

# Supplementary Material for X-Nav: Learning End-to-End Cross-Embodiment Navigation for Mobile Robots

## A. Reward Function for Expert Policies

The reward function is computed as  $r = r_{task} * \exp(c_{reg} * r_{reg})$  [1]. The definition of the task reward  $r_{task}$  and regularization rewards  $r_{reg}^{wheel}$  and  $r_{reg}^{quad}$  are defined below in Table A.I.

TABLE A.I: REWARD FUNCTION

Reward	Expression	Explanation
$r_{task}$	$r_{pos,(soft/hard)} = \frac{c_{1,(soft/hard)}}{1 + \left\  \frac{d_{goal}}{c_{2,(soft/hard)}} \right\ ^2} \cdot \mathbb{1}(t > T - T_r)$ $r_{fwd} = c_{fwd} \cdot \exp\left(-\frac{\ v_{ref} - v_x\ ^2}{\sigma_{vel}}\right) \cdot \mathbb{1}(\delta_{goal} < \sigma_{direct})$	<p>Encourages the robot to move to the goal position [2]. <math>T_r</math> denotes the duration of timesteps for which <math>r_{pos}</math> is activated. <math>r_{pos,soft}</math> is a dense reward used to encourage exploration using a larger <math>c_2</math>, <math>c_{2,soft}</math>, while <math>r_{pos,hard}</math> is a sparse reward for accurate position tracking using a smaller <math>c_2</math>, <math>c_{2,hard}</math>.</p> <p>Encourages the robot to move forward at the reference speed <math>v_{ref}</math>. <math>\delta_{goal}</math> is the heading error relative to the goal, and <math>\sigma_{direct}</math> is an angle threshold for the robot heading in the correct direction.</p>
$r_{reg}^{wheel}$	$r_{reg}^{wheel} = c_a \ \dot{a}\ ^2$ $r_{collide} = c_{collide} \cdot \mathbb{1}_{collide}$ $r_{stop} = \frac{c_{1,stop}}{1 + \left\  \frac{a}{c_{2,stop}} \right\ ^2} \cdot \mathbb{1}(\delta_{goal} < \sigma_{direct})$	<p><math>\dot{a}</math> denotes the action rate. <math>\mathbb{1}_{collide}</math> indicates whether a collision has happened between the robot and obstacles.</p> <p>Ensures the robot stops at its goal position. <math>\sigma_{hard}</math> is a distance threshold used to determine when the robot has reached its goal position.</p>
$r_{reg}^{quad}$	$r_{stop} = \frac{c_{1,stop}}{1 + \left\  \frac{q - q_{nom}}{c_{2,stop}} \right\ ^2} \cdot \mathbb{1}(d_{goal} < \sigma_{hard})$ $r_{collide} = c_{collide} \cdot \mathbb{1}_{collide}$ $r_{\ddot{q}} = c_{\ddot{q}} \ \ddot{q}\ ^2$ $r_{v_z} = c_{v_z} v_z^2$ $r_{\omega} = c_{\omega} (\omega_{xy}^2)$ $r_{\tau} = c_{\tau} \ \tau\ ^2$ $r_{\ddot{a}} = c_{\ddot{a}} \ a_t - a_{t-1}\ ^2$ $r_{dev} = c_{dev} \cdot \ q - q_{nom}\ ^2$ $r_{\ddot{a}} = c_{\ddot{a}} \ a_t - 2a_{t-1} + a_{t-2}\ ^2$ $r_{swing} = c_{swing} \cdot \sum_{i=1}^4 (1 - s_i) \cdot (1 - \exp(-f_i^2 / \sigma_{force}))$ $r_{stance} = c_{stance} \cdot \sum_{i=1}^4 s_i \cdot (1 - \exp(-\ v_{foot,i}\ ^2 / \sigma_{foot}))$ $r_{slide} = c_{slide} \cdot \sum_{i=1}^4 \ v_{foot,i}^{xy}\  \cdot \mathbb{1}(f_i > f_{thresh})$	<p>Ensures the robot stops at its goal position. <math>q_{nom}</math> are the nominal joint positions, defined as the initial joint angles of the robot when it stands in its default posture.</p> <p><math>\ddot{q}</math> is the joint acceleration.</p> <p><math>v_z</math> is the vertical velocity of the robot base, and <math>\omega_{xy}</math> is the angular velocity of the base around <math>x</math> and <math>y</math> axes.</p> <p><math>\tau</math> is the joint torque.</p> <p>Penalizes joint deviation from the nominal joint positions.</p> <p>Encourages the quadruped feet to track the leg swing phase [1]. <math>f_i</math> is the contact force of the <math>i</math>-th foot of the quadruped. <math>s_i</math> is the desired contact state of the foot.</p> <p>Encourages the quadruped feet to track the leg stance phase [1]. <math>v_{foot,i}</math> is the velocity of the <math>i</math>-th foot.</p> <p>Avoids feet sliding on the ground. <math>f_{thresh}</math> is a force threshold for determining feet contact.</p>

