Local Graph-based Distributed Control for Safe Highway Platooning

Sven Gowal, Riccardo Falconi, Alcherio Martinoli

Abstract—**Using graph theory, this paper investigates how a group of vehicles, endowed with local positioning capabilities (range and bearing to other vehicles), can keep a predefined formation. We propose a longitudinal and lateral controller that stabilizes a system of several vehicles as well as a collision avoidance mechanism. The stability of our approach is supported by a mathematical analysis as well as realistic simulations.**

I. Introduction

Since the 1990s, and the start of the California's Partners for Advanced Transit and Highways (PATH) project [1], several works have shown that when multiple cars drive together in *platoons*, traffic throughput and safety can be increased [19]. *Platooning* is a complex task that requires automobiles to be able to drive in a controlled and coordinated fashion. One aspect of platooning is to control formations of vehicles. Managing formations of non-holonomic vehicles has received a lot of attention in the last decade and is known to be difficult when only local positioning is available [7, 17, 20]. Our work considers that vehicles have only range and bearing information to the neighboring vehicles and no communication available. Hence, we inspire ourselves from potential fields [8] and graph theoretic [3] approaches and use a decentralized *Laplacian feedback control* [2] to solve a *consensus* problem [13] with the ultimate goal of guiding a formation of vehicles. Unlike most of the research on formation control focusing on differential drive robots, we consider here non-holonomic vehicles which can only move forward or backward in a direction tangent to their orientation.

As in [14, 15], we propose a control strategy and analyze its stability when driving multiple cars. Such strategies always rely on longitudinal and lateral control that are intrinsically linked due to non-holonomicity. In Section III, we augment an existing lateral control policy explained in [9] with a simple longitudinal PI (Proportional, Integral) controller and we prove its stability on a single vehicle. Further (in Section IV), we use the graph theory and the *Laplacian feedback control* (explained in Section II) in conjunction with our single car controller to keep a predefined formation of multiple vehicles. The stability of the whole system is also proven.

Finally, using the mathematical framework provided by graph theory, we add an active collision avoidance mechanism that makes use of both longitudinal and lateral axes. Not only does it enable the agents to avoid obstacles on the road, but also to avoid themselves while converging to the desired

Sven Gowal and Alcherio Martinoli are with the Distributed Intelligent Systems and Algorithms Laboratory of the Ecole Polytechnique Fédérale de Lausanne. `svenadrian.gowal@epfl.ch`, `alcherio.martinoli@epfl.ch`

Riccardo Falconi is with the Dipartimento di Elettronica, Informatica e Sistemistica of the Università di Bologna. `riccardo.falconi@unibo.it`
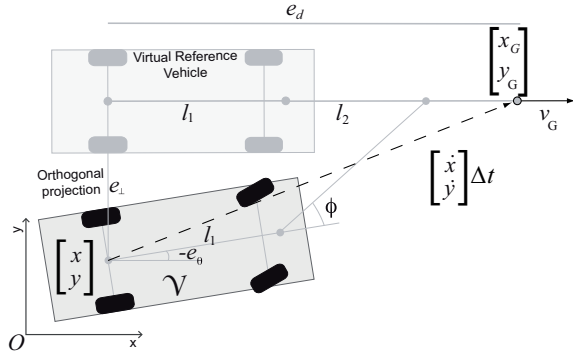
formation. In Section V, we test our approach in Webots [10], a realistic robotic simulator for which we built a car physics plugin. Section VI concludes this paper.

II. Background

A. Basic Notions of Graph Theory

In this section the main graph theory notions that we will use are summarized. A directed simple graph with N elements is defined as a pair $\mathcal{G} = (V, E)$, where

- $V = \{v_i, i = 1 \ldots N\}$ is the *vertex set*,
- $E \subseteq V \times V$ is the *edge set*.

The elements of E are ordered pairs of elements $e_k = (v_i, v_j)$ with $k = \{1 \ldots |E|\}$. The i-th node *neighbors subset* is defined as $\mathcal{N}_i = \{\forall v_j \in V : (v_i, v_j) \in E\}$. Given such a graph, we can define the *incidence matrix* $\mathcal{I} \in \mathbb{R}^{N \times |E|}$ as:

$$\mathcal{I}_{i,k} = \begin{cases} -1 & \text{if } e_k = (v_i, v_j) \\ 1 & \text{if } e_k = (v_j, v_i) \\ 0 & \text{otherwise} \end{cases}$$

where e_k is the k-th edge of $\mathcal{G}$. When we deal with undirected graphs a random orientation for the edges can be chosen and the *incidence matrix* calculated. The definition of the *incidence matrix* allows us to define the *Laplacian matrix* as

$$\mathcal{L} = \mathcal{I} \cdot \mathcal{W} \cdot \mathcal{I}^T \qquad (1)$$

where the *weight matrix* $\mathcal{W} \in \mathbb{R}^{|E| \times |E|}$ is a diagonal matrix whose element $\mathcal{W}_{k,k}$ relates to the importance of each edge e_k. In particular, if at least one weight $\mathcal{W}_{k,k}$ differs from 1, the Laplacian matrix is addressed as the *weighted Laplacian matrix*.

