Traffic control and route choice;  
a new model which designs signal timings

Mike Smith (mike@st-pauls-square.demon.co.uk)
Department of Mathematics, University of York, UK

1. Introduction

The main purpose of this paper is to encourage the consideration and development of new models which design signal timings and take account of route choices. The aim is that such controls (1) actively encourage “economical” routeing patterns and (2) form part of a stable interaction involving routeing, queueing and control; taking account of the scarcity of junction space. The obvious scarce commodity in congested networks is “junction capacity”.

The central idea is to make use of the natural dynamical system illustrated in figure 1 within a model to design signal timings.

Figure 1. A dynamical system arising when a responsive control system is utilised either in reality or within a model. Current route-flows change green-times (according to some responsive control policy or algorithm) and current green-times generate delays which change route-flows (as travellers seek quicker / cheaper routes). The loop is traversed indefinitely. Here we assume that the dynamical system is to be implemented within a model.

Section 2 below shows how a simple dynamical mathematical model and a simple thought experiment may be used to justify a special responsive signal control strategy which does take advantage of the dynamical system illustrated in figure 1; by encouraging congestion reducing route changes and also leading to a stable interaction between route swaps, queueing delay changes and green-time swaps between signal stages.

The dynamical model designs signal timings which take some account of route choice. All of the components of the dynamical system follow the proportional adjustment process outlined in Smith (1984a). A much fuller version of section 2 is in an appendix.

Section 3 gives a brief context and section 4 gives a brief conclusion.

2. Outline of a dynamical routeing / control / queueing model using $P_0$ and a proportional adjustment process

2.1 The simple network

We consider here the simple network in Fig. 2 with one node signalised.
2.2. A simple network and the $P_0$ control policy or algorithm

In this paper

$s_i$ vehicles per hour is the saturation flow at the link $i$ exit and
$b_i$ hours is to be the delay at the link $i$ exit.

Consider a junction with just two approaches or stages: link 1 and link 2. Then the $P_0$ policy selects green times which equalise the following two values:

$s_1b_1$ and $s_2b_2$.

Dynamically, if faced with two approaches where

$s_1b_1 < s_2b_2$

the $P_0$ signal control policy swaps green-time from approach 1 (or stage 1) to approach 2 (or stage 2) until

$s_1b_1 = s_2b_2$.

2.3. A simple network and the proportional switch re-routeing method.

Here we make the above rather vague adjustments more precise using a route switch method specified in Smith (1984a) as a simple and reasonable-looking day to day re-routeing process. This has been called a “proportional-switch adjustment process”, or PAP, by He et al (2010); it allows the stability or otherwise of a given traffic equilibrium to be studied. This is important since unstable equilibria are unlikely to persist. Also if an existing equilibrium might be moved by a control intervention to a “better” equilibrium then dynamical considerations are essential.

Later in the paper we use the same switching process to change stage green times and bottleneck delays. So all switches in this paper follow the same proportional adjustment process.

We start here in this section 2.3 with route switches only to make the principles clear. [Smith (1984b, c) extends Smith (1984a) aiming to be quicker computationally.]

Suppose that, in the network in figure 3,

$X_1(t)$ = the flow on route 1 on day $t$ (in vehicles per hour) and
$X_2(t)$ = the flow on route 2 on day $t$ (in vehicles per hour).

Suppose also that costs depend on flows and that

$C_i(X_i(t)) =$ the cost of travel via route 1 on day $t$ (in hours); and
\( C_2(X_2(t)) = \) the cost of travel via route 2 on day \( t \) (in hours).

Suppose finally that on a specific day \( t \), \( C_i(X_i(t)) > C_d(X_d(t)) \). How might the day to day system evolve?

A natural assumption is that the traveller flow swapping from route 1 on day \( t \) to route 2 on day \( t + 1 \) is an increasing function of both

- the flow \( X_i(t) \) on the more expensive route 1 on day \( t \) and
- the difference \( C_i(X_i(t)) - C_d(X_d(t)) \) in route costs on day \( t \).

Here \( X(t) = [X_1(t), X_2(t)] \) is the route flow vector on day \( t \) and \( C(X(t)) = [C_1(X(t)), C_2(X(t))] \) is the route-cost vector on day \( t \). The simplest natural swapping hypothesis, following on from above, is then that the traveller flow swapping from route 1 to route 2 will be proportional to the product of the two factors above. If the only swaps are from higher to lower cost routes then this “proportional to the product” assumption means that for some constant \( k > 0 \), the changes \( \Delta_1(X(t)), \Delta_2(X(t)) \) in the traveller flows on routes 1 and 2 will be given by the formulae:

\[
\Delta_1(X(t)) = -kX_1(t) [C_1(X(t)) - C_d(X(t))] \quad (2.1)
\]

and

\[
\Delta_2(X(t)) = +kX_2(t) [C_2(X(t)) - C_d(X(t))] . \quad (2.2)
\]

