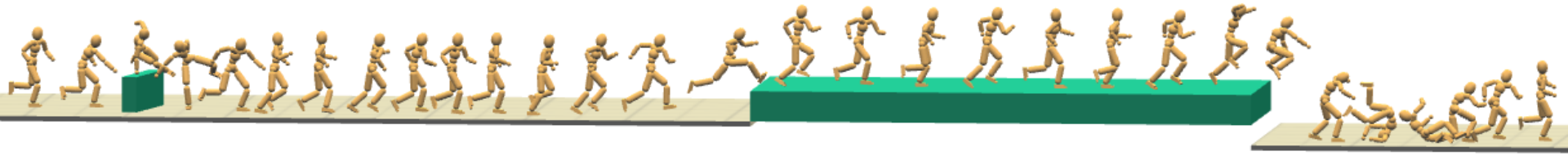
Terrain Runner: Control, Parameterization, Composition, and Planning for Highly Dynamic Motions

Libin Liu* KangKang Yin† Michiel van de Panne‡ Baining Guo §

*Tsinghua University †National University of Singapore

‡University of British Columbia § Microsoft Research Asia



Sponsored by ACM SIGGRAPH

SIGGRAPH ASIA 2012

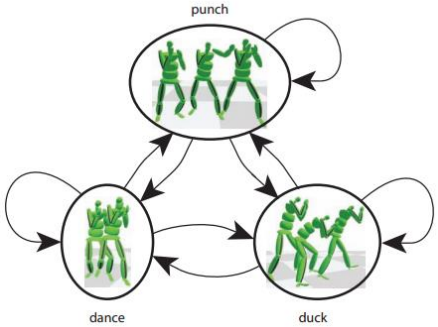
Motivation

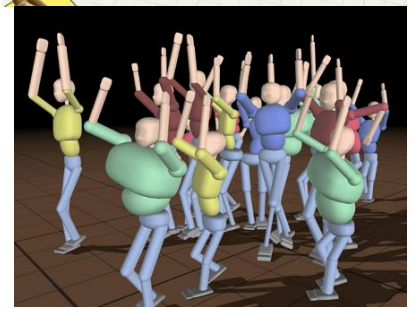
[clips are from YouTube, uploaded by 3runTube, l1consolable, ParkourGenerations, rubenparkour, traceurelements]

Outline

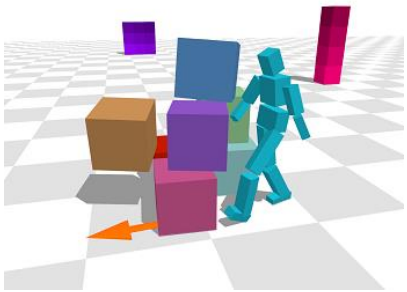
- Motivation
- Related work
- Controller synthesis pipeline + results
- Conclusion

