2010 13th International IEEE
Annual Conference on Intelligent Transportation Systems
Madeira Island, Portugal, September 19-22, 2010

Urban Traffic Control System using Self-organization

Gianfilippo Slager, Michela Milano

Abstract— **This paper presents a Urban Traffic Control (UTC) system, inspired to techniques exploited by social insects to coordinate themselves and specialize their behaviour, without any centralized coordination or explicit communication. Local traffic is handled at an intersection by a controller, executing simple local reactive rule-based policies. Every intersection controller chooses on its own which policy to use, according to stimuli perceived from the environment through sensors, thus being influenced by other controllers indirectly. The execution of simple reactive local policies lets an overall traffic control to emerge, with complex behaviours not defined a priori and unaware to the single controller.**

Simulations demonstrate that the exploiting of emergent behaviours is desirable, as the system outperforms traditional traffic control methods. Dynamic specialization makes the system able to cope with traffic variation and to perform traffic shaping in a positive way.

I. Introduction

TRAFFIC is a serious problem for every city, as traffic congestions are common in everyday life. There are many technical and social aspects regarding mobility that affect traffic and need improvements, one of these is how traffic lights regulate vehicle flows at intersections. Since the introduction of traffic lights, engineers deployed different solutions for a good traffic control; one of the oldest approaches is to synchronize lights phases to achieve the so-called "green-wave". Synchronization approaches were refined over the year, but nowadays traffic is so changing over time and days that no static solutions are feasible anymore, thus the trend is to replace them with the introduction of various kinds of dynamic traffic controllers.

There are many different ways to design and deploy UTC systems; these can be roughly divided into three categories: centralized, partially decentralized and fully distributed systems.

- In centralized systems, a control centre collects information from road sensors scattered through the city, to have a comprehensive knowledge about the controlled scenario. This knowledge concurs to the evaluation of a traffic plan, i.e. to the assignment of phases and timings to traffic lights. In case of need, traffic experts may override control system decisions: since optimization is based on limited-horizon forecasts, there could be mispredictions needing manual adjustments to the plan. Different solutions basically differ in the evaluation of the control strategy and inherit in some way from the TRANSYT off-line optimization model [Rob69]; for example, SCOOT [Rob90] is a largely deployed centralized solution.
- In decentralized systems there is more than one entity capable to take decisions, while a master entity still exists to coordinate and communicate directives to the behaviour or goal to pursue. This approach is adopted by two systems, currently in production: SCATS [Sim79] and UTOPIA [Pee02].
- Fully distributed systems do not have any control centre to coordinate operations and generate traffic plans: every single intersection controller decides on its own, taking initiatives of neighbours into account. The distributed approach for urban traffic control systems is still under research and development, as there is no evidence of large scale deployment. Many proposals about distributed solutions exist: they are essentially agent-based, where an agent is in charge of handling traffic lights at an intersection and performs actions with regards to the local traffic situation only, as in [Pri09], while in other cases the information coming from surrounding agents is also taken into account [Miz08]. Another possibility would be to let vehicles commerce the passage on their own, without any broker, as in [Mil10]: this would be as interesting as visionary, since several concerns about technical feasibility, communication standardisation and security, costs and safety would be involved.

As stated before, static optimization is not able to adapt to changing needs in traffic. Instead, centralized solutions are able to reach good performances, but require significant investments: a control centre is expensive to be installed, maintained and supervised by qualified personnel; the control centre needs to be connected to every sensor and to every traffic light controller to apply specific traffic plans; centralized solutions are based on traffic models coming from empirical studies, thus needing tuning as traffic trends evolve. When controlling big scenarios, centralized systems do not scale well and require more and more investments in hardware; decentralized systems are an answer to scaling needs. On the other side, distributed approaches would be

