Green splits with priority to selected movements

RAHMI AKÇELIK

REFERENCE:

NOTE:
This paper is related to the intersection analysis methodology used in the SIDRA INTERSECTION software. Since the publication of this paper, many related aspects of the traffic model have been further developed in later versions of SIDRA INTERSECTION. Though some aspects of this paper may be outdated, this reprint is provided as a record of important aspects of the SIDRA INTERSECTION software, and in order to promote software assessment and further research.
Green splits with priority to selected movements

by Rahmi Akçelik, Chief Scientist, Australian Road Research Board

Introduction. This article describes a green split computation method that allows for priority specification for selected movements. The method is an extension of the traditional Webster's1, Miller2, and Akçelik's3 line of methods, although its detailed formulation differs from previous methods in the concepts of required green times and excess green time.

The general green split computation method described in this paper allows for:
(a) unequal practical (target) degrees of saturation;
(b) minimum and maximum green times; and
(c) green split priority for selected movements, which is particularly relevant to arterial signal co-ordination (e.g. see Moskaluk and Parsonson4), arterial call or semi-actuated signal control methods.

The method is simple in principle, and is suitable for manual calculations. However, its implementation allowing for various combinations of minimum and maximum times and priority specifications may appear to be somewhat complex. The method was developed for, and implemented in, the SIDRA 3 computer program.

Background
The green split computation method described in ARR Research Report ARR No. 1234 was developed from the original Webster method. The Webster method uses flow ratios (y values) as a basis for green split calculations for critical (or representative) movements at the intersection. This method distributes the total available green time (c) over the cycle time and L is the sum of critical movement lost times) to critical movements in proportion to their y values, and this results in equal degrees of saturation for critical movements. This method was also used in the traditional Australian methods.

Instead of flow ratios, the ARR 123 method distributes the total available green time to critical movements in proportion to their required green time ratios (u values) calculated as

\[ u = \frac{y}{x_p} \]  

(1)

where \( u \) = required green time ratio (ratio of the required green time to the cycle time)

\( y \) = flow ratio (ratio of flow to saturation flow).

\( x_p \) = practical (target) degree of saturation

This method allows the use of unequal practical degrees of saturation for different movements at the intersection (e.g. \( x_p = 0.90 \) for major movements and \( x_p = 0.95 \) for minor movements), and implies equal \( x/x_p \) ratios for critical movements. Where equal \( x_x \) values are used for all movements, the results are identical to those from Webster's y-value method. Thus, Webster's method implies an equal degree of saturation solution as a special case of the ARR 123 method. As a green split computation strategy, this is equivalent to minimising the intersection degree of saturation.

The general critical movement identification method introduced in ARR 123 (and implemented in SIDRA) as an extension of earlier methods enables the handling of complicated overlap movement cases in cycle time and green split calculations.

The ARR 123 green split computation method also introduced an allowance for minimum green time effects. The method is simply to allocate the minimum green time to each movement whose required green time based on Equation (1) is less than its minimum green time. Such movements are excluded from green split calculations, and their minimum green times added to the total lost time. This adjusted total lost time (\( L' \)) is used to calculate a new total available green time (\( c-L' \)) which is distributed to remaining critical movements to achieve equal \( x/x_p \) ratios.

In SIDRA 2, this method was extended to allow for maximum green time constraints in a similar fashion.

The green split computation method using the excess green time concept (SIDRA 3) was developed to achieve more balanced degrees of saturation with minimum and maximum green constraints and, at the same time, to allow for green split priority allocation to selected movements.

The general method allowing for green split priorities is presented following a simple formulation of the method without green split priority. A simple example is given to explain various aspects of the method.

The required green times and excess green time

Instead of using the required green time ratio given by (Equation (1)), the new method is formulated in terms of required green times calculated from:

\[ g = \frac{q_c}{s} \]

(2)

\[ g = g_m \text{ if } g > g_m \]

(3)

where \( g \) = original required green time (this is the flow-based required green time which corresponds to Equation (1) through \( u = \frac{y}{x_p} \))

\( g_m \) = minimum green time

\( g_m \) = maximum green time

\( y \) = flow ratio (ratio of flow to saturation flow, q/s)

\( c \) = cycle time

\( x_p \) = practical (target) degree of saturation

A useful explanatory feature of Equation (2) is that the required green times are seen to depend on cycle time (this is not immediately obvious with the y-value method). Firstly, assuming constant saturation flows (y values), the dependence of required green times on cycle time reflects what happens in real life; longer green times are needed to achieve the same degree of saturation if cycle time is longer. This is because longer cycle times mean longer red times, and hence, longer queues to clear. Expressing this in terms of Equation (2), a longer cycle time means that more vehicles arrive during one cycle (qSC vehicles) and a longer green time is required to meet this demand.

