field trial in Camden Town, London showed bus delay savings averaging 5 seconds per junction (22%). The maximum delay saving to buses was achieved using local extensions and central recalls. Delay to other traffic was small with local extensions at all traffic levels, but recalls at higher traffic levels caused disbenefit to other traffic averaging 5 seconds per vehicle.

1. INTRODUCTION
Normally, Urban Traffic Control systems such as SCOOT treat all vehicles equally, but for both policy and economic reasons it is often desirable to give priority to public service vehicles such as buses. Previous versions of SCOOT could give only 'passive' priority to buses by giving a general preference to all traffic on a particular route such as a bus route. Research within this project has now developed and tested a new facility to give buses 'active' priority as they arrive at the traffic signal.

Bowen et al (1994) describe the system in general. Detection of buses can be by selective vehicle detectors (SVD) or by an automatic vehicle location (AVL) system. Once detected, buses are modelled as queuing with other traffic and are given priority by means of green extensions or early recall of succeeding stages. After priority has been given, a recovery period takes place to allow the signal timings to revert to the normal SCOOT optimisation. A more detailed description is provided in chapter 2.

Most of the work on Bus Priority in UTC Systems has been performed in conjunction with the DRIVE II project PROMPT, involving collaboration with partners in the UK, Italy and Sweden. In the UK, work has concentrated on the SCOOT system in London, with work in Southampton mainly performed under the DRIVE II project ROMANSE. The original design of bus priority in SCOOT was derived from earlier work in SELKENT. The main features of the final system were identified, namely:

- green-time extensions, both local and central
- priority recalls
- constraints based on degree of saturation

Other features became clearer during the simulation, such as:

- importance of recovery (or 'resynchronisation') logic
- modelling of queueing delay to buses

2. DESCRIPTION OF THE BUS PRIORITY SYSTEM IN SCOOT

2.1 GENERAL

The logic of bus priority is contained within the SCOOT kernel which has been modified by TRL. The bus priority optimiser implements extensions and recalls; the decision to make an extension or recall and the limits imposed on them is decided with reference to the degree of saturation or spare capacity of the network, and so makes full use of
2.2 BUS DETECTION

The logic of bus priority does not depend on the method of detecting buses, which can vary from system to system. The method of detection can be based on transponders; it can be derived from an Automatic Vehicle Location (AVL) system; or it can be based on some other system which provides suitable information. SCOOT requires an indication of the presence of a bus on a link, the free flow journey time of the bus to the stopline, and the queue clear time of a vehicle queue reaching to the bus detection position. This enables SCOOT to model the bus arrival at the stopline, the delay to the bus caused by any vehicle queue ahead of the bus, and the delay caused by the traffic signals. But SCOOT will not model a bus waiting at a bus stop, so any detection information should relate to a bus after a bus stop.

2.2.1 Transponders

Buses are detected by a special bus detector which does not detect other vehicles. The detector is situated on a link downstream of any bus stop, and provides one bit per second indicating the presence or absence of a bus. The presence of a bus can be indicated in consecutive seconds and this is taken to be one bus only; a gap of one second will allow the detection of another bus. Transponders can be used to provide accurate information at a position on the link which is pre-determined to give the best information.

2.2.2 Automatic Vehicle Location

AVL can provide more information than a transponder system, and can be more flexible in how and when it provides the information. It is not known how reliable the information will be, and this may vary between different systems.

SCOOT will make the best use it can of information supplied to it, but will not initiate a request for information nor will it schedule the supply of information; these features are considered to be a function of the AVL system. The information supplied does not have to mimic that of a transponder system; in particular, the presence of a bus does not have to be indicated at a fixed location on a link, and information on a bus can be presented more than once on the same link. It is important, however, that the bus has passed any bus stop as SCOOT cannot model the time spent at bus stops.

2.3 THE QUEUE MODEL

Buses are modelled on the same links as other traffic. A record is made by SCOOT whenever information is supplied about a bus by the detection routines. If there is no identity of the bus has previously been recorded by SCOOT, the information is taken to be an update of information about the same bus and the queue model of that bus is adjusted to take into account this further information. If the bus is still on the same link, the time from stopline is updated, but if the bus is now on a different link from that modelled by SCOOT, the record of the bus is also moved to the new link.

