CoSign: A Fictitious Play Algorithm for Coordinated Traffic Signal Control

Shih-Fen Cheng
Marina A. Epelman
Robert L. Smith

Technical Report number 04-08
July 29, 2004

University of Michigan
Industrial and Operations Engineering
1205 Beal Avenue
Ann Arbor, MI 48109

CoSIGN: A Fictitious Play Algorithm for Coordinated Traffic Signal Control

Shih-Fen Cheng, Marina A. Epelman, and Robert L. Smith

Department of Industrial and Operations Engineering
University of Michigan, Ann Arbor, MI

Abstract

The problem of finding efficient coordinated signal timing plans for a large number of traffic signals is a challenging problem because of the exponential growth in the number of joint timing plans that need to be explored as the network size grows. In this paper, we employ the game-theoretic paradigm of fictitious play to iteratively converge to a locally optimal coordinated signal timing plan. Since there is only one traffic simulation required per iteration, the resulting algorithm is robustly scalable to realistic size networks modelled with high fidelity simulations. We report the results of a case study for the the city of Troy, Michigan where we experienced delay and throughput savings in excess of 10 percent for a network model of 75 signalized intersections.

1 Introduction

Providing an optimal, real-time signal control scheme to an array of traffic signals continues to be a major issue in traffic management. Optimization of signal timing plans at an isolated signal has already been well-studied. However, optimization of signal timing plans for a group of coordinating signals is still an active research area. Because it is computationally intractable to exactly solve the coordinated signal control problem even for a moderate-size network, most previous approaches propose approximation schemes through restrictions on the original problem. These restrictions can be imposed on the number of signals considered, the flexibility of the signal timing patterns, and the adaptiveness of the signal plans. Depending on the situation, traffic planners may choose various possible combinations of these restrictions.

Email addresses: chengsf@umich.edu (Shih-Fen Cheng), mepelman@umich.edu (Marina A. Epelman), rlsmith@umich.edu (Robert L. Smith).

Summarizing, we can classify research in traffic signal control into the following major categories:

Isolated intersections with cyclic schemes: The first such method developed is Webster's method [1]. Webster's method and its later variants have been used widely in practice. They optimize the cycle time and also the split of green times among different phases [1].

The signal plans provided by Webster's method and its variants are non-adaptive. Later researchers proposed adjusting the parameters of these cyclic plans frequently in order to reflect the dynamics of changing traffic conditions. These approaches are examples of off-line adaptive systems. SCOOT [2] and SCATS [3] are two notable examples that have been developed and used more recently.

Isolated intersections with non-cyclic schemes: If we think of signal plans as sequences of decisions as to which phase to give green time to, we are able to anticipate and react to the dynamic environment more rapidly. This enables the design of systems that are able to take into account real-time traffic data in computing optimal plans. UTOPIA [4], PRODYN [5], SPPORT [6], OPAC [7], ALLONS-D [8], and COP [9] all belong to this class. A more complete review of these adaptive non-cyclic control schemes can be found in [10]

A network of intersections: Long before the introduction of digital computers, traffic engineers have tried to manually coordinate a series of signals by following the so called *maximum through-band* design. This design aims at finding the length of the cycle time and the offsets of the beginning of the cycles for a series of signals. Its goal is to allow platoons of vehicles to pass the series of signals without being stopped. For most research on cyclic schemes at isolated intersections, this is a natural extension. SCATS and SCOOT mentioned earlier can both be extended to conduct this kind of coordination. Of course, real time traffic data can also be used to adaptively adjust the coordination parameters. REALBAND [11] and SCOOT are examples of such approach. On observing real-time flow data from the detectors, REALBAND and SCOOT will try to form progression bands that minimize any delay- or stop-related performance measures. In traffic networks where main arteries can be identified easily, progression-based systems will work well. However, for a generic traffic network where main arteries cannot be identified in a straightforward way, a more general methodology is required.