B. The Consensus Problem

The *consensus problem* [13] is a well-known and widely studied problem in the field of decentralized control. It starts by considering all the agents of a group as holonomic kinematic models:

$$\dot{x}^i = u^i$$

where x^i is the state of the i-th agent. The solution of the consensus problem for a graph of N agents, whose goal is to drive the whole system to a final common state, can be solved with a Laplacian-based feedback method if the graph is connected. The feedback control is in the form

$$\dot{x} = u = -\mathcal{L}x. \qquad (2)$$

To extend the consensus problem to more than one dimension, the *Kronecker product* can be used:

$$\begin{bmatrix} \dot{x} \\ \dot{y} \end{bmatrix} = -\mathcal{L} \otimes I_2 \begin{bmatrix} x \\ y \end{bmatrix} \qquad (3)$$

Fig. 1. Representation of the problem to solve.

Fig. 2. Phase portrait showing a region of the $e_\perp, e_\theta, e_d$-space, generated with $l_1 = L = 3.0$, $l_2 = 4.0$, $l_3 = 1.0$ and $\phi_{\max} = 0.45$.

where I_2 is the 2×2 identity matrix. We can conclude that if we consider the state $x^i(t)$ of each robot i (or agent) to be its position, we can drive a group of holonomic robots to a rendez-vous point in x, y-coordinates with Equation 3.

III. FROM COORDINATES TO CONTROL VARIABLES

The solution of the consensus problem briefly outlined above is based on the assumption that the agents are modeled as bodyless, holonomic vehicles. This section is dedicated to finding the appropriate translation between the holonomic control variables $\dot{x}(t)$, $\dot{y}(t)$ and the non-holonomic control variables of a simplistic car, namely $v(t)$ and $\phi(t)$, the speed and steering angle respectively.

A. Vehicle Model

Although this assumption holds only for low speed and small steering angles, we will consider throughout this work the following bicycle model as our car model:

$$\begin{cases} \dot{x} &= \cos(\theta) \cdot v \\ \dot{y} &= \sin(\theta) \cdot v \\ \dot{\theta} &= \frac{\tan(\phi)}{L} \cdot v \end{cases}$$

where $[x, y]^T$ defines the position of the midpoint of the rear axle in an Euclidean reference frame, θ means the orientation of the car relative to the x-axis, L is the wheelbase, ϕ is the steering angle (the angle of the front wheels relative the car's local x-axis) and v is the current speed. We note that in this paper all the mathematical analyses consider an unbounded steering angle whereas all simulations assume that the steering angle is limited ($|\phi| < \phi_{\max}$).

B. Lateral Control

Let us consider the problem represented on Figure 1. We are given (from the *Laplacian control feedback*) a horizontal and vertical displacement rate $\dot{x}$ and $\dot{y}$. This displacement enables us to create a line (the *goal line*) that passes through the $[x_G, y_G]^T = [x, y]^T + \Delta t [\dot{x}, \dot{y}]^T$ point (the *goal point*) where Δt is a positive time horizon (in the sequel we set $\Delta t = 1$ [s]). We assume here for simplicity that the wanted final direction for all vehicles is parallel to the global x-axis (we assume that vehicles are able to determine this direction, which could
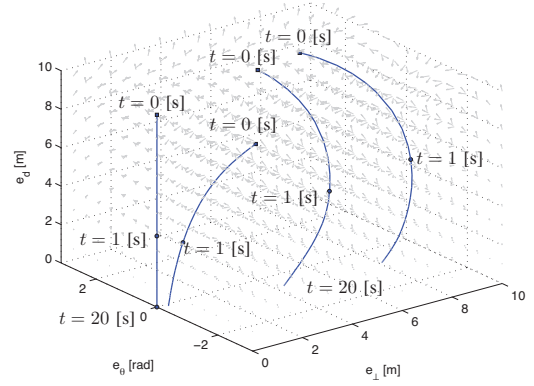
be done by analyzing the road markings for example). Hence the orientation θ of a vehicle is equal to the heading error $-e_\theta$ with respect to the *goal line*. The lateral and longitudinal errors with the *goal line* are $e_\perp = y_G - y$ and $e_d = x_G - x$ respectively. The goal of this section is to create a control that reaches the *goal point* with the correct orientation.

From the vehicle model and as partly explained in [9], we can deduce that:

$$\begin{cases} \dot{e}_\perp &= \sin(e_\theta) \cdot v \\ \dot{e}_\theta &= -\frac{\tan(\phi)}{L} \cdot v \\ \dot{e}_d &= -\cos(e_\theta) \cdot v \end{cases} \quad (4)$$

[9] explains a lateral controller able to bring $e_\perp$ and e_θ to 0. If the vehicle moves forward ($v > 0$), the control law sketched for ϕ in Figure 1 is stable. To reach the *goal line*, we can apply:

$$\tan(\phi(t)) = \frac{-\cos(e_\theta(t))e_\perp(t) - (l_1 + l_2)\sin(e_\theta(t))}{l_1 - (l_1 + l_2)\cos(e_\theta(t)) + \sin(e_\theta(t))e_\perp(t)} \quad (5)$$

where l_1 and l_2 are two positive control constants. If the vehicle moves backward the control law needs to be slightly modified. In this work, our mathematical developments focus only on a forward motion of the vehicles. Nevertheless, they are also valid for the backward motion if instead of using (5) we use

$$\tan(\phi(t)) = \frac{-\cos(e_\theta(t))e_\perp(t) - (l_1 + l_2)\sin(e_\theta(t))}{l_1 + (l_1 + l_2)\cos(e_\theta(t)) - \sin(e_\theta(t))e_\perp(t)}.$$