2.3.1 Queueing by buses

Every second, each bus is moved one second nearer to the stopline. Once the time from the stopline is zero, a copy is made of the vehicle queue on the main link, and the bus is not modelled as crossing the stopline until the vehicle queue ahead of the bus has discharged.

The queue on the link which is in front of the bus detector is modelled as blocking the bus while that queue lasts. This is indicated as 'Exit Blocking' although it refers to blocking by a queue on the same stretch of road as the bus, and ensures that the bus does not clear the lights until traffic ahead of it has discharged. By this means, a bus is modelled as queueing in the same queue as the main traffic.

2.3.2 Queueing by other vehicles

Queueing by other vehicles on the link is not affected by buses except insofar as they are detected on the normal vehicle detector. The queue model now operates once per second on each link, instead of the previous method whereby each link was processed every four seconds. In other respects the queue model for ordinary traffic operates as before.

2.3.3 Link saturated green length

The saturated green length of a link is calculated as follows:

\[ g_{sat} = \frac{Q_r + q_{RG}}{s} \]

where:

- \( g_{sat} \) = the saturated green length of a link (s)
- \( Q_r \) = the queue at start of red for the link (lpu)
- \( q_{RG} \) = the inflow during the red-green period (lpu)
- \( s \) = the saturation occupancy of the link (lpu s\(^{-1}\))

This is then smoothed every green to give the smoothed saturated green length as follows:

\[ \bar{g}_{sat} = \alpha \bar{g}_{sat} + (1 - \alpha) g_{sat} \]

where:

- \( \bar{g}_{sat} \) = the smoothed saturated green length of a link (s)
- \( \alpha \) = the smoothing factor (0.5).

This smoothed value is used in the recall logic and in the extension logic.
2.3.4 The link extension saturated green length

For each link, the extension saturated green length $g_e$ is calculated as follows:

$$\bar{g}_e = \frac{100\underline{g}_{sat}}{x_e}$$

where:

$x_e$ = the extension target saturation of a node (%)

2.3.5 Node required cycle length

The node required cycle length $c_{req}$ is then calculated from the extension saturated green lengths $\bar{g}_e$ of each link at the node. Consider Example 1 below.

SCOOT indicates that bus priority is in operation at a node by setting the 'bus active' flag, and there is a SCOOT message to output this flag. Other SCOOT messages provide more detailed information on the operation of bus priority. There may also be displays which summarise bus information on one screen, such as the btsj (bus rest set for junction) display in London.

**2.4 BUS OPTIMISATION**

2.4.1 Overview

Bus priority optimisation occurs when there is a change in information, such as when:

- a bus is detected
- a bus clears the stopline (as indicated by the bus priority model)
- exit blocking occurs (as indicated by the SCOOT model)

The bus priority optimisation process decides whether to give an extension or recall for a bus. If possible, an extension is granted. An extension extends the green on the current stage so as to allow a bus through the signals. If this is not possible, because the current stage is red to the bus or because the maximum possible extension would not allow the bus through the signals, a recall is granted. A recall causes succeeding stages to occur earlier than they otherwise would so that a bus clears the signals earlier.

**EXAMPLE 1**

<table>
<thead>
<tr>
<th>Link</th>
<th>$\bar{g}_e$</th>
<th>Start stage</th>
<th>End stage</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>10</td>
<td>1</td>
<td>2</td>
</tr>
<tr>
<td>B</td>
<td>50</td>
<td>1</td>
<td>3</td>
</tr>
<tr>
<td>C</td>
<td>25</td>
<td>2</td>
<td>3</td>
</tr>
<tr>
<td>D</td>
<td>15</td>
<td>3</td>
<td>1</td>
</tr>
</tbody>
</table>

Number of stages: 3
Intergreens: 5
Area start lag: 2
Area end lag: 3

So each stage requires $5 + 2 \times 3 = 4$ seconds more than its green length to ensure that no link is saturated to more than the extension target saturation $x_e$.

This makes the stage length requirements as follows:

![Diagram showing stage length requirements]

which gives a $c_{req}$ of 73 seconds. This is used in the optimisation when determining whether an extension can be granted.
If more than one bus is detected, the buses are handled in stage order, so that buses requiring an extension of the current stage are given priority first, and buses requiring a recall to a later stage are handled later. In this way, extensions are preferred as they disrupt other traffic less than recalls.

