2024 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)
October 14-18, 2024. Abu Dhabi, UAE

The Design of the Barkour Benchmark for Robot Agility

Wenhao Yu*, Ken Caluwaerts*, Atil Iscen*, J. Chase Kew*, Tingnan Zhang*, Daniel Freeman†, Lisa Lee†, Stefano Saliceti†, Vincent Zhuang†, Nathan Batchelor, Steven Bohez, Federico Casarini, Jose Enrique Chen, Erwin Coumans, Adil Dostmohamed, Gabriel Dulac-Arnold, Alejandro Escontrela, Erik Frey, Roland Hafner, Deepali Jain, Bauyrjan Jyenis, Yuheng Kuang, Edward Lee, Ofir Nachum, Ken Oslund, Francesco Romano, Fereshteh Sadeghi, Baruch Tabanpour, Daniel Zheng, Michael Neunert, Raia Hadsell, Nicolas Heess, Francesco Nori, Jeff Seto, Carolina Parada, Vikas Sindhwani, Vincent Vanhoucke, Jie Tan, Kuang-Huei Lee

Abstract— **In this paper, we describe the design of the Barkour benchmark for measuring robot agility in navigating complex environments. Despite the growing interest in developing agile robot locomotion skills, the field lacks systematic benchmarks to measure the performance of robotic control systems and hardware in agility-focused tasks. This motivated us to propose the Barkour benchmark, an obstacle course designed to quantify agility across various robotic platforms. Inspired by dog agility competitions, the course features diverse obstacles and a time-based scoring mechanism, encouraging researchers to develop controllers that enable robots to move quickly, precisely, and with adaptability. This benchmark is challenging as it demands diverse motion skills and the time-based scoring requires control precision at high speed. Along with the design details presented in the paper, we release our simulated environment setups in MuJoCo-XLA and the CAD model of a custom-designed quadruped robot to facilitate future research to reproduce the Barkour setup (available at sites.google.com/view/barkour). We hope these together will accelerate the pace of robot agility research.**

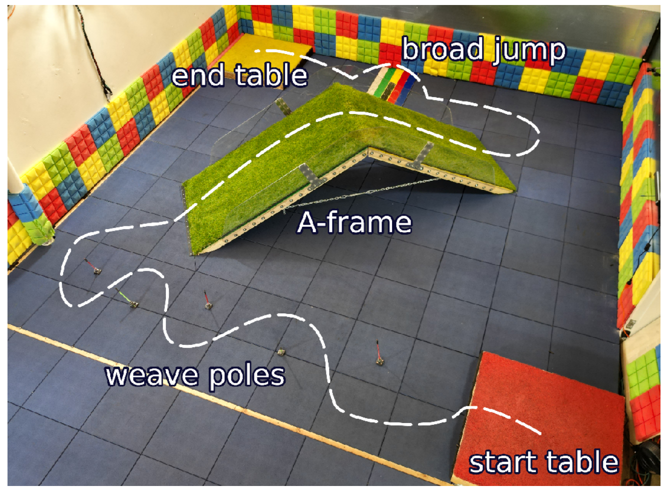

Fig. 1: The Barkour physical setup (5 m x 5 m course).

I. Introduction

Robot development has seen significant advancement inspired by animal mobility. Recent notable examples include the ETH ANYmal [1], the MIT Mini Cheetah [2], the KAIST RaiBo [3], Unitree A1/Go1 [4], and the Boston Dynamics Spot robots [5]. An important research question in this field is how to develop a controller that enables legged robots to exhibit animal-level agility while also being able to generalize across various obstacles and terrains. Through the exploration of both learning and traditional control-based methods, there has been significant progress in enabling robots to walk across a wide range of terrains [6], [7], [8], [9], [10]. These robots are now capable of walking in a variety of indoor and outdoor environments, such as up and down stairs, through bushes, and over unpaved roads and rocky, or even sandy beaches.

Despite advances in robot hardware and control, a major challenge in the field is the lack of standardized and intuitive methods for evaluating the effectiveness of locomotion controllers. Ad-hoc metrics are often used to present results, which complicates the comparing of results. To address this issue, it is essential to establish metrics that can accurately measure robot agility and to define a standard set of tasks that can serve as a common evaluation framework, similar to how the DeepMind Control Suite [11] has been widely adopted in the field of reinforcement learning (RL).

*These authors contributed equally. †These authors contributed equally. All authors are affiliated with Google DeepMind.

A good benchmark set for agile legged locomotion should (1) be non-trivial or not easily exploitable, and (2) require a diverse set of primitive behaviors that quadrupeds showcase in real environments. Well-established benchmarks already exist to measure animal performance, for example, dog agility competitions [12]. In these competitions, participants race their dogs through a preset obstacle course. A variety of obstacles including weave poles, jumps, tunnels, A-frames, seesaws, and pause tables test diverse locomotion skills. The performance is evaluated based on time and there are penalties for errors such as completing obstacles in the wrong order, tackling an obstacle from the wrong direction, or touching the jump bars while leaping.

Inspired by dog agility competitions, we introduce Barkour (Fig. 1), a challenging parkour course designed specifically for quadruped robots. We propose Barkour as a comprehensive benchmark suite for evaluating the agility of legged robots. We select a few representative obstacles (the weave poles, an A-frame, a jump board, and starting/ending pause tables) and fit the setup in a $25\,\mathrm{m}^2$ area. We design an agility scoring system inspired by the dog competition rules: To get a high score, the legged robot must complete the entire Barkour course within a time limit determined by the configuration of the obstacles. The faster the robot, the higher the final score.

The benchmark is effective for several reasons. Firstly, diverse motion capabilities are required to complete the course, which effectively exposes potential limitations on agile skill discovery. Secondly, due to the complex routes required to finish the course and the score being tied to completion time, the benchmark effectively tests the maneuverability and control precision of the locomotion controller at high speeds. If the robot misses the correct gate in the weave poles section, touches the jump board, or is simply too slow, the overall score is penalized. Lastly, we found that the benchmark is well-suited for real animal counterparts, as demonstrated in our experiments involving two small dogs, a Pomeranian/Chihuahua and a Dachshund, in Section IV. The comparison between the performance of the real dogs and the robots highlights opportunities for improvement in both hardware (such as the need for flexible spines) and algorithmic approaches.

