Auton Agent Multi-Agent Syst (2009) 18:342–375
DOI 10.1007/s10458-008-9062-9

Opportunities for multiagent systems and multiagent reinforcement learning in traffic control

Ana L. C. Bazzan

Published online: 7 September 2008

Abstract The increasing demand for mobility in our society poses various challenges to traffic engineering, computer science in general, and artificial intelligence and multiagent systems in particular. As it is often the case, it is not possible to provide additional capacity, so that a more efficient use of the available transportation infrastructure is necessary. This relates closely to multiagent systems as many problems in traffic management and control are inherently distributed. Also, many actors in a transportation system fit very well the concept of autonomous agents: the driver, the pedestrian, the traffic expert; in some cases, also the intersection and the traffic signal controller can be regarded as an autonomous agent. However, the "agentification" of a transportation system is associated with some challenging issues: the number of agents is high, typically agents are highly adaptive, they react to changes in the environment at individual level but cause an unpredictable collective pattern, and act in a highly coupled environment. Therefore, this domain poses many challenges for standard techniques from multiagent systems such as coordination and learning. This paper has two main objectives: (i) to present problems, methods, approaches and practices in traffic engineering (especially regarding traffic signal control); and (ii) to highlight open problems and challenges so that future research in multiagent systems can address them.

Keywords Multiagent systems · Multiagent learning · Reinforcement learning · Coordination of agents · Game-theory · Traffic signal control

1 Introduction

The second half of the last century has witnessed the beginning of the phenomenon of traffic congestion. This arose due to the fact that the demand for mobility in our society has increased constantly. Traffic congestion is a phenomenon caused by too many vehicles trying to use the same infrastructure at the same time. The consequences are well-known: air pollution,

A. L. C. Bazzan (✉)
Instituto de Informática, Universidade Federal do Rio Grande do Sul, CP 15064,
91501-970 Porto Alegre, RS, Brazil
e-mail: bazzan@inf.ufrgs.br

decrease in speed, delays, and unsatisfaction of drivers. The latter may lead to risk maneuvers thus reducing safety for pedestrians as well as for other drivers.

The increase in transportation demand can be met by providing additional capacity. However, this might no longer be economically or socially attainable or feasible. Thus, the emphasis has shifted to improving the existing infrastructure without increasing the overall nominal capacity, by means of an optimal utilization of the available capacity. Two complementary measures can be taken: coupling management systems with telecommunication and information technology, and improving the management via control techniques. This set of measures is framed as Intelligent Transportation Systems (ITS).

This paper focus on the second (control). However, we note here that Advanced Traveler Information Systems are becoming increasing popular as they aim at controlling traffic by broadcasting information to road users via radio, variable message signals installed on the road, internet, mobile phones, on-board systems, and so on.

Management, control, and optimization of traffic have received the attention of researchers outside the area of traffic engineering. In fact, computer scientists, physicists, and mathematicians have proposed several different approaches to the problem. It is therefore an interesting question, why these approaches do not raise the interest of practitioners in traffic engineering, not to speak of deployment. In this paper it is claimed that researchers outside the traffic engineering community tend to approach the problem in a naive way only to find out that, in order to apply their approaches, they have to make simplifying assumptions that render the "solution" uninteresting from the point of view of engineering. In reality, not even the "simplest" problem, namely the optimization of traffic in a single intersection, is solved to a satisfactory level. Learning techniques which were developed for multiagent systems, can potentially give decisive contributions to control and management of traffic systems, as they meet the demands of dynamic, changing systems of many heterogeneous actors with different goals, distinct cognitive capabilities and learning pace.

Thus the aim of the present paper is twofold. It is intended to serve as a survey regarding problems, methods, and practices in traffic engineering (especially regarding traffic signal control), as well as on approaches coming from other fields (computer science, physics, etc.), while also stating the basic and open problems and challenges so that future research can be directed to them, in order to reduce the gap between theory and practice. Second, it focuses on multiagent systems (MAS) and artificial intelligence (AI) aspects such as learning (and its complexity), presenting both past proposed solutions as well as a discussion about issues that have to be improved in order to increase the practitioners' acceptability of such solutions.

It is also shown that there are several open questions regarding modeling, simulation, management, and control of traffic systems. Thus there are many opportunities for using multiagent systems methods and techniques, and in fact the problems posed are challenging and now ripe for non trivial, non naive approaches, especially as to what regards learning. Therefore, approaches proposed by computer scientist in general and from the artificial intelligence community in particular may achieve the level of deployment among the transportation engineering community, which is, understandably, highly concerned with operational and security issues.

In order to keep the focus and be able to present and discuss issues involved in a detailed level, this paper concentrates on traffic control and, to a lesser extent, simulation issues. It does *not* deal with logistics and freight, sea and air transportation, air traffic control, public transportation, and pedestrian and crowds simulation. Also, only some references are given here for AI and MAS based approaches to traffic management systems and to route guidance systems (Sect. 3.1). However, for completeness, Sect. 3 provides an overview to traffic engineering, including references for readers interested in investigating details of sub-fields not

covered here. Before, Sect. 2 discusses the current state-of-the art in multiagent reinforcement learning (MARL). Sections 4–6 describe classical approaches to the particular case of control and optimization of traffic via traffic signals (mostly deployed), and new approaches arising from AI, multiagent systems, learning-based, as well as those proposed by physicists and by the operations research community. In Sect. 7 a classification of the methods discussed before is presented. Section 8 returns to the second goal of this paper which is to focus on MARL. First the goals and challenges of MARL regarding control of traffic signals are summarized. Then, it is shown that this problem is an excellent testbed for MARL as the inherent dimensionality and complexity of these scenarios render the most popular MARL approaches proposed so far unsuitable given computational performance issues. Also the application and challenges of one popular approach to MARL, stochastic games, is discussed and it is shown that existing approaches cannot cope with the dimension of the problem of learning for controlling a network of traffic signals. Thus, new approaches are necessary, possibly based on heuristics and approximate solutions to partially observed Markov decision processes, as well as mixed approaches involving evolutionary and reinforcement learning techniques. Concluding remarks appear in Sect. 9.

2 Multiagent learning: questions and answers

2.1 Single agent reinforcement learning

Usually, single agent Reinforcement Learning (RL) problems are modeled as Markov Decision Processes (MDPs). These are described by a set of states, $\mathcal{S}$, a set of actions, $\mathcal{A}$, a reward function $R(s, a) \rightarrow \Re$ and a probabilistic state transition function $T(s, a, s') \rightarrow [0, 1]$. An experience tuple $\langle s, a, s', r\rangle$ denotes the fact that the agent was in state s, performed action a and ended up in s' with reward r.

Reinforcement learning methods can be divided into two categories: model-free and model-based. Model-based methods assume that the transition function T and the reward function R are available. Model-free systems, such as Q-learning, on the other hand, do not require that agents have access to information on how the environment works. Q-Learning works by estimating state–action values, the Q-values, which are numerical estimators of quality for a given pair of state and action. More precisely, a Q-value $Q(s, a)$ represents the maximum discounted sum of future rewards an agent can expect to receive if it starts in s, chooses action a and then continues to follow an optimal policy. The Q-Learning algorithm approximates $Q(s, a)$ as the agent acts in a given environment. The update rule for each experience tuple $\langle s, a, s', r\rangle$ is:

$$Q(s, a) \leftarrow Q(s, a) + \alpha(r + \gamma \max_{a'} Q(s', a') - Q(s, a)) \tag{1}$$

where α is the learning rate and γ is the discount for future rewards. If all pairs state-action are visited infinitely often, then Q-learning is guaranteed to converge to the correct Q-values with probability one [91].

In scenarios where exploration is too costly, it may make sense to either have a model of the environment or profit from transfer of knowledge. This aims at enhancing learning using knowledge already acquired (e.g. when performing one task) in another, related, target task (see [74] for an introduction, as well as references therein).

Knowing the states to which each action will take the agent, it may select the one with higher utility. Each time the agent acts, it observes the environment and updates its transition and reward models. This is the basis of model-based algorithms for reinforcement learning such as Dyna [76] and Prioritized Sweeping [49].

2.2 Multiagent reinforcement learning: stochastic games

Learning in systems with two or more players has a long history in game-theory. Specifically, the connection between multiagent systems and game-theory as to what regards learning has been explored at least from the 1990s onwards. Thus, it seems natural to the reinforcement learning community to explore the existing formalisms behind stochastic (Markov) games (SG) as an extension of Markov decision processes (MDP). Despite the inspiring results achieved so far, it is not clear whether this formalism is completely suitable for multiagent learning [69,70,75]. Besides, as discussed later in Sect. 2.3, SG is not the only approach available [75]. Problems posed by many agents in multi-agent reinforcement learning are inherently more complex than those regarding single-agent reinforcement learning (SARL). This complexity has many consequences.

First, approaches proposed for the case of general sum SG require that several assumptions be made regarding the game structure (agents' knowledge, self-play etc.). These assumptions restrict the convergence results to common payoff games and other special cases such as zero-sum games, besides focusing on two-agent stage games. Otherwise, an oracle is needed if one wants to deal with the problem of equilibrium selection when two or more equilibria exist.

Second, despite recent results on formalizing MARL using SG, these cannot be used for systems of more than a few agents, *if any flavor of joint-action is explicitly considered*, unless the obligation of visiting all pairs of state-action is relaxed, which has impacts on the convergence. The problem with having a high number of agents happens mainly due to the exponential increase in the space of *joint* actions. In fact, most of the literature concentrates on repeated games with two-players and a single state. However, this modeling is not suitable for interactions in traffic systems.

Third, while agents themselves must not be cooperative, we may be interested in improving the system's performance. This is a well-known issue. Tumer and Wolpert [80] for instance have shown that there is no general approach to deal with the complex question of collectives.

Up to now, these issues have prevented the use of MARL in real-world, large-scale problems, unless simplifications are made, such as letting each agent learn *individually* using single-agent based approaches. As it is known, this approach is not effective, since agents converge to sub-optimal states.

2.2.1 Stochastic games: formal setting

The generalization of a MDP for n agents is an SG, represented by the tuple (N, S, A, R, T) where:

$N = 1, \ldots, i, \ldots, n$ is the set of agents
$S = \times S^j$, $1 \leq j \leq m$ is the discrete state space (set of n-agent stage games)
$A = \times_{i \in N} A^i$ is the discrete action space (set of joint actions)
R is the reward function (R determines the payoff for agent i as $r^i \colon S^1 \times \cdots \times S^m \times A^1 \times \cdots \times A^n \rightarrow \Re$)
T is the transition probability map (set of probability distributions over the state space S).

2.2.2 Stochastic game based approaches

This section reviews some approaches to MARL without any pretension of being comprehensive. In fact, this is impossible nowadays given the number of approaches suggested. As many of them deal with particular issues, we will rather concentrate on seminal works,

on more general approaches, and especially on those which will be later discussed in the context of traffic signals control. We will show that most of the approaches are not completely appropriated for this context.

Most of the research on SG so far is based on a static, single state stage game (i.e. a repeated game) with common payoff (payoff is the same for agent and opponent). In single state SG, since there is no state transition, one cannot speak of a dynamic game in strict sense.

One of the common formulations is that of n-player, single state team (common payoff) game. Here, each $i \in N$ has a finite set of actions A_i and repeatedly play a single stage game. To denote joint actions we use $A = \times_{i \in N} A_i$; each joint action $a \in A$ is associated with a reward $R(a) \in \Re$ and because it is a team game, payoffs are the same for all agents. In fact, this is normally a two player game played pairwise by $|N|$ agents.

Claus and Boutilier [22] discuss some factors that influence the dynamics of the learning process in a coordination game where learning is performed by means of a Q-learning-like mechanism. In a first setting this is done independently by each player; in a second setting, players keep beliefs about strategies of other agents, a kind of opponent modeling. They call this second setting "joint action learners". In their experiments they concluded that independent learners converge quickly but not necessarily to the same equilibrium. Joint action learners did not perform much better.

The zero-sum (ZSG) case of the stage game is discussed by Littman in [46]. In a two-player ZSG, one agent's payoff is the opposite of the other agent. One agent minimizes over others' action and then selects its own action so as to maximize its own payoff.

For the general sum game with two players, as the minimax Q-learning cannot be used, Hu and Wellman [39] propose that both agents execute actions (a^1, a^2) in state s and follow their Nash equilibrium strategies (π_1, π_2) thereafter. This of course assumes that information is perfect, i.e. one agent can observe other agents' actions and rewards. For a more comprehensive description of general approaches, we refer the reader to [58,70] and references therein. Here we focus on particular approaches that could be extended for control of traffic signal purposes. Verbeeck et al. [85] tackle a particular kind of game (coordination game) by means of an exploration technique based on learning automata and reduction of the action space. The approach of Vu et al. [86] deals with multiple opponents but they assume that the full game structure and payoffs are known to all agents. Besides, the algorithm is based on joint strategy for all self-play agents (those who learn using the same algorithm) so that the action space is exponential in the number of self-play agents. Specifically for traffic, a simple stage game is presented in [5], while Camponogara and Kraus [18] have studied a simple scenario with two intersections, using stochastic game-theory and reinforcement learning. Both are discussed in more details in Sect. 5.