C. Adding the Longitudinal Control

The key point is to control the speed $v(t)$ of the car so as to reach the *goal point* without making the whole system unstable. Let us use a simple proportional control:

$$v(t) = l_3 e_d(t)$$

with l_3 a positive constant. Figure 2 shows four trajectories of this new controller in the $e_\perp, e_\theta, e_d$-space. In this phase portrait, the steering angle was limited to 0.45 [rad]. We can observe (and it is expected) that the error e_d goes to 0, setting the speed v to 0, thus stopping the convergence of $e_\perp$ and e_θ.
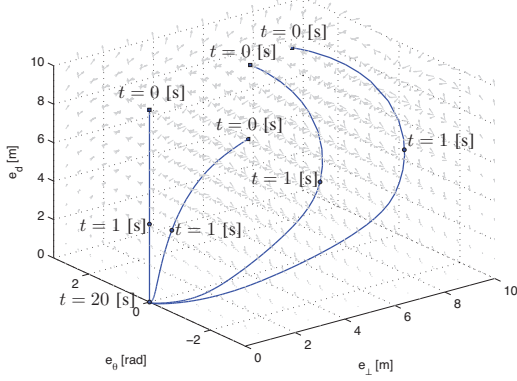
Fig. 3. Phase portrait showing a region of the $e_\perp, e_\theta, e_d$-space, generated with $l_1 = L = 3.0$, $l_2 = 4.0$, $l_3 = 1.0$, $v_G = 1.0$, $K_I = 0.1$ and $\phi_{\max} = 0.45$.

As we are investigating a platooning technique where vehicles try to reach a predefined speed, let us assume that the *goal point* moves along the *goal line* at a speed $v_G > 0$ (see Figure 1) known by all vehicles. In practice (and if this speed is not known by all cars), vehicles can use a PI controller to estimate it [5]:

$$v(t) = l_3 e_d(t) + K_I \cdot \int_0^t e_d(\tau) \mathrm{d}\tau \qquad (6)$$

Intuitively, if v_G stays constant, $K_I \cdot \int_0^t e_d(\tau)\mathrm{d}\tau$ should reach v_G as the time t tends to infinity. The error dynamics (4) become:

$$\begin{cases} \dot{e}_\perp(t) &= & \sin(e_\theta(t)) \cdot v(t) \\ \dot{e}_\theta(t) &= & -\frac{\tan(\phi(t))}{L} \cdot v(t) \\ \dot{e}_d(t) &= & -\cos(e_\theta(t)) \cdot v(t) + v_G \end{cases}$$

with $v(t) = l_3 e_d(t) + v_G$ and $\phi(t)$ as in Equation 5. This system has two sets of fixed points in the $e_\perp, e_\theta, e_d$-space: $(0, 2k\pi, 0)$ and $(0, \pi + 2k\pi, -2v_G/l_3)$ with $k \in \mathbb{Z}$. The linearized Jacobian of our system around $(0, 2k\pi, 0)$ is:

$$J_0 = \begin{bmatrix} 0 & v_G & 0 \\ -\frac{v_G}{Ll_2} & -\frac{(l_1+l_2)v_G}{Ll_2} & 0 \\ 0 & 0 & -l_3 \end{bmatrix}.$$

It yields as eigenvalues:

$$\begin{cases} \lambda_1 &= & -l_3 \\ \lambda_{2,3} &= & -v_G \frac{l_1+l_2 \pm \sqrt{-4Ll_2 + (l1+l2)^2}}{2Ll_2} \end{cases}$$

Under our current constraints ($l_1, l_2, l_3, L, v_G > 0$), the real part of all the eigenvalues are negative. We conclude that the fixed points $(0, 2k\pi, 0)$ are asymptotically stable. Furthermore, as the real part of two eigenvalues of the Jacobian around the other set of fixed points are positive, the fixed points $(0, \pi + 2k\pi, -2v_G/l_3)$ are unstable. This not only demonstrates that our system is stable when operating around $(0, 0, 0)$ but also
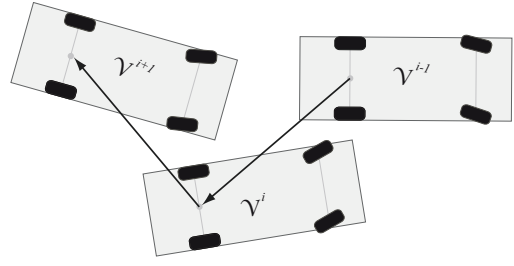

Fig. 4. Chain or string of vehicles solving the *rendez-vous problem*.

that it converges to this nominal operation point. Figure 3 shows the same four trajectories as in Figure 2 converging to a moving *goal point* using the PI speed controller of Equation 6 in the $e_\perp, e_\theta, e_d$-space.

We have seen in this section that given some displacement $\Delta t[\dot{x}, \dot{y}]^T$ with respect to a car position, we can build a *goal point* $[x_G, y_G]^T$, that if moving ($v_G \neq 0$), can be reached with the controller given by Equations 5 and 6.