General coordinated traffic signal control scheme allows arbitrary signal timing plans at all signals. Due to its intractable nature (as we will see in the

[1] A *phase* is a collection of traffic movements that receive right-of-way simultaneously. Therefore, all movements within a phase must be non-conflicting.

next section), it can not be solved directly. Therefore, in most related research, heuristics have been proposed, usually without theoretical convergence guarantees. In this paper, we present a model and an algorithm that searches for locally optimal non-cyclic signal timing plans for a group of coordinated signals (we call this approach **CoSIGN**). To limit the number of decisions, the planning horizon is discretized into a manageable amount of equal-length time periods. The applicability of our approach is demonstrated by a test case based on the real traffic network of Troy, Michigan.

The paper is organized as follows. In section 2, the motive for using a game-theoretic approach and an intuitive description of our methodology are presented. In section 3, we formally introduce the technical background underlying our approach. In section 4, we formally state the model. In section 5, the test case and the results of the experiment are discussed. Future work is proposed in section 6.

2 Motive for a Game-Theoretic Approach

In this section we briefly describe the motive and the intuition for using *game theory* in solving coordinated traffic signal control problems. Although some game theory related terms are mentioned throughout this description, their formal definitions will be deferred to the next section. The intuition behind our approach is emphasized here.

As mentioned in the introduction, we assume that the signal plan we would like to come up with for a single signal is a sequence of decisions over a discretized time horizon, where the decision for each time period is which phase to give green time to. Let the upper bound on the number of phases be $S_{\max}$, the number of time periods be N and the number of intersections be I. The number of possible signal plans for the group of signals is then bounded by $(S_{\max})^{I \cdot N}$. This problem quickly become intractable as we increase I and N. However, if we decompose the problem into smaller subproblems, we may be able to find a sufficiently good solution in a reasonable amount of time. The decomposition of the problem can be done by assuming that each signal at each period is an independent decision maker (or agent). By performing this decomposition, the centralized decision problem that involves a decision set with $(S_{\max})^{I \cdot N}$ possible decisions can then be transformed into $(I \cdot N)$ small subproblems, each with at most $S_{\max}$ decisions. However, if we decompose the problem without considering the interactions among these independent decision makers, we are just solving $(I \cdot N)$ isolated signal control problems.

In order to effectively deal with this coordination problem for a large number of decision makers which in general have conflicting interests, we turn to *game* *theory* that originates from economics. Modern *game theory* was created after von Neumann and Morgenstern [12] in 1944 and quickly became a powerful tool in explaining and predicting the behavior of a group of rational decision makers when their well-beings are associated with the joint actions of all decision makers. If each decision maker who controls a time period for a signal is viewed as a *player* in the *game*, and the travel time for all vehicles in the traffic network is viewed as a *common payoff* for every player, the traffic signal problem can then be formulated as a *game* of identical interests. The notion of a solution to a game is that of a Nash equilibrium, which can be viewed as a coordinate-wise local optimum for our game of identical interests. Intuitively, a joint decision is a *Nash equilibrium* if no individual player can improve its payoff value by unilaterally deviating from the original joint decision.

It is a well-known fact that finding Nash equilibria is a hard problem [13]. One of the earliest algorithms used to find Nash equilibria is an iterative process called *fictitious play* [14,15]. The primary pitfall of fictitious play (FP) is that in general it does not converge to an equilibrium. However, Monderer and Shapley [16] showed that for a special class of games, namely *games with identical interest*, FP will converge to equilibrium. Since almost all unconstrained discrete optimization problems can be formulated as *games with identical interest*, this result has recently inspired researchers in optimization [17] to introduce FP as an optimization tool. In this paper, after the traffic signal control problem is formulated as a game, we can then apply the FP algorithm in order to find a solution for the problem.

3 Game Theory and the FP Algorithm

In this section, we formally define a game and the solution concept of a Nash equilibrium, and discuss how one can use FP to find a Nash equilibrium of a game.

3.1 Game Theory Fundamentals

Game theory studies how independent decision makers would act under the assumption that individual's payoff will be determined by the joint actions. We now define the components of a game.

- **Players:** Each independent decision maker in the game is defined as a player. Every player has a finite set of decisions called **strategies** that it can choose from. A **(mixed) strategy** is a probability distribution over the set of the player's strategies. A **joint strategy** is a specification of (mixed) strategies for all players.