As seen, stochastic games have been used successfully to model multiagent encounters which fit the somehow limited frame of stage games. There has been a discussion among several researchers whether or not this is the right avenue for multiagent learning in general [69,70,75]. Shoham et al. single out some problems due to focusing on what they call the "Bellman heritage".

Related to this discussion, two issues discussed in [69] are important from the perspective of traffic control. The first is the focus on convergence to equilibrium regarding the stage game: "If the process [of playing a game] does not converge to equilibrium play, should we be disturbed?" Also, most of the research so far has focused on the play to which agents converge, not on the payoff agents obtain. The second issue is that "In a multi-agent setting, one cannot separate learning from teaching" because agent i's action selections both arise from information about agent j's past behavior, as well as impact j's future actions' selections. Unless i and j are completely unaware of the presence of each other, both can teach and

learn how to play in mutual benefit. Therefore it is suggested that a more neutral term would be *multi-agent adaptation* (rather than learning). This is an important point because it agrees with a view that some issues related to operational control are more a quest of adaptation than of optimization (see Sect. 6.2). Since the latter is hard to achieve within a short time frame, it is often the case that this cannot be done in real-time. One more point in favor of adaptation is that many works on MARL have been assuming static environments. In this kind of environment it may make sense to evaluate MARL algorithms by the criteria proposed in [13], namely convergence to a stationary policy, and convergence to a best response if the opponent converges to a stationary policy. Although other criteria are being proposed (see [86]), it certainly makes little sense to evaluate a learning or adaptation algorithm by such criteria when the environment is itself dynamic, as it is the case of the traffic scenario discussed here. This issue is further detailed in Sect. 8.

2.3 Beyond stochastic games

The previous section has reviewed selected relevant multiagent learning research based on paradigms of game-theory. However there are other works where: (i) convergence to an equilibrium is not a goal in and of itself, and (ii) a game theoretic formulation yields little progress towards a solution. Stone [75] mentions RoboCup Soccer as an example where there are more than one opponent (11), while each agent has several teammates (10). Here clearly both non cooperative and cooperative game-theory could be involved. Besides, decisions by players are made continuously, are based on incomplete information, they must be made in highly stochastic environments, and have strong sequential dependencies. So posed, the scale of this problem completely discourages a game theoretic formulation. Even when such a formulation proved successful, it was sometimes based on abstraction of complex multiagent interactions to game-theory terms.

In traffic engineering there has been some successful models based on SG (Sect. 5). However two major issues play an important role. First, a high level of abstraction is necessary in order to avoid the combinatorial explosion mentioned. Second, as in RoboCup Soccer (and possibly any other kind of n-agent encounter where $n > 2$), there is the issue of local versus global optimum. In the example of the player learning "how to pass [the ball] and where to pass in the presence of specific adversaries" [75], there are many possible formulations regarding the reward of this player. Either it gets a (possibly artificial) reward for a good pass, which would be its local reward, or it is rewarded later when and if the team scores (team reward). In real life soccer, it is frequently the case that player A makes a wonderful pass to player B who then misses the chance to score. How shall A be rewarded? And B?

Similarly, in traffic a local control decision by traffic signal A may lead to the best performance at intersection level but clogs intersection B downstream. Which reward structure shall be considered? Drivers receiving green indication at A, who are not bound to B, are certainly satisfied as their delays were minimized. Those bound to B would have to stop and wait anyway at B. The agent at intersection B is definitively not happy to detect an increasing flow of incoming vehicles. Rewards at global level (whatever global here means) are even more complex. Is average travel time a fair measure? Why do traffic signal agents have to be cooperative?

In this paper it is claimed that traffic control scenarios discussed in the context of learning (Sect. 5) can indeed be framed as a modified SG if assumptions about visiting infinitely often all pairs of state-action are relaxed, and if the focus is not put on convergence to an equilibrium. This way, some theoretical basis of SG as well as its nice formalization remain, while more complex scenarios can be tackled.

Also [83] discusses Shoham and et al. agenda, this time from the perspective of evolutionary game-theory (EGT). They consider what each of the agenda's point means in term of EGT. In particular, regarding the issue of normative and descriptive agendas, EGT represents a shift in game-theory as to what regards moving away from classical solution concepts—Nash equilibrium is meant—towards an evolutionary stable strategy. In fact, this shift fits very well multiagent learning in general and learning in dynamic scenarios (such as traffic control) as already detected in [4,5,81,82].

Despite the achievements in single-agent reinforcement learning, in model-based approaches, in SG, and in other multiagent learning techniques, most of these cannot be readily employed in traffic control without significant simplifications. Model-free techniques cannot cope with dynamically changing environments in a fast way, while SG based approaches cannot cope with the explosion in the number of states. These issues are discussed in details in Sect. 5 where examples of use of these and other techniques are discussed. In Sect. 8 the open challenges are discussed.

3 Conceptual and organizational aspects in traffic engineering

Modern transportation networks are becoming more and more congested, especially in urban areas. To turn the problem even worse, this congestion is not evenly spatially distributed. If we abstract the problem of distribution of traffic for a moment, we may consider it just from the perspective of a supply–demand problem. Here, two concepts are important to understand traffic in transportation systems. The demand between places must either be known (e.g. via origin-destination (OD) matrices) or estimated; see [56] for details. Similarly, the supply within the network must either be known or estimated for all possible routes between points in the network that generate or attract trips.

However, per se, those estimates say little about *how* traffic is distributed in the network. The *assignment problem* deals with the distribution of traffic in a network considering demands between several locations, and the supply and capacity in that network. Assignment methods must consider not only the distribution of traffic in a network, but also a set of constraints related to cost, time, and preferences of road users. Classically, this is done via network analysis. To this aim, it is assumed that individual road users seek to optimize their individual costs regarding trips they make by selecting the "best" route. This is the underlying idea behind traffic network analysis based on Wardrop's equilibrium [90].

An example of traffic assignment is related to the following classical commuting scenario: several commuters want to go from location A to location B around some specific time of the day. The network offers a set of route (path) choices. A typical commuter will then select the one with the least time, although other criteria can be used. Given that thousands of commuters make these decisions every day, the assignment of commuters to routes becomes a highly complex task, especially due to the fact that commuters are likely to adjust their decisions to their past experience and to information they may be able to gather. On the other hand, transportation authorities also collect information about the state of the network in order to use them to adjust their transport supply. Unfortunately, given topological constraints, it is not possible to change the supply in a way flexible enough that matches the demand entirely. Therefore transportation authorities must employ several kinds of traffic management systems, involving both information broadcast as well as control and optimization.

AI and multi-agent techniques have been used in many stages of these processes. These approaches can be classified into three levels: integration of heterogeneous traffic management systems, traffic flow control, and traffic guidance. The first of these levels is

discussed in several papers, e.g. the platform called Multi-Agent Environment for Constructing Cooperative Applications—MECCA/UTS—[36], as well as in [57,67,84]. The other two levels are discussed next.

3.1 Traffic management via information broadcasting for route choice

Although travelers information systems are not the focus of this paper, for the sake of directing the reader, basic concepts and some references regarding the use of AI and MAS techniques are given. Further references can be found in [35,66].

It is generally believed that information-based ITS strategies are among the most cost-effective investments that a transportation agency can make. These strategies, also called Advanced Traveler Information Systems (ATIS), include highway information, broadcast via radio, variable message signs (VMS), telephone information services, web/internet sites, kiosks with traveler information, and personal data assistant and in-vehicle devices. Many other new technologies are available now to assist people with their travel decisions.

Multi-agent techniques have been used for modeling and simulation of effects of the use of these technologies, as well as the modeling of behavioral aspects of drivers and reaction to information. Details can be found in [2,9,12,17,30,41,42,61,62,68,79,87].

3.2 Traffic management via traffic control

Several strategies of traffic control exist, with most of them fitting the control loop described by Papageorgiou [59,60]. The basic elements of this control loop are: the physical network, its model, the model of demand and disturbances (can be measured, detected or forecasted); control devices (traffic signals, variable message signs, etc.); surveillance devices (e.g. loop detectors); and the control strategy. Some of these are computational entities: models, surveillance and control strategies. In this paper the focus is exactly on the latter. As mentioned, several strategies will be discussed emphasizing those based on learning.

This control loop applies to any kind of traffic network if one is able to measure traffic as the number of vehicles passing on a link in a given period of time. Techniques and methods from control theory are applied in traffic control in a fine grained way. This leads to the problem that those techniques can be applied only to single intersections or to a small number of them. For arbitrarily big networks, if no simplifications are made, the real-time solution of the control loop faces a number of apparently insurmountable difficulties [59].

In spite of this, with the current developments in communication and hardware, computer-based control is now a reality, especially as to what regards the control of traffic. These are also known as advanced transportation management systems (ATMS), whose main goals are: to maximize the overall capacity of the network; to maximize capacity of critical routes and intersections which represent bottlenecks; to minimize negative impacts of traffic on the environment and on energy consumption; to minimize travel times; and to increase traffic safety. In order to achieve these goals, one of the possible measures is to use devices to control the flow of vehicles (e.g. traffic signals). Traffic signals can vary from hard-wired logic to computerized control, either centralized or not.

Because of the characteristics of transportation networks, there are two major types of traffic flows: uninterrupted traffic (regulated by vehicle-vehicle interactions and interactions between vehicles and the transport infrastructure) such as a highway; and interrupted traffic (regulated by devices such as a traffic signal) which leads to queue formation.

Next, we focus on control of interrupted traffic by means of traffic signals.

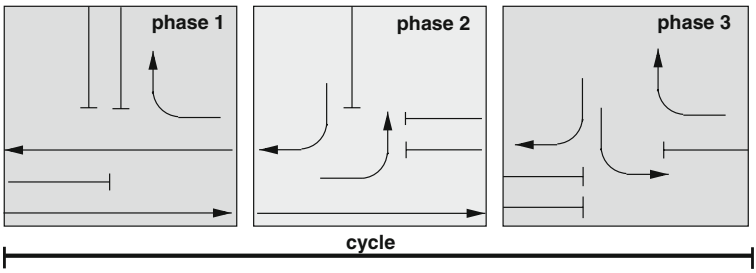

Fig. 1 Signal plan with three phases

3.3 Traffic signal controllers

Signalized intersections[1] are operated by traffic signal controllers that combine hardware and software in order to implement the signal timing. According to [66], the most fundamental unit in signal design and timing is the cycle. This is defined as a complete rotation through all the green indications. In general, all legal vehicular movements (as for example in Fig. 1) receive a green indication during each cycle. Intervals are periods of time during which no signal indication changes. The most common intervals are: the change interval (yellow indication), the clearance interval (all movements receive red indication), and the green and red intervals. Besides the cycle and the intervals, another component of the signal timing is the signal phase. It consists of a green interval plus the change and clearance intervals.

Traffic signals can be operated in a variety of modes, classified according to the following three main dimensions: fixed-time (pre-timed) basis versus traffic responsive (actuated) basis; isolated intersection control versus coordinated control; locally controlled by a simple microprocessor versus remote, computerized control.

In pre-timed operation, the cycle length, the phase sequence, and the timing of each interval are constant and follow a predefined plan designed to deal with a traffic volume that is computed based on historical data. In semi-actuated operation it is also necessary to acquire data from buried detectors (e.g. of loop-induced type) or other devices. These detectors are only placed in minor approaches to the intersection (no detector in the main street). In full-actuated operation, every lane of every approach has a detector and green time is allocated in accordance to specific rules, so that the cycle length, the sequence of phases and the green time split may vary from cycle to cycle. A fixed-time controller is the more affordable and logical choice for networks with stable or predictable traffic behavior. However, this kind of controller cannot cope with unexpected changes in traffic flow.

In a computer controlled system the computer acts as a master, coordinating timings of the signals. This master selects or calculates an optimal coordinated plan either based on inputs from detectors or on a time-of-the-day basis. For coordination to be effective, all signals must use the same cycle (or multiples). Thus it is difficult to maintain this coordination if cycle lengths or phase splits are allowed to vary. Several plans are normally required for an intersection (or set of intersections in the case of a coordinated system) to deal with changes in traffic flow.

[1] The terms intersections, crossing, junction, traffic signal, and traffic light are used interchangeably since in each intersections, *only one signal-timing plan* runs in a set of traffic lights so that the set of traffic lights that provide the actual indications must be seen as a single entity.

