Feasible trajectory planning algorithm for a skid-steered tracked mobile robot subject to skid and slip phenomena

V. A. Nardi A. Ferraro V. Scordamaglia

DIIES - Università degli Studi di Reggio Calabria - Reggio Calabria(RC) -ITALY

Università degli Studi **Mediterranea** di Reggio Calabria

MMAR 2018

AUGUST 27 - 30, 2018

AMBER BALTIC HOTEL
MIEDZYZDROJE
POLAND

23RD INTERNATIONAL CONFERENCE
ON METHODS AND MODELS
IN AUTOMATION AND ROBOTICS

Introduction

Summary

- This paper proposes an approach for the motion planning of a constrained skid-steered tracked mobile robot under the hypothesis of non-negligible skid and slip phenomena
- Operating environment is discretized with a finite dimensional grid. Then, a weighted graph is defined whose nodes are the above mentioned grid points, and whose arcs denote the trajectory segments
- A modified A* shortest path search algorithm is then proposed to find a trajectory, in terms of succession of arcs, connecting starting and ending nodes. Trajectory feasibility is guaranteed by recurring to set-based arguments
- In order to show the effectiveness of the proposed approach, some numerical examples are finally discussed

Mathematical Modelling

- Be $q = \left[x\, y\, \theta\right]^T$ the vector of coordinates for a skid-steered tracked mobile robot expressed in a given inertial reference frame **E**.
- At low velocities a a kinematic model can be used. The classical first-order kinematic model is

$$\dot{q} = G(q) \cdot \tilde{u} \tag{1}$$

being

$$G(q) = \begin{bmatrix} \cos\theta & 0 \\ \sin\theta & 0 \\ 0 & 1 \end{bmatrix} \tag{2}$$

and $\tilde{u} = \left[\tilde{V}\ \tilde{\omega}\right]^T$ where $\tilde{V}$ ($\tilde{\omega}$) denotes the *effective* forward (rotational) velocity.
- According to the reference coordinates the kinematic relation between $\tilde{u}$ and the vector of effective angular velocities of the tracks sprockets $\tilde{w} = \left[\tilde{w}_R\ \tilde{w}_L\right]^T$ is:

$$\tilde{u} = J \cdot \tilde{w} \tag{3}$$

being

$$J = \begin{bmatrix} R/2 & R/2 \\ R/d & -R/d \end{bmatrix} \tag{4}$$

where R represents the radius of track sprocket and d the distance between the two tracks.

- During rotation a skid-steered tracked vehicle experiences both skidding (inner wheel) and slipping (outer wheel) effects.
- The skidding and slipping effects can be modelled as terrain-dependent possibly time-varying friction coefficients μ_R and μ_L for right and left track respectively.
- A kinematic relation between $\tilde{w}$ and the controlled tracks sprockets angular velocities $w = \begin{bmatrix} w_R & w_L \end{bmatrix}^T$ is

$$\tilde{w} = H(\mu) \cdot w \tag{5}$$

where

$$H(\mu) = \begin{bmatrix} \mu_R & 0 \\ 0 & \mu_L \end{bmatrix} \tag{6}$$

and $\mu = \begin{bmatrix} \mu_R & \mu_L \end{bmatrix}^T$.

- Finally, by assembling (1)-(5) the following nonlinear kinematic model representing the motion of a skid-steered tracked robot in presence of skidding and slipping effects is considered

$$\dot{q} = G(q) \cdot J \cdot H(\mu) \cdot J^{-1} \cdot u = f(q, \mu, u) \tag{7}$$

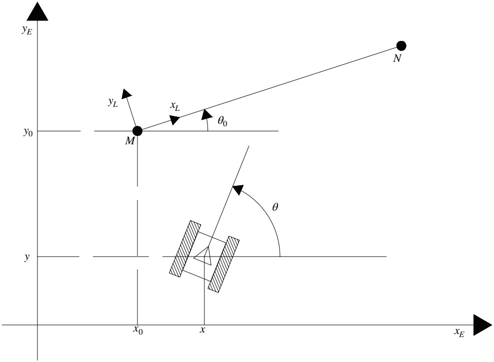
being $u = \begin{bmatrix} V & \omega \end{bmatrix}^T$ the vector where V (ω) denotes the forward (rotational) control velocity.