Furthermore, unlike the assumption in Webster's method as used in most traditional methods (e.g. the well-known TRANSYT program), saturation flows are highly dependent on signal timings (cycle time and green times) due to short lanes, opposed turns and lane blockage in shared lanes, as clearly demonstrated through the capacity modelling developed for SIDRA.

As a result, the flow ratios will also depend on signal timings. In this sense, longer cycle times may lead to reduced saturation flows and increased y values, and, in turn, result in longer required green times.

The dependence of required green times on cycle time, in fact, explains the tendency...
of vehicle-actuated traffic signals towards longer cycle times (through long maximum green times). This dependence also enforces the need for iterative calculations in capacity and timing analysis for traffic signals as adopted in SIDRA.

The first step in the green split computation method is to calculate the excess green time in the cycle as the difference between the total available green time (c-L) and the sum of adjusted required green times for critical movements:

$$\Delta g = (c-L) - \sum \Delta g$$  \(\ldots(4)\)

where $\sum \Delta g$ is the sum of adjusted required green times for critical movements.

The algorithm used in SIDRA 3 is structured according to the values of the excess green time, $\Delta g$. Some movements may be assigned their minimum or maximum green times depending on the value of the excess green time. The excess green time is then distributed to other critical movements in proportion to their original required green times and added to the original required green times. The resulting green times are checked against the minimum and maximum green times, and set to those values if the minimum or maximum constraint is violated.

Green splits without priority

The method is described below in terms of distributing the total available green time in proportion to the required green times directly, rather than distributing the excess green time as used in the general method. This helps to demonstrate the similarity of the method given here to the previous methods described in the literature.

For critical movements, calculate green times from:

$$g_i = A \frac{g_i}{\sum g_i}$$  \(\ldots(6)\)

where

$$A = \frac{(c-L)}{\sum g_i}$$  \(\ldots(6a)\)

where summation is for all critical movements.

Check green times, $g_i$, for critical movements and recalculate if necessary:

- If, for a movement, $g_i < g_{min}$, set

$$g_i = g_{min}$$  \(\ldots(7a)\)

- or if $g_i > g_{max}$, set

$$g_i = g_{max}$$  \(\ldots(7b)\)

Calculate an adjusted total lost time by adding to the lost time the green times of all movements whose required green times have been set equal to minimum or maximum values:

$$L' = L + \sum_{g_i \text{min}} + \sum_{g_i \text{max}}$$  \(\ldots(8)\)

Then recalculate green times for other movements from

$$g_i = A' \frac{g_i}{\sum g_i}$$  \(\ldots(9)\)

where

$$A' = \frac{(c-L)}{\sum g_i}$$  \(\ldots(9a)\)

The resulting degrees of saturation:

The if the excess green time from Equation (4) equals zero, $\Delta g = 0$, then all critical movements will have $x = x_p$. The excess green time is then distributed to other critical movements in proportion to their original required green times and added to the original required green times. The resulting green times are checked against the minimum and maximum green times, and set to those values if the minimum or maximum constraint is violated.

The sum of original required green times is $\Sigma g_i = 65.6$, and from Equation (6) the ratio of available green time to the sum of original

$\text{Example 1:}$ Example for green split calculation (c = 120 sec., all movements with lost time (sec), $g_{min} = 0.90$).

$\text{Movement 1:}$ A, B, C, $\Delta g = 0.00$,

$\text{Movement 2:}$ A, B, C, $\Delta g = 0.00$,

$\text{Movement 3:}$ A, B, C, $\Delta g = 0.00$.

