2017 IEEE Intelligent Vehicles Symposium (IV)
June 11-14, 2017, Redondo Beach, CA, USA

Stable Timed Elastic Bands with Loose Ends

Fritz Ulbrich[1], Daniel Goehring[1], Tobias Langner[1], Zahra Boroujeni[1], and Raúl Rojas[1]

Abstract— **In this paper we propose a trajectory planning approach for autonomous vehicles in highly dynamic traffic scenarios, capable of exploiting the observed trajectories of other vehicles. For this purpose, we introduce a novel variant of the timed elastic bands (TEB) approach by using fixed time intervals and a flexible goal position. We tested our method with a simulated merge-into-traffic scenario and compare it to a reference TEB implementation with focus on the impact of important parameters and the stability of planned trajectories. The results show that our method is an improvement over TEB in terms trajectory smoothness and stability.**

I. Introduction

Most planning approaches for autonomous driving rely heavily on accurate maps. However, there are many situations where the observed world mismatches the mapped world. Other vehicle drivers may decide to not follow the mapped lanes, because the pathway of roads has changed, lane markings are missing, or because lanes are temporarily blocked. In such scenarios, it is certainly safer to adopt the ego trajectory to that of other vehicles nearby.

A simple and effective strategy is to follow other vehicles, which are driving in the desired general direction. These vehicles may be driving on an adjacent lane and our planning goal is to merge into the lane defined by the other vehicles' trajectories. Such merge-in maneuvers are very important and complex in high density traffic, because the kinematics and dynamics of the own and other vehicles have to be taken into account. In the following, we mainly focus on this scenario to keep the evaluation tractable.

In this paper, we utilize timed elastic bands since they provide beneficial properties for our planning task: They consider temporal aspects of the trajectory, generate plans with respect to steering and velocity, and consider dynamic constraints in addition to obstacles. All constraints must be formulated as target function creating a huge set of parameters, which need to be found and tuned. The actual energy minimization adjusts all poses and time intervals with respect to these functions. However, such a path is unnecessarily restrictive. Usually, the exact point of lane transition is irrelevant. We only require the goal pose to be somewhere on the trajectory to be followed.

Therefore, we propose a modification to the timed elastic band approach addressing this aspect while stabilizing the generated paths. We evaluate some of the most influential parameters based on the simulation of a merge into traffic scenario, as one of the most prominent maneuvers requiring an adoption to other vehicles trajectories.

The sections of this paper are structured as follows: Section II gives a brief overview of most common planning approaches, Section III provides details about our planning and control architecture. In Section IV we describe the necessary parameters of the used timed elastic bands approach. Important experimental results are presented in V, followed by a conclusion in VI.

II. Related Work

Most planning algorithms for self-driving vehicles are either applied in unstructured environments, i.e., without a map, or in structured areas with roads where maps are usually available. Very prominent examples for path planning algorithms come from the A* search family, as, e.g., ARA* from Likhachev et. al for road vehicles, c.f. [10] or RRTs from LaValle et al., c.f. [8].

Path planning approaches for autonomous vehicles often take advantage of prerecorded maps, as for the CMU Boss vehicle at the DARPA Grand Urban Challenge [17], as in the work of Czerwionka et. al. [3], or as Wang et al., following predefined routes [18]. In this case, the velocity on a given trajectory was calculated in a second pass, with respect to dynamic obstacles, as described in [4]. Heinrich et al. proposed an "approach for automated vehicle motion planning systems that introduces the likelihood of an information gain at future positions to trajectory optimization" [6]. Optimal trajectories for time-critical street scenarios using discretized terminal manifolds are presented in [19]. Planning methods for vehicles which take advantage of lattices are described, e.g., in [20] .

In contrast, path planning without a map is often applied to unstructured or outdoor environment, as well as to swarm-like maneuver planning, as in [16]. The work of Keller et al. [7] describes the generation of "optimal trajectories for vehicle collision avoidance with a Timed Elastic Band (TEB) framework". The idea of elastic bands reaches back to Quinlan et al. [12], who used them to refine coarse trajectories in mobile robot navigation.

The timed elastic band approach is an extension to elastic bands and has been designed originally for optimal trajectory planning for non-holonomic mobile robots by Rösman et al. [13]. In [14] an extension is proposed to avoid local minima by planning multiple topologically distinctive trajectories simultaneously and [15] combines timed elastic bands with MPC. Timed elastic bands have been sucessfully applied to planning dynamic maneuvers for CMUs BOSS self-driving vehicle in [9]. More details about the AutoNOMOS project,

[1]DCMLR, Computer Science Institute, Freie Universität Berlin, Germany, {fritz.ulbrich | daniel.goehring | tobias.langner | zahra.boroujeni | raul.rojas}@fu-berlin.de

for which the presented approach was designed can be found in [1].