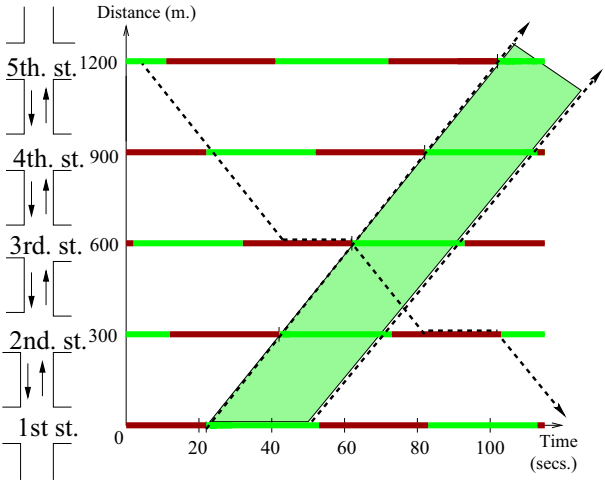

Fig. 2 Time-space diagram of a progression in an arterial

3.4 Coordinated systems

3.4.1 Synchronization in arterials: basics

The goal of coordinated systems (also called synchronized or progressive systems) is *to synchronize traffic signals along an arterial* in order to allow platoon of vehicles, traveling at a given speed, to cross the arterial without stopping at red lights. Thus, coordination here means that if appropriate signal plans are selected to run at adjacent traffic signals, a "green wave" is built.

Apart from the parameters discussed in Sect. 3.3, a coordinated system also needs the so-called *offset* (time between the beginning of the green phase of two consecutive traffic signals) that is computed based on the desired speed and on the distance between intersections. Another important concept is the bandwidth. It is the time difference between the first and the last vehicle that can pass through without stopping. Figure 2 shows a space-time diagram of the synchronized or progressive system in an arterial. In order to keep the example simple, only a few intersections are shown. Also, this progression is a particular, simple, case in which the bandwidth is designed to use all the green time in one direction (assume that this direction has a much higher traffic volume).[2] Thus one may expect the other direction to suffer. In fact, the progression is efficient for vehicles traveling from the First to the Fifth Street. One bandwidth is shown: vehicles entering the arterial in the intersection with the First Street from time 22 to 54 will be able to pass the whole arterial without stopping.

The classical problem concerning synchronization systems is to find the optimal bandwidth for different cycle times and speeds. Popular solutions use hill climbing [63,78] and mixed-integer linear programming [50].

Well designed synchronized signal plans can achieve acceptable results in undersaturated arterial, in one flow direction. However the bandwidth of the progression decreases in more constrained problems. For example when the progression is to be set in two directions of an arterial, the bandwidth generally decreases. If, additionally, the progression is to be calculated for a network of arterials that cross themselves in a more complex way, the bandwidth

[2] Readers can find a discussion on how to design green waves in more complicated cases in [66].

is further reduced. The difficulty is that the geometry of the arterial is fixed and with it the spacing between adjacent intersections.

If one wants to have a long bandwidth in one traffic direction (e.g. bandwidth is equal to the green time), this may have consequences in other directions. Figure 2 shows this. From the perspective of a southbound vehicle, it is possible to see that this vehicle is fortunate and does not stop at the Fourth st. as it arrives there when the progression (which was originally designed for northbound vehicles) just allows green indication at that intersection. However, upon arriving at the Third st., the vehicle has to stop until time around 40, departing at time 60 etc. At the end there is a significant delay, compared to northbound vehicles. As aforementioned, this is an extreme example as it sets the bandwidth equal to the green time for northbound vehicles. As discussed later, there are several algorithms that can optimize more than one bandwidth, thought the fact remains that, the more the constraints (e.g. in the form of more directions of progressions), the smaller the bandwidth, and depending on operation mode (e.g. real time) this problem may become a difficult one, resulting in bandwidths that are too short.

3.4.2 Operation modes

First generation of coordinated systems are based on computation of synchronized signal plans for fixed times of the day such as morning and afternoon peaks. Even if the computation itself is not manual (see next section), this is a complex task that requires a lot of expert knowledge as well as historical data. Once the traffic expert can generate a library of plans, the next task is to decide which one to select for each situation. This is effective only in networks with well-behaved traffic patterns.

In second generation coordinated systems, plans are computed in real time, based on forecasts of traffic conditions using detector data as input to a prediction algorithm. A third generation mentioned in the literature, namely highly responsive control, is based on relaxing of a cycle-based system. It is of course much more difficult to maintain a progressive pattern where cycle length or phase splits are allowed to vary. This can be overcome with queue management at critical intersections, requiring a high number of detectors. Due to all these difficulties, systems of the third generation are not yet completely deployed.

Even more flexible operation modes can be thought such as one that changes the direction of coordination *and the design* of the coordination (who is coordinating with whom in which direction); or fully traffic responsive coordinated systems capable of dealing not only with a main arterial and with a given coordination direction, but also in a grid.

The common reality is that coordinated systems are implemented almost always as fixed-time. This is so because the computer controlled traffic responsive system relies on detectors. These are unaffordable for huge cities in developing countries and difficult to maintain (since buried). Despite this, some successful cases of use of computer-controlled coordinated systems are reported in Washington, DC, Toronto, Sydney, UK, etc.

3.4.3 Classical algorithms and software

In the 1960 and 1970 some algorithms were proposed to analyze traffic patterns and to set traffic signal cycle length, cycle splits, and offset in order to maximize the bandwidth. TRANSYT [63,78] is an off-line optimization tool that generates optimal coordinated plans for fixed-time operation. Inputs are the geometry of the arterial, saturation flows, link travel times, turning rates at each intersection, demands (which are assumed to be constant), a set of

pre-specified timings for the intervals, minimum green duration, and initial values for cycle time, splits and offsets. For given values of the latter three parameters, a model based on platoon dispersion is run, and a performance index is computed based on a combination of delays and total number of stops. The optimization is performed via hill-climbing. Of course the main drawback of this method is that plans are computed for a static situation, based on historical data. It is known that nowadays demand may not be constant, neither between days, nor during the day, nor in a given traffic direction. Besides, the operation of networks close to the saturation flow increases the chances of accidents which, in turn, contribute to unexpected patterns.

SCOOT (*Split Cycle and Offset Optimization Technique*) [40] is similar to TRANSYT but it is traffic-responsive (uses data from detectors located at upstream end of the link).

SCATS (*Sydney Coordinated Adaptive Traffic System*) [47] is also based on real-time data. The main difference to SCOOT is that it is a hierarchical and distributed system. Data collection is local, based on detectors. For control purposes, an area is divided into smaller subsystems (1–10 intersections) that perform the control independently most of the time, i.e. appropriate cycle time and offsets are computed.

Prodyn [38] as well as OPAC [32] and UTOPIA [23] are also adaptive programs in which control is not centralized. These do not consider explicit splits, offsets and cycles. In Prodyn for instance, a decision is taken at each 5 s concerning whether to change phases or not. In a typical case, each intersection simulates all possible situations using detector information in adjacent areas. This information propagates from intersection to intersection with a decreasing weight. Both the relatively complex computation and the communication system can increase the cost of implementation.

The TUC (*Traffic-responsive Urban Traffic Control*) [24] was conceived for large scale networks. The coordination strategy consists of changing split, cycle, or offset (or all), as well as priorization of public transportation. Authors report results in two scenarios (small network, real world) simulating morning peaks. The performance was positive compared to a situation with fixed time synchronization. On the other hand, the computation is centralized and conflits are solved either by a traffic expert or by pre given priority rules, in clear contrast to a multiagent system point of view.

3.4.4 Metrics

The effectiveness of a coordinated system can be measured in various levels of abstraction. As a general measure, one generally seeks to optimize a weighted combination of stops and delays, a measure of the density (vehicles/unit of length) in the arterial or network, or travel time. However, particular characteristics must be observed. Number of stops and delay are acceptable measures for *undersaturated* arterials or networks. Here, queues are generally dissipated. In the oversaturated network, there is an excess of demand relative to the capacity, thus queues tend to expand over time, eventually blocking intersections. Control policies for oversaturated networks have the maximization of the throughput as primary objective.

3.4.5 Limitations of coordinated systems

The simplest methodology for creating a signal-timing makes a lot of simplifying assumptions, including a nominal volume of vehicles that reflects a “typical” condition. However, this is normally not the case. Besides the already mentioned issue of cost and maintenance,

there are several other reasons why this approach may fail. In traffic networks without well-defined traffic flow patterns like for instance morning and afternoon peaks, this approach may not be effective. This is clearly the case in large cities where business centers are no longer located exclusively downtown. Rather, there are several locations that serve as attractors for traffic so that no clear patterns exist. Also, in some cities, "secondary" streets have become as important as traditional arterials due to the saturation of these. Traffic patterns can also be affected by accidents, floods, snow, etc. Moreover, optimization of several intersecting arterials simultaneously in a grid-like network using real time data and under congestion is difficult.

These issues show that simple offline (or even on-line) optimization of the synchronization in *arterials* alone cannot cope with changing traffic patterns. With an increasing volume of traffic, the situation becomes more and more complex. Thus, flexible and robust approaches are necessary.

Finally, according to [66], there are limitations on how responsive a system of traffic signals can be. The implementation of a new cycle length or new offsets cannot be done instantaneously. Rather, the transition may take some time. In any case, during the transition, patterns are disrupted and large queues may begin to form. In general, it is not feasible to implement many different coordination patterns within a short period of time.

4 Approaches based on AI and MAS

In this section, AI and multiagent system-based approaches and frameworks to traffic signal control are presented and discussed. There is an expressive number of publications using various AI techniques in traffic control, such as genetic algorithms and fuzzy inference, which are not included here because they do not tackle the problem from a distributed and/or decentralized point of view. Thus the criteria to select the works presented next was primarily whether or not techniques from multiagent systems were used, are possible, or at least considered.

The taxonomy introduced in Sect. 3.3 is used to classify approaches whenever possible. Approaches for isolated, non-actuated intersections do not really fit into the spirit of AI. Thus it is no surprise that no non-actuated approach is discussed here. The first approach (next section) is the reservation-based system for isolated intersections, which does not specifically deal with conventional traffic signals as it is intended to provide orderly movement for *autonomous* vehicles. Sections 4.2–4.4 deal with approaches targetting isolated but actuated intersections, with coordinated systems, and with non-coordinated but networked intersections respectively. In particular, the main interest of this paper is on learning based approaches (both in single agent or multiagent flavors). These are covered in more details in Sect. 5.

4.1 Isolated intersections

4.1.1 Reservation-based

A reservation-based intersection control is proposed in [26] for a simplified version of real-world intersection without conventional traffic signals: autonomous guided vehicles (called AGV here) are not allowed to turn, do not change lanes, and all begin traveling roughly at the same speed. The reservation is performed as follows. First, the AGV informs the intersection manager (IM) the time it will arrive at the intersection, the velocity, direction, maximum and minimum acceleration and other properties of the vehicle. Then, the IM simulates the journey of the AGV, given the IM's knowledge about other similar reservations. If the space

requested by the AGV (for a given time) is already occupied, then the request is rejected, in which case the AGV must decelerate and try again. If the request is accepted, it must be kept or canceled by the AGV (in case it cannot be met).

The research presented in [26] has left as open questions: what happens if the driver has to make sudden changes; whether the AGV is really committed to the deceleration; what happens if conventional vehicles participate in the system as well; and what happens outside the look of the intersection manager after AGV's have to decelerate in response to a denial of request (e.g. what happens with other vehicles behind this one which have to decelerate too? What happens with their previous reservations?). Although authors claimed that those simplifications do not detract from the fundamental challenges of the problem, this is only valid in the situation in which only AGV's are using the intersection. As soon as conventional guided vehicles are present, these are likely to disturb the nice dynamic shown in the simulations, especially in what regards lane changing, an important (and difficult to model) issue in the theory of traffic flow (see e.g. [45]).

In order to cope with these issues, in [27] some of the previous assumptions were relaxed, especially the requirement that AGV's maintain a constant velocity in the intersection and do not make turns. AGV's are told to accelerate by the intersection manager. The improved protocol proposed is based on rules that vehicles are expected to follow: a vehicle may not enter the intersection without a reservation; the vehicle must try to follow actions prescribed by the intersection manager; the AGV cannot try to improve its own journey (as the manager will ignore a new request if the AGV has already a reservation granted).

In [29] a simulation is discussed in which the rate of human drivers to AGV's increases. The good news are that the delay decreases dramatically when only AGV's are present and, even better, seem to be unaffected by the increase in the traffic load. The bad news however is that the delay increases rapidly if more than 10% of human drivers are sharing the road with AGV's. In [29] a new component—a light model—was added to each IM so that it has different control policies to select from, such as first come, first served with all lights red, and rotation of green indication through all lanes. According to the authors, this light model does not work very well if most of the vehicles are human-driven, but it is useful for intersections which deal with mostly autonomous vehicles.

