tion [44, 98, 45, 48, 47, 74, 73, 19, 106], jumping [46], and parkour [50, 107]. More advanced capabilities, such as dancing [105, 32, 10], loco-manipulation [25, 53, 15, 24], and even backflipping [78], have also been demonstrated. Meanwhile, the humanoid character animation community has achieved highly expressive and agile whole-body motions in physics-based simulations [71, 86, 55], including cartwheels [67], backflips [69], sports movements [104, 90, 56, 91, 92], and smooth object interactions [86, 17, 22]. However, transferring these highly dynamic and agile skills to real-world humanoid robots remains challenging due to the dynamics mismatch between simulation and real-world physics. To address this challenge, our work focuses on learning and compensating for this dynamics mismatch, enabling humanoid robots to perform expressive and agile whole-body skills in the real world.

B. Offline and Online System Identification for Robotics

The dynamics mismatch between simulators and real-world physics can be attributed to two primary factors: inaccuracies in the robot model descriptions and the presence of complex real-world dynamics that are difficult for physics-based simulators to capture. Traditional approaches address these issues using system identification (SysID) methods [39, 5], which calibrate the robot model or simulator based on real-world performance. These methods can be broadly categorized into *offline SysID* and *online SysID*, depending on whether system identification occurs at test time. *Offline SysID* methods typically collect real-world data and adjust simulation parameters to train policies in more accurate dynamics. The calibration process may focus on modeling actuator dynamics [85, 29, 99], refining robot dynamics models [36, 2, 18, 21, 30], explicitly identifying critical simulation parameters [102, 9, 13, 96], learning a distribution over simulation parameters [75, 27, 4], or optimizing system parameters to maximize policy performance [64, 76]. *Online SysID* methods, in contrast, aim to learn a representation of the robot's state or environment properties, enabling real-time adaptation to different conditions. These representations can be learned using optimization-based approaches [101, 103, 43, 70], regression-based methods [100, 40, 89, 19, 31, 60, 14, 72, 61, 41, 62, 42], next-state reconstruction techniques [65, 51, 54, 94, 83], direct reward maximization [47], or by leveraging tracking and prediction errors for online adaptation [66, 57, 28, 16]. Our framework takes a different approach from traditional SysID methods by learning a residual action model that directly compensates for dynamics mismatch through corrective actions, rather than explicitly estimating system parameters.

C. Residual Learning for Robotics

Learning a residual component alongside learned or predefined base models has been widely used in robotics. Prior work has explored residual policy models that refine the actions of an initial controller [84, 34, 8, 1, 12, 20, 3, 33, 42]. Other approaches leverage residual components to correct inaccuracies in dynamics models [66, 35, 38, 82, 23] or to model residual trajectories resulting from residual actions [11] for achieving precise and agile motions. RGAT [35] uses a residual action model with a learned forward dynamics to refine the simulator. Our framework builds on this idea by using RL-based residual actions to align the dynamics mismatch between simulation and real-world physics, enabling agile whole-body humanoid skills.

VII. Conclusion

We present ASAP, a two-stage framework that bridges the sim-to-real gap for agile humanoid control. By learning a universal delta action model to capture dynamics mismatch, ASAP enables policies trained in simulation to adapt seamlessly to real-world physics. Extensive experiments demonstrate significant reductions in motion tracking errors (up to 52.7% in sim-to-real tasks) and successful deployment of diverse agile skills—including agile jumps and kicks—on the Unitree G1 humanoid. Our work advances the frontier of sim-to-real transfer for agile whole-body control, paving the way for versatile humanoid robots in real-world applications.

VIII. Limitations

While ASAP demonstrates promising results in bridging the sim-to-real gap for agile humanoid control, our framework has several real-world limitations that highlights critical challenges in scaling agile humanoid control to real-world:

- **Hardware Constraints**: Agile whole-body motions exert significant stress on robots, leading to motor overheating and hardware failure during data collection. Two Unitree G1 robots were broken to some extent during our experiments. This bottleneck limits the scale and diversity of real-world motion sequences that can be safely collected.
- **Dependence on Motion Capture Systems**: Our pipeline requires MoCap setup to record real-world trajectories. This introduces practical deployment barriers in unstructured environments where MoCap setups are unavailable.
- **Data-Hungry Delta Action Training**: While reducing the delta action model to 4 DoF ankle joints improved sample efficiency, training the full 23 DoF model remains impractical for real-world deployment due to the large demand of required motion clips (e.g., >400 episodes in simulation for the 23 DoF delta action training).

Future directions could focus on developing damage-aware policy to mitigate hardware risks, leveraging MoCap-free alignment to eliminate the reliance on MoCap, and exploring adaptation techniques for delta action models to achieve sample-efficient few-shot alignment.

Acknowledgments

We thank Zhenjia Xu, Yizhou Zhao, Yu Fang for help with hardware. We thank Pulkit Goyal, Hawkeye King, Peter Varvak and Haoru Xue for help with motion capture setup. We thank Ziyan Xiong, Yilin Wu and Xian Zhou for support on Genesis integration. We thank Rui Chen, Yifan Sun and Kai Yun for help with G1 hardware. We thank Xuxin Cheng, Chong Zhang and Toru Lin for always being there to help with any problem and answer any question. We thank Unitree Robotics for help with G1 support.