After the extension or recall has finished, recovery takes place which causes the signal timings to become resynchronised with the main SCOOT optimisers.

The operator can specify target saturations for extensions, recalls and recovery. These restrain bus priority so that it does not cause the saturation of other traffic to become greater than the appropriate target. This means that bus priority can be loosely constrained when there is plenty of spare capacity at a junction, but heavily constrained or not permitted at all when the junction is oversaturated.

SCOOT maintains its normal optimisation process and decides on the stage change times in the normal way, but the Bus Priority Optimiser overrides these SCOOT stage change times when it operates an extension or a recall.

### 2.4.2 Calculation of time to clear stopline

The second a bus is detected, the bus optimiser performs a calculation and determines whether an extension or recall is wanted to assist the bus. The optimiser calculates how long the bus needs to clear the stopline as follows:

\[ t = t_c + t_Q - (t_{RX} + t_{AE} + t_{LE}) + \delta t \]

where:

- \( t \) = Time required from now for bus to clear stopline.
- \( t_c \) = The journey time (cruise time) of the bus from the detector to the stopline.
- \( t_Q \) = The bus queueing time. This is the time the bus has to queue behind a queue of ordinary traffic being modelled on the main link.
- \( t_{RX} \) = The RX lag. The transmission delay from detection to computer.
- \( t_{AE} \) = The area end lag. The lag of usually 3 seconds from the start of amber to end of effective green.
- \( t_{LE} \) = The link end lag. A ± adjustment to the area end lag specific to the link. Usually 0.
- \( \delta t \) = The bus vary, typically about 3 seconds.

The bus vary \( \delta t \) is a constant value which allows for several factors:

- variations in the cruise time \( t_c \) for different buses
- variations in the RX lag \( t_{RX} \)
- the amount of leeway allowed to a bus at the end of effective green

Typically the area end lag is 3 seconds and the amber is 3 seconds long, so the end of effective green occurs at the end of the amber on street. A value for the bus vary \( \delta t \) of 3 seconds would allow a critical bus to receive an amber just as it crosses the stop line. Higher values can be used if it is wished to allow for variations in the cruise time and RX lag. Essentially, the bus vary is an additive constant to the cruise time but it is preferable to maintain it as a separate variable to allow easy comparison of street and model.

The value of \( t \) for a particular link varies only with the bus queueing time \( t_Q \), and if this is 0 then \( t \) is constant for any particular link.

The following two conditions need to be satisfied before an extension can be granted:

\[ t - t_{NS} \leq a \]
\[ t - t_{NS} \leq c - c_{req} \]

where:

- \( a \) = The bus authority. This is the maximum permitted extension and would typically be a value in the range 10 to 20 seconds.
- \( t_{NS} \) = The time to next stage.
- \( c \) = The node cycle time.
- \( c_{req} \) = The node required cycle length.

This states that the actual extension required does not exceed the maximum allowed by the bus authority, and that there is sufficient spare capacity at the node. The amount of the extension is considered as being taken from the succeeding stages. If these conditions are satisfied the bus optimiser indicates that an extension is appropriate, otherwise it indicates a recall.

### 2.4.3 Calculation of bus queueing time

The bus queueing time is calculated by running the model on the main link forward in time for the bus journey time. This is necessary because in SCOOT all modelling is performed as at the stopline. At this point the front and back of queue on the main link are determined, and any queue which exists in front of the bus detector is divided by the saturation occupancy of the main link to give the length of time the bus will be delayed, i.e. the bus queueing time. It is possible to run the model forward in time because the SCOOT detector on the main link is before the bus detector, and so future information is already available in the short-term profile.
2.4.4 Extensions

This may not actually affect the running stage at all, depending on the time to next stage and whether SCOOT makes optimisation decisions which affect the start time of the next stage. An extension causes the current stage to continue by inhibiting any stage change requested by SCOOT until all buses on the bus link have cleared the stopline. If the main link becomes exit blocked the extension is abandoned, as otherwise the extension could last indefinitely. Once the extension has completed, recovery takes place. The amount of time allocated for an extension is constrained by the ‘bus authority’ which is set by the operator. If a bus is detected which would not clear the signals with the constrained extension, a recall is requested instead.