For each real number \( x \) we put \( x_+ = \max \{x, 0\} \). Using this notation (2.1), (2.2) become:

\[
\Delta_1(X(t)) = -kX_1(t) [C_1(X(t)) - C_d(X(t))] + kX_2(t) [C_2(X(t)) - C_d(X(t))], \quad (2.3)
\]

and

\[
\Delta_2(X(t)) = -kX_2(t) [C_1(X(t)) - C_d(X(t))] + kX_1(t) [C_2(X(t)) - C_d(X(t))]. \quad (2.4)
\]

Given these rates (2.3) and (2.4), the idealized day to day dynamical system becomes:

\[
X_1(t+1) = X_1(t) + \Delta_1(X(t)) \quad \text{and} \quad X_2(t+1) = X_2(t) + \Delta_2(X(t))
\]

or, if we put \( \Delta(X(t)) = [\Delta_1(X(t)), \Delta_2(X(t))] \),

\[
X(t+1) = X(t) + \Delta(X(t)). \quad (2.5)
\]

We also need a start point so we suppose that \( X(0) = X^0 \) for some suitable \( X^0 \). Throughout we suppose \( k \) is small.

Since only \( t \) occurs above as an argument in \( X(t) \) in the right hand side of equations (2.5), the dynamical system is autonomous. So let

\[
\Delta_{12} = [-1,1] = (-1,1)^T = [0,1] - [1,0] \quad \text{and} \quad \Delta_{21} = [1,-1] = (1,-1)^T = [1,0] - [0,1].
\]

( \( \Delta_{12} \) is the swap from route 1 to route 2 vector and \( \Delta_{21} \) is the swap from route 2 to route 1 vector) and write \( \Delta(X) \) in the form:

\[
\Delta(X) = k[X_1] [C_1(X) - C_d(X)], \Delta_{12} = X_2 [C_2(X) - C_d(X)], \Delta_{21}.
\]

This is a useful and slightly shorter form of (2.3) and (2.4); it applies naturally to a general network.

It may be shown (Smith (1984a, b)) that under suitable conditions (\( C \) monotone and smooth) and with suitable step lengths adjustments (2.5) converges to an equilibrium state.

1.4. Extending the PAP route swap dynamical system to embrace bottleneck delays

Consider again the simple two route network in figure 3, and consider two bottleneck delays explicitly; one of these bottleneck delays is at the exit of link 1 where route 1 meets node 1 and the other is at the exit of link 2 where route 2 meets node 1. Bottleneck delays affect flows and so including bottleneck delays explicitly means that these delays must be added to the cost vector \( C \) already discussed above, and the total cost (running cost plus bottleneck delay) will then be felt by the route flow vector \( X \). Also, the bottleneck delays themselves will be affected by route flows and so as to equilibrate these bottleneck delays we write down a natural dynamical system similar to the PAP dynamical system above.

*We now consider the dynamical systems to be artificial and not realistic.* So we go from iteration \( t \) to iteration \( t+1 \) rather than from day \( t \) to day \( t+1 \).

In the network in figure 3 let

\[
b_1(t) = \text{the bottleneck delay on link 1 at the start of iteration } t \text{ (in hours)} \quad \text{and}
\]

\[
b_2(t) = \text{the bottleneck delay on link 2 at the start of iteration } t \text{ (in hours)}.
\]
We wish to suggest the values of these bottleneck delays at the end of iteration $t$ or the start of iteration $t+1$.

Consider $b_1(t)$ and suppose that $b_1(t) > 0$. It seems natural to suggest that in our model:

$$
\begin{align*}
&\text{if } x_1 - s_1 > 0 \text{ then } b_1(t+1) > b_1(t); \\
&\text{if } x_1 - s_1 < 0 \text{ then } b_1(t+1) < b_1(t); \text{ and} \\
&\text{if } x_1 - s_1 = 0 \text{ then } b_1(t+1) = b_1(t).
\end{align*}
$$

(2.7)

There are many dynamical systems which follow these simple rules (2.7). In the appendix we show how the PAP dynamical system may be used here.

### 2.5. Extending the PAP route and delay change dynamical system to embrace signal green-times

Return now to the simple two route network in figure 2. There are two stages as shown in figure 2 and so there are two green-times $G_1$ and $G_2$. $G_1$ applies on link 1 (stage 1) where route 1 meets node 1 and $G_2$ applies on link 2 (stage 2) where route 2 meets node 2. These two green-times will affect flows and bottleneck delays and so these must now be included in the dynamical system we are creating. How this may be done is shown in detail in the appendix.

Here we are considering a dynamical system so just like flows and delays above it is natural to suppose $G_1$, $G_2$, $s_1b_1$ and $s_2b_2$ are known at the start of iteration $t$ and then during iteration $t$ to change $G_1$ and $G_2$ aiming to more closely fit the P0 rule at the end of iteration $t$. We again follow PAP to swap green-time away from the approach with the smaller of the two $s_1b_1$ and $s_2b_2$ values, adding the same amount to the approach with the greater $s_1b_1$ value.