Authors also discuss (in [28]) how to tackle other shortcomings such as deadlocks and delays caused by drivers putting reservations in a suboptimal way because the intersection manager processes requests on a first come, first served basis. The solution proposed is that the IM waits until it had received all n requests in order to reorder them such that as many vehicles as possible make it through the intersection. This extension requires AGV's to keep communicating with the IM in order to inform new events. Since these events are highly interrelated for close AGV's (which is the case since the IM is dealing with a time slot for these AGV's to cross the same intersection), this turns into a kind of dynamic scheduling problem. In order to deal with this, a market-oriented approach is proposed: using some prepayment or credit system, vehicles could be required to pay for their reservations. The intersection can then be designed to maximize its revenue. In the market scenario, an AGV would be able to learn in order to minimize its delay while also minimizing the cost it incurs by making the appropriate reservations (e.g. those that are likely to be granted).

Regarding challenges related to multiagent systems, most learning approaches and in particular reinforcement learning are expected to have low performance given the large and distributed nature of the system. Besides, and probably more important, in large systems where each agent is trying to maximize its private utility, there is no guarantee that the global utility will also be maximized so that an individual-based approach could cause a suboptimal situation in terms of overall delay. In fact, this is a well known phenomena in traffic

management and control. Since the concept of Wardrop's equilibrium [90], traffic engineers face the problem of deleterious oscillatory behaviors caused by externalities and agents not facing the consequence of their individual choices [8,9,14].

Another extension proposed is to have multiple intersections coordinated in a more flexible way. Their experiments have shown that the system can move between different control policies smoothly and safely. For similar, thought not for AGV's, approaches to this proposal, see Sects. 4.3 and 5.2.

4.2 Isolated intersections, actuated

4.2.1 Learning classifier systems

Bull et al. [16] use learning classifier systems (LCS) to control a single intersection. Authors consider an intersection with pre-defined flow of vehicles on each of the four approaches. Signals at each of the four intersections in the network have two phases, one permitting north–south and the other east–west movements. This means that for each intersection, the controller decision determines the duration of each of the two phases. Each intersection is controlled by an LCS that receives a binary string as stimulus. Various aspects of the implementation of LCS for signal control were considered such as the way in which traffic data is presented to the LCS as stimulus for the rules, the form of the LCS reward for performance of its rules, a range of objectives for the control policy, and the choice of control action for the LCS rules to vary the signal control. Each of these was found to affect the performance of the LCS system to some extent. When tested with a detailed simulation of signal controlled road traffic, some implementations of the LCS methodology outperformed standard control methods in certain cases. This indicates that the LCS approach has potential for application to road traffic control. However, the control is again for single intersections; the paper does not mention how to integrate the control or tackle a network scenario.

4.3 Coordinated systems

Classical approaches to synchronization of traffic signals (see Sect. 3.3) work mostly off-line and focus on synchronization of traffic signals in an arterial. The main difficulty to extend the synchronization to a network or to more directions of traffic is the fact that due to traffic patterns, in some key intersections conflicts may appear because different directions compete for bandwidth. The traditional approach is to let a traffic expert decide which synchronization to implement. This section presents some approaches that either seek to replace the traditional arterial green wave by "shorter green waves" in segments of the network, or try to let agents dynamically negotiate over the question of which traffic direction shall be synchronized. In both cases the aim is to have more flexible solutions.

4.3.1 Swarm-based dynamic coordination

In [55] an approach based on swarm intelligence is proposed. Each intersection (plus its traffic signals) behaves like a social insect that grounds its decision-making on mass recruitment mechanisms found in social insects [34,64]. Signal plans are seen as tasks to be performed by the insect without any centralized control or task allocation mechanism. Stimuli to perform or to change tasks, are provided by vehicles that, while waiting for their next green indication, continuously produce "pheromone". Thus the volume of traffic coming from one direction

can be evaluated by the intersection agent, and this may trigger some signal plan switching. No other information is available to agents.

The approach was implemented and tested in a microscopic traffic simulator. Traffic signal agents perceive the pheromone trails and select an appropriate signal plan. Average density in the arterial was measured in order to compare the following situations: (i) traffic signals are not coordinated; (ii) they are coordinated in the classical way, i.e., using a central decision component (normally the traffic engineer) that determines a joint synchronization for all intersections; (iii) they are free to decide, at local level, whether or not to coordinate. Quantitatively, when agents are free to decide according to the swarm approach the system behaves almost as a central decision support system. Experiments show that agents achieve synchronization without any central management. However, the time needed to converge to a stable coordination can be high, which is a negative aspect especially in highly dynamic environments.

4.3.2 Coordination via constraint optimization

An approach based on an algorithm for distributed constraint optimization problems (DCOP) using cooperative mediation is proposed in [54]. It has the same purpose as the approach just described [55], and it is intended to be a compromise between totally autonomous coordination with implicit communication and the classical centralized solution (e.g. TRANSYT or SCOOT).

In the traffic signal scenario modeled via a DCOP, each agent is assigned to one ore more variables and these have interdependencies. The problem is to find an assignment such that the global cost is minimized. Agents can extend the context they use for local decision-making using a relationship graph. Within its graph, one of the agents has to act as a mediator, computing a solution for the extended context and recommending values for variables associated with the agents involved in the mediation session.

In the synchronization of traffic signals scenario, variables of the DCOP are the coordination direction for each traffic signal. Thus, the domain for all variables is given by two possible directions of coordination. Constraints in this problem arise from the fact that, in each node of the graph, a traffic signal cannot necessarily or efficiently coordinate with neighbors located in a different direction at the same time. A conflict occurs when two neighbors want to coordinate in different traffic directions.

The main results were: agents start the mediation and eventually reach a configuration of minimum cost; after the stabilization, only minor changes occur. When there are changes in traffic pattern, costs increase and a new mediation starts. Besides, comparing the DCOP-based approach to a situation without any mediation (i.e. with fixed coordination), the latter performs well only in the case coordination fits the traffic volume. One shortcoming of the mediation-based method is that the mediation may end up being performed by a single agent, thus in a centralized way. Additionally, the mediation process may take time.

4.4 Network, non-coordinated

4.4.1 Hierarchical multiagent system

A hierarchical multiagent system with three levels is proposed in [31]. In the first level, local traffic agents (LTAs) represent intersections; these are responsible for providing appropriate traffic signal timing which allows traffic to pass based only on local traffic patterns. Because the local optimum may not be a good one when observed from another perspective, there is a second level in the system, in which a coordinator traffic agent (CTA) supervises a few

LTAs. According to the authors, a CTA should provide a means by which the optimal local signal pattern can be slightly modified to accommodate the global performance. This is hardly a trivial question. In the paper, it is not discussed how this can be achieved; authors only mention that by means of communication between the local components and the CTAs and by means of storing all relevant information in a third level—controlled by an information traffic agent or ITA—it is possible to handle congestions. Also, some assumptions seem to be too strong. LTAs are assumed to be cooperative; they will always give up their optimal local control strategy, accept and implement the solution computed by a CTA, which is suppose to compute a "global" optimal solution. Thus LTAs have no autonomy. The authors also mention that once a LTA calculates its local optimum, the corresponding CTA is informed and adjusts local solutions to meet concerns at the CTA level. However, details are not given.

Another issue is that it is not mentioned what a global solution in this case is. If it is global in the sense of the CTA-controlled area, then a situation may arise in which two or more CTAs compute their "locally global" solution and, again, these may not be compatible or optimal from the network's (thus, global) point of view. Of course other level of CTAs could be implemented but ultimately the topmost CTA (at network level) would have to perform the whole computation. Assuming that an efficient, real time mechanism exists for this computation, the question would arise why the top most level does not then perform the computation in the first place, just informing LTAs what they have to do.

According to the authors, when an intersection is congested, neighbors are informed in order to respond accordingly. It is not clear how eventual conflicts are solved. Moreover, temporal relationships are not discussed: are two neighboring LTAs A and B trying to respond based on the intended (thus not yet implemented) control measures, or based on control measures just implemented, which proved not successful? What happens then if B sees conflict also with C, upstream?

In the paper two scenarios are discussed in which, due to congestion, traffic is halted. This might decrease the *overall* density but, as they notice, the speed also decreases and it is questionable whether this would not be better handled by giving detour advice to drivers instead of blocking their ways.

4.4.2 Multi-layer architecture

In [65] a multi-layered architecture is proposed. In the first layer, changes in the traffic patterns are detected on-line (via traffic detectors) and appropriate signal plans are selected. This is done by having several parameters specifying different aspects of the control (e.g. minimum or maximum duration of phases etc.) and using a LCS as in Sect. 4.2.1. The input data for this LCS is the observed volume of vehicles crossing the intersection. The problem is that this parameter only applies to a single intersection so that one can expect a local optimum at best. A second layer deals with previously unknown situations by searching for a signal plan based on off-line optimization. Situations not covered in layer 1 are reported to layer 2 and an evolutionary algorithm generates populations of parameters using a microscopic simulation model. A new classifier is created, mapping the observed situation to the parameters in the optimization process.

The paper reports no evaluation of the proposed multi-layered architecture. Also, given that a high computing time is required in layer 2, it would be interesting to see a discussion about whether a new generated solution still applies to the traffic situation when it is employed. Finally, the method looks promising (if performance issues are treated) and can be more useful when applied to a network of nodes instead of a single intersection because then the optimization of several parameters makes more sense.

4.4.3 History-based

Balan and Luke propose history-based controllers [1] intended to provide a sort of global fairness. According to the authors, the original inspiration was to allow drivers to let traffic signals know whether they are in a hurry or not, departing from previous work (e.g. [92]) that has explored what happens when traffic signals know about the trip plans of the drivers. In the history-based approach, information about vehicles' recent performance is collected by traffic signals. The authors base their approach on the notion of historical fairness by allowing vehicles to store credits they receive when waiting at red lights, and cash credits in when passing through intersections. Traffic signals base their decisions on credits of various vehicles at the intersection. When a vehicle reaches its destination, it reports its average waiting time over all intersections. This time is then used as one metric to assess the efficiency of the control. One issue is that vehicles have no interest in doing this and, if they do, there is no incentive to report the true commuting time. To overcome this, as future work the authors propose one action-based approach in which the intersection would grant green time to vehicles paying more.

4.4.4 Fuzzy inference

In [44], real-time simulation, multi-agent control, and fuzzy inference are combined to control a group of phases, each modeled as an agent which can changed the lights of the group to green when requested by traffic and when permitted by other agents. Hence there is a need for negotiation between agents about how to operate together. This approach has the same objectives as in [3], except that instead of using a game-theoretic approach based on coordination via game equilibria, a fuzzy inference is used.

To allow agents to negotiate regarding control decisions, they must exchange their local traffic and control data. There are plenty of possibilities related to how agents could reach a common control strategy. One possibility is that agents in each intersection negotiate regarding the extension of the green indication. In coordinated operation this negotiation process is affected by external signals from neighboring intersections. The fuzzy decision has only two options: extend or terminate green. The paper does not provide more details about how the fuzzy control is performed when it comes to negotiating with neighboring agents.

5 Reinforcement learning based approaches

5.1 Isolated intersections

5.1.1 Model-based learning with context detection

As noted in Sect. 2.1, due to the dynamic and non-stationary nature of flow patterns, one solution would be to keep multiple models of the environment (and their respective policies). Partial models have been used for the purpose of dealing with non-stationarity in [19,25]. However, these approaches require a fixed number of models, and thus implicitly assume that the approximate number of different environment dynamics is known a priori. Since this assumption is not always realistic in traffic, an alternative is to incrementally build new models as in [71,72]. In this approach (RL-CD for Reinforcement Learning with Context Detection), it is assumed that environmental changes are restricted to a small number of *contexts* (traffic patterns) which are stationary environments with distinct dynamics; that the

current context cannot be directly observed, but can be estimated according to the types of transitions and rewards observed; that the environmental context changes are independent of agent's actions; and that context changes are relatively infrequent. In a traffic scenario, these assumptions mean that flow patterns are non-stationary but they can be divided in stationary dynamics that need not be known a priori. In fact, one of the interesting aspects of the method is exactly its capability of automatically partitioning the environment dynamics into relevant partial models.

Each model is assigned to an optimal policy (which is a mapping from traffic patterns to signal plans), and to a trace of prediction error of transitions and rewards, aiming at estimating the quality of a given partial model. The creation of new models is controlled by a continuous evaluation of the prediction errors generated by each partial model. A partial model contains estimated transition and estimated reward functions.

For each partial model, classic model-based reinforcement learning methods such as Prioritized Sweeping and Dyna may be used to compute a locally optimal policy. If the environment changes and a local policy turns suboptimal (congestion increases over a threshold), then the system creates a new model. Whenever possible, the system reuses existing models instead of creating new ones. New models are created only when there are no models with trace error smaller than a defined threshold. Results show that the RL-CD mechanism is more efficient than a greedy strategy and other model-based reinforcement learning approaches. Although this mechanism was tested in a network of nine traffic signals, it remains a single-agent based learning method and an extension is necessary in order that agents map states and joint actions to rewards.