IV. VEHICULAR CONSENSUS

To simplify our upcoming multi-vehicle analysis we will first transform our system and control law to coordinates relative to the *goal point* (i.e. the translation along the x-axis induced by the speed v_G is ignored):

$$\begin{cases} \dot{x}(t) &= & \cos(\theta(t)) \cdot v(t) - v_G \\ \dot{y}(t) &= & \sin(\theta(t)) \cdot v(t) \\ \dot{\theta}(t) &= & \frac{\tan(\phi)}{L} \cdot v(t) \end{cases}$$

with

$$\tan(\phi(t)) = \frac{-\cos(\theta(t))\Delta y(t) - (l_1 + l_2)\sin(\theta(t))}{-l_1 + (l_1 + l_2)\cos(\theta(t)) - \sin(\theta(t))\Delta y(t)}$$

$$v(t) = l_3(x_G(t) - x(t)) + v_G.$$

where $\Delta y(t) = y(t) - y_G(t)$. $x_G(t)$ and $y_G(t)$ become inputs to the system. We note that $[x_G(t) + v_G \cdot t, y_G(t)]^T$ represents the *goal point* position in time. After linearization of the system around the nominal regime ($x(t) = x_G(t)$, $y(t) = y_G(t)$ and $\theta(t) = 0$), we obtain:

$$\begin{bmatrix} \dot{x}(t) \\ \dot{y}(t) \\ \dot{\theta}(t) \end{bmatrix} = A \cdot \begin{bmatrix} x(t) \\ y(t) \\ \theta(t) \end{bmatrix} + B \cdot \begin{bmatrix} x_G(t) \\ y_G(t) \end{bmatrix}$$

with

$$A = \begin{bmatrix} -l_3 & 0 & 0 \\ 0 & 0 & v_G \\ 0 & -\frac{v_G}{Ll_2} & -\frac{(l_1+l_2)v_G}{Ll_2} \end{bmatrix}, B = \begin{bmatrix} l_3 & 0 \\ 0 & 0 \\ 0 & \frac{v_G}{Ll_2} \end{bmatrix}.$$

A. String Stability

Let us consider now a fleet of N vehicles that have to solve the *rendez-vous problem*. We propose to link each vehicle as in Figure 4, thus forming a chain. The idea is to give as input
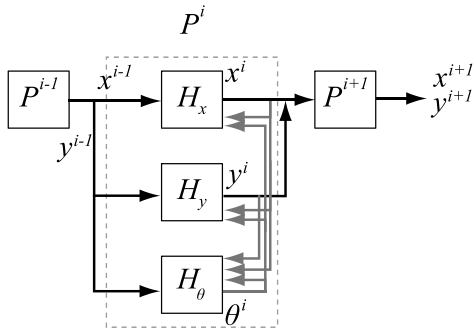
Fig. 5. Block diagram of the controller of each individual vehicle i within the car chain.
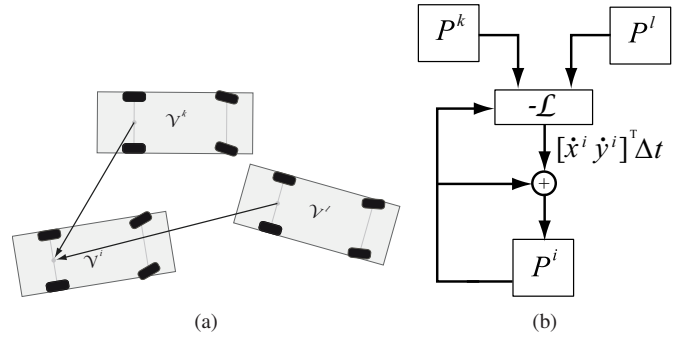

Fig. 6. Integration of the *Laplacian feedback control* into our system of cars. (a) Assuming that the car $\mathcal{V}^i$ is *connected* to the other cars $\mathcal{V}^k$ and $\mathcal{V}^l$, (b) we create the block diagram that aggregates all car positions and generates a new *goal point* to reach.

to the controller of each car $\mathcal{V}^i$ the position of the previous car $\mathcal{V}^{i-1}$, thus yielding a linearized system P^i for each vehicle:

$$\begin{bmatrix} \dot{x}^i(t) \\ \dot{y}^i(t) \\ \dot{\theta}^i(t) \end{bmatrix} = A \cdot \begin{bmatrix} x^i(t) \\ y^i(t) \\ \theta^i(t) \end{bmatrix} + B \cdot \begin{bmatrix} x^{i-1}(t) \\ y^{i-1}(t) \end{bmatrix}$$

This system is sketched in Figure 5 where we have separated the system in three sub-blocks H_x, H_y and H_θ representing the transfer functions from the input $[x^{i-1}(t), y^{i-1}(t)]^T$ to the output $[x^i(t), y^i(t)]^T$.

This system is identical to the one explained in Section III-C and is stable. The question now is whether the whole system (with its N vehicles) is stable – in other words, whether the disturbances grow or attenuate as they propagate through our system of vehicles [12]. This is regarded in litterature as *string stability* [18]. Sheikholeslam and Desoer in [16] state that a cascaded system of identical vehicles (with a transfer function $P(s)$) is string stable if $|P(j\omega)| < 1$ for all ω. We note that this result is a sufficient condition to prove *string stability*, but it is not a necessary condition. Hence if we can prove that our transfer functions H_x, H_y and H_θ all have gains lower than 1, then our chain of vehicles is stable. The transfer functions of our system (7) are:

$$H_x(s) = \frac{l_3}{s + l_3}$$

$$H_y(s) = \frac{v_G^2}{Ll_2 s^2 + v_G(l_1 + l_2)s + v_G^2}$$

$$H_\theta(s) = \frac{sv_G}{Ll2 s^2 + v_G(l_1 + l_2)s + v_G^2}.$$

Their gains are:

$$|H_x(j\omega)| = \frac{l_3^2}{\omega^2 + l_3^2}$$

$$|H_y(j\omega)| = \frac{v_G^4}{L^2 l_2^2 \omega^4 + ((l1 + l2)^2 - 2Ll_2)v_G^2 \omega^2 + v_G^4}$$

$$|H_\theta(j\omega)| = \frac{v_G^2 \omega^2}{L^2 l_2^2 \omega^4 + ((l1 + l2)^2 - 2Ll_2)v_G^2 \omega^2 + v_G^4},$$

yielding the following two sufficient (and resonable) condi-

tions:

$$L \leq \frac{(l_1 + l_2)^2}{2l_2} \tag{7}$$

$$l_1 \geq 1 \tag{8}$$

such that all gains are lower than 1 for all ω. Hence the system is *string stable* if it satisfies Equations 7 and 8. Additionally, we note that each element of the string is BIBO (Bounded Input Bounded Output) stable.

B. Back to Graph Theory

The key result just obtained allows us to link vehicles together in any way (provided that only bounded inputs are given to each controller) whilst keeping our graph of agents stable. In particular, we can use Equation 3 as a way to use the *Laplacian feedback control* to achieve the consensus. Consider the transformation of the system depicted Figure 5 to the one on Figure 6. Positions for all vehicles (e.g., $\mathcal{V}^i$, $\mathcal{V}^k$, $\mathcal{V}^l$) can be aggregated through the Laplacian equation resulting in a displacement vector $\Delta t[\dot{x}, \dot{y}]$. Adding the corresponding displacement vector to each car position will produce each *goal point*.

1) Decentralization: In the context of platooning on highways without any additional road infrastructure, we need to decentralize the *Laplacian feedback control* presented in Section II-B, as well as use only relative coordinates. To do so, we make the assumption that any vehicle $\mathcal{V}^i$ is able to measure its distance $e_{i,j}$ and azimut $\alpha_{i,j}$ to other neighboring vehicles $\mathcal{V}^j$. Hence we can transform Equation 3 for each vehicle $\mathcal{V}^i$:

$$\begin{cases} \dot{x}^i = -\sum_{v_j \in \mathcal{N}_i} \mathcal{L}_{i,j} \cdot e_{i,j} \cos(\alpha_{i,j}) \\ \dot{y}^i = -\sum_{v_j \in \mathcal{N}_i} \mathcal{L}_{i,j} \cdot e_{i,j} \sin(\alpha_{i,j}) \end{cases} \tag{9}$$

In Figure 7, we have linked two vehicles using the above decentralized law with the *incidence matrix* $\mathcal{I} = [1, -1]^T$ and the *weight matrix* $\mathcal{W} = [1]$. We have also enforced their speed to be $v_G = 1.0$ [m/s]. Both cars converge to $(0, 0, 0)$ in the $e_\perp, e_\theta, e_d$-space and reach the same position in the global coordinate frame.
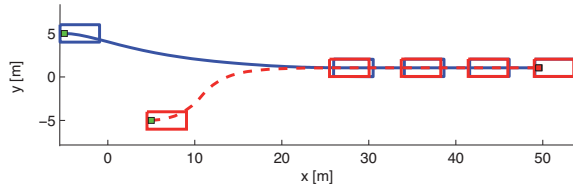
Fig. 7. Trajectories of two cars engaged in a consensus and using the *Laplacian feedback control* in the x, y, θ-space. We have used $l_1 = L = 3.0$, $l_2 = 4.0$, $l_3 = 1.0$, $v_G = 1.0$, $K_I = 0.1$ and $\phi_{\max} = 0.45$.
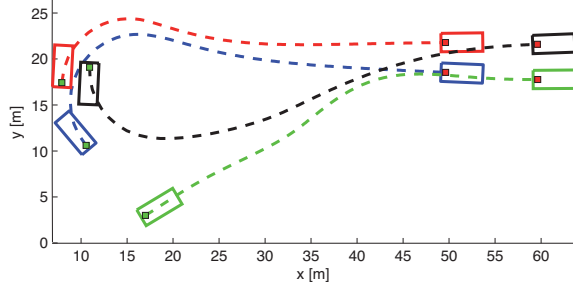

Fig. 9. Safety bounding box defined as the dangerous region where the vehicle engages collision avoidance.

Fig. 8. Trajectories of four cars converging to a rectangular formation.
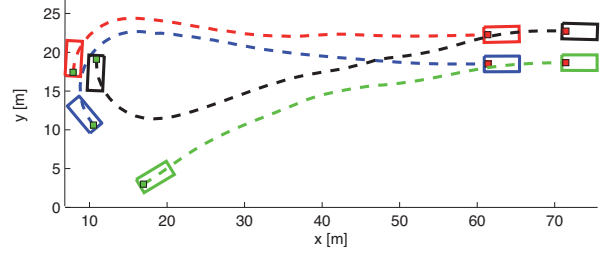
Fig. 10. Trajectories of four cars converging to a rectangular formation using collision avoidance. The maximal deceleration was set to $a_{\max} = 4.0 \; [\mathrm{m/s^2}]$.

