Planning Autonomous Vehicles in the Absence of Speed Lanes Using an Elastic Strip

Rahul Kala and Kevin Warwick

***Abstract*—Planning of autonomous vehicles in the absence of speed lanes is a less-researched problem. However, it is an important step toward extending the possibility of autonomous vehicles to countries where speed lanes are not followed. The advantages of having nonlane-oriented traffic include larger traffic bandwidth and more overtaking, which are features that are highlighted when vehicles vary in terms of speed and size. In the most general case, the road would be filled with a complex grid of static obstacles and vehicles of varying speeds. The optimal travel plan consists of a set of maneuvers that enables a vehicle to avoid obstacles and to overtake vehicles in an optimal manner and, in turn, enable other vehicles to overtake. The desired characteristics of this planning scenario include near completeness and near optimality in real time with an unstructured environment, with vehicles essentially displaying a high degree of cooperation and enabling every possible (safe) overtaking procedure to be completed as soon as possible. Challenges addressed in this paper include a (fast) method for initial path generation using an elastic strip, (re-)defining the notion of completeness specific to the problem, and inducing the notion of cooperation in the elastic strip. Using this approach, vehicular behaviors of overtaking, cooperation, vehicle following, obstacle avoidance, etc., are demonstrated.**

***Index Terms*—Cooperative systems, intelligent vehicles, motion analysis, multirobot systems.**

Glossary

For any general vehicle R_i (vehicle being planned denoted by Q, with all variables indexed q)

$L_i(x'_i, y'_i, \theta'_i)$	Position (X', Y', orientation).
ΔL_i	Uncertainty.
v_i	Linear speed.
vpref_i	Preferred linear speed.
Δv_i	Uncertainty.
ω_i	Angular speed.
$\omega_i^{\max}$	Maximum angular speed.
$\mathrm{acc}_i^{\max}$	Maximum acceleration.
agg_i	Aggression factor.
$\zeta_{\mathrm{free}}^{\mathrm{static}}$	Free workspace.

For trajectory τ

$\tau(t)$	Planned position at time t.
τ_{obs}	Trajectory considering only static obstacles.
$\Delta\tau_{\mathrm{obs}}$	Deviation between τ and τ_{obs}.
τ_{strat}	Strategy (+1/left or −1/right) to avoid obstacles.
$\tau_{\mathrm{strat}}(i)$	Strategy to overcome obstacle i.
T	Time until which motion is planned.
$\tau(T)\{\tau(T)[X'], \tau(T)[Y'], \tau(T)[\theta']\}$	Last planned position.
d_{unc}	Minimum distance to maintain to overcome uncertain speed changes of vehicles.
d_o	Minimum distance to overcome a static obstacle ahead.
$d_o^{\min}$	Minimum distance to be maintained from the static obstacle while overcoming it.

For any general state s considered in planning

$[v_b^s, v_a^s]$	Speed bounds for safe travel.
R	Set of vehicles considered for feasibility.
$(d_f, d_{\mathrm{ls}}, d_{\mathrm{rs}}, d_{\mathrm{fl}}, d_{\mathrm{fr}}, d_b)$	Distance from obstacle (ahead, left, right, forward left, forward right, back).
(p_f, p_s, p_d, p_b, LP)	(forward, side, diagonal, back, lateral) potential.
$\mathrm{sen}_{X'}, \mathrm{sen}_{Y'}, \mathrm{sen}_{X'Y'}, \mathrm{coop}$	LP parameters denoting sensitivities along axes/cooperation.

For any general waypoint τ'_i in trajectory

τ'_0	Initial position/waypoint.
τ_{obs}^i	Point in τ_{obs} closest to τ'_i.
$(F_l, F_s, F_{\mathrm{coop}}, F_o, F_{\mathrm{total}})$	(lateral, spring extension, cooperation, drift, total) force.
$k_l, k_s, k_{\mathrm{coop}}, k_o$	Optimization parameters denoting contributions of each force.

Manuscript received February 7, 2013; revised April 12, 2013 and May 30, 2013; accepted May 31, 2013. Date of publication June 28, 2013; date of current version November 26, 2013. The work of R. Kala was supported by the Commonwealth Scholarship and Fellowship Plan (2010), U.K. under Award INCS-2010-161. An earlier version of this paper was presented at the 2012 Intelligent Vehicles Symposium, Alcalá de Henares, Spain, June 3–7. The Associate Editor for this paper was M. Vaizquez.

The authors are with the School of Systems Engineering, University of Reading, Reading, RG6 6AY U.K. (e-mail: rkala001@gmail.com; k.warwick@reading.ac.uk).

Color versions of one or more of the figures in this paper are available online at http://ieeexplore.ieee.org.

Digital Object Identifier 10.1109/TITS.2013.2266355

I. Introduction

THE problem of planning autonomous vehicles deals with all aspects of decision-making, which include selecting the route to reach the goal [1], selecting the manner of avoiding obstacles and other vehicles [2], generating a trajectory for motion [3], determining the lane of travel [4], dealing with special incidents and blockages [5], etc. Planning may be different for scenarios of straight roads, junctions, intersections, diversions, etc., [6], [7]. The specific problem addressed in this paper is planning for straight roads.

In general, navigation in most countries consists of vehicles that are laterally organized within speed lanes. The advantages of organized traffic are a higher degree of safety, clearer intentions of other vehicles, and fewer lane changes or lateral movements, signifying a more comfortable driving experience and, hence, shorter travel distances and times.

However, a number of countries allows unorganized traffic where vehicles may place themselves laterally anywhere in the road. In this scenario, the planner can construct trajectories, keeping the vehicle anywhere inside the road boundaries while varying the speed of travel. The advantages of this traffic system include higher traffic bandwidth and more overtaking (with each overtake signifying better travel for the vehicle that completed the maneuver). However, unorganized traffic has an apparent higher risk of accidents due to uncertain movements of other vehicles. Consider Indian traffic as an example. Bicycles, motor bikes, three-wheeled auto rickshaws, small cars, buses, overloaded trucks, hand-pulled vehicles, etc., all share the same road (see [8] and [9] for the details of the dynamics of this traffic).

This paper looks into the problem of navigating autonomous vehicles for nonlane-based traffic. The motivation is to move autonomous vehicles toward present-day unorganized traffic. Further, an increase in the number of autonomous vehicles may bring increased diversity to the currently organized traffic landscape, which may continually diminish the use of speed lanes up to the point when the entire traffic becomes unorganized.

Completeness is an essential requirement in the choice of the algorithm for solving any problem. This refers to the guarantee of the algorithm in finding a solution if a solution thus exists. However, it is usually possible to make reasonable assumptions to enable an algorithm to compute the results in a limited time. These algorithms are referred to as near complete as they "almost guarantee" the generation of a result, as long as the assumptions hold. Similarly, optimality is another essential requirement that refers to the guarantee of the algorithm to return the best solution if one thus exists. Algorithms of this type in which reasonable assumptions or approximations must be made are referred to as near optimal.