Fig 2-1 shows an example of an extension. The symbol ★ indicates the detection of a bus and the symbol □ indicates the bus crossing the stopline. The extension is shown as starting when the bus is detected; this is because from that time bus priority is controlling the signal timings.

A bus is detected which requires stage 1. This results in stage 1 being extended, and stage 2 appearing later than would otherwise be the case. The signals are then timed so that stage 2 is called at the earliest moment while allowing the bus to cross the stopline. If the bus journey time given to SCOOT, the queue model on the link is accurate, and the bus vary is given as 0, the bus should cross the stopline at the end of the end lag, ie as the lights turn from amber to red. The bus vary would normally be given a value of about 2 to give some flexibility and to allow for variability in journey time.

2.4.4.1 Local extensions

The SCOOT kernel decides, based on the degree of saturation of a junction, when it is permissible for the controller to implement an extension locally and sends an EP (Extension Permit) bit to street during this time. The controller will implement an extension when a bus is detected during the green provided the EP bit is present. The controller then sends an EC (Extension Confirm) bit back to SCOOT to indicate that it is implementing an extension locally.

Implementing extensions locally avoids the transmission delay to and from the central SCOOT computer and so is expected to give enhanced benefits to buses.

2.4.5 Recalls

A recall will cause succeeding stages to be called until a stage runs which is green to the bus. When calling succeeding stages the following conditions apply:

- All SCOOT stages are called in sequence and no stages are skipped by SCOOT.
- Each stage must run for at least its minimum length.
- If exit blocking is detected, which would nullify any predictions about when the bus would clear the stopline, any bus priority in operation is abandoned and recovery is started.
- A recall will not cause a stage to run longer than its SCOOT stage length as determined in the normal way by the SCOOT optimisers.
- If a recall target saturation has been specified (ie it is not 0), each stage will be maintained until each link has received sufficient green to satisfy the target saturation or until the normal SCOOT stage length has been reached.

Fig 2-2 shows an example of a recall. A bus is detected during stage 2, so stage 2 is terminated early, stage 3 runs short and stage 1 appears earlier than would otherwise be the case. Then stage 1 is maintained until the bus can cross the stopline. At this point the recall is over, and recovery can take place.

If a recall target saturation is specified, a recall green length is calculated as follows for each non-faulty link:

\[
g_r = \frac{100g_{sat}}{x_t}
\]

where:

- \(g_r\) = recall green length for the link (s)
- \(x_t\) = recall target saturation (%)

The SCOOT stage will not change unless the SCOOT stage length has expired or each link which is green has received at least its appropriate \(g_r\) amount of green.

<table>
<thead>
<tr>
<th>Without</th>
<th>1</th>
<th>2</th>
</tr>
</thead>
<tbody>
<tr>
<td>With</td>
<td>1</td>
<td>★</td>
</tr>
<tr>
<td>Bus</td>
<td></td>
<td>□</td>
</tr>
<tr>
<td>Extension</td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

Fig. 2-1 Example of an extension
When the desired stage has been reached, it is then maintained until all buses on the link have cleared the stopline in the same way as for an extension. At this point the recall has completed and recovery starts.

### 2.4.6 Recovery

Recovery is the process of resynchronisation with the normal SCOOT stage timings after bus priority has finished. Four methods of recovery have been implemented. Which method operates in any particular case is configurable; different methods can be configured for use after extensions and after recalls. The configuration can be on a node basis but normally the same methods would be used throughout the area.

In the following examples, the bus is assumed to require stage 1 and an extension is given when the bus arrives during this stage; otherwise a recall is given. The symbol ★ indicates the detection of the bus, the symbol □ indicates the bus crossing the stopline, and the duration of events is shown by ⌊---⌋. A recall is marked as ⌊---⌋ where the first part of the line indicates the recall proper and the second part of the line show the phase where the desired stage has been reached and is maintained until the bus crosses the stopline.

Recovery occurs after the extension or recall has finished. Recovery is not complete until the correct stage is operating and its minimum length has expired; the period of recovery is shown by recovery ⌊---⌋.

The period where SCOOT is overridden is shown by override ⌊---⌋. The period of override starts when bus priority imposes a stage which is different from the stage which SCOOT optimisation would request, and ends at the end of recovery when the normal SCOOT stage is again operating and its minimum length has expired.