- **Payoff function:** For every player, its associated payoff function is defined as a mapping from joint strategies to the corresponding payoffs this player will get were these joint strategies played (or expected payoffs, if mixed strategies are played). In general, players may have different payoff functions. However, in this paper, all players will be assumed to have identical payoff functions.
- **Best reply function:** Given an arbitrary joint strategy, a player's best reply function will return the strategy that gives this player its highest payoff value, assuming that all other players use the strategies specified in this joint strategy. As we will see later, this is the critical operation in our approach.
- **Nash equilibrium:** A joint strategy is a Nash equilibrium if no individual player can improve its payoff by unilaterally deviating from the play of the original joint strategy. More precisely, a joint decision is a Nash equilibrium if for every player, its best reply against this joint strategy is its current decision. In other words, Nash equilibrium is a fixed point to the best reply function.

The first important existence theorem, proposed by Nash [18], stated that every finite game in strategic form [2] has a mixed strategy equilibrium.

For complete treatment of these introductory terms and concepts, we refer to Fudenburg and Tirole [19].

3.2 FP Algorithm

Computing Nash equilibria can be a difficult task. McKelvey and Mclennan's work on GAMBIT [13] is an excellent source for various computational methods in finding Nash equilibria. In this research, we will use a simple-to-implement iterative algorithm which is a variation of FP [17].

The convergence results for the FP algorithm and its variants are stated in [16,17]. Since in this paper we are mainly interested in solving the traffic signal control problem, most technical details are neglected here. We will refer interested readers to [17] for complete treatment.

The intuition behind FP lies in the theory of learning in games. In a FP process, every player assumes that other players are playing unknown stationary

[2] A game is said to be in strategic form if it has a finite set of players, each player has a nonempty strategy set, and each player's payoff functions are defined for all joint strategies. A strategic game is finite if the number of players and all players' strategy sets are finite.

mixed strategies and they will try to learn them iteratively. This unknown stationary mixed strategy is represented as *beliefs*, and is shared among all players. The *belief* for player i is calculated by finding the relative frequency of all strategies from the history of its past plays. During each iteration, each player will try to find a *best reply* against its *beliefs* on how other players will play. These *best replies* are then included in the history of past plays and the beliefs are updated accordingly. To start the FP process, an arbitrary joint strategy is used.

The FP algorithm doesn't converge in general. However, for games with identical interest as in our case, the FP algorithm is guaranteed to converge in *beliefs* to equilibrium [16].

FP is computationally expensive to be implemented in practice. Lambert et al. [17] thus suggested a variant they called *sampled fictitious play* (SFP) that is computationally practical. The convergence result for this variant is also proved in [17]. SFP is very similar to FP except the best reply evaluation in each iteration is done against a random *sample* drawn from the belief distribution instead of the belief distribution itself. In practice, one uses a sample of size one.

The SFP algorithm, with sample size one, is described as follows:

(1) **Initialization:** An initial joint strategy profile is chosen arbitrarily. It is then stored in the history.
(2) **Sample:** A strategy is independently drawn from the history of each player (i.e., each past play in the history is selected with equal probability).
(3) **Best Reply:** For every player, the best reply is computed by assuming that all other players play the strategies drawn in step 2).
(4) **Update:** The best replies obtained in step 3) are stored in the history.
(5) **Stop?** Check if the stopping criterion is met, if not go to step 2), otherwise stop.

The SFP algorithm was first implemented and applied to a dynamic traffic routing assignment problem by Garcia et al. [20]. When compared to previously established method, SFP algorithm is able to obtain a solution of same quality significantly faster. Lambert and Wang [21] further demonstrated the superiority of the SFP algorithm over simulated annealing and random search in a communication protocol design problem.

4 Traffic Signal Control Problem

As mentioned in the introduction, due to the intractability of the problem of finding optimal signal timing plans even for moderate-size networks, researchers either restrict the space of solutions by searching for parameters of predetermined cyclic patterns, or limit the number of signals considerably.

Instead of trying to find solutions for restricted versions of the problem, our approach will try to find solutions for the full-scale coordinated signal planning problem by using the SFP algorithm.