Elastic strips [10] have been used for the planning of a mobile robot. A number of homotopies may lead the robot from its source to the goal. The preferable homotopy is selected and represented as a strip that marks the robot's trajectory. For computational reasons, the strip is discretized to a number of waypoints. A change in the environment is marked by movement of these points, the addition of points, or deletion of points, such that the resulting trajectory is collision free. The intent is to have the least waypoints that are mostly near the obstacles. Each waypoint is acted on by a repulsive force from the obstacles, which makes the trajectory lie far from the obstacles. There is an additional internal force that pulls the waypoints toward each other, resulting in a shortening of the path. Given that two obstacles do not eventually intersect such that the robot is planned to travel in between them, the resulting travel plan is complete and reactive to real-time changes.

Given an initial optimal travel plan, the algorithm is near optimal to small changes in the environment. With some additional computational time, the algorithm is, as a result, better than the artificial potential field method [11], [12] and better than similar approaches that are neither complete nor optimal. The algorithm is in fact similar to the elastic roadmap [13], [14] and related approaches that maintain a roadmap in a dynamic environment.

An earlier version of this paper was presented at the 2012 Intelligent Vehicles Symposium [15]. The paper used lateral potentials (LPs) for planning movement of a vehicle. This paper tackles two major limitations of that work. First, potential methods are neither near complete nor near optimal. This paper solves both problems. Second, the paper answers the questions of deciding on a strategy of avoiding obstacles; this was done earlier using heuristics, which could be problematic in many situations.

The key contributions of this paper are: 1) providing the design of a method to quickly compute the optimal strategy for obstacle avoidance and the resulting trajectory; 2) providing real-time optimization of the trajectory as the vehicle moves, making the resulting plan near optimal; 3) using heuristics to ensure the travel plan is near complete; 4) providing the design of the algorithm to allow cooperation between vehicles; 5) enabling vehicles to travel at near-optimal speeds; 6) displaying complex behaviors of overtaking, vehicle following, obstacle avoidance, cooperation, etc., in a multivehicle scenario.

II. Related Research

Most studies in this area are based on organized or lane-oriented traffic. We list here some notable research on unorganized traffic or research that is extendable to unorganized traffic. Kuwata *et al.* [16] used rapidly exploring random trees with a biased sampling technique for the planning of a single vehicle. The algorithm, however, lacks global optimality. Treating vehicles and obstacles alike can lead to loss of completeness. Anderson *et al.* [17] solved the problem of trajectory generation for a single vehicle using constrained Delaunay triangles for a structured environment. Chu *et al.* [18] constructed a number of candidate paths from which the best path was selected. The strategy can be used for avoiding obstacles by a single maneuver only. Other limitations of these algorithms include noncooperative vehicles and the possibility of a vehicle steer causing a collision with another vehicle to the rear in a diverse vehicle speed scenario. In lane-based travel, this is equivalent to the fact that a (slower) vehicle may suddenly change lanes and push in front of a (faster) vehicle, thus requiring sudden braking.

Kala and Warwick [19] designed a multilevel graph search algorithm for the planning of multiple vehicles based on the assumption that vehicles were connected via a communication framework. Layers corresponded to route planning, obstacle avoidance strategy computation, vehicle coordination (placement), and trajectory generation. However, the resulting algorithm was not scalable. The application of elastic bands for the problem can be found in [20], where it was used to model vehicle-following behavior. The band was attached to the vehicle being followed. The approach is clearly not extendable to overtaking. Moreover, the strategy used for obstacle avoidance for the leading vehicle may not be the same for the following vehicle.

While the notion of cooperation seems to be a challenge for nonlane-based navigation algorithms, for lane-based traffic, a comparative study for cooperative overtaking was shown by Frese and Beyerer [21]. The authors studied mixed-integer programming, tree search, elastic bands, random priorities, and optimized priorities. However, a direct implementation in nonlane-based environments can be computationally expensive. Kala and Warwick [22] designed a set of discrete behaviors for the cooperative navigation of a vehicle in unorganized traffic. The discrete nature of the definitions was a limitation as the transitions could sometimes be rough.

For planning in a lane-based system, automaton techniques are widely used. Furda and Vlacic [23] modeled the problem using deterministic state automata with multicriterion decision-making. Schubert *et al.* [24] used lane markings and distances from vehicles as inputs to decide the optimal lane of travel.

Interesting similar algorithms exist in the domain of planning for mobile robots. Baxter *et al.* [25] used the artificial potential field method for planning multiple robots. The authors enabled robots to share potentials to rectify environmental perceptions. Gayle *et al.* [26] modeled the cooperation by social potential, where different types of robots applied different potentials. The social interaction framework was displayed within an adaptive roadmap in [27]. This approach is cooperative, near real time, near optimal, and near complete (with the exception of robots mutually causing a deadlock or congestion). These and similar reactive approaches are not applicable for a narrow-road-like structure where the robot may be found oscillating within the road, not allowing possible overtaking (where two robots are symmetrically ahead and behind each other), or poorly allowing possible overtaking (where two robots are almost ahead and behind each other). Further, unlike mobile robots, any turn or lateral movement (lane change) in a vehicle scenario threatens a collision with a vehicle to the rear.

Based on these works, it is clear that generalized planning algorithms for vehicles are noncooperative, and general lane-based approaches cannot be applied for nonlane-based travel. The presence of a narrowly bounded road structure, overtaking, and vehicle following as the primary underlying dynamics, and the unknown time and place of emergence of vehicles in a continuous traffic scenario make the problem of autonomous vehicle planning fundamentally different from multi-robot path planning. Any algorithm design needs to be validated against these differences and may possibly use this knowledge about the operational scenario as heuristics for computational speedup. Due to this heuristic, the algorithm presented here also performs better than any probabilistic sampling method that may explore much more or a graph search approach that may be computationally expensive for high-resolution maps, whereas (due to the nature of the problem) resolution across the lateral axis cannot be reduced.

III. Problem Definition

A limited map of the road is assumed to be given. It is considered that the road does not contain any junctions or diversions. The road is characterized by its left and right boundaries. It is assumed that all obstacles (including vehicles) can be sensed with some degree of certainty. Let any such general vehicle R_i be located at position $L_i(x'_i, y'_i, \theta'_i)$, where θ'_i denotes the heading direction, and (x'_i, y'_i) corresponds to the center, with the vehicle traveling at speed v_i. Let, at any general time, the vehicle being planned Q be at position $L_q(x'_q, y'_q, \theta'_q)$ with linear speed $v_q(\leq \text{vpref}_q)$ and angular speed $\omega_q(\leq \omega_q^{\max})$, where vpref_q is the preferred (maximum) speed of travel, and $\omega_q^{\max}$ is the maximum angular speed. The linear acceleration is bounded by $\text{acc}_q^{\max}$. All positions are denoted using the longitudinal (X') and lateral (Y') axis systems [15]. Let the free workspace be given by $\zeta_{\text{free}}^{\text{static}}$, which excludes any region with static obstacles or that are outside the road boundaries/road segment. Static obstacles and moving vehicles behave differently and are handled separately. Broken-down vehicles cannot move; hence, these are taken as static obstacles.

A. Objectives

The purpose of the algorithm is to construct a travel plan τ. Let $\tau(t)\{= (x'_q, y'_q, \theta'_q)\}$ be the planned position at time t, and let T denote the time up to which the vehicle is planned. The objectives, in (nonstrictly) decreasing order of importance, are as follows.