Manuscript received the 20th of July 2010.
Gianfilippo Slager is with the DEIS, the Department of Electronics, Computer Science and Systems of the University of Bologna, Italy. He received his Master Degree degree in Computer Science Engineering in 2010 (phone: +39 338 6612497; e-mail: slager@fastwebnet.it).
Michela Milano is associate professor at DEIS, the Department of Electronics, Computer Science and Systems of the University of Bologna, Italy. She received her Ph.D. degree in Computer Science in 1997. (e-mail: mmilano@deis.unibo.it)

simpler and scale better, but some of them need heavy communication among their parts to coordinate and pursue global traffic control: using MARL (Multi Agent Reinforcement Learning), for example, the communication needs would grow exponentially with the number of agents, as stated in [Baz09] and in [Bus08]. Communication requires mutual knowledge, thus reconfiguration of neighbouring agents is needed when an agent is introduced or removed in the network: in MARL the costs for a control centre are removed, but not for communication.

Following principles regarding Complex Adaptive Systems (CAS) and after a study of complex systems common properties (the traffic is indeed a complex open system) we conjecture that a smart and planned global traffic flow could emerge as the result of local decisions, automatically made by local controllers, executing simple policies in an emergent fashion. Emergent systems are very common in Nature and colonies of social insects are one of the most interesting examples for our purposes. Following these principles, our UTC system offers unlimited scalability, adaptability to traffic conditions and maximizes road network capabilities, while totally removing at the same time the costs associated to the control centre with trained personnel, and to communication infrastructure.

We take inspiration from two academic works: [Ger07] presents local policies able to reach global traffic control through emergence, but, since every policy is thought for some traffic densities and do not perform well on other ones, there is still the problem of choosing the right policy; on the other side [Oli04] discusses a mechanism that offers the capability to choose among different local policies with respect to traffic density, but executes non-reactive policies. Taking the best from these two approaches, we develop a model to implement and deploy an effective and simple traffic light control system. The conceived solution is able to choose among different local polices, which react to traffic conditions in real-time. To achieve this, local policies are seen as tasks performed by a social insect in a swarm-based environment: as specialization leads to better performance of a swarm of insects, the capability of intersection controllers to perform the right policy gives our system a good ability to cope far different traffic needs with success.

The structure of the paper is as follows: Section II presents local policies capable to achieve emergent behaviours, while Section III describes how policies are selected according to the traffic conditions. Section IV describes experimental results. A discussion and open issues conclude the paper.

II. LOCAL POLICIES FOR EMERGENT GLOBAL BEHAVIOURS

According to [Gol99], emergence is the way complex systems and behavioural patterns arise out from the sum of simple local interactions between elements, which are unaware of their role in the whole: in this paper the interactions among competing traffic flows are driven by traffic lights, that are unaware of the system global state, but act so that the emerging result is an overall traffic behaviour that seems to be smart and planned. Local policies are conceived thinking the system as a whole without giving any entity the notion of the whole: this is the key for scalability and adaptability.

Once a traffic control policy is chosen and executed by an intersection controller, the input and output lanes are sensed for cars and traffic lights state is accordingly assigned to the intersection. This happens in a continuous loop like in every classic digital control system: sensing, evaluation, action.

The concept of timing and phases for traffic lights is abandoned: a light does not change following any clock but after sensing, while the traffic lights state is chosen from a limited set of states that fulfils safety requirements for the intersection; state transitions also follow the common requirements for a safe handling of traffic flows. Reactive policies have the capability to fulfil incoming and outgoing traffic needs with the slowest delay: traditional phases have only drawbacks, because they need the system to guess what the traffic will be after a while, instead a reactive control makes decisions basing on what is really happening.

Every policy is a mediation between needs of upstream and downstream intersections: on low demands an intersection controller tries to indulge upstream neighbours, while on heavier traffic the will is to be more respectful of downstream ones.

A. Low traffic

When an intersection is charged with low traffic, the most important need is to react fast to incoming vehicles, giving them the right to cross as soon as possible, at best without having them to stop and lose their kinetic energy. In very low traffic, crossing conflicts have a low probability, but when vehicle density slightly increases, it is important to reduce the chance of conflicts on intersections through what literature calls "traffic entropy reduction": intersections group vehicles in clusters, such that the probability of conflicts is reduced. SOTL-Request and SOTL-Platoon [Ger07] are the control policies that realize these concepts.