C. Creating Formations

Until now we have considered that all cars should converge to the same spatial location. This is, of course, neither possible nor desirable: the vehicles should keep instead predefined distances with their neighbors. In [4] it is explained how to use the *Laplacian approach* to achieve formation control. Equation 2 can be modified to accomodate a *bias vector b* enabling the system to reach a specific configuration:

$$\dot{x} = u = -\mathcal{L}(x - b).$$

Hence Equation 9 becomes:

$$\begin{cases} \dot{x}^i = -\sum_{v_j \in \mathcal{N}_i} \mathcal{L}_{i,j} \cdot \left(e_{i,j} \cos(\alpha_{i,j}) - b_{i,j}^x \right) \\ \dot{y}^i = -\sum_{v_j \in \mathcal{N}_i} \mathcal{L}_{i,j} \cdot \left(e_{i,j} \sin(\alpha_{i,j}) - b_{i,j}^y \right) \end{cases}.$$

with $b_{i,j}^x$ and $b_{i,j}^y$ being the desired longitudinal and lateral offset between the vehicles $\mathcal{V}^i$ and $\mathcal{V}^j$ respectively.

Figure 8 shows a group of vehicles converging to a rectangular formation. In this example, the *Laplacian matrix* is:

$$\mathcal{L} = \begin{bmatrix} 3 & -1 & -1 & -1 \\ -1 & 3 & -1 & -1 \\ -1 & -1 & 3 & -1 \\ -1 & -1 & -1 & 3 \end{bmatrix}$$

and the *bias matrices* composed of each *bias vector* are:

$$b^x = \begin{bmatrix} 0 & 0 & -10 & -10 \\ 0 & 0 & -10 & -10 \\ 10 & 10 & 0 & 0 \\ 10 & 10 & 0 & 0 \end{bmatrix}, b^y = \begin{bmatrix} 0 & -4 & 0 & -4 \\ 4 & 0 & 4 & 0 \\ 0 & -4 & 0 & -4 \\ 4 & 0 & 4 & 0 \end{bmatrix}.$$

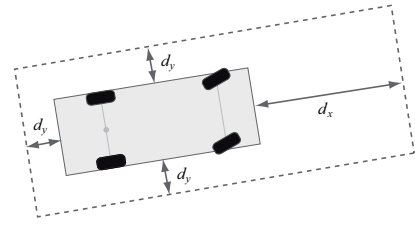Thus, by modifying the values of the *bias matrices* our system is able to reconfigure into any shape.

D. Collision Avoidance

Although they are not visible, the example depicted on Figure 8 engendered two collisions (i.e. at least two of the cars' bounding boxes overlapped) at 1.2 and 3.2 seconds into the run. Fortunately, using graph theory and the *Laplacian feedback control*, we can seamlessly add on top of our formation keeping behavior a distributed collision avoidance control. Let us define $\overline{E}$ as the set of all edges (v_i, v_j) such that the vehicle $\mathcal{V}^j$ is not within the *dangerous region* of $\mathcal{V}^i$. We denote by *dangerous region* the spatial region that is covered by a car driving at its current speed $v(t)$ and suddenly decelerating at its maximum acceleration $a_{\max}$ until halted. For simplicity and as shown in Figure 9, we have defined it as a simple bounding box where the lateral and rear distances d_y are constant and the frontal distance $d_x(t)$ is dependant on time and equal to:

$$d_x(t) = \frac{v(t)^2}{2a_{\max}}$$

Finally, the behavior between pairs of potentially colliding vehicles should become repulsive instead of attractive. Hence we could define the weight $\mathcal{W}_{k,k}$ of each edge e_k to be:

$$\mathcal{W}_{k,k} = \begin{cases} 1/N & \text{if } e_k \in \overline{E} \\ -\delta/d_{i,j} & \text{otherwise} \end{cases}$$

where δ is a positive constant and $d_{i,j}$ is the distance between the bodies of car $\mathcal{V}^i$ and car $\mathcal{V}^j$. An adjustment of the *bias vector* is also needed as the repulsion point should be the same for both cars. Thus if $e_k = (v_i, v_j) \notin \overline{E}$ then $b_{i,j}^x = b_{i,j}^y = 0$.

Figure 10 shows the trajectories generated by four cars having the same initial conditions as in Figure 8. We note that each car has a limited acceleration and deceleration of $a_{\max} = 4.0 \; [\mathrm{m/s^2}]$. We can observe that trajectories are slightly different and result in no collisions anymore.
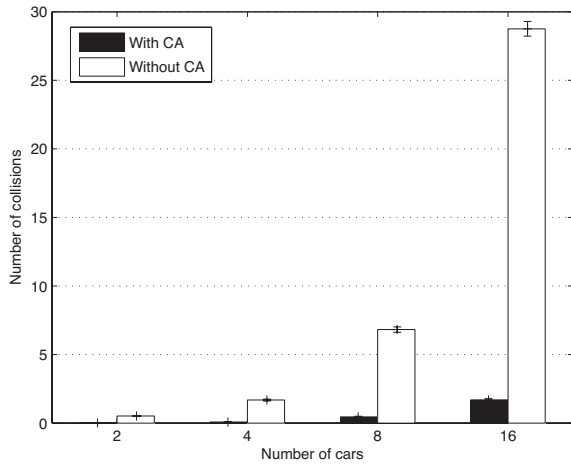
Fig. 11. Average number of collisions detected with and without the collision avoidance mechanism. The error bars represent 95% confidence intervals.