The required green times are $A = 105/65.6 = 1.60061$. From Equation (6), the green times are calculated as $g_i = 1.60061 \times (9.6 + 15.37 \text{ sec})$, $g_i = 1.60061 \times (20.0 + 32.01 \text{ sec})$, and $g_i = 1.60061 \times (36.0 + 57.62 \text{ sec})$. All movements satisfy minimum green time constraints and hence these green times are acceptable. Note that the adjusted required green time of Movement 1 is $\Delta g_i = 0$, and a longer green time has been assigned to this movement. This results in equal degrees of saturation of $x = 0.562$ for all movements.

The integer green time constraints given in Figure 1 are from SIDRA 3 which adjusts critical movement green times in 1-second increments so as to minimise the largest $x_p$ ratio for the intersection. With equal $x_p$ values, this process is equivalent to balancing (equalising) movement degrees of saturation. The result of balancing for this example has been to round the green time for Movement 1 up (i.e. 16 sec.). The rounding of green times results in slightly unequal degrees of saturation, as seen in Figure 1.

The $\text{ARR 123 method}$: In this method, Movement 1 is assigned its adjusted required green time (minimum) value, $g_i = g_{min} = 12$ sec., and an adjusted total lost time is calculated as $L' = L + g_i = 15 + 12 = 27$ sec. If implemented in a way similar to the SIDRA method by excluding Movement 1, the $\text{ARR 123 method}$ would give $\Sigma g_i = g_i + g_i = 56.0$ sec., $c-L' = 120 - 27 = 93$ sec., $A = 93/56.0 = 1.6007$, and therefore $g_i = 1.6007 \times (20.0 + 33.21 \times 33.21) = 59.79 \times 60$ sec. The resulting degrees of saturation are $x_1 = 0.720$, $x_2 = 0.545$, and $x_3 = 0.540$.

It is seen that this method results in a higher degree of saturation to Movement 1 because it is assigned a green time equal to its minimum and the excess green time available in the cycle is split between Movements 2 and 3. The resulting delays for Movements 1 to 3 are $57.7, 37.1$ and $20.5$ sec., respectively, and the average delay to all vehicles is $30.2$ sec. Thus the results are close to the equal degree of saturation results shown in Figure 1, although this is not always the case.

Green splits with priority to selected movements:

A general method for green split computation allowing for high green split priority for selected movements, in addition to allowing for minimum and maximum green times and unequal practical degrees of saturation, is to calculate green times for critical movements from:

$$g_i = g_{min} + \Delta g_i$$  \(\ldots(10)\)

where

$$\Delta g_i = \frac{(c-L)}{\sum g_i}$$  \(\ldots(10a)\)

where

<table>
<thead>
<tr>
<th>Mov.</th>
<th>q</th>
<th>s</th>
<th>y</th>
<th>3</th>
<th>2</th>
<th>1</th>
<th>0</th>
<th>g_i</th>
<th>x_i</th>
<th>\text{d}</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>108</td>
<td>1500</td>
<td>0.722</td>
<td>9.6</td>
<td>0.90</td>
<td>12.0</td>
<td>0.90</td>
<td>15.37</td>
<td>0.562</td>
<td>16</td>
</tr>
<tr>
<td>2</td>
<td>266</td>
<td>1700</td>
<td>0.150</td>
<td>20.0</td>
<td>0.90</td>
<td>32.0</td>
<td>0.90</td>
<td>32.01</td>
<td>0.562</td>
<td>35</td>
</tr>
<tr>
<td>3</td>
<td>486</td>
<td>1800</td>
<td>0.270</td>
<td>36.0</td>
<td>0.90</td>
<td>57.62</td>
<td>0.90</td>
<td>57.62</td>
<td>0.562</td>
<td>57</td>
</tr>
<tr>
<td>Total</td>
<td>849</td>
<td>4092</td>
<td>65.6</td>
<td>68.0</td>
<td>105.00</td>
<td>105</td>
<td>30.5</td>
<td></td>
<td></td>
<td></td>
</tr>
</tbody>
</table>

July/August 1990
where \( \Delta g = \) extra green time allocated to ith movement (can be negative);
\( g_i^e = \) original required green time (as calculated from Equation (2)) for ith movement which is to be allocated extra green time; and
\( g_i = \) adjusted required green time (as calculated from Equation (3)) for ith movement which has been eliminated from the process of allocating extra green time due to a minimum or maximum green time, or a low or high priority specification. The elimination method is explained below.