To make progress towards the tasks in our Barkour benchmark suite, we introduce a simulation setup in MuJoCo-XLA, which includes the environment and a custom-built quadruped robot (Figure 4). As demonstrated in Caluwaerts et al. [13], locomotion policies learned in this simulation setup can be directly transferred onto a real-world Barkour course in zero-shot to enable agile locomotion. We have released the simulated setup in both MuJoCo [11] and Brax [14], as well as the CAD models of the robot at sites.google.com/view/barkour (also see Section V for details).

II. Related Work

Benchmarks are a driving force behind the development of artificial intelligence methods, such as ImageNet [15] for computer vision and the OpenAI Gym [16] for Reinforcement Learning. In the field of robot agility, while many prior works focused on creating new algorithms and controllers, limited effort has been directed towards creating a systematic benchmark to assess the performance of these controllers, especially in the context of agility [17], [18], [19], [20]. Among these efforts, Eckert & Ijspeert [18] proposed a suite of 13 metrics to measure different aspects of the agility of legged robots including leaping, standing, balancing, and climbing. Their metrics are carefully designed and measure robot performance in a comprehensive way. Barkour is complementary to these metrics, a key difference being the inclusion of a standardized and extensible obstacle course environment and a single metric to measure the overall performance of the controller in an intuitive way.

The W-Prize [19] was proposed in 2007 to demonstrate efficiency and dexterity in machine locomotion. The goal was to have a robot walk $10\,\mathrm{km}$ under $10\,000\,\mathrm{s}$ while using no more than $10\,\mathrm{kJ}$ per kg of machine mass. The obstacle course consisted of stairs, stepping blocks and missing blocks. While the overall goal is very similar, our course and metrics, inspired by dog agility competitions, focuses on faster and more dynamic motions such as jumping, without the energy restrictions. Recently, there has been an uptick in robot competitions related to agility and tackling of diverse terrains and obstacles, such as the Quadruped Robot Challenge [21] and Cybathlon [20].

There has been an extensive set of research works that push the boundaries of agile, robust, and dynamic motions [10], [9], [22], [23], [24], [25], [26], [27], [28], [29], [30], [31], [32], [33], [34], [35]. Many of these works rely on adhoc evaluation metrics, while Caluwaerts et al. [13] introduced experiments on a systematic *Barkour* benchmark. In this paper, we deep-dive into the design of Barkour and release the assets to facilitate the collective endeavor from the research community.

III. The Barkour Benchmark

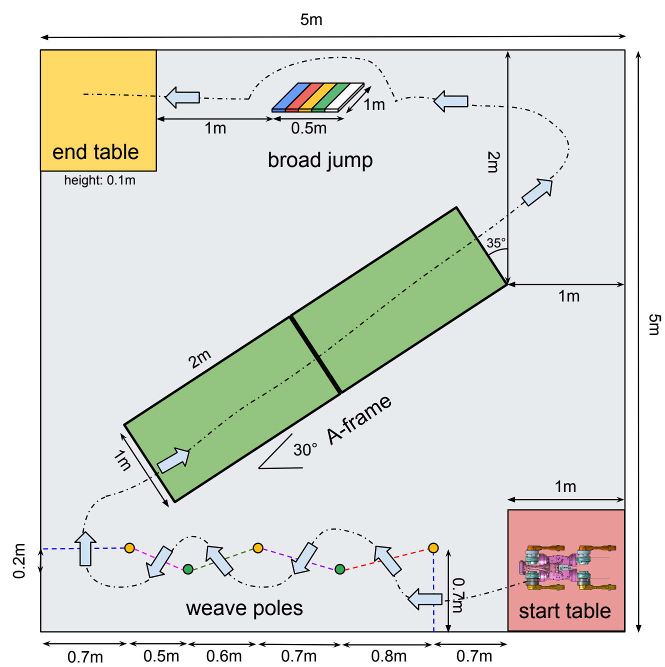

Fig. 2: Barkour course design composed of four different obstacles: (start and end) pause tables, weave poles, an A-frame, and a broad jump.

Inspired by dog agility competitions, the Barkour benchmark is designed to provide a setting to quantitatively evaluate robot agility. Dog agility competitions feature diverse sets of obstacles that require a wide range of agile locomotion skills and behaviors. They have well-defined rules and clear metrics based on timing and behaviors to compare the performances of different dogs. It is natural to think that all of these can apply to benchmark robot agility as well. Hence, we propose a similar design, including a set of obstacles that require diverse skills, a time-based metric, and penalties for rule violations.

A. The Barkour Course and Obstacles

We use an area of $5\,\mathrm{m} \times 5\,\mathrm{m}$ within which we place four unique obstacles (Figs. 1 and 2). This is a denser setup with a smaller footprint than a typical dog competition to enable repeated and controlled deployment in a standard robotics lab. We chose the obstacle types that provide diversity over required primitive skills such as running, sideways movement, climbing and jumping, while keeping the smaller footprint. Beginning at a pause table (the start table), the robot must weave through a set of poles, traverse an A-frame, jump across a broad jump, and finally make its way onto another pause table (the end table). All obstacles are fixed in place. The main floor area of the Barkour course consists of *PlaySafer Classic Tile Blue 20”x20”x2.75”* from RubberMulch.com. In the following, we describe the details of each obstacle, including physical setup and success criteria.

1) Pause Tables: A pause table is a $1\,\mathrm{m}$ x $1\,\mathrm{m}$ x $0.1\,\mathrm{m}$ (width, depth, height) solid obstacle. The top consists of a hard, non-slip surface ($12.7\,\mathrm{mm}$-$15.8\,\mathrm{mm}$ rubber granules and chunks from RubberMulch.com attached with epoxy to plywood). There is one pause table at the start (start table) of the course and one at the end (end table).

To successfully step off a table, the robot must move the center of its torso $0.7\,\mathrm{m}$ away from the center of the table. To step onto the a table, the robot must move the center of its torso to within $0.4\,\mathrm{m}$ of the center of the table.

2) Weave Poles: The weave poles obstacle consists of 5 small flexible poles spaced $0.5\,\mathrm{m}$ and $0.8\,\mathrm{m}$ apart. The unequal distances between poles increase the difficulty towards the end of the obstacle and predispose the robot to make sideways motions. The fixed part of each cone is approximately $0.05\,\mathrm{m}$ in diameter and $0.05\,\mathrm{m}$ high.