III. Path Planning and Control Architecture

The path planning approach described below was implemented in the ROS framework [11] and tested in a simulation environment. An overview of the involved modules is shown in Fig. 1.

The simulator modules - or respectively the I/O modules for sensors and actuators in a live environment - provide the current pose x (position and orientation) as well as the current velocity v (linear and angular) to all other modules. Information about obstacles (size, pose, velocity and acceleration) is provided to the path planning module.

Based on this data, the path planning module calculates a plan consisting of a list of poses over time. A path following module calculates a lookahead point p and extracts a desired velocity v_d from the current plan based on the current configuration x and velocity v. Details on how the lookahead point is calculated can be found in [4]. The controller then calculates the control output u, i.e., actual steer momentum and values for the throttle/brake actuators.

Due to performance reasons, the path planning and control stack was implemented as a hybrid of open and closed loop control concepts. While there is a direct feedback to the path planning module in each iteration, all other modules can be run at different (possibly higher) frequencies. In such a case, they are working on the same unchanged plan for some iterations.

A key advantage of this architecture is that the path planning module can be run at a lower frequency while the control output can still be generated at higher frequencies - which is necessary to follow the planned trajectory precisely and avoid oscillations. For the approach described in this paper, path planning ran at a frequency of 10 Hz and all other modules at 100 Hz. On the downside, this architecture tends to slightly cut curves when following the plan due to the calculation of a lookahead point [4].

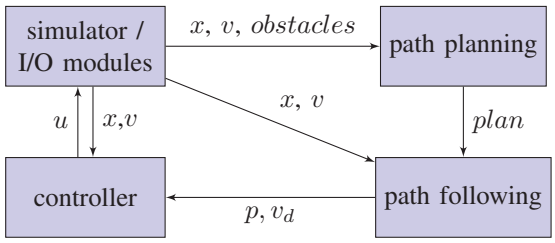


Fig. 1. Module overview and data flow

IV. Stable Timed Elastic Bands with Loose Ends

The path planning approach with stable timed elastic bands with loose ends (STEBLE) proposed in this paper largely builds on the timed elastic band (TEB) framework introduced by Rösman et. al. [13]. The TEB approach was implemented and published by Rösman as central part of the ROS package teb_local_planner [2]. It extends the elastic band approach proposed by Quinlan et. al. [12] with regard to temporal aspects.

The original elastic band is described solely as a sequence $Q = \{X_i\}_{i=0}^{n}$ of robot configurations in a three-dimensional workspace with $\mathbf{x}_i = (x_i, y_i, \beta_i)$ (two coordinates for the position on a plane plus one angle for the orientation). TEB augments Q by an additional sequence of time intervals $\tau = \{\Delta T_i\}_{i=0}^{n-1}$, where each time interval ΔT_i denotes the time needed by the robot to transit from configuration $\mathbf{x}_i$ to configuration $\mathbf{x}_{i+1}$.

The resulting timed elastic band $B = (Q, \tau)$ is then optimized in terms of spatial configurations and temporal intervals with regard to an weighted multi-objective function $f(B)$, which is the weighted sum of components f_x described in the following. The TEB approach allows to formulate objective functions regarding dynamic aspects, such as velocity, acceleration or moving obstacles. The resulting optimization problem is represented as a hyper-graph (explained in detail in [13]) and solved using the g2o framework [5].

During the optimization process, the first and last configuration are always fixed in the TEB approach. This property is disbanded for the goal configuration in the STEBLE approach.

A. Objective Functions

Most of the components f_k of the weighted multi-objective function

$$f(B) = \sum_k \gamma_k f_k(B) \tag{1}$$

used in this paper were proposed by Rösman et. al. and are described in detail in [13], i.e., restrictions for maximal translational velocity, maximal translational and rotational acceleration, non-holonomic kinematics and fastest path.

Also, existing functions from the implementation in [2] were used to penalize backward-driving, falling below the minimum turning radius and maintaining a specific velocity at the start and goal configuration.

Limiting Centripetal Acceleration: Additionally, we added an objective function to restrict the centripetal acceleration a_c, which implicitly restricts the maximal rotational velocity ω depending on the velocity v and the radius r of the arc segment between two consecutive configurations. The centripetal acceleration a_{ci} can be computed directly from the translational velocity v_i and the rotational velocity ω_i. The radius r_i does not have to be calculated explicitly, since it is equal to v_i/ω_i.

$$v_i = \frac{1}{\Delta T_i} \left\| \begin{pmatrix} x_{i+1} - x_i \\ y_{i+1} - y_i \end{pmatrix} \right\| \tag{2}$$

$$\omega_i = \frac{\beta_{i+1} - \beta_i}{\Delta T_i} \tag{3}$$

$$a_{ci} = \frac{v_i^2}{r_i} = v_i \cdot \omega_i \tag{4}$$

Limiting the centripetal acceleration has the advantage of allowing higher rotational velocities at lower translational

velocities. This enables sharp turns at low speed while still maintaining a comfortable trajectory at higher speed.

