Global Path Planning for Competitive Robotic Cars*

Tizar Rizano[1], Daniele Fontanelli[1], Luigi Palopoli[1], Lucia Pallottino[2] and Paolo Salaris[2]

Abstract— In this paper, we consider the optimal motion planning problem for an autonomous race car. A competitive autonomous car must acquire environmental and opponent information to compute in real time the minimum time collision free path and the low level control to track the chosen path. To cope with those requirements, we first solve the problem for a car running in isolation considering the optimal sequence of manoeuvres to approach bends and straight stretches of track. We then propose a discrete abstraction to derive a problem of graph optimisation that lends itself to a very efficient, albeit suboptimal, solution. In this context, an overtake manoeuvre against a slower car will be obtained excluding from the path the arcs that could potentially generate a collision. Finally, a control algorithm is used to ensure that the car always remains close to the planned path.

I. INTRODUCTION

The problem addressed in this paper can be summarised in the following terms: *given an autonomous car–like vehicle that runs on a known track sharing it with other vehicles, plan a trajectory that allows the vehicle to complete a given number of laps on the track, time avoiding collisions with other vehicles*. Path planning in a competition track is very different from path planning in a urban or extra urban road [1]. First, the environment is strongly structured and well known upfront. Second, the track is accessible only to a small and predetermined number of cars. Third, all the cars in the track have very similar spatial footprint and dynamic characteristics. This reduces the amount of real–time data that the planner requires. On the other hand, the high speed of the car requires the information acquisition, the optimal planning with collision avoidance and the control phases to be executed within tight real–time constraints.

a) Related Work: Autonomous driving for cars is a very popular theme amongst a multidisciplinary research community [2], [3]. In particular, trajectory optimisation for race car simulators has received a constant attention, with the production of research papers and of patents [4], [5], [6].

In the robotic community, the path of shortest length has been investigated by Dubins[7], by Reeds&Sheep [8] and by Alexander et al. [9] for car–like vehicles, and by other authors for differential drive robots with limited field of

[1]T. Rizano, D. Fontanelli and L. Palopoli are with the Dep. of Information Engineering and Computer Science, University of Trento, Via Sommarive 5, Trento, Italy {rizano,fontanelli,palopoli} at disi.unitn.it

[2] L. Pallottino and P. Salaris are with the Interdepartmental Research Center "E. Piaggio", University of Pisa, via Diotisalvi 2, Pisa, Italy {l.pallottino,paolo.salaris} at centropiaggio.unipi.it
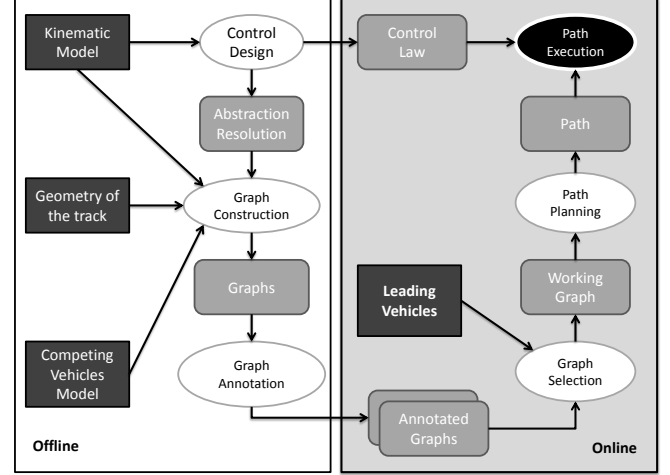
Fig. 1. Overview of the approach

view sensors [10], [11]. The minimum wheel rotation for differential drive robots has been investigated by Lavalle et al. [12]. The minimum time trajectories have been investigated for differential drive [13], omnidirectional [14] and mobile robots with trailers using bounded controls [15] and robots with independently driven wheels [16].

The approach we advocate in this paper builds on the extremal manoeuvres, i.e. that verify the Pontryagin Minimum Principle necessary conditions for optimality [17], to steer the car from one configuration to another in minimum time. Geometric constraints imposed by the boundaries of the track must be taken into account, as in the work of Velenis et al. [18], who analyse different strategies for local path planning (making a bend) based on the dynamic behaviour of the suspension and of the tyre. For the car, we adopt a kinematic model described in the existing literature [19].

Finally, a very important inspiration for this work is the technique usually referred to as "discrete abstraction" [20], [21], [22], [23], [24], [25]. whereby a system with dense state is translated into a discrete system (essentially a state machine) to simplify planning and verification of properties.

b) Overview on the approach: The approach of this paper is pictorially represented in Figure 1, where squares denote information, ovals denote steps and squares with rounded corners denote results of the steps. Part of the steps are carried out offline and part are carried out on-line. The backbone of our approach is an abstraction that allows us to reformulate the path planning problem in a discrete graph–based setting. Such abstraction is constructed starting from way lines that are distributed on the track. By quantising the

euvres that steer the vehicle from the initial to the final cell. The sequence of manoeuvres that solves this local path planning problem is discussed in a different paper [26]. Each arc has a cost given by the time required by the sequence of manoeuvres that executes the transition between the two cells. If the car runs in isolation, a suboptimal motion plan is constructed by solving on-line the shortest path problem on the graph. The control law guarantees the correct execution of the planned path: if the plan includes a transition from cell A to cell B, it ensures that if the initial state is within cell A then its final state will be within cell B. As a consequence, the control law determines the minimum resolution of the quantisation.