- Consider a local reference system **L** and be $q_0 = \begin{bmatrix} x_0 & y_0 & \theta_0 \end{bmatrix}^T$.
- The following transformation from **E** to **L** holds

$$q_L = R_E^L(\theta_0) \cdot (q - q_0) \tag{8}$$

being

$$R_E^L(\theta_0) = \begin{bmatrix} cos(\theta_0) & sin(\theta_0) & 0 \\ -sin(\theta_0) & cos(\theta_0) & 0 \\ 0 & 0 & 1 \end{bmatrix} \tag{9}$$



Notations

- Let
$$T^D(\cdot) = \begin{bmatrix} q^D(\cdot) & u^D(\cdot) \end{bmatrix} \quad (10)$$
be a desired trajectory expressed in **L** reference system.
- $T^D(\cdot)$ is defined in terms of feasible couples of state and control inputs compliant with non-holonomic constraints (1) over the time window $[0, \hat{t}]$ such that $q^D(0) = \begin{bmatrix} 0 & 0 & 0 \end{bmatrix}$ and $u^D(t) = \begin{bmatrix} V^D & 0 \end{bmatrix}$.
- V^D represents the mobile robot forward velocity.
- The desired state trajectory at the time $t \in \begin{bmatrix} 0, \hat{t} \end{bmatrix}$ is denoted as follows: $q^D(t) = \begin{bmatrix} V^D \cdot t & 0 & 0 \end{bmatrix}^T$.

Discrete LTI representation

- By recurring to classical linearization and discretization arguments the following discrete linear time invariant system, representing trajectory tracking error dynamic, is obtained:
$$e(t_{k+1}) = Ae(t_k) + B\delta u(t_k) + B_D d(t_k) \quad (11)$$
being $e = q - q^D$, $\delta u = u - u^D$, $d = \mu - \mu^D$, being μ^D the nominal value of friction coefficients

Trajectory Tracking Control Design

Constraints

Given the discrete linear time invariant representation of tracking error (11), the following ellipsoidal constraints are considered:

$$e(t) \in \Omega_e, \ \Omega_e \triangleq \{e \in \mathcal{R}^3 : e^T e \leq e_{max}^2\} \tag{12}$$

$$\delta u(t) \in \Omega_u, \ \Omega_u \triangleq \{\delta u \in \mathcal{R}^2 : \delta u^T \delta u \leq u_{max}^2\} \tag{13}$$

Constrained Control Problem (CCP)

Given (11) find the state feedback control action $\delta u(\cdot) = K \cdot e(\cdot)$ fulfilling the prescribed constraints (12)-(13) for all the realizations of external disturbance $d \in \Omega_D$ being

$$\Omega_D \triangleq \{d \in \mathcal{R}^2 : d^T d \leq d_{max}^2\} \tag{14}$$

CCP Solution

1. Compute a stabilizing state-feedback control law $\delta u(\cdot) = Ke(\cdot)$ accomplishing (12) and (13) within the ellipsoidal positively invariant region[1]

$$\Gamma_0 = \{e \in \mathcal{R}^3 : e^T P_0 e \leq 1 \;\; P_0 \geq 0\} \tag{15}$$

2. Define the maximal ellipsoidal d-invariant subset[2]

$$\Gamma_\infty = \{e \in \mathcal{R}^3 : e^T P_\infty e \leq 1 \;\; P_\infty \geq 0\} \subseteq \Gamma_0 \tag{16}$$

$$\text{such that } e(t_k) = \Phi^k e(t_0) + \sum_{h=0}^{k-1} \Phi^{k-1-h} B_D d_h \in \Gamma_0 \tag{17}$$

$\forall t_k \geq 0 \; \forall e(t_0) \in \Gamma_\infty$ and $\forall d(t_j) \in \Omega_D \;, t_j < t_k$

$$\text{being } \; e(t_{k+1}) = \Phi e(t_k) + B_D d(t_k) \tag{18}$$

the closed loop tracking error dynamic with $\Phi = (A + B \cdot K)$.

Remark

The above two step procedure provides a solution (K, Γ_∞) of **CCP** in terms of feedback control action $\delta u(\cdot) = Ke(\cdot)$ within Γ_∞ fulfilling the prescribed constraints (12), (13) for all admissible $d \in \Omega_D$.