1) The vehicle should go as far as possible or should maximize $\tau(T)[X']$, which is the longitudinal position at the end of travel.
2) Maximize the minimum lateral clearance of the trajectory, where the lateral clearance is the minimum distance of the vehicle from any obstacle measured in the lateral axis.
3) Minimize T.
4) Maximize the lateral cooperation, i.e., the net lateral movement (measured in lateral axis) of a vehicle that is solely traversed with the intention of enabling some other vehicles to the rear to obtain a better plan.
5) In the case of overtaking an obstacle being equally advantageous (as per the aforementioned objectives) from both the left and right sides, the strategy used in [15] (see Section IV) is regarded as better.

B. General Speed Bounds

Consider a point-sized object at state s moving with speed $v_q(\leq \text{vpref}_q)$. Consider that obstacle i (in static obstacles and a pool of vehicles R) lies ahead of it at longitudinal distance d_f. The maximum speed v_a^s by which the object can move to avoid a collision with i is given by

$$v_a^s = \min\left(v_i + \sqrt{2 \cdot \text{acc}_q^{\max} \cdot \text{agg}_q \cdot d_f}, \text{vpref}_q\right). \quad (1)$$

v_d^s should be low enough to allow the object to stop before a collision takes place if there is no other alternative than following i, whereas it is assumed that i continues to travel with its current speed (v_i if a vehicle and 0 if an obstacle). This corresponds to slowing down with the maximum uniform retardation of $\text{agg}_q \cdot \text{acc}_q^{\max}$.

Here, $\text{agg}_q(0 < \text{agg}_q \leq 1)$ is the aggression factor that limits the planned acceleration. Lower values would indicate a more comfortable drive, whereas higher values sacrifice comfort for travel time. A minimum threshold distance of d_{unc} must always be maintained, which is excluded from the measured d_f. This is employed to overcome uncertain speed changes of the vehicle in front or other uncertain environment changes.

Further consider that obstacle i (in static obstacles and a pool of vehicles R) lies behind a point-sized object at a longitudinal distance of d_b. The minimum speed v_b^s by which the object can move to avoid collision with i is given by

$$v_b^s = \max\left(v_i - \sqrt{2 \cdot \mathrm{acc}_i^{\max} \cdot \mathrm{agg}_i \cdot d_b}, 0\right). \tag{2}$$

Using the concepts from (1), v_b^s should be high enough to allow stopping i traveling with speed v_i (0 if obstacle) before it collides with the object if i follows the object.

Hence, for safe travel, $v_b^s \le v_q \le v_a^s$.

C. Plan Feasibility

A driver only considers vehicles ahead of it while formulating his/her travel plan, as is the case in deciding the feasibility of travel plan τ. The only exception is making a turn (lane change) when one might accidently drive in front of a vehicle that may not have enough time to slow down to avoid collision. Hence, the resulting pool of vehicles considered for feasibility consists of all vehicles R_i, which either lie completely ahead of Q (or whose longitudinal coverage is completely ahead of the longitudinal coverage of Q) or do not lie in the same lane as Q (or whose lateral coverage is completely disjoint from the lateral coverage of Q). The set of vehicles is given by

$$R = \left\{ \begin{array}{l} R_i : L_i \otimes R_i[X'] > L_q \otimes Q[X'] \\ \vee L_i \otimes R_i[Y'] \cap L_q \otimes Q[Y'] = \phi \end{array} \right\}. \tag{3}$$

Plan τ is called a feasible travel plan if, first, no collisions occur with static obstacles or road boundaries, i.e.,

$$\tau(t) \otimes Q \in \zeta_{\mathrm{free}}^{\mathrm{static}} \qquad \forall t \le T. \tag{4}$$

Second, plan τ is called a feasible travel plan if no collisions occur with the projected motion of other vehicles, i.e.,

$$\tau(t) \otimes Q \notin \bigcup_{i, R_i \in R} ((L_i \pm \Delta L_i) + t(v_i \pm \Delta v_i)) \otimes R_i, \ \forall t \le T. \tag{5}$$

All vehicles are projected to travel straight (longitudinally), maintaining their current orientation, from the sensed initial position in the range of $(L_i - \Delta L_i, L_i + \Delta L_i)$, with a constant sensed speed in the range of $(v_i - \Delta v_i, v_i + \Delta v_i)$. A collision is said to have occurred if Q intersects with any projected position of the vehicle. Here, ΔL_i and Δv_i denote the uncertainty in measurements of position and speed of vehicle R_i.

Finally, at all times, the speeds are within the desirable bounds, i.e.,

$$v_b^s \le v_q \le v_a^s \qquad \forall s \in \tau(t) \otimes Q, \quad t \le T. \tag{6}$$

Checking speed bounds ensures that the trajectory being followed can be terminated at any instance (due to changed environment dynamics), and the vehicle can be made to drive straight ahead, giving enough time for every vehicle to adjust.

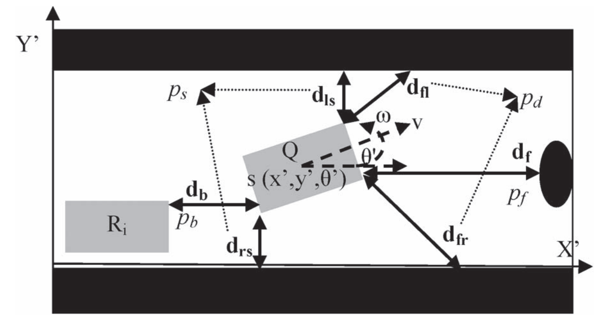

Fig. 1. Sources of potential. d_f, d_{ls}, d_{rs}, d_{fl}, d_{fr}, and d_b denote distance from ahead, left, right, forward-left, forward-right, and back obstacles, respectively. p_f, p_s, p_d, and p_b denote forward, side, diagonal, and back potentials, respectively.

IV. Planning Using LPs

Here, we summarize the approach used in [15]. The algorithm used an LP to compute the desirable angle to the longitudinal axis, based on which steering attempt was made (e.g., LP(s) for state s). The potential is computed from four sources, which are the forward potential, the side potential (the left side and the right side), the diagonal potential (forward left and forward right), and the back potential. The various sources are shown in Fig. 1.

The forward potential at point s is a reaction to seeing obstacle i longitudinally ahead (along the length of the road, irrespective of vehicle orientation) at a distance of d_f. In [15], a simple heuristic was used to decide on the direction. In the case of a static obstacle, the strategy was to turn right if the obstacle was more toward the left of the road and *vice versa*. In the case of a vehicle, the direction was to the right if the other vehicle was more laterally toward the left and *vice versa*. However, this heuristic is neither complete nor optimal. In the current approach, let τ_{strat} be a bookkeeping variable that directly returns the direction of turn $\tau_{\mathrm{strat}}(i)$ for obstacle i. Time to collision, instead of the commonly used distance, was used as a metric, whereas the potential was taken to be inversely proportional to the square of the metric (time to collision), which is given by

$$p_f = \begin{cases} 0, & v_i \ge \mathrm{vpref}_q \\ \tau_{\mathrm{strat}}(i) \left(\frac{\mathrm{vpref}_q - v_i}{d_f}\right)^2, & v_i < \mathrm{vpref}_q. \end{cases} \tag{7}$$