Dynamic Obstacles: The reference TEB implementation handles dynamic obstacles by penalizing the minimal separation of a configuration $\mathbf{x}_i$ and the predicted position of the obstacle (compare f_{ob} from [13]). In our approach, we also use the minimal separation of a configuration $\mathbf{x}_i$ and a dynamic obstacle, but we represent both the obstacle and the ego vehicle as a pair of a line segment l and a radius r (compare Fig. 2). This is more appropriate for road vehicles, which typically have different width and length, but still reduces the computational cost compared to, e.g., matching polygonal shapes. This representation also allows different lateral and longitudinal safety margins by scaling either the length l or the radius r.

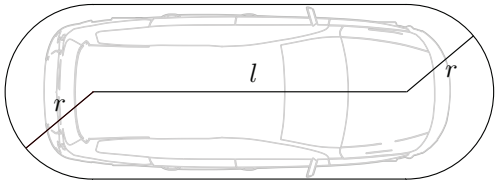


Fig. 2. The shape representing the ego vehicle and dynamic obstacles is defined by a line segment l and a radius r.

The trajectory of dynamic obstacle j is predicted by multiplying the components of its velocity vector $v_j = (v_x, v_y)$ with an time interval ΔT_n, which is determined for each configuration $\mathbf{x}_i$ of the band by summing up all time intervals up to index i.

Following Trajectories: A central aspect of the STEBLE approach is a variation of the objective function f_{path} (compare [13]) for the attainment of intermediate waypoints. The original function is based on penalizing the minimal separation $d_{min,j}$ between the TEB and an waypoint z_j, if it is above an threshold $r_{p_{max}}$. To reflect the intention of driving on the trajectory of another vehicle, we penalize the separation (i.e., the euclidean distance) of each configuration $\mathbf{x}_i$ to a sequence of line segments representing the other vehicles' trajectories.

B. Flexible Goal Position and Fixed Time Intervals

Although the experiments with the modified objective function described above provided promising results, the generated trajectories tended to create unnecessary maneuvers in order to reach the goal location. However, the restriction of having a fixed goal position is unnecessary, because many positions on the other vehicle's trajectory are equally suitable to fulfill the following task. The general idea is that the objective function f_{path}, which attracts the band to the other vehicles trajectories, provides enough stability to the planned paths, so that we can loosen the end of the elastic band, i.e., the goal configuration can be changed during the optimization process.

Furthermore, we can subsequently reduce the complexity of the optimization problem drastically by using a static value for ΔT_i between all configurations. Since we have a flexible goal position, we do not need the flexibility in time. As the experiments below have shown, this is also the most important factor in stabilizing the band.

Although this can be seen as going a step back to the original elastic band approach, we are keeping all the objective functions with respect to ΔT, so that the expression "timed" is still justified. Dynamic changes of the velocity throughout the band are still possible by varying the euclidean distance between configurations. Also, it is a preferable feature to have constant time intervals on the planned path, as some calculations can be simplified (e.g. the prediction of obstacles, compare IV-A).

C. Initializing and Adjusting the Band

The original TEB approach (compare [13]) relies on an "outer loop" to adjust the number of configurations and time intervals of the band to the planning horizon by adding or removing them based on a hysteresis on ΔT. Therefore, the band can be initialized with just two configurations (start and goal) and one time interval between those - although initializing the band with an "initial path" is possible and preferable for most scenarios.

For the STEBLE approach we cannot use a heuristic for adjusting the band based on ΔT, since all time intervals ΔT_i are fixed. Adjusting the number of configurations based on distance over space would certainly be a valid approach, but on the other hand, with a flexible goal position we do not need to adjust the number of configurations at all. Subsequently, the initialization of the band becomes more important, since it determines the final length of the band (with regard to time).

For the experiments described below the band was initialized with a rather simple heuristic. The number of initial configurations n is determined by ΔT_{init} (the initial - and in case of fixed time intervals final - value for all time intervals), the average of the start and goal velocity v_{avg} and the euclidean distance between the start and goal d_{sg}.

$$n = \frac{d_{sg}}{v_{avg} \cdot \Delta T_{init}} \tag{5}$$

The initial positions are then sampled uniformly along the direct path between start and goal, the initial orientation is pointing towards the goal for all configurations.

V. Experimental Results

A. Scenario

Due to the huge amount of parameters influencing the performance of the proposed approach, a rather simple scenario was chosen. Nonetheless, it is a scenario occurring often in the real world at highway entries. It consists of three dynamic obstacles driving with a constant velocity on a straight line with a distance of 30 m between them. The obstacles as well as the ego vehicle are modeled with a width of 2 m and a length of 5 m. The ego vehicle is starting with the same orientation in 4 m distance to the trajectory of the obstacles (compare Fig. 3).

