A Universal Footstep Planning Methodology for Continuous Walking in Challenging Terrain Applicable to Different Types of Legged Robots

Alexander Stumpf* and Oskar von Stryk*

***Abstract*— In recent years, the capabilities of legged locomotion controllers have been significantly advanced enabling them to traverse basic types of uneven terrain without visual perception. However, safely and autonomously traversing longer distances over difficult uneven terrain requires appropriate motion planning using online collected environmental knowledge. In this paper, we present such a novel methodology for generic closed-loop preceding horizon footstep planning that enables legged robots equipped with capable locomotion controllers to autonomously traverse previously unknown terrain while continuously walking long distances. Hereby, our approach addresses the challenge of online terrain perception and soft real-time footstep planning. The proposed new formulation of the search-based planning problem makes no specific assumptions about the robot kinematics (e.g. number of legs) or the used locomotion control schemes. Therefore, it can be applied to a broad range of different types of legged robots. Unlike current methods, the proposed new framework can optionally consider the floating base as part of the state-space. It is possible to configure the complexity of the planner online, from efficiently solving tasks in flat terrain to using non-contiguous contacts in highly challenging terrain. Finally, the presented methodology is successfully applied and evaluated in virtual and real experiments on state of the art bipedal, quadrupedal, and a novel eight-legged robot.**

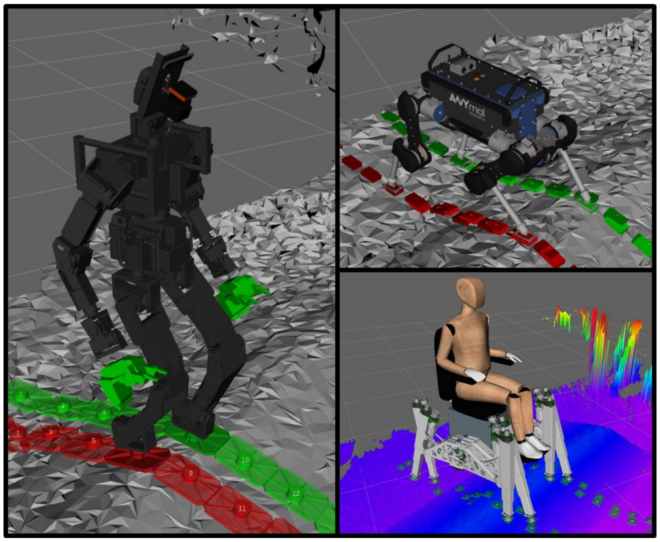

Fig. 1. The novel footstep planning methodology we present is suitable for a broad range of different types of legged robots, e.g.: THORMANG3 (left), ANYmal (top), Walkerchair (bottom).

I. INTRODUCTION

In the last decade, the locomotion capabilities of legged robots have improved significantly and now offer the potential for first real-world applications. New Whole-Body Control (WBC) approaches for bipedal robots [1][2][3][4] have been applied to capable robots such as Boston Dynamics' Atlas or NASA's Valkyrie [5] and demonstrated exceptional capabilities in traversing terrain [6] up to walking on edges with partial footholds [7].

Spot [8], ANYmal [9], and many other quadrupedal robots have also demonstrated sophisticated WBC capabilities [10][11][12] enabling them to cross challenging real-world terrain. As almost all quadrupeds currently use ball-shaped feet, their practical use in real-world seems to be reduced when walking over compressible soil or slippery surfaces [13] anticipating the use of planar-shaped feet [14], which requires more sophisticated contact planners.

In general, solving the WBC problem is a computationally difficult task that is usually solved in hard real-time using predetermined contacts [15][16][17][18]. Such walking control schemes can also be deployed in perception-less applications [2][19], although any deviation from the assumed terrain results in a considerable perturbation that requires sophisticated balance control.

However, perception-less walking and relying on reactive balance controls is strongly discouraged because of the imminent risk of serious fall overs:

- Capture steps should account for obstacles at low-level control as in [20], otherwise stepping on another obstacle will still result in a fall.
- Reactive walk controllers can get stuck in repetitive motions without making progress [21].
- Climbing over large obstacles may not be performed safely due to unexpected contacts.
- Knowledge about the expected foothold, contact area, and resulting exerted forces improve surefootedness [2][7][22].

These observations strongly motivate using contact planners. When the entire robot body is taken into account, such contact planners can solve the broadest range of locomotion tasks (see Fig. 2) but are usually computationally demanding [23] or require precomputed world models to efficiently solve convex optimization problems [15]. Such whole-body contact planners often first seek a guiding path along which a feasible sequence of contacts is then generated. However, finding such a guiding path that guarantees the existence of a feasible sequence of contacts (equilibrium feasibility) is a challenging task and can be approximated by reachability condition, as shown in [24].

As in most cases, walking is a ground-contact-constrained

---

*Department of Computer Science, Technical University of Darmstadt
`stumpf,stryk@sim.tu-darmstadt.de`

task, where finding a valid contact sequence can be done more efficiently by focusing on footstep planning. With Mixed Integer Quadratically Constrained Quadratic Program (MIQCQP), this problem is efficiently solved [25] but requires that all constraints be in a quadratic convex form. Since we live in a highly non-convex world, this may require expensive upstream computing steps [26].

In contrast, this work is based on search-based planners that traverse a graph dynamically constructed from a predefined action set. Based on task-specific cost and heuristic functions, the best path is found using graph-based algorithms such as ARA* [27]. A suitably chosen action set can easily bypass the equilibrium feasibility problem but may not take into account the robot's full locomotion capabilities. Search-based footstep planning has a rich history of publications [28][29][30] but few deal with uneven flat surfaces [31] or rough terrain [6][32]. In this work, we even demonstrate a solver for irregular terrain tasks that includes curved and even non-contiguous contacts (see Fig. 2).

