Momentum Based Traversal of Mobility Challenges for Autonomous Ground Vehicles

Camilo Ordonez, Nikhil Gupta, Oscar Chuy, Jr., and Emmanuel G. Collins, Jr.

Abstract— Autonomous ground vehicles operating in the field are likely to be faced with several mobility challenges such as piles of rubble, water crossings, steep hills, mud, and stiff vegetation patches. These scenarios are particularly critical for smaller robots with torque and power limited actuators, which as experimentally shown in this work can easily fail to accomplish their tasks in these environments. This paper motivates and provides a methodology that integrates the robot, actuator and terrain models with an efficient motion planner to exploit the vehicle momentum as a way to successfully traverse these difficult terrains. In particular, experimental results showing the efficacy of the proposed methodology are presented for a vegetation patch and a steep hill. Finally, a discussion of the necessary perception work to fully automate the process is included.

I. INTRODUCTION

Autonomous ground vehicles (AGVs) are increasingly being considered and used for challenging outdoor applications. These tasks range from agriculture, mining and military applications to urban, polar and space exploration. Some of these tasks can be addressed by developing very mechanically robust and usually heavy and energy inefficient robots. On the other hand, a more difficult yet more beneficial approach, consists of developing smaller, more energy efficient and smarter robots. However, in order to achieve similar operation standards as the bulkier robotic counterparts, smaller robots need to be able to exploit their dynamics while executing tasks that can easily lead to saturation of their power and torque limited actuators.

In many of the unstructured environments mentioned above, it is likely for robots to have to deal with mobility challenges that force the vehicle to decelerate such as high and stiff vegetation, steep hills, rubble, mud patches and water crossings. Two illustrative examples of task completion failure on two of these scenarios are presented in Figs. 1 through 3, where the FAMU-FSU Bot is facing a patch of stiff (artificial plastic) vegetation and a steep hill. In both experiments the robot was located close to the beginning of the mobility challenge and was commanded to follow a velocity of 1.0m/s for the vegetation and 1.5m/s for the hill. (The commanded velocities could have been the same values; notice that in both cases the maximum velocity achieved by the robot was below 0.6m/s). However, as shown in the snapshots (a)-(h) of Fig. 1, and in the position and velocity profiles of Figs. 2 and 3, the robot became immobilized in the middle of the mobility challenges. The included torque profiles clearly show that the torques are saturated and the robot did not have enough momentum to traverse the mobility challenges. It is important to clarify that the maximum torque was set to 25% for the stiff vegetation and 40% for the steep hill. This is similar to adding a load to the robot and was done to guarantee that the terrains constitute indeed mobility challenges for the FAMU-FSU Bot. (See Section II-A for further discussion.)

The exploitation of robot dynamics has been demonstrated in present years mainly on bio-inspired robots, which while limited in size and actuation, can achieve outstanding locomotion tasks like running on horizontal and vertical surfaces, hopping, and flying [1], [2], [3]. However, the manipulation of dynamics by these type of robots has been typically embedded in their mechanical designs through the usage of compliant limbs and simple open-loop controllers and not at the higher motion planning level. Agile motion has also been achieved in wheeled mobile robots at the low level through manipulation of internal mass and inertial properties during locomotion [4].

At the motion planning level, relevant work has been conducted for underactuated robots. In particular, complex motions that incorporate joint and torque limits have been proposed for multi-link manipulators [5]. Additional related work includes motion planning for autonomous door opening with a mobile manipulator [6], [7]. There the proposed motion planner disallowed states where the door required more force than the robot could exert from a given configuration. Here, we are interested in regimes of operation where the robot actuators cannot provide sufficient torque to accelerate the robot for some periods of the mission (e.g., when faced with the mobility challenges described above).