Suppose that, in the network in figure 2,

- $G_1(t)$ = the green-time on route / stage 1 at the start of iteration $t$ (a proportion and so dimensionless) and
- $G_2(t)$ = the green-time on route / stage 2 at the start of iteration $t$ (a proportion and so dimensionless).

Suppose also that delays at the start of iteration $t$ are

- $b_1(t)$ = delay on link 1 (in hours); and
- $b_2(t)$ = delay on link 2 (in hours).

Suppose finally that at the start of a specific iteration $t$, $s_1b_1(t) < s_2b_2(t))$. Just as with route flows above a natural assumption is that green time swaps from route / stage 1 to route / stage 2 during iteration $t$ as follows.

$$\Delta G_1(t) = -kG_1(t) [s_2b_2(t) - s_1b_1(t)]$$

and

$$\Delta G_2(t) = kG_1(t) [s_2b_2(t) - s_1b_1(t)]$$

and so

$$G(t+1) = G(t) + [\Delta G_1(t), \Delta G_2(t)]$$

Again the description here is expanded in the appendix giving many details.

### 3. Traffic signal control and routeing: a context

Webster (1958) was one of the first to seek to model signal timings and their effect on traffic flow at a single junction (assuming that flows are essentially fixed). Robertson (1969) gives a model of a whole network (TRANSYT) allowing whole network optimisation of traffic signals (for know OD inputs and known routes). Hunt et al (1982) developed the real time control system SCOOT; essentially from the TRANSYT model. The subject is a very large one; Wood (1993) provides a review of certain urban traffic control systems.

Allsop (1974) pointed out the importance of allowing for route choices when considering the impacts of signal control changes. Gartner (1976) considers area traffic control and network equilibrium. Dickson (1981) showed that optimising signals for fixed flows does not give optimum timings when route choices are variable. Signal-controlled networks, allowing route choices to vary, are considered by Fisk (1980) and Sheffi and Powell (1983). Smith (1979a) proposed the $P_0$ traffic control policy and Smith (1987) extended the $P_0$ policy to allow for vertical queueing delays.

Van Vuren and Van Vliet (1992), Smith and van Vuren (1993), Yang and Yagar (1995) and Yang (1996) have considered in detail the interaction between signal control and routeing. Meneguzzo (1996,
1997) reports computational experiments with combined traffic assignment and control models and provides a review of models linking signal control and route choice. Hu and Mahmassami (1997) have studied (within a model) day to day evolution of network flows under real-time information and reactive signal control. Taale and van Zuylen (2001) provide interesting discussions of their own work and the work of others combining signal control and route choice.

More recently, Heydecker (2004) and Heydecker et al (2007) propose an adaptive dynamic control system for traffic signals and also considered possible future objectives for traffic signal control. Mounce (2009, 2003) has shown that a time-varying equilibrium exists with responsive control (under certain conditions, which prohibit blocking back). Smith and Mounce (2011) present a splitting rate model embracing in a simplified way both traffic re-routing and signal control adjustments.

LINSIG (2010) software generates signal timings for given flows; this software is often used in real life for junctions and small networks, and may involve small scale routeing considerations. Smith (2010) suggests a way of designing signal timings, but without explicit queues.

The need for models with more realistic explicit queues has been recently emphasised by Bliemer et al (2012), Daganzo (1998) and others. Vertical or spatial queueing assignment models have been proposed by Thompson and Payne (1975), Larsson and Patriksson (1995), Nesterov and de Palma (2003), Nie et al (2004), Smith (2012, 2013). Capacitated models which have signal timings are given by Smith (1979a, b, c, 1987), Smith et al (1987), Smith (2011) and Smith et al (2013). The last paper has capacity constraints, explicit queues which take up space and signal control.

4. Conclusion

This paper has given a stable dynamical model embracing traffic control, queueing delays and routeing; the model is deigned so that an implementation in software becomes a signal green-time design tool. Signal timings generated by the model (1) make some reasonable systematic allowance for travellers’ travel choices (including route choices) and (2) encourage congestion-reducing travel choices in the future.

A brief and selective context of academic papers concerning traffic control / route choice has also been provided.

5. References


He et al., 2010. A link-based day to day traffic assignment model. Transportation Research Part B 44 (4), 597 – 608.


Robertson, D. I., 1969. TRANSYT: a traffic network study tool. RRL Lab. report LR253, Road Research Laboratory, Crowthorne, UK.
This appendix has overlaps with section 2 in the main paper above

2 (Expanded) The proportional-switch adjustment process (PAP) as an iterative process

Here we make the above rather vague delay and green time adjustments more precise by following in detail a route switch method specified in Smith (1984a). This was then intended as a simple and reasonable-looking day to day re-routing process and has been called a “proportional-switch adjustment process”, or PAP, by He et al (2010). This routeing adjustment process allows the stability or otherwise of a given traffic equilibrium to be studied; this is important since unstable equilibria are unlikely to persist. Also if an existing equilibrium might be moved by a control intervention to a “better” equilibrium then dynamical considerations are essential.

