Predictive–Tentative Transit Signal Priority with Self-Organizing Traffic Signal Control

SayedBahman Moghimidarzi, Peter G. Furth, and Burak Cesme

Reducing bus delay beyond what can be achieved with conventional transit signal priority requires making and responding to longer-range predictions of bus arrival time, which include dwell time at an upstream stop. At the same time, priority decisions based on such uncertain predictions should be reversible if the dwell time should be much longer than expected. Rules for applying these concepts are proposed for application in the framework of self-organizing traffic signal control developed by authors Cesme and Furth. Predicted arrival time is based on a calculation of expected remaining dwell time and is compared with the earliest time the bus phase can be expected to return to green. One possible decision is to expedite return to green so that secondary extensions (a feature of self-organizing control logic) are inhibited. The other is to hold the green; however, this decision can be reversed if updated predictions of expected remaining dwell time indicate that the bus will arrive after the maximum green extension has expired. Simulation tests on a corridor with nine signalized intersections showed a 75% reduction in bus delay, to only 5 s per intersection, with only a 3% increase in general traffic delay.

Transit signal priority (TSP) can be a powerful method for traffic management and promotion of sustainable transportation. Buses need only a few seconds of green to pass through an intersection, and by giving them green when they arrive, it should in principle be possible to reduce their delay to near zero with almost no impact on other traffic, at least when intersections are undersaturated.

However, in practice, application of TSP often leads to meager delay reduction for transit (*1*). One reason is the short notice of an approaching bus, giving the controller little time to switch to a phase serving the bus while complying with constraints such as pedestrian clearance and minimum green. The rationale behind this short notice is that dwell time at stops and delay at intersections are random, and a reliable prediction of bus arrival time at an intersection cannot usually be made until the bus has passed the closest upstream stop and signalized intersection. Where stops are on the near side of intersections, the prediction range is only a few seconds. If stops are on the far side—a configuration advantageous for signal priority (*2*)—the typical stop spacing in urban areas still limits the prediction range to 15 s or less.

S. Moghimidarzi and P. G. Furth, Department of Civil and Environmental Engineering, College of Engineering, Northeastern University, Room 400 SN, 360 Huntington Avenue, Boston, MA 02115. Current affiliation for S. Moghimidarzi: Department of Civil Engineering, University Transportation Research Center, City College of New York, Marshak Building, Suite 910, 138th Street and Convent Avenue, New York, NY 10031. B. Cesme, Kittelson & Associates, Inc., 300 M Street SE, Suite 810, Washington, DC 20003. Corresponding author: P. G. Furth, pfurth@coe.neu.edu.

Transportation Research Record: Journal of the Transportation Research Board, No. 2557, Transportation Research Board, Washington, D.C., 2016, pp. 77–85.
DOI: 10.3141/2557-08

Other researchers have explored the idea of predicting transit arrivals further in advance with explicit recognition of the uncertainty involved, an approach called predictive priority. Wadjas and Furth (*3*) proposed a long-range (2- to 3-min) prediction for light rail transit operating in a median reservation, using that information to shorten or lengthen intervening cycles, with the goal of having the train arrive in the middle of a green period. Zlatkovic et al. (*4*) and Islam et al. (*5*) also developed predictive methods for light rail transit with good performance, but the small variability in light rail transit dwell time compared with bus dwell time allowed the researchers to ignore uncertainty in the travel time prediction.

Predictive priority research on bus transit includes work by Kim and Rilett, who used regression models to predict an upper and lower bound for dwell time at nearside bus stops on the basis of bus load, headway, and schedule adherence (*6*). They also developed logic for choosing a priority action that maximizes the chance of zero signal delay. Ekeila et al. developed a prediction model for bus arrival time and a method of choosing the best priority action; however, they assumed a dwell time standard deviation of only 3 s (*7*). Lee et al. also developed a predictive model for dwell time, using headway as an independent variable (*8*). All of these researchers report improved performance but still come short of the goal of near-zero transit delay.

A second reason for the poor performance of TSP in many applications is that conventional arterial control, accomplished with coordinated–actuated control, uses fixed cycle lengths and offsets and usually involves long cycles. Conventional arterial control offers little flexibility for serving buses that tend to arrive outside the main vehicle platoon because they serve stops (*1*). Popular adaptive control methods including SCOOT, SCATS, and ACS Lite have the same weakness.

Janos and Furth responded to this need by developing control logic that is highly interruptible (*9*). Each intersection is free to cycle independently, and input is taken from upstream intersections to help promote coordination. With TSP added, the researchers found far better performance not only for transit but also for other traffic because control responded to buses stopping in a lane, blocking flow. Surtrac, which uses local controllers optimizing a scheduling problem over a rolling horizon, also allows intersections to cycle independently and responds to buses blocking a lane but has not yet been applied with TSP (*10*).

Cesme and Furth developed a flexible, adaptive control method called self-organizing control (*11, 12*). The method is based on standard actuated control logic, with two added tactics intended to promote progression: secondary extension and—in small zones of closely spaced intersections—dynamic coordination. The researchers hypothesized that self-organizing logic would be a better platform for TSP than coordinated–actuated control because it leads to shorter cycles and because it recovers well from interruptions, which in turn allow more aggressive priority tactics. In simulation tests, the researchers found that self-organizing control leads to lower delay for general traffic and, when applied with TSP, for transit as well. However, the goal of achieving near-zero bus delay was not achieved; transit delay remained in the range of 10 to 15 s per intersection.

The objective of this research was to develop predictive priority logic that could be applied on a background of self-organizing control to further reduce bus delay. The hypothesis is that with flexible, self-organizing control logic and a TSP method based on longer-range detection, near-zero delay for transit will be achievable without substantial increase in delay to general traffic.