We have also performed systematic 60 [s] simulation runs where 2, 4, 8 and 16 vehicles were asked to achieve a formation on two lanes: pairs of cars are asked to stay 10 [m] apart and cars in the same pair 4 [m] apart (similiar to the experiments in Figures 8 and 10). Vehicles were initially randomly placed in a 20×20 [m] area (collisions due to cars overlapping at time $t = 0$ are ignored) with random orientations and random speeds varying between 0 and 20 [m/s]. Results are summarized on Figure 11. We observe that collisions are significantly reduced and note that some collisions (due to inadequate initial conditions) were unavoidable as we limited the maximum acceleration and deceleration.

Finally, we have obtained a complete system able to keep a predefined formation at a specified speed whilst actively avoiding collisions.

E. Notes about Assumptions

Throughout this work we have made several assumptions. In this section, we explain how to alleviate some of them.

1) Straight trajectory: To simplify our analyses, we have proposed that vehicles need to reach a *goal line*. As such our formations are only able to accomodate trajectories with a small curvature. Fortunately there are no restrictions on the curve to reach and the notion of *goal curve* is valid. The state of each car simply needs to be converted beforehand from rectangular coordinates to curvilinear coordinates (as explained in [15]).

2) Leaderless formations: Formation speed and direction of motion was a priori known by all vehicles. Hence there were no need to have a formation leader (not even a virtual leader). In some sense, this approach is quite unique and provides an approach that is fully distributed and robust to failure. We can note that optionally, any number of cars within the formation can be declared as leaders. The leaders do not use the *Laplacian feedback control* and do not need to be known by the other vehicles (the followers). The followers will automatically adapt and try their best to keep the formation

with the leaders as anchor points. This strategy has already been used in [5].

3) Unique identifiers: The *Laplacian feedback control* assumes that vehicles are uniquely identified. To perform platooning, the relative position of the vehicles within the formation is enough to compute their identifiers. Let us consider a formation on two lanes, each vehicles can have either 3 (if it is in leading or trailing row) or 5 neighbors. They are also able to determine whether they are on the left or right side of the formation. Hence they are capable of knowing their role within the formation and assign identifiers to their neighbors and themselves.

4) Obstacle-free roads: Obstacles can be easily integrated into our framework. If a vehicle detects an obstacle it will add it to its list of neighbors and handle it as a replusive agent.

V. EXPERIMENTS

A. Simulated Vehicles

Experiments are conducted in Webots [10], a realistic mobile robotic simulator for which we have developed a realistic car simulator plugin [6]. Webots carefully reproduces discrete sensors and actuators with their calibrated nonlinearities and noise features. Our car model incorporates basic rigid dynamics properties including typical steering dynamics response. Although not calibrated with an actual vehicle, we performed several validation runs and tried to bring the friction (of the tires) and throttle/brake (engine) model close to the ones of a real vehicle based on literature information..

The main differences between this real Ackermann steering vehicle and the simulated model (of Equation 4) are that steering and speed dynamics include a time lag (dependent on physical properties) and that the speed is only controllable through the throttle and brake pedals which have a nonlinear and noisy relation with the acceleration. Hence we will now control the speed $v(t)$ given by Equation 6 with:

$$\tau(t) = f^{-1}(l_4 \cdot (v(t) - v_{\text{current}}(t)))$$

where $\tau(t)$ is the throttle (if positive) and brake (if negative) positions, $v_{\text{current}}(t)$ is the current speed, $f(\cdot)$ is the nonlinear relation between the throttle position and the desired acceleration at the current speed and l_4 is a positive constant.

We have equiped our simulated vehicles with four simulated SICK LMS 291 sensors as to cover a $360°$ field of view. The SICK LMS 291 is a laser rangefinder, which scans at 75 Hz over $180°$ with a $0.25°$ angular resolution. Their sensing range can go up to 80 [m] with an error of about 1 [cm] at 30 [m]. We have implemented a dynamic object detection and tracking method similar to the one proposed in [11] where *synthetic scans* are created and areas of difference between consecutive scans are tracked by particle filters.

B. Setup

For each experimental run, four intelligent vehicles are placed in an area free of obstacles. Vehicles have to perform the rectangular formation already explained in Section IV-D for Figure 10 at a speed $v_G = 10$ [m/s]. Their initial pose
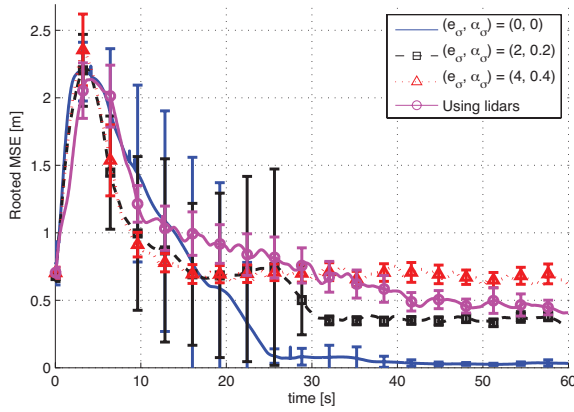
Fig. 12. Average and standard deviation of the rooted mean square error of the distances between each car and the desired distances depending on time.

is randomly drawn from a normal distribution around their *desired pose* with a standard deviation of 2 [m] in each direction and of $\pi/4$ [rad] for the orientation. We define by *desired pose* the pose such that the longitudinal and lateral distances between each vehicle satisfy the final formation. Additionally their speed is randomly drawn from a uniform distribution between 0 and 20 [m/s].