2.1. The proportional switch adjustment process in a very simple network

Appendix

This appendix has overlaps with section 2 in the main paper above

2 (Expanded) The proportional-switch adjustment process (PAP) as an iterative process

Here we make the above rather vague delay and green time adjustments more precise by following in detail a route switch method specified in Smith (1984a). This was then intended as a simple and reasonable-looking day to day re-routing process and has been called a “proportional-switch adjustment process”, or PAP, by He et al (2010). This routeing adjustment process allows the stability or otherwise of a given traffic equilibrium to be studied; this is important since unstable equilibria are unlikely to persist. Also if an existing equilibrium might be moved by a control intervention to a “better” equilibrium then dynamical considerations are essential.

2.1. The proportional switch adjustment process in a very simple network

Figure 4. A two route network
Suppose that, in the network in figure 4,
\[ X_1(t) = \text{the flow on route 1 on day } t \text{ (in vehicles per hour) and} \]
\[ X_2(t) = \text{the flow on route 2 on day } t \text{ (in vehicles per hour).} \]
Suppose also that costs depend on flows and that
\[ C_1(X_1(t)) = \text{the cost of travel via route 1 on day } t \text{ (in hours);} \]
\[ C_2(X_2(t)) = \text{the cost of travel via route 2 on day } t \text{ (in hours).} \]
Suppose finally that on a specific day \( t \), \( C_j(X_j(t)) > C_i(X_i(t)) \). How might the day to day system evolve?

A natural assumption is that the traveller flow swapping from route 1 on day \( t \) to route 2 on day \( t + 1 \) is an increasing function of both
- the flow \( X_1(t) \) on the more expensive route 1 on day \( t \) and
- the difference \( C_1(X_1(t)) - C_2(X_2(t)) \) in route costs on day \( t \).

Here \( X(t) = [X_1(t), X_2(t)] \) is the route flow vector on day \( t \) and \( C(X(t)) = [C_1(X_1(t)), C_2(X_2(t))] \) is the route-cost vector on day \( t \). The simplest natural swapping hypothesis, following on from above, is then that the traveller flow swapping from route 1 to route 2 will be proportional to the product of the two factors above. If the only swaps are from higher to lower cost routes then this “proportional to the product” assumption means that for some constant \( k > 0 \), the changes \( \Delta_1(X(t)), \Delta_2(X(t)) \) in the traveller flows on routes 1 and 2 will be given by the formulae:

\[
\Delta_1(X(t)) = -kX_1(t)[C_1(X_1(t)) - C_2(X_2(t))] \quad \text{and} \quad \Delta_2(X(t)) = kX_2(t)[C_1(X_1(t)) - C_2(X_2(t))].
\]

For each real number \( x \) we put \( x_+ = \max[x, 0] \). Using this notation (2.1), (2.2) become:

\[
\Delta_1(X(t)) = -kX_1(t)[C_1(X_1(t)) - C_2(X_2(t))], \quad \Delta_2(X(t)) = kX_2(t)[C_1(X_1(t)) - C_2(X_2(t))].
\]

(2.3)

Given these rates (2.3) and (2.4), the idealized day to day dynamical system becomes:

\[ X_1(t + 1) = X_1(t) + \Delta_1(X(t)) \quad \text{and} \quad X_2(t + 1) = X_2(t) + \Delta_2(X(t)) \]

or, if we put \( \Delta(X(t)) = [\Delta_1(X(t)), \Delta_2(X(t))] \),

\[ X(t + 1) = X(t) + \Delta(X(t)). \]

(2.5)

We also need a start point so we suppose that \( X(0) = X^0 \) for some suitable \( X^0 \).

Since only \( t \) occurs above as an argument in \( X(t) \) in the right hand side of equations (2.5), the dynamical system is autonomous. So let

\[ \Delta_{12} = [-1, 1]^T = [-1, 1] - [1, 0] \quad \text{and} \quad \Delta_{21} = [1, -1]^T = [1, 0] - [0, 1]. \]

(\( \Delta_{12} \) is the swap from route 1 to route 2 vector and \( \Delta_{21} \) is the swap from route 2 to route 1 vector) and define \( \Delta(X) \) in this case instead by:

\[ \Delta(X) = \sum_{(i,j) \neq (i', j')} k_i[X_i[C_j(X) - C_{i'}(X)] - X_i(C_j(X) - C_{i'}(X))]. \]

(2.6)

This is a useful and slightly shorter form of (2.3) and (2.4).