The presence of multiple vehicles is dealt with by constructing a graph for each vehicle on the track (the graph depends on the kinematic parameters of the vehicle). When a slower vehicle is in a leading position, the motion planner operates with two graphs (one for each vehicle). The arcs of the two graphs are considered pairwise: whenever a pair of arcs taken by the two vehicles could lead to a collision, it is annotated with the minimum inter-arrival time that permits to avoid the accident. The motion plan algorithm is modified to make sure that the shortest path selected on the graph does not violate such temporal constraints.

The paper is organised as follows. In Section II we introduce the most important concepts the paper revolves around and propose a formal statement for the problem. In Section III we discuss our graph based discrete abstraction of the problem. In Section IV, we discuss the construction of arc between the nodes summarising the results of the local path planning problem [26]. In Section V, we support our technique by a large set of simulation results. Finally, In Section VI, we offer our conclusions and announce future directions of work.

II. Problem Definition

In this section we first introduce the race car model and the geometric model of the track. Then, we formalise the global motion planning problem addressed in the next sections.

A. Car Model

A realistic model of the car, comprising the kinematics and the most important dynamic effects has been proposed by Hoffman et al. [19] and used in our work, adding aerodynamical drag forces, on local motion planning [26]. As shown by Hoffman et al., the motion planning problem can be approached restricting to the kinematics of the vehicle as far as an appropriate controller is applied to the dynamics. For this reason, we will simply focus here on the kinematics.

Let $\langle W \rangle = \{O_w, X_w, Y_w, Z_w\}$ be a right–handed fixed reference frame (see Figure 2). The configuration of the vehicle is described by $\xi(t) = (x(t), y(t), \theta(t), \varphi(t))$, where $p(t) = (x(t), z(t))$ is the position in $\langle W \rangle$ of the rear–wheel
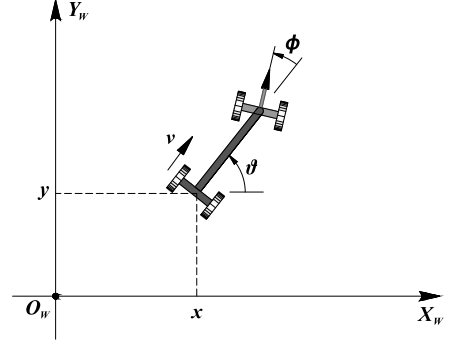

Fig. 2. Mobile robot and system coordinates

axis midpoint, $\theta(t)$ is the orientation of the vehicle with respect to the X_w axis and $\varphi(t)$ is the steering angle of the front wheel with respect to the vehicle, see Figure 2. Let $v(t)$ be the linear velocity of rear–wheel axis midpoint. Choosing φ and the traction acceleration a as control inputs, the car–like model adopted in this work is

$$\begin{bmatrix} \dot{x} \\ \dot{y} \\ \dot{\theta} \\ \dot{v} \end{bmatrix} = \begin{bmatrix} v \cos \theta \\ v \sin \theta \\ \frac{\tan \varphi}{L} v \\ a - c_a v^i \end{bmatrix}, \quad (1)$$

where the term

$$F_{aero} = m c_a v^i$$

accounts for the aerodynamic force, m is the total mass of the vehicle, $c'_a = m c_a$ is the aerodynamic drag coefficient and $i = 1$ for air laminar motion, while $i = 2$ for turbulent motion. The parameter L is the distance between the wheel axes. The control inputs are constrained in the sets $\varphi \in [-\overline{\varphi}, \overline{\varphi}]$ and $a \in [\underline{a}, \overline{a}]$ (with $\underline{a} < 0 < \overline{a}$, i.e. maximum braking and acceleration applicable actions), respectively. Since we are not interested in parking or docking manoeuvres we assume the velocity $v \in [\underline{v}, \overline{v}]$ with $\underline{v} > 0$. For practical reasons, the maximum velocity is assumed to be less than $\frac{\overline{a}}{c_a}$ for the laminar regime, and can be identified numerically for the turbulent regime.

For bending paths, the constraint imposed by the maximum lateral acceleration a_l of the vehicle is additionally considered. In fact, given a planned curve to travel, we have

$$v^2 \leq a_l R \quad (2)$$

where $R = \frac{L}{\tan \varphi}$. Notice that both a_l and b are function of the aerodynamic load as well as the tires grip (which depends on the ground characteristics, e.g. dry or wet asphalt, off road, etc.), and provides a constraint depending on the state variable v and the control variable φ.