TSP TACTICS

Active TSP changes the signal display to favor a bus or other transit vehicle in response to detection of the bus. Until now, tests of TSP with self-organizing control logic (*11*) used the tactics of green extension, in which a signal that is already green is held until a bus passes; early green, in which the green signals of conflicting phases are cut short to promote an early return to green for the bus phase; and phase rotation, in which a leading left turn is changed to a lagging left turn so that a through or left-turn phase serving a bus can begin earlier (*1*). This paper adds a fourth TSP tactic, early red, in which the signal for a bus phase is cut short in one cycle so that it may return to green earlier in the next cycle.

SELF-ORGANIZING SIGNAL CONTROL LOGIC

A system is described as self-organizing when local elements, interacting with each other, dynamically achieve a global function or behavior. Gershenson and Rosenblueth first showed the self-organizing property of local, actuated control logic (*13*). Following this principle, Cesme and Furth developed practical control logic for urban arterial road networks with two-way streets, irregular intersection spacing, irregularly distributed traffic volumes, and multiphase control that achieves coordination without coordination being imposed exogenously (*11, 12*).

Self-organizing logic is based on standard actuated control, switching from green to red when gaps are detected (gap out) or the maximum green is exceeded (max out), skipping minor phases when there is no call, and truncating green—subject to minimum green requirements—when spillback is detected on the departure leg. Self-organizing logic uses four key features of snappy actuated control: an upstream extension detector, a short critical gap, a short minimum green, and nonsimultaneous gap out (*14*). For multilane approaches, it uses multiheadway gap-out logic, which is more effective than traditional detection at distinguishing when a standing queue has been discharged on a multilane approach (*15*). Self-organizing logic relies on upstream and stop line detectors to count cars in the trap between those detectors to estimate queue length and thus determine a minimum dynamic green.

To promote progression, self-organizing logic uses a mechanism called secondary extension, in which the green is extended for an imminently arriving platoon. Platoons are detected by maintaining a profile of arrivals expected over the next 20 s or so on the basis of input from upstream detectors. Willingness to grant a secondary extension, calculated at the moment of gap out, is based on lost time per vehicle in the approaching platoon (L^*), given by

$$L^* = \min_t L_v(t)$$

and

$$L_v(t) = \frac{t - n(t) * h_{\text{sat}}}{n(t)} \tag{1}$$

where

t = time measured from the moment of gap out,
$n(t)$ = number of vehicles expected to pass the stop line if the green phase is further extended by t, and
h_{sat} = saturation headway.

L^* decreases as the arriving platoon becomes larger, dense, and more imminent. A secondary extension is granted if L^* is less than affordable lost time, which is a function of an intersection's excess capacity and is calculated through the empirical formula

$$L_{\text{affordable}} = \min\left[2, 2\left(\frac{1}{\text{v/c}} - 1\right)\right] \tag{2}$$

where the intersection's volume-to-capacity (v/c) ratio, is calculated as follows:

$$\frac{\text{v}}{\text{c}} = \frac{\sum_{\text{critical}} \frac{v_i}{s_i}}{1 - \frac{L_{\text{sum}}}{C}} \tag{3}$$

where

v_i = arrival rate as measured (using detectors) over the past five cycles;
s_i = saturation flow rate (given), with the sum over critical movements only;
L_{sum} = sum of lost time for critical movements; and
C = maximum desirable cycle length, which equaled 90 s in this application.

Intersections that are so closely spaced that their queues are likely to interact (typically up to 500 to 600 ft) are put into groups called coupled zones. Within each coupled zone, signals follow dynamic coordination logic in which they follow the critical intersection within their zone, preventing both spillback and starvation at the critical intersection. That way the zone as a whole cycles as needed by the traffic demands at the critical intersection, with good progression between zones. Self-organizing logic also includes special features for preserving capacity during times of oversaturation, thus promoting throughput (*12*).

Published tests of self-organizing logic have shown that it performs well compared with coordinated–actuated control, even on an arterial on which the geometry is well suited to coordinated–actuated control (*11, 12*). Tests in which TSP was applied found a strong reduction in bus delay with nearly no impact to general traffic. Buses are detected at a check-in detector about 15 s upstream of an intersection. If the bus phase is green, green extension is applied; otherwise, early green is applied. If the predicted green start with early green is later than the predicted bus arrival time and the bus phase is normally lagging, phase rotation is applied to change the bus phase to leading. The actuated logic embedded into self-organizing control naturally compensates traffic streams for which green was shortened by a priority action, allowing them a longer green in the next cycle so that queues caused by priority interruptions quickly dissipate.

PREDICTIVE–TENTATIVE PRIORITY LOGIC

Even with self-organizing control, however, traditional TSP tactics based on short notice still leave the average bus delay well above the goal of near zero. Along with flexible control logic and aggressive priority tactics, better priority for transit also needs a longer range prediction of bus arrival time. The authors propose extending the prediction range to include dwell time at the bus stop upstream of the intersection. At the same time, because predictions that involve dwell time contain considerable uncertainty, actions taken at this level should be tentative. In particular, if a decision is made to hold the green for a bus, that decision should be reversible if the dwell time becomes so long that it becomes likely that the green extension will max out before the bus arrives. The authors call this proposal predictive–tentative priority logic.

Expected remaining dwell time is calculated from historical dwell time data gathered by detectors. From a cumulative distribution of dwell time, and knowing how long the bus has already been at the stop, one can readily calculate the expected remaining dwell time, given by

$$\text{E}[\text{Remaining\_DT}] = \left(\sum_{t_c}^{t_{\max}} \frac{p(t)\,t}{(1 - F(t_c))} \right) \tag{4}$$

where

t_c = current length of dwell time,
$p(t)$ = P(dwell time = t), and
$F(t)$ = P(dwell time ≤ t).