3) A-frame: The A-frame consists of a ramp with a $30°$ incline. The top of the obstacle is $1\,\mathrm{m}$ high and each side is $1\,\mathrm{m}$ wide. The gap between the floor and the flat part of the ascending and descending ramps shall be less than $20\,\mathrm{mm}$.

To successfully complete the A-frame obstacle, the robot must pass (center of torso) across the line segment defined by the part of the A-frame touching floor on the first side, move across the top of the A-frame, and finally pass across the line segment at the bottom of the opposite end.

The A-frame is challenging as the relatively small feet of the robot do not provide enough friction for semi-static behaviors. The lack of friction also penalizes jittery motions that lose contact with the ground. Moreover, the sensitivity to friction and restitution of the surface and the feet deformations also make the problem harder to model perfectly in simulation. The A-frame requires the robot to approach with some speed and maintain good contact to generate sufficient friction, while keeping forward momentum. The robot also has to keep its balance and transition to a soft landing during the downhill segment.

On the surface of the A-frame, we put artificial turf on top of a 9mm rubber (gym) mat and nailed these to a plywood frame. The rubber mat prevents the artificial grass from slipping and dampens the sound. Initially, we used the same surface as for the pause tables, but subsequently found that it is harsh on the robot’s feet and can cause loud noise when the robot moves at higher speeds.

4) Broad Jump: The broad jump is $0.5\,\mathrm{m}$ long and $1\,\mathrm{m}$ wide and sits flush with the floor. The robot has to clear it without touching. Jumping across a gap significantly exceeding the torso length is difficult for small quadruped robots to achieve from a static pose due to power/torque limits. To implement the broad jump in a repeatable and robust way, we embedded 4 motion capture markers in a plywood panel mounted on compression springs and flag a jump as failed in case any motion is detected.

B. Barkour Score Calculation

We define an *agility score* R_{agility} to measure how fast a robot can successfully complete all obstacles in the benchmark. The scoring is simplified from what is implemented in real dog competitions[1]. A score of 1.0 indicates that the robot solved the entire course within the *allotted course time* t_{allotted}. Starting from 1.0, the robot can receive two types of penalties: 1) a 0.1 penalty for each failed or skipped obstacle (N_{fail}) and 2) a 0.01 penalty for each full second the robot exceeds t_{allotted}. An episode is completed when the robot reaches the end table, otherwise it is terminated when R_{agility} reaches 0. The score[2] calculation is given by:

$$R_{\text{agility}} = 1.0 - 0.1 * N_{fail} - 0.01 * \max(t_{\text{run}} - t_{\text{allotted}}, 0) \quad (1)$$

t_{run} is the time to complete, rounded to the nearest integer. The allotted course time t_{allotted} is the target amount of time to finish the entire course. To compute this, we first measure the nominal sizes d_{obstacle} of different obstacles and sum them to obtain the full course length (Table I). We then calculate the target finish time t_{allotted} by dividing it with the target average speed. The nominal size is an estimate of the length of typical trajectory to complete a given obstacle, including the distance leading up to and away from it. Based on the rule book for real dog competitions, we set the target average speed to $v_{\text{target}} = 1.69\,\mathrm{m\,s^{-1}}$. The allotted time is calculated by:

$$t_{\text{allotted}} = \frac{\sum_{\text{obstacles}} d_{\text{obstacle}}}{v_{\text{target}}} \quad (2)$$

Table I lists the nominal sizes and allotted times of the four types of obstacles in Barkour.

Obstacle	d_{obstacle}	t_{allotted}
pause table (x2)	$1\,\mathrm{m}$	$0.59\,\mathrm{s}$
weave poles	$6\,\mathrm{m}$	$3.55\,\mathrm{s}$
A-frame	$6\,\mathrm{m}$	$3.55\,\mathrm{s}$
broad jump	$4\,\mathrm{m}$	$2.36\,\mathrm{s}$
full course	$18\,\mathrm{m}$	$10.64\,\mathrm{s}$

TABLE I: Nominal obstacle sizes for score calculation.

Note that real dog competitions often include other types of penalties (e.g., smaller deductions if a dog retries an obstacle), whereas Barkour only deducts points for failed/skipped obstacles or excess time. This makes implementing the scoring mechanism in both simulation and real easier to implement and significantly less error prone.

We also note that R_{agility} should be evaluated over multiple runs. This is similar to real dog competitions for which a dog has to achieve a qualifying score multiple times (typically $85/100$) before advancing to the next higher division.

[1]Regulations for Agility Trials and Agility Course Test (ACT) rules/scoring from the American Kennel Club (AKC). See standard course time for 8-inch Division Novice A and B Agility Standard Class.

[2]Typically represented as $[0 - 100]$ points in real dog competitions.

To understand how difficult the setting we design is to real dogs, we invited two real dogs that have similar sizes to the robot we use to test Barkour. They never failed at any of the obstacles and achieved top scores (Table III).

C. Adapting Barkour Scoring Based on Robot and Course

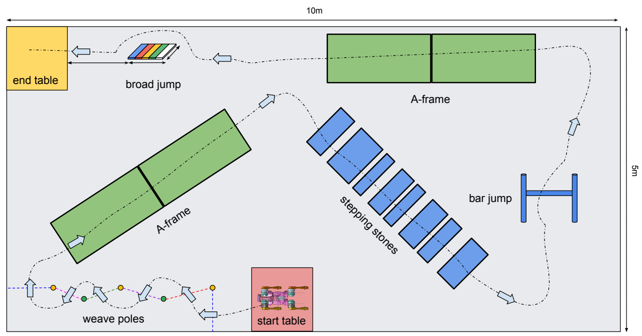

Fig. 3: Alternative obstacle course example to illustrate the flexibility of the Barkour scoring mechanism. This sample course contains two additional obstacle type (a sequence of stepping stones and a bar jump) and changes the order of the obstacles. Equation (1) can be used to compute R_{agility} based on an updated t_{allotted}.

In the base Barkour setting we have, we focused on a fixed obstacle configuration and the "8-inch Division Novice A and B Agility Standard Class" from the American Kennel Club's agility rule book in order to compute a Barkour score. However, the scoring mechanism can be customized for new obstacle sequences or types. Similarly, one can consider different agility classes or divisions based on the size or capabilities of the robots attempting a course.