### 2.2 Extending PAP, given by (2.6) to a more general network

Equation (2.6) may be extended to the case where there are several routes joining a single OD pair by putting

\[ \Delta(X) = \sum_{(i,j) \neq (i', j')} k_i[X_i[C_j(X) - C_{i'}(X)] - X_i(C_j(X) - C_{i'}(X))]. \]

(2.7)

Equation (2.7) may then be easily extended to the case where there are several routes joining several OD pairs by putting, for OD pair \( q \):

\[ \Delta_q(X) = \sum_{(i,j) \neq (i', j')} k_{iq}[X_{iq}[C_{jq}(X) - C_{iq}(X)] - X_{iq}(C_{jq}(X) - C_{iq}(X))]. \]

(2.8)

Here the routes joining OD pair \( q \) are labelled \( q_1, q_2, q_3, \ldots, q_N \) so that \( N_q \) routes join OD pair \( q \). “Adding over \( q \)” we obtain:

\[ \Delta(X) = \sum_{q=1}^{N_q} \Delta_q(X), \ldots, \Delta_q(X), \ldots, \Delta_q(X); \]

(2.9)

here the number of OD pairs is \( K \).
We may now recover the dynamical system (2.5) by putting:

\[ X(t+1) = X(t) + \Delta(X(t)) \text{ for } t = 0, 1, 2, 3, \ldots ; \]  
\[ X(0) = X^0 \]  

where \( X^0 \) is a given starting route flow vector meeting a given fixed demand and \( \Delta(X) \) is given by (2.8) and (2.9). Thus evolution (2.10) defines a proportional adjustment process or PAP in a many OD pair many route network.

2.3. Departure from equilibrium

Following Smith (1984a), let

\[ V_q(X) = \sum_{(r,s) \in \mathcal{E}} X_{qr} \{ [C_{qr}(X) - C_{sr}(X)]^2 + X_{qr} [C_{qr}(X) - C_{sr}(X)] \} \tag{2.11} \]

\[ V(X) = \sum_q V_q(X) \tag{2.12} \]

for all \( X \in D \). Then \( V \) given in (2.11) and (2.12) is a measure of departure from equilibrium and it is easy to see that

\[ V(X) = 0 \text{ if and only if } \{ \text{for all } q, r, s, X_{qr} [C_{qr}(X) - C_{sr}(X)]_i^2 = 0 \} \]

\[ \text{if and only if } \{ \text{for all } q, r, s, [C_{qr}(X) - C_{sr}(X)] > 0 \Rightarrow X_{qr} = 0 \} \]

\[ \text{if and only if } X \text{ is a Wardrop equilibrium.} \]

The set \( E \) of Wardrop equilibria (see Wardrop (1958)) may thus be specified as follows:

\[ E = \{ X \in D; V(X) = 0 \}. \tag{2.13} \]

It is natural to consider approximate equilibria; so, for any \( \varepsilon > 0 \), let

\[ E_{\varepsilon} = \{ X \in D; V(X) \leq \varepsilon \}. \tag{2.14} \]

Suppose that \( C(.) \) is monotone and directionally differentiable. In this case \( \Delta(X) \) given in (2.8) and (2.9) is a descent direction for \( V \). See Smith (1984a).

2.4. Extending the PAP route swap dynamical system to embrace bottleneck delays

Consider again the simple two route network in figure 4, and consider two bottleneck delays explicitly; one of these bottleneck delays is at the exit of link 1 where route 1 meets node 1 and the other is at the exit of link 2 where route 2 meets node 1. Bottleneck delays affect flows and so including bottleneck delays explicitly means that these delays must be added to the cost vector \( C \) already discussed above, and the total cost (running cost plus bottleneck delay) will then be felt by the route flow vector \( X \). Also, the bottleneck delays themselves will be affected by route flows and so as to equilibrate these bottleneck delays we need to write down a natural dynamical system similar to the PAP dynamical system above.

We now consider the dynamical systems to be artificial and not realistic. So we go from iteration \( t \) to iteration \( t+1 \) rather than from day \( t \) to day \( t+1 \).

In the network in figure 4 let

\[ b_1(t) = \text{ the bottleneck delay on link 1 at the start of iteration } t \text{ (in hours)} \]
\[ b_2(t) = \text{ the bottleneck delay on link 2 at the start of iteration } t \text{ (in hours)} \]

We wish to suggest the values of these bottleneck delays at the end of iteration \( t \) or the start of iteration \( t+1 \).

Consider \( b_1(t) \) and suppose that \( b_1(t) > 0 \). It seems natural to suggest that

\[ \text{if } x_1 - s_1 > 0 \text{ then } b_1(t+1) > b_1(t); \]
\[ \text{if } x_1 - s_1 < 0 \text{ then } b_1(t+1) < b_1(t); \]
\[ \text{and} \]
\[ \text{if } x_1 - s_1 = 0 \text{ then } b_1(t+1) = b_1(t). \tag{2.15} \]

There are many dynamical systems which follow these simple rules. To follow PAP above we first estimate an upper bound on \( b_1 \) and call this \( \max b_1 \); \( \max b_1 \) is to be fixed. Then for \( k > 0 \) and small we define:

\[ \Delta b_1 = k \{ x_1 - s_1 \}, (\max b_1 - b_1(t+1)}, b_1(t+1) = b_1(t) + \Delta b_1(t) \text{ for } t = 1, 2, 3, \ldots \]  
\[ \Delta b_1 = k \{ x_1 - s_1 \}, (\max b_1 - b_1(t)), b_1(t) \Delta b_1 \text{ for } t = 1, 2, 3, \ldots \]  

This (2.16), is a little like (2.7) – (2.10) and clearly obeys the simple rules (2.15) above.