B. Average traffic

On average traffic, i.e. when it is not possible to avoid or reduce conflicts, the main goal is to reduce variance (in a statistical sense) of road traffic densities, i.e. not to have empty intersections and congested ones. This is achieved regulating traffic lights to have on output lanes a balanced and uniform throughput of vehicles; following intersections receive a controlled flow because traffic bursts are filtered and congestions become less probable as incoming traffic increase.

SOTL-Phase and SOTL-Marching [Ger07] policies achieve these goals and homogeneously scatter traffic over the whole controlled network; the latter performs best for more dense traffic. SOTL-Marching is a "blind" policy, i.e. does not take into account information from sensors, but simply execute phases to limit the passing of vehicles to the maximum throughput allowed on the output lane.

C. *Heavy-congested traffic*

Traffic on peak hours can be so heavy that it is not possible to avoid jams. There are also cases where local congestions happen, a typical example is the end of sport events. In these situations of emergency it is important to resolve local congestions as fast as possible, giving less fairness on input lanes queues to maximize throughput on output lanes. Deadlocks have to be absolutely avoided, i.e. no vehicles may remain jammed in the middle of an intersection, obstructing other directions. This is achieved allowing vehicles to cross only if there is room in the output lanes the vehicles are going to.

Since [Ger07] does realize no such policy, and nothing has found in literature to fulfil these requirements, a new control policy is developed for that purpose, named SOTL-Congested. It follows the simple rules stated in the following:

SOTL-Congested rules

1. *If there is a jammed car in the middle of the intersection, or if there is no room for cars in the output lanes, every light should be and remain red until the intersection clears.*
2. *If there is a green light, check if there are conditions to change lights;*
    a. *Check if there is no room for cars in the active output lanes; if so put the lights red.*
    b. *If there is a red light without cars in the output lane, while the green light has no cars in the input lane, put green that red light.*
3. *If there are no green lights, check if there are conditions to put a light green: If there is room for cars on output lanes, put green the light with less cars on the corresponding output lane.*

III. DYNAMIC POLICY SELECTION

Traffic control policies have different performance because developed for different traffic densities. Therefore a set of mechanisms able to choose the right policy in every traffic condition is needed. [Oli04] is an interesting work that addresses exactly this aspect, taking inspiration from [The97], the original research from natural sciences. In this paper we refer to the original model. [The97] proposes a model of division of labour in insect societies, in which every insect from a colony has to decide what to do, without any centralized decision maker. In Nature the environment stimulates social insects to perform tasks, as well as other insects do through a mechanism called stigmergy [Gra59]. For example, ants stimulate others through the release of pheromone.

Despite social insects, intersections and traffic lights (the actors) are steady, while vehicles (the objects of actions) are moving; another difference is that vehicles release pheromone on the roads, not traffic lights, but this does not affect the validity of the model.

Summarizing, in our abstraction vehicles release pheromone passing on the roads, while control systems at the intersections are stimulated by this pheromone through mechanisms described below.

A. *Pheromone as traffic density*

Pheromone is used as a metaphor, realized by the control system itself, counting incoming and outgoing vehicles at the intersection. Its level indicates the density of traffic in a road: the more the pheromone the more the traffic, following the time-discrete law

$$p_l(k+1) = \beta\, p_l(k) + \gamma\, n(l,k);\; p_l(0) = 0$$

$p_l(k)$ is the pheromone level for lane l at time $k\Delta t$, Δt is the sampling interval of road sensors, β is the pheromone dissipation rate, γ the amount of pheromone left by any vehicle, $n(l,k)$ is a function that counts the number of vehicles in a lane at the time $k\Delta t$.

It is easy to verify that, considering $n(l,k)$ upper-bounded, $p_l(k)$ has maximum value

$$p_{max} = {}^{\gamma\, n_{max}}\!/_{1-\beta}$$

and that the error due to initial condition decreases asymptotically according to β.