B. Track Description

The track contains straight sectors and turn sectors. A *straight sector* contains a straight line and is modelled as two parallel lines (B_r and B_l) at distance W, while the *turn sector* contains a left or a right bend consisting of a sequence of two (short) parallel lines L_{in} (the approach to the bend),
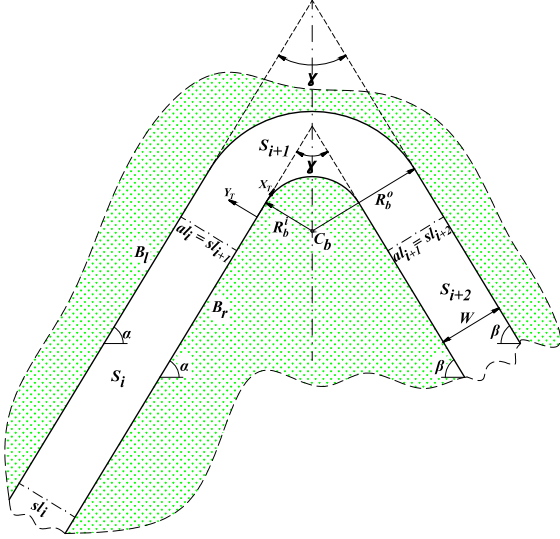
Fig. 3. Circuit sectors.

two concentric arcs of circle (the bend) and other two (short) parallel lines L_{out} (the exit of the bend).

The bend boundaries (B_r and B_l) are characterised by the centres C_b and the radii R_b, see Figure 3. In particular, R_b^i refers to the inner boundary, while R_b^o to the outer where $R_b^o - R_b^i = W$. The angle γ is the characteristic curve angle, i.e. the angle between the two straight line segments of each boundary. The angles α and $\pi - \beta$ are the orientations of L_{in} and L_{out} w.r.t. the orthogonal of the γ bisector, respectively. The entire track is hence a sequence of sectors S_i where each sector can be of straight or turn type. Furthermore, a sector S_i is characterised by two segments (way-lines) of width W orthogonal to the sector borders. The way-line sl_i is the segment at the beginning of sector S_i and is considered as the starting line of S_i and the final way-line of S_{i-1} while the orthogonal segment al_i at the end of sector S_i is considered as starting line of S_{i+1} and final line of S_i, i.e. $sl_{i+1} = al_i$, see Figure 3. Since in each turn sector the track bend is preceded and followed by a straight stretch of track we may assume that the car will always cross the starting and final lines sl_i and al_i perpendicularly, i.e. $\theta(t_i) = \theta(t_{i+1}) = 0$. Finally, considering a circuit of n_s sectors, we assume $S_{n_s+1} = S_1$. In other words, the track re-starts over at the end of each lap.

C. Reference frames

Let $\langle T \rangle = \{O_T, X_T, Y_T, Z_T\}$ be the *track reference frame* with O_T a point on B_r, X_T tangent to B_r and Y_T pointing towards the left boundary B_l (see Fig. 3). This frame is a Frenet frame attached on the the right boundary of the track. Consider the *manoeuvre initial reference frame* $\langle I_m \rangle$ as $\langle T \rangle$ with O_T placed on the first straight line of B_r and the *manoeuvre final reference frame* $\langle E_m \rangle$ as $\langle T \rangle$ with O_T placed on the second straight line of B_r. Similarly, $\langle I \rangle$ and $\langle E \rangle$ represent the *track initial reference frame* and the *track*

final reference frame, respectively. Notice that, in general, $\langle I \rangle \equiv \langle E \rangle$.

A point $^T p_r$ on X_T has a corresponding point on B_l given by $^T p_l = {}^T p_r + [0, W]^T$. In particular, any point inside the track at the same distance of $^T p_r$ from the curve can be expressed as $^T p = {}^T p_r + k[0, W]^T$, where $0 \leq k \leq 1$. We will denote by $\mathcal{P}$ the set of all points lying inside the track, for which there exist a Frenet frame in which their X coordinate is 0 and their Y coordinate is smaller than W: $\mathcal{P} = \{v | \exists \langle T \rangle \text{ s.t. } {}^T v = k[0, W]^T, 0 \leq k \leq 1\}$.

D. The optimal control problem

The goal of this paper is to find the sequence of manoeuvres that allow the car to complete a generic number of laps in minimum time. Hence, the cost function to be minimised over the track is

$$\int_0^T L(q, a, \varphi) dt, \tag{3}$$

where $L(q, a, \varphi) = 1$.

Given a configuration $\xi = [x, y, \theta, \varphi]$, $p_r = [x, y]$ denotes the position of the midpoint of the rear axle, while the midpoint of the front axle is $p_f = (x + L\cos\theta, y + L\sin\theta)$. Let Σ be the sequence of sectors to be traversed. In other words, Σ is the region of configurations such that the vehicle is inside the track; such configuration requires that both p_f and p_r are inside the track: $\Sigma = \{\xi | p, p_f \in \mathcal{P}\}$. Moreover, at time t_i the configuration is supposed to lie on $sl_i = \{\xi | {}^T p_f = k[0, W]^T, 0 \leq k \leq 1, p \in \mathcal{P}\}$, i.e. the starting region of sector S_i. If the circuit has n_s sectors and the number of laps is n_l, Σ comprises n_l ordered sequences of n_s sectors. We can now state the *Track Optimal Problem*:

Problem 1: Track Optimal Problem

$$\min_{a(t),\varphi(t)} \sum_{i=0}^{n_l n_s - 1} \int_{t_i}^{t_{i+1}} dt, \text{ subject to}$$

$$q(t) \text{ solution of } (1),$$
$$\xi(t) \in \Sigma, \xi(t_i) \in sl_i, \forall i = 1, \ldots, n_l n_s$$
$$v(t) \in [\underline{v}, \overline{v}]$$
$$v^2(t) \tan\varphi \leq a_l L$$
$$a(t) \in [\underline{a}, \overline{a}]$$
$$\varphi(t) \in [-\overline{\varphi}, \overline{\varphi}].$$

At time $t_1 = 0$ the initial configuration is supposed to lie on $sl_1 = \{\xi | {}^I p_f = k[0, W]^T, 0 \leq k \leq 1, p \in \mathcal{P}\}$, with $sl_1 = sl_{n_s}$ while $sl_{n_l n_s} = \{\xi | {}^E p_f = k[0, W]^T, 0 \leq k \leq 1, p \in \mathcal{P}\}$. Such problem refers to a single car requiring that the kinematic model, and the different geometric and dynamic constraints are respected. An additional requirement (addressed in the final part of the paper) is that no collision happen with the other vehicles, assuming that they also adopt a time optimal strategy.