One approach to deal with actuator limitations consists of the wise usage of momentum. This approach is commonly used by high jump athletes and weight lifters [8]. However, little research has been conducted on the usage of momentum to traverse the type of mobility challenges in the previously mentioned terrains. A related problem was considered in [9], where motion planning was employed to climb a steep hill. However, here we consider a more complex and relevant problem. First, in this work we develop trajectories that optimize a minimum time cost function whereas the cost function in [9] was simply distance. Second, in [9], the robot was allowed to reach the top of the hill at any velocity. Here, the task of reaching the goal at zero (or more generally, a specified) velocity is considered. These constraints make the problem much harder. That is, to achieve fast computations,
this problem requires the employment of a minimum time optimistic heuristic that takes into account the goal velocity. Furthermore, in order to achieve efficient computations, the heuristic must be fast to compute and should not be overly optimistic. Finally, here we also extend the approach to the domain of stiff vegetation patches.

The main contributions of the paper are the formulation of and proposed solution to the problem of mobility challenge traversal in minimum time, with actuator constraints, and a desired goal velocity. The proposed solution is developed by integrating motion planning with simple yet accurate enough physics-based models, which allows for the removal of the many typically ad-hoc heuristics employed by motion planners. In addition, the paper experimentally demonstrates the effectiveness of the proposed solution on a robotic platform traversing a stiff vegetation patch and a steep hill. The paper also proposes a preliminary methodology to obtain the terrain parameters required for this task and points out perception challenges that should be addressed to fully automate the process and increase the system robustness.

The remainder of the paper is structured as follows. Section II describes the experimental platform and presents the proposed vehicle and terrain models. Section III describes the motion planner employed in this research. Section IV provides experimental results. Finally, Section V presents concluding remarks and future directions for this research.

II. EXPERIMENTAL SETUP AND MODELING

This section details the experimental setup and develops the robot dynamic model, which includes a model of the actuators and motor controllers. In addition, models of the vegetation and wooden surfaces employed in the experiments are presented.

A. Experimental Setup and Assumptions

1) Experimental Setup: The FAMU-FSU Bot shown in Fig. 1 is a skid-steered robot, which employs 2 mechanically coupled Pittman GM 9236 brushed DC motors per side. Each pair of motors is controlled using a current control approach by a Maxon motor controller (4-Q-dc). The motor controllers are configured to provide a maximum current of 5A, which corresponds to a maximum torque of about 4.35Nm. However, since the FAMU-FSU Bot is not particularly rugged for vegetation traversal, the robot torque was limited to 25% of its maximum value, which allowed us to perform the experimental validation on more benign terrains for the robot structure. Similarly, the torque for each motor was limited to 40% of the maximum value for steep hill climbing. The hill can be made steep enough that it cannot be traversed using the maximum actuator torques, but due to the simple ramp used, this introduces impact dynamics when entering

Fig. 2: Position, velocity and torque profiles (the torque corresponds to the average of the two robot sides), corresponding to the robot immobilization on the stiff vegetation of Fig. 1. Notice that once the torques saturate, the robot does not have enough momentum to traverse the vegetation. Therefore, the velocity goes to zero and the position remains within the shaded area, which represents the location of the vegetation patch.
the ramp that are not modeled in the current model.

The mobility challenges employed in this research are shown in Fig. 1. They are a wooden platform that contains a stiff vegetation patch and a steep hill both of 1.22m in length. The vegetation patch consists of artificial plants which are stiff enough to provide significant resistance to the robot as it was shown in the motivational experiment of Figs. 1-3.

2) Simplifying Assumptions: At this stage of the research the following assumptions are considered: 1) the robot has a priori knowledge of the length and slope of the mobility challenges, 2) the considered surfaces can provide enough traction to the robot, and therefore wheel odometry is accurate enough to perform robot localization, and 3) motion planning is performed off-board and at the beginning of the mission. Once the trajectory is planned, it is transferred to the robot for execution.

B. Robot, Surface and Motor Modeling

In this subsection, we describe the dynamic model of the robot, which includes the friction models for the wooden and vegetation surfaces. In addition, the actuator model is presented in detail.