Related Work

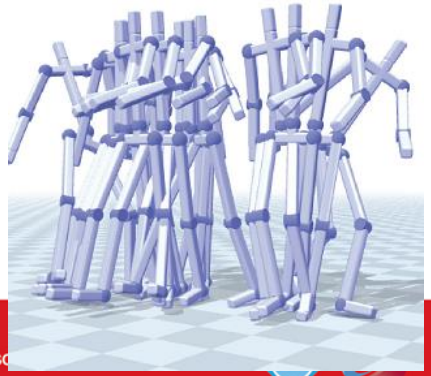




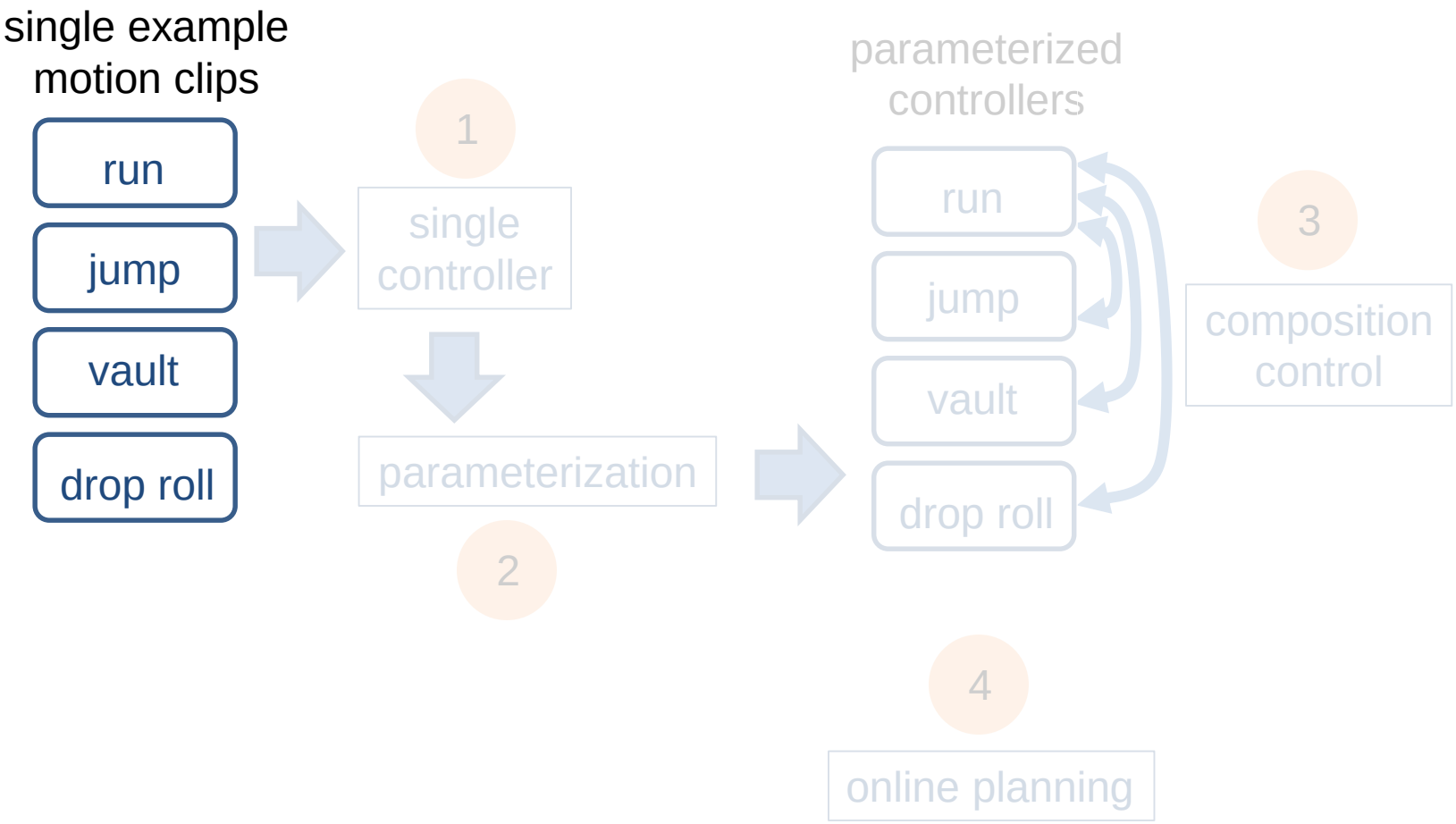
- Kinematic Methods
  - [Kovar et al. 2002; Heck and Gleicher 2007; Min et al. 2009; Treuille et al. 2007; Lee et al. 2009]



- Physics-based Methods
  - Single controllers: [Hodgins et al. 1995; Zhao and van de Panne 2005; Muico et al. 2009; Coros et al. 2010; Lee et al. 2010; Wang et al. 2009]
  - Control Composition: [Faloutsos et al. 2001, Sok et al. 2007, da Silva et al. 2009, Muico et al. 2011, Coros et al. 2011]