The side potential is computed using the free lateral distance on the left side d_{ls} and that on the right side d_{rs}. Each side applies a potential in the opposite direction given by

$$p_s = p_{\mathrm{ls}} + p_{\mathrm{rs}} = -\max\left\{(1/d_{\mathrm{ls}})\right\}^2 + \max\left\{(1/d_{\mathrm{rs}})\right\}^2. \tag{8}$$

The diagonal potential has magnitude indicated by the diagonal-free distances of forward left d_{fl} and forward right d_{fr}, whereas the directions are opposite to the side considered, which is given by

$$p_d = p_{\mathrm{fl}} + p_{\mathrm{fr}} = -(1/d_{\mathrm{fl}})^2 + (1/d_{\mathrm{fr}})^2. \tag{9}$$

The back potential is responsible for the cooperative behavior of the vehicle. In [15], the direction was given by the heuristic, whether the vehicle behind was more toward the left or right. Here, the same heuristic is used; however, cooperation is only applied when the direction of overtaking is clear. Similar to (7), the magnitude is indicated by the time to collision with vehicle i to the rear (if any), which is given by

$$p_b = \begin{cases} \left(\frac{v_i - \text{vpref}_q}{d_b}\right)^2, & v_i > \text{vpref}_q,\ L_i[Y'] < s[Y'] \\ -\left(\frac{v_i - \text{vpref}_q}{d_b}\right)^2, & v_i > \text{vpref}_q,\ L_i[Y'] > s[Y'] \\ 0, & \text{otherwise.} \end{cases} \tag{10}$$

$s[Y']$ denotes the Y' component of state s.

The potential directions are based on the $X'Y'$ coordinate axis system and are independent of vehicle orientation. Each potential is measured across all possible points, and the maximum potential recorded is used for computation of the resulting potential, which is given by

$$\text{LP} = \text{sen}_{X'} \cdot p_f + \text{sen}_{Y'} \cdot p_s + \text{sen}_{X'Y'} \cdot p_d + \text{coop} \cdot p_b \tag{11}$$

where sen_X, $\text{sen}_{X'Y'}$, and $\text{sen}_{Y'}$ denote the various sensitivities, and coop denotes the cooperation factor. Fig. 1 shows the sources corresponding to the maximum potential values.

V. Algorithm

Let the trajectory being followed at any time be τ. The algorithm additionally defines τ_{obs} to denote the trajectory constructed by considering only static obstacles. This term is only defined if the vehicle cannot compute a collision-free trajectory, which assures collision-free avoidance of static obstacles. τ_{strat} denotes the operational strategy and is a specification of direction (left or right) by which any obstacle need to be overcome. Initially, τ contains the immediate position only, whereas τ_{obs} and τ_{strat} are both null.

A. Plan Extender

Consider a travel plan τ (known to be feasible as per the current traffic scenario) constructed using strategy τ_{strat}. The task is to extend τ. The assumption is that the entire extended plan thereafter would be followed at prespecified speed v_q. The extension is carried using an LP (see Section IV) with the following differences.

1) The speed of travel is kept constant. The sampling frequency is taken to be inversely proportional to the speed indicated by the LP. The sampling time indicates the time span after which the steering action of the LP is applied. When the (projected) vehicle is nearer to obstacles, the speed indicated by the LP is smaller; hence, frequent LP actions are applied and *vice versa*. Every call to LP consumes computation. This methodology is hence adaptive to obstacle placements.
2) The strategy parameters appearing in Section VI are looked up in τ_{strat}. For every new obstacle witnessed (which does not have an entry in τ_{strat}), all combinations of strategies (e.g., in Fig. 2) are separately computed, and the best one, as per the performance criterion set in Section III, is chosen.

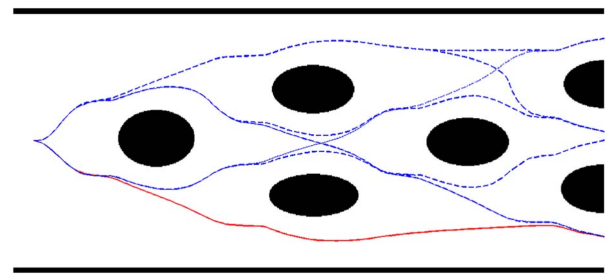

Fig. 2. Plan extension by every strategy. The blue (dashed) lines represent all possible strategy plans (only strategies experienced by the vehicle). The red (continuous) line shows the optimal plan.

At every step, the extended plan needs to be feasible as per the criterion stated in Section III. The plan extension may terminate for any of the following three reasons: 1) the end of a road segment being planned is reached; 2) it is no longer possible to move any further due to feasibility criterion due to a static obstacle; and 3) it is no longer possible to move any further due to the feasibility criterion of another vehicle.

The algorithm can be regarded as complete for the cases that end in the first reason. In the case of the third reason, the algorithm has an option to travel as per the computed plan and then to start following the vehicle in front. Human drivers can be reliably followed, whereas following an automated vehicle can be regarded as near complete if the vehicle ahead is following a near complete algorithm. Since the recurrence is applied to a forward vehicle, the planning algorithm due to termination because of the third reason can be regarded as complete.

In the case of termination due to the second reason, an assurance that the static obstacle is avoided at a later time is needed. Hence, an attempt is made to compute plan τ_{obs}, considering only static obstacles and all ways of avoiding them. Out of the range of different possibilities, the best plan is considered. While completeness in construction of τ_{obs} cannot be guaranteed, all possible ways of avoiding static obstacles are considered, and no other vehicle is considered; the algorithm may therefore be stated as near complete. It may be assumed that the vehicles ahead would eventually clear, making way for the vehicle to follow the plan τ_{obs}. In lieu of these points, the entire algorithm is near complete and (combined with Section V-B) near optimal.

Algorithm 1: Extend1$(\boldsymbol{\tau}, \boldsymbol{\tau}_{\text{strat}}, \boldsymbol{v}_q)$
While τ is feasible
  $\tau \leftarrow \tau \cup$ LP(τ[end], τ_{strat}, v_q)
  if τ[end] encounters obstacle i and $\tau_{\text{strat}}(i) = \text{null}$
    $\langle \tau^a, \tau^a_{\text{strat}} \rangle \leftarrow \text{Extend1}(\tau, \tau_{\text{strat}} \cup \langle i,\ \text{left} \rangle, v_q)$
    $\langle \tau^b, \tau^b_{\text{strat}} \rangle \leftarrow \text{Extend1}(\tau, \tau_{\text{strat}} \cup \langle i,\ \text{right} \rangle, v_q)$
    $\langle \tau, \tau_{\text{strat}} \rangle \leftarrow$ better than $\langle \tau^a, \tau^a_{\text{strat}} \rangle$ and $\langle \tau^b, \tau^b_{\text{strat}} \rangle$
    break while
  end if
end while
return $\langle \tau, \tau_{\text{strat}} \rangle$

Algorithm 2:Extend$(\boldsymbol{\tau}, \boldsymbol{\tau}_{\text{strat}}, \boldsymbol{v}_q)$
$\langle \tau, \tau_{\text{strat}} \rangle \leftarrow \text{Extend1}(\tau, \tau_{\text{strat}}, v_q)$ with all obstacles
if τ ends with static obstacle
  $\langle \tau_{\text{obs}}, \tau_{\text{stratObs}} \rangle \leftarrow \text{Extend1}(\tau, \tau_{\text{strat}}, v_q)$ with only static obstacles
  $\langle \tau, \tau_{\text{strat}} \rangle \leftarrow \text{Extend1}(\tau, \tau_{\text{stratObs}}, v_q)$ with all obstacles
end if
return $\langle \tau, \tau_{\text{strat}}, \tau_{\text{obs}} \rangle$

The extension is hence given by Algorithms 1 and 2. Algorithm 1 extends a plan consisting of τ and τ_{strat}. Every time a new obstacle is discovered, both avoidance from the left and avoidance from the right are tried, and the best is retained. For obstacles encountered previously, the strategy indicated by τ_{strat} is used. Algorithm 2 computes τ_{obs}, in the case when τ, as computed by Algorithm 1, ends in a static obstacle. τ_{stratObs} denotes the strategy to avoid static obstacles. Extend1 is called again to make τ and τ_{strat} consistent with the strategy used in τ_{obs}, i.e., τ_{stratObs}. In implementation, all strategies may be stored in line 1 of Extend$(\cdot)$ and fetched in line 4 instead of a new function call.