1) Robot Modeling: The longitudinal dynamics of the robot are expressed in vehicle coordinates as:

\[ M \ddot{x} + C(x, \dot{x}) + G(x) = F, \]  

(1)

where \( M \) is the robot mass, \( \dot{x} \) and \( \ddot{x} \) respectively denote the robot position, velocity and acceleration, \( C(x, \dot{x}) \) is the frictional term, \( G(x) \) is the gravitational component, and \( F \) is the applied tractive force.

2) Surface Modeling: Surface modeling of mobility challenges is difficult for two main reasons: First, due to the actuator limitations, experiments conducted to estimate the terrain model parameters can lead to robot immobilization and task completion failure. Second, for robots operating in the field, it is desirable to keep to a minimum the number of required experiments to determine the model parameters. These two challenges are addressed here by employing reduced order friction models and by following an experimental methodology to estimate the terrain parameters while dealing with actuator limitations.

The general form of the friction models is given by

\[ C(x, \dot{x}) = b(x) \dot{x} + R_r(x), \]  

(2)

where \( b \) is a damping term and \( R_r \) corresponds to the rolling resistance. The frictional term \( C \) captures the combined effect of the friction in the drive system and the friction due to the terrain. It is important to clarify that this friction model is only used here as a first approximation as it models reasonably well the situation were the robot is always traversing the same part of terrain. However, the friction experienced by the robot is dependent on the density of the vegetation, which changes with the direction of traversal. Ongoing work involves the improvement of this model to account for this uncertainty.

It is also important to mention that the modeling of a benign (wooden) surface and a challenging (vegetation) surface is conducted using two different approaches. In the case of the wooden surface, which the robot can traverse without difficulty, both the damping and rolling resistance were determined by running the robot over the surface at different constant speeds while monitoring the actuator torques. The estimated parameters were \( R_{rw} = 7.86N \) and \( b_w = 3.36N/m \). For the vegetation surface, the situation is more complicated because due to the actuator limitation, the robot is always forced to decelerate while traversing this terrain, which, as shown in the motivation experiment, can lead to robot immobilization if the robot has little momentum when it comes in contact with the vegetation.

In the field, it is expected that robots will learn the properties of these difficult terrains by first traversing at high speeds terrains of similar characteristics. Then, when faced with
new perceptually similar challenging terrains, the robot will employ its past observations as initial guesses of the terrain parameters. To mimic this scenario, the following experiment was conducted. Starting from the wooden surface, the robot was allowed to accelerate to a relatively high speed of about 1.0 m/s, which gave it enough momentum to traverse the vegetation. During this process, actuator torques, and robot velocity were measured. Fig. 4 shows the velocity profile of this maneuver, which clearly shows that during the traversal of the patch, the robot is forced to decelerate. There is also a noticeable high variance in the velocity caused by the non-homogeneity of the terrain. Due to all this uncertainty and the fact that in the field, experiments should be kept to a minimum, it was decided to simplify the model of the vegetation and include only a high rolling resistance term, which was estimated by \( R_v = F - M \ddot{x} \), where \( \ddot{x} \) corresponds to the average deceleration of the robot while traversing the vegetation, and \( F \) is the applied tractive force. The estimated resistance obtained from the experiment of Fig. 4 was 27.59 N (damping was neglected in the vegetation experiments).

In addition to estimating the model parameters, it is important to approximate the transitions between two different terrains. Here, we employ linear functions of position for both parameters \( b \) and \( R_v \). This transition model was selected because it represents the physical situation in which the robot has the front set of wheels on the vegetation and the rear wheels on the wood. The complete surface model is illustrated in Fig. 5 and is expressed by

\[
R_v = \begin{cases} 
R_{rw} & \text{if } x < -l \\
R_{rw} + \frac{R_{w0} - R_{rw}}{2l} (x - l) & \text{if } -l \leq x < l \\
R_v & \text{if } l \leq x < d - l \\
R_v + \frac{R_{v0} - R_v}{2d} (x - d) & \text{if } d - l \leq x < d + l \\
R_{rw} & \text{if } x \geq d + l,
\end{cases}
\]  

(3)

where \( 2l \) is the terrain transition size, which was set equal to the robot length, \( d \) is the length of the vegetation patch, and subindexes \( w \) and \( v \) stand for wood and vegetation respectively. Finally, in the case of steep hills, the gravity term needs to be included. For the positions \( x \) when all the wheels are on the hill, this term is given by \( G(x) = \frac{W}{r} \sin(\theta) \), where \( W \) is the robot weight and \( \theta \) is the hill inclination.