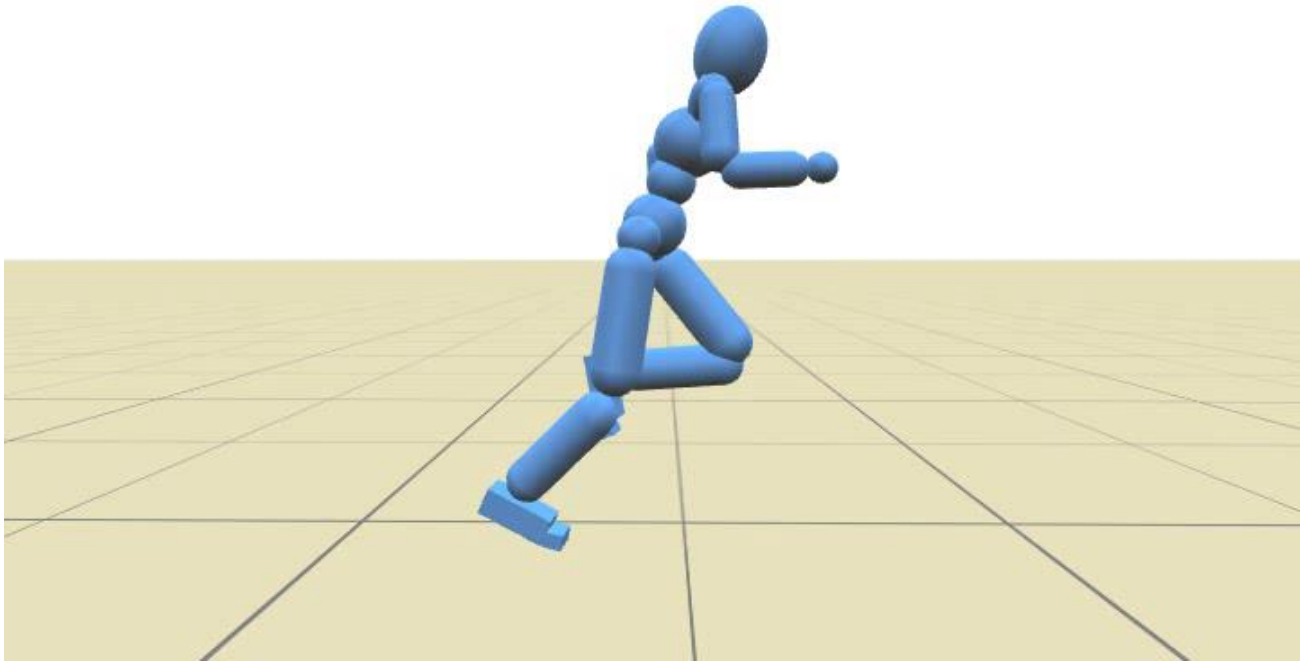
System Overview



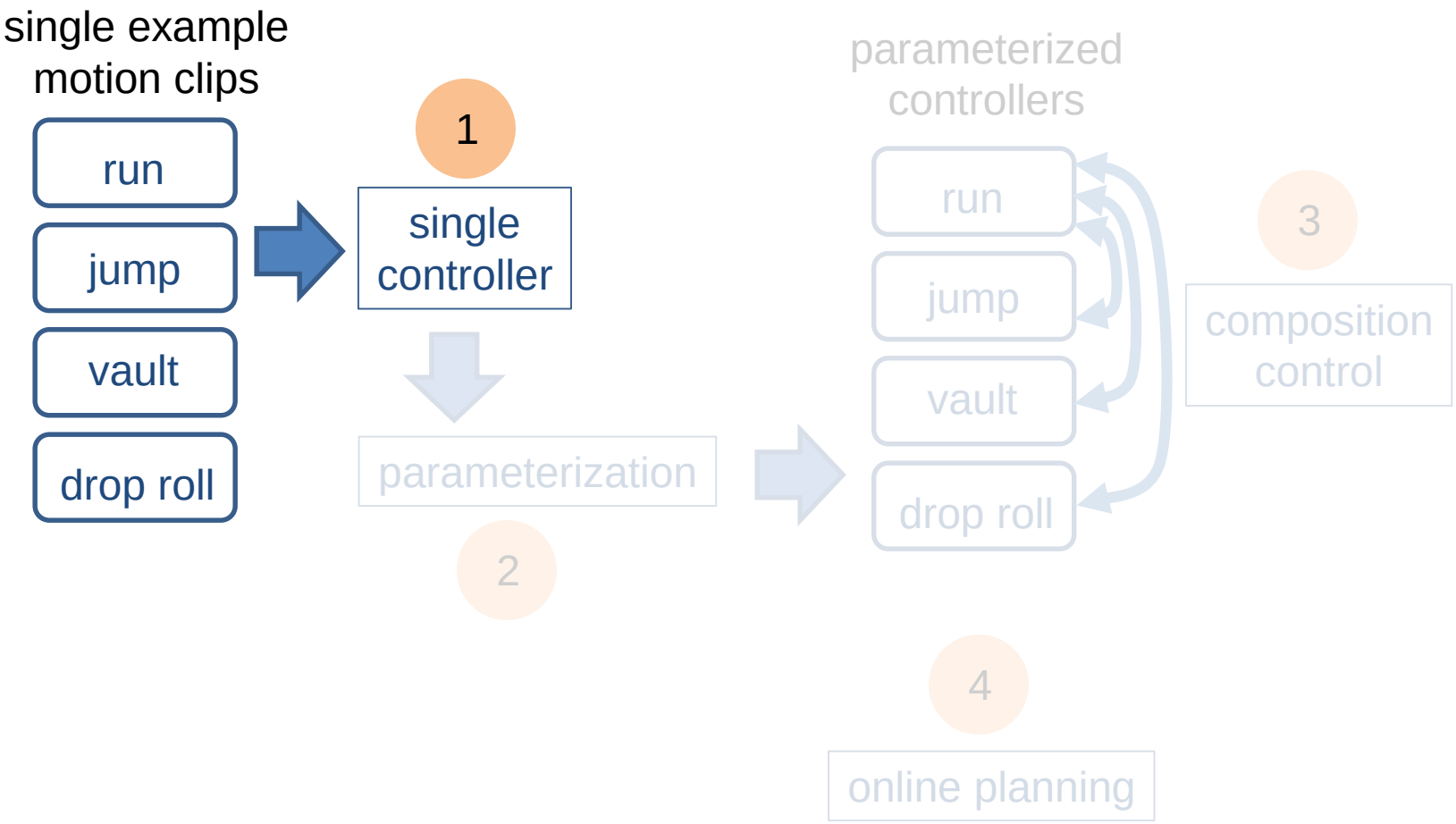
Motion Examples

single example motion clips

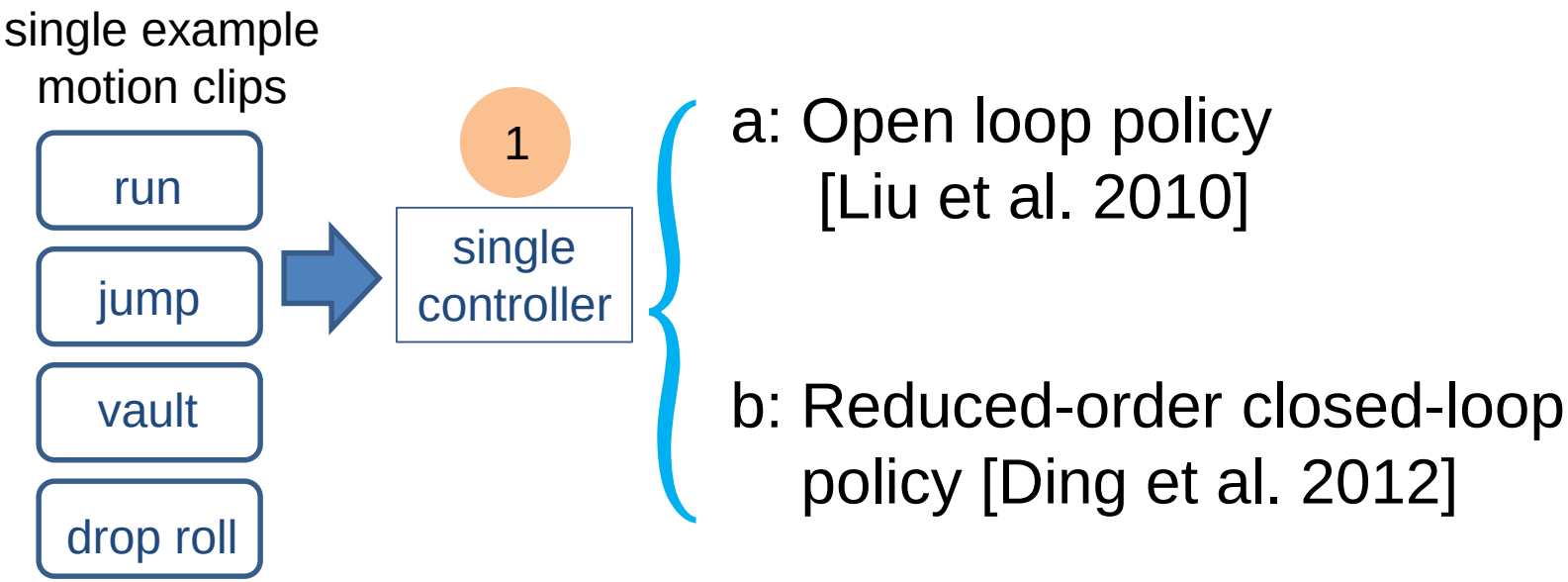
- run
- jump