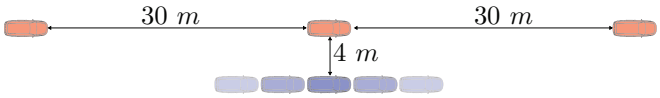


Fig. 3. Starting configuration of the ego vehicle (blue) and three dynamic obstacles (red). For some of the experiments the longitudinal offset of the ego vehicle to the obstacles is varied.

The longitudinal offset of the ego vehicle to the closest obstacle varies for some experiments. If not stated otherwise it is set to 0 m.

The initial velocity of the obstacles was set to 25 m/s (90 km/h). Nonetheless, due to the architecture of the simulation, which does not guarantee a specific timing in the sequence of initialization, and the control parameters of the simulated obstacles, there are minimal variations in the actual velocity ($\sigma_{v\_ob} \approx 0.02m/s$). The mean of the velocity of the obstacles is slightly below the initial value ($\mu_{v\_ob} \approx 24.84m/s$). The ego vehicle is starting with the same initial velocity as the obstacles, if not stated otherwise.

The (initial) goal for the path planning is set to a position 200 m ahead of the closest obstacle, i.e., in 4 m lateral distance and 200 m longitudinal distance from the ego vehicle. The desired orientation and velocity at the goal position are chosen so that the ego vehicle tries to match the respective values of the obstacles.

B. Parameters, Thresholds and Weights

In this section, we describe the parameters used for the experiments described below. Although other parameter sets worked equally well, the following values were set as default for the experiments throughout this paper.

The number of iterations for optimizing the underlying graph (compare [5]) was set to 100 for the STEBLE approach. For the experiments using the original TEB approach of initializing the band with only two configurations and adjusting the number of configurations in an outer loop, the number of iterations of this outer loop as well as the iterations for the graph optimization was set to 10, which also results in 100 overall iterations optimizing the graph.

The initial - and in case of STEBLE final - value for all ΔT_i was set to $\Delta T_{init} = 0.1$ s. The threshold for adding or removing configurations was set to 0.01 s for the original TEB approach (compare IV-C).

An overview of the thresholds used for the objective functions is given in table I. The ϵ value (compare [13]) was set to 0 where applicable. Some values are directly determined by the vehicle dimension and dynamics, others depend on user preferences regarding safety distance and driving comfort.

For a minimal separation from obstacles, several values were tested. Fig. 4 depicts the results for 2 m, 3 m and 4 m (with the objective function also taking into account the dimensions of the obstacles and the ego vehicle - compare IV-A). It can be clearly seen that the resulting trajectories are laterally shifted - nonetheless the general behavior is quite similar. A threshold of 2 m pushes the trajectory closer to the obstacles, which then enables an (on average) earlier transition to the obstacles' lane. The other values produce similar trajectories, but lane changes tend to happen later. Additionally, the impact of any threshold is tightly connected to the weight of the corresponding objective function. As a value of 2 m still preserves a collision free path for all tested scenarios - and the default weights described below - the value was chosen as default for the experiments.

TABLE I
THRESHOLDS OF THE OBJECTIVE FUNCTIONS

f_x	Threshold	Description
f_v	30 m/s	maximum velocity
f_{acc_x}	2 m/s^2	maximum longitudinal acceleration
f_{acc_y}	2 m/s^2	maximum centripetal acceleration
f_{acc_theta}	1 $1/s^2$	maximum rotational acceleration
f_{path}	0 m	attraction to obstacle trajectories
f_{ob}	2 m	separation from obstacles
f_{turn}	4 m	minimum turning radius

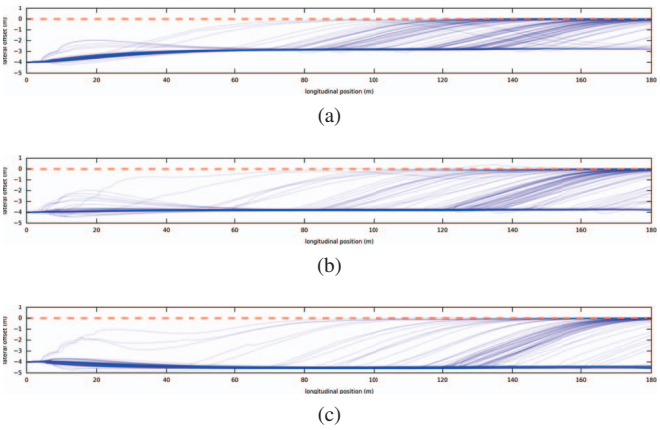


Fig. 4. Each subplot shows trajectories of the simulated ego vehicle from 100 repetitions (blue lines) of our scenario and the obstacles' trajectory (red dashed line). For each simulation we used the default parameter set described in Section V-B, except for the minimal distance to obstacles threshold, which was set to (a) $2m$, (b) $3m$ and (c) $4m$. Variations of the trajectories within a subplot result from minor differences in the velocity of the obstacles (compare Section V-A).