For the transitions regions from level to inclined ground that occur at the base and top of the hill, the gravitational and damping terms are modeled in a similar fashion to (3).

3) Motor Modeling: In order to exploit the vehicle dynamics, it is necessary to embed into the equations of motion a proper model of the motors, gears, and motor controller. Fig. 6 shows a speed-torque curve for the motor reflected at the robot wheels. That is, \( w \) is the wheel speed \([rad/s]\) and \( \tau \) is the torque at the wheel \([Nm]\). The operating line \( \nabla_m \) corresponds to the maximum voltage at the output of the motor controller, which is 14 V. The variable \( \nabla \) is the maximum output torque at the operating speed \( w \) and is given by

\[
\nabla = 2 \tau_n \left[ 1 - \frac{w}{w_{nl}} \right],
\]  

(4)

where \( w_{nl} \) corresponds to the no load speed of the wheel and \( \tau_n \) is the stall torque at the wheel; the latter two variables are computed using \( w_{nl} = \frac{k_m \tau_n}{g_r} \) and \( \tau_n = \frac{k_m \tau_{nl} g_r}{w_{nl}} \), where \( g_r \) is the gear ratio and \( \tau_{nl} \), and \( w_{nl} \) are the stall torque and no load speed of the motor at a nominal voltage of 12 V. For the particular motor state depicted in Fig. 6, the motor would be able to accelerate while the vehicle is on terrain \( a \) but it would be forced to decelerate on terrain \( b \) (here we assume that \( \tau_{na} \) and \( \tau_{nb} \) represent torque loads due to the two different terrains).

As mentioned in Section II-A, the control system allows the inclusion of current (equivalently, torque) limitation in the motor controllers \((\tau_{mc})\). Therefore, the maximum wheel torque at an operating wheel speed \( w \) is given by \( \tau_{max} = \min(\nabla, \tau_{mc}) \), which in terms of traction force, and assuming a wheel radius \( r \), can be expressed as:

\[
F_{max} = \min\left( \frac{\nabla}{r}, \frac{\tau_{mc}}{r} \right).
\]  

(5)

Notice that, we are assuming that the surface can generate that tractive force. In the future, more detailed wheel-terrain interaction models will be considered to cope with surfaces where traction is limited (it is important to point out that invertibility of the model is not a requirement for the selected motion planning approach).

III. MOTION PLANNING

The proposed solution to the momentum based traversal of mobility challenges is here formulated by utilizing sampling-based model predictive optimization (SBMPO), which is a sampling based motion planning algorithm capable of planning efficiently with dynamic models [10]. SBMPO employs A*-type optimization and produces fast computations.
if provided with a properly designed optimistic heuristic. In the SBMPO algorithm, a graph is created from start to goal and each vertex of the graph keeps track of the states of the system, the control input, and cost associated with the state.

The following are the main steps of SBMPO in the context of momentum based planning. (For details about SBMPO and a comparison with other motion planning algorithms [11], please refer to [10]; here we emphasize only the steps that are crucial to the problem at hand).

1) **Select a node with highest priority:** The nodes are collected in an Open List, which ranks them based on their priority for expansion. If the selected node is the goal SBMPO terminates, otherwise go to step 2.

2) **Sample control space:** Generate a sample of the control space. Since the vehicle model (1) is invertible, it was decided to sample the vehicle acceleration \( \ddot{x} \). However, it would also be possible to sample the applied joint torques. It is important to clarify that during execution, the robot does not measure acceleration directly. It employs wheel encoders and a PD control law to follow the desired position, velocity, and acceleration profiles.