Currently all our results are based on simulations, and our approach aims at solving the traffic signal control problem for a fixed traffic network. However, with fast enough execution speed, we will be able to update the traffic network and recompute the signal timing plans in a short time, thus allowing the implementation of a rolling-horizon control scheme. I.e., for some fixed interval, the latest traffic condition will be updated and the complete plan will be recomputed, and the resultant plan will then be used until next update.

Porche and Lafortune [22] utilize a Cournot adjustment process for coordinated signal control analogous to that used for route guidance in Wunderlich et al. [23]. However this algorithm is subject to the same failure to converge. In our work, a game-theoretic model is used and the solution is found by an algorithm (SFP) which is known to converge. The convergence of SFP algorithm does not depend on any property of the objective function; therefore even values returned by black-box-type simulators can be used. Since simulators are often used to evaluate performance, it is important to have a convergence guarantee under very general performance functions.

In the following sections, we will describe the game-theoretic model and the performance measure.

4.1 Formulating Coordinated Traffic Signal Control As a Game

The following notation will be used in describing a traffic signal control problem:

(1) $\mathbf{I} = \{1, 2, \ldots, I\}$: set of signalized intersections.
(2) $\mathbf{N} = \{1, 2, \ldots, N\}$: set of time periods (with the assumption that each time period has an equal length of δ seconds).
(3) $\mathbf{S}_i = \{1, 2, ..., S_i\}$: set of permissible signal phases for intersection i, $i \in \mathbf{I}$.

To formulate the problem as a game, the following important elements must be specified:

- **Player:** each tuple (i, n), where $i \in \mathbf{I}$, $n \in \mathbf{N}$, is a player and is written as $P_{i,n}$.
- **Strategy Space:** for each player $P_{i,n}$, its strategy space will be the set $\mathbf{S}_i$. Player $P_{i,n}$'s decision is denoted as $D(i, n)$.
- **Payoff function:** by collecting decisions $D(i, n)$ from all players, a signal timing plan for the planning horizon is formed. By sending this plan to the traffic simulator, we can find the total travel time experienced by all drivers as the payoff function value for all players.

4.2 Simulation by INTEGRATION-UM

Note that a simulator will be required in order to evaluate total travel time. In our experiment, the simulation is done by INTEGRATION-UM, developed by Van Aerde [24] and modified by researchers in the ITS RCE at the University of Michigan. INTEGRATION-UM is an event-based, meso-scopic and deterministic traffic simulator. A detailed description of specifications of INTEGRATION-UM can be found in Wunderlich's PhD dissertation [25].

The signal timing plans in INTEGRATION-UM are specified by giving parameters that define cyclic patterns (i.e., cycle length, green split, offset, and lost time). INTEGRATION-UM was modified in order to take players' joint strategy as input. Unlike cyclic signal timing plans, where the lost time is incurred during phase transition, the timing plans specified by joint players' decisions incur lost time at one intersection only when two consecutive players (i, n), $(i, n+1)$ have different decisions. Note that with a short enough time period (δ), the player model can emulate any cyclic pattern.

We selected INTEGRATION-UM as our traffic simulator purely on the basis of convenience of the implementation since its source code was available to us. We would like to emphasize that since our system architecture is flexible with regard to the type of simulator used, any traffic simulator could be used here. The only requirement is that it must be able to accept the signal timing plan generated by our algorithm as input, and output necessary information to our SFP solver, as described below.

4.3 Best Reply Approximation

Given a sampled joint strategy, every player's best reply can be computed exactly by invoking the simulator for each strategy. However, since the execu-

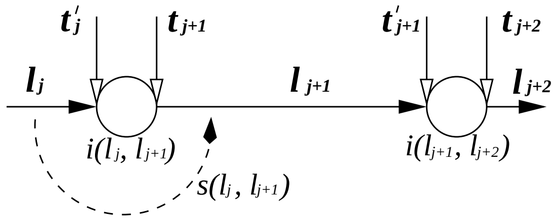

Fig. 1. Illustration of temporal relations in best-reply approximation.

tion of the simulator is computationally expensive, we would like to come up with better ways of evaluating best replies, preferably with fewer simulation executions.