5.2 Coordinated systems

5.2.1 Game-theoretic approach

In [4,5] techniques of evolutionary game theory and SG are used: individually-motivated agents (traffic signals) act in a dynamic environment in which not only their own local goals but also a global one can be taken into account. This is achieved with each agent having only local knowledge that it obtains from sensoring its local environment. This way agents are able to respond to their local environment state. However, they also perform experimentation and, according also to the experimentation performed elsewhere in the neighborhood, they receive a reward. Stochastic events that may take place in the network are modelled by mutations. During the learning process, a fitness for each strategy is computed and it influences the next generation of strategies which will be used by agents to perform the experimentation. Depending on the frequency of the stochastic events, agents are able to coordinate better towards the global goal. Agents neither observe the distribution of strategies in the population nor are able to calculate best responses. Due to the lack of communication among agents, the general traffic pattern remains unknown to them. For instance, if the trend is that the traffic volume is predominantly westbound, agents performing executing a policy that provides longer green indication for that traffic direction are better paid than those executing other policies. At each period, there is a probability that each agent learns how good the set of strategies played in the near past was. Besides the learning probability, at each stage agents have also a small probability of mutating (selecting a different action).

Several scenarios have been simulated, varying the learning and mutation parameters (see [5] for details). Results showed that a central synchronization performs better in stable scenarios with the flow of vehicles being clearly higher in one direction than in the opposite since

few or no conflict occurs. However, in scenarios where the volume of vehicles is nearly equal in both directions, the central progression does not perform well compared to the agent-based mechanism. This can be explained by the fact that the agent-based mechanism is adaptive and allows agents to break with the synchronization in order to cope with their local traffic conditions for a short time period, if necessary.

A shortcoming of the approach is that payoff matrices (or at least utilities and preferences of agents) have to be explicitly formalized by the designer of the system. This makes the approach time consuming when many different options of coordination are possible as for example all four traffic directions have to be considered.

5.3 Network, non-coordinated

5.3.1 Co-learning based on waiting time

Wiering [92] describes the use of reinforcement learning by traffic light agents in order to minimize the overall waiting time of vehicles in a small grid. Those agents learn a value function that estimates expected waiting times of vehicles given different settings of traffic lights. One interesting issue tackled in this research is that a kind of co-learning is considered: value functions are learned not only by traffic signals, but also by the vehicles that can thus compute policies to select optimal routes to their destinations.

The ideas and results presented in this paper are important. However, there are some issues that may be significant if such an approach is to be deployed. First, the kind of communication and knowledge (or more properly communication *for* knowledge formation) may have a high cost. Also, it is questionable whether vehicles (drivers) themselves "would exactly know their waiting time until they arrive at the destination address given that their traffic light is currently set to red or green". Second, it seems that traffic signals can shift from red to green and opposite at each time step of the simulation. Although the size of this time step is not given, the text conveys the idea that it is a small fraction of time (say seconds). In the practice of traffic engineering, changes are introduced only in a smooth way as for instance when phases are allowed to be extended. Third, there is no account of experience made by drivers based on their local experiences only. It seems that their past experiences in terms of route choice and travel time are not considered. Finally, drivers being autonomous, it is not obvious to expect that all will use the best policy, which, in this case, was computed by the traffic signal and not by the driver itself.

As for the experiments and results, the paper investigates the use of reinforcement learning in different flavors and under different saturation conditions in a grid-like network. It is shown that reinforcement learning starts to payoff when the grid starts to saturate.

A similar RL-based method for controlling traffic lights is presented in [73] to minimize the total travel time of all vehicles in the network. Thus, the control perspective is a global one, although actions are local to the agents. Agents here are the traffic signals but the learning task is formulated in a way that the state representation is vehicle-based (waiting times for individual vehicles), aggregated over all vehicles around the intersection. Another issue is that the more information about the individual vehicle, the bigger the state space. The paper also investigates other forms of state representation and different learning abilities. Experiments were performed with different volumes of traffic and the evaluation was performed regarding a global measure, namely average waiting time.

5.3.2 Co-evolution

As seen in Sect. 3, classically, assignment is done via network analysis (e.g. via Wardrop equilibrium). However equilibrium-based concepts are not completely adequate as they overlook the within-day variability for example. In [7,11] authors investigate what happens when different actors interact, each having its own goal and learning algorithm. The objective of *local* traffic control is obviously to minimize queues in a spatially limited area (e.g. around a traffic light). The objective of drivers is (normally) to minimize their travel times. The scenario used is a typical commuting scenario modeled as a 6×6 grid, where drivers repeatedly select a route to go from an origin to a destination. These routes are not so simple as a two-route (binary decision) scenario; it is possible to set arbitrary origins and destinations. Thus drivers have a large set of routes to select from.

The control is done via decentralized traffic signals. Each has a default signal plan that divides the cycle time equally between two phases. The actions of traffic signals are to keep this default plan or to priorize one phase. Strategies are: (i) always keep the default signal plan; (ii) greedy (run green time for the phase with the higher occupancy); (iii) use single agent Q-learning. Regarding the drivers, these may use three strategies: (i) select a route randomly (each time it departs); (ii) select a route greedily (always pick the one with best average travel time so far); (iii) select a route in an adaptive way meaning that average travel times so far are used to compute a probability to select the route to use.

Results show an improvement regarding travel time and occupancy when all actors co-evolve especially in large-scale situations involving hundreds of drivers. This was compared to situations in which either only drivers or only traffic signals evolve, in different scenarios.

One issue left open by the authors is *en-route* adaptation. This was not initially considered because approaches in which there are more than two routes between two locations, and agents can change their routes on the fly are not trivial. An operational problem is the generation of reasonable alternatives. The problem of generation of routes for route choice models is well known in traditional discrete choice approaches: the n shortest paths may differ only marginally. Additionally, all approaches consider one route as one complete option to choose from.

Therefore the issue of on-the-fly re-routing is investigated in [10]. Here drivers react to their perception of jammed links. Notice however that the focus is on adaptation, not on learning, with drivers using different criteria to decide whether to deviate from the originally planned route. Although the authors have shown that re-routing may compensate an eventual inefficient traffic control (by the traffic signal agents), it remains an open question how this can be combined with reinforcement learning techniques by the drivers.

5.3.3 Stochastic-game based

The task of operating a traffic network as a distributed, stochastic game in which agents solve reinforcement-learning problems is investigated by Camponogara and Kraus [18]. They use a variation of stochastic games in which states are only partially observable, and mention that one cannot expect to compute a set of policies, one for each agent, that is optimal. They also note that this set of policies, even if it exists, may conflict with the performance yielded by an optimal, centralized policy that maximizes the sum of rewards of all agents, which is the principal goal in operating a traffic network. This relates to the issue of collectives discussed in Sect. 2.2.

From the standpoint of each agent, its control task could be thought of as an ordinary reinforcement problem by regarding other agents as part of the environment, except that this environment is not stationary as it depends on the policy implemented by other agents. Despite this, authors employ standard reinforcement learning algorithms (called distributed Q-learning in their paper) to reach a set of distributed control policies.

For the experiments, a small network of two intersections is used, where roads have limited capacities. Traffic conditions were varied, as well as the kind of policy followed: uniformly random policy (assigns the same probability to all actions available to an agent); best-effort policy (green indication to the lane with the longest queue); Q-learning implemented by agent-1 (agent-1 applies the Q-learning algorithm to control traffic signals at its intersection while agent-2 follows the uniformly random policy); Q-learning implemented by agent-2. Results of Q-learning against uniformly random policy show that a reduction of 18% in the average waiting time is induced by agent-1 or agent-2, if either agent implements Q-learning. Authors also report a reduction in the waiting time of 43% when both agents run Q-learning, compared to best-effort policy, and conclude that the best-effort policy outperforms the random policy for low traffic densities. However the former incurs higher waiting times under heavy traffic conditions.

5.3.4 Learning in heterogeneous groups

The question of how heterogeneous groups of agents can benefit from communication to improve their learning skills is investigated in [52], where information from several sources during learning is used in a simplified simulation of a traffic control problem. Here, teams of agents are in charge of two connected crossings in a certain area. Each crossing is controlled by a different agent. Members of a team may communicate with their partner in the same area or with members of other teams that are solving similar problems in different areas (up to three areas were used in the experiments). The reward at each timestep is the weighted sum of two terms, which represent the compromise between improving the individual performance at a crossing, and the global quality in the area controlled by the team.

Different types of agents were used: EA-agents (evolution of population of neural networks), QL-agents (connectionist Q-learning), and H-agents (heuristic agents). H-agents respond to occupation thresholds, changing to green the traffic-lights of lanes that have higher occupation rates. Besides, agents can learn from advice given by their peers. An advice is composed of a state experienced by the advisee agent, an action proposed by an advisor agent, the reward the advisor would expect to achieve by taking the proposed action, and the confidence the advisor estimates for the information. Advice is generated upon request. When an agent decides to request advice it sends the observed state to the selected advisor. To estimate the reward that will be obtained, as well as the confidence in the action proposed for this state, the advisor uses information previously saved. The quality of a previous advice given by a certain peer can be seen as a measure of trust. Trust, in this case, is related to how accurate the estimated reward given for previous advice was. When information is requested concerning a given state, the advisor also gives the reward that it would expect to achieve using the proposed action.

Experiments were conducted based on real data from Lisbon. Teams of agents, each using a different decision mechanism (EA-agents, QL-agents and H-agents) were employed. H-agents do not request advice, but they respond to all advice requests. The following simplifications were assumed: vehicles do not change lane or turn; forward movement is of type follow-the-leader (observing the desired maximum speed of each vehicle); vehicles can break to zero-speed instantly; stopped vehicles, in crossings, do not prevent others from passing

in other directions. Traffic-lights are prompted for a new decision every 20 s, which may be too frequent. A simulation runs first in training mode. After, the best parameters saved are reloaded and it runs in test mode (without communication and learning). The state that agents observe is composed of parameters representing occupation rate in each of the four incoming directions, incoming traffic from a given direction, the current color of traffic lights, and the time since the last change in the traffic-light.

Results indicate that EA-agents may reach scores higher than QL-agents. This might be explained by the size of the search space that QL-agents have. EA-agents may perform well even when none of their peers is able to reach scores at the same level. Authors note that advice exchange does improve the average performance of agents with lower scores, although not all agents reach the same performance levels and some advisees do not reach the level of performance of advisors.

5.3.5 Learning in groups with advice

The idea of giving advice is also explored in [6]. It is argued that one possible way to reduce the complexity of the problem is to have agents organized in groups of limited size so that the number of joint actions is reduced. These groups are then coordinated by another agent, a tutor or supervisor. The paper investigates and compares the task of multiagent reinforcement learning for control of traffic signals under the following two situations: agents act individually (individual learners), and agents can be "tutored", meaning that another agent with a broader sight will recommend a joint action. Results show that supervision pays off: when there is no supervision, agents just learn using individual Q-learning and in this case the number of stopped vehicles is higher than when supervisors give advice to the traffic signal agents.

6 Approaches based on cellular automata, self-organization, and optimization

Approaches that were proposed by physicists and mathematicians (especially those related to operations research and optimization) are discussed next. In these, artificial intelligence plays a minor role, if any. Also, they are not necessarily aimed at traffic signal control. More generally, they deal with microscopic as well as macroscopic models for simulation of movement of vehicles (next subsection), and optimization of traffic flow (Sects. 6.2 and 6.3).

6.1 Microscopic models of vehicular traffic

Most of the works published by physicists tackle the "coarse-grained" fluid dynamics related to the description of traffic flow. This means that this is a macroscopic modeling where platoons, not individual vehicles are treated. These are not discussed here.

Microscopic models are a trend because they allow fine-grained description of all actors involved. In microscopic approaches traffic is treated as a system of interacting particles driven far from equilibrium. Therefore the main focus of these publications is to study various fundamental aspects of nonequilibrium systems under the light of statistical physics. Analytical as well as numerical techniques of statistical physics are being used to investigate transitions from one dynamical phase to another, criticality and self-organized criticality, etc. For example, Helbing and Huberman [37] discuss simulations based on a discretized follow-the-leader algorithm resulting in the existence of cooperative, coherent states arising from competitive interactions.

The prototypical microscopic approach is based on cellular automata (CA). Details can be found in [51] (the seminal paper by Nagel and Schreckenberg where the CA model was proposed) and in a review, written from the perspective of statistical physics [20]. The CA model represents a minimal model in the sense that it is capable of reproducing several basic features of real traffic using only a few behavioral rules. The road is subdivided in cells that can be either occupied by one vehicle, which has an integer speed $v_i \in \{0, \ldots, v_{\max}\}$ with a maximum speed $v_{\max}$. The dynamics of vehicle motion is described by four simple rules (see [51]). Every driver described by the Nagel–Schreckenberg model can be seen as a reactive agent: autonomous, situated in a discrete environment, and having (potentially individual) characteristics such as its maximum speed $v_{\max}$ and deceleration probability p. However, real drivers do not react in this relatively simple way. Rather, sometimes they vary their driving behavior for no obvious reasons. Thus, there is room for multiagent system-based formulations of this modeling as in [9,12,42,79,87–89].