Putting this another way let

\[ \Delta b_1 = k \{ x_1 - s_1 \}, (\max b_1 - b_1(t)), b_1(t) \Delta b_1 \text{ for } t = 1, 2, 3, \ldots \]
where

\[ \Delta_{01} = [1-0] \text{ and } \Delta_{10} = [0-1]. \]

(\( \Delta_{01} \)) is the swap from 0 to 1 (a scalar or 1-vector) and (\( \Delta_{10} \)) is the swap from 1 to 0 (also a scalar or 1-vector; both corresponding to link 1). This is to be done for each link \( a \), and we will now write \( \Delta_{b_{link}} \) for \( \Delta_{b_01} \) above (and so on), so that for the general link \( a \):

\[
\Delta_{b_{linka}} = k \{(x_a - s_d) + (\max b_a - b_d) \Delta_{01} + b_a \Delta_{10}\}
\]

(2.17)

where

\[ \Delta_{01} = [1-0] \text{ and } \Delta_{10} = [0-1]. \]

Using these link delay swap vectors, two for each link, define \( \Delta_{LINKS}(X, b) \) by:

\[
\Delta_{LINKS}(X, b) = \left[ \Delta_{blink1}, \Delta_{blink2}, \ldots, \Delta_{blinka}, \ldots, \Delta_{blinkm} \right]
\]

This is a vector of dimension \( m \) = the number of links in the network.

Bottleneck delays also affect flows so in (2.8) and (2.9). To take account of these we need the route-link incidence matrix \( A \):

\[
A_{aqr} = \begin{cases} 1 & \text{if link } a \text{ is part of route } qr \\ 0 & \text{otherwise} \end{cases}
\]

\( C_{qr}(X) \) must now be replaced by

\[ C_{qr}(X) + \sum_a A^{T}_{aqr} b_a = C_{qr}(X) + \left[ A^{T} b \right]_{qr}, \]

so that now

\[
\Delta_{ODS}(X, b) = \sum_{\{(r,a),s,t\}} \left\{ X_{wr} \left[ (C_{qr}(X) + [A^{T} b]_{qr}) - (C_{qs}(X) + [A^{T} b]_{qs}) \right] \cdot \Delta_{qrs} \\
+ X_{ng} \left[ (C_{qr}(X) + [A^{T} b]_{qr}) - (C_{qs}(X) + [A^{T} b]_{qs}) \right] \cdot \Delta_{qsr} \right\},
\]

(2.18)

and finally

\[
\Delta(X, b) = \left[ \Delta_{ODS}(X, b), \Delta_{STAGES}(X, b), \Delta_{LINKS}(X, b) \right].
\]

(2.19)

2.5. Extending the PAP route and delay change dynamical system to embrace signal green-times

\[
\Delta(X, G, b) = \left[ \Delta_{ODS}(X, G, b), \Delta_{STAGES}(X, G, b), \Delta_{LINKS}(X, G, b) \right].
\]

(2.21)

To specify \( \Delta_{STAGES}(X, G, b) \), which gives the direction of motion of all signal stage green-time vectors under the effects of the flows and delays, we here suppose that the queueing delay version of the P0 control policy is applied (see Smith (1987)) in a dynamical manner. Given bottleneck delays \( b_1 \) and \( b_2 \), the control policy P0 allocates green-times \( G_1, G_2 \) to links 1 and 2 so that

\[
s_1 b_1 = s_2 b_2.
\]

(2.22)

(Green-times are a proportion and so must add to 1 here. In a real life context it would here be assumed that green-times influence bottleneck delays.)

Here we are considering a dynamical system so just like flows and delays above it is natural to suppose \( G_1, G_2, s_1 b_1 \) and \( s_2 b_2 \) are known at the start of iteration \( t \) and then during iteration \( t \) to change \( G_1 \) and \( G_2 \), aiming to more closely fit the rule (2.22) at the end of iteration \( t \). Here we again follow PAP again and swap green-time away from the approach with the
smaller of the two values \( s_1b_1 \) and \( s_2b_2 \), adding the same amount to the approach with the greater \( sb \) value.

Suppose that, in the network in figure 1,

\[
G_1(t) = \text{the green-time on route 1 at the start of iteration } t \text{ (a proportion and so dimensionless)} \quad \text{and} \quad G_2(t) = \text{the green-time on route 2 at the start of iteration } t \text{ (a proportion and so dimensionless)}.
\]

Suppose also that delays at the start of iteration \( t \) are

\[
b_1(t) = \text{delay on link 1 (in hours); and} \quad b_2(t) = \text{delay on link 2 (in hours)}.
\]

Suppose finally that at the start of a specific iteration \( t \), \( s_1b_1(t) < s_2b_2(t) \). Just as with route flows above a natural assumption is that green time swaps from route 1 to route 2 on iteration \( t \) as follows.