Similarly, the value for maximum centripetal acceleration does obviously change the trajectories, but does not change the general behavior, so plots are omitted. It also depends on the preference how smooth the trajectories should be. The chosen default value of 2 m/s^2 proved to be comfortable when using it in a real vehicle in previous works. Correspondingly, the values for maximum lateral and longitudinal as well as rotational acceleration were chosen to provide comfortable trajectories rather than exploiting the vehicle limits.

The weights for the objective functions were mostly determined heuristically. Their influence on the actual trajectory is tightly connected to the weights and thresholds of other objective functions. If several objective functions compete in a certain configuration, the relative weight difference defines the priority of the objectives. Given the large space of configurations, it is difficult to determine optimal values,

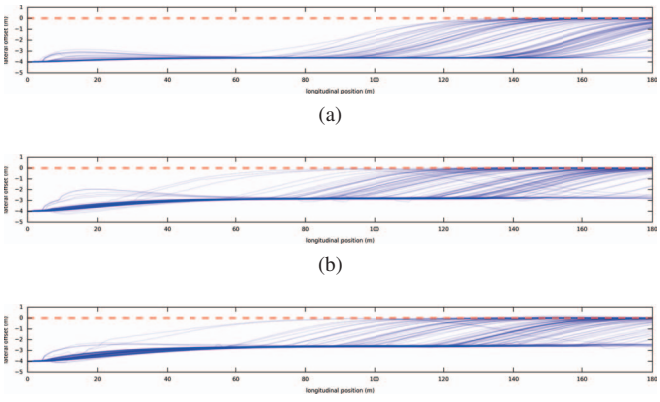
especially for different scenarios. The default values chosen for the experiments are listed in table II.

TABLE II
WEIGHTS OF THE OBJECTIVE FUNCTIONS

Weight	Value	Description
γ_v	30	velocity (maximum, start, goal)
γ_{acc_x}	30	maximum longitudinal acceleration
γ_{acc_y}	30	maximum centripetal acceleration
γ_{acc_theta}	30	maximum rotational acceleration
γ_{time}	100	penalize sum of ΔT_i (prefer faster paths)
γ_{path}	300	attraction to obstacle trajectories
γ_{ob}	1000	separation from obstacles
γ_{nhk}	10000	non-holonomic kinetics
$\gamma_{forward}$	10000	penalize backward driving
γ_{turn}	10000	minimum turning radius

While objective functions relating to acceleration and velocity are clearly not as important as avoiding obstacles, maintaining the kinematic constraints was chosen as top priority - since we do not want to plan trajectories which are not drivable. The specific ratio between the different weight classes was chosen heuristically - other ratios provided similar results.

As can be seen in Fig. 5, the result of changing the weight γ_{path} for the objective function for following the obstacles trajectory is quite similar to changing the threshold for minimal obstacle separation. Although the trajectories are shifted towards the obstacles, the general outcome does not change significantly. The default value for the experiments was set to $\gamma_{path} = 300$, but different values worked also well.

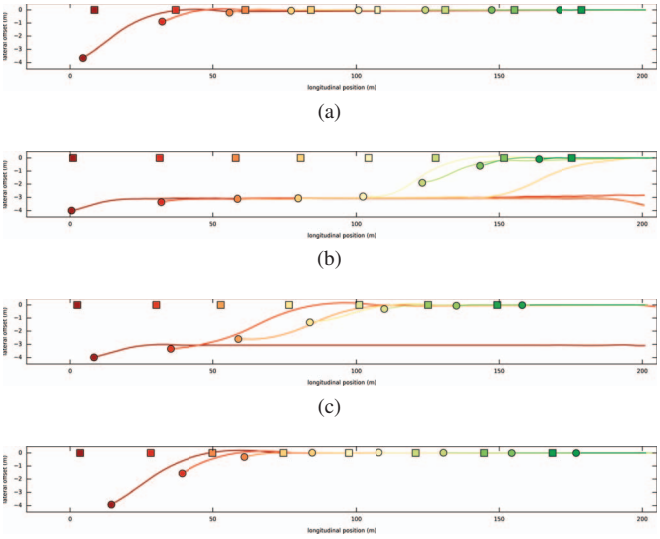


(a) (b) (c)

Fig. 5. Each subplot shows trajectories of the simulated ego vehicle from 100 repetitions (blue lines) of our scenario and the obstacles' trajectory (red dashed line). For each simulation we used the default parameter set described in Section V-B, except for the the weight γ_{path}, which was set to (a) $\gamma_{path} = 100$, (b) $\gamma_{path} = 300$ and (c) $\gamma_{path} = 500$. Variations of the trajectories within a subplot result from minor differences in the velocity of the obstacles (compare Section V-A).