#### 2.4.6.1 DN recovery

Do nothing (DN) recovery is the simplest in concept. It amounts to accepting the change which has taken place in the signal timings due to the action of bus optimisation and updating SCOOT with these timings. The SCOOT stage lengths are kept as they were determined by SCOOT, and the offset is taken from the actual offset reached.

Fig 2-3 shows an example of do nothing recovery after an extension. As soon as the bus has left the link, the stage changes to stage 2 and the SCOOT timings are reset to maintain the SCOOT stage lengths and to make the present second the stage change time for stage 2.

Fig 2-4 shows an example of do nothing recovery after a recall. As soon as the bus has left the link, the SCOOT timings are reset to maintain the SCOOT stage lengths and to retain the previous change time of stage 1.

Do nothing recovery has the advantage of being quick, and of causing minimal disruption to the green splits, but it can considerably disrupt the offset. It may be a good method of recovery when the offset is not important.
2.4.6.2 MS recovery

Minimum stage (MS) recovery consists of running stages to a minimum to resynchronise with SCOOT timings.

Fig 2-5 shows an example of minimum stage recovery after an extension. After stage 1 has run long because of the extension, stage 2 is run to a minimum and stage 3 is run short to get into step with the SCOOT timings.

Fig 2-6 shows an example of minimum stage recovery after a recall. After the bus has left the link, stages 2, 3 and 1 are called in quick succession until the timings are in step with SCOOT.

Minimum stage recovery has the advantage of getting into step quickly and of not disrupting offsets permanently, but has the disadvantage that shortened stages may become oversaturated.

2.4.6.3 DS recovery

Degree of saturation (DS) recovery is similar to MS recovery but instead of running stages to a minimum, stages are run short to a length constrained by the degree of saturation until they are resynchronised with the SCOOT timings. Stage lengths (not shorter than the minimum stage lengths) are calculated so that no link is saturated to more than the recovery target saturation $x_r$; this value can be configured but is typically about 90%. These stage lengths are then shortened if necessary to allow recovery to be completed within one node cycle time.
Fig 2-7 shows an example of degree of saturation recovery after an extension. After stage 1 has run long because of the extension, stages 2 and 3 are run short (but not oversaturated) to get into step with the SCOOT timings.

Fig 2-8 shows an example of degree of saturation recovery after a recall. After the bus has left the link and subject to stage 1 not becoming oversaturated, stages 2, 3 and 1 are called in succession until the timings are in step with SCOOT.

This method has the advantage of not disrupting offsets permanently, but the disadvantage of taking a long time to get into step.

### 2.4.6.4 LS recovery

Long stage (LS) recovery consists of running stages long to resynchronise with SCOOT timings. Stages are still subject to their maximum stage length.

Fig 2-9 shows an example of long stage recovery after an extension. Stage 1 runs long until the SCOOT timings catch up with the bus priority timings. Because this results in stage 1 being green for longer than one cycle, this method of recovery is not usually appropriate for use after extensions.

Fig 2-10 shows an example of long stage recovery after a
recall. Stage 1 runs long until the SCOOT timings catch up. This results in a long green for stage 1 but only to the extent that stages 2 and 3 have already been shortened during the recall. Advantages of this method of recovery are that the timings are quickly back in step with SCOOT and that the offset is not permanently disrupted. If a target saturation for recall has been specified, stages 2 and 3 will not have become oversaturated, and stage 1 merely makes use of the spare green from those two stages.

### 2.4.6.5 Summary of types of recovery

Table 1 below shows whether each type of recovery is good (✓), bad (×) or very bad(XX) for the splits and offsets obtained by the SCOOT optimisers and for speed of recovery. The annotation ‘good’ means that the SCOOT splits and offsets as optimised are retained after the recovery is complete, or that the recovery happens quickly compared to other methods of recovery, and ‘bad’ the opposite.

The defaults chosen, degree of saturation recovery after extensions and long stage recovery after recalls, maintain the SCOOT splits and offsets, but other recovery methods can be configured if required.

### 2.4.6.6 End of recovery

When the stages are back in step, bus priority is no longer active, and the normal SCOOT logic is back in control.