B. Plan Optimizer

Consider travel plan τ, which as per the current traffic scenario, needs to be followed with a prespecified travel speed of v_q. The travel plan is optimized as the vehicle moves (and scenarios change). Plan τ is converted into a set of coarsely located waypoints τ', vaguely representing plan τ. The optimization of τ' is based on the analogy of a spring, with each waypoint τ'_i representing a virtual vehicle with a movable clamp. τ'_i is modeled as a clamp attached to the lateral axis Y'. Hence, by the application of forces, τ'_i can move along Y' but not along X'. The initial position τ'_0 is fixed. This constraint disallows two waypoints to come close to each other, which may slow down the optimization process. τ'_i is influenced by the following four forces.

1) *Lateral force.* The force is applied by obstacles that are laterally left and laterally right in opposing directions, and magnitude F_l is given by (8).
2) *Spring extension force:* Each way point τ'_i is attracted by the waypoint ahead τ'_{i+1} (if any) and behind τ'_{i-1} (if any) with a force proportional to the extension given by

$$F_s = \left(\left\|\tau'_{i+1} - \tau'_i\right\| - \left(\tau'_{i+1}[X'] - \tau'_i[X']\right)\right) \cdot u\left(\tau'_{i+1} - \tau'_i\right) + \left(\left\|\tau'_{i-1} - \tau'_i\right\| - \left(\tau'_i[X'] - \tau'_{i-1}[X']\right)\right) \cdot u\left(\tau'_{i-1} - \tau'_i\right). \tag{12}$$

Here, $u(x)$ denotes the unit vector in the direction of x. Two points clamped to their lateral axis can have a minimal separation equal to their longitudinal separation. Any separation in excess is considered as an extension.
3) *Cooperation force:* Plan τ may initially be made by only considering the vehicles in the scenario. Additional vehicles may appear later at the rear, and they might then aim to overtake. Extension of τ does not account for cooperation in overtaking; hence, the same is modeled in optimization. Force F_{coop} is the same as that given by (10).
4) τ_{obs} *drift force:* A nonnull value of τ_{obs} indicates that the current plan τ cannot overcome a static obstacle, whereas plan τ_{obs} can overcome the static obstacle subjected to the absence of the other vehicles. Following τ would mean ending up close to the static obstacle and then having to steeply steer to avoid it, whenever feasible. Following τ_{obs} by waiting for the vehicles to clear and, at every step, computing the highest possible speed may mean excessive slowing down initially or for a large part of the journey. An attempt is made to induce advantages of both the techniques by following τ, but slowly drifting it toward τ_{obs}. Force F_o is proportional to the distance between the closest waypoint in τ_{obs} (denoted τ^i_{obs}) applied in same direction. This is given by

$$F_o = \left\|\tau^i_{\text{obs}} - \tau'_i\right\| \cdot u\left(\tau^i_{\text{obs}} - \tau'_i\right). \tag{13}$$

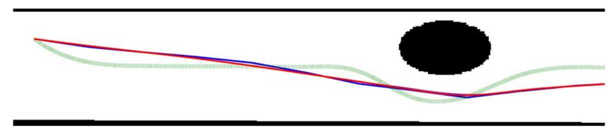

Fig. 3. Optimization of the plan. The blue line represents the initial plan, which after optimization is given by the red line. If optimized with the aim of maximizing average clearance, the plan is given by the green (dashed) line.

The total force is given by

$$F_{\text{total}} = k_l \cdot F_l + k_s \cdot F_s + k_{\text{coop}} \cdot F_{\text{coop}} + k_o \cdot F_o. \tag{14}$$

Here, k_l, k_s, k_{coop}, and k_o are the associated weights of the different factors.

The lateral component of F_{total} is used for deviating τ'_i. Only changes resulting in a feasible plan are admitted. For a sample path, the optimization is shown in Fig. 3.

C. Complete Framework

The basic hypothesis behind the algorithm is simple. Compute the highest speed v_q that the vehicle can have as per the current scenario (step A), use v_q to trim (step B), extend/construct the plan (step C) such that the resulting plan is feasible, and optimize the plan (step D). The vehicle, at any time, may have two modes of operation, which are Mode I traveling, with a plan ending at some static obstacle ($\tau_{\text{obs}} =$ null), and Mode II traveling, with a plan not ending at some static obstacle ($\tau_{\text{obs}} \neq$ null). Each of the steps is applied for both modes (denoted I A, I B, ..., II D), whereas switching between modes (denoted I $\rightarrow$ II and II $\rightarrow$ I) is monitored. These steps are applied as the vehicle moves and the scenarios change. Hence, at any time, the vehicle may be seen to show initial signs of reacting to any new obstacle (or an obstacle whose motion has changed a lot as per expectation), deliberating over later course of actions, adapting the plans to any changes in the scenario, and optimizing any previously suboptimal plan. The algorithm is summarized in Fig. 4 and Algorithm 3.

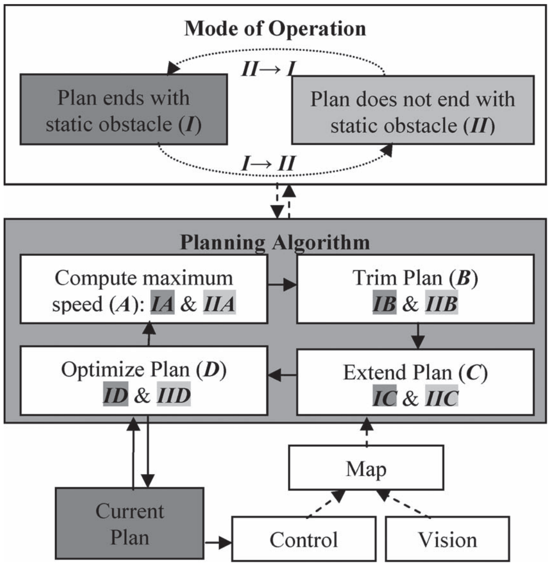

Fig. 4. General algorithm framework.