The distribution of t is discretized by seconds, and expected remaining dwell time is updated with each passing second. (If bus load and headway data are available, a more sophisticated dwell time model could be developed to improve dwell time prediction.)

The proposed logic for predictive–tentative priority, meant to be run every second or similarly small time step, is shown in Figure 1. It is activated only when a bus has been detected, the bus phase has a green signal, and the bus phase has gapped out. No action is taken if the bus phase has a red signal. Because it deals with predictions of bus arrival time that include considerable uncertainty, predictive–tentative priority logic does not truncate the green of any phase that has not yet gapped out to preserve the actuated control properties of compensation and self-healing. Once the bus departs from the stop being monitored, normal TSP logic takes over.

With self-organizing control, there are two common ways in which predictive–tentative priority logic can first become applicable:

- The bus phase is green even though it has gapped out (because it is running a secondary extension) when a bus arrives at the upstream stop. Predictive–tentative logic will then decide whether to truncate the secondary extension and expedite return to green in the next cycle or continue to extend the green for the bus while incidentally continuing the secondary extension.
- While a bus is at the upstream stop, its green phase gaps out. At this moment, predictive–tentative logic will decide whether to end the green and expedite return to the next green—inhibiting secondary extension—or to hold the green for the approaching bus.

The default action is to expedite return to green in the next cycle, and so the first condition tests whether it might be possible for the bus to arrive on green in the next cycle:

$$\text{E}[\text{Remaining\_DT}] + \text{TT} > \text{Earliest\_expected\_return} \tag{5}$$

where TT is the travel time from the bus's current position to the stop, not counting dwell time, and Earliest_expected_return is the smallest duration of time needed for the signal to cycle through intervening phases and begin the bus phase's next green. Its calculation is based on estimating the needed split of each conflicting phase, and, using those estimated splits, finding the longest path through the ring diagram to the next bus phase green.

For calculating estimated splits, first a dynamic minimum green is calculated from the number of cars in the trap between the extension detector and the stop line detector:

$$\text{dynamic minimum green} = \text{startup lost time} + \frac{(\#\ \text{queued cars})}{s} \tag{6}$$

where s = saturation flow rate. Estimated split is 0 if a phase has no call and can be skipped, as may be the case for some left-turn phases; otherwise it is

$$\text{Estimated\_split} = \max \begin{bmatrix} (\text{dynamic minimum green}) * \left(\dfrac{1}{1 - \dfrac{v}{s}} \right), \\ \text{PedMinGreen} \end{bmatrix} + \text{change interval} \tag{7}$$

where v is the phase's approach volume and PedMinGreen is the minimum green needed to provide a pedestrian interval if the pedestrian phase has a call, and 0 otherwise. The factor $1/(1 - v/s)$ accounts for vehicles expected to join the queue while the signal is green.

If Equation 5 returns true, the green will be allowed to end, and the condition Expedited_Return becomes true, which inhibits secondary extensions on the bus phase in the current cycle as well as on all conflicting phases. (Allowing the green to end means that green will end immediately unless waiting for a parallel phase—programmed to cross a barrier with it in the ring diagram—to gap out.) Expedited return does not, in itself, limit intervening phases to their minimum green. However, once the bus phase has gone to red, the signal is subject to other priority logic, including early green.

If Equation 5 returns false, three other conditions are checked. If any of them is true, green return will be expedited. Those conditions follow:

- PreviousMaxedOut. This condition is true if either of the through phases of the cross street in the intersection maxed out in the last cycle. (Turning phases are accounted for as part of a different condition.) This criterion prioritizes intersection throughput when there is an indication of a capacity shortfall because once