III. GRAPH–BASED CIRCUIT REPRESENTATION

A solution to Problem 1 can be found using nonlinear optimal control theory or MPC–like tools. However, a complete characterisation of the solution as well as the high computational cost required for such solutions (in particular, if the algorithm has to be executed on an autonomous robotic

vehicle endowed with limited computing resources) leads to a more manageable solution. In this paper, we decide to represent the track using a discrete abstractions of its possible configurations. More precisely, a possible representation of the track can be given in terms of a graph.

Consider a discretisation of dimension d_w of the width W of the track. The d_w points laying on the orthogonal lines sl_i are the starting points of S_i and final points of S_{i-1}, while the points laying on the orthogonal lines $al_i = sl_{i+1}$ are considered as starting points of S_{i+1} and ending points of S_i. Considering a circuit of n_s sectors the total number of nodes in the graph is $(n_s + 1)d_w$ where we assume $S_{n_s+1} = S_1$. In other words, to close the circuit, nodes of S_1 are considered twice: one as initial nodes of S_1 and one as final nodes of S_{n_s}. The arcs of the graph represent the sectors and hence any node on sl_i is connected through an arc to any nodes on $sl_{i+1} = al_i$. Therefore, the graph consists of $n_s d_w^2$ arcs. A cost $c_{i,j}$ equal to the minimum time required to go from a node i to j is associated to arc (i,j). However, combination of minimum time sectors does not results in a minimum time lap. Nevertheless, it is sufficient to reach al_i from sl_i along S_i in the minimum time *and* with the maximum velocity in al_i. In other terms, it is sufficient for sector S_i to minimise the following cost index

$$\int_{t_i}^{t_{i+1}} L(q, a, \varphi)dt, \qquad (4)$$

where $L(q, a, \varphi) = 1$ in each track sector.

Therefore, for a sector S_i, the initial and final configurations assumed by the vehicle are constrained on two lines, defined by $sl_i = \{\xi | {}^{Im}p_f = k[0, W]^T, 0 \le k \le 1, \ p \in \mathcal{P}\}$ and $al_i = \{\xi | {}^{Em}p_f = k[0, W]^T, 0 \le k \le 1, \ p \in \mathcal{P}\}$, respectively. Hence, in place of Problem 1 the following set of optimal control problems can be defined

Problem 2: Optimal Control Problem

$$\min_{a(t),\varphi(t)} \ t_{i+1} - t_i, \text{ subject to}$$

$$q(t) \text{ solution of } (1),$$
$$\xi(t) \in \Sigma, \xi(t_i) \in sl_i, \xi(t_{i+1}) \in al_i$$
$$v(t_i) = v_i^i, \ v(t_{i+1}) = v_i^f$$
$$v(t) \in [\underline{v}, \overline{v}]$$
$$v^2(t)\tan \varphi \le a_l L$$
$$a(t) \in [\underline{a}, \overline{a}]$$
$$\varphi(t) \in [-\overline{\varphi}, \overline{\varphi}].$$

It is worth noting that, with respect to Problem 1, the optimal solution in each sector must be determined given the speeds v_i^i and v_i^f at the beginning and at the end of the sector S_i. Based on the principle of optimality, those constraints are introduced to obtain the concatenation of optimal solutions of Problem 2 for each sector that is the optimal solution of Problem 1.

To solve this problem, a discretisation of dimension d_v of the speed space v is also provided, so that each point on sl_i can be crossed at d_v different speed values. Hence, $d_w d_v$ nodes are associated to any initial segment $sl_i, \forall i$. With n_s sectors, the total number of nodes in the graph is now $(n_s +$

$1)d_w d_v$, while the number of arcs turns to $n_s d_w^2 d_v^2$. More formally, a node k is represented by a triplet $k = (S_k, p_k, v_k)$ where S_k is the sector, p_k is one of the d_w position of the point represented by k on sl_k, and v_k is one of the d_v speeds pertaining to a point p_k. Given nodes i and j the arc (i, j) belongs to the graph if and only if $S_j = S_{i+1}$. The cost $c_{i,j}$ associated to arc (i, j) is equal to the minimum time required to go from a node i to j (with the corresponding speeds). Hence, the cost $c_{i,j}$ equal to the solution of the minimum time Problem 2 with $v(0) = v_i$, $v(T) = v_j$ from p_i to p_j.