A footstep planner is often strongly tied to a specific robot platform or at least to the basic kinematic structure, such as for bipeds [6][30][33] or for quadrupeds [22]. Our preceding modular and reusable approach also applies only for bipedal robots [32]. There have been few attempts to provide a hardware abstraction layer for motion planners such as Free Gait [34]. Footstep plans are typically passed directly to the motion controller using specialized bridging software that is not designed for reuse for other robot systems [35][36]. However, due to locomotion drift and incomplete knowledge of the terrain, step plans over long distances are not feasible. While drift can be compensated using localization [37], continuous locomotion approaches such as [38] address both problems by using short horizon periodic planning that is feasible in soft real-time. This work addresses such limitations and challenges.

A. Contributions

This paper provides a basic overview of the novel Legged Locomotion Library (L3) framework. L3 provides generic methodologies for autonomous legged locomotion that apply to many different types of robots, as depicted in Fig. 1. We present a new holistic architecture for autonomous continuous walking that covers the full sense-plan-act cycle:

- L3 Terrain Model Generator: Efficient and modular terrain perception and mapping,
- L3 Footstep Planner: Generalized modular 3D footstep planning framework using a novel state-action modeling approach while counteracting state-space complexity,
- L3 Step Controller: Hardware abstraction layer to handle very different types of legged robots and control schemes while providing closed-loop footstep planning for autonomous continuous walking.

This work builds on our predecessor plugin-based 3D footstep planner for bipedal robots in rough terrain [32]. We demonstrate how the extensions made by L3 enables autonomous continuous walking with low migration overhead for different types of legged robots thanks to the plugin system (see Fig. 6) used for cost estimation, heuristic, collision checks, etc. [32], and a novel state-action modeling approach. In addition to current footstep planners as in [6][22][25][30][32], our approach does not rely on cyclic gaits and allows finding a suitable floating base pose for each foothold configuration and vice-versa, providing better initialization for trajectory optimization.

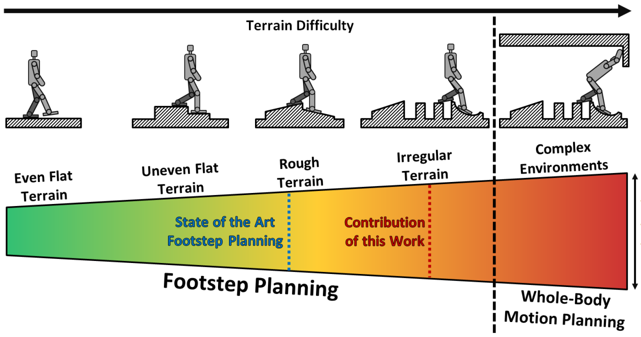

Fig. 2. Classification of different terrain challenges used in this paper.

Given the complexity of the presented topic, this paper focuses on the high-level perspective of how the L3 framework solves these challenging tasks. More in-depth explanations and analyses of the applied methods are beyond the scope of this paper and will be left to follow work. However, for advanced research, we make L3 available as a generic open-source tool[1]. By taking this step, we aim to support the research community to further advance this research topic, as many current and future integrated methods can be easily shared and applied to many different types of legged robots with little or no overhead. This framework allows us to focus on open research topics, such as the method presented in this work for finding step plans using footholds with non-contiguous contacts in irregular terrain (see Fig. 2), which to the best of the authors' knowledge has not been attempted before.

B. Paper Outline

Sec. II summarizes the prerequisites for the proposed approach presented in Sec. III. The results of applying the L3 framework to different types of legged robots are then demonstrated in Sec. IV. We conclude the presented work in Sec. V.

II. PREREQUISITES

A typical locomotion control architecture consists of contact planning, centroidal dynamics, and whole-body control [39]. As this work focuses on contact planning of footsteps, it relies on these assumptions:

- locomotion platform is a legged robot,
- appropriate controls for the intended locomotion task,
- control system accepts predetermined foot contacts,
- 3D terrain awareness and self-localization available,

while neither specific kinematics (e.g. number of legs), fixed gait cycles, nor specific motion control schemes are required.

---

[1] https://github.com/tu-darmstadt-ros-pkg/legged_locomotion_library

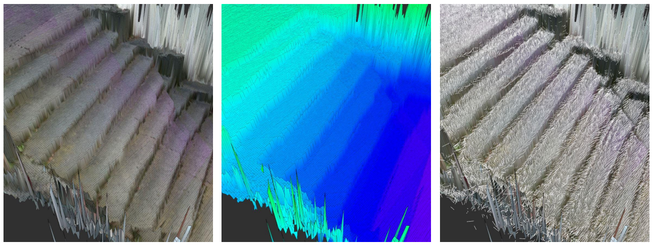

Fig. 3. Example of acquired data from a staircase leading to a sufficient accurate colored 3D world model (left) using an elevation map (middle) and providing surface normals (right).

III. METHODOLOGY

In the following, an overview of the main methods and components of the new L3 framework are described in a highly condensed form based on [40]. While the L3 framework aims to improve locomotion autonomy of legged robots by providing a holistic solution that addresses the respective sense, plan, act components, this paper focuses mainly on the footstep planning part. All presented methods are implemented as generic reusable plugins provided in our open-source release.

A. Sense: Efficient & Sophisticated Foothold Estimation

The L3 terrain model generator is the successor of [32], which now uses a plugin system that allows us to freely compose our world modeling processing chain using different data sources such as 3D laser scanners or RGBD-cameras. In this work, we use the grid map from [41] and the surface normal estimation based on Principal Component Analysis (PCA) from our previous work [32] (see Fig. 3).