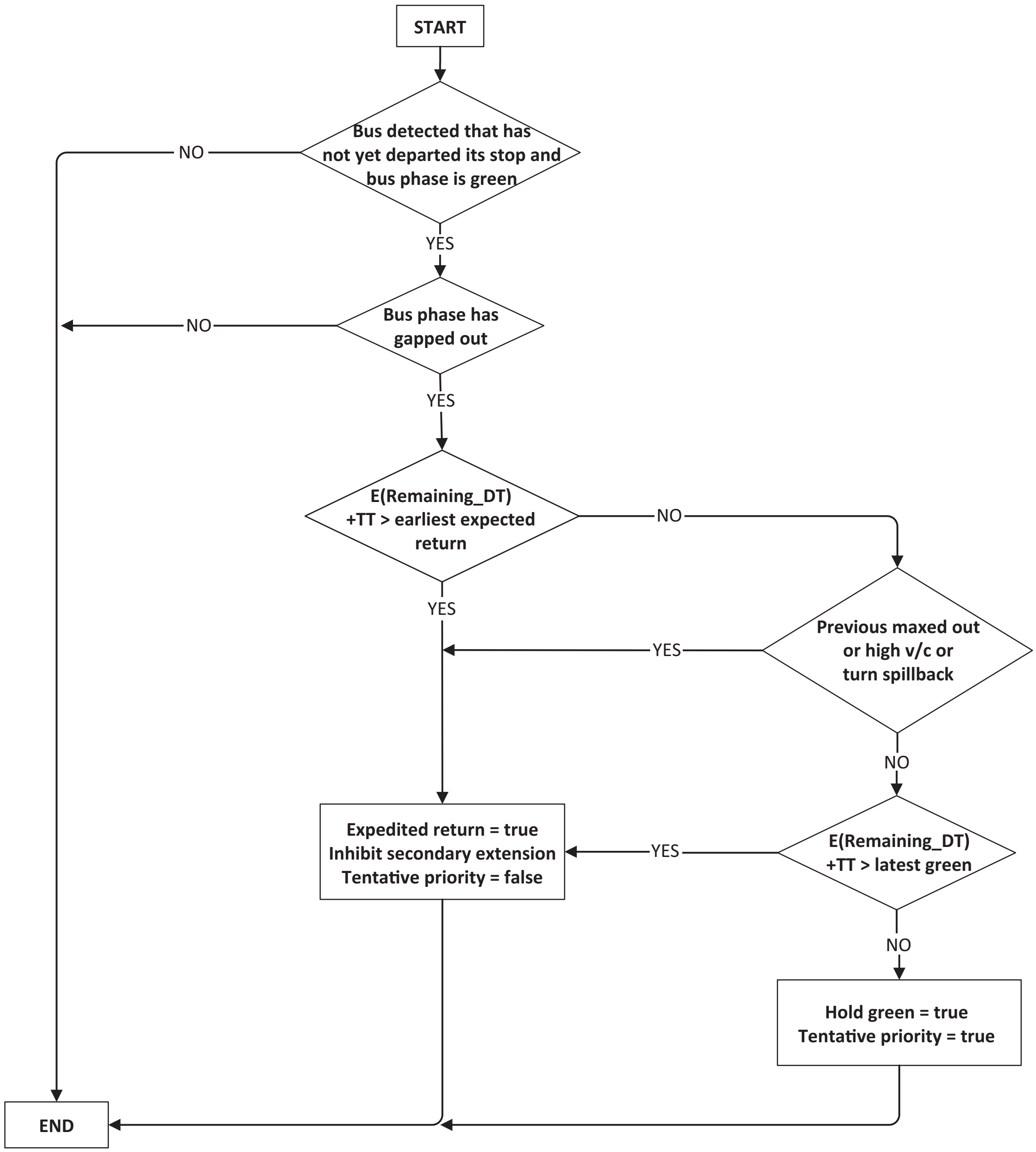

FIGURE 1 Flowchart of predictive–tentative priority logic.

large queues start to form as a result of overcapacity, TSP becomes impotent.

- High v/c. This condition is true if the intersection v/c (Equation 3) exceeds a user-specified threshold (0.9 was used in this test). This condition is optional because its function can also be served by the previous condition.
- TurnSpillback. This condition is true if a detector at the rear of a turn pocket is occupied for more than 5 s. Spillback from a turn pocket creates a safety hazard and a precipitous drop in capacity. If this condition is true, green extension will be inhibited to serve the turn phase as soon as possible.

If none of the previous conditions is true, a fourth test is applied:

$$E[\text{Remaining\_DT}] + TT > \text{LatestGreen} \tag{8}$$

If true, green return is expedited. The term on the left is the time until the bus's predicted arrival at the stop line, and LatestGreen is the time until the current green phase reaches a length equal to the phase's maximum green plus a preset maximum green extension (15 s in this test). Thus, Equation 8 forecasts whether a bus granted a green extension will be so late that the green extension will have maxed out. If so, no extension is given. If Equation 8 is false and the other conditions leading to it are satisfied, green will be held.

As the flowchart indicates, if it can be arranged for the bus phase to be green when the bus arrives either with green extension or expedited return, the latter is chosen. This choice is consistent with the self-organizing control principle of keeping cycles short. As long as the green is held, predictive–tentative logic—which includes updating expected remaining dwell time—is reapplied every time step until the bus departs from its stop or its predicted arrival time becomes too late for the bus to be served by green extension, in which case the green hold action is reversed in favor of expediting return to green.

Figure 2 shows how predictive–tentative logic can lead to three possible outcomes: an initial decision to expedite return to green, a decision to hold green that is not reversed, and a decision to hold green that is later reversed. Consider three buses—A, B, and C—which are serving a stop when their phase at the downstream intersection gaps out. In Case A, the bus is detected soon after stopping. Its expected remaining dwell time plus the travel time to the stop line will make it arrive later than the earliest expected return to green; therefore, expedited return is chosen, including a prohibition on secondary extension, so that phase can become green again in the next cycle by the time the bus arrives.

In Case B, at the moment of gap out, the expected remaining dwell time plus the travel time is predicted to get the bus to the stop line before the earliest expected return to green. Therefore, assuming the other four conditions are satisfied, green will be extended. If the dwell time matches what was expected, the bus will pass during the green extension.

Case C begins the same as Case B, and so initially gets a green extension. However, dwell time continues longer than expected. At time t_c the bus is still dwelling at the upstream stop, and the bus's predicted arrival time becomes later than the limit on green extension, and so green extension is canceled and expedited return is applied.

Case C involves some wasted green time: the signal was held for a bus that could not use that extension, imposing some delay on vehicles waiting on conflicting phases without directly reducing bus delay. That delay is the price of giving priority on the basis of the prediction of an uncertain arrival time, which is a reason to inhibit predictive priority when there is little slack capacity. Moreover, that price is minimized by continuing to check whether conditions for green extension are satisfied, and canceling them as soon as they are not, instead of just allowing the green to extend to its maximum time.

A modified rule for predictive–tentative priority is also proposed for the case of a bus stop that lies in the interior of a coupled zone. Normally, self-organizing logic aims to provide good two-way progression in a coupled zone by providing near-simultaneous green for the through arterial phases. If a bus enters the zone during the early part of the green period, secondary extensions should be welcomed (from the perspective of bus priority) because there is a good chance that, with the green being extended by the secondary extension, the bus will be able to pass through the zone within the same cycle. Therefore, Equation 5 is assumed to return false for any bus stopped in a coupled zone that passed through its last intersection before its phase gapped out.