B. *Stimulus functions*

A stimulus function is the way to compute the current level of stimulus for an intersection to execute a policy , with respect to pheromone levels: the more desirable the policy, the higher the stimulus. A general stimulus function is defined as

$$s_{i,j}\colon [0, p_{max}]x[0, p_{max}] \to \mathbb{R}^+$$

where $s_{i,j}$ is the stimulus to adopt the policy j in the intersection i. The function has a limited domain because pheromone levels are bounded and has a limited co-domain because of normalization.

It is mandatory that every stimulus function is normalized over the domain, such that

$$\iint_{[0,p_{max}]x[0,p_{max}]} s_{i,j}(p_{in}, p_{out})\, dp_{in} dp_{out} = 1$$

This guarantees that more specialized policies dominate less-specialized ones in the neighbourhood of their maximum effectiveness.

C. *Choosing a policy*

Once a stimulus is computed, a policy is chosen with a non deterministic procedure. As an ant decides its moves driven by probability, an intersection chooses the policy. The probability of choosing a policy j in the intersection i is given by the fraction

$$P(i,j) = \frac{T_{\theta_{i,j}}(s_{i,j})}{\sum_j T_{\theta_{i,j}}(s_{i,j})}, \text{ where } T_{\theta_{i,j}}(s_{i,j}) = \frac{s_{i,j}^2}{s_{i,j}^2 + \theta_{i,j}}$$

$\theta_{i,j}$ is the response threshold for the intersection i to execute the plan j. Following the model from [The97], an intersection continues to perform a previously chosen policy, and, after a fixed amount of time, decides to inspect and possibly choose a new policy only with probability p_{change}, whatever the stimulus.

D. Response thresholds reinforcement

Thresholds are dynamic and belong to

$$[\theta_{min}\ \theta_{max}];\ 0 \le \theta_{min} < \theta_{max} \le 1.$$

Thresholds give the capability to learn from the past and specialize the way to react to traffic dynamics. Every intersection has its own set of thresholds, one for each policy, and updates them at run-time. When executing a policy j, the corresponding threshold is lowered:

$$\theta_{i,j} = \theta_{i,j} - \xi\Delta t$$

ξ is the learning coefficient, Δt is the fixed amount of time between two decisions. Instead, when not executing a policy j, the corresponding threshold is increased:

$$\theta_{i,j} = \theta_{i,j} + \rho\Delta t$$

where ρ is the forgetting coefficient.

In the model describing insect societies [The97] thresholds reinforcement has the purpose to model the specialization of insects in performing a task. In this paper an intersection does not specialize in executing a single policy, but can learn and consider some policies as preferable, being more reactive in executing them, but without losing the possibility to choose and execute a less common policy, if it is the case to use it.

IV. Simulation and Benchmarks

From a software engineering point of view, the controller is modelled and implemented to be easily put into production, for example using a common single-threaded Programmable Logic Controller.

The control system is implemented into two simulation environments: the NetLogo environment [Wil99], and the SUMO environment [Kra02].

In NetLogo the "Gridlock model" [Wil03] is extended, simulating simple networks to perform in-line cognitive experiments. SUMO is extended to realize our traffic control system and assess performance on scenarios closer to the real world: this traffic micro-simulator is able to load complex road networks and perform more realistic simulations. Benchmarks are repeated over different topologies of non-homogeneous networks, having up to 64 intersections, and over different kind of traffic load; every test is repeated over several road topologies chosen randomly and average results are taken into account to avoid any biasing of results.

Performance assessment is performed comparing our control system against a static green-wave approach, a reactive one [Fou04] and single policies proposed in [Ger07]. Performance metrics are the average speed of vehicles, the percentage of vehicles stopped at intersections, average waiting time when stopped at an intersection, and the time needed to solve a local congestion, due to a sudden burst of traffic (referred in tables as TT-640-10, i.e. the time to have only 10 vehicles inside a district, starting from a local congestion of 640 vehicles willing to leave). Performances are tested over five different traffic demands, from low to congested, plus the case of the local burst of traffic to be managed.

Simulations showed our system performs well, as it inherits all the advantages from single policies and avoids disadvantages. It is interesting that for average traffic the system performs even better than any single policy from [Ger07], because a good mix of them is spontaneously adopted, according to local traffic needs.