Obstacle	d_{obstacle}	t_{allotted}
pause table (x2)	1 m	0.59 s
weave poles	6 m	3.55 s
A-frame (x2)	6 m	3.55 s
broad jump	4 m	2.36 s
bar jump	4 m	2.36 s
stepping stones	6 m	3.55 s
full alternative course	34 m	20.12 s

TABLE II: Nominal obstacle sizes for score calculation for alternative course from Figure 3. t_{allotted} shows the obstacle time for the "8-inch Division Novice A and B Agility Standard Class" as used throughout the main text. To compute R_{agility}, use Equation (1).

For designing novel agility courses and adapt our base Barkour setting to them, we propose the following practical guidelines based off of our experience:

- Take inspiration from dog agility competitions, but keep new obstacle designs and evaluation simple. Real dog agility competitions often specify in great detail how a specific obstacle should be completed. Such detailed instructions are difficult to evaluate consistently and robustly in a robotics setting and add little practical value. Example: The pause table obstacle requires dogs to touch the top of the table with all 4 paws, which is hard to evaluate on a real robot. Instead, we simply track the position of the robot's torso. This leaves room for a number of edge cases, but policies that depend on these tend to fail under repeated evaluation.
- A simple rule of thumb to define the nominal size of an obstacle is to use the actual length of the obstacle plus $1\,\mathrm{m}$, rounded up to the nearest integer.
- Courses with more obstacles are harder in general, because the penalty for a failed or skipped obstacle is constant (0.1). We suggest first successfully evaluating the original Barkour environment as a baseline over multiple trials prior to expanding to a larger number of obstacles (this can be done with a setup like Figure 3). This is similar to the approach taken for real dog agility competitions in which a dog has to achieve a qualifying score (typically 85/100) at multiple competitions before being allowed to enter a higher division (e.g. more obstacles).
- The American Kennel Club rule book states higher target speeds for taller dogs in agility competitions. However, biological and mechanical scaling laws are complex and a robot's height does not accurately reflect its moving speed. Instead, we suggest keeping the target speed constant (8-inch novice division, $1.69\,\mathrm{m\,s^{-1}}$ as in the main text) until more data is available on how other quadruped designs perform. We also encourage larger and more diverse course designs that further the generalization capabilities and robustness of policies.

Figure 3 shows an example of an possible alternative course. This course adds two new types of obstacles (bar jump and stepping stones), modifies the order of the obstacles and repeats the A-frame. In Table II, we show how we would compute the Barkour agility score R_{agility} for this course.

IV. Current Progress in Tackling Barkour

In this section we summarize and discuss the current process that we made on tackling the Barkour benchmark with a quadruped robot in Caluwaerts et al. [13]. We first present a setting where specialist policies are trained for each obstacle and sequenced together for the entire course. We then discuss a method that trains a single policy for all obstacles. We focus on presenting the results from these methods to show the progress that has been made towards Barkour. More details can be found in the original tech report [13].

A. Robot Hardware

Effectively exploring the Barkour benchmark necessitates considerable controller development and thorough real hardware experimentation. This poses significant challenges on the reliability and repeatability of the robot hardware, especially given the highly agile movements we strive for. Moreover, quadruped animals exhibit diverse body configurations compared to typical quadruped robots, which can significantly affect the robot's capacity for agile motion. Consequently, we believe that hardware optimization and customization are critical in bridging the agility gap between legged robots and their animal counterparts.

As a result, in Caluwaerts et al. [13] we developed a small quadruped robot in-house to evaluate its learned agile skills using the Barkour benchmark. The robot (Figure 4) is similar in size to the Unitree A1 [4] and MIT Mini-Cheetah [2] robots. The robot weighs $11.5\,\mathrm{kg}$, and has $220\,\mathrm{mm}$ upper limbs and $190\,\mathrm{mm}$ lower limbs. The front-to-back hip-to-hip distance is $380\,\mathrm{mm}$, and the distance between the left and right legs is $290\,\mathrm{mm}$. The robot uses T-Motor AK80-6 actuators, which are controlled by Elmo G-SOLTWIR50/100SE2S motor drivers running at $24\,\mathrm{V}$. The actuators provide a peak output torque of $12\,\mathrm{N\,m}$ at each joint. As mentioned, we release both the CAD models and simulated implementations of this robot (Section V).

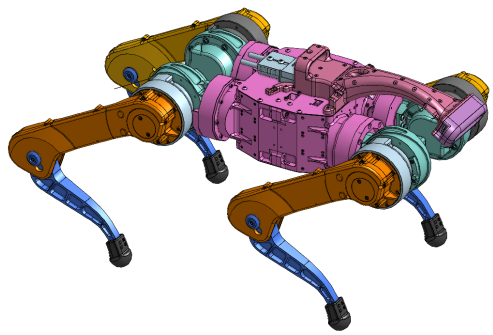

Fig. 4: Custom small quadruped robot for hardware evaluation.

Caluwaerts et al. [13] used both the robot's onboard sensors (Parker 3DMCX5-AHRS IMU, joint position, velocity, torque) as well as an external motion capture system (Phasespace X2E) to track the robot's position and orientation in the obstacle course.

B. Combined Specialist Policies

In the specialist setting, Caluwaerts et al. [13] trained three specialist policies to cover all the core agility skills required in the Barkour benchmark: fast omnidirectional walking on uneven terrain, climbing up and down a slope, and jumping over a board. The first policy was used to solve the start/end tables and weave pole tasks, the second solves the A-Frame task, and the third solves the Broad Jump task. All the specialist policies were trained using PPO [36].

Figure 5 and Figure 6 shows a sample episode using all these specialist policies combined together to tackle the Barkour course. The robot completes the course by traversing all of the obstacles in $20.4\,\mathrm{s}$. Although the robot completes all the obstacles, this specific run only receives 0.91 agility score due to the extra time it took beyond the allotted time of $10.64\,\mathrm{s}$. One can see that the robot completes the weave poles section at a $90°$ angle, which is enforced by our high level controller, due to the narrow passages between the last few poles. The velocity of the robot during the same run is shown

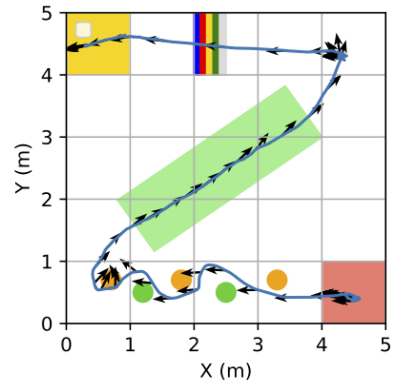

Fig. 5: Top-down view of the robot's trajectory during a single high performance evaluation where the robot completes the course in $20.4\,\mathrm{s}$ scoring 0.91. The arrows show the robot's heading at $0.5\,\mathrm{s}$ intervals.