CASE STUDY: RURAL ROAD, TEMPE, ARIZONA

The test site is a 3.1-mi section of Rural Road in Tempe, Arizona, between Warner Road and Minton Drive, depicted in Figure 3 and previously analyzed by Cesme and Furth (*11*). It has two to three lanes per direction as well as a median left-turn lane and a speed

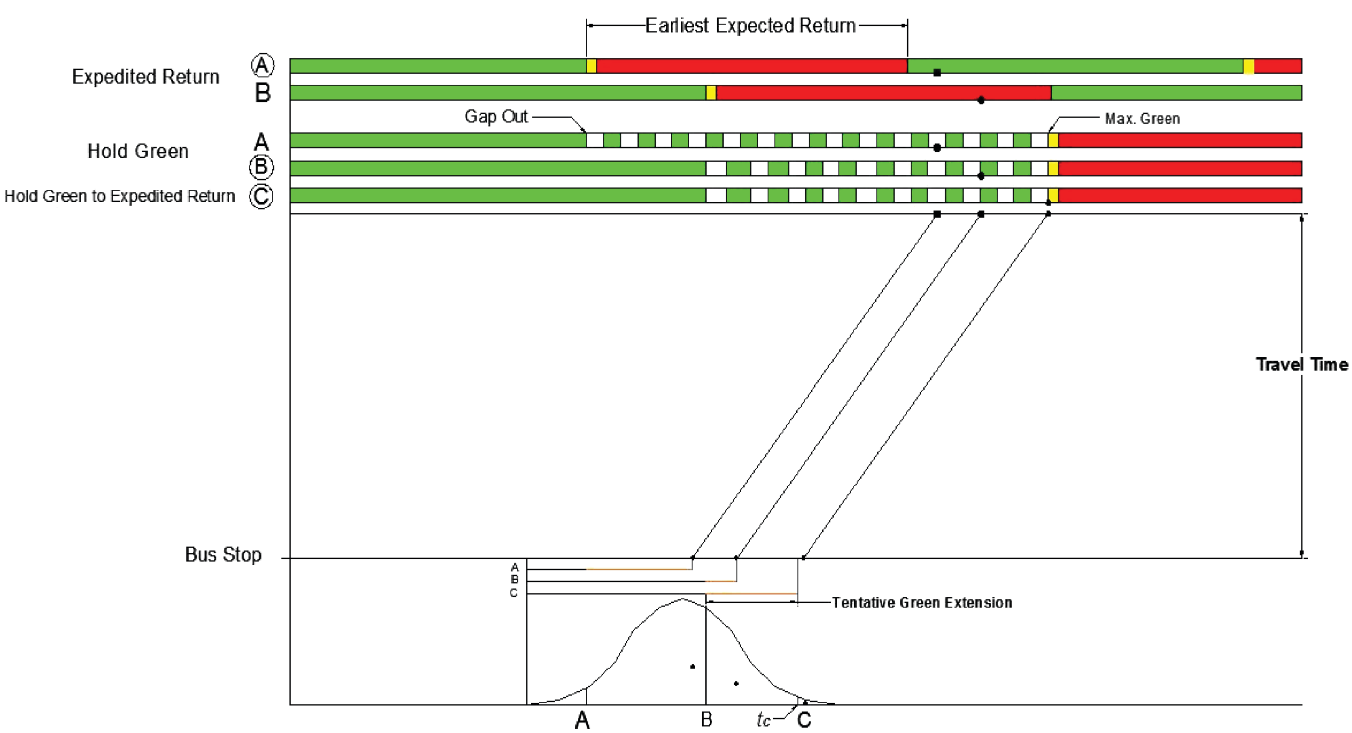

FIGURE 2 Predictive–tentative priority scheme.

FIGURE 3 Rural Road corridor, Tempe (NB = northbound; SB = southbound).

limit of 45 mph. There are nine signalized intersections, of which Intersections 8 and 9—600 ft apart—are treated as a coupled zone in self-organizing logic. Traffic counts and signal timing information were obtained from the Maricopa Association of Governments. Left-turn phases are always leading while the opposing through phase lags. Although the existing bus headway is 20 min, it was set to 10 min to test performance with frequent transit priority interruptions.

Analysis used Vissim, a microsimulation model. For self-organizing control, control logic was programmed in C++. Control logic interacted with the simulation through Vissim's application programming interface, in which, at every time step, detector information is passed from simulation to controller and information about signal states is returned to the simulation program. Actuated–coordinated logic was modeled with Vissim's ring barrier controller.

The corridor has 29 bus stops, of which 18 (one per signalized intersection in each direction) are part of predictive priority logic because they are the last stop before a signalized intersection. As shown in Table 1, most stops are midblock or farside stops; however, because of the short block length between Intersections 8 and 9, the stop in that block functions as a farside stop for one intersection and a nearside stop for the other. Dwell time was modeled as having a normal distribution with mean of 20 s and standard deviation of 10 s.

To test performance in a transit-oriented environment with high pedestrian demand, pedestrian phases were set to recall. Because the arterial has seven to eight lanes, pedestrian minimum green—which ranged from 25 to 33 s—controls for most cross street phases.

Call detectors were located at the stop line, and extension detectors were located 2 s (140 ft) before the stop line. Detectors for requesting secondary extension were located about 20 s before the stop line, except within the coupled zone (Intersections 8 and 9), where they were 11 s upstream. For standard TSP, bus check-in detectors were placed about 15 s before the stop line or just after the previous bus stop, whichever was closer. This placement implied that the maximum green extension can be 15 s. Bus detectors were also placed at the 18 bus stops upstream of a signalized intersection, at which dwell time measurement began when a bus had occupied the detector for more than 3 s and had zero speed.