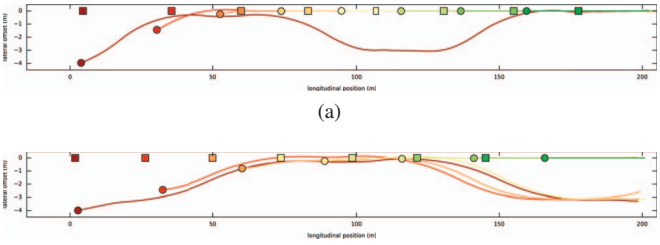
C. Different Start Configurations

The STEBLE approach generated smooth trajectories for several variations of the scenario described in Section V-A. In this section we provide plots of the trajectories and velocity profiles for some exemplary starting configurations. Fig. 6 depicts the results for different starting positions. If the ego vehicle starts in a configuration, where it can change on the obstacles trajectory directly with enough separation from the obstacles, it does so as expected. If not, the ego vehicle decelerates or accelerates depending on the longitudinal offset and falls behind or goes ahead the closest obstacle respectively.



(a) (b) (c) (d)

Fig. 6. Exemplary trajectories and the planned paths with the ego vehicle starting at different longitudinal offsets (a) $-5\ m$, (b) $0\ m$, (c) $5\ m$ and (d) $10\ m$ to the closest obstacle. Circles represent the position of the ego vehicle. Squares represent the position of the closest obstacle. Lines represent the planned paths. Colors indicate the elapsed time (trajectories planned at the beginning are red, those planned later are yellow and green). Newer paths may occlude older ones.

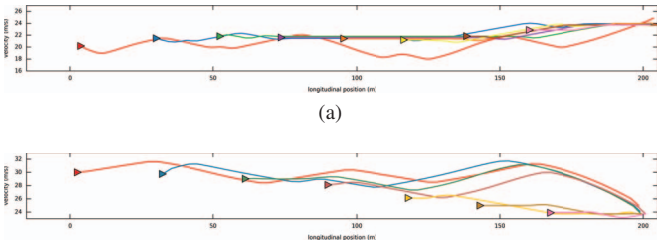


(a) (b)

Fig. 7. Exemplary trajectories and the planned paths with the ego vehicle starting at longitudinal offset $0\ m$ to the closest obstacle with velocity (a) $v = 20\ m/s$ ($5\ m/s$ slower than obstacles) and (b) $v = 30\ m/s$ ($5\ m/s$ faster than obstacles).

Varying the ego vehicles velocity in the start configuration relative to the obstacles velocity yields similar results. The ego vehicle is adapting to the velocity of the obstacles, while using the difference in momentum in the beginning to get into the gap between two obstacles. The corresponding planned paths are depicted in Fig. 7, velocity profiles in Fig. 8.

If there is a large difference between the obstacles' and the ego vehicle's velocities, the planned paths transitions forth and back between the two lanes, in order to avoid a collision



(a)

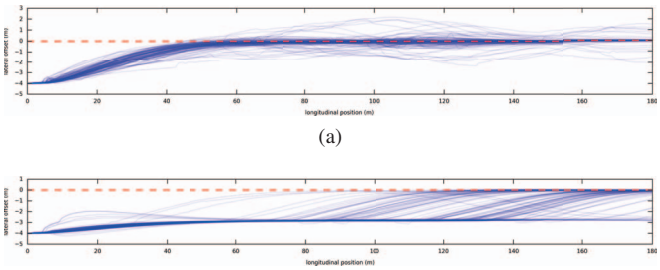
(b)

Fig. 8. Velocity profiles of the planned paths with starting velocity (a) $v = 20\ m/s$ ($5\ m/s$ slower than obstacles) and (c) $v = 30\ m/s$ ($5\ m/s$ faster than obstacles). Triangles represent the velocity of the ego vehicle at a specific longitudinal position. Lines represent the corresponding planned velocities.

with obstacles further behind or ahead. Depending on the weights of the objective functions, it is favorable to switch on the obstacles trajectory even for short distances. If frequent lane changes are undesired behavior, it can be avoided by tuning the thresholds and weights for maximum accelerations or attraction to the obstacles trajectory.

D. TEB vs. STEBLE

In this section we compare the proposed approach with fixed time intervals and a flexible goal position to the original TEB approach (using flexible time intervals and a fixed goal position) over several iterations of the same scenario and parameter set. However, as described in Section V-A, there are minor variations in the velocity (and subsequently position) of the obstacles, which cause different trajectories. As can be seen in Fig. 9, there is no variance in the trajectories when obstacles are ignored.