1) Center-Contact Point (CCP) Foothold Estimation: We have presented in [42] the CCP foothold estimation method, which snaps the foot to the ground by obtaining the height from the elevation map and using surface normals precomputed with PCA. This method has been shown to be very efficient in 3D footstep planning [43]. We have also successfully tested this approach with our footstep planner by autonomously traversing the most difficult terrain type of the DRC Finals 2015 using an Atlas robot. This efficiency comes with the disadvantage that non-contiguous foot contacts cannot be determined due to the constraint of ensure ground-contact with the center of the sole.

2) Multi-Contact Point (MCP) Foothold Estimation: To take advantage of a planar-shaped foot, a feasible foot support polygon using non-contiguous contacts in irregular terrain must be found, which is a highly challenging task.

We adapted the foot landing state estimation approach [44] for our purposes of efficient 3D footstep planning. A section of the 2D footprint from the elevation map at the given position is converted into a point cloud of which the convex hull, represented as facets, is computed using the Quickhull algorithm [45]. The highest facet projected into the x/y-plane and encircling a given point (e.g. a reference Center of Pressure (COP)) is then selected as the resulting planar support polygon on which the foothold is snapped. The robustness of this approach is further improved by merging all nearby near-coplanar facets, resulting in the largest feasible convex planar support polygon for the foothold. The feasibility of the solution is validated by verifying if the resulting foot support polygon encloses a predefined stability support polygon that represents the desired minimum foot contact area required for a stable walk, which should also consider a reasonable margin for balance control.

This novel MCP foothold estimation approach has been successfully tested in generating suitable non-contiguous foot contacts in one of the most challenging scenarios we named the “Bottleneck Test” (see Fig. 4). A more detailed evaluation shows that the MCP method significantly improves the usable search-space for valid footholds in such irregular terrain compared to the CCP method (see Fig. 5). However, the current MCP implementation takes about 40 times more time than CCP (0.01ms vs. 0.40ms) to evaluate a single foot position, which will be addressed in future work.

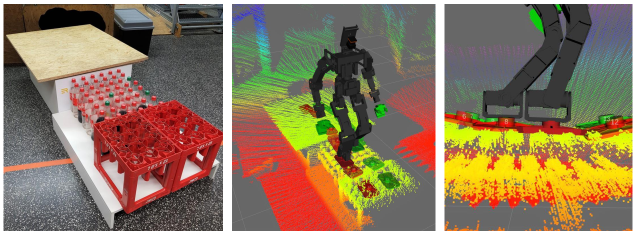

Fig. 4. The “Bottleneck Test” consists of an array of bottles representing small isolated contact areas (left). The MCP approach can find suitable step plans (middle) where individual steps form non-contiguous contacts (right).

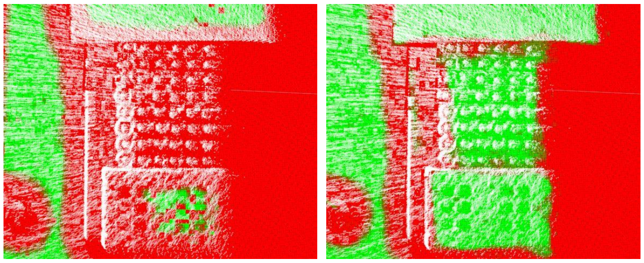

Fig. 5. Available search-space for valid footholds (green areas): The MCP (right) clearly outperforms the CCP (left) in the “Bottleneck Test”.

B. Plan: Multi-Legged Footstep Planning

The following section gives an overview of contributed footstep planning methods, including a novel variadic state-action representation as used by our SBPL[2]-based L3 footstep planner that currently supports A*, ARA*, and R*. It also provides the flexibility needed to deal with very different types of legged robots, leaving aside any assumptions about kinematics (especially the number of legs).

1) State: A state $\mathcal{S}$ is now defined as the set of all unique feet (footholds) $f \in \mathcal{F}$, which are in contact with the environment

$$\mathcal{S} \subseteq \mathcal{F}, \tag{1}$$

where $\mathcal{F}$ denotes all possible footholds and a foot index $f_{idx} \in \mathcal{F}_{idx}$ must occur only once in $\mathcal{S}$. Note that feet

---

[2] http://sbpl.net

without contact do not belong to the state, so $f \notin \mathcal{S}$. The flight phases $\mathcal{S} = \emptyset$ for the feet are valid states as long as the floating base is moving, which will be further elaborated in future work.

2) Gait Generation: A possible transition between states is primarily defined by the gait function. We define a generic gait function as

$$\mathcal{G} : F_{idx} \in \mathcal{P}(\mathcal{F}_{idx}) \mapsto \mathcal{N} \subseteq \mathcal{P}(\mathcal{F}_{idx}), \tag{2}$$

where $\mathcal{P}(A)$ denotes the powerset of the set A. This function maps a set of foot indexes F_{idx}, representing all moved legs during the last step, to all possible subsequent movements of legs $F'_{idx} \in \mathcal{N}$. In this way, the planner can account for acyclic gaits and optionally perform gait optimization. Note that the quantity $|\mathcal{N}|$ affects the planning time exponentially.

3) Transition: In contrast to our previous work, an action $a \in \mathcal{A}$ is now parameterized by the gait function defining the next moving feet $F'_{idx} \in \mathcal{G}(F_{idx})$, leading to the new transition model t to generate all successor states $\mathcal{S}'$:

$$succ(\mathcal{S}) = \{\mathcal{S}' \mid \mathcal{S}' = t(\mathcal{S}, a(\mathcal{G}(F_{idx})), a \in \mathcal{A}\}. \tag{3}$$

In the case of $|\mathcal{G}(F_{idx})| > 1$ the planner automatically performs gait optimization but this can result in an exponentially growing number of successors.