The control system global adaptability is analyzed: traffic changes are sensed and handled with a small delay, because of pheromone dynamics and the effect of parameter p_{change}; this is a desirable property because the system shall absorb small burst without changing policies; in typical real scenarios traffic has not sudden but smooth changes and the system behaves very well to these kinds of variations.

It is clear results presented here are meant not to be definitive, because a more appropriate tuning of parameters could lead to even better performances. Table I shows the average improvement or worsening percentage of our control system compared to a static optimization based on green waves, while Table II is a comparison with a reactive control [Fou04] using a fixed policy (there are improvements when the average speed increases, or when the number of waiting cars decreases, as well as the average waiting time decreases; an improvement is shown also by a decreased value of the TT-640-10 metric).

TABLE I
OUR SYSTEM VERSUS STATIC OPTIMIZATION

Traffic Demand	Average Speed	Stopped %	Average Waiting Time
Low	*15,9%*	*-42,1%*	*-76,4%*
Average	*6,4%*	*-28,6%*	*-65,0%*
Heavy	*10,0%*	*-9,5%*	*-37,1%*
Congested	*21,6%*	*2,1%*	*22,5%*
Deadlock	*48,7%*	*1,5%*	*218,2%*
TT-640-10	-32,9%		

Values are truncated for convenience.

TABLE II
OUR SYSTEM VERSUS REACTIVE CONTROL

Traffic Demand	Average Speed	Stopped %	Average Waiting Time
Low	*6,8%*	*-13,3%*	*-34,6%*
Average	*3,9%*	*-4,1%*	*8,1%*
Heavy	*240,4%*	*-59,0%*	*-83,3%*
Congested	*873,0%*	*-45,9%*	*-69,7%*
Deadlock	*45,7%*	*9,0%*	*218,6%*
TT-640-10	-22,9%		

Values are truncated for convenience.

Our solution shows notable improvements in average speeds for every traffic condition, thus it achieves a higher network throughput. The stopped percentage and waiting time of vehicles at intersections decrease for every condition, except when traffic is heavy or congested: this is expected as the will is to offer the biggest throughput possible, at the cost of some unfairness to road network users. Thanks to this approach, our control system performs very well to resolve local congestions, as the TT-640-10 metric shows a strong improvement over both concurrent solutions, in resolving local congestions.

V. CONCLUSION AND FURTHER IMPROVEMENTS

A. Discussion

This paper presented a distributed control system in which every intersection controller makes independent decisions to pursue common goals and is able to improve global traffic performance. This solution is low cost and widely applicable to different urban scenarios.

It is reactive and adaptable, as every action is decided with respect to sensors, thus there is no risk of wasted green time at intersections. It is self-organized, leading to the spontaneous emergence of complex behaviours, like the dynamic creation of traffic corridors similar to green waves, load balancing over routes of heavy traffic, effective handling of local congestions due to peek demands (e.g., end of sport events), smoothing of shock waves. In addition, the control system is able to adapt spontaneously to traffic rerouting due to road works, protests, accidents, or permanent changes in road topology as the introduction of roundabouts in place of a signaled intersection.

Unlike both centralized and distributed control systems requiring communication, ours scales without any effort from a single intersection to a metropolis. The self-organizing pattern enables each agent to be totally unaware of their role in the whole. Therefore, self-organization positively impacts the extensibility. Smart intersections might be added in a plug-and play fashion when new budget is available without affecting or reconfiguring the control system already in use. Public transports, pedestrians and emergency vehicles could be supported easily with a pre-emption mechanism that gives priority to these classes of road users.

From an economical point of view, our system has low cost for installation, maintenance and reconfiguration, as every controller and its sensors are technically an independent system. Our solution is meant to be deployable using present technologies and to be open for future cheaper ones.

In the future traffic will be more and more dynamic, thanks to information sharing systems that will make road users able to prevent congestions, preferring alternate routes. Our solution is developed to handle present and future traffic scenarios, thanks to the adaptability of self-organizing systems.