3) **Generate Neighbor Nodes:** Integrate the system model with the control samples to determine the neighbors of the current node. Here we assume that the time \( t \in [NT, NT + T] \), where \( N \) is some positive integer and \( T \) is the sampling time. During this interval the desired acceleration \( \ddot{x}_d(t) \) is held constant at its sampled value \( \ddot{x}_d \). In this step each new sampled acceleration \( \ddot{x}_d \) is validated with the forward vehicle dynamic model (1), by computing the required traction force \( F_d \) to achieve the desired acceleration. That is:

\[
F_d = M \ddot{x}_d(NT) + C(x_d(NT), \dot{x}_d(NT)) + G(x_d(NT))
\]

if \( F_d > F_{\text{max}} \), with \( F_{\text{max}} \) given by (5) then the desired acceleration is modified by

\[
\ddot{x}_d \leftarrow M^{-1}[F_{\text{max}} - C(x_d(NT), \dot{x}_d(NT)) - G(x_d(NT))] \]

Finally, the acceleration is integrated to find the new vehicle velocity and position as follows:

\[
\dot{x}_d(t) = \ddot{x}_d(NT) + \dot{x}_d(t) t, \quad t \in [NT, NT + T),
\]

\[
x_d(t) = x_d(NT) + \dot{x}_d(NT) t + \frac{1}{2} \ddot{x}_d(t) t^2, \quad t \in [NT, NT + T)
\]

4) **Add new node to the graph.**

5) **Evaluate new node cost:** Use an A*-like heuristic to evaluate the cost of generated nodes based on the desired objective and add it to the priority queue based on the node’s cost. In the current work a minimum time heuristic inspired by [12] and detailed below is employed. This heuristic is a bound on the chosen cost and plays a key role in the efficient computation of trajectories that end with zero velocity.

Consider a system described by

\[
\ddot{x} = u; \quad x(0) = x_0, \quad \dot{x}(0) = v_0; \quad -a \leq u \leq b. \tag{10}
\]

The state space description of (10) is given by

\[
\dot{x}_1 = x_2, \quad \dot{x}_2 = u; \quad x_1(0) = x_{1,0}, \quad x_2(0) = v_0 \triangleq x_{2,0}. \tag{11}
\]

It is desired to find the minimum time needed to transfer the system from the initial state \((x_{1,0}, x_{2,0})\) to the final state \((x_1,0,0)\). Since the solution for transferring the system from \((x_{1,0}, x_{2,0})\) to the origin \((0,0)\) is easily

Fig. 7: Minimum time heuristic.

Fig. 8: Snapshots of the robot backing up to gather momentum to traverse the vegetation patch (Scenario 1).
extended to the more general case by a simple change of variable, for ease of exposition we assume below that $x_{1,f} = 0$.

Generalizing the results of [12], it is possible to show that the minimum time is the solution $t_f$ of

$$
\begin{align*}
L_f^2 - \frac{2x_{2,0}}{a} L_f &= \frac{x_{2,0}^2 + 2(a + b)x_{1,0}}{ab} \\
&\quad \text{if } x_{1,0} + \frac{x_{2,0} \cdot x_{2,0}}{2b} < 0,
\end{align*}
$$

and

$$
\begin{align*}
L_f^2 + \frac{2x_{2,0}}{b} L_f &= \frac{x_{2,0}^2 - 2(a + b)x_{1,0}}{ab} \\
&\quad \text{if } x_{1,0} + \frac{x_{2,0} \cdot x_{2,0}}{2a} > 0. \tag{12}
\end{align*}
$$

The minimum time ($t_f$) computed using (12) corresponds to a “bang-bang” optimal controller illustrated in Fig. 7, which shows switching curves that take the system to the origin using either the minimum or maximum control input (i.e., $u = -a$ or $u = b$).

For example, if $(x_{1,0}, x_{2,0})$ corresponds to point $p_1$ in Fig. 7, then the control input should be $u = -a$ until the system reaches point $p_2$ on the switching curve, corresponding to $u = b$. At this point the control input is switched to $u = b$, which will take the system to the origin.

It is also important to note that the heuristic defined by $t_f$ is optimistic, especially when the real system cannot implement the optimal bang-bang control due to actuator limitations at some points of the state space.