Algorithm 3: Plan$(\tau_{\mathrm{obs}}, \tau, v_q)$

I B	if (I) and τ_{obs} is not collision-free τ_{obs}=null, τ=null
II A	if (II), compute v_q using (1) with acceleration/ aggression limits
I/II B	trim τ till it is feasible
	if (I)
I C	$<\tau, \tau_{strat}> \leftarrow$ Extend1(τ, τ_{strat}, max(v_q, v_{min}))
I→II	if τ satisfies (II), τ_{obs}=null
	else
II C, II→I	$<\tau, \tau_{strat}, \tau_{obs}> \leftarrow$ Extend(τ, τ_{strat}, max(v_q, v_{min}))
	end if
I/II D	Optimize(τ, v_q)
I A	if $\tau_{obs} \neq$ null $v_q \leftarrow$ maximum speed to stop at $\|\tau\|$-d_{obs}
	return $<\tau_{obs}, \tau, v_q>$

We first discuss each of the four steps (A to D) for each mode of operation (I and II), and then discuss how these assemble into Algorithm 3. Step A is different for both modes I and II. In II A, speed v_q is the highest speed possible for the vehicle as per the feasibility plan, taking into account the acceleration limits, aggression factor, and deceleration limits. This update rule is formulated in realization of the fact that a fast-moving vehicle at a distance, on being unable to overtake a slow moving vehicle in front, would be found to exhibit high speedup to the point when it comes close to the slower vehicle, after which a nonaggressive (as prespecified) deceleration takes place. Hence, at every step, the vehicle displays maximum speed, which naturally plays a role in decreasing the total travel time.

An optimal travel plan consists of optimal trajectory computation and optimal speed settings. Working in the joint space of trajectory and speed is not possible due to computational and modeling constraints. However, by constant adjustment of speeds and trajectories separately, a near-optimal path can be obtained.

For I A, speeding up is disallowed, which arises from the commonly seen behavior that drivers tend to slow down upon seeing a static obstacle if a lane change is not allowed. It is expected that the vehicle would come to a standstill at a distance of d_{obs} before the static obstacle, where d_{obs} is the minimum longitudinal distance needed to overcome the static obstacle given by

$$d_{\mathrm{obs}} = d_{\mathrm{obs}}^{\mathrm{min}} + \Delta\tau_{\mathrm{obs}} \tag{15}$$

and accordingly, the speed is iteratively reduced.

Here, $d_{\mathrm{obs}}^{\mathrm{min}}$ is a small fixed distance to be maintained from the static obstacle at all times, and $\Delta\tau_{\mathrm{obs}}$ is the deviation between τ and τ_{obs}, which is taken as the distance between the last point in τ and the closest point in τ_{obs}. If the vehicle continues to follow τ, it is expected that it would have to stop at the end due to no possible subsequent moves and wait for the other vehicles to clear. The distance d_{obs} gives enough scope for a vehicle to turn to avoid the static obstacle, in the absence of other vehicles. If the vehicle is directly ahead of the static obstacle, a larger steer would be required as compared with the case when only a small part of the vehicle is ahead of the static obstacle. A larger steer implies a requirement of a greater longitudinal distance, which is modeled by $\Delta\tau_{\mathrm{obs}}$. In the case in which the vehicle actually stops or almost stops, the subsequent extend operation (to move the vehicle when path is clear) is called with a small predefined speed v_{min}.

Trimming without ending at static obstacles or II B is simply based on the feasibility of the plan τ as per the changed environment (if any) using the set feasibility constraints. For I B, this needs to be performed for both τ and τ_{obs}. An infeasible τ_{obs} (Algorithm 3, Line 1) implies emergence of a new static obstacle; in which case, everything needs to be invalidated and recomputed to ensure a collision-free τ_{obs}, whereas τ follows the same concepts as II B. In conception, *extend* or step C is the same for both modes; however, they are differently called in the algorithm to eliminate recomputation of τ_{obs}, which is already known. The *extend* algorithm also monitors for mode changes. Step D is the same for both modes.

From Algorithm 3, it is shown that the four steps are performed one after the other for mode II. Mode checking conditions are introduced if a step is particular to a mode. For mode I, the algorithm starts with I B and ends with I A through I D. I B (Algorithm 3, Line 1) is placed at the start since its action invalidates the plan, whereas I A is kept at the end since, in mode I, speed is trajectory dependent, although once initiated, it does not matter which was the first step of the loop and which was the last. An exception to the algorithm is when the vehicle is initially to be found placed in an infeasible manner, or when not even a single step is feasible; in which case, the immediate move is as directed by LP, with maximum possible deceleration of $-\mathrm{acc}_q^{\mathrm{max}}$.

VI. Results

A. Simulations

To test the algorithm, a number of diverse scenarios were constructed, each aimed at testing a different aspect of the algorithm. In all the experiments, vehicles are named on the order of their appearance. We have included a supplementary video

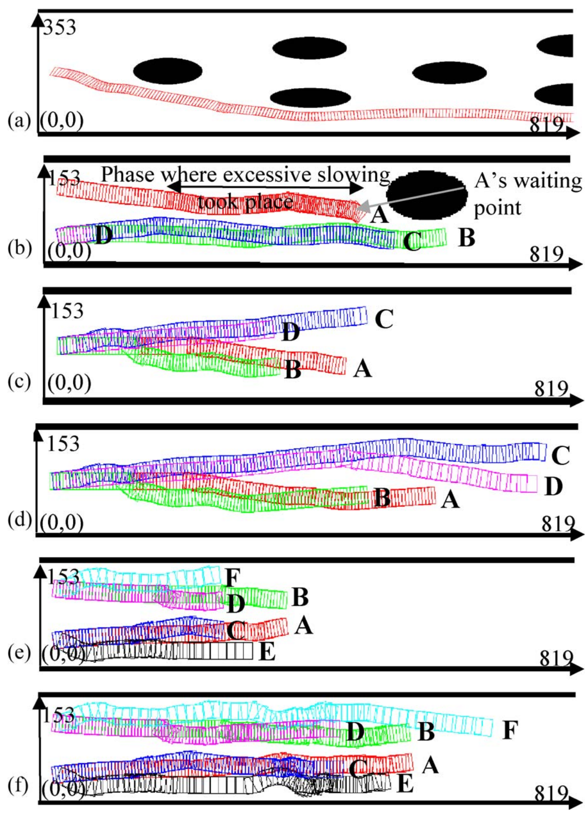

Fig. 5. Simulation results. (a) Computing the optimal path in a complex grid. (b) Slowing down to avoid a static obstacle while waiting for vehicles to clear. (c)–(d) Multiple overtaking, each from the optimal side. (e)–(f) Multiple overtakings, each after waiting for the correct time and in the right order.

file showing the results. This is available at http://ieeexplore.ieee.org. In all of the experiments, initially, the road segment is empty with only the static obstacles (if any). The vehicles are made to emerge at the positions and emergence time as visible in the results/supplementary video. All vehicles are generated in the direction of the road segment. Nothing is known about a vehicle by any other vehicle before it is visible in the scenario. In other words, the vehicles suddenly appear in the scenario requiring other vehicles to adjust their plans accordingly.

The first experiment tests the ability of the algorithm to optimally drive a vehicle within a grid of complex obstacles [see Fig. 5(a)]. An initial expectation was that the vehicle would take the central route; however, the extra maneuvers around the obstacles made the trajectory larger. The vehicle always maintained comfortable distances from all the obstacles while traveling with the maximum permissible speed.