4) Step: The explicit consideration of steps allows us to model e.g. reachability checks and cost functions more easily and effectively. A step $\boldsymbol{S}$ describes the contact changes between two successive states $\mathcal{S}$ and $\mathcal{S}'$ by providing the step data $\Delta(f, f') \in \Delta(\mathcal{S}, \mathcal{S}')$ for each moved leg:

$$\Delta(f, f') = \langle f, f', \Delta\boldsymbol{P} \rangle, \tag{4}$$

where $\Delta\boldsymbol{P}$ is the positional delta of the two foothold poses $\boldsymbol{P}_f$ and $\boldsymbol{P}_{f'}$ given as homogeneous transformation.

$$\Delta\boldsymbol{P} \leftarrow \boldsymbol{P}_f^{-1} \cdot \boldsymbol{P}_{f'}. \tag{5}$$

In addition, all footholds of non-moving (support) legs:

$$\Box(\mathcal{S}, \mathcal{S}') = \{f_i \mid f_i = f'_i\}, \tag{6}$$

are included as well resulting in the new step representation:

$$\boldsymbol{S}(\mathcal{S}, \mathcal{S}') = \langle \Delta(\mathcal{S}, \mathcal{S}'), \Box(\mathcal{S}, \mathcal{S}'), id \rangle, \tag{7}$$

where $id \in \mathbb{N}_0$ is the assigned step index.

5) Planning State: This formulation leads to the definition of the planning state, which is now composed by the successor state $\mathcal{S}'$, the predecessor state $\mathcal{S}$ and the step $\boldsymbol{S}(\mathcal{S}, \mathcal{S}')$:

$$\boldsymbol{p} = \langle \mathcal{S}', \mathcal{S}, \boldsymbol{S}(\mathcal{S}, \mathcal{S}') \rangle. \tag{8}$$

Unlike our earlier work, the planning state does not handle discretized values or state hashing, which have been offloaded to the new state manager.

In summary, this novel variadic state-action formulation leads to generic and effective modeling for legged locomotion planning that can cope with a broad range of different robot types. Moreover, the proposed plugin infrastructure enables the easy introduction of additional methods and details required for the specific locomotion task.

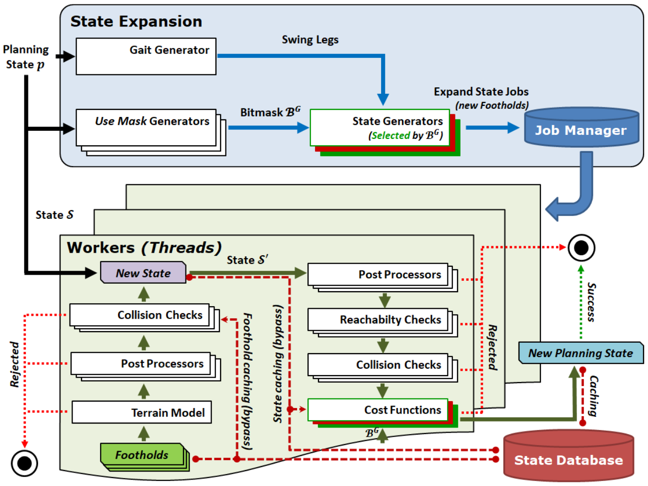

Fig. 6. The processing pipeline of the state extension as implemented by the *L3* footstep planner. Stacked boxes indicate that multiple instances of the plugin type can be loaded simultaneously.

6) State Manager: Footholds are discretized in an x/y/z/yaw lattice, and by applying a recursive hashing strategy, each previously generated foothold, state, and planning state is efficiently addressed in a hash table used by the state manager. During state expansion, the planner can omit computational demanding processing steps if the foothold or state is already known to the state manager (see Fig. 6).

7) State Expansion: The highly multithreaded state expansion runs in two phases (see Fig. 6). First, the job manager stores all by the gait function and action set generated possible successors in an x/y-planar state-space represented as jobs. After all jobs have been generated, those are processed by parallelized workers, which determine the full 3D foothold (e.g. using MCP), generate, and evaluate the resulting states. Hereby, the workers omit computational demanding steps when either the foothold or the resulting state has been already stored in the state manager.

8) Intelligent Adaptive State Expansion: The L3 footstep planner introduces a unique concept to solve planning tasks smarter (and not harder). A situation-aware intelligent decision process can dynamically enable or disable the use of specific plugins during planning. For this purpose, each plugin can be controlled by an assigned "use mask" controlled by the "use mask generator" (see Fig. 6), which implements task-specific decision rules to (de-)activate specific plugins. Fig. 7 exemplifies the significant reduction of state expansions by introducing sparse sampling in obstacle-free regions. This system can also be used to detect and overcome depression regions where the planner easily gets stuck, e.g., by adapting heuristics and gait generation for a quadrupedal robot when turning on the spot.

9) Floating Base State Expansion: For simplicity, we have not considered the floating base in the previous notation. However, the planner can also plan floating base poses, since the state $\mathcal{S}$ consists not only of footholds but also contains a floating base pose. The gait function can signal that the floating base should be moved alone or in combination with a

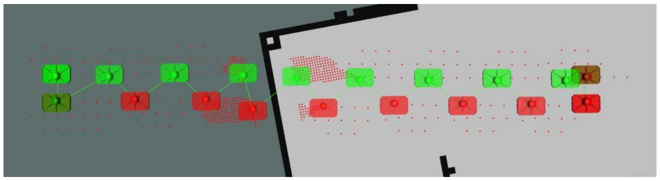

Fig. 7. Example of adaptive state sampling considering an occupancy grid map: Sparse sampling is performed in free space, while sampling density is increased near walls.