- vault
- drop roll



System Overview

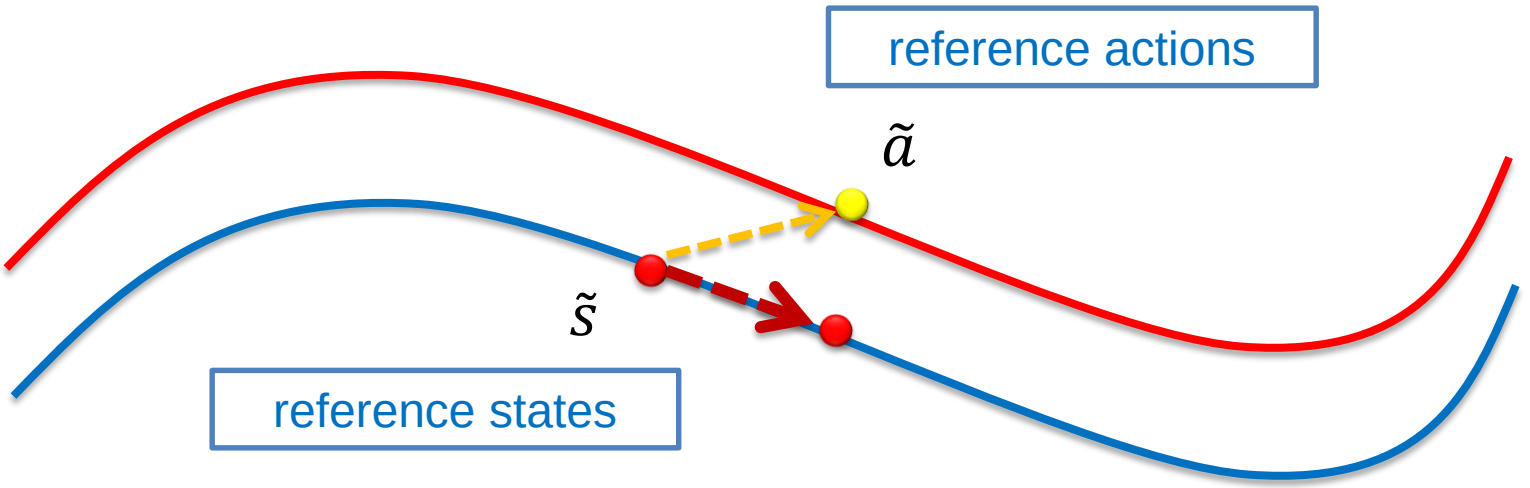


Stage 1: Single Controller Construction



Stage 1a: Open-loop Policy

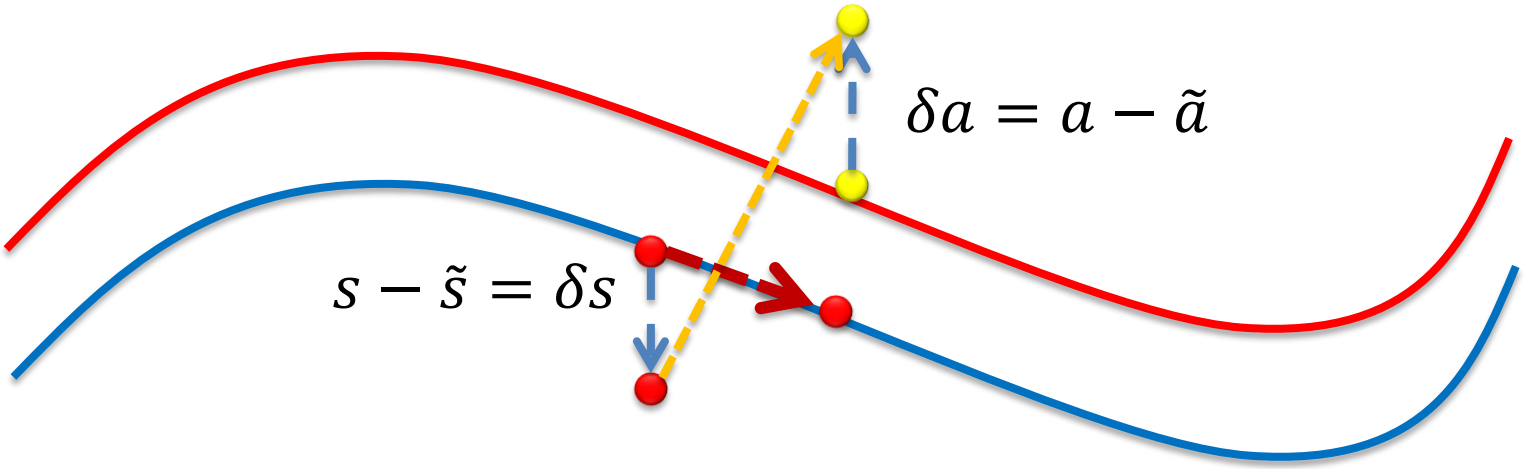