\[
\Delta G_1(t) = -kG_1(t) \{ s_2b_2(t) - s_1b_1(t) \}
\]

and

\[
\Delta G_2(t) = kG_1(t) \{ s_2b_2(t) - s_1b_1(t) \}
\]

(2.23)

and so

\[
G(t+1) = G(t) + [\Delta G_1(t), \Delta G_2(t)]
\]

(2.25)

To allow for \( s_1b_1(t) = s_2b_2(t) \), put \( x_r = \max \{ x, 0 \} \). Then (2.23), (2.24) become:

\[
\Delta G_1(t) = -kG_1(t) [s_2b_2(t) - s_1b_1(t)] + kG_2(t) [s_1b_1(t) - s_2b_2(t)].
\]

\[
\Delta G_2(t) = -kG_2(t) [s_1b_1(t) - s_2b_2(t)] + kG_1(t) [s_2b_2(t) - s_1b_1(t)].
\]

(2.26)

The dynamical system (2.25) then becomes:

\[
G_1(t+1) = G_1(t) + \Delta G_1(t) \quad \text{and} \quad G_2(t+1) = G_2(t) + \Delta G_2(t) \quad \text{or}
\]

(2.27)

\[
G(t+1) = G(t) + \Delta G(t).
\]

Let

\[
\Delta_{12} = [-1,1] = [-1,1]^T = [0, 1] - [1, 0] \quad \text{and} \quad \Delta_{21} = [1,-1] = (1,-1)^T = [1, 0] - [0, 1].
\]

(2.28)

(2.29)

\[
\Delta_{12} = [-1,1] = (-1,1)^T = [0, 1] - [1, 0] \quad \text{and} \quad \Delta_{21} = [1,-1] = (1,-1)^T = [1, 0] - [0, 1].
\]

\[
\Delta_{12} \text{ is now the swap from stage 1 to stage 2 vector and } \Delta_{21} \text{ is now the swap from stage 2 to stage 1 vector) and define } \Delta G \text{ in this case instead by:}
\]

\[
\Delta G(t) = kG_1(t) \{ s_2b_2(t) - s_1b_1(t) \}, \Delta_{12} + kG_2(t) \{ s_1b_1(t) - s_2b_2(t) \}, \Delta_{21}
\]

(2.27)

\[
\text{Extending the dynamical green-time system to a more general network}
\]

Equation (2.27) may be extended to the case where there are several stages at a single junction by letting \( G \) be the vector of stage green-times and \( B \) be the link-stage incidence matrix defined by putting

\[
B_{ar} = s_a \text{ if link } a \text{ is part of stage } r \text{ and } = 0 \text{ otherwise.}
\]

The we put

\[
\Delta G(t) = \sum_{(r,s) \in r < s} \{ kG_{sr}(t) ((Bb)_r - (Bb)_s)_r \}, \Delta_{sr} + kG_s(t) ((Bb)_r - (Bb)_s)_s \}, \Delta_{rs}
\]

(2.28)

Equation (2.28) may be extended to the case where there are several stages at several junctions by letting \( B \) be the link-stage incidence matrix defined by putting

\[
B_{ah} = s_a \text{ if link } a \text{ is part of stage } Jr \text{ and } = 0 \text{ otherwise}
\]

and

\[
\Delta_{\text{JUNCTION}}(G, b(t)) = \sum_{(r,s) \in r < s} \{ kG_{sr}(t) ((Bb)_r - (Bb)_s)_r \}, \Delta_{hr}, + kG_s(t) ((Bb)_r - (Bb)_s)_s \}, \Delta_{hr}
\]

(2.29)

Here the dependence on \( G \) and \( b \) at iteration \( t \) is made explicit and the stages at junction \( J \) are labelled \( J1, J2, J3, \ldots, JNS(J) \), where there are \( NS(J) \) stages at junction \( J \). Then we may “add over all junctions” obtaining:

\[
\Delta_{\text{JUNCTION}}(G, b(t)) \quad \text{for } t = 0, 1, 2, 3, \ldots
\]

We may now recover a dynamical system like (2.5) by putting:

\[
G(t+1) = G(t) + \Delta_{\text{JUNCTION}}(G, b(t)) \quad \text{for } t = 0, 1, 2, 3, \ldots
\]

(2.30)
\[ [G, b](0) = [G, b]^0 \]

where \([G, b]^0\) is a given starting stage green-time vector and \(\Delta_{\text{JUNCTION}}([G, b](t))\) is given by (2.29) and (2.30). Thus evolution (2.31) defines a proportional green-time adjustment process in a many junction many stages network.