(a)

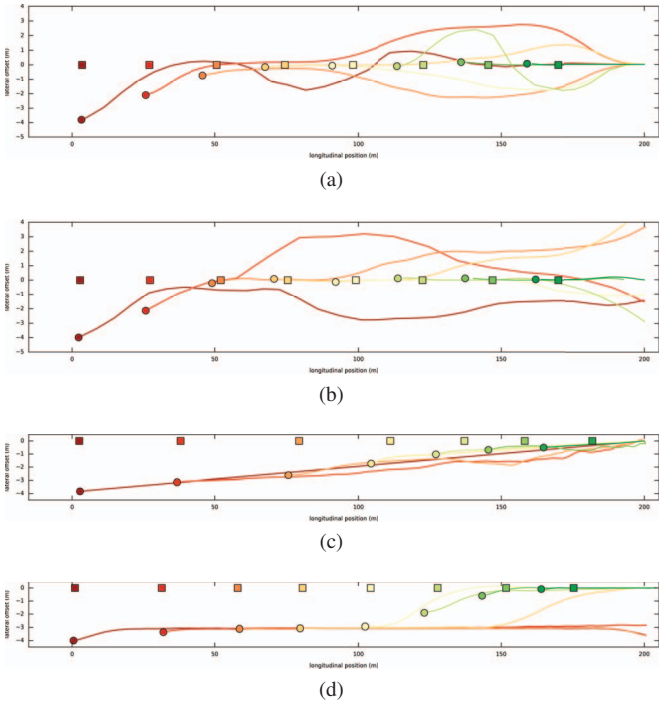
(b)

(c)

Fig. 9. The actual trajectories driven by the simulated ego vehicle in 100 repetitions of the scenario with the default parameter set (compare Section V-B) and minor variations in the velocity of the obstacles (compare Section V-A). (a) The original TEB approach (fixed goal position, flexible time intervals). (b) The STEBLE approach (flexible goal position, fixed time intervals). (c) The trajectory without considering obstacles (identical for both approaches)

Both approaches generate sensible trajectories for the examined scenario. The most obvious difference is that, while the original approach is performing the change to the obstacles trajectory as soon as possible, the STEBLE approach performs the actual lane change at intervals widely scattered over the latter half of the distance. In some cases ($\sim$ 5%) it does not change at all. While this may seem non-optimal, the generated trajectories are generally safer - in terms of separation from obstacles - as the ego vehicle is not forced on the obstacles trajectory. With a fixed goal, there are several configurations, where the planner is forced to violate some of the constraints to reach the goal.

Furthermore, the trajectories generated by the proposed approach can be seen as much more stable, since the actual change to the other lane resembles a very similar sigmoidal curve for all iterations - despite the longitudinal shifts. In contrast, the original TEB approach sticks closer to the trajectory generated without considering obstacles. However, the trajectories are scattered along this path, because the ego vehicle gets ”pushed out” of the seemingly optimal path much more. This is probably due to the optimization falling in many (nearly) equally costing local minima of the objective function.



(a)

(b)

(c)

(d)

Fig. 10. Exemplary trajectories and the planned paths for different graduations between the original TEB and the proposed STEBLE approach. (a) Fixed goal position, flexible time intervals, initialization with two configurations. (b) Flexible goal position, flexible time intervals, initialization with two configurations. (c) Flexible goal position, flexible time intervals, initialization with number of configurations based on heuristic. (d) The STEBLE approach (flexible goal position, fixed time intervals, initialization with number of configurations based on heuristic).

Fig. 10 depicts exemplary planned paths for different graduations between the original TEB and the proposed approach. It can be clearly seen that the most stabilizing factor is the fixing of the time intervals. While all variations with flexible time intervals plan largely varying paths over the course of one iteration, the STEBLE variant plans straight in the beginning and at some point switches to sigmoidal curves closely resembling the actual trajectories (compare

Fig. 9).

Nonetheless, it can also be noticed that the initialization of the band with the simple heuristic described in Section IV-C also impacts the stability of the planned paths (and subsequently the resulting trajectories) with flexible time intervals. The optimization seems to fall into a relatively small set of local minima close to the initial path in this case.

Another interesting observation is that the STEBLE approach does not suffer from the invalidation of the objective function f_{time}, which always returns the same value for fixed time intervals. With flexible time intervals, for low values of γ_{time} the ego vehicle tends to slow down rapidly, switch on the obstacles trajectory with a sharp curve and then accelerate to the goal velocity. This behavior is prevented when using fixed time intervals (i.e., a fixed length of the plan in terms of time), because slowing down (when not on the obstacles trajectory) leads to proportionally less configurations on the band which are close to the obstacles trajectory - which then is penalized by f_{path}.

VI. Conclusions

It can be concluded, that the proposed STEBLE approach works well for the simulated scenario tested in this paper, but obviously has to be further tested in more complex scenarios - mainly involving different trajectories of the obstacles, e.g. with curvature and dynamic acceleration - and with real vehicles. It has been shown, that STEBLE is much less prone to changes in the parameters and weights (in terms of always producing similar, more "human-like" paths) than the original TEB approach.