[1] V. Kothare et al., Robust constrained model predictive control using linear matrix inequalities 1996

[2] I. Kolmanovsky et al. Theory and computation of disturbance invariant sets for discrete time linear systems 1998

Motion Planning Algorithm

Assumptions

- Assume the trajectory tracking problem was tackled and a solution of *CCP* was provided in terms of a couple (K, Γ_∞)
- Assume a 2D operational scenario $\Delta \subseteq \mathcal{R}^2$.
- Δ is firstly discretized with a finite dimensional grid of feasible positions.
- An undirected weighted graph $\mathcal{G}$ is then defined whose nodes $V \in R^2$ are the above mentioned grid points.

Δ-compatibility

Two nodes $E_1 = (x_1, y_1) \in V$ and $E_2 = (x_2, y_2) \in V$ are Δ-compatible if $\forall \alpha \in [0, 1]$

$$P_\alpha = (x_\alpha, y_\alpha) \in \Delta$$

with $x_\alpha = (1-\alpha)x_1 + \alpha x_2$ and $y_\alpha = \eta_\alpha x_\alpha + \tau_\alpha$ being $\eta_\alpha = \frac{y_1 - y_2}{x_1 - x_2}$ and $\tau_\alpha = \frac{x_1 y_2 - x_2 y_1}{x_1 - x_2}$

Arcs of graph $\mathcal{G}$

- Be $A, B \in V$ two Δ-compatible nodes.
- Arc connecting A and B is denoted as

$$T^{AB}(\cdot) = \begin{bmatrix} q^{AB}(\cdot) & \tilde{u}^{AB} \end{bmatrix} \tag{19}$$

It represents, in a local reference frame centred in A and such that B belongs to robot positive x-axis, the trajectory crossing the segment $\bar{AB}$ at the constant velocity V_{AB} for N_{AB} time steps with a null angular velocity being $q^{AB}(t) = \begin{bmatrix} V_{AB} \cdot t & 0 & 0 \end{bmatrix}$ and $\tilde{u}^{AB} = [V_{AB}\ 0]^T$ solution of (1) over the time horizon $t \in [0, T_s \cdot N_{AB}]$. N_{AB} is the maximum positive integer such that

$$N_{AB} \leq \frac{d_{AB}}{V_{AB} \cdot T_s} \tag{20}$$

Arc Cost

- Arc cost is assumed to be d_{AB} the length of segment $\bar{AB}$.

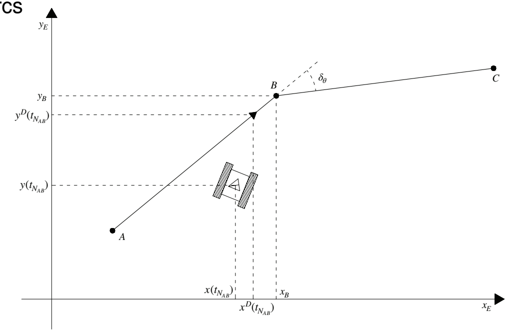
- Given three nodes $A, B, C \in V$ suppose the segments A, B and B, C are both Δ-compatibles. Consider a path including the two adjacent arcs T^{AB} and T^{BC}
- If the robot has to track planned trajectory, a switch from T^{AB} to T^{BC} at switching time $t_{N_{AB}} = T_s \cdot N_{AB}$ is required
- Switching is considered admissible if the following condition is fulfilled:

$$(e(t_{N_{AB}}) + \Pi) \in \Gamma_\infty \tag{21}$$

where $e(t_{N_{AB}})$ is the tracking error at switching time and

$$\Pi = \begin{bmatrix} x_B - x^D(t_{N_{AB}}) & y_B - y^D(t_{N_{AB}}) & \delta_\theta \end{bmatrix}^T \tag{22}$$

- A trajectory is feasible if every switch between trajectory segments is admissible, feasibility is guaranteed by a conservative check involving admissibility of switches between arcs



Lemma 1

Suppose the tracking error at the time instant $e(t_0) \in S_A$ being

$$S_A = \{e \in \Gamma_\infty : e^T P_A e \leq 1,\ \ P_A \geq 0\} \tag{23}$$

The ellipsoidal set

$$S_{N_{AB}} = \{e \in \Gamma_\infty : e^T \Psi e \leq 1,\ \ \Psi \geq 0\} \tag{24}$$

representing the set of allowable tracking error $e(t_{N_{AB}})$ for every admissible realization of the disturbance $d(t_h) \in \Omega_D$ with $h = 0 \cdots (N_{AB} - 1)$ can be obtained by solving the following SDP minimization problem:

$$\min_{\Psi, \tau_0, \cdots \tau_{N_{AB}}} \quad logdet \Psi^{-1} \tag{25}$$

s.t.