As mentioned, physicists have been mainly concerned with equilibrium issues and/or with the investigation of critical phenomena in simulation of traffic flow at network level, centered on vehicles movement. Optimization and adaptation of traffic flow has become a topic of interest only recently. In [15] the impact of a coordination among traffic signals is simulated in a cellular-automata-based model. This model was originally designed for highway traffic [51] and later extended [21] for detailed city traffic. The goal was to test the extended CA model to find optimal cycle times for traffic signals in the network, which has a simple square lattice geometry. Results refer to constant density in traffic volume in both directions. In their case, due both to the lattice geometry and to similar densities in all directions, green waves with large bandwidth can be set in both directions. However the case in which one direction gets more traffic load is not discussed so that the optimization problem only applies to static and well-behaved situations.

6.2 Self-organization-based approach

Gershenson [33] approaches the problem of improving traffic flow not from the optimization point of view but from the adaptation one. Here, traffic lights self organize by means of three methods, with no direct communication between them. It is shown that the adaptation to traffic conditions reduces waiting times and number of stopped vehicles. The quest of adaptation instead of optimization is similar to the one in [5,55]. However, in [33] this is done for isolated intersections, with no form of coordination among agents as e.g. in a progressive system.

6.3 Approaches based on mixed integer programming

Möhring et al. [48] deal with fixed-time signal control, introducing two approaches to minimize total delay modeled as a mixed-integer problem. Their approach is useful for arterial optimization; however, the mixed-integer formalization of the problem is associated with a large computing time.

In Köhler et al. [43], authors describe a model for offline optimization of the offset in synchronized systems. Their solution is modeled as a mixed-integer problem and supports non-uniform cycle lengths operating near saturated conditions. One drawback is that only fixed-time signal control is considered, thus not tackling adaptation to changing environments. On the other hand, problems related to the computational complexity of previous approaches is overcome with piecewise linearization of the delay function.

7 Summary and classification of approaches

The approaches described in Sects. 4–6 are summarized here according to some dimensions: kind of technology used; whether they apply to single intersection, coordinated systems or networks; whether or not they are traffic responsive (actuated); whether or not they are decentralized; and which is the main technique used. Table 1 includes only conventional approaches.

In Table 2, the approaches are not conventional ones; most are developed as testbeds for some technique from AI or other area, and none is deployed. The two first lines in the table refer respectively to approaches proposed by physicists and mathematicians, and from AI and multiagent systems, excluding learning based approaches. These are presented in the last line.

Table 3 lists approaches that deal with non conventional, not yet fully deployed technologies such as autonomous guided vehicles, GPS tracking, wireless communication, etc.

8 Open challenges for MAS-based approaches

8.1 Agents as traffic signals: to cooperate or not to cooperate

Conventional control in isolated intersections uses mathematical programming and operations research tools based on constraints and optimization of operational parameters (length of

Table 1 Classical approaches: coordinated systems

Actuated	Non-actuated
SCOOT [40], SCATS [47], PRODYN [38], OPAC [32], UTOPIA [23], TUC [24]	TRANSYT [78]

Table 2 Actuated approaches: centralized, from physics and optimization (row I); decentralized, from AI and MAS, except learning (row II); decentralized, learning based (row III)

	Isolated intersections	Coordinated systems	Two or more intersections
I	Gershenson [33]	CA-based [15]; Köhler et al. [43]	–
II	Bazzan [3]; Bull et al. [16]	Oliveira et al. [53–55]	France and Ghorbani [31]; Kosonen [44]; Rochner et al. [65]
III	Silva et al. [72,71]	Bazzan [4,5]	Bazzan et al. [7,10,11]; Camponogara and Kraus [18]; Nunes and Oliveira [52]; Steingrover et al. [73]; Wiering [92]

Table 3 New technologies

Isolated intersections
Balan and Luke [1]
Dresner and Stone [27]

cycle, split, minimum and maximum green time). Several tools from AI such as evolutionary computation, learning, and MAS-based approaches have proven useful in this problem (Sects. 4 and 5), as they are able to deal with data collected from sensors in a more intelligent and adaptive way.

However, major challenges lie in control of arterial and networks, especially when several intersections must work in a synchronized way because this means contention and conflicts between agents. The reason is that the several actors must coordinate their actions in order to reach an effective performance not only at local level. This task is further complicated by the fact that the task itself is not of a cooperative nature. In [77] opportunities for cooperation in MARL are identified such as communication of instantaneous information (perception, actions, rewards); sequences of triples ⟨sensation, action, reward⟩; learned policies. Communication and cooperation are natural in the prey-predator scenario used to illustrate opportunities for cooperation. However, it is not obvious what "intelligent cooperation" means in the traffic control scenario.

Additionally, in synchronized systems, due to topological constraints discussed in Sect. 3.4.1, the direction of the synchronization is a non local, possibly conflicting decision, that also includes non-local performance. Thus, one important issue is whether or not self-motivated, possibly non cooperative agents can handle this problem. In [4] it was shown that this can be done, at the cost of the designer having to formulate payoff matrices.

Alternative approaches that alleviate this issue were proposed but have other drawbacks. The swarm-based approach for synchronization [55] requires time to build a green wave. The cooperative mediation approach [54] assumes cooperative agents and centralizes the mediation in a few (often only one) agent. Besides, the computational complexity of a DCOP based method prevents its use in large networks. The other approaches discussed all have the problem that they do not explicitly deal with conflict resolution (e.g. in which traffic direction to synchronize traffic signals as in [31,44,65]) or base this decision on pre-specified rules or on traffic experts [24].

The decision about which traffic direction to synchronize, and thus give priority, can be further tackled in many ways:

1. By explicit communication among traffic signals, which is nowadays possible due to technical advances in communication devices (e.g. wireless networks). Using new technologies, it is possible to exchange sensor and control data. However, just data exchange is not sufficient for at least two reasons. First, it is not clear how having more data could help since there is no computationally efficient method to resolve conflicts, since this involves data acquisition (from sensors) and processing in real time. Second, it is not obvious why intersection agents would sacrifice local performance in favor of a global one. More philosophically, this question leads to the issue of to what extent those agents would still be autonomous if they were to follow a hierarchically superior agent (e.g. an arterial manager) in order to implement the synchronization of traffic signals in the traffic direction imposed by the manager. This is not a trivial question especially as it involves other agents, namely drivers, who can then react to inefficient local control policies. This issue of co-adaptation is further addressed in Sect. 8.3.
2. Instead of explicitly exchanging data, agents could do this implicitly by sensing the environment and adapt (or react) to it accordingly and eventually form several sub groups of synchronization as in [5,55]. These approaches use neither explicit communication nor a manager agent to resolve conflicts. There is of course a trade-off between time taken to adapt and having immediate conflict resolution by communication and by a manager agent. In highly dynamic environments, an agent might not have the time to

perceive a change and adapt to it before the environment has already changed again. In reasonably well behaved scenarios, the adaptive approach with formation of several mini green waves has performed well compared to arterial-based, fixed green waves.

3. Synchronization can be achieved by traffic signal agents that learn, e.g. using reinforcement learning and game-theoretic approaches. Regarding the SG defined in Sect. 2.2.1, this has to be adapted for the traffic control setting. In particular, for traffic control, each agent i has a partial observation of the whole environment so that there is no single state vector; rather, S is actually the cartesian product over all partial observed states and all agents ($\times S^i$).[3] Thus, if only traffic signals are considered (the situation with driver as agents is discussed in Sect. 8.3), the action space is the cartesian product over all set of actions of the traffic signals.

As seen, the problem here is already very complex for standard reinforcement learning based approaches; it does not scale up to more than a dozen traffic signal agents. In the next subsection, some possible solutions are discussed together with their strengths and drawbacks.

8.2 Traffic signal control: challenges for multiagent reinforcement learning

The case of isolated intersection (thus single agent reinforcement learning) was discussed in Sect. 5.1. It was shown that the main problem occurs when the environment is non-stationary. Besides this problem, even in the case of stationary environments, the reinforcement learning task is further complicated by the potential scale of the problem. Assume that, in a single intersection, the following holds: there are four sets of approaching lanes (four traffic directions), the traffic signal agent is able to sensor the traffic volume in all approaching lanes, each of these is discretized in s states, and the set of actions consists of k signal plans to select among. The number of states is s^4, and the number of pairs state-action is $s^4 \times k$. Already for small s and k this can be a hard task for reinforcement learning methods, especially in non-stationary environments. Moreover, many of these state-action pairs should not even be visited as they are known to be suboptimal (e.g. allowing a long green time to lanes with low traffic, while letting short green time to another direction that has a heavy traffic). Therefore, heuristics used by traffic experts are likely to perform better here.

This picture changes when it comes to several intersections considered all together because either it is not so obvious how these heuristics look like, or the traffic expert is likely to have heuristics only for a particular, small portion of the traffic network. Besides, as already said, MARL is more complex than SARL. Due to the specific characteristics of traffic control scenarios, in several cases, already established and tested approaches for RL cannot be used. The environment is generally dynamic and hard to predict not only due to the stochastic nature of traffic patterns, but also because other agents may be learning concurrently; the reward that one agent receives strongly depends on the actions performed by typically many other agents; not all actions are observable.

Hence, as also noted by Stone [75], it makes little sense to expect convergence to an equilibrium. Therefore the question remains how to design such agents so that they achieve some degree of system effectiveness. One possibility is to simplify and model the problem as several MDP's, one per agent. This could be done in several ways:

1. All agents have local set of actions, the reward is local but they all perceive the overall system state, i.e. the performance of the whole system is discretized over one parameter

[3] The majority of papers reviewed in Sect. 2.2.2 assume that all agents are able to observe the state of the whole environment so that obviously $|S| \ll |\times S^i|$; besides most works on repeated games assume $|S| = 1$.

or over a combination of several parameters resulting in state values such as overall congestion between 0% and 10% etc. (or average travel time, delay, queue, or any other global metric). In current traffic scenarios, this can be seen as a strong assumption, since this would mean informing all traffic signals about the current state of the network. However, new technologies are being developed. For instance, it will soon be possible to measure average velocity and other global variables in a traffic network: The city of Stuttgart uses the fleet of taxis as floating cars; data is collected to a Floating Car Data (FCD) system. This way, it is possible to know position and velocity of these floating cars. If the programming of the traffic signals's controllers is centralized or if there is at least some communication channel between a central of control and the traffic signals, then it is reasonable to assume that traffic signals can be periodically informed about the overall state of the traffic network.

2. Similar to the item above but here everything is local: set of actions, rewards, and states. With the current technology, traffic signals can perceive their own states (via detectors) so this would pose no problem.
3. Stateless learning, learning automata: only actions are mapped to rewards.

All three approaches above are simple to implement but are questionable from one or more of the following issues, to a lesser or greater extent: (i) assumptions made about informational state of agents, and (ii) the assumption (for Q-values computation) that actions by each agent are independent of the action selected by other agents. Regarding the latter, any kind of sophistication in the (re-)definition of Bellman's equations such as $Q^i(s, \mathbf{a}) \leftarrow (1 - \alpha)Q^i(s, \mathbf{a}) + \alpha[r^i(s, \mathbf{a}) + \gamma V^i(s')]$ (where the exponent i refers to agent i and $\vec{a}$ is the vector of *joint* actions) leads to the problem of how to update the V term (i.e. the $\max_{\mathbf{a}\in\times A} Q^i(s', \mathbf{a})$), not to mention the increasing in the size of the action space. As seen in Sect. 2.2.2, when approaches based on simple application of multiagent Q-learning are used, there is the problem of ending up in a suboptimal solution (e.g. [22]), as agents cannot distinguish the effect of their own actions.

Using one Q table per "opponent" (as in [39]) would cause each traffic signal in a grid scenario to keep, each, at least three Q tables: its own, that of the neighbor to which it is connected in the north-south direction, and that of the neighbor in the east-west direction.[4] Additionally, this assumes that each agent knows actions and rewards of other agents, an assumption which may lead to communication bottlenecks. Even if we deal with stateless versions of those approaches, assuming observation of all joint actions remains a problem.

8.3 Drivers and traffic signals as learning agents

Most of the discussion above holds when the subject of MARL is the driver or when both kinds of agents act in a traffic network (which is the real case). However, the scale of the problem is much bigger then, since the number of drivers tend to be in the order of thousands. Besides, the number of drivers in the network changes within time, an additional reason why convergence to an equilibrium is not a good metric.

When considering traffic signals and drivers as learning agents, $N = \mathcal{D} \cup \mathcal{T}$ (set of agents is the union of the set of drivers and set of traffic signal agents). Thus, formally, the SG defined in Sect. 2.2.1 has to be modified:

- $N = \mathcal{D} \cup \mathcal{T}$;
- each agent i has only a local, individual observation of the whole environment so that the state space is actually the Cartesian product over the individual state sets ($\times S^i$);

[4] This decreases to two tables in the case of an one-way arterial rather than a grid.