### 3. IMPLEMENTATION

#### 3.1 SIMULATION

McLeod et al. (1994) describe simulation tests performed at TRL using the STEP simulation provided by TRL. The STEP simulation was enhanced by TRL to model buses and bus priority. Vehicles can be categorised as buses or cars, the effect of bus stops is included, and bus detectors are modelled. The simulation was run for a range of operating conditions, with congestion levels, bus demands, bus conflicts and target saturation levels being varied and different combinations of extensions and recalls being allowed. The simulation showed savings in bus passenger delay of typically 20% to 30%, with greater levels being possible at lightly trafficked junctions. Worthwhile benefits were obtained even with bus flows as high as 100 buses per hour. In one simulated network, extensions were shown to produce delay savings of about 24% to buses with the least disruption to other traffic (no significant increase in car passenger delay); recalls produced further savings of about 8% in bus passenger delay but were more disruptive to the other traffic (about 4% increase in car passenger delay).

The simulation produced recommendations for the SCOOT system to be tested during the field trials, covering the target saturation levels and recovery logic to be used. The following recommendations were made.

### TABLE 1

<table>
<thead>
<tr>
<th>Optimisation</th>
<th>Recovery type</th>
<th>Split</th>
<th>Offset</th>
<th>Speed</th>
</tr>
</thead>
<tbody>
<tr>
<td>extension</td>
<td>DN do nothing</td>
<td>✓</td>
<td></td>
<td>✓</td>
</tr>
<tr>
<td>extension</td>
<td>MS minimum stage</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>extension</td>
<td>DS degree of sat</td>
<td>✓</td>
<td>✓</td>
<td>(default)</td>
</tr>
<tr>
<td>extension</td>
<td>LS long stage</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>recall</td>
<td>DN do nothing</td>
<td>✓</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>recall</td>
<td>MS minimum stage</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>recall</td>
<td>DS degree of sat</td>
<td>×</td>
<td>✓</td>
<td>✓</td>
</tr>
<tr>
<td>recall</td>
<td>LS long stage</td>
<td>✓</td>
<td></td>
<td>✓</td>
</tr>
</tbody>
</table>
extension target saturation $x_e = 110\%$
recall target saturation $x_r = 90\%$
recovery target saturation $x_r = 90\%$
extension recovery method = degree of saturation (DS)
recall recovery method = long stage (LS)

3.2 INSTALLATIONS

3.2.1 London
The SCOOT bus priority system in London was installed by TCSU. At the time of the trial there were about 5000 London buses fitted with transponders. Special bus detectors were located about 70 metres upstream of the junction, depending on the location of bus stops. Microprocessor-based traffic signal controllers contain bus detection equipment, and new logic to perform local green extensions for bus priority has been included. The central computer contains the new SCOOT kernel produced by TRL together with software produced by TCSU to support the bus priority database and parameter values, and software to communicate with the bus priority facilities in the controllers. Bus detection is transmitted as a single bit to the central computer. The central computer transmits stage requests to the controller in the normal way, together with an EP bit if a local extension is permitted. The controller returns an EC bit when a local extension is in operation. During the field trials, Sharp palm-top computers were installed in the controllers to record full bus information from the transponders, including time and bus identifier.

3.2.2 Southampton
The UTC system in Southampton was installed by STCL. It contains the same SCOOT kernel as in London, but surrounding software has been written to interface between SCOOT and the AVL system used to detect buses. The AVL system is the Bus-Tracker system supplied by GEC, which was installed as part of the ROMANSE project with the primary objective of supplying passenger information at bus stops. Nine junctions in Southampton have been configured to allow bus priority within SCOOT. The system is at present set up to give priority to all detected buses, but it can be easily changed so that only buses running late are given priority. No local extensions are used in Southampton so no modifications are required to the traffic signal controllers.

3.3 FIELD TRIAL
McLeod and Hounsell (1995) describe the field trial in Camden Town, London, and the evaluation of results. Camden Town is a relatively congested network, and has a mixture of one-way and two-way streets, bus lanes and mixed traffic, and variations in bus flows and congestion. Priority was provided on 19 links with bus flows varying between 5 and 70 buses per hour per link.

The target saturation values used for the trial were 110% for extensions, 95% for recalls and 95% for recovery, similar to those recommended from the simulation runs. The recovery method used was 'degree of saturation' (DS) for both extensions and recalls; the 'long stage' (LS) method recommended for recalls from the simulation runs was not considered to be suitable by TCSU engineers after observation of the long stage lengths produced.