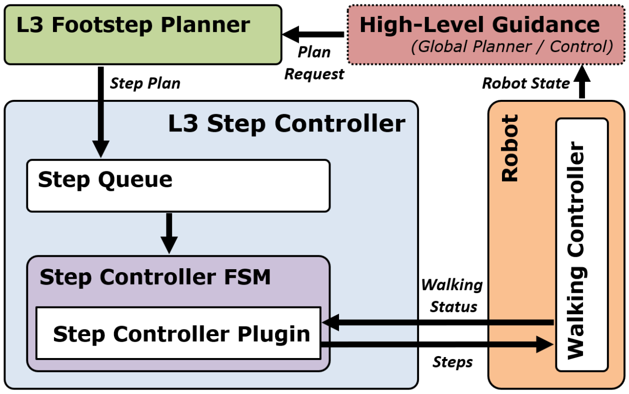

Fig. 8. Continuous walking cycle using the step controller.

particular foot pattern. The entire processing chain is capable to generate a suitable floating base pose and to take this into account in all processing steps, e.g. in evaluating cost functions, heuristics, and collision checks.

C. Act: Hardware Abstraction & Continuous Walking

Continuous walking requires smooth synchronization of the motion execution and the planning pipeline. The new L3 step controller provides the crucial logic for this task as pre-implemented Finite State Machine (FSM) while providing a generic infrastructure to bridge both systems (see Fig. 8). A smart step queue seamlessly stitches sequential footstep plans. The signal processing of the FSM is implemented by a robot-specific step controller plugin that directly interfaces with the robot's motion control and bi-directionally forwards all necessary data to realize continuous walking behavior.

A high-level guidance system (e.g., a global path planner) can now easily close the autonomous locomotion control loop. In this work, we provide a basic "carrot on a stick" behavior that can be replaced by more complex strategies. Based on the state of the robot and the feedback from the step controller, the guidance system can create an appropriate planning request with suitable start and goal states to update and extend the current step queue (see Fig. 8).

IV. RESULTS

In this section, the proposed methods are systematically evaluated with different types of legged robots. The experiments were performed using a Intel® Core™ i7-4800MQ machine (4 cores, 8 threads) with either a real robot system or a physically realistic simulation in Gazebo.

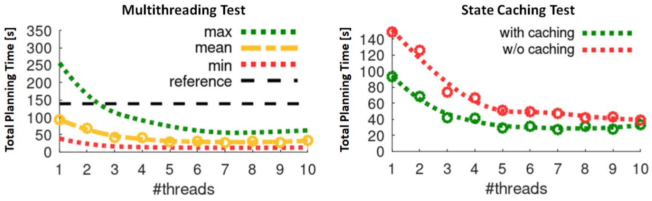

Fig. 9. Evaluating multithreading (left) and state caching (right) shows a significant performance increase. The black line shows the reference value for single-threaded planning.

A. Handling State-Space Complexity

The effectiveness of multithreading and state caching is evaluated using a high-density (0.01 m) search-space and a high-branching factor scenario where each iteration of ARA* generates 6714 different footsteps with a bipedal robot model. For each thread count, 10 trials are performed with deterministic, randomly selected start and target poses for the planning request. The results in Fig. 9 show the performance improvement of both methods. Note that the observed stagnation above the real core count of 4 is likely due to hardware shared by the virtual cores.

B. Application to Different Types of Legged Robots

We have successfully applied the L3 to very distinct types of legged robots, where the proposed genericity and modularity allow us to easily reuse the just presented methods for all demonstrated robots. Here, the planner is configured to use a maximum time budget of 10s for plan optimization.

1) Bipedal Robot: We have fully migrated L3 to the THORMANG3 robot built by ROBOTIS. Here, the L3 step controller is seamlessly connected to a ZMP-based motion controller, which allows us to execute the generated footstep plans with the real robot, as shown in Fig. 10.

2) Quadrupedal Robot: We evaluated our approach using the ANYmal-B robot with two different versions: ball and planar-shaped feet, both seamlessly supported by our planner. The step controller interfaces ANYmal's WBC via Free Gait [34] and allows the execution of step plans using a receding horizon controller, as shown in Fig. 11. Rotation on the spot with quadrupeds quickly leads to a depression region, which our planner can overcome using the adaptive state expansion approach described above.

3) Eight-legged Robot: Walkerchair is an eight-legged robot that can carry heavy loads and is currently under

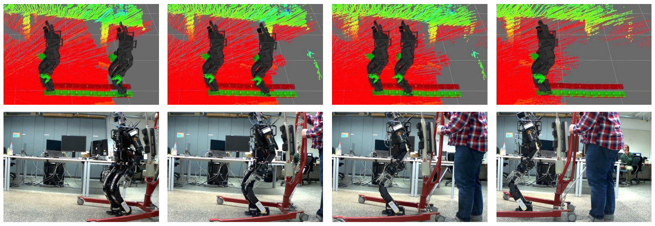

Fig. 10. The top row shows the planned path by L3 using the goal depicted by the robot on the left. The bottom row shows the execution of the step plan with the L3 step controller on the real robot.

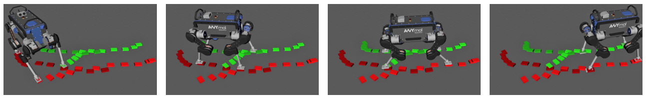

Fig. 11. Step plan for the ANYmal using planar-shaped feet. This image sequence shows the execution in a physically realistic simulation in Gazebo.