$$1 - \sum_{h=0}^{N_{AB}-1} \tau_h d_{max}^2 - \tau_{N_{AB}} \geq 0 \tag{26}$$

$$\begin{bmatrix} -\hat{\Phi}^T \Psi \hat{\Phi} + \tau_{N_{AB}} P_A & -\hat{\Phi}^T \Psi H \\ * & -H^T \Psi H + \sum_{h=0}^{N_{AB}-1} \tau_h J_h^T J_h \end{bmatrix} \geq 0 \tag{27}$$

being τ_h with $h = 0 \cdots N_{AB}$ positive scalars, $\Psi \geq 0$, $\hat{\Phi} = \Phi^{N_{AB}}$, $H = \begin{bmatrix} \Phi^{N_{AB}-1} B_D & \Phi^{N_{AB}-2} B_D & \cdots & B_D \end{bmatrix}$, J_h the matrix of proper dimension such that $J_h \underline{d} = d(t_h)$ being $\underline{d} = \begin{bmatrix} d(t_0)^T & d(t_1)^T & \cdots & d(t_{N_{AB}})^T \end{bmatrix}^T$

Motion Planning Algorithm

Given a solution of Lemma 1, tracking error switch Π is assumed admissible

$$S_{N_{AB}} \bigoplus \Pi \subseteq \Gamma_\infty \tag{28}$$

and then T^{BC} can follow T^{AB} in a feasible trajectory if there exists $\tau > 0$ such that the following LMI holds

$$\begin{bmatrix} 1-\tau-\Pi^T P_\infty \Pi & * \\ -P_\infty \Pi & -P_\infty + \tau P_A \end{bmatrix} \geq 0 \tag{29}$$

Considerations

- All the above considerations can be readily recast into a recursive algorithm.

Numerical Results

Robot Specifications

Consider a skid-steered tracked mobile robot with a radius of track sprocket $R = 8cm$ and a distance between the two tracks $d = 50cm$. The following constraints are considered:

a the terrain-dependent friction coefficients are supposed bounded

$$\mu_R, \mu_L \in (0.7, 1.2) \tag{30}$$

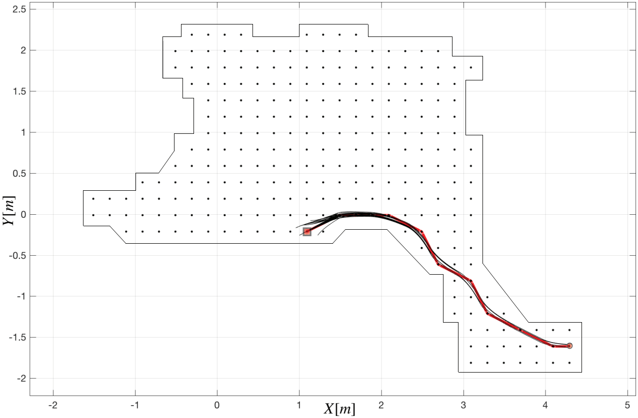
b the following constraints on the control velocities are considered

$$0 \leq V \leq 1.4 \left[m/s\right] \text{ and } -2.15 \leq \omega \leq 2.15 \left[rad/s\right] \tag{31}$$

Assumptions

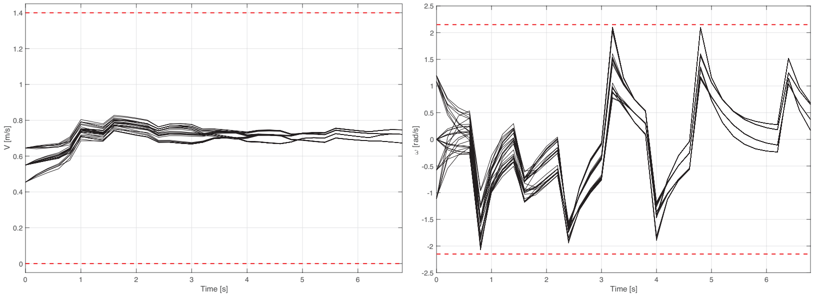
- The two-dimensional space domain Δ was discretized by recurring to a regular grid of $0.2m$ with about 270 resulting nodes.
- In order to construct the graph $\mathcal{G}$ a maximum length of $0.5m$ is assumed for trajectory segment connecting two $\Delta - compatibles$ nodes.
- A nominal forward velocity is assumed constant along the trajectory segments $V^D = 0.7$ [m/s].