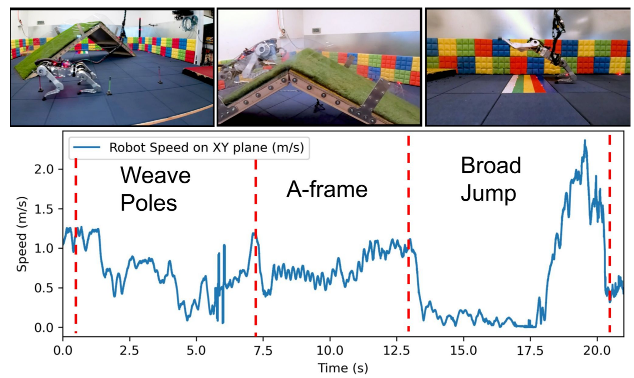

Fig. 6: Top: Robot behaviors overcoming three different obstacles as in Figure 5. Bottom: The speed of the robot during the same trial.

on the right side in Figure 6 (Bottom). The robot's forward velocity varies between $0.5\,\mathrm{m\,s^{-1}}$ and $1\,\mathrm{m\,s^{-1}}$ during the weave poles. It reaches a velocity as high as $2.3\,\mathrm{m\,s^{-1}}$ during the jumping phase. For comparison, the same course was tested with two small dogs: a Pomeranian/Chihuahua ($3.2\,\mathrm{kg}$, $33\,\mathrm{cm}$ height at withers) and a Dachshund ($4\,\mathrm{kg}$, $20\,\mathrm{cm}$ height at withers). After getting familiar with the course, both dogs reached a maximum score (1.0) by reaching the final pause table in approximately $9\,\mathrm{s}$.

Policy Type	Barkour Score	Completion Time	Forward Velocity
Specialist	0.77 ± 0.064	24.6 ± 1.1 s	0.74 ± 0.04 m/s
Locomotion-Transformer	0.73 ± 0.062	25.8 ± 1.6 s	0.69 ± 0.04 m/s
Small dog	1.00 ± 0	9.02 ± 0.65 s	

TABLE III: Results for full Barkour course with specialist policies and the generalist Locomotion-Transformer. We report mean and one standard deviation for each metric.

The robustness of the combined controller was tested by running it 71 times. Figure 8 shows that the robot completes the course in most of these runs, but fell down or tipped over 6 times. In 40 of the runs, the robot touches the jump board

	Weave Poles	A-Frame	Broad Jump
Completion Time	$9.27 \pm 0.87\mathrm{s}$	$7.95 \pm 0.73\mathrm{s}$	$2.45 \pm 0.54\mathrm{s}$
Nominal Size	$6\,\mathrm{m}$	$6\,\mathrm{m}$	$4\,\mathrm{m}$
Forward Velocity	$0.73 \pm 0.06\,\mathrm{m\,s^{-1}}$	$0.68 \pm 0.06\,\mathrm{m\,s^{-1}}$	1.7 ± 0.24
Success Rate	100%	100%	38%

TABLE IV: Results for specialist policies with individual obstacles. We report mean and one std dev for each metric.

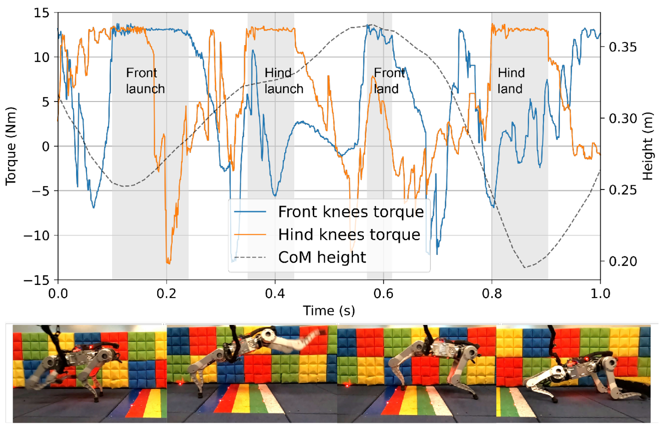

Fig. 7: Broad jump example. Top: Average front and hind knee torque during various phases of the jump. Bottom: Main jump phases: accelerating, lift-off, landing the front legs, landing the rear legs.

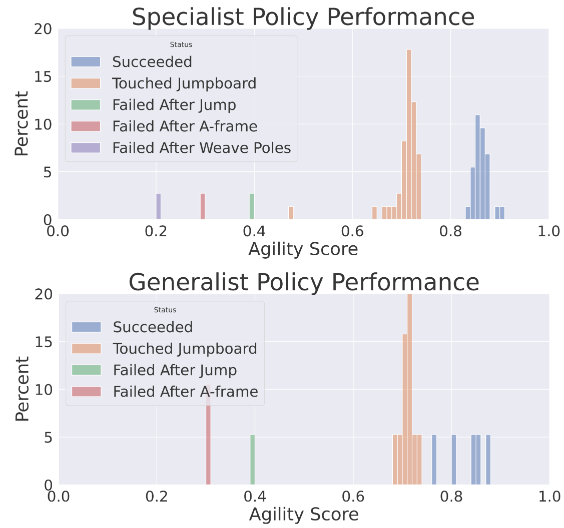

Fig. 8: Statistical results for Barkour episodes with specialist policies. Top: Distribution of the scores for 71 trials using specialist policies. Bottom: Distribution of the scores for 19 trials using generalist Locomotion-Transformer architecture.

and still scores approximately 0.73. In 25 of these runs, the robot completes all 5 obstacles scoring 0.87 on average. As shown in Table III, the mean score of all the runs is around 0.77 with an average base forward velocity of $0.74\,\mathrm{m\,s^{-1}}$.