The a.m. peak hour was simulated with constant input volumes. The simulation was run for 6 h (after a 15-min warm-up) to get more statistically meaningful results, with 36 buses passing in each direction. Volumes are such that the corridor is not oversaturated, so that successive time intervals (for example, of 15 min) are effectively independent. Vissim's calculated vehicle delay excludes dwell time, but it includes delay during deceleration and acceleration, which can be caused by traffic control but also by bus stops and car following. To isolate control delay, a special simulation was run without signals or priority rules. In later analyses, this control–independent acceleration and deceleration delay was subtracted from calculated delay to yield net delay, which is what this paper reports.

Five control alternatives were tested. The first two alternatives are actuated–coordinated logic without and with TSP, for which TSP

TABLE 1 Bus Delay for Five Control Alternatives

Segment	Bus Stop's Location from Stop Line (ft) (mb, fs, ns)	Acceleration–Deceleration Delay (s)	Average Net Delay (s): Actuated–Coordinated, No TSP	Actuated–Coordinated, TSP	Self-Organizing, No TSP	Self-Organizing, TSP	Self-Organizing, PT–TSP
Bus-NB-Int_1	1,360 (mb)	2.8	34.0	27.8	31.1	11.5	6.9
Bus-NB-Int_2	1,150 (mb)	5.5	15.1	6.3	9.2	3.9	5.3
Bus-NB-Int_3	1,400 (mb)	3.3	26.0	17.3	24.3	8.9	4.4
Bus-NB-Int_4	1,750 (mb)	3.7	15.6	1.3	9.2	3.3	5.3
Bus-NB-Int_5	1,150 (mb)	5.5	35.1	12.1	20.2	10.8	5.2
Bus-NB-Int_6	1,130 (fs)	5	5.9	3.8	11.8	6.8	3.5
Bus-NB-Int_7	1,000 (mb)	6.2	11.2	2.9	10.3	6.0	4.9
Bus-NB-Int_8	1,190 (mb)	3.9	21.2	21.7	23.7	10.5	5.6
Bus-NB-Int_9[a]	380 (ns)	3.9	40.3	16.2	31.1	28.4	10.9
Bus-SB-Int_9	1,300 (mb)	4.1	0.8	0.8	11.9	4.7	5.5
Bus-SB-Int_8[a]	415 (ns)	4.9	47.8	18.4	39.5	15.0	3.8
Bus-SB-Int_7	1,110 (mb)	5.4	4.8	2.3	8.2	2.1	1.5
Bus-SB-Int_6	1,020 (fs)	4.4	3.2	4.6	10.1	2.9	0.2
Bus-SB-Int_5	1,120 (fs)	3.3	22.0	10.6	27.8	11.3	6.9
Bus-SB-Int_4	1,130 (mb)	5.1	0.4	1.1	4.5	2.1	3.5
Bus-SB-Int_3	1,780 (mb)	2.6	32.2	18.7	24.3	13.2	5.6
Bus-SB-Int_2	1,350 (mb)	4.1	12.8	6.5	9.4	4.9	3.5
Bus-SB-Int_1	1,450 (mb)	0.2	28.8	12.6	31.3	14.8	5.4
Bus, junctions with 2 critical phases		4.9	8.6	3.6	9.1	4.0	3.5
Bus, junctions with 3 or 4 critical phases		3.5	28.8	15.6	26.5	12.9	6.0
Bus, all junctions		4.1	19.9	10.3	18.8	9.0	4.9
Bus delay reduction versus no TSP				48%		52%	74%

NOTE: mb = midblock stop, fs = farside stop, ns = nearside stop; PT = predictive–tentative.
[a] Under self-organizing control, upstream bus stop is within the coupled zone.

includes green extension and early green. Because most cross streets are constrained by a pedestrian minimum green, early green has little effect. The other three control alternatives use self-organizing control logic without TSP; with standard TSP (green extension, early green, and phase rotation); and with predictive–tentative TSP logic as well as standard TSP.

Table 1 shows net control delay for buses for the five control alternatives. For both coordinated–actuated and self-organizing control, TSP reduces bus delay by about 50%, but still leaves an average delay of 9 or 10 s per intersection. Predictive–tentative priority reduces average bus delay to only 5 s per junction. Delay is smaller (3 s on average) at intersections with only two critical phases; these intersections tend to have lower cycle lengths and shorter red intervals than those with three or four critical phases.

The greatest delay reduction is for northbound buses at Intersection 9 and southbound buses at Intersection 8, for which the upstream stop lies between two coupled intersections. With the original self-organizing logic, the coupled intersections were constrained to have arterial green waves, and buses stopping in the middle of the zone were often unable to progress through the second signal. The improved logic presented in this paper lowers bus delay from 31 to 11 s in one direction and from 39 to 4 s in the other.

Table 2 shows the impact on general traffic by junction for all traffic and for full-corridor traffic (vehicles that travel the full length of the corridor without turning). Without TSP, self-organizing logic offers substantially less overall delay (21 s versus 27 s per vehicle per intersection) than coordinated–actuated logic. This difference is similar to that found in the previous analysis by Cesme and Furth of the same corridor modeled without pedestrian recall. For arterial traffic traveling the full length of the corridor, without TSP, coordinated–actuated control gives slightly less delay than self-organizing control; however, coordinated–actuated control imposes so much more delay on cross street and turning traffic that its average delay per vehicle, considering all vehicles, is 34% greater than self-organizing control.

When standard TSP is added, neither coordinated–actuated control nor self-organizing control shows any substantial change in overall average vehicle delay, even though both reduce bus delay by about 50%. With coordinated–actuated control, TSP additionally lowers delay to full-corridor traffic at the expense of turning and cross street traffic. With self-organizing control, TSP leaves both through arterial traffic and cross traffic essentially unaffected. Predictive–tentative priority logic increases general traffic delay by 3%.