Equation (10) can also be written in the form of Equations (6) and (9):

\[
\begin{align*}
\dot{g}_i &= A \dot{g}_i^e \\
A &= (c - 1 - \sum g_i) \sum g_i^e
\end{align*}
\]

where the sum \( \sum g_i^e \) is for all critical movements except those which have been eliminated (due to a minimum or maximum green time, or a low or high priority specification), and the sum \( \sum g_i \) is for all eliminated critical movements.

The Elimination Method. Case a — Without high green split priority specification:
In this case, all critical movements are treated equally according to the value of excess green time per cycle, \( \Delta g \) , calculated from Equation (4). There are two sub-cases according to whether the excess green time is positive or negative.

**Case a.1 — The excess green time is positive (\( \Delta g > 0 \)):** In this case, the movements with \( g_i = g_{i\min} \) are eliminated from the process (\( g_i = g_{i\max} \) is set) as a first step.

The green times, \( g_i \), are then calculated from Equation (10) or (11), and checked against minimum green times. If \( g_i < g_{i\min} \) is found for any movement, it is eliminated from the process (\( g_i = g_{i\min} \) is set), and calculations are repeated by reapplying Equation (10) or (11) to remaining movements.

This case means that there is excess time in the cycle, and this time is distributed to all critical movements except those which have \( g_i = g_{i\max} \). Thus, all critical movements will have \( x < x_p \) unless \( g_i = g_{i\max} \).

**Case a.2 — The excess green time is negative (\( \Delta g < 0 \)):** In this case, the movements, the movements with \( g_i = g_{i\min} \) are eliminated from the process (\( g_i = g_{i\min} \) is set) at the start.

The green times, \( g_i \), are calculated from Equation (10) or (11), and they are checked against maximum green times. If \( g_i > g_{i\max} \) is found for any movement, it is eliminated from the process (\( g_i = g_{i\min} \) is set), and Equations (10) or (11) are reapplied to remaining movements.

This case means that there is insufficient time in the cycle to achieve \( x = x_p \), and time is taken out of all critical movements except those which have \( g_i = g_{i\min} \). Thus, all critical movements will have \( x < x_p \) unless \( g_i = g_{i\min} \).
On the other hand, a comparison of individual movement delays show that the SIDRA solution ($d_1 = 48.6 \, \text{sec}$, $d_2 = 38.0 \, \text{sec}$, and $d_3 = 27.2 \, \text{sec}$) gives more equitable delays, or a lower percentage of the largest average delay to any movement, compared with the minimum delay solution. This leads to the consideration of a green split strategy which gives equal delays to all critical movements (equivalent to minimising the largest average delay to any movement). The difference between this strategy and the minimum overall delay strategy is its similarity to user-optimising and system-optimising strategies in traffic assignment.

The equal delay solution for the example in Fig 1 is $g_1 = 30 \, \text{sec}$, $g_2 = 33 \, \text{sec}$, and $g_3 = 22 \, \text{sec}$, resulting in $d_1 = 36.6 \, \text{sec}$, $d_2 = 37.1 \, \text{sec}$, and $d_3 = 37.2 \, \text{sec}$. (approximately equal within the constraint of integer green times), and an average overall delay of $d = 37.1 \, \text{sec}$.

For this example, the equal degree of saturation strategy used in SIDRA is seen to give a solution closer to the minimum overall delay strategy.

An interesting discussion of several green split calculation strategies in the context of combined traffic assignment and signal timing optimization has been presented by Van Uren, Smith and VanVleet. In addition to the equal degree of saturation strategy (referred to as the Webster Policy) and the minimum overall delay strategy, Van Uren et al. considered a strategy which aims to equalise the product of average delay and saturation flow (equal $s_d$ policy) so as to allocate longer green times to major roads. For the example in Fig 1, the equal $s_d$ solution is $g_1 = 36 \, \text{sec}$ and $g_2 = 36 \, \text{sec}$, and $g_3 = 44 \, \text{sec}$, resulting in $d_1 = 40.5 \, \text{sec}$, $d_2 = 34.6 \, \text{sec}$, and $d_3 = 33.9 \, \text{sec}$ with an average overall delay of $d = 34.9 \, \text{sec}$. In contrast with the equal degree of saturation strategy, this solution is closer to the equal delay solution than the minimum overall delay solution.