B. Parameters Tuning

Presenting the model, many parameters are introduced. They affect every aspect of the control system: β and γ regulate the sensitivity to changes in densities, θ_{min} and θ_{max} how much thresholds can be reinforced, ξ and ρ how fast this reinforcement is. Besides, every single policy has its own parameters needing to be correctly tuned, as well as properly-designed stimulus functions are fundamental for a correct policy selection.

In this paper, parameters regarding dynamic policy selection are tuned using knowledge coming from control systems theory, experiments and suggestions taken from [The97] (it describes how thresholds and reinforcement affect specialization). Stimulus functions are designed after benchmarking in isolation every single policy, observing for what traffic density the policy performs best and how large is its range of effectiveness.

Our control system, designed to be reactive and spontaneously adaptive, depends on experiments and human intuition to be tuned, by now. This is a weakness, as emergent systems are inherently difficult to be predicted and developed with traditional engineering techniques. As a further improvement, it would be highly desirable to implement some kind of mechanism to tune the system and strengthen its performance automatically, exploiting recent achievements in the field of parameter tuning.

REFERENCES

[Baz09] A. L. C. Bazzan, "Opportunities for multiagent systems and multiagent reinforcement learning in traffic control", in Autonomous Agents and Multi-Agents Systems Volume 18 Number 3, June 2009, pp. 342–375.

[Bus08] L. Busoniu, R. Babuska, B. De Schutter, "A Comprehensive Survey of Multiagent Reinforcement Learning", in IEEE Transactions on systems, man and cybernetics Part C: Applications and reviews Volume 38, Number 2, March 2008, pp. 156–172.

[Fou04] M. E. Fouladvand, Z. Sadjadi, M. R. Shaebani, "Optimized Traffic Flow at a single intersection: Traffic responsive signalization", 2004.

[Ger07] Carlos Gershenson, "Self-Organizing Traffic Lights", 2007.

[Gol99] J. Goldstein, "Emergence as a construct: History and issues" in Emergence Volume 1, Issue 1, March 1999, pp. 49–72.

[Gra59] P. P. Grassé, "La théorie de la Stigmergie: essai d'interpretationdu Comportement des Termites Constructeurs", 1959.

[Kra02] D. Krajzewicz, G. Hertkorn, C. Rössel, P. Wagner, "SUMO (Simulation of Urban MObility); An open-source traffic simulation", 2002.

[Mil10] V. Milanés, J. Pérez, E. Onieva, and C. González, "Controller for Urban Intersections Based on Wireless Communications and Fuzzy Logic", in IEEE Transactions on intelligent transportation systems, Vol. 11, No. 1, 2010.

[Miz08] K. Mizuno, Y. Fukui, S. Nishihara, "Urban Traffic Signal Control Based on Distributed Constraint Satisfaction", in Proceedings of the 41st Hawaii International Conference on System Sciences, 2008.

[Oli04] D. de Oliveria, P. R. Ferreira Jr., A. L. C. Bazzan, F. Klügl, "Reducing traffic jams with a swarm-based approach for selection of signal plans", 2004.

[Pee02] Peek Traffic Scandinavia, "UTOPIA/SPOT-

Technical Reference Manual", 2002.

[Pri09] C. Priemer, B. Friedrich, "A Decentralized Adaptive Traffic Signal Control Using V2I Communication Data", in Proceedings of the 12th International IEEE Conference on Intelligent Transportation Systems, 2009.

[Rob69] D.I. Robertson, "TRANSYT: a network study tool", 1969.

[Rob90] D.I. Robertson, R.D. Bretherton, "Optimizing Networks of Traffic Signals in Real Time – The SCOOT Method", 1990.

[Sim79] A.G. Sims, K. W. Dobinson, "the Sydney Coordinated Adaptive Traffic system", 1979.

[The97] G. Theraulaz, E. Bonabeau, J. L. Deneubourg "Response threshold reinforcement and division of labour in inset societies", 1997.

[Wil99] U. Wilensky, "NetLogo", 1999.

[Wil03] U. Wilensky, W. Stroup, "NetLogo HubNet Gridlock model", 2003.