The decomposition of these runs (Table IV) into the different obstacles shows that the specialist policies complete the weave poles at around the nominal (target) speed, while the broad jump is faster and the A-frame is slower. The robot completes the weave poles and A-frame with a 100% success rate. The broad jump, which goes up to $70\,\mathrm{cm}$ in height and requires more agile behavior, succeeds only 38% of the time.

C. Generalist Policy using Locomotion-Transformer

In the generalist setting, Caluwaerts et al. [13] trained a single generalist locomotion policy that can tackle all obstacles, thereby removing the necessity for ad hoc switching between individual specialist policies and promoting generalization capabilities to different obstacle and terrain configurations. The generalist policy, *Locomotion-Transformer*, was trained by distilling the specialist policies via behavioral cloning with a Transformer sequence model [37], similar to [38], [39], [40]. Although prior work [41], [42] has demonstrated that generalist policies can also be learned via reinforcement learning by simultaneously training on multiple environments, the diverse curricula and reward structures in our problem setting makes it challenging in our experience.

The Locomotion-Transformer policy was evaluated on the real Barkour setup with 19 trials, which can be seen in Figure 8. In general, the Locomotion-Transformer policy exhibits slightly lower performance compared to the combined specialist policies (also seen in Table III). This is understandable because the Locomotion-Transformer policy does not have information about the obstacle type being tackled and has to infer relevant information from terrain and proprioceptive observations, which increases the difficulties of the task.

On important property of training a generalist policy is its generalization capability to novel situations. To demonstrate this, Caluwaerts et al. evaluated the policies in the following two novel scenarios.

In the first scenario, the order of the Barkour obstacles was adjusted to create a new course: the robot must perform fast 90° turning, then go to the end of the A-Frame and perform the A-Frame task backwards. Next, the robot must do a broad jump towards the start table and climb up the table. With specialist policies, one would need to reassign different policies to different sub-goals and fine-tune the transitions between different policies; the generalist Locomotion-Transformer policy can generalize to this new route by simply providing a set of new sub-goals to denote the desired route. The results in Figure 9 (left) show that the robot succeeds in climbing the A-frame from the opposite end and going over the broad jump in a different location.

In the second scenario, the A-Frame was modified by adding random steps to the slope, creating a terrain that mixes slope and steps as shown in Figure 9 (right). This requires the policy to combine the skills for handling individual terrains. Both the specialist and the generalist Locomotion-Transformer policies were evaluated in this new environment in simulation. Locomotion-Transformer was able to significantly outperform the specialist policy and achieve an average score of 0.66 ± 0.091 over 10 runs, in contrast to 0.34 ± 0.211 for the specialist policy.

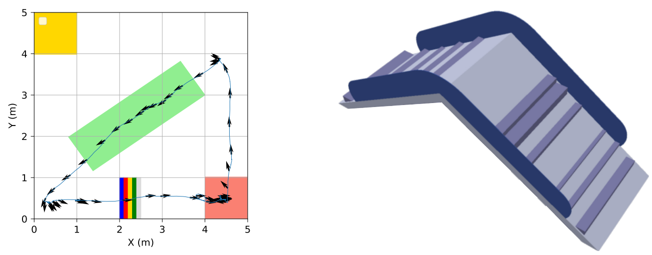

Fig. 9: Generalization tests of the generalist Locomotion-Transformer policy. (Left) a modified course where the layout and order of obstacle locations are changed. While the policy is not aware of the obstacle types, the robot can still climb up the A-frame from the opposite end or achieve the broad jump in different locations. (Right) a modified A-Frame task that blends both slope and steps, requiring the policy to handle both terrain types simultaneously. The Locomotion-Transformer is able to successfully handle this task by distilling knowledge from both skills and significantly outperforms the specialist policy.

V. Open Source Assets

We open-source the key assets of the Barkour benchmark at **sites.google.com/view/barkour** and our GitHub repo, **github.com/google-deepmind/barkour_robot**, to enable future research efforts to build upon Barkour:

- **Barkour course MuJoCo model:** We release a simulation model of the main Barkour course in the MuJoCo Menagerie [43]. MuJoCo Menagerie is a collection of high-quality models for the MuJoCo physics engine. We also release a standalone script to compute the Barkour score given a complete trajectory of the robot CoM and orientation. Researchers can use the script to compute the score from robot trajectories and the simulation environment to visualize the result or perform training. MuJoCo Menagerie also provides access to a variety of legged robots that can facilitate explorations of Barkour benchmark with different robots, including the one used in Caluwaerts et al. [13].
- **MuJoCo-XLA training example:** To provide a starting point for training locomotion policies on the Barkour benchmark, we release an MuJoCo-XLA training environment, a revised version of the IsaacGym-based training environment used in Caluwaerts et al. [13]. With the MuJoCo-XLA environment, training can achieve a high-quality omni-direction walking policy on the real robot within 6 minutes of training on an A100 GPU.
- **Robot CAD model (OnShape):** In addition to the simulation version of the robot in the MuJoCo Menagerie, we also open-source the detailed design of the robot with OnShape CAD model to enable reproducing the robot hardware. The structural parts of the robot were machined out of 6061-T6 aluminum. The feet were 3D printed using Loctite 3D IND402 material.

VI. Conclusions

We presented Barkour, a benchmark to evaluate the agility of quadruped robots. Inspired by dog agility competitions, Barkour is a test-bed with an intuitive scoring mechanism that requires the combination of various agile skills. Furthermore, it can be easily adapted to robots of various sizes or extended by adding or rearranging obstacles and still retain the same metrics.

We believe that providing a benchmark for legged robotics, especially in regards to agility, is an important first step to quantify progress towards animal-level agility for quadruped robots. The Barkour benchmark is far from solved. While our baseline solutions can reach a peak agility score of 0.91 and complete the course in approx. 20 s, an untrained dog achieved a 1.0 agility score and completed the course in half the robot's time (approx. 9 s). There is still a big gap in agility between robots and their animal counterparts, as demonstrated in this benchmark. Additionally, the fact that state-of-the-art RL methods fail to learn a single policy to complete the course further underscores the complexity and value of Barkour as a benchmark.

It is worth noting that the current baselines use privileged information such as the CAD model of the environment and the position of the robot in the world frame. An important future work direction is to explore Barkour using only on-board sensors for both low-level locomotion skills and high-level navigation controller. An equally exciting direction for future research on Barkour is to evaluate the impact of modifications to robot hardware, different form-factors, and sensors on performance or training speed.

