Supplementary Material
RINA: Rapid Introspective Neural Adaptation for Out-of-Distribution Payload Configurations on Quadruped Robots

Oscar Youngquist and Hao Zhang

I. Whole-Body Impulse Control Details

This section provides details on the Whole-Body Impulse Control (WBIC) [1] controller used as $\Phi(\cdot)$ in RINA. This controller considers the following general equations of motion:

$$\mathbf{A}\begin{bmatrix}\ddot{\mathbf{p}}\\ \ddot{\mathbf{q}}\end{bmatrix} + \mathbf{b} + \mathbf{g} = \begin{bmatrix}\mathbf{0}^6\\ \mathbf{a}^{\mathbf{ex}}\end{bmatrix} + \mathbf{J}_c^\top \mathbf{f}^r \tag{1}$$

where $\mathbf{A}$, $\mathbf{b}$, $\mathbf{g}$, $\mathbf{J}_c^\top$, and $\mathbf{f}^r$ are the generalized mass matrix, Corilois force vector, gravitation force vector, contact Jacobian, and ground reaction force vector respectively. Additionally, $\ddot{\mathbf{p}}$ and $\ddot{\mathbf{q}}$ are the accelerations of the torso and leg joints and $\mathbf{a}^{\mathbf{ex}}$ are the joint torques. Lastly, $\mathbf{0}^6$ represents a six dimensional vector of zeros, corresponding to the six not directly controlled degrees of freedom of the torso.

The WBIC controller in [1] leverages the above definition of the whole body dynamics to track an optimal ground reaction force profile along side acceleration commands. This accomplished by first using the current state of the robot and the expected torso locomotion commands $\mathbf{b}^{ex}$ (defined as expected linear and angular velocities as in the main text) to find ground reaction forces that allow the robot to follow the desired trajectory. The optimal ground reaction force profile is computed using the convex MPC (cMPC) formulation presented in [2] using a simplified lumped mass model. The ground force reaction profile generated by the cMPC, denoted as $\mathbf{f}^r_{\text{MPC}}$, are then used to compute expected joint position $\mathbf{q}^{ex}$, velocity $\dot{\mathbf{q}}^{ex}$, acceleration $\ddot{\mathbf{q}}^{ex}$ and torque $\mathbf{a}^{ex}$ commands.

The WBIC computes the expected $\ddot{\mathbf{q}}^{ex}$, $\dot{\mathbf{q}}^{ex}$, and $\mathbf{q}^{ex}$ using an hierarchical inverse kinematics algorithm that uses null space projection to enforce a strict task priority over three control objectives: torso orientation, torso position, and foot position, in that order. The quadratic program in Eq. (2) is then used to refine $\mathbf{f}^r_{\text{MPC}}$ and $\ddot{\mathbf{q}}^{ex}$ using the whole body dynamics in Eq. (1).

$$\begin{aligned}
\min_{\delta_{\mathbf{f}^r},\delta_f} \quad & \delta_{\mathbf{f}^r}^\top Q \delta_{\mathbf{f}^r}^\top + \delta_f^\top Q \delta_f^\top \\
s.t. \quad & \\
& \mathbf{S}^f\left(\mathbf{A}\begin{bmatrix}\ddot{\mathbf{p}}^{ex}\\ \ddot{\mathbf{q}}^{ex}\end{bmatrix} + \mathbf{b} + \mathbf{g}\right) = \mathbf{S}^j \mathbf{J}_c^\top \mathbf{f}^r \\
& \ddot{\mathbf{q}} = \begin{bmatrix}\ddot{\mathbf{p}}^{ex}\\ \ddot{\mathbf{q}}^{ex}\end{bmatrix} + \begin{bmatrix}\delta_f\\ \mathbf{0}^{n_j}\end{bmatrix} \\
& \mathbf{f}^r = \mathbf{f}^r_{\text{MPC}} + \delta_{\mathbf{f}^r} \\
& W\mathbf{f}^r \geq \mathbf{0} \qquad \text{(contact force constraint)}
\end{aligned} \tag{2}$$

In Eq. (2), $\delta_{\mathbf{f}^r}$ and δ_f are slack variables used to adjust the ground reaction forces and torso acceleration respectively. Additionally, W denotes the contact constraint matrix and n_j the number of leg joints. Finally, using the results from Eq. (2), the WBIC computes the expected joint torques via:

$$\begin{bmatrix}-\\ \mathbf{a}^{ex}\end{bmatrix} = \mathbf{A}\begin{bmatrix}\ddot{\mathbf{p}}\\ \ddot{\mathbf{q}}\end{bmatrix} + \mathbf{b} + \mathbf{g} - \mathbf{J}_c^\top \mathbf{f}^r \tag{3}$$

II. Training Details

The learning model used in RINA, $\Psi(\cdot)$, was implemented in PyTorch [3] as a five layer MLP with dimensions $[45, 100, 128, 100, 16]$ with ReLU activations between layers. RINA was trained using gradient descent based on the ADAM optimizer [4] over 10,000 epochs on an 8-core i9 machine with 64 GB of memory and a NVIDIA RTX-2080 GPU, taking roughly four hours. The values of the training hyperparameters for $\Psi(\cdot)$, which can seen in Table I. The MLP baseline shared the same architecture as RINA expect for the final layer, which is of size 12 in the MLP to predict the joint torque residual dynamics directly. Additionally, the MLP baseline was not trained using layer-wise spectral normalization. The value of γ was held constant between training and execution.

*This work was partially supported by NSF CAREER Award IIS-2308492 and DARPA Young Faculty Award (YFA) D21AP10114-00.

Oscar Youngquist and Hao Zhang are with Human-Centered Robotics Lab, University of Massachusetts Amherst, Amherst, MA 01002, USA. Email: {oyoungquist, hao.zhang}@umass.edu

TABLE I

TRAINING HYPERPARAMETERS

Symbol	**Description**	**Value**
-	Learning Rate	0.0007
-	Training Batch $\mathbf{B}^T$ Size	2048
K	Adaptation Batch $\mathbf{B}^A$ Size	512
λ	Discriminator Regularization Strength	0.1
γ	Maximum Magnitude of $\mathbf{M}$	10.0
κ	Maximum Layer-wise Spectral Norm. of $\Psi(\cdot)$	6.0

REFERENCES

[1] 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.

[2] 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," *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS)*, 2018.

[3] A. Paszke, S. Gross, F. Massa, A. Lerer, J. Bradbury, G. Chanan, T. Killeen, Z. Lin, N. Gimelshein, L. Antiga *et al.*, "Pytorch: An imperative style, high-performance deep learning library," *Advances in Neural Information Processing Systems*, 2019.

[4] D. P. Kingma, "Adam: A method for stochastic optimization," *arXiv preprint arXiv:1412.6980*, 2014.