A minor drawback is the property that for some configurations the ego vehicle tends to change relatively late on the obstacles trajectory - and sometimes not at all - due to the optimization falling into local minima. This is a general problem of the elastic band approach and can certainly be mitigated by generating several plans simultaneously in different homotopy classes (as proposed by Rösman et al. [14]) at the cost increased processing time.

There is also some potential in tuning the weights and parameters, but probably the most influential improvement would be additional objective functions. A natural candidate is a function penalizing driving with no longitudinal offset to other vehicles - a heuristic experienced human drivers often follow instinctively.

Furthermore, future research can be done on the topic, where to set the initial goal for the presented approach. Although the goal position is not fixed, it directly impacts the initialization of the band and subsequently the overall length of the band with regard to time - which then leads to potentially different planned paths. While for the experiments above a static position was chosen to reduce the complexity of the analyzed scenario, an obvious extension would be to change the initial goal position dynamically with relation to the observed obstacles and a specified planning horizon with regard to time.

Acknowledgment

This work was supported by the Deutsche Forschungsgemeinschaft (DFG), SPP 1835 - Kooperativ interagierende Automobile.

References

[1] Autonomos project. http://www.autonomos-labs.de/. Accessed: 2017-01-10.

[2] Teb local planner package for ros. http://http://wiki.ros.org/teb_local_planner. Accessed: 2017-01-10.

[3] P. Czerwionka. A three dimensional map format for autonomous vehicles. Master's thesis, Freie Universität Berlin, 2014.

[4] D. Göhring. Controller architecture for the autonomous cars: Madeingermany and e-instein. Technical report, Freie Universität Berlin, 2012.

[5] G. Grisetti, H. Strasdat, K. Konolige, and W. Burgard. g2o: A general framework for graph optimization. In *IEEE International Conference on Robotics and Automation*, 2011.

[6] S. Heinrich, J. Stubbemann, and R. Rojas. Optimizing a driving strategy by its sensor coverage of relevant environment information. In *Intelligent Vehicles Symposium (IV), 2016 IEEE*, pages 441–446. IEEE, 2016.

[7] M. Keller, F. Hoffmann, C. Hass, T. Bertram, and A. Seewald. Planning of optimal collision avoidance trajectories with timed elastic bands. *IFAC Proceedings Volumes*, 47(3):9822–9827, 2014.

[8] S. LaValle and J. K. Jr. Randomized kinodynamic planning. In *International Conference on Robotics and Automation*, pages 473–479, 1999.

[9] M. Likhachev and D. Ferguson. Planning long dynamically-feasible maneuvers for autonomous vehicles. *The International Journal of Robotics Research*, 28(8), July 2009.

[10] M. Likhachev, G. Gordon, and S. Thrun. Ara*: Anytime a* with provable bounds on sub-optimality. *Advances in Neural Information Processing Systems (NIPS)*, 16, 2016.

[11] O. S. R. F. (OSRF). Robot operating system (ros). http://www.ros.org/. Accessed: 2017-01-10.

[12] S. Quinlan and O. Khatib. Elastic bands: Connecting path planning and control. In *Robotics and Automation, 1993. Proceedings., 1993 IEEE International Conference on*, pages 802–807. IEEE, 1993.

[13] C. Rösmann, W. Feiten, T. Wösch, F. Hoffmann, and T. Bertram. Trajectory modification considering dynamic constraints of autonomous robots. In *Robotics; Proceedings of ROBOTIK 2012; 7th German Conference on*, pages 1–6. VDE, 2012.

[14] C. Rösmann, F. Hoffmann, and T. Bertram. Planning of multiple robot trajectories in distinctive topologies. In *Mobile Robots (ECMR), 2015 European Conference on*, pages 1–6. IEEE, 2015.

[15] C. Rösmann, F. Hoffmann, and T. Bertram. Timed-elastic-bands for time-optimal point-to-point nonlinear model predictive control. In *Control Conference (ECC), 2015 European*, pages 3352–3357. IEEE, 2015.

[16] S. Rotter. Swarm behaviour for path planning. Master's thesis, Freie Universität Berlin, 2014.

[17] C. Urmson, J. Anhalt, D. Bagnell, C. Baker, R. Bittner, M. Clark, J. Dolan, and D. D. et al. Autonomous driving in urban environments: Boss and the urban challenge. *Journal of Field Robotics Special Issue on the 2007 DARPA Urban Challenge*, 25(1):425–466, 2008.

[18] M. Wang. *A Cognitive Navigation Approach for Autonomous Vehicles*. mbv, 2012.

[19] M. Werling, S. Kammel, J. Ziegler, and L. Grll. Optimal trajectories for time-critical street scenarios using discretized terminal manifolds. *The International Journal of Robotics Research Robotics Research*, 2011.

[20] Ziegler and C. Stiller. Spatiotemporal state lattices for fast trajectory planning in dynamic on-road driving scenarios. In *IEEE/RSJ International Conference on Intelligent Robots and Systems*, pages 1879–1884, 2009.