We believe that Barkour will serve the robotics community as an important test-bed for different learning and control methods and different hardware designs.

Acknowledgments

We'd like to thank Marissa Giustina, Gus Kouretas, nubby Lee, James Lubin, Sherry Moore, Thinh Nguyen, Krista Reymann, Satoshi Kataoka, Trish Blazina, Omar Cortes, Linda Luu, Jason Powell, Diego Reyes, Ron Sloat, Daniel Freeman, Erwin Coumans, Ofir Nachum, Alejandro Escontrela, Fereshteh Sadeghi, and the Google DeepMind Robotics team.

References

[1] 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 *2016 IEEE/RSJ international conference on intelligent robots and systems (IROS)*. IEEE, 2016, pp. 38–44.

[2] B. Katz, J. Di Carlo, and S. Kim, "Mini cheetah: A platform for pushing the limits of dynamic quadruped control," in *2019 international conference on robotics and automation (ICRA)*. IEEE, 2019, pp. 6295–6301.

[3] S. Choi, G. Ji, J. Park, H. Kim, J. Mun, J. H. Lee, and J. Hwangbo, "Learning quadrupedal locomotion on deformable terrain," *Science Robotics*, vol. 8, no. 74, p. eade2256, 2023.

[4] "Unitree Go1." [Online]. Available: https://shop.unitree.com/products/unitreeyushutechnologydog-artificial-intelligence-companion-bionic-\companion-intelligent-robot-go1-quadruped-robot-dog

[5] "Spot® - The Agile Mobile Robot." [Online]. Available: https://www.bostondynamics.com/products/spot

[6] J. Di Carlo, P. M. Wensing, B. Katz, G. Bledt, and S. Kim, "Dynamic locomotion in the mit cheetah 3 through convex model-predictive control," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 1–9.

[7] J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning quadrupedal locomotion over challenging terrain," *Science robotics*, vol. 5, no. 47, p. eabc5986, 2020.

[8] A. Kumar, Z. Fu, D. Pathak, and J. Malik, "Rma: Rapid motor adaptation for legged robots," *arXiv preprint arXiv:2107.04034*, 2021.

[9] A. Agarwal, A. Kumar, J. Malik, and D. Pathak, "Legged locomotion in challenging terrains using egocentric vision," in *Conference on Robot Learning, CoRL 2022, 14-18 December 2022, Auckland, New Zealand*, ser. Proceedings of Machine Learning Research, K. Liu, D. Kulic, and J. Ichnowski, Eds., vol. 205. PMLR, 2022, pp. 403–415. [Online]. Available: https://proceedings.mlr.press/v205/agarwal23a.html

[10] T. Miki, J. Lee, J. Hwangbo, L. Wellhausen, V. Koltun, and M. Hutter, "Learning robust perceptive locomotion for quadrupedal robots in the wild," *Science Robotics*, vol. 7, no. 62, p. eabk2822, 2022.

[11] Y. Tassa, Y. Doron, A. Muldal, T. Erez, Y. Li, D. d. L. Casas, D. Budden, A. Abdolmaleki, J. Merel, A. Lefrancq *et al.*, "Deepmind control suite," *arXiv preprint arXiv:1801.00690*, 2018.

[12] A. K. Club, *Regulations for Agility Trials and Agility Course Test (ACT)*. American Kennel Club, 2023.

[13] K. Caluwaerts, A. Iscen, J. C. Kew, W. Yu, T. Zhang, D. Freeman, K.-H. Lee, L. Lee, S. Saliceti, V. Zhuang *et al.*, "Barkour: Benchmarking animal-level agility with quadruped robots," *arXiv preprint arXiv:2305.14654*, 2023.

[14] C. D. Freeman, E. Frey, A. Raichuk, S. Girgin, I. Mordatch, and O. Bachem, "Brax–a differentiable physics engine for large scale rigid body simulation," *arXiv preprint arXiv:2106.13281*, 2021.

[15] J. Deng, W. Dong, R. Socher, L.-J. Li, K. Li, and L. Fei-Fei, "Imagenet: A large-scale hierarchical image database," in *2009 IEEE conference on computer vision and pattern recognition*. Ieee, 2009, pp. 248–255.

[16] G. Brockman, V. Cheung, L. Pettersson, J. Schneider, J. Schulman, J. Tang, and W. Zaremba, "Openai gym," 2016.

[17] D. Torricelli, R. S. Mizanoor, J. Gonzalez, V. Lippi, G. Hettich, L. Asslaender, M. Weckx, B. Vanderborght, S. Dosen, M. Sartori *et al.*, "Benchmarking human-like posture and locomotion of humanoid robots: a preliminary scheme," in *Biomimetic and Biohybrid Systems: Third International Conference, Living Machines 2014, Milan, Italy, July 30–August 1, 2014. Proceedings 3*. Springer, 2014, pp. 320–331.

[18] P. Eckert and A. J. Ijspeert, "Benchmarking agility for multilegged terrestrial robots," *IEEE Transactions on Robotics*, vol. 35, no. 2, pp. 529–535, 2019.

[19] T. McGeer, "Dynamic walking robots and the w prize [industry/research news]," *IEEE Robotics and Automation Magazine*, vol. 14, no. 2, pp. 13–15, 2007.

[20] L. Jaeger, R. d. S. Baptista, C. Basla, P. Capsi-Morales, Y. K. Kim, S. Nakajima, C. Piazza, M. Sommerhalder, L. Tonin, G. Valle, R. Riener, and R. Sigrist, "How the cybathlon competition has advanced assistive technologies," *Annual Review of Control, Robotics, and Autonomous Systems*, vol. 6, no. 1, pp. 447–476, 2023. [Online]. Available: https://doi.org/10.1146/annurev-control-071822-095355

[21] A. Jacoff, J. Jeon, O. Huke, D. Kanoulas, S. Ha, D. Kim, and H. Moon, "Taking the first step toward autonomous quadruped robots: The quadruped robot challenge at icra 2023 in london [competitions]," *IEEE Robotics & Automation Magazine*, vol. 30, no. 3, pp. 154–158, 2023.

[22] N. Rudin, D. Hoeller, P. Reist, and M. Hutter, "Learning to walk in minutes using massively parallel deep reinforcement learning," in *Conference on Robot Learning*. PMLR, 2022, pp. 91–100.