In order to apply standard shortest paths algorithms such as Dijkstra [27] it is necessary to introduce two nodes. An initial node I and a final node F and all arcs (I, i) and (j, F) where $S_i = S_1$ and $S_j = S_{n_s+1}$. The associated costs are null, i.e. $c_{Ii} = c_{jF} = 0$. With the introduction of such nodes algorithm such as Dijkstra provide shortest path from any node of the graph to F. In particular, from any point on S_1 to S_{n_s+1}.

Determining the minimum path from the nodes associated to the circuit starting line to the same set of nodes considered ad nodes of the arrival line the minimum time lap can be determined with the associated sequence of manoeuvres described in previous section. It is worth noting that with this approach the best trajectory for the qualifying lap is determined. If we are interesting in 2 or more minimum time laps the graph must be extended duplicating the nodes and the arcs. Indeed, a graph associated to two laps on the circuit must be taken into account considering S_i of the first lap different from S_{n_s+i} of the second lap. A graph with $2n_s d_w d_v$ nodes and $2(n_s + 1)d_w^2 d_v^2$ arcs is hence considered. The same construction of nodes I and F with associated arcs and cost can be followed and the optimal trajectory for a generic lap of the race can be found.

The discretisation of the width W and of the speed v will obviously provide a suboptimal solution. However, a finer quantisation provides a better solution but with the drawback of having a huge graph and hence a higher computational costs.

A. Avoiding obstacles

The graph abstraction can easily be applied to account for the presence of other (slower) cars in front of the vehicle that are not cooperative (i.e., do not facilitate the overtake). This is done working with two graphs (one for each vehicle). The first step is to creare a relations between the arcs of the two graphs. A pair of arcs belongs to the relation if it is possible to have a collision when the arcs are taken with a wrong timing. Suppose that vehicle A follows and vehicle B leads, and assume that a couple of arcs (a_A, b_A) (belonging to the graph of A), (a_B, b_B) (belonging to the graph of B) potentially lead to a collision. Using simple kinematic considerations (which we do not detail for the sake of brevity) it is possible to find a minimum interarrival time $\tau_{(a_A, b_A) \to (a_B, b_B)}$ such that if A enters the arc (a_A, b_A) after a time $\tau_{(a_A, b_A) \to (a_B, b_B)}$ elapsed since B entered (a_B, b_B), then the collision is avoided.

The algorithm for path planning can be modified as follows. At the beginning A detects the position of B and assumes that it will use its graph for a minimum time path planning. As a consequence it knows the position of B in its graph for any time in the future. As a first step A finds the shortes path using its graph and annotates it with the time each node is reached. If the path thus found contains an arc (a_A, b_A) that is in relation with another arc (a_B, b_B) that B used, the algorithm checks if the minimum interarrival time is respected. If not the arc (a_A, b_A) is removed from the graph and the Djikstra algorithm is repeated on the updated graph. These steps are repeated until a "clear" path is found. The algorithm also checks when A becomes the leader. From that point on the arcs that are possibly removed during the algorithm execution are reinserted and A can use all of its graph.

IV. TOWARDS AN OPTIMAL SOLUTION

Instrumental to the construction of the graph is the solution of the local planning problem: how to steer the car from a configuration $\xi(t_I)$ where each configuration is define by position and velocity. This problem has been addressed in a different paper [26]. We report here the essential results for the sake of completeness.

The solution of the local planning problem requires: 1) identification of an alphabet of elementary maneuvers (extremals) arising form the application of Pontryagin Minimum Principle (see e.g. [17] and [28] for the constraints on the state and control variables); 2) identification of the optimal concatenation of extremals; 3) computation of the optimal free parameters.

Computation of the extremals

The application of the Pontryagin Minimum Principle to the kinematic–model disregarding the constraints on the configurations ξ that impose that the vehicle is on the track leads to the following set of extremals:

1) Straight line $\mathcal{S}$, travelled with any velocity profile, compatible with the maximum and minimum accelerations, i.e. $\overline{a}$ and $\underline{a}$, respectively;
2) Circular curve $\mathcal{C}_{\overline{r}}$ travelled with constant maximum velocity $\overline{v_x}$; the radius is fixed to the maximum value $\overline{r}$ that is compatible with the constraint on the lateral acceleration: $\overline{r} = \overline{v_x}^2/a_l$, where a_l is the maximum lateral acceleration;
3) Circular curve $\mathcal{C}_{\underline{r}}$ travelled at (possibly time–varying) velocity $\underline{v_x} \leq v_x \leq \sqrt{\frac{a_l L}{|\tan(\overline{\varphi})|}} = v_{x\varphi}$; the radius is fixed to the minimum possible value allowed by the vehicle : $\underline{r} = \frac{L}{|\tan(\overline{\varphi})|}$;
4) Variable radius curves $\mathcal{V}_{\overline{a}}$ and $\mathcal{V}_{\underline{a}}$ executed with maximum or with minimum acceleration respectively, which always verify the relation $v_x^2|\tan(\varphi)| = a_l L$.