Two important observations can assist us in designing appropriate best reply approximation:

(1) When trying to find the best reply for any player, the exact payoff is not important for our purpose. A preference order over the strategies will be enough for us.
(2) Since each player only governs a time period of length δ at one signal, when all other players' strategies are fixed, the impact of single player's decision change is very limited.

These observations provide useful guidelines in designing a best reply approximation. From observation 2), we can locally change one player's strategy choice in order to measure the relative benefit of choosing different strategies. According to observation 1), the order of computed benefits will be enough for us to find the best reply.

Assume all necessary information is generated by running the simulator with the sampled strategy. Best reply approximation for every player $P_{i,n}$ can be summarized as follows (refer to Fig. 1 for an illustration on important temporal relationships):

- Identify the cars that are *scheduled* [3] to go through the signalized intersection i during time period n, and characterize the remaining portions of their routes as follows:
  - · Let C be the set of all vehicles that are scheduled to pass the signalized intersection i.
  - · For all $c \in C$, let $L_c(l)$ be the next link car c will visit, given that it is currently on link l. If link l is car c's destination, $L_c(l) = 0$.

[3] INTEGRATION-UM internally generates an estimated departure time for every vehicle when it enters any link; note that the real departure time may be greater then the estimated time due to various reasons [25]. However, any vehicle that has estimated departure time between $(n-1)\delta$ and $n\delta$ will be said to be *scheduled* to go through intersection i during period n.

- Let $V(l,t)$ be the estimated travel time to traverse link l for a vehicle that enters link l at time t.
- Let $i(l_1,l_2)$ be the intersection connecting link l_1 to link l_2.
- Let $S(l_1,l_2)$ be the set of phases of the signal at intersection $i(l_1,l_2)$ allowing the movement from link l_1 to link l_2.

- For every phase $s \in \mathbf{S}_i$, compute the estimated remaining travel time Δ_s for all the vehicles that are scheduled to go through the signalized intersection i during time period n, if green time were assigned to phase s as $\Delta_s = \sum_{c \in C} R_c$, where R_c is calculated by following procedure:

(1) Assume that car c is on link l_0 at the beginning of period n. Let t'_0 be the scheduled departure time from link l_0 (by definition, $(n-1)\delta < t'_0 \le n\delta$). Let $j = 0$, $R_c = 0$, $D(i,n) = s$.
(2) If $L_c(l_j) = 0$, end the calculation and return R_c.
(3) If $L_c(l_j) \neq 0$, let $l_{j+1} = L_c(l_j)$, $t_{j+1} = \min\{t : t \ge t'_j, D(i(l_j,l_{j+1}), \lceil t/\delta \rceil) \in S(l_j,l_{j+1})\}$.
(4) Let $t'_{j+1} = t_{j+1} + V(l_{j+1},t_{j+1})$, $R_c = R_c + (t'_{j+1} - t'_j)$.
(5) $j = j+1$, go to step 2.

- Approximate best reply for player $P_{i,t}$ will then be:

$$s^*_{i,t} = \arg\min_{s \in \mathbf{S}_i} \Delta_s \tag{1}$$

When there is a tie in (1), it's not obvious how it should be broken. Two possible tie-breaking schemes are: keep the original decision, or break the tie randomly. These tie-breaking schemes were tested empirically, and the random tie-breaking scheme performed significantly better. Therefore in our later case study, we used a random tie-breaking rule.

The algorithm **CoSIGN** is a combination of the SFP algorithm and the best reply approximation, both mentioned previously.

5 Case Study - The Troy, Michigan Network

In order to test the performance of the CoSIGN algorithm, we used a realistic traffic network built by Wunderlich [23,25]. This case study model was constructed based on the real traffic network of Troy, Michigan. To ensure fidelity, this model was carefully calibrated against empirical measurements. To maintain this fidelity, we did not modify the model in any way except to insert the signal plan we generated. In our experiments we compared the traffic timing plan generated by the CoSIGN algorithm to the cyclic plan provided in Wunderlich's model.

The network topology of the Troy, Michigan network is shown in Fig. 2.