6) Repeat 2-5 for B (“Branch-out factor”) number of successors.

7) Repeat 1-6 until one of the stopping criteria is true:

Steps 1-6 will be repeated until the goal is found or the maximum number of allowable iterations is achieved.

IV. RESULTS

Before proceeding with the experimental validation of the proposed approach. It is important to clarify some aspects about the “Branch out” factor used by SBMPO for the problem considered in this paper, it is possible to argue that during each iteration of SBMPO, it would suffice to consider only three deterministic inputs each time. That is, given the current vehicle state, one can use the dynamic models of Section II to compute the maximum, minimum and 0 acceleration control inputs and let SBMPO generate the resulting trajectory based on these. In effect, this approach is effective for these type of scenarios, and should be used whenever possible due to its low computational time. However, for more general motion, where in addition to exploiting the momentum, the vehicle should avoid obstacles, it would be required to increase the number of control samples. Motivated by this reasoning, a branch out factor of 10 was selected for the following scenarios. In all experiments, the sampling time $T$ was set to 0.05s, which results in velocity profiles at 20Hz. However, since the robot is controlled at 1kHz (note: the robot is controlled using QNX real time operating system, and exhibits negligible latency), linear interpolation between consecutive samples was employed. Finally, to get a better understanding of the experiments please observe the video that accompanies the paper.

In Scenario 1, the motivational experiment of Section I was repeated for the stiff vegetation patch. However, in this case as shown in the snapshots of Fig. 8, the robot exploits its dynamics and finds a trajectory from the initial state $(x_0, x_0) = (-0.23m, 0.0m/s)$ to the final state $(x_f, x_f) = (1.8m, 0.0m/s)$, which requires the robot to back up to gather the required momentum to traverse the vegetation and reach the goal in near minimum time and at zero velocity. Fig. 9 compares the desired and actual position and velocity profiles.

It is interesting to point out that as predicted by the dynamics, and clearly shown in the velocity profile of Fig. 9, the slopes of the acceleration ($t < 1.6s$) and deceleration ($t < 2.3s$) regions are not symmetric. This is due to the fact that while the robot is decelerating, the rolling resistance is helping the motors. Figure 9 also shows the torque profiles, which illustrate the effectiveness of the manipulation of the vehicle momentum as a strategy to traverse the patch. Notice that for most of the region where the vehicle is traversing the patch, the wheel torques are saturated and the vehicle is decelerating (the observed variability in the torque correspond to the discrete location of the vegetation stems).

In Scenario 2, the robot was confronted with the vegetation patch but its initial state was set to $(-1.06m, 0m/s)$, which corresponds to the distance that the motion planner asked the robot to back up in the first scenario. The goal was kept at $(1.8m, 0.0m/s)$. As expected, the generated minimum time trajectory shown in Fig. 10 does not require the robot to back
Fig. 10: Position, velocity, and torque profiles (the torque corresponds to the average of the two robot sides), corresponding to Scenario 2. The robot starts its mission far from the vegetation patch and has time to gather the necessary momentum to traverse it and reach the goal in minimum time and at zero velocity.

up and the robot successfully completed its mission.

Scenario 3, corresponds to the steep hill experiment of the motivational experiment of Section I. The hill inclination remained at $10^\circ$ and the goal state was $(1.8m, 0.0m/s)$. As shown in the trajectory profiles of Fig. 11, the robot backed up the necessary distance to gather the required momentum to conquer the hill and reach the goal state in minimum time.

Scenario 4, corresponds to a steep hill scenario with an inclination of $16^\circ$ and serves to motivate important future research in the area of robust integration of perception and control. Two different experiments were conducted in this setup. Initially, accurate perception (i.e., correct hill inclination and correct initial and goal states) was assumed and as shown in the profiles of Fig. 12, the robot successfully conquered the hill by gathering the required momentum to reach the goal state $(1.8m, 0.0m/s)$ in minimum time. However, in a second experiment on the same setup, an erroneous estimate of the hill inclination was given to the planner. That is, the robot was told that the slope was $10^\circ$ when the actual value stayed at $16^\circ$. Notice from the trajectories of Fig. 12 that the robot failed to complete its mission due to the introduced perceptual error (it is important to clarify that for all the experiments corresponding to scenario 4, the maximum torque was limited to 60%). An additional experiment where the inclination of the hill was overestimated by $4^\circ$ was conducted and in that case the robot accomplished the mission successfully.