The strong reduction in bus delay produced by predictive–tentative priority coupled with a barely detectable change in general

TABLE 2 Average Intersection Delay and Cycle Length with Actuated–Coordinated, Self-Organizing, and Predictive–Tentative TSP Logic

Average Intersection Delay and Average Cycle Length (s)		Actuated–Coordinated				Self-Organizing					
		No TSP		With TSP		No TSP		With TSP		PT–TSP	
Node	Number of Critical Phases	Delay	Cycle Length	Delay	Cycle Length	Delay	Cycle Length	Delay	Cycle Length	Delay	Cycle Length
Intersection 1	4	33.4	120.0	34.1	120.0	25.0	76.5	25.3	77.9	25.8	77.5
Intersection 2	2	12.0	60.0	12.9	60.0	14.6	82.8	14.4	83.0	14.6	83.4
Intersection 3	4	31.6	120.0	33.4	120.0	25.2	79.9	25.4	81.4	25.5	81.1
Intersection 4	2	20.8	120.0	21.0	120.0	15.0	77.9	15.6	79.8	15.3	79.7
Intersection 5	4	40.6	120.0	40.1	120.0	32.2	91.8	32.9	93.1	32.8	92.9
Intersection 6	2	24.6	120.0	24.2	120.0	15.9	61.9	17.7	63.9	16.4	61.1
Intersection 7	2	25.2	120.0	22.7	120.0	14.1	66.5	14.5	67.9	13.8	66.9
Intersection 8	4	29.0	120.0	27.3	120.0	21.5	89.0	22.0	90.1	22.5	91.6
Intersection 9	3	28.7	120.0	29.1	120.0	20.7	89.0	21.2	90.1	22.1	89.6
Overall average		27.4	113.3	27.2	113.3	20.5	79.5	21.0	80.8	21.0	80.4
NB full-corridor traffic		142.1		104.1		150.3		147.4		153.6	
SB full-corridor traffic		129.6		118.4		141.6		150.4		143.0	

traffic delay demonstrates the efficacy of both predictive–tentative priority logic and self-organizing control logic, supporting the hypothesis stated by Cesme and Furth that self-organizing control logic, by virtue of its flexibility and capacity for self-healing, can enable aggressive, effective TSP without substantial negative impact to other traffic (*11*).

Table 2 also shows that self-organizing control leads to far shorter cycle lengths than coordinated–actuated control (about 80 s versus 120 s), implying substantially less delay for pedestrians in addition to the delay reductions to motorists and bus users already mentioned. Cycle length at any given intersection shows a coefficient of variation in a range of 12% to 25%. Table 2 also shows that with self-organizing control, application of TSP, with or without predictive–tentative priority logic, does not appreciably change average cycle length, thanks to its demand-responsive logic. Although TSP logic often extends the green signal for buses, the extra time given to the arterial phases in one cycle is balanced by a shorter arterial phase in the subsequent cycle because vehicles that would have been served in the subsequent cycle were served during the green extension.

To provide more insight into how predictive–tentative priority works, Table 3 shows the frequency with which a bus detection triggers the three priority actions: expedited return, green hold with the decision not being reversed, and green hold that is later changed to expedited return. The table also shows how long green was extended, on average, for the last two cases. One can see that predictive–tentative priority logic applies to about 40% of the buses, those that arrive at an upstream stop during a secondary extension or are at an upstream stop when the bus phase gaps out. Expedited return, with its inhibition of secondary green extension for platoons, was applied directly in about one-third of those cases. In the other cases, green extension was initially applied, and of those, the action was reversed in 24 of 168 cases when dwell time became too long. On average, when a green-hold action was reversed, the green was held for 25 s before the action was reversed. Reversal was most frequent at Intersection 9, which is part of a coupled zone.

TABLE 3 Frequency of Priority Actions and Mean Green Extension Lengths

		Expedited Return	Hold Green		Hold Green to Expedited Return	
Intersection	Buses Detected	Frequency	Frequency	Mean (s)	Frequency	Mean (s)
1	72	9 (13%)	20 (28%)	8.9	3 (4%)	21.6
2	72	14 (19%)	7 (10%)	10.5	3 (4%)	15.3
3	72	9 (13%)	22 (31%)	8.9	4 (6%)	15.6
4	72	12 (17%)	3 (4%)	14	1 (1%)	26.9
5	72	7 (10%)	20 (28%)	13.9	0 (0%)	na
6	72	12 (17%)	22 (31%)	10.6	0 (0%)	na
7	72	13 (18%)	11 (15%)	11.3	1 (1%)	28.1
8	72	5 (7%)	21 (29%)	17.4	2 (3%)	17
9	72	4 (6%)	18 (25%)	9.8	10 (14%)	34.6
Sum	648	85 (13%)	144 (22%)	11.6	24 (4%)	25.3

NOTE: na = not applicable.

TABLE 4 Average Bus Net Delay with Different Criteria for Predicted Dwell Time

Average Net Delay (s)	Using *p*th Percentile Remaining Dwell, for *p* Equal to							Remaining DT Using Expected Value
	20%	30%	40%	50%	60%	70%	80%	
Intersections with 2 critical phases	4.0	4.4	4.4	4.4	3.8	4.0	4.1	3.5
Intersections with 3 or 4 critical phases	6.8	6.8	7.1	6.9	7.4	6.9	6.5	6.0
All intersections	5.5	5.7	5.9	5.8	5.8	5.6	5.5	4.9

SENSITIVITY ANALYSIS

The predictive–tentative priority logic presented until now uses a predicted bus arrival time that is based on expected remaining dwell time. To test whether better performance can be obtained with an upward or downward biased estimate of remaining dwell time, a sensitivity study was conducted, in which arrival time was based on the *p*th percentile remaining dwell time, with *p* varying from 20 to 80. Computationally, a conditional cumulative distribution function can be calculated from a historical distribution of dwell time for any given amount of dwell time, from which any percentile of remaining dwell time can be determined.