– the action space is the Cartesian product over all set of actions of drivers and traffic signals.

With these figures, it is obvious that, if we use a SG-based approach that considers all states and actions, each agent needs to keep tables whose sizes are exponential in the number of agents: $|S^1| \times \cdots \times |S^k| \times |A^1| \times \cdots \times |A^k|$. Assuming a very simple discretization of states, namely that all traffic signal agents can map local states to either jammed or not jammed, i.e. $|S^i| = 2$ for $i = 1, \ldots, |\mathcal{T}|$, and that drivers cannot perceive more than one state, the Cartesian product over the states has a size $2^{|\mathcal{T}|} \times 1^{|\mathcal{D}|}$. Assuming also that traffic signals have only two actions (two signal plans) and drivers have at most five actions (five routes to choose from), the size of Q tables is $2^{|\mathcal{T}|} \times 2^{|\mathcal{T}|} \times 5^{|\mathcal{D}|}$. Already the last term makes this approach computationally intractable as the number of drivers tends to grow, not to speak about the communication demand.

Therefore, one solution is to treat drivers as part of the environment and let traffic signals learn. Even this can be prohibitive if the number of traffic signals is bigger than, say, three to five.

Letting all drivers act locally but have knowledge of the overall system state (e.g. the performance of the whole system in terms of overall congestion) is a strong assumption: even with the increasing deployment of ATIS (Sect. 3.1), it is known that drivers cannot process much information. When this assumption is dropped and perceptions are local, this is again questionable because the concept of what a state is for the drivers is quite vague. An action, a choice of route, is rewarded or not (in terms of travel time or any other metric) and this is probably what matters. Therefore, the stateless formulation of a model free approach to reinforcement learning would do fine, in spite of the known problems associated with learning in congestion games. An intermediate version of both items above would be to let drivers only do the action-reward mapping, in a stateless Q-learning or learning automata fashion.

In summary the issue of SG-based learning in such scenarios is not trivial and improvements in the current algorithms regarding learning and collectives are necessary.

9 Concluding remarks

This paper has discussed some open challenges for employing techniques from AI and MAS in traffic control. A survey of classical methods, as well as of methods coming from AI, computer science, operations research, and physics was presented, emphasizing challenges and open problems with these approaches. An important point is that the area of traffic control is an attractive testbed for MARL approaches as well as for other techniques from MAS, especially coordination. The problem of how to synchronize traffic signals (while also keeping large bandwidths) is non trivial. Other open issues were discussed in Sect. 8.

Summarizing, the three more important directions are whether or not to tackle traffic control as a cooperative problem/domain, meaning that local optima generally conflicts with the global (network) optimum; the dimensionality problem associated with learning in a system with many agents, where each implicitly affects others' decisions; co-evolution and the role of the behavior of drivers in a transportation network.

Acknowledgements The author would like to thank co-authors of papers related to the subject of this survey, especially F. Klügl, K. Nagel, and J. Wahle, as well as all students who have worked in related projects, and V. Lesser for his support. All these projects could not have been carried out without the financial support of CNPq (grant 306892) and CAPES (Brazil); DLR, BMBF, DAAD, and Alexander von Humboldt Foundation (Germany).

References

1. Balan, G., & Luke, S. (2006). History-based traffic control. In H. Nakashima, M. P. Wellman, G. Weiss, & P. Stone (Eds.), *Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems* (pp. 616–621). New York: ACM Press.
2. Balmer, M., Cetin, N., Nagel, K., & Raney, B. (2004). Towards truly agent-based traffic and mobility simulations. In N. Jennings, C. Sierra, L. Sonenberg, & M. Tambe (Eds.), *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multi Agent Systems, AAMAS*, July 2004 (Vol. 1, pp. 60–67). New York: IEEE Computer Society.
3. Bazzan, A. L. C. (1995). A game-theoretic approach to distributed control of traffic signals. In *Proceedings of the 1st International Conference on Multi-Agent Systems (ICMAS)* (p. 439, extended abstract). San Francisco.
4. Bazzan, A. L. C. (1997). *An evolutionary game-theoretic approach for coordination of traffic signal agents*. PhD thesis, University of Karlsruhe.
5. Bazzan, A. L. C. (2005). A distributed approach for coordination of traffic signal agents. *Autonomous Agents and Multiagent Systems, 10*(1), 131–164.
6. Bazzan, A. L. C., de Oliveira, D., & da Silva, B. C. (2008). *Learning in groups of traffic signals*. Technical report, UFRGS.
7. Bazzan, A. L. C., de Oliveira, D., Klügl, F., & Nagel, K. (2008). Adapt or not to adapt—Consequences of adapting driver and traffic light agents. In K. Tuyls, A. Nowe, Z. Guessoum, & D. Kudenko (Eds.), *Adaptive agents and multi-agent systems III*, Lecture notes in artificial intelligence (Vol. 4865, pp. 1–14). New York: Springer-Verlag.
8. Bazzan, A. L. C., & Junges, R. (2006). Congestion tolls as utility alignment between agent and system optimum. In H. Nakashima, M. P. Wellman, G. Weiss, & P. Stone (Eds.), *Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems*, May 2006 (pp. 126–128). New York: ACM Press.
9. Bazzan, A. L. C., & Klügl, F. (2005). Case studies on the Braess paradox: Simulating route recommendation and learning in abstract and microscopic models. *Transportation Research C, 13*(4), 299–319.
10. Bazzan, A. L. C., & Klügl, F. (2008). Re-routing agents in an abstract traffic scenario. In G. Zaverucha & A. L. da Costa (Eds.), *Advances in artificial intelligence*, Lecture notes in artificial intelligence (Vol. 5249, pp. 63–72). Berlin: Springer-Verlag.
11. Bazzan, A. L. C., Klügl, F., & Nagel, K. (2007). Adaptation in games with many co-evolving agents. In J. Neves, M. Santos, & J. Machado (Eds.), *EPIA 2007*, Lecture notes in artificial intelligence (Vol. 4874, pp. 195–206). Berlin: Springer-Verlag.
12. Bazzan, A. L. C., Wahle, J., & Klügl, F. (1999). Agents in traffic modelling—From reactive to social behavior. In *Advances in artificial intelligence*, Lecture notes in artificial intelligence (Vol. 1701, pp. 303–306). Berlin/Heidelberg: Springer. Extended version appeared in *Proceedings of the U.K. Special Interest Group on Multi-Agent Systems (UKMAS)*, Bristol, UK.
13. Bowling, M. H., & Veloso, M. M. (2001). Rational and convergent learning in stochastic games. In B. Nebel (Ed.), *Proceedings of the 17th International Joint Conference on Artificial Intelligence* (pp. 1021–1026). Seattle: Morgan Kaufmann.
14. Braess, D. (1968). Über ein Paradoxon aus der Verkehrsplanung. *Unternehmensforschung, 12*, 258.
15. Brockfeld, E., Barlovic, R., Schadschneider, A., & Schreckenberg, M. (2001). Optimizing traffic lights in a cellular automaton model for city traffic. *Physical Review E, 64*(5), 056132.
16. Bull, L., Sha'Aban, J., Tomlinson, A., Addison, J. D., & Heydecker, B. G. (2004). Towards distributed adaptive control for road traffic junction signals using learning classifier systems. In L. Bull (Ed.), *Applications of learning classifier systems*, Studies in fuzziness and soft computing (Vol. 150, pp. 276–299). New York: Springer.
17. Burmeister, B., Doormann, J., & Matylis, G. (1997). Agent-oriented traffic simulation. *Transactions Society for Computer Simulation, 14*, 79–86.
18. Camponogara, E., & Kraus, W. Jr. (2003). Distributed learning agents in urban traffic control. In F. Moura-Pires & S. Abreu (Eds.), *EPIA* (pp. 324–335). Beja, Portugal.
19. Choi, S. P. M., Yeung, D.-Y., & Zhang, N. L. (2001). Hidden-mode markov decision processes for nonstationary sequential decision making. In R. Sun & C. L. Giles (Eds.), *Sequence learning: paradigms, algorithms, and applications* (pp. 264–287). Berlin: Springer.
20. Chowdhury, D., Santen, L., & Schadschneider, A. (2000). Statistical physics of vehicular traffic and some related systems. *Physics Reports, 329*, 199–329.
21. Chowdhury, D., & Schadschneider, A. (1999). Self-organization of traffic jams in cities: Effects of stochastic dynamics and signal periods. *Physical Review E, 59*(2), R1311–R1314.

22. Claus, C., & Boutilier, C. (1998). The dynamics of reinforcement learning in cooperative multiagent systems. In *Proceedings of the 15th National Conference on Artificial Intelligence* (pp. 746–752). Madison, Wisconsin.
23. Di Taranto, M. (1989). UTOPIA. In *Proceedings of the IFAC-IFIP-IFORS Conference on Control, Computers, Communication in Transportation*, Paris. ifac.
24. Diakaki, C., Papageorgiou, M., & Aboudolas, K. (2002). A multivariable regulator approach to traffic-responsive network-wide signal control. *Control Engineering Practice, 10*(2), 183–195.
25. Doya, K., Samejima, K., Katagiri, K., & Kawato, M. (2002). Multiple model-based reinforcement learning. *Neural Computation, 14*(6), 1347–1369.
26. Dresner, K., & Stone, P. (2004). Multiagent traffic management: A reservation-based intersection control mechanism. In N. Jennings, C. Sierra, L. Sonenberg, & M. Tambe (Eds.), *The 3rd International Joint Conference on Autonomous Agents and Multiagent Systems*, July 2004 (pp. 530–537). New York: IEEE Computer Society.
27. Dresner, K., & Stone, P. (2005). Multiagent traffic management: An improved intersection control mechanism. In F. Dignum, V. Dignum, S. Koenig, S. Kraus, M. P. Singh, & M. Wooldridge (Eds.), *The 4th International Joint Conference on Autonomous Agents and Multiagent Systems*, July 2005. New York: ACM Press.
28. Dresner, K., & Stone, P. (2006). Multiagent traffic management: Opportunities for multiagent learning. In K. Tuyls, P. J. Hoen, K. Verbeeck, & S. Sen (Eds.), *LAMAS 2005*, Lecture notes in artificial intelligence (Vol. 3898, pp. 129–138). Berlin: Springer Verlag.
29. Dresner, K., & Stone, P. (2007). Sharing the road: Autonomous vehicles meet human drivers. In *The 20th International Joint Conference on Artificial Intelligence*, January 2007 (pp. 1263–1268). Hyderabad, India.
30. Elhadouaj, S., Drogoul, A., & Espié, S. (2000). How to combine reactivity and anticipation: The case of conflicts resolution in a simulated road traffic. In *Proceedings of the Multiagent Based Simulation (MABS)* (pp. 82–96). New York: Springer.
31. France, J., & Ghorbani, A. A. (2003). A multiagent system for optimizing urban traffic. In *Proceedings of the IEEE/WIC International Conference on Intelligent Agent Technology* (pp. 411–414). Washington, DC: IEEE Computer Society.
32. Gartner, N. H. (1983). OPAC—A demand-responsive strategy for traffic signal control. *Transportation Research Record, 906*, 75–81.
33. Gershenson, C. (2005). Self-organizing traffic lights. *Complex Systems, 16*(1), 29–53.
34. Gordon, D. (1996). The organization of work in social insect colonies. *Nature, 380*, 121–124.
35. Hall, R. W. (Ed.). (2003). *Handbook of transportation science* (2nd ed.). Dordrecht: Kluwer Academic Pub.
36. Haugeneder, H., & Steiner, D. (1993). MECCA/UTS: A multi-agent scenario for cooperation in urban traffic. In *Proceedings of the Special Interest Group on Cooperating Knowledge Based Systems* (pp. 83–98). Keele, UK.
37. Helbing, D., & Huberman, B. A. (1998). Coherent moving states in highway traffic. *Nature, 396*, 738.
38. Henry, J., Farges, J. L., & Tuffal, J. (1983). The PRODYN real time traffic algorithm. In *Proceedings of the International Federation of Automatic Control (IFAC) Conference*, Baden-Baden: IFAC.
39. Hu, J., & Wellman, M. P. (1998). Multiagent reinforcement learning: Theoretical framework and an algorithm. In *Proceedings of the 15th International Conference on Machine Learning* (pp. 242–250). Los Altos: Morgan Kaufmann.
40. Hunt, P. B., Robertson, D. I., Bretherton, R. D., & Winton, R. I. (1981). *SCOOT—A traffic responsive method of coordinating signals*. TRRL Lab. Report 1014, Transport and Road Research Laboratory, Berkshire, 1981.
41. Klügl, F., & Bazzan, A. L. C. (2004), Simulated route decision behaviour: Simple heuristics and adaptation. In R. Selten & M. Schreckenberg (Eds.), *Human behaviour and traffic networks* (pp. 285–304). New York: Springer.
42. Klügl, F., Bazzan, A. L. C., & Wahle, J. (2003). Selection of information types based on personal utility—A testbed for traffic information markets. In *Proceedings of the 2nd International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)* (pp. 377–384). Melbourne, Australia: ACM Press.
43. Köhler, E., Möhring, R. H., & Wünsch, G. (2004). Minimizing total delay in fixed-time controlled traffic networks. In H. Fleuren, D. den Hertog, & P. Kort (Eds.), *Proceedings of Operations Research (OR), Operations Research Proceedings* (p. 192), Tilburg: Springer.
44. Kosonen, I. (2003). Multi-agent fuzzy signal control based on real-time simulation. *Transportation Research C, 11*(5), 389–403.
45. Leutzbach, W. (1988). *Introduction to the theory of traffic flow*. Berlin: Springer.