Optimal sequences of extremals

The computation of the optimal sequence of extremals is very difficult to be addressed geometrically due to the nature of the variable radius curves $\mathcal{V}_{\overline{a}}$ and $\mathcal{V}_{\underline{a}}$. In the paper [26] we identified a heuristic solution derived from geometric considerations on simplified cases, that are:

- For straight sector:
$$\mathcal{S}\mathcal{T}_{\underline{a}}\mathcal{C}_{\underline{r}}\mathcal{S}\mathcal{C}_{\underline{r}}\mathcal{T}_{\overline{a}}\mathcal{S};$$
- For a turn sector:
$$\mathcal{S}\mathcal{T}_{\underline{a}}\mathcal{C}_{\underline{r}}\mathcal{S}[\mathcal{C}_{B_r}|\mathcal{C}_{B_l}]\mathcal{S}\mathcal{C}_{\underline{r}}\mathcal{T}_{\overline{a}}\mathcal{S}.$$

In this case we make use of the $\mathcal{T}$ manoeuvres, i.e.,
$$\mathcal{T}_{\underline{a}} = \mathcal{C}_{\overline{r}} \circ \mathcal{V}_{\underline{a}},$$
$$\mathcal{T}_{\overline{a}} = \mathcal{V}_{\overline{a}} \circ \mathcal{C}_{\overline{r}}.$$

In addition, we can have either a curve with maximum or with minimum radius $[\mathcal{C}_{B_r}|\mathcal{C}_{B_l}]$ of the sector to account for the geometry of the track. Notice that each extremal can be of zero length.

Additionally, each curved maneuver has a superscript equal to $-$ or $+$ depending on the fact that the curve turns in the clockwise or counterclockwise direction (thus decreasing or increasing the value of the angle θ according to the right–hand rule).

Computation of the parameters

Each extremal is characterised by a set of parameters. For instance a straight line $\mathcal{S}$ is associate with its length, initial velocity, final velocity. When the extremals are concatenated together some of the choices become bound by the previous extremal in the sequence. For instance, if a straight line is followed by a bend, the initial velocity of the bend will have to be equal to the final velocity of the straight line. Similar constraints holds for the trajectory tangents in the concatenation point. Additional constraint are obviously imposed by the initial and the final configuration. Nevertheless some of the parameters remain open and can be considered as decision variables in an optimisation procedure aiming for the minimum time solution. Efficient solution strategies for this challenging problem are still under investigation. In the simulation section below we report solutions obtained with a combination of simulated annealing and gradient discent.

The optimal solution of this problem is the weight of the arc connecting the two nodes of the graph.

V. SIMULATIONS

The simulations are performed with two cars running on a track for 5 laps. Both cars have equal size ($3.5m$ in length and $1.8m$ wide) and same maximum and minimum accelerations $\overline{a} = 34.5m/s^2$ and $\underline{a} = -20m/s^2$ respectively. A minimum curvature radius of $15m$ is imposed on both cars to account for their minimum turning radius.

The cars have different viscous friction parameter (b) which in turn affects their maximum speed ($\overline{v}$) and different maximum lateral acceleration (a_l). For the first car: $b = 0.45$, $\overline{v} = 0.95\frac{\overline{a}}{b} = 72.89m/s$, and $a_l = 67.5m/s^2$, while for the second car: $b = 0.4$, $\overline{v} = 82m/s$, and $a_l = 60m/s^2$. Such parameters have been taken from a race car simulator[1].

The control laws used in the following simulations in order to track each elementary manoeuvre of the simplified scenario, i.e. straight line and circular arcs, are those provided by [19].
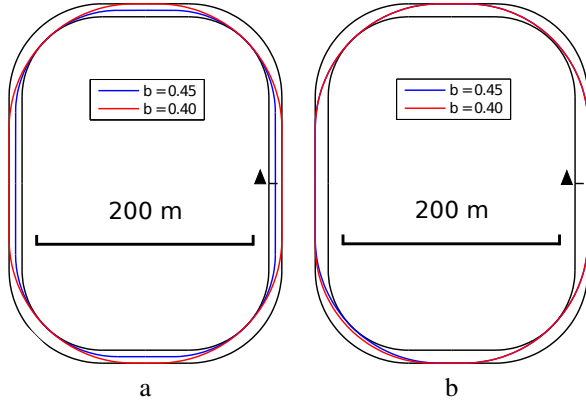
---

[1]http://torcs.sourceforge.net/

Fig. 4. a) Trajectories from two independent simulations of a car with different viscous frictions (b) b) trajectories for reduced lateral acceleration $a_l^b = a_l^a/2$

car id	road	d_w	d_v	lap time (s)		
				1	2 - 4	5
1	dry	3	6, 12	14.20	13.14	13.14
		6	6, 12	14.12	13.04	13.04
2	dry	3	6, 12	13.27	11.84	11.84
		6	6 ,12	13.27	11.84	11.84

TABLE I

TIME TO COMPLETE EACH ONE OF THE 5 LAPS WITH DIFFERENT
CHOICES OF QUANTISATION AND ROAD CONDITIONS FOR TRACK IN
FIGURE 4.

A. Static Graph

The optimal trajectory of a vehicle is affected by the choice of parameter values (maximum velocity, maximum acceleration, length of the track, etc). Therefore, the graph differs for different set of parameter values. Figure 4.a shows the result of two independent simulations where the viscous friction of the car differs.

Figure 4.b shows the result of a similar simulation with a reduced (half) value of a_l. Due to the reduced maximum lateral acceleration, both cars cannot find a trajectory where they can reach the maximum velocity along the curve.

Table I offers simulation results for different choices of quantisation parameters. Table I is related to the track shown in Figure 4 where the length of rectilinear paths are $160m$ and $80m$ while the curvature radii are $80m$.