[23] H.-W. Park, P. M. Wensing, and S. Kim, "High-speed bounding with the mit cheetah 2: Control design and experiments," *The International Journal of Robotics Research*, vol. 36, no. 2, pp. 167–192, 2017.

[24] G. Bledt, M. J. Powell, B. Katz, J. Di Carlo, P. M. Wensing, and S. Kim, "Mit cheetah 3: Design and control of a robust, dynamic quadruped robot," in *2018 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018, pp. 2245–2252.

[25] D. Kim, J. Di Carlo, B. Katz, G. Bledt, and S. Kim, "Highly dynamic quadruped locomotion via whole-body impulse control and model predictive control," *arXiv preprint arXiv:1909.06586*, 2019.

[26] G. B. Margolis, G. Yang, K. Paigwar, T. Chen, and P. Agrawal, "Rapid locomotion via reinforcement learning," *arXiv preprint arXiv:2205.02824*, 2022.

[27] M. Chignoli, S. Morozov, and S. Kim, "Rapid and reliable quadruped motion planning with omnidirectional jumping," in *2022 International Conference on Robotics and Automation (ICRA)*. IEEE, 2022, pp. 6621–6627.

[28] Q. Nguyen, M. J. Powell, B. Katz, J. Di Carlo, and S. Kim, "Optimized jumping on the mit cheetah 3 robot," in *2019 International Conference on Robotics and Automation (ICRA)*. IEEE, 2019, pp. 7448–7454.

[29] H. Li, W. Yu, T. Zhang, and P. M. Wensing, "Zero-shot retargeting of learned quadruped locomotion policies using hybrid kinodynamic model predictive control," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 11 971–11 977.

[30] A. Iscen, G. Yu, A. Escontrela, D. Jain, J. Tan, and K. Caluwaerts, "Learning agile locomotion skills with a mentor," in *2021 IEEE International Conference on Robotics and Automation (ICRA)*. IEEE, 2021, pp. 2019–2025.

[31] G. B. Margolis, T. Chen, K. Paigwar, X. Fu, D. Kim, S. Kim, and P. Agrawal, "Learning to jump from pixels," *arXiv preprint arXiv:2110.15344*, 2021.

[32] W. Yu, D. Jain, A. Escontrela, A. Iscen, P. Xu, E. Coumans, S. Ha, J. Tan, and T. Zhang, "Visual-locomotion: Learning to walk on complex terrains with vision," in *5th Annual Conference on Robot Learning*, 2021.

[33] K.-H. Lee, O. Nachum, T. Zhang, S. Guadarrama, J. Tan, and W. Yu, "PI-ARS: Accelerating Evolution-Learned Visual-Locomotion with Predictive Information Representations," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*. IEEE, 2022, pp. 1447–1454.

[34] Z. Zhuang, Z. Fu, J. Wang, C. Atkeson, S. Schwertfeger, C. Finn, and H. Zhao, "Robot parkour learning," *arXiv preprint arXiv:2309.05665*, 2023.

[35] X. Cheng, K. Shi, A. Agarwal, and D. Pathak, "Extreme parkour with legged robots," *arXiv preprint arXiv:2309.14341*, 2023.

[36] J. Schulman, F. Wolski, P. Dhariwal, A. Radford, and O. Klimov, "Proximal policy optimization algorithms," *arXiv preprint arXiv:1707.06347*, 2017.

[37] A. Vaswani, N. Shazeer, N. Parmar, J. Uszkoreit, L. Jones, A. N. Gomez, Ł. Kaiser, and I. Polosukhin, "Attention is all you need," *Advances in neural information processing systems*, vol. 30, 2017.

[38] K.-H. Lee, O. Nachum, S. Yang, L. Lee, C. D. Freeman, S. Guadarrama, I. Fischer, W. Xu, E. Jang, H. Michalewski *et al.*, "Multi-game decision transformers," in *Advances in Neural Information Processing Systems*, 2022.

[39] A. Brohan, N. Brown, J. Carbajal, Y. Chebotar, J. Dabis, C. Finn, K. Gopalakrishnan, K. Hausman, A. Herzog, J. Hsu, J. Ibarz, B. Ichter, A. Irpan, T. Jackson, S. Jesmonth, N. J. Joshi, R. Julian, D. Kalashnikov, Y. Kuang, I. Leal, K. Lee, S. Levine, Y. Lu, U. Malla, D. Manjunath, I. Mordatch, O. Nachum, C. Parada, J. Peralta, E. Perez, K. Pertsch, J. Quiambao, K. Rao, M. S. Ryoo, G. Salazar, P. R. Sanketi, K. Sayed, J. Singh, S. Sontakke, A. Stone, C. Tan, H. Tran, V. Vanhoucke, S. Vega, Q. Vuong, F. Xia, T. Xiao, P. Xu, S. Xu, T. Yu, and B. Zitkovich, "RT-1: robotics transformer for real-world control at scale," in *Robotics: Science and Systems XIX, Daegu, Republic of Korea, July 10-14, 2023*, K. E. Bekris, K. Hauser, S. L. Herbert, and J. Yu, Eds., 2023. [Online]. Available: https://doi.org/10.15607/RSS.2023.XIX.025

[40] S. Reed, K. Zolna, E. Parisotto, S. G. Colmenarejo, A. Novikov, G. Barth-Maron, M. Gimenez, Y. Sulsky, J. Kay, J. T. Springenberg *et al.*, "A generalist agent," *arXiv preprint arXiv:2205.06175*, 2022.

[41] L. Espeholt, H. Soyer, R. Munos, K. Simonyan, V. Mnih, T. Ward, Y. Doron, V. Firoiu, T. Harley, I. Dunning *et al.*, "Impala: Scalable distributed deep-rl with importance weighted actor-learner architectures," in *International conference on machine learning*. PMLR, 2018, pp. 1407–1416.

[42] N. Rudin, D. Hoeller, M. Bjelonic, and M. Hutter, "Advanced skills by learning locomotion and local navigation end-to-end," in *2022 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2022, pp. 2497–2503.

[43] K. Zakka, Y. Tassa, and MuJoCo Menagerie Contributors, "MuJoCo Menagerie: A collection of high-quality simulation models for MuJoCo," 2022. [Online]. Available: http://github.com/google-deepmind/mujoco_menagerie