Generally, the equal $s_d$ strategy will give results which are close to the equal delay strategy when critical movement saturation flows ($s$) are close. In such cases, both the equal degree of saturation and the minimum overall delay strategies are better in terms of achieving the objective of allocating longer green times to major movements. Green splits with priority to major movements, or with unequal practical degrees of saturation (higher $s_{p}$ values for minor movements), or both, could be used to achieve this objective in a more effective and efficient way.

In the literature, the minimum-delay and other signal timing computation strategies are often discussed without considering the type of intersection control (vehicle-actuated, fixed-time, co-ordinated or unco-ordinated) used in real-life situations. Many aspects of the practical signal timing method adopted in the SIDRA program (including the equal degree of saturation strategy for green time computation presented in this paper) relate to vehicle-actuated control better (see Akçelik for techniques to determine the optimal green and calculate a practical cycle time). Unlike the simple example given in this paper, non-critical movements exist in real-life intersection cases, and this contributes to the difficulty of expressing general relationships between alternative computation strategies. However, it appears that the equal degree of saturation strategy gives a solution between the minimum overall delay and the equal delay strategies, which might be considered a good compromise between the user-optimising and system-optimising strategies.

More rigorous discussions on the relevance of various signal timing computation strategies and optimisation criteria (delay, queue length, a performance index, fuel consumption, cost, etc.) to real-life signal control systems and traffic conditions are needed.

ACKNOWLEDGMENT

The author thanks Dr Ian Johnston, the Executive Director of the Australian Road Research Board, for permission to publish this article. The views expressed in the article are those of the author, and not necessarily those of ABB.

REFERENCES


ARRB updates SIDRA

The Australian Road Research Board has released an updated version of SIDRA, its intersection modelling software package. Version 3.2 contains many new features which make it easier to use and increase the range of options and outputs: for example, users now have full control over specifications of output tables and can use new features such as unit conversions and variable flow scales. The package also includes new documentation in the form of four User Manuals, and SIDRA 3.2 also has an option to implement the U.S. Highway Capacity Manual methods.

According to ARRB’s Executive Director Dr Ian Johnston, SIDRA has been bought by over 140 organisations in 23 countries and has been used under a wide variety of operating conditions, enabling continuous development to take place in response to feedback from many practising engineers and planners. Work is proceeding on roundabout modelling and, with the support of VicRoads and the University of Melbourne, a Road Intersection Data Editing System (RIDIS), a special graphics-based input data preparation program. Both features will be included in SIDRA 3.3.

SIDRA 3.2 can be used on any IBM PC or compatible, and is released under the terms of a software agreement. In Australia and New Zealand it costs A$1,400 for commercial organisations and A$900 for researchers — contact Dr Rahmi Akçelik at the Board, P.O. Box 156, Nunawading, Victoria 3131. The University of Florida’s McTrans Center for Microcomputers in Transportation distributes the package in all other territories worldwide under a non-exclusive marketing arrangement.

In addition to References 13 and 14 given in the article above, recent ARRB documents by Dr Akçelik, SIDRA author, and Mark Besley, its programmer, include DN 1708, Installation and running instructions for the PC-Version; and DN 1709, User notes for the Highway Capacity Manual option.

Road traffic:

5 per cent growth, 1-3.90

Motor traffic was 5 per cent higher in the first quarter of 1990 than in the same quarter last year, according to provisional estimates issued by the U.K. Department of Transport in June.

Although the overall rate of growth in traffic seems to have been slowing during the last few quarters, motorway traffic was an estimated 12 per cent higher than in the first three months of 1989. There were some signs of a slowdown in the rapid rise in light van traffic. Growth by this group was lower than that of heavy goods vehicles for the first time in two years. However, the rise of the so-called articulated lorries continued to rise at the expense of other HGV groups. The mileage of 4-axled rigid vehicles (used mainly for bulk haulage) was estimated to have fallen compared with the same time last year.

On a seasonally-adjusted basis traffic showed a 6 per cent quarter-on-quarter growth. However, the higher traffic levels (and the associated increases in road tolls) have not necessarily indicated a real upturn in economic activity. The underlying trend.

Traffic in Great Britain — 1st Quarter 1990 (Statistics Bulletin 90 (6), 6), is available from the Data Sales Unit in Building 1, Victoria Road, South Ruislip, Middlesex HA4 0NZ (Tel: 081-844 3425).