Simulation Scenario #1



- Red solid line represents the optimal feasible trajectory resulting from the proposed algorithm connecting starting (square) and ending (circle) nodes
- Black solid lines denote the robot trajectories obtained by means of numerical simulations at varying robot initial pose and constant friction coefficients.

Simulation Scenario #1



Forward and rotational velocities

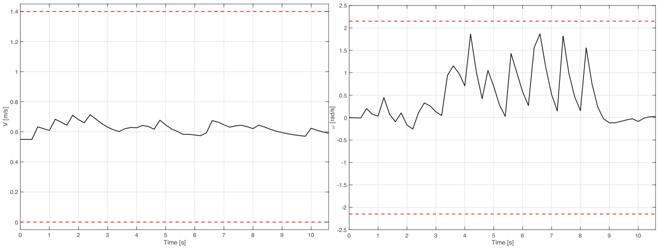
- The whole trajectory includes 10 segments with an overall length of about 3.93m
- Red-dashed lines represent control velocities constraints
- Black solid lines denote the forward and angular control velocities obtained at varying robot initial pose.

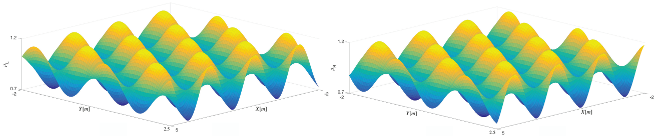
Simulation Scenario #2

- Three additional obstacles (black boxes) are considered in the operating environment.
- Red line denotes an optimal feasible trajectory including 14 segments with a full length of about 6.5*m*.
- In such a scenario the optimal trajectory requires a loop to perform required 90*deg* turn.

Simulation Scenario #2

Forward and rotational velocities





μ_r and μ_l

Conclusions

- In this paper the problem of motion planning of a skid- steered tracked mobile robot subject to skid and slip phenomena is tackled;
- A feedback control action is firstly designed to achieve trajectory tracking of mobile robot accounting for constraints on trajectory tracking error and robot control velocities;
- In order to find optimal feasible trajectory in terms of succession of segments to cross with an assigned nominal velocity, a procedure based on the A* shortest path algorithm is proposed;
- Trajectory feasibility is guaranteed by recurring to set based arguments involving the solution of SDP minimization problems;
- Finally, some numerical simulations are discussed to show effectiveness of the proposed approach.

Feasible trajectory planning algorithm for a skid-steered tracked mobile robot subject to skid and slip phenomena

V. A. Nardi   A. Ferraro   V. Scordamaglia

DIIES - Università degli Studi di Reggio Calabria - Reggio Calabria(RC) -ITALY

Università degli Studi
Mediterranea
di Reggio Calabria

MMAR 2018

AUGUST 27 - 30, 2018
AMBER BALTIC HOTEL
MIEDZYZDROJE
POLAND

23RD INTERNATIONAL CONFERENCE
ON METHODS AND MODELS
IN AUTOMATION AND ROBOTICS

$A^* - like$

- $A^* - like$ algorithm is adopted to compute a shortest path (if any) in terms of a succession of arcs connecting starting $V_0 \in V$ and ending $V_F \in V$ nodes.
- A^* algorithm is a classical approach in searching the best path in a graph. It relies upon exploration of the most promising arc according to an heuristic function estimating the lower-bound of the cost of path including arcs to be explored.
- A^* guarantees the exploration of fewer nodes than any other algorithm using the same heuristic if the heuristic function never overestimates the cost of path.
- The considered heuristic function is sum of two terms: a) the cost of computed path; b) the euclidean distance between the last explored node and destination node.
- If a succession of arcs W_{OF} connecting V_O and V_F is explored, cost of path γ_{OF} becomes an upper bound of the heuristic function. Thus, a path stops being explored when its heuristic exceeds the current upper bound.
- Upper bound must be updated if a shorter path reaching the destination node is found.