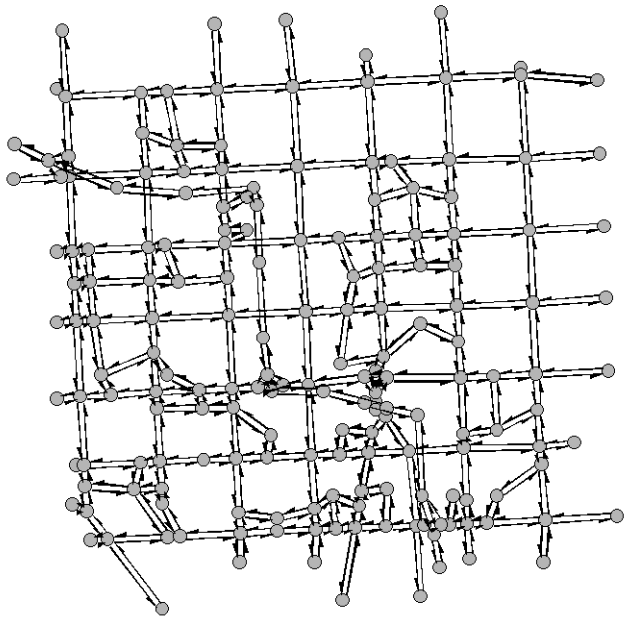

Fig. 2. The Troy network, composed of 529 links, 200 nodes and 50 zone centroids that can serve as origins or destinations.

Here are the settings used in our experiment:

- Length of time periods (δ): 10 seconds.
- Number of time periods: $N = 720$
- Number of signalized intersections: $I = 75$
- Number of players: 54,000
- Maximum number of CoSIGN iterations: 20

The original cyclic pattern was used as the initial solution. We assumed that all vehicles will follow shortest free-flow paths from their origins to destinations.

Since mixed strategy cannot be interpreted intuitively here, the best pure strategy seen so far will be reported as the solution, as in Lambert and Wang's work [21].

The termination criterion for the algorithm was as follows: if the algorithm saw 10 non-improving iterations before reaching the maximally-allowed iterations, it terminated. Whenever the algorithm has terminated, if the best value was improved within this run, we performed another run of the algorithm, initialized with the best decision found in the current run. We repeatedly restarted the algorithm until no improvement was observed between two consecutive runs.

In Table 1, two cases are reported. Case I is the comparison between solution delivered by CoSIGN and the cyclic pattern under original traffic condition. In order to test how robustly CoSIGN performs in a dynamic environment,

Table 1
Experiment Result Summary

	Cyc. Pattern	CoSIGN Sol.	Diff.
Case I: Original traffic flow scenario			
Total travel time (min.)	490,597	433,678	11.6%
Throughput (# vehicles)	9,320	10,265	10.1%
Case II: Randomly generated traffic flow scenarios			
Total travel time (min.)	487,626	431,763	11.4%
Throughput (# vehicles)	8,970	10,273	14.5%

we tested it in various traffic scenarios and summarize the result in Case II. These random traffic scenarios were generated by assuming that the actual number of cars flowing into the traffic network (per origin/destination pair) is a Poisson random variable. The realization of these Poisson random variables is generated by using flow rate in the original model as the mean.

Note that since random sampling is involved in the algorithm (step 2 of the algorithm), to obtain statistically significant results, we ran the algorithm multiple times. For Case I, the test instance was executed 15 times, and the average is reported. For Case II, we randomly generated 12 different traffic profiles and each profile was also executed 15 times. These results were then averaged and reported.

In Table 1, both total travel time and throughput are reported. The throughput is calculated by counting the number of cars reaching their destinations from the 30$^{\text{th}}$ minute to 75$^{\text{th}}$ minute.

From the summary of the experiment in both cases, we can see that solutions delivered by CoSIGN is pretty robust in both criterions. As for the cyclic signal plan, although its total travel time is roughly the same, about 4% drop in the throughput can be seen.

Also, from Fig. 3, we can see that the CoSIGN algorithm performs better then cyclic signal plan, in terms of the throughput, during most of the peak hour (from 30$^{\text{th}}$ to 75$^{\text{th}}$ minute). Since most vehicles reach their destinations by 68$^{\text{th}}$ minute in CoSIGN's solution, that explains why cyclic solution has higher throughput after 68$^{\text{th}}$ minute.