Scenario 5, illustrates an experiment in which a perception error of $1.03m$ was introduced in the initial location of the robot with respect to the vegetation patch. As shown in the position and velocity profiles of Fig. 13, the robot failed to complete its mission as it did not have the required momentum when it contacted the patch.

Finally, Table I summarize the motion planning results performed in an Intel i7 – 2600, 3.4GHz. It is important to emphasize that efficient computations of the minimum time trajectories were possible thanks to the properly selected
TABLE I: Motion Planning Results (Scenarios 1b and 2b correspond to trajectories using only maximum, minimum, and 0 accelerations. Scenario 4b corresponds to the experiment with erroneous perception of the hill inclination.)

<table>
<thead>
<tr>
<th>Scenario</th>
<th>Sampling Frequency[Hz]</th>
<th>Branch Factor</th>
<th>Computation time[s]</th>
</tr>
</thead>
<tbody>
<tr>
<td>1</td>
<td>20</td>
<td>10</td>
<td>0.253</td>
</tr>
<tr>
<td>1b</td>
<td>20</td>
<td>3</td>
<td>0.085</td>
</tr>
<tr>
<td>2</td>
<td>20</td>
<td>10</td>
<td>0.163</td>
</tr>
<tr>
<td>2b</td>
<td>20</td>
<td>3</td>
<td>0.043</td>
</tr>
<tr>
<td>3</td>
<td>20</td>
<td>10</td>
<td>0.141</td>
</tr>
<tr>
<td>4</td>
<td>20</td>
<td>10</td>
<td>0.119</td>
</tr>
<tr>
<td>4b</td>
<td>20</td>
<td>10</td>
<td>0.028</td>
</tr>
<tr>
<td>5</td>
<td>20</td>
<td>10</td>
<td>0.803</td>
</tr>
</tbody>
</table>

minimum time heuristic.

V. CONCLUSIONS AND FUTURE WORK

In this work, a strategy to exploit momentum as a way to deal with mobility challenges typically encountered by mobile robots in the field was proposed. The presented methodology integrates dynamic models of the robot, actuators, and terrain with an efficient motion planner. In particular, the approach was demonstrated in scenarios in which the robot faces patches of stiff vegetation and steep hills. In addition to overcome the mobility challenge, the robot completes its mission in near minimum time and with a desired velocity at the goal.

Future work involves the extension of the proposed approach to curvilinear motion. Additionally, the proposed approach will be evaluated in outdoor mobility challenges that demand more detailed wheel-interaction models. This include slippery hills, water bodies, sand and mud patches. As pointed out in Section II-B.2, it is required to integrate the current research work with perception to achieve full autonomy in the traversal of these challenging scenarios. Concretely, the perception system should be able to detect the presence of mobility challenges and characterize them through association with previously experienced and perceptually similar terrains. Another big challenge, motivated by the experiments of Scenarios 4 and 5, consists in the development of principled ways to translate the perceptual data to the motion planner and low level controllers in a way that preserves robustness against perception errors and model uncertainty but still avoids overly conservative solutions. Finally, online learning techniques will be exploited to adapt the vehicle and friction models as the robot navigates over the different mobility challenges.

ACKNOWLEDGMENT

This work was supported by the collaborative participation in the Robotics Consortium sponsored by the U.S. Army Research Laboratory under the Collaborative Technology Alliance Program, Cooperative Agreement DAAD 19-01-2-0012, and funded by National Science Foundation under Award CMMI-0927040. The U.S. Government is authorized to reproduce and distribute reprints for Government purposes not withstanding any copyright notation thereon.

REFERENCES