Fig. 5(b) shows a different scenario where A could not overcome a static obstacle and hence had to slow down and wait for the other vehicles (B and C) to clear. At a later stage, it comfortably places itself before D. The scenario tests the algorithm's ability to restrain a vehicle from accidently pushing in front of another vehicle, which it may not be prepared for. Consider that D's entry is made much earlier; in which case, A would wait for it as well. If A is synthetically made to move, D would obediently follow. If D is synthetically made to slow down, A would not wait any longer. Hence, individually, the vehicles show expected behaviors; however, having already waited for two vehicles to pass through A would be called too courteous to be waiting for D, which the algorithm currently simulates.

The next scenario tests the ability of the vehicle to decide the optimal direction for overtaking. The overtaking vehicles have infeasible entry conditions, and the plans are made in multiple extension operations. C overtaking the two vehicles (A and B) in front is visible in Fig. 5(c), whereas it is likely to be overtaken itself by D. Fig. 5(d) shows how D overtakes C, whereas the other vehicles (A and B) themselves orient to enable the overtaking procedure to happen. All vehicles travel nearly at their maximum permissible speeds.

The last scenario is introduced to consider the traffic dynamics when overtaking is not initially possible and later is still rather difficult due to competing vehicles. E entered the scenario before F and hence overtook the initial set of vehicles (C and D) earlier [see Fig. 5(e)]. However, it appeared that there was no room for overtaking the vehicle set ahead (A and B); hence, E had to rapidly slow down to follow A ahead, whereas F succeeded in overtaking the initial set of vehicles (C and D). Subsequently, F occupied enough room and succeeded in overtaking the vehicle set ahead [A and B, see Fig. 5(f)]. Initially, F slowed down while space was being created, but later accelerated during overtaking. Henceforth, the vehicles drifted to the other side to accommodate overtaking E. It must be emphasized that the scenario is largely driven by LPs.

B. Parameters

The effects of the parameters of LP was demonstrated in [15]. The settings used in this paper, however, favor more sensitive settings, which led to the disadvantages of introducing oscillations and steep turns, whereas the advantages included the capability to avoid obstacles, however far or near. The limitations are however eliminated by the optimize algorithm, whereas the advantages remain.

Trajectory planning always involves the problem of a tradeoff between path length and average clearance. These factors are controlled by the parameters in the optimize algorithm. Experiments over a single obstacle scenario were carried out, for which the participating parameters are k_l and k_s. Both a large clearance and a short path length cannot be simultaneously achieved, which is clearly shown in Fig. 6. A large clearance leads to the vehicle quickly placing itself in the middle of the road and then between the obstacle and the road edge, whereas attempts to minimize the path length led to the vehicle traveling very close to the obstacle (see Fig. 3). Only parameter settings leading to feasible results are plotted. k_{coop} has a role similar to that presented in [15].

The worst-case complexity of the extend algorithm is $O(2^{(|O|+|R|)L})$, where $|O|$ represents the number of static obstacles ahead, $|R|$ represents the number of vehicles ahead with a speed lower than the vehicle being planned, and L is the length of the segment being planned. In the presented scenarios, computing the trajectory for each vehicle takes less than a second, whereas the initial plan generation can take 2.5 s if τ_{obs} also needs to be computed. Scenario 1 takes 8 s to compute in total due to the computation of a single trajectory overcoming all static obstacles.

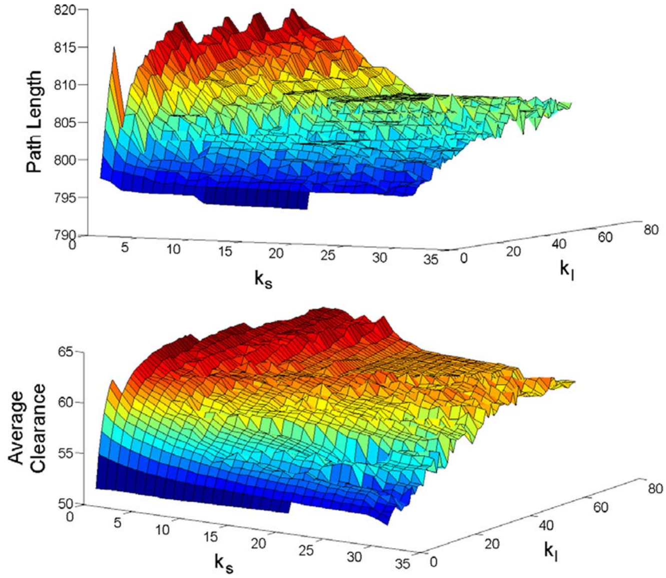

Fig. 6. Effect of change of algorithm parameters on obstacle avoidance. (a) Path length. (b) Average clearance.

The high complexity is not a factor of concern considering that the length of segment planned can be adjusted to account for the number of obstacles. Further, for a high number of obstacles, the algorithm may fail to find a single trajectory that simultaneously avoids all the obstacles, forcing the algorithm to terminate.

The algorithm does not (in initial plan construction) have the liberty to reduce the speed. The uncertainties in projecting a vehicle's future motion increase with the vehicle's speed and with longer projections into the future. High uncertainties show as elongated projected positions of the vehicles that may be hard to surpass. In this case, the algorithm would construct a trajectory to overcome the initial set of obstacles first, and as the vehicle moves, the trajectory would be modified to overcome latter obstacles.

Avoiding closer obstacles is easier due to the high degree of certainty in position and speed, and preferable speed settings as compared with the obstacles that are much further away. The inability to overtake a vehicle may however result in vehicle following.

VII. Conclusion

This paper has emphasized the notion of the unorganized nature of traffic to make travel efficient in a diverse transportation system. The problem of planning on a straight road was solved using the elastic strip concept. The algorithm was near complete and near optimal, whereas the computational cost was only larger than reactive techniques. Using the algorithm, the ability to maneuver in a complex obstacle framework was presented while showing complex overtaking, vehicle following, and waiting behaviors. This paper is hence a step toward bringing autonomy to currently unorganized traffic systems, while also providing a possible future alternative for currently organized traffic systems.

In the future, maximum lateral acceleration needs to be added in the feasibility definition to prevent trajectories being generated, which cannot be traced. The simulation framework also needs to be extended to a real city map, including intersections, diversions, crossings with traffic lights, etc. Currently, there are no data sets recording unorganized traffic movement to validate such algorithms, which may further give better intuition of the traffic. Many present algorithms for vehicle tracking, obstacle detection, localization, etc., are currently lane based, and these need to become nonlane based for these algorithms to be implemented on real roads.

References

[1] R. Claes, T. Holvoet, and D. Weyns, "A decentralized approach for anticipatory vehicle routing using delegate multiagent systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 12, no. 2, pp. 364–373, Jun. 2011.

[2] F. Jiménez and J. E. Naranjo, "Improving the obstacle detection and identification algorithms of a laserscanner-based collision avoidance system," *Transp. Res. Part C, Emerg. Technol.*, vol. 19, no. 4, pp. 658–672, Aug. 2011.

[3] S. Glaser, B. Vanholme, S. Mammar, D. Gruyer, and L. Nouveliere, "Maneuver-based trajectory planning for highly autonomous vehicles on real road with traffic and driver interaction," *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 589–606, Sep. 2010.