2.7 Combining the three dynamical systems

First we amend evolution of the bottleneck delays in (2.17) above to allow for green times. So now

\[
\Delta_{\text{b,linka}} = k \left( [x_a - s_{g,a}, (\text{max}b_a - b_a)]_{\Delta_{01}} + \{s_{g,a} - x_a, b_a \} \Delta_{10} \right) \]

and:

\[
\Delta_{\text{LINKS}}(X, G, b) = [\Delta_{\text{b,link1}}, \Delta_{\text{b,link2}}, \ldots, \Delta_{\text{b,linka}}, \ldots, \Delta_{\text{b,linkn}}].
\]

We then let

\[
\Delta[X, G, b] = [\Delta_{\text{OD}}(X, b), \Delta_{\text{JUNCTION}}(G, b)(t), \Delta_{\text{LINKS}}(X, G, b)]
\]

2.8 An objective function for the whole dynamical system

Given a triple \([X, G, b]\), consider OD pair \(q\), junction \(J\) and link \(a\). The three directions

\[
\Delta_{\text{OD}}(X, b), \Delta_{\text{JUNCTION}}(X, G, b) \quad \text{and} \quad \Delta_{\text{LINKS}}(X, G, b)
\]

specify the direction of motion of \(X, G\) and \(b\), and these all combine to give \(\Delta[X, G, b]\), giving the direction of motion of the whole vector \([X, G, b]\) and so giving rise to the following dynamical system:

\[
[X, G, b](t + 1) = [X, G, b](t) + \Delta[X, G, b](t) \quad \text{for} \ t = 0, 1, 2, 3, \ldots ;(2.34)
\]

\[
[X, G, b](0) = [X, G, b]^0
\]

Associated measures of departure from equilibrium

\[
V_{\text{OD}}(X, b), V_{\text{JUNCTION}}(X, G, b) \quad \text{and} \quad V_{\text{link}}(X, G, b)
\]

are given below. These measure the extent to which

\[
\Delta_{\text{OD}}(X, b), \Delta_{\text{JUNCTION}}(X, G, b), \Delta_{\text{LINKS}}(X, G, b)
\]

departs from the zero vector.

The directions (2.33) and associated measures of departure from equilibrium are:

\[
\Delta_{\text{OD}}(X, b) = \sum_{(r, s) \neq (r, s)} k \{ X_{qr} [(C_{qr}(X) + [A^T b]_{qr}) - (C_{qr}(X) + [A^T b]_{qr})] \Delta_{qs} 
\]

\[
+ X_{qs} [(C_{qs}(X) + [A^T b]_{qs}) - (C_{qs}(X) + [A^T b]_{qs})] \Delta_{qs} \}.
\]

\[
V_{\text{OD}}(X, G, b) = \sum_{(r, s) \neq (r, s)} k \{ X_{qr} [(C_{qr}(X) + [A^T b]_{qr}) - (C_{qr}(X) + [A^T b]_{qr})]^2 
\]

\[
+ X_{qs} [(C_{qs}(X) + [A^T b]_{qs}) - (C_{qs}(X) + [A^T b]_{qs})]^2 \}.
\]

\[
\Delta_{\text{JUNCTION}}(G, b) = \sum_{(r, s) \neq (r, s)} \{ k G_{jr} \ (Bb)_{jr} - (Bb)_{jr} \}_{\Delta_{jr}} + k G_{jr} \ (Bb)_{jr} - (Bb)_{jr} \}_{\Delta_{jr}} \}
\]

\[
V_{\text{JUNCTION}}(G, b) = \sum_{(r, s) \neq (r, s)} \{ k G_{jr} \ (Bb)_{jr} - (Bb)_{jr} \}_{\Delta_{jr}}^2 + k G_{jr} \ (Bb)_{jr} - (Bb)_{jr} \}_{\Delta_{jr}}^2 \}
\]

\[
\Delta_{\text{b,linka}}(X, G, b) = k \left( [(AX)_{a} - (BG)_{a}], (\text{max}b_a - b_a)]_{\Delta_{01}} + [(BG)_{a} - (AX)_{a}], b_a \Delta_{10} \right)
\]

\[
V_{\text{link}}(X, G, b) = k \left( \text{max}b_a - b_a \right) [(AX)_{a} - (BG)_{a}]^2 + b_a [(BG)_{a} - (AX)_{a}]^2
\]

Finally to obtain a measure of departure from equilibrium for the whole dynamical system (2.34) we add the different components of \(V\) given above to obtain:

\[
V(X, G, b) = \sum_q V_{\text{OD}}([X, G, b]) + \sum_J V_{\text{JUNCTION}}([X, G, b]) + \sum_a V_{\text{link}}([X, G, b])
\]

(2.38)
2.9. Monotonicity and stability

The underlying cost function is now \([C(X) + A^Tb, BG - AX, -B^Tb]\). It is easy to see that if \(C(X)\) is monotone (perhaps constant) then

\[C(X) + A^Tb, -B^Tb, BG - AX\] is a monotone function of \([X, G, b]\).

It follows that the stability results in Smith (1984a, b) may be utilised. Provided suitable step lengths are used the dynamical system (2.34) must converge to the set of equilibrium \([X, G, b]\), where \(V\) in (2.35) equals zero. If at such an equilibrium the upper bounds are not binding (or \(b < \text{max} b\)) this equilibrium will yield

- equilibrium route flows,
- equilibrium stage green-times and
- equilibrium bottleneck delays.

Even with constant small step lengths (2.34) will converge to approximate equilibrium.