Results are presented in Table 4. Average bus delay remains flat as *p* changes and is clearly worse than when arrival time is based on expected remaining dwell time.

CONCLUSION AND FUTURE RESEARCH

Predictive–tentative priority adds a longer-range prediction element to TSP that—when applied on top of self-organizing control—reduces bus delay to near zero with almost no impact on general traffic delay. For the test case, average bus delay was reduced 75% compared with a no-priority case to just under 5 s per intersection. This result confirms the hypothesis that the flexibility of self-organizing control—together with intelligent and aggressive logic for TSP—is able to give excellent service to transit while also giving better service to general traffic than is offered by the dominant paradigm of coordinated–actuated control with a fixed cycle length.

Future research should test these algorithms in a variety of travel corridors and under a variety of demand conditions. Possible directions for improved algorithmic development include developing logic for still longer-range predictions of bus arrival time, similar to work by Wadjas and Furth (*3*), and using more aggressive priority tactics to map the trade-off between bus delay and general traffic delay.

REFERENCES

1. Furth, P. G., B. Cesme, and T. Rima. Signal Priority near Major Bus Terminal: Case Study of Ruggles Station, Boston, Mass. In *Transportation Research Record: Journal of the Transportation Research Board, No. 2192,* Transportation Research Board of the National Academies, Washington, D.C., 2010, pp. 89–96.
2. Ngan, V., T. Sayed, and A. Abdelfatah. Impacts of Various Parameters on Transit Signal Priority Effectiveness. *Journal of Public Transportation,* Vol. 7, No. 3, 2004, pp. 71–93.
3. Wadjas, Y., and P. G. Furth. Transit Signal Priority Along Arterials Using Advanced Detection. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1856,* Transportation Research Board of the National Academies, Washington, D.C., 2003, pp. 220–230. DOI:10.3141/1856-24.
4. Zlatkovic, M., P. T. Martin, and A. Stevanovic. Predictive Priority for Light Rail Transit: University Light Rail Line in Salt Lake County, Utah. In *Transportation Research Record: Journal of the Transportation Research Board, No. 2259,* Transportation Research Board of the National Academies, Washington, D.C., 2011, pp. 168–178. DOI:10.3141/2259-16.
5. Islam, M. T., J. Tiwana, A. Bhowmick, and T. Z. Qiu. Design of Light-Rail Transit Signal Priority to Improve Arterial Traffic Mobility. Presented at 91st Annual Meeting of the Transportation Research Board, Washington, D.C., 2012.
6. Kim, W., and L. R. Rilett. Improved Transit Signal Priority System for Networks with Nearside Bus Stops. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1925,* Transportation Research Board of the National Academies, Washington, D.C., 2005, pp. 205–214. DOI:10.3141/1925-21.
7. Ekeila, W., T. A. Sayed, and M. El Esawey. Development of Dynamic Transit Signal Priority Strategy. In *Transportation Research Record: Journal of the Transportation Research Board, No. 2111,* Transportation Research Board of the National Academies, Washington, D.C., 2009, pp. 1–9. DOI:10.3141/2111-01.
8. Lee, J., A. Shalaby, J. Greenough, M. Bowie, and S. Hung. Advanced Transit Signal Priority Control with Online Microsimulation-Based Transit Prediction Model. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1925,* Transportation Research Board of the National Academies, Washington, D.C., 2005, pp. 185–194. DOI:10.3141/1925-19.
9. Janos, M., and P. G. Furth. Bus Priority with Highly Interruptible Traffic Signal Control: Simulation of San Juan's Avenida Ponce de Leon. In *Transportation Research Record: Journal of the Transportation Research Board, No. 1811,* Transportation Research Board of the National Academies, Washington, D.C., 2002, pp. 157–165. DOI:10.3141/1811-19.
10. Xie, X.-F., S. F. Smith, G. J. Barlow, and T.-W. Chen. Coping with Real-World Challenges in Real-Time Urban Traffic Control. Presented at 93rd Annual Meeting of the Transportation Research Board, Washington, D.C., 2014.
11. Cesme, B., and P. G. Furth. Self-Organizing Traffic Signals Using Secondary Extension and Dynamic Coordination. *Transportation Research Part C: Emerging Technologies,* Vol. 48, 2014, pp. 1–15. DOI:10.1016/j.trc.2014.08.006.
12. Cesme, B., and P. G. Furth. Self-Organizing Control Logic for Oversaturated Arterials. In *Transportation Research Record: Journal of the Transportation Research Board, No. 2356,* Transportation Research Board of the National Academies, Washington, D.C., 2013, pp. 92–99. DOI:10.3141/2356-11.
13. Gershenson, C., and D. A. Rosenblueth. Modeling Self-Organizing Traffic Lights with Elementary Cellular Automata. Complex Systems, Vol. 19, No. 4, 2011, pp. 305–322. http://arxiv.org/abs/0907.1925.
14. Furth, P. G., B. Cesme, and T. H. J. Muller. Lost Time and Cycle Length for Actuated Traffic Signal. In *Transportation Research Record: Journal of the Transportation Research Board, No. 2128,* Transportation Research Board of the National Academies, Washington, D.C., 2009, pp. 152–160. DOI:10.3141/2128-16.
15. Cesme, B., and P. G. Furth. Multiheadway Gap-Out Logic for Actuated Control on Multilane Approaches. In *Transportation Research Record: Journal of the Transportation Research Board, No. 2311,* Transportation Research Board of the National Academies, Washington, D.C., 2012, pp. 117–123. DOI:10.3141/2311-11.

The Standing Committee on Traffic Signal Systems peer-reviewed this paper.