[4] R. W. Hall and C. Caliskan, "Design and evaluation of an automated highway system with optimized lane assignment," *Transp. Res. Part C, Emerg. Technol.*, vol. 7, no. 1, pp. 1–15, Feb. 1999.

[5] J. B. Sheu and S. G. Ritchie, "A new methodology for incident detection and characterization on surface streets," *Transp. Res. Part C, Emerg. Technol.*, vol. 6, no. 5/6, pp. 315–335, Dec. 1998.

[6] J. Lee and B. Park, "Development and evaluation of a cooperative vehicle intersection control algorithm under the connected vehicles environment," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 1, pp. 81–90, Mar. 2012.

[7] M. Sarvi and M. Kuwahara, "Microsimulation of freeway ramp merging processes under congested traffic conditions," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 3, pp. 470–479, Sep. 2007.

[8] D. Mohan and P. S. Bawa, "An analysis of road traffic fatalities in Delhi, India," *Accid. Anal. Prev.*, vol. 17, no. 1, pp. 33–45, Feb. 1985.

[9] L. Vanajakshi, S. C. Subramanian, and R. Sivanandan, "Travel time prediction under heterogeneous traffic conditions using global positioning system data from buses," *IET Intell. Transp. Syst.*, vol. 3, no. 1, pp. 1–9, Mar. 2009.

[10] O. Brock and O. Khatib, "Elastic strips: A framework for integrated planning and execution," in *Experimental Robotics VI*. New York, NY, USA: Springer-Verlag, 2002, pp. 329–338.

[11] O. Khatib, "Real-time obstacle avoidance for manipulators and mobile robots," in *Proc. IEEE Int. Conf. Robot. Autom.*, St. Louis, MO, USA, 1985, vol. 2, pp. 500–505.

[12] Y. K. Hwang and N. Ahuja, "A potential field approach to path planning," *IEEE Trans. Robot. Autom.*, vol. 8, no. 1, pp. 23–32, Feb. 1992.

[13] Y. Yang and O. Brock, "Elastic roadmaps-motion generation for autonomous mobile manipulation," *Autonom. Robots*, vol. 28, no. 1, pp. 113–130, Jan. 2010.

[14] S. Quinlan and O. Khatib, "Elastic bands: Connecting path planning and control," in *Proc. IEEE Int. Conf. Robot. Autom.*, 1993, pp. 802–807.

[15] R. Kala and K. Warwick, "Planning autonomous vehicles in the absence of speed lanes using lateral potentials," in *Proc. IEEE Intell. Veh. Symp.*, Alcala de Henares, Spain, Jun. 2012, pp. 597–602.

[16] Y. Kuwata, S. Karaman, J. Teo, E. Frazzoli, J. P. How, and G. Fiore, "Real-time motion planning with applications to autonomous urban driving," *IEEE Trans. Control Syst. Technol.*, vol. 17, no. 5, pp. 1105–1118, Sep. 2009.

[17] S. J. Anderson, S. B. Karumanchi, and K. Iagnemma, "Constraint-based planning and control for safe, semi-autonomous operation of vehicles," in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2012, pp. 383–388.

[18] K. Chu, M. Lee, and M. Sunwoo, "Local path planning for off-road autonomous driving with avoidance of static obstacles," *IEEE Trans. Intell. Transp. Syst.*, vol. 13, no. 4, pp. 1599–1616, Dec. 2012.

[19] R. Kala and K. Warwick, "Multi-level planning for semi-autonomous vehicles in traffic scenarios based on separation maximization," *J. Intell. Robot. Syst.*, vol. 72, no. 3/4, pp. 559–590, Dec. 2013.

[20] S. K. Gehrig and F. J. Stein, "Collision avoidance for vehicle-following systems," *IEEE Trans. Intell. Transp. Syst.*, vol. 8, no. 2, pp. 233–244, Jun. 2007.

[21] C. Frese and J. Beyerer, “A comparison of motion planning algorithms for cooperative collision avoidance of multiple cognitive automobiles,” in *Proc. IEEE Intell. Veh. Symp.*, Jun. 2011, pp. 1156–1162.

[22] R. Kala and K. Warwick, “Motion planning of autonomous vehicles in a non-autonomous vehicle environment without speed lanes,” *Eng. Appl. Artif. Intell.*, vol. 26, no. 5/6, pp. 1588–1601, May/Jun. 2013.

[23] A. Furda and L. Vlacic, “Enabling safe autonomous driving in real-world city traffic using multiple criteria decision making,” *IEEE Intell. Transp. Syst. Mag.*, vol. 3, no. 1, pp. 4–17, Spring 2011.

[24] R. Schubert, K. Schulze, and G. Wanielik, “Situation assessment for automatic lane-change maneuvers,” *IEEE Trans. Intell. Transp. Syst.*, vol. 11, no. 3, pp. 607–616, Sep. 2010.

[25] J. L. Baxter, E. K. Burke, J. M. Garibald, and M. Normanb, “Shared potential fields and their place in a multi-robot co-ordination taxonomy,” *Robot. Autonom. Syst.*, vol. 57, no. 10, pp. 1048–1055, Oct. 2009.

[26] R. Gayle, W. Moss, M. C. Lin, and D. Manocha, “Multi-robot coordination using generalized social potential fields,” in *Proc. IEEE Int. Conf. Robot. Autom.*, 2009, pp. 106–113.

[27] R. Gayle, A. Sud, E. Andersen, S. J. Guy, M. C. Lin, and D. Manocha, “Interactive navigation of heterogeneous agents using adaptive roadmaps,” *IEEE Trans. Vis. Comput. Graphics*, vol. 15, no. 1, pp. 34–48, Jan./Feb. 2009.

Rahul Kala received the B.Tech and M.Tech degrees in information technology from Indian Institute of Information Technology and Management, Gwalior, India. He is currently working toward the Ph.D. degree from the School of Systems Engineering, University of Reading, Reading, U.K.

He is the author of three books and over 60 papers. His recent book is on robotic planning titled *Intelligent Planning for Mobile Robotics: Algorithmic Approaches* (IGI-Global Publishers, 2013).

Mr. Kala is a member of the Association for the Advancement of Artificial Intelligence. He received the Commonwealth Scholarship from the Association of Commonwealth Universities and the British Council, the Lord of the Code scholarship from RedHat and the Indian Institute of Technology Bombay, and the GATE scholarship from the Indian Ministry of Human Resource Development.

Kevin Warwick received the B.Sc. degree from Aston University, Birmingham, U.K.; the Ph.D. and D.Sc. degrees from Imperial College, London, U.K.; and the D.Sc. degree from Czech Academy of Sciences, Prague, Czech Republic.

He is a Professor of Cybernetics with the University of Reading, Reading, U.K., where he is involved in research on artificial intelligence, control, robotics and cyborgs. He is the author or coauthor of more than 600 research papers and is perhaps best known for his experiments using implant technology.

Dr. Warwick is an Honorary Member of the St. Petersburg Academy of Sciences. He received The Future of Health Technology Award in the Massachusetts Institute of Technology, Cambridge, MA, USA, the IEE Senior Achievement Medal in 2004, the Mountbatten Medal in 2008, the Golden Eurydice in 2009, and the Ellison–Cliffe Medal in 2011. He has also received Honorary Doctorates from the Universities of Aston, Coventry, Bradford, Robert Gordon, Portsmouth, and Bedfordshire.