In Fig. 4, we see how fast the CoSIGN algorithm converges. In this example, the CoSIGN algorithm is restarted two times (the third run is not plotted since no improvement was observed). We can see that about half of the improvement was gained in the first 5 iterations. This suggests that the CoSIGN algorithm can guide us to a good solution very quickly.

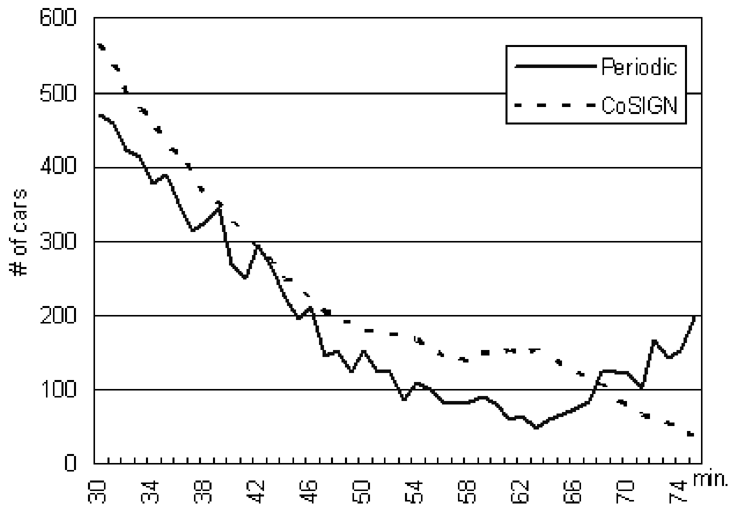

Fig. 3. Number of vehicles arriving at their destinations from 30^{th} to 75^{th} minute (for one of the random traffic flow scenarios).

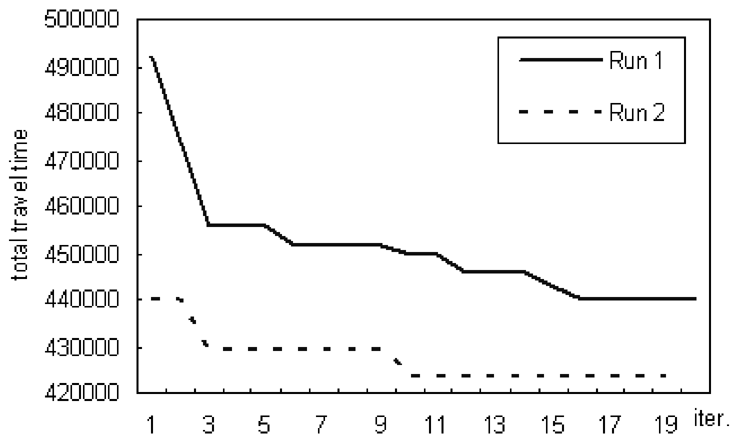

Fig. 4. The best value found at each iteration of the CoSIGN algorithm.

Note that above test is performed on a Pentium III, 800MHz PC with 256MB RAM, running Windows 2000. On average, time spent in each run is 19.65 minutes. However, if we integrate solver and simulator, thus eliminating time-consuming disk I/O, we can save up to 50% execution time.

6 Future Work

A larger and more detailed real-world test case, based on the urban traffic network for metropolitan Seattle is available from Mitretek Systems, Inc. Our hope is to test CoSIGN algorithm on this large scale instance to verify the magnitude of savings in trip time we experienced in the Troy model.

Garcia et al. [20] suggests the use of a FP algorithm in vehicle routing. A natural extension of the results from this paper and Garcia et al's result will be to combine both traffic signal control and vehicle routing. In doing so, each driver who belongs to the guided class will be treated as a player in the SFP algorithm. Since in this model, both drivers and signals will anticipate each others' decisions, we can expect that the model will guide them to choose their decisions in a coordinated way. The improvement we can achieve by combining route guidance and coordinated signal timing could be quite substantial.

7 Acknowledgements

The authors gratefully acknowledge Karl Wunderlich's valuable suggestions and advice. This work was partially supported by the National Science Foundation under Grant DMI-0217283.