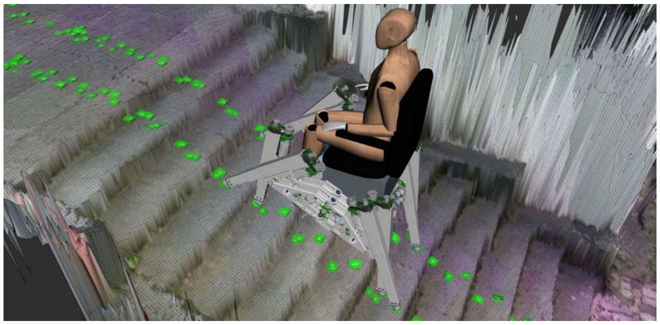

Fig. 12. Our approach to footstep planning also applies to the eight-legged Walkerchair using real-world terrain data.

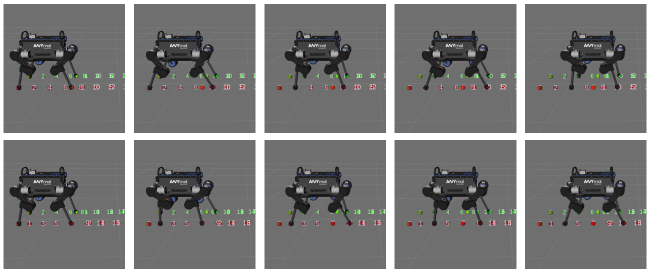

Fig. 13. Visualization of different quadruped gaits applied by our planner: Gallop (upper row) and Trot (lower row).

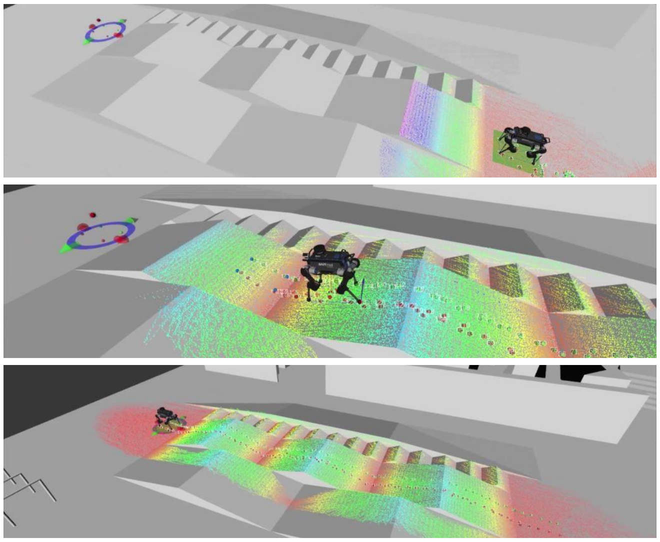

Fig. 14. The ANYmal robot autonomously traverses previously unknown ramps in a physically realistic simulation in Gazebo using the proposed holistic L3 framework.

development in our research group. Fig. 12 shows initial results of applying the L3 footstep planner to an early version of this robot. Here, our planner explicitly considers the floating base to provide more postural stable poses. The collision check of the legs is not yet integrated, but can easily be provided e.g. as an additional collision check plugin.

C. *Gait Generation*

The novel state-space formulation allows for the easy application of different gaits, as shown in Fig. 13. However, the resulting exponential number of states emphasizes the use of the proposed multithreading and adaptive state generation.

D. *Continuous Walking*

The presented step control architecture allows THOR-MANG3 and ANYmal to walk continuously, although each uses different control systems. In Fig. 14 a long-term experiment shows how ANYmal explores the terrain, continuously re-plans and extends the step queue until it reaches the goal. This experiment shows the importance of continuous walking since the robot cannot see the terrain in the direction of the goal from the beginning and therefore needs a reactive planning loop. Here, the planner is configured to use CCP and a maximum time budget of 2s for planning, which corresponds to a typical cycle for a single step. When goals are placed in close proximity (<2m), the initial result is found very quickly (<1s), leaving a reasonable amount of time to optimize the step plan. This experiment was conducted in real-time using the Gazebo simulation, which shows that the L3 planner generates short horizon plans sufficiently fast to keep the receding horizon controller running.

V. CONCLUSION & OUTLOOK

In this paper, we presented the novel generic footstep planning methodology implemented in our L3 framework that enables autonomous legged locomotion for different types of robots. It provides the required tools necessary to generate a terrain model online, find suitable footstep sequences in highly challenging irregular terrain, and controlling closed-loop footstep execution. A novel, generalized state-space formulation that neither relies on specific kinematic assumptions (e.g., the number of legs) nor requires a specific control scheme to execute the step plan achieves the challenging goal of supporting a broad range of different legged robots. We have successfully demonstrated methods such as multithreading, state caching, and intelligent adaptive state expansion to counteract the exponential growth of state-space caused by increasing the number of legs. Our generic step controller enables autonomous continuous walking while interfacing robot-specific motion libraries and supporting receding horizon controllers. All proposed methods have been successfully evaluated with three different types of legged robots.

In future work, we would like to provide more real-world examples using our approach, we are especially excited to perform the “Bottleneck Test” with a real legged robot.

VI. ACKNOWLEDGMENT

The presented research has been supported in parts by the Distr@l program of the state of Hesse (Germany). Part of this work has been conducted as part of ANYmal Research, a community to advance legged robotics.

REFERENCES

[1] S. Feng, X. Xinjilefu, C. G. Atkeson, and J. Kim, "Optimization based controller design and implementation for the Atlas robot in the DARPA Robotics Challenge Finals," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2015, pp. 1028–1035.

[2] T. Koolen, S. Bertrand, G. Thomas, T. De Boer, T. Wu, J. Smith, J. Englsberger, and J. Pratt, "Design of a momentum-based control framework and application to the humanoid robot Atlas," *International Journal of Humanoid Robotics*, vol. 13, no. 01, 2016.