Bus delay savings averaged 5 seconds per bus per junction (22%), and reached 10 seconds per bus per junction (70%) at lightly trafficked junctions. The maximum delay saving to buses was achieved using local extensions and central recalls. Delay to other traffic was small with local extensions at all traffic levels, but recalls at higher traffic levels caused disbenefit to other traffic averaging 5 seconds per vehicle.

Strategies tested were:

<table>
<thead>
<tr>
<th>Strategy</th>
<th>Description</th>
</tr>
</thead>
<tbody>
<tr>
<td>A</td>
<td>normal SCOOT without bus priority</td>
</tr>
<tr>
<td>B</td>
<td>central extensions</td>
</tr>
<tr>
<td>C</td>
<td>local extensions</td>
</tr>
<tr>
<td>D</td>
<td>local extensions + central recalls</td>
</tr>
<tr>
<td>E</td>
<td>central extensions + central recalls</td>
</tr>
</tbody>
</table>

When analysing the results of the trial, cases studied were:

<table>
<thead>
<tr>
<th>Case</th>
<th>Routes</th>
<th>Junctions</th>
<th>Periods</th>
<th>Ave % sat</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>route 4</td>
<td>2</td>
<td>all</td>
<td>50</td>
</tr>
<tr>
<td>2</td>
<td>all</td>
<td>10</td>
<td>am only</td>
<td>73</td>
</tr>
<tr>
<td>3</td>
<td>all</td>
<td>10</td>
<td>all</td>
<td>85</td>
</tr>
</tbody>
</table>

Table 2 on the following page shows results from the trial.

Case 1 was chosen to illustrate the effect of bus priority during lower traffic levels.

The delay savings (+) or losses (-) shown compare each strategy with strategy A, the base case. Except where indicated by †, values are statistically significant at the 95% level.

The economic valuation was based on an evaluation period of 1 year (250 working days) and used Department of Transport statistics for 1992, factored to 1995 prices and
factored for London conditions and recommended by London Transport for evaluations of this nature. More details are provided by McLeod and Hounsell (1995). This gives the resource cost of a bus as £93 hr⁻¹ and the resource cost of a car as £7.70 hr⁻¹ 1995 prices, and gives the economic values shown in Table 3 (only statistically significant results are shown).

Other trials of active bus priority have been conducted outside this project in Edgware Road, London and in Southampton.

4. CONCLUSIONS

Bus priority in SCOOT has been successfully implemented in London and Southampton and has been tested by field trials. The system accepts bus information either from selective vehicle detectors and from an automatic vehicle location system.

In Camden Town, buses were shown to benefit on average by 5 seconds per bus per link, which rose to 10 seconds in lighter traffic conditions. Maximum benefits to buses occurred at all levels of traffic when using local extensions plus recalls. Local extensions gave an average of 2 seconds per bus per link higher savings than central extensions, but at the expense of engineers’ time required to configure the controllers.

Disappointingly, other traffic generally suffered large disbenefits particularly at higher traffic levels and when recalls were allowed. But extensions on their own did not show statistically significant disbenefits to other traffic, and at lower traffic levels there was little disbenefit to other traffic from either extensions or recalls. Where other traffic suffered disbenefits an economic evaluation generally showed that these outweighed the benefits to buses.

Four methods of recovery have been tested in simulation but further work would be beneficial in this area to deter-

### TABLE 2

<table>
<thead>
<tr>
<th>Case</th>
<th>Veh type</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Bus</td>
<td>+4.8</td>
<td>+7.5</td>
<td>+7.8</td>
<td>+7.5</td>
</tr>
<tr>
<td></td>
<td>Car</td>
<td>+0.1†</td>
<td>+1.2†</td>
<td>+0.6†</td>
<td>+1.6†</td>
</tr>
<tr>
<td>2</td>
<td>Bus</td>
<td>+2.1†</td>
<td>+4.4</td>
<td>+8.2</td>
<td>+6.8</td>
</tr>
<tr>
<td></td>
<td>Car</td>
<td>-2.3†</td>
<td>-2.1†</td>
<td>-5.5</td>
<td>-5.0</td>
</tr>
<tr>
<td>3</td>
<td>Bus</td>
<td>+0.2†</td>
<td>+4.1</td>
<td>+4.8</td>
<td>+3.7</td>
</tr>
<tr>
<td></td>
<td>Car</td>
<td>-1.7†</td>
<td>-0.4†</td>
<td>-5.0</td>
<td>-5.3</td>
</tr>
</tbody>
</table>