\*  $c_{1,(soft/hard)}$ ,  $c_{2,(soft/hard)}$ ,  $c_{fwd}$ ,  $c_{1,stop}$ ,  $c_{2,stop}$ ,  $c_{collide}$ ,  $c_a$ ,  $c_{v_z}$ ,  $c_{\omega}$ ,  $c_{\tau}$ ,  $c_{\ddot{q}}$ ,  $c_{dev}$ ,  $c_{\ddot{a}}$ ,  $c_{swing}$ ,  $c_{stance}$ ,  $c_{slide}$  and  $c_{reg}$  are all constants.

## B. Parameters and Value Ranges for Embodiment Randomization

TABLE B.I: PARAMETERS FOR EMBODIMENT RANDOMIZATION

Type	Parameters	Range	Parameters	Range
Small-Sized Quadrupeds	Base length	[0.24, 0.91] m	Thigh mass	[0.56, 1.69] kg
	Base width	[0.16, 0.39] m	Calf radius	[0.02, 0.05] m
	Base height	[0.06, 0.21] m	Calf length	[0.12, 0.39] m
	Base mass	[4.8, 19.5] kg	Calf mass	[0.12, 0.39] kg
	Thigh radius	[0.02, 0.05] m	Motor P gain	[0.7, 1.3]
	Thigh length	[0.16, 0.46] m	Motor D gain	[0.7, 1.3]
Large-Sized Quadrupeds	Base length	[0.56, 1.04] m	Thigh mass	[2, 5.2] kg
	Base width	[0.28, 0.52] m	Calf radius	[0.02, 0.04] m
	Base height	[0.14, 0.26] m	Calf length	[0.24, 0.36] m
	Base mass	[24, 39] kg	Calf mass	[0.4, 0.6] kg
	Thigh radius	[0.03, 0.05] m	Motor P gain	[0.5, 1.3]
	Thigh length	[0.24, 0.39] m	Motor D gain	[0.5, 1.3]
Wheeled Robots	Base length	[0.3, 0.8] m	Base height	[0.15, 0.3] m
	Base width	[0.2, 0.65] m	Base mass	[5, 20] kg

## C. Parameters and Value Ranges for Domain Randomization

TABLE C.I: PARAMETERS AND VALUE RANGES FOR DOMAIN RANDOMIZATION

Parameters	Ranges	Parameters	Ranges
Static friction	[0.7, 1.1]	Push disturbance	[-0.5, 0.5] m/s
Dynamic friction	[0.6, 1.0]	Push Interval	[4, 8] s
Added mass	[0.0, 2.0] kg	Ray distance noise	[-0.1, 0.1] m
Linear velocity noise	[-0.1, 0.1] m/s	Angular velocity noise	[-0.1, 0.1] rad/s
Projected gravity noise	[-0.05, 0.05] m/s <sup>2</sup>	Joint velocity noise	[-1.0, 1.0] rad/s
Joint position noise	[-0.01, 0.01] rad	Motor Delay	[0.02, 0.08] s

## D. Hyperparameters and Values

TABLE D.I: HYPERPARAMETERS AND VALUES

Parameter	Value	Parameter	Value	Parameter	Value
$x_{scan}$	2.5 m	$y_{scan}$	2.5 m	$c_{collide}$	-40
$c_{1,soft}$	10	$c_{2,soft}$	5	$T_r$	3 s
$c_{1,hard}$	15	$c_{2,hard}$	0.5	$c_{dev}$	-1
$v_{ref}$	[0.5, 1.0] m/s	$c_{\omega}$	-0.05	$T$	12 s
$\sigma_{direct}$	1.75 rad	$c_{\tau}$	-0.0002	$c_{v_z}$	-2
$\sigma_{hard}$	0.5 m	$c_{\dot{a}}$	-0.01	$\sigma_{close}$	0.5 m
$c_{1,stop}$	10	$c_{\ddot{q}}$	-2.5e-7	$\sigma_{far}$	3.0 m
$c_{2,stop}$	0.2	$c_{\ddot{a}}$	-0.05	$c_{fwd}$	2
$c_{swing}$	-5	$c_{stance}$	-5	$c_{slide}$	-5

## References

- [1] G. B. Margolis and P. Agrawal, “Walk These Ways: Tuning Robot Control for Generalization with Multiplicity of Behavior,” in *Proc. Conf. Robot Learn.*, 2022, pp. 1-14.
- [2] J. Jin *et al.*, “Resilient Legged Local Navigation: Learning to Traverse with Compromised Perception End-to-End,” 2023, *arXiv:2310.03581*.