References

[1] F. V. Webster, B. M. Cobbe, Traffic Signals, Road Res. Tech. Rep. 39, HMSO, London (1958).

[2] P. B. Hunt, D. I. Robertson, R. D. Bretherton, R. I. Winton, SCOOT - a traffic responsive method for coordinating signals, in: Laboratory Report no. LP 1014, Transportation and Road Research, Crowthorne, Berkshire, England, 1981.

[3] A. G. Sims, The sydney coordinated adaptive traffic system, in: Urban Transport Division of ASCE Proc., Engineering Foundation Conference on Research Directions in Computer Control of Urban Traffic Systems, New York, NY, 1979, pp. 12–27.

[4] V. Mauro, D. DiTaronto, UTOPIA, in: Proc. of the 6th IFAC/IFIP/IFORS Symposium on Control, Computers and Communication in Transportation, no. 12, 1990, pp. 245–252.

[5] J. J. Henry, J. L. Farges, J. Tuffal, The PRODYN real time traffic algorithm, in: B. Baden (Ed.), 4th IFAC-IFIP-IFORS Conference on Control in Transportation System, Germany, 1983.

[6] S. Yargar, B. Han, A procedure for real-time signal control that considers transit interference and priority, Transpn Res.-B 28 (1994) 315–331.

[7] N. H. Gartner, OPAC: A demand-responsive strategy for traffic signal control, Transpn Res. Rec. 906 (1983) 75–81.

[8] I. Porche, S. Lafortune, Adaptive look-ahead optimization of traffic signals, Tech. Rep. CGR 97-11, Dept. of Electric Engineering and Computer Science, The University of Michigan (1997).

[9] S. Sen, K. L. Head, Controlled optimization of phases at an intersection, Transpn Sci. 31 (1997) 5–17.

[10] I. Porche, Dynamic traffic control: Decentralized and coordinated methods, Ph.D. thesis, Univ. Michigan (1998).

[11] P. Dell'Olmo, P. B. Mirchandani, REALBAND: An approach for real-time coordination of traffic flows on a network, Transpn Res. Rec. 1494 (1995) 106–116.

[12] J. von Neumann, O. Morgenstern, Theory of Games and Economic Behavior, 2nd Edition, Princeton Univ. Press, Princeton, 1947.

[13] D. McKelvey, A. McLennan, Computation of equilibria in finite games, in: Handbook of Computational Economics, Vol. 1, Elsevier, 1996.

[14] G. W. Brown, Iterative solution of games by fictitious play, in: Activity Analysis of Production and Allocation, John Wiley, New York, 1951, pp. 374–376.

[15] J. Robinson, An iterative method of solving a game, Annals of Mathematics 54 (1951) 296–301.

[16] D. Monderer, L. S. Shapley, Fictitious play property for games with identical interests, J. Econom. Theory 68 (1) (1996) 258–265.

[17] T. J. Lambert, M. A. Epelman, R. L. Smith, A fictitious play approach to large-scale optimization, Oper. Res.Forthcoming.

[18] J. Nash, Equilibrium points in n-person games, in: Proc. of the National Academy of Sciences, Vol. 36, 1950, pp. 48–49.

[19] D. Fudenberg, J. Tirole, Game Theory, MIT Press, 1991.

[20] A. Garcia, D. Reaume, R. L. Smith, Fictitious play for finding system optimal routings in dynamic traffic networks, Transpn Res.-B 34 (2) (2000) 146–157.

[21] T. J. Lambert, H. Wang, Fictitious play approach to a mobile unit situation awareness problem, Tech. rep., Univ. Michigan (2003).

[22] I. Porche, S. Lafortune, A game-theoretic approach to signal coordination (1997).

[23] K. E. Wunderlich, D. E. Kaufman, R. L. Smith, Link travel time prediction for decentralized route guidance architectures 1 (1) (2000) 4–14.

[24] M. V. Aerde, J. Voss, G. McKinnon, INTEGRATION Simulation Model User's Guide, Queen's University (1989).

[25] K. E. Wunderlich, Link travel time prediction for dynamic route guidance in vehicular traffic networks, Ph.D. thesis, Univ. Michigan (1994).