As shown in Table I, the time to complete a lap of Car 1 does not change by increasing the dimension d_v of the velocity space. This is due to the fact that both cars can reach their maximum velocity and hold the velocity until the end of the lap. However, increasing the dimension d_w enables Car 1 to reduce the lap time by having a trajectory with shorter length. Conversely, the time to complete a lap of Car 2 on a dry road condition is unaffected by changing d_w or d_v. This is due to the fact that there is only one trajectory that enables Car 1 to reach the highest feasible velocity.

B. Obstacle Avoidance

A more interesting example is the one where we have more than one car on the track. Similarly to the previous
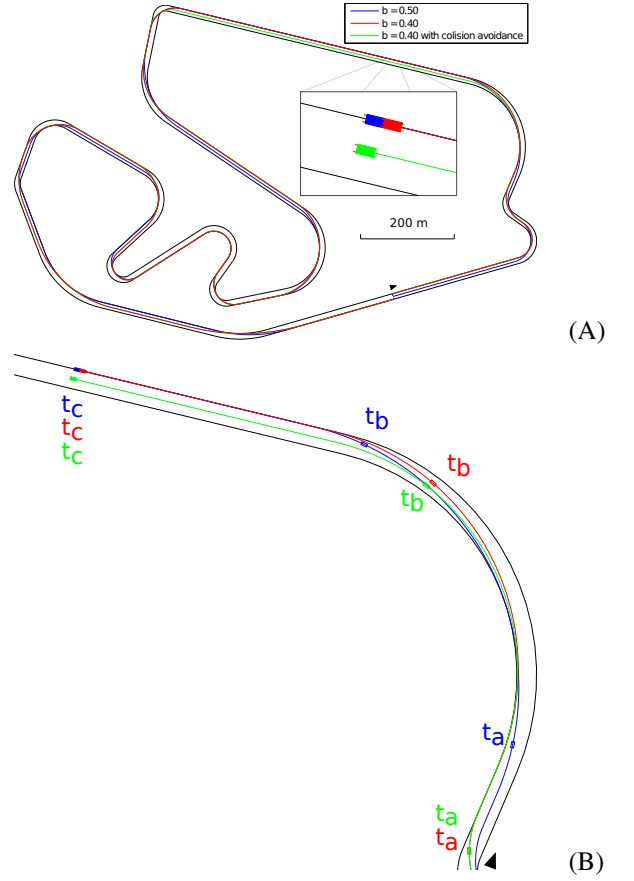

(A)

(B)

Fig. 5. (A) Trajectories of two cars, on the Interlagos-like circuit, with different viscous friction b parameters before graph pruning (red) and after pruning (green), (B) Closeup of the trajectories of two cars, on the Interlagos circuit, with different viscous friction b parameters before graph pruning (red) and after pruning (green)

example a graph of feasible trajectories is built for each car with the parameters given in the beginning of this section. Two optimal trajectories are produced for each graph. We did this for a track inspired to Interlagos Formula 1 track, which was divided in 14 sectors. Figure 5.(A) shows the two sets of optimal trajectories. The blue line represents the trajectory of car 1 and the red line represents the trajectory of car 2 before applying the collision avoidance algorithm described in Section III-A, while the green line represents the new optimal trajectory of car 2 with collision avoidance. The inset shows the configurations of the two cars at time t_i before and after the path planning. A more detailed view is depicted in Figure 5.(B) where we took a snapshot of the cars configurations at three different times.

We applied the path planning algorithm on the two graphs and removed edges from the graph of the car at the rear if the two cars collide along the trajectories represented by the edges. The algorithm produces two new optimal paths where the two cars do not collide as shown in Figure 5.(B).

VI. CONCLUSION

In this work, we have proposed a motion planning technique that applies to robot race cars. By using a discrete abstraction, we are able to generate a motion plan that

optimises the completion time of the race in a short time, both when the car runs in isolation and when it has to overtake a slower car.

The leading car is not cooperative but is not competitive as well (it does not try to obstruct the overtake manoeuvre). An obvious future direction is to implement game theoretic strategies that allows the car to overtake even if the leader opposes. Another important area for future work will be on the implementation of this idea on a robotic vehicle.

From the control theoretical point of view, the authors are attracted by the study of the apparent ideal line that the vehicles seem to converge to during the race.

REFERENCES

[1] D. Kogan and R. Murray, "Optimization-based navigation for the darpa grand challenge," in *Conference on Decision and Control (CDC)*, 2006.

[2] M. B. Issue edited by: Karl Iagnemma, "Special issue: Special issue on the darpa grand challenge, part 1," *Journal of Field Robotic*, vol. 23, no. 8, p. 461652, 2006.

[3] S. S. Issue edited by: Martin Buehler, Karl Lagnemma, "Special issue: Special issue on the 2007 darpa urban challenge, part i," *Journal of Field Robotic*, vol. 25, no. 8, p. 423566, 2008.

[4] M. Tipping, M. Hatton, and R. Herbrich, "Racing line optimization," 2011, u.S. Patent 20110137629.

[5] M. Botta, V. Gautieri, D. Loiacono, and P. Lanzi, "Evolving the optimal racing line in a high–end racing game," in *Computational Intelligence and Games (CIG), 2012 IEEE Symposium on*, 2012, pp. 108–115.