C. Results

We perform four sets of experiments: three sets where the range and bearing measurements are given to the vehicles with a zero mean gaussian noise with variance e_σ^2, α_σ^2 respectively and a last set where these measurements are gathered from the dynamic object tracker using the laser rangefinders. The position of each car is monitored during a run and each run lasts 60 seconds. After 100 runs, the average rooted mean square error (MSE) between the actual distances between each pair of cars and the desired distances is computed. Figure 12 shows the rooted MSE for $(e_\sigma, \alpha_\sigma) = \{(0,0), (2, 0.2), (4, 0.4)\}$. As the noise grows the final MSE gets larger, but we observe that in all cases the formation is stable. The average error on each vehicle link is lower than a meter even with a noise as high as 4 [m] and 0.4 [rad]. Figure 12 also shows that the convergence of the last set is steady. When we used the laser rangefinders, we measured errors of -0.4176 ± 1.4492 [m] for the range and 0.0449 ± 0.2143 [rad] for the bearing.

Overall, these simulation results show the good performances of our approach and its ability to stabilizes under challenging conditions (videos showing simulated vehicles are available on http://disal.epfl.ch/research/context_aware_its/videos/).

VI. CONCLUSION

In this paper, we demonstrated that we could drive a group of automobiles in a specific formation. We have proved mathematically that our approach is stable. In particular, we solved the *consensus* problem for kinematically constrained vehicles whilst providing an efficient and active collision avoidance mechanism by exploiting only local measurements between neighboring vehicles. We also tested the robustness of our control under realistic conditions.

REFERENCES

[1] California partners for advanced transit and highways. http://www.path.berkeley.edu.

[2] L. Cremean and R. Murray. Stability analysis of interconnected nonlinear systems under matrix feedback. In *IEEE Conf. on Decision and Control (CDC 2003)*, volume 3, pages 3078–3083, 2003.

[3] J. Desai, J. Ostrowski, and V. Kumar. Modeling and control of formations of nonholonomic mobile robots. *IEEE Trans. on Robotics and Automation*, 17(6):905–908, 2001.

[4] M. Egerstedt. *Graph-theoretic methods for multi-agent coordination*. ROBOMAT, 2007.

[5] R. Falconi, S. Gowal, and A. Martinoli. Graph based distributed control of non-holonomic vehicles endowed with local positioning information engaged in escorting missions. In *IEEE Conf. on Robotics and Automation (ICRA 2010)*, pages 3207–3214, 2010.

[6] S. Gowal, Y. Zhang, and A. Martinoli. A realistic simulator for the design and evaluation of intelligent vehicles. In *IEEE Conf. on Intelligent Transportation Systems (ITSC 2010)*, 2010, To appear.

[7] M. Ji and M. Egerstedt. Distributed coordination control of multiagent systems while preserving connectedness. *IEEE Trans. on Robotics*, 23(4):693–703, 2007.

[8] N. Leonard and E. Fiorelli. Virtual leaders, artificial potentials and coordinated control of groups. In *IEEE Conf. on Decision and Control (CDC 2001)*, volume 3, pages 2968–2973, 2001.

[9] M. Linderoth, K. Soltesz, and R. Murray. Nonlinear lateral control strategy for nonholonomic vehicles. In *American Control Conference*, pages 3219–3224, 2008.

[10] O. Michel. Webots: Professional Mobile Robot Simulation. *Journal of Advanced Robotic Systems*, 1(1):39–42, 2004.

[11] M. Montemerlo et al. Junior: The stanford entry in the urban challenge. *Journal of Field Robotics*, 25(9):569–597, 2008.

[12] R. M. Murray. Recent research in cooperative control of multivehicle systems. *Journal of Dynamic Systems, Measurement, and Control*, 129(5):571–583, 2007.

[13] R. Olfati-Saber, J. A. Fax, and R. M. Murray. Consensus and cooperation in networked multi-agent systems. *Proc. of the IEEE*, 95(1):215–233, 2007.

[14] S. Ramaswamy and S. Balakrishnan. Formation control of car-like mobile robots: A lyapunov function based approach. In *American Control Conference*, pages 657–662, 2008.

[15] M. Saska, J. Mejia, D. Stipanovic, and K. Schilling. Control and navigation of formations of car-like robots on a receding horizon. In *IEEE Conf. on Control Applications (CCA 2009)*, pages 1761–1766, 2009.

[16] S. Sheikholeslam and C. Desoer. Longitudinal control of a platoon of vehicles with no communication of lead vehicle information: a system level study. *IEEE Trans. on Vehicular Technology*, 42(4):546–554, 1993.

[17] B. S. Smith, J. Wang, M. Egerstedt, and A. Howard. Automatic formation deployment of decentralized heterogeneous multi-robot networks with limited sensing capabilities. In *IEEE Conf. on Robotics and Automation (ICRA 2009)*, pages 2130–2135, 2009.

[18] D. Swaroop and J. Hedrick. String stability of interconnected systems. *IEEE Trans. on Automatic Control*, 41(3):349–357, 1996.

[19] H. S. Tsao, R. W. Hall, and B. Hongola. Capacity of automated highway systems: effect of platooning and barriers. PATH Research Report UCB-ITS-PRR-93-26, 1994.

[20] F. Zhang and S. Haq. Boundary following by robot formations without gps. In *IEEE Conf. on Robotics and Automation (ICRA 2008)*, pages 152–157, 2008.