Average delay savings (s veh⁻¹ link⁻¹)
† not significant at 95% level

### TABLE 3

<table>
<thead>
<tr>
<th>Case</th>
<th>Veh type</th>
<th>B</th>
<th>C</th>
<th>D</th>
<th>E</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>Bus</td>
<td>+16</td>
<td>+25</td>
<td>+26</td>
<td>+25</td>
</tr>
<tr>
<td></td>
<td>Car</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>-</td>
<td>-</td>
<td>-</td>
<td>-</td>
</tr>
<tr>
<td>2</td>
<td>Bus</td>
<td>-</td>
<td>+49</td>
<td>+92</td>
<td>+76</td>
</tr>
<tr>
<td></td>
<td>Car</td>
<td>-</td>
<td>-</td>
<td>-297</td>
<td>-270</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>-</td>
<td>-</td>
<td>-205</td>
<td>-194</td>
</tr>
<tr>
<td>3</td>
<td>Bus</td>
<td>-</td>
<td>+92</td>
<td>+107</td>
<td>+83</td>
</tr>
<tr>
<td></td>
<td>Car</td>
<td>-</td>
<td>-</td>
<td>-574</td>
<td>-608</td>
</tr>
<tr>
<td></td>
<td>Total</td>
<td>-</td>
<td>-</td>
<td>-467</td>
<td>-525</td>
</tr>
</tbody>
</table>

Average delay savings (£k yr⁻¹)
mine the best method to use in different situations. There is also scope for considering a wider range of techniques to benefit buses under congested conditions. It is also desirable to consider ways of reducing the disbenefit to other traffic.

Further research is required, particularly in the best use of information from AVL systems, the recovery methods, and the use of bus priority in congested conditions.

5. REFERENCES


MORE INFORMATION

The Transport Research Laboratory has published the following other reports on this area of research:

LR1014  SCOOT - a traffic responsive method of co-ordinating signals. P B Hunt, D I Robertson, R D Bretherton and R I Winton. £10.

PR41   Urban traffic control: systems review. Dr K Wood. Price code J.


TRL240 UTC strategies for congested networks. Dr K Wood. Price code E.

If you would like copies, photocopy and fill in the slip below. There is a 20% discount if you take all the reports listed above. Prices include postage and are correct at the time of publication. Please see the enclosed letter for current price code values and handling charge. Enquiries to TRL Library Services, Tel: 01344 770784, Fax: 01344 770193.

To: Publication Sales, TRL Library, PO Box 303, WOKINGHAM, Berkshire, RG45 6YX.

Please send me the following Transport Research Laboratory reports (state report Nos and quantity)

Report no ........................................ Quantity ........................................
Report no ........................................ Quantity ........................................
Report no ........................................ Quantity ........................................

Name ............................................................................................................................
Address ..............................................................................................................................
...............................................................................................................................
Postcode .........................................................................................................................
Telephone .........................................................................................................................
...............................................................................................................................
Credit card address (if different from above) .................................................................

PAYMENT:

• I enclose a cheque for £ ........................................ payable to TRL Ltd.

• Please debit my Deposit Account

• Please debit my Credit Card by £ ........................................

• Credit card no ........................................ Expiry date ........................................

Signature .........................................................................................................................

USE OUR EXPERTISE

The Transport Research Laboratory's researchers and facilities are available at competitive rates.

Our 330 scientists and engineers include many world-class experts on highways design and maintenance, transport structures, traffic systems, vehicle safety, road safety and the environment.

The Transport Research Laboratory facilities include a 3.8 km test track, a fully interactive driving simulator, an all weather facility for impact testing of vehicles, large structures test halls for static and fatigue testing, dynamic pavement test facility, dynamic and low cost impact test rigs, a pedestrian impact test facility, as well as advanced computer systems and a large specialist library with online access to worldwide information.

If you are planning a project where we may be able to help, contact the Transport Research Laboratory's Business Directorate at Crowthorne, Berkshire RG45 6AU, telephone 01344 770004, fax 01344 770356.