[3] M. Kudruss, M. Naveau, O. Stasse, N. Mansard, C. Kirches, P. Soueres, and K. Mombaur, "Optimal control for whole-body motion generation using center-of-mass dynamics for predefined multi-contact configurations," in *2015 IEEE-RAS 15th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2015, pp. 684–689.

[4] Russ Tedrake and the Drake Development Team. (2019) Drake: A planning, control, and analysis toolbox for nonlinear dynamical systems. http://drake.mit.edu. Accessed: 2021-09-15.

[5] N. A. Radford, P. Strawser, K. Hambuchen, J. S. Mehling, W. K. Verdeyen, A. S. Donnan, J. Holley, J. Sanchez, V. Nguyen, L. Bridgwater *et al.*, "Valkyrie: Nasa's first bipedal humanoid robot," *Journal of Field Robotics*, vol. 32, no. 3, pp. 397–419, 2015.

[6] R. J. Griffin, G. Wiedebach, S. McCrory, S. Bertrand, I. Lee, and J. Pratt, "Footstep planning for autonomous walking over rough terrain," *arXiv preprint arXiv:1907.08673*, 2019.

[7] G. Wiedebach, S. Bertrand, T. Wu, L. Fiorio, S. McCrory, R. Griffin, F. Nori, and J. Pratt, "Walking on partial footholds including line contacts with the humanoid robot Atlas," in *Humanoid Robots (Humanoids), 2016 IEEE-RAS 16th International Conference on*. IEEE, 2016, pp. 1312–1319.

[8] Boston Dynamics. (2018) Spot. https://www.bostondynamics.com/spot. Accessed: 2021-09-15.

[9] M. Hutter, C. Gehring, D. Jud, A. Lauber, C. D. Bellicoso, V. Tsounis, J. Hwangbo, K. Bodie, P. Fankhauser, M. Bloesch *et al.*, "ANYmal–A highly mobile and dynamic quadrupedal robot," in *Intelligent Robots and Systems (IROS), 2016 IEEE/RSJ International Conference on*. IEEE, 2016, pp. 38–44.

[10] C. D. Bellicoso, F. Jenelten, P. Fankhauser, C. Gehring, J. Hwangbo, and M. Hutter, "Dynamic locomotion and whole-body control for quadrupedal robots," in *2017 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS 2017)*. IEEE, 2017.

[11] M. Hutter, H. Sommer, C. Gehring, M. Hoepflinger, M. Bloesch, and R. Siegwart, "Quadrupedal locomotion using hierarchical operational space control," *The International Journal of Robotics Research*, vol. 33, no. 8, pp. 1047–1062, 2014.

[12] O. Villarreal, V. Barasuol, P. Wensing, and C. Semini, "Mpc-based controller with terrain insight for dynamic legged locomotion," *arXiv preprint arXiv:1909.13842*, 2019.

[13] M. Tranzatto, F. Mascarich, L. Bernreiter, C. Godinho, M. Camurri, S. Khattak, T. Dang, V. Reijgwart, J. Loeje, D. Wisth *et al.*, "Cerberus: Autonomous legged and aerial robotic exploration in the tunnel and urban circuits of the darpa subterranean challenge," *arXiv preprint arXiv:2201.07067*, 2022.

[14] R. Käslin, H. Kolvenbach, L. Paez, K. Lika, and M. Hutter, "Towards a passive adaptive planar foot with ground orientation and contact force sensing for legged robots," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 2707–2714.

[15] J. Carpentier and N. Mansard, "Multicontact locomotion of legged robots," *IEEE Transactions on Robotics*, vol. 34, no. 6, pp. 1441–1460, 2018.

[16] M. Geisert, T. Yates, A. Orgen, P. Fernbach, and I. Havoutis, "Contact planning for the ANYmal quadruped robot using an acyclic reachability-based planner," *arXiv preprint arXiv:1904.08238*, 2019.

[17] B. Ponton, A. Herzog, S. Schaal, and L. Righetti, "A convex model of humanoid momentum dynamics for multi-contact motion generation," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2016, pp. 842–849.

[18] C. Mastalli, R. Budhiraja, W. Merkt, G. Saurel, B. Hammoud, M. Naveau, J. Carpentier, L. Righetti, S. Vijayakumar, and N. Mansard, "Crocoddyl: An efficient and versatile framework for multi-contact optimal control," in *2020 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2020, pp. 2536–2542.

[19] C. D. Bellicoso, C. Gehring, J. Hwangbo, P. Fankhauser, and M. Hutter, "Perception-less terrain adaptation through whole body control and hierarchical optimization," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2016, pp. 558–564.

[20] A.-C. Hildebrandt, R. Wittmann, D. Wahrmann, A. Ewald, and T. Buschmann, "Real-time 3D collision avoidance for biped robots," in *2014 IEEE/RSJ International Conference on Intelligent Robots and Systems*. IEEE, 2014, pp. 4184–4190.

[21] D. Belter, "Optimization-based approach for motion planning of a robot walking on rough terrain," *Journal of Automation Mobile Robotics and Intelligent Systems*, vol. 7, 2013.

[22] P. Fankhauser, M. Bjelonic, D. Bellicoso, T. Miki, and M. Hutter, "Robust rough-terrain locomotion with a quadrupedal robot," in *IEEE International Conference on Robotics and Automation (ICRA 2018)*. IEEE, 2018.

[23] K. Bouyarmane and A. Kheddar, "Multi-contact stances planning for multiple agents," in *IEEE International Conference on Robotics and Automation (ICRA)*, 2011, pp. 5246–5253.

[24] S. Tonneau, A. Del Prete, J. Pettré, C. Park, D. Manocha, and N. Mansard, "An efficient acyclic contact planner for multiped robots," *IEEE Transactions on Robotics*, vol. 34, no. 3, pp. 586–601, 2018.