[6] D. Casanova, R. Sharp, and P. Symonds, "Minimum time manoeuvring: The significance of yaw inertia," *Vehicle System Dynamics*, vol. 34, no. 2, pp. 77–115, 2000.

[7] L. E. Dubins, "On curves of minimal length with a constraint on average curvature, and with prescribed initial and terminal positions and tangents," *American Journal of Mathematics*, pp. 457–516, 1957.

[8] J. A. Reeds and L. A. Shepp, "Optimal paths for a car that goes both forwards and backwards," *Pacific Journal of Mathematics*, pp. 367–393, 1990.

[9] J. Alexander, J. Maddocks, and B. Michalowski, "Shortest distance paths for wheeled mobile robots," *Robotics and Automation, IEEE Transactions on*, vol. 14, no. 5, pp. 657 –662, oct 1998.

[10] P. Salaris, D. Fontanelli, L. Pallottino, and A. Bicchi, "Shortest paths for a robot with nonholonomic and field-of-view constraints," *IEEE Transactions on Robotics*, vol. 26, no. 2, pp. 269–281, April 2010.

[11] P. Salaris, L. Pallottino, and A. Bicchi, "Shortest paths for finned, winged, legged, and wheeled vehicles with side-looking sensors," *The International Journal of Robotics Research*, vol. 31, no. 8, pp. 997–1017, 2012.

[12] H. Chitsaz, S. M. LaValle, D. J. Balkcom, and M. Mason, "Minimum wheel-rotation for differential-drive mobile robots," *The International Journal of Robotics Research*, pp. 66–80, 2009.

[13] H. Wang, Y. Chan, and P. Souères, "A geometric algorithm to compute time-optimal trajectories for a bidirectional steered robot," *IEEE Transaction on Robotics*, vol. 25, no. 2, pp. 399–413, 2009.

[14] D. Balkcom and M. Mason, "Time-optimal trajectories for an omni-directional vehicle," *The International Journal of Robotics Research*, vol. 25, no. 10, pp. 985–999, 2006.

[15] M. Chyba and S. Sekhavat, "Time optimal paths for a mobile robot with one trailer," in *Intelligent Robots and Systems, 1999. IROS '99. Proceedings. 1999 IEEE/RSJ International Conference on*, vol. 3, 1999, pp. 1669 –1674 vol.3.

[16] D. B. Reister and F. G. Pin, "Time-optimal trajectories for mobile robots with two independently driven wheels," *The International Journal of Robotics Research*, vol. 13, no. 1, pp. 38–54, 1994.

[17] L. Pontryagin, V. Boltyanskii, R. Gamkrelidze, and E. Mishchenko, *The mathematical theory of optimal processes*. Interscience Publishers New York, 1962.

[18] E. Velenis and P. Tsiotras, "Minimum time vs maximum exit velocity path optimization during cornering," in *Industrial Electronics, 2005. ISIE 2005. Proceedings of the IEEE International Symposium on*, vol. 1, 20-23, 2005, pp. 355 – 360.

[19] G. M. Hoffmann, C. J. Tomlin, M. Montemerlo, and S. Thrun, "Autonomous automobile trajectory tracking for off-road driving: Controller design, experimental validation and racing," in *American Control Conference, 2007. ACC'07*. IEEE, 2007, pp. 2296–2301.

[20] A. Girard and G. Pappas, "Approximate bisimulation relations for constrained linear systems," *Automatica*, vol. 43, no. 8, pp. 1307–1317, 2007.

[21] G. Pola, A. Girard, and P. Tabuada, "Approximately bisimilar symbolic models for nonlinear control systems," *Automatica*, vol. 44, no. 10, pp. 2508–2516, 2008.

[22] S. Pancanti, L. Pallottino, D. Salvadorini, and A. Bicchi, "Motion planning through symbols and lattices," in *Robotics and Automation, 2004. Proceedings. ICRA'04. 2004 IEEE International Conference on*, vol. 4. IEEE, 2004, pp. 3914–3919.

[23] M. Prandini, J. Hu, J. Lygeros, and S. Sastry, "A probabilistic approach to aircraft conflict detection," *Intelligent Transportation Systems, IEEE Transactions on*, vol. 1, no. 4, pp. 199–220, 2000.

[24] R. Asaula, D. Fontanelli, and L. Palopoli, "Safety provisions for human/robot interactions using stochastic discrete abstractions," in *Proc. IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2010, pp. 2175–2180.

[25] R. Passerone, J. R. Burch, and A. L. Sangiovanni-Vincentelli, "Refinement preserving approximations for the design and verification of heterogeneous systems," *Formal Methods in System Design*, vol. 31, no. 1, pp. 1–33, August 2007.

[26] T. Rizano, D. Fontanelli, L. Palopoli, L. Pallottino, and P. Salaris, "Local motion planning for robotic race cars," in *Conference on Decision and Control (CDC)*, 2013, submitted. [Online]. Available: http://www.centropiaggio.unipi.it/sites/default/files/CDC13-LocalPlanning.pdf

[27] E. Dijkstra, "A note on two problems in connexion with graphs," *Numerische mathematik*, vol. 1, no. 1, pp. 269–271, 1959.

[28] A. Bryson and Y. Ho, *Applied optimal control*. Wiley New York, 1975.