46. Littman, M. L. (1994). Markov games as a framework for multi-agent reinforcement learning. In *Proceedings of the 11th International Conference on Machine Learning, ML* (pp. 157–163). New Brunswick, NJ: Morgan Kaufmann.
47. Lowrie, P. (1982). The Sydney coordinate adaptive traffic system—Principles, methodology, algorithms. In *Proceedings of the International Conference on Road Traffic Signalling*, Sydney, Australia.
48. Möhring, R. H., Nökel, K., & Wünsch, G. (2006). A model and fast optimization method for signal coordination in a network. In *Proceedings of the 11th IFAC Symposium on Control in Transportation Systems*, August 2006. Delft, The Netherlands.
49. Moore, A. W., & Atkeson, C. G. (1993). Prioritized sweeping: Reinforcement learning with less data and less time. *Machine Learning, 13*, 103–130.
50. Morgan, J. T., & Little, J. D. C. (1964). Synchronizing traffic signals for maximal bandwidth. *Operations Research, 12*, 897–912.
51. Nagel, K., & Schreckenberg, M. (1992). A cellular automaton model for freeway traffic. *Journal de Physique I, 2*, 2221.
52. Nunes, L., & Oliveira, E. C. (2004). Learning from multiple sources. In N. Jennings, C. Sierra, L. Sonenberg, & M. Tambe (Eds.), *Proceedings of the 3rd International Joint Conference on Autonomous Agents and Multi Agent Systems, AAMAS*, July 2004 (Vol. 3, pp. 1106–1113). New York: IEEE Computer Society.
53. Oliveira, D., & Bazzan, A. L. C. (2006). Traffic lights control with adaptive group formation based on swarm intelligence. In M. Dorigo, L. M. Gambardella, M. Birattari, A. Martinoli, R. Poli, & T. Stuetzle (Eds.), *Proceedings of the 5th International Workshop on Ant Colony Optimization and Swarm Intelligence, ANTS 2006*, Lecture notes in computer science, September 2006 (pp. 520–521). Berlin: Springer.
54. Oliveira, D., Bazzan, A. L. C., & Lesser, V. (2005). Using cooperative mediation to coordinate traffic lights: A case study. In *Proceedings of the 4th International Joint Conference on Autonomous Agents and Multi Agent Systems (AAMAS)*, July 2005 (pp. 463–470). New York: IEEE Computer Society.
55. Oliveira, D., Ferreira, P. R., Jr., Bazzan, A. L. C., & Klügl, F. (2004). A swarm-based approach for selection of signal plans in urban scenarios. In *Proceedings of 4th International Workshop on Ant Colony Optimization and Swarm Intelligence—ANTS 2004*, Lecture notes in computer science (Vol. 3172, pp. 416–417). Berlin, Germany.
56. Oppenheim, N. (1995). *Urban travel demand modeling: From individual choices to general equilibrium*. New York, NY: Wiley.
57. Ossowski, S., Fernández, A., Serrano, J. M., Pérez-de-la-Cruz, J. L., Belmonte, M. V., Hernández, J. Z., et al. (2005). Designing multiagent decision support systems for traffic management. In F. Klügl, A. L. C. Bazzan, & S. Ossowski (Eds.), *Applications of agent technology in traffic and transportation*, Whitestein series in software agent technologies and autonomic computing (pp. 51–67). Basel: Birkhäuser.
58. Panait, L., & Luke, S. (2005). Cooperative multi-agent learning: The state of the art. *Autonomous Agents and Multi-Agent Systems, 11*(3), 387–434.
59. Papageorgiou, M. (2003). Traffic control. In R. W. Hall (Ed.), *Handbook of transportation science* (Chap. 8, pp. 243–277). Dordrecht: Kluwer Academic Pub.
60. Papageorgiou, M., Diakaki, C., Dinopoulou, V., Kotsialos, A., & Wang, Y. (2003). Review of road traffic control strategies. *Proceedings of the IEEE, 91*(12), 2043–2067.
61. Paruchuri, P., Pullalarevu, A. R., & Karlapalem, K. (2002). Multi agent simulation of unorganized traffic. In *Proceedings of the 1st International Joint Conference on Autonomous Agents and Multi-Agent Systems (AAMAS)* (Vol. 1, pp. 176–183). Bologna, Italy: ACM Press.
62. Rigolli, M., & Brady, M. (2005). Towards a behavioural traffic monitoring system. In F. Dignum, V. Dignum, S. Koenig, S. Kraus, M. P. Singh, & M. Wooldridge (Eds.), *Proceedings of the 4th International Joint Conference on Autonomous Agents and Multiagent Systems* (pp. 449–454). New York: ACM Press.
63. Robertson, D. I. (1969). *TRANSYT: A traffic network study tool*. Rep. LR 253, Road Res. Lab., London.
64. Robinson, G. E. (1992). Regulation of division of labor in insect societies. *Annual Review of Entomology, 37*, 637–665.
65. Rochner, F., Prothmann, H., Branke, J., Müller-Schloer, C., & Schmeck, H. (2006). An organic architecture for traffic light controllers. In C. Hochberger & R. Liskowsky (Eds.), *Proceedings of the Informatik 2006—Informatik für Menschen*, Lecture notes in informatics (Vol. P-93, pp. 120–127). Berlin: Köllen Verlag.
66. Roess, R. P., Prassas, E. S., & McShane, W. R. (2004). *Traffic engineering*. Englewood Cliffs, NJ: Prentice Hall.
67. Rossetti, R., & Liu, R. (2005). A dynamic network simulation model based on multi-agent systems. In F. Klügl, A. L. C. Bazzan, & S. Ossowski (Eds.), *Applications of agent technology in traffic and transportation*, Whitestein series in software agent technologies and autonomic computing (pp. 181–192). Basel: Birkhäuser.

68. Rossetti, R. J. F., Bordini, R. H., Bazzan, A. L. C., Bampi, S., Liu, R., & Van Vliet, D. (2002). Using BDI agents to improve driver modelling in a commuter scenario. *Transportation Research Part C: Emerging Technologies, 10*(5–6), 47–72.
69. Shoham, Y., Powers, R., & Grenager, T. (2003). Multi-agent reinforcement learning: A critical survey. Unpublished survey.
70. Shoham, Y., Powers, R., & Grenager, T. (2007). If multi-agent learning is the answer, what is the question? *Artificial Intelligence, 171*(7), 365–377.
71. Silva, B. C. d., Basso, E. W., Bazzan, A. L. C., & Engel, P. M. (2006). Dealing with non-stationary environments using context detection. In W. W. Cohen & A. Moore (Eds.), *Proceedings of the 23rd International Conference on Machine Learning ICML*, June 2006 (pp. 217–224). New York, ACM Press.
72. Silva, B. C. d., Oliveira, D. d., Bazzan, A. L. C., & Basso, E. W. (2006). Adaptive traffic control with reinforcement learning. In A. L. C. Bazzan, B. Chaib-Draa, F. Klügl, & S. Ossowski (Eds.), *Proceedings of the 4th Workshop on Agents in Traffic and Transportation (at AAMAS 2006)*, May 2006 (pp. 80–86). Hakodate, Japan.
73. Steingrover, M., Schouten, R., Peelen, S., Nijhuis, E., & Bakker, B. (2005). Reinforcement learning of traffic light controllers adapting to traffic congestion. In K. Verbeeck, K. Tuyls, A. Nowé, B. Manderick, & B. Kuijpers (Eds.), *Proceedings of the 17th Belgium-Netherlands Conference on Artificial Intelligence (BNAIC 2005)*, October 2005 (pp. 216–223). Brussels, Belgium: Koninklijke Vlaamse Academie van Belie voor Wetenschappen en Kunsten.
74. Stone, P. (2007). Learning and multiagent reasoning for autonomous agents. In *The 20th International Joint Conference on Artificial Intelligence*, January 2007 (pp. 13–30).
75. Stone, P. (2007). Multiagent learning is not the answer. It is the question. *Artificial Intelligence, 171*(7), 402–405.
76. Sutton, R. S. (1990). Integrated architectures for learning, planning, and reacting based on approximating dynamic programming. In *Proceedings of the 7th International Conference on Machine Learning* (pp. 216–224). Austin, Texas.
77. Tan, M. (1993). Multi-agent reinforcement learning: Independent vs. cooperative agents. In *Proceedings of the 10th International Conference on Machine Learning (ICML 1993)*, June 1993 (pp. 330–337). Los Altos, CA: Morgan Kaufmann.
78. TRANSYT-7F. (1988). *TRANSYT-7F user's manual*. Transportation Research Center, University of Florida.
79. Tumer, K., Welch, Z. T., & Agogino, A. (2008). Aligning social welfare and agent preferences to alleviate traffic congestion. In L. Padgham, D. Parkes, J. Müller, & S. Parsons (Eds.), *Proceedings of the 7th International Conference on Autonomous Agents and Multiagent Systems*, May 2008 (pp. 655–662). Estoril: IFAAMAS.
80. Tumer, K., & Wolpert, D. (2004). A survey of collectives. In K. Tumer & D. Wolpert (Eds.), *Collectives and the design of complex systems* (pp. 1–42). New York: Springer.
81. Tuyls, K. (2004). *Learning in multi-agent systems, an evolutionary game theoretic approach*. PhD thesis, Vrije Universiteit Brussel.
82. Tuyls, K., Hoen, P. J., & Vanschoenwinkel, B. (2006). An evolutionary dynamical analysis of multi-agent learning in iterated games. *Autonomous Agents and Multiagent Systems, 12*(1), 115–153.
83. Tuyls, K., & Parsons, S. (2007). What evolutionary game theory tells us about multiagent learning. *Artificial Intelligence, 171*(7), 406–416.
84. van Katwijk, R. T., van Koningsbruggen, P., Schutter, B. D., & Hellendoorn, J. (2005). A test bed for multi-agent control systems in road traffic management. In F. Klügl, A. L. C. Bazzan, & S. Ossowski (Eds.), *Applications of agent technology in traffic and transportation*, Whitestein series in software agent technologies and autonomic computing (pp. 113–131). Basel: Birkhäuser.
85. Verbeeck, K., Nowé, A., Peeters, M., & Tuyls, K. (2005). Multi-agent reinforcement learning in stochastic games and multi-stage games. In D. K. et al. (Eds.), *Adaptive agents and MAS II*, LNAI (Vol. 3394, pp. 275–294). Berlin: Springer.
86. Vu, T., Powers, R., & Shoham, Y. (2006). Learning against multiple opponents. In H. Nakashima, M. P. Wellman, G. Weiss, & P. Stone (Eds.), *Proceedings of the 5th International Joint Conference on Autonomous Agents and Multiagent Systems* (pp. 752–760). Hakodate, Japan.
87. Wahle, J., Bazzan, A. L. C., & Kluegl, F. (2002). The impact of real time information in a two route scenario using agent based simulation. *Transportation Research Part C: Emerging Technologies, 10*(5–6), 73–91.
88. Wahle, J., Bazzan, A. L. C., Klügl, F., & Schreckenberg, M. (2000). Anticipatory traffic forecast using multi-agent techniques. In D. Helbing, H. J. Hermann, M. Schreckenberg, & D. Wolf (Eds.), *Traffic and granular flow* (pp. 87–92). New york: Springer.

89. Wahle, J., Bazzan, A. L. C., Klügl, F., & Schreckenberg, M. (2000). Decision dynamics in a traffic scenario. *Physica A, 287*(3–4), 669–681.
90. Wardrop, J. G. (1952). Some theoretical aspects of road traffic research. In *Proceedings of the Institute of Civil Engineers* (Vol. 2, pp. 325–378). UK.
91. Watkins, C. J. C. H., & Dayan, P. (1992). Q-learning. *Machine Learning, 8*(3), 279–292.
92. Wiering, M. (2000). Multi-agent reinforcement learning for traffic light control. In *Proceedings of the 17th International Conference on Machine Learning (ICML 2000)* (pp. 1151–1158). Stanford.