[25] R. Deits and R. Tedrake, "Footstep planning on uneven terrain with mixed-integer convex optimization," in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2014, pp. 279–286.

[26] ——, "Computing large convex regions of obstacle-free space through semidefinite programming," in *Algorithmic foundations of robotics XI*. Springer, 2015, pp. 109–124.

[27] M. Likhachev, G. J. Gordon, and S. Thrun, "ARA*: Anytime A* with provable bounds on sub-optimality," in *Advances in neural information processing systems*, 2004, pp. 767–774.

[28] J. J. Kuffner, K. Nishiwaki, S. Kagami, M. Inaba, and H. Inoue, "Footstep planning among obstacles for biped robots," in *Intelligent Robots and Systems, 2001. Proceedings. 2001 IEEE/RSJ International Conference on*, vol. 1. IEEE, 2001, pp. 500–505.

[29] J. E. Chestnutt, K. Nishiwaki, J. Kuffner, and S. Kagami, "An adaptive action model for legged navigation planning," in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2007, pp. 196–202.

[30] A. Hornung, D. Maier, and M. Bennewitz, "Search-based footstep planning," in *ICRA Workshop Progress & Open Problems in Motion Planning & Navigation for Humanoids*. IEEE, 2013.

[31] D. Maier, C. Lutz, and M. Bennewitz, "Integrated perception, mapping, and footstep planning for humanoid navigation among 3D obstacles," in *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2013, pp. 2658–2664.

[32] A. Stumpf, S. Kohlbrecher, D. C. Conner, and O. von Stryk, "Open source integrated 3D footstep planning framework for humanoid robots," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2016, pp. 938–945.

[33] S. Kajita, F. Kanehiro, K. Kaneko, K. Fujiwara, K. Harada, K. Yokoi, and H. Hirukawa, "Biped walking pattern generation by using preview control of zero-moment point," in *2003 IEEE International Conference on Robotics and Automation*, vol. 2. IEEE, 2003, pp. 1620–1626.

[34] P. Fankhauser, C. D. Bellicoso, C. Gehring, R. Dubé, A. Gawel, and M. Hutter, "Free Gait - an architecture for the versatile control of legged robots," in *2016 IEEE-RAS 16th International Conference on Humanoid Robots (Humanoids)*. IEEE, 2016, pp. 1052–1058.

[35] C. Knabe, R. Griffin, J. Burton, G. Cantor-Cooke, L. Dantanarayana, G. Day, O. Ebeling-Koning, E. Hahn, M. Hopkins, J. Neal *et al.*, "Team Valor's ESCHER: A novel electromechanical biped for the darpa robotics challenge," *Journal of Field Robotics*, vol. 34, no. 5, pp. 912–939, 2017.

[36] D. Conner, S. Kohlbrecher, P. Schillinger, A. Romay, A. Stumpf, S. Maniatopoulos, H. Kress-Gazit, and O. von Stryk, *Collaborative Autonomy Between High-Level Behaviors and Human Operators for Control of Complex Tasks with Different Humanoid Robot*, ser. Springer Tracts in Advanced Robotics. Springer, 2018, vol. 121, pp. 429–494.

[37] I. Kumagai, M. Morisawa, S. Nakaoka, T. Sakaguchi, H. Kaminaga, K. Kaneko, and F. Kanehiro, "Perception based locomotion system for a humanoid robot with adaptive footstep compensation under task constraints," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2018, pp. 713–719.

[38] M. F. Fallon, P. Marion, R. Deits, T. Whelan, M. Antone, J. McDonald, and R. Tedrake, “Continuous humanoid locomotion over uneven terrain using stereo fusion,” in *IEEE-RAS International Conference on Humanoid Robots (Humanoids)*, 2015, pp. 881–888.

[39] J. Carpentier and P.-B. Wieber, “Recent progress in legged robots locomotion control,” *Current Robotics Reports*, pp. 1–8, 2021.

[40] A. Stumpf, “An integrated concept for footstep planning and navigation for different types of multi-legged robots in challenging environments,” Ph.D. dissertation, Technical University Darmstadt, Department of Computer Science, January 29 2020. [Online]. Available: https://doi.org/10.26083/tuprints-00014549

[41] P. Fankhauser and M. Hutter, “A universal grid map library: Implementation and use case for rough terrain navigation,” in *Robot Operating System (ROS)*. Springer, 2016, pp. 99–120.

[42] A. Stumpf, S. Kohlbrecher, D. C. Conner, and O. von Stryk, “Supervised footstep planning for humanoid robots in rough terrain tasks using a black box walking controller,” in *2014 IEEE-RAS 14th International Conference on Humanoid Robots (Humanoids)*, Madrid, Spain, Nov 2014, pp. 287–294.

[43] D. Kanoulas, A. Stumpf, V. Raghavan, C. Zhou, A. Toumpa, O. von Stryk, D. Caldwell, and N. Tsagarakis, “Footstep planning in rough terrain for bipedal robots using curved contact patches,” in *Proc. IEEE Int. Conf. on Robotics and Automation (ICRA)*, Brisbane, Australia, May 2018.

[44] N. Kita, M. Morisawa, and F. Kanehiro, “Foot landing state estimation from point cloud at a landing place,” in *2013 13th IEEE-RAS International Conference on Humanoid Robots (Humanoids)*. IEEE, 2013, pp. 252–259.

[45] C. B. Barber, D. P. Dobkin, D. P. Dobkin, and H. Huhdanpaa, “The quickhull algorithm for convex hulls,” *ACM Transactions on Mathematical Software (TOMS)*, vol. 22, no. 4, pp. 469–483, 1996.