[Liu et al. 2010]: Sampling-based Contact-rich Motion Control, SIGGRAPH 2010



Stage 1b: Reduced-order Closed-loop Policy

[Ding et al. 2012]: Learning reduced-order feedback policies for motion skills. Tech. Rep. University of British Columbia.

$$\delta a = M\,\delta s + \hat{a}$$



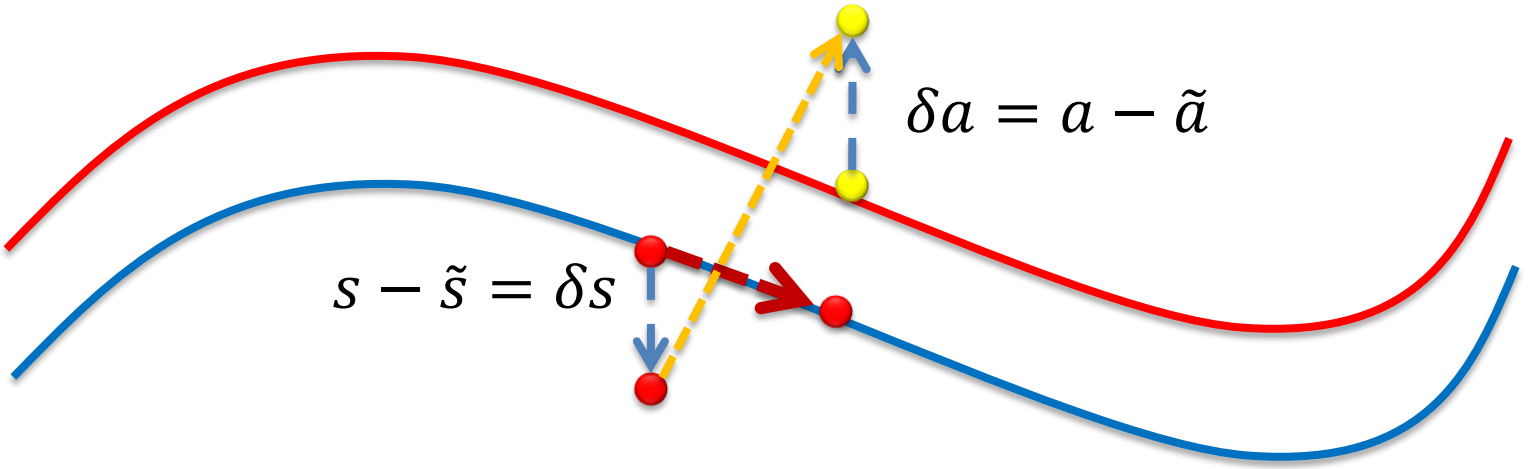
Stage 1b: Reduced-order Closed-loop Policy

$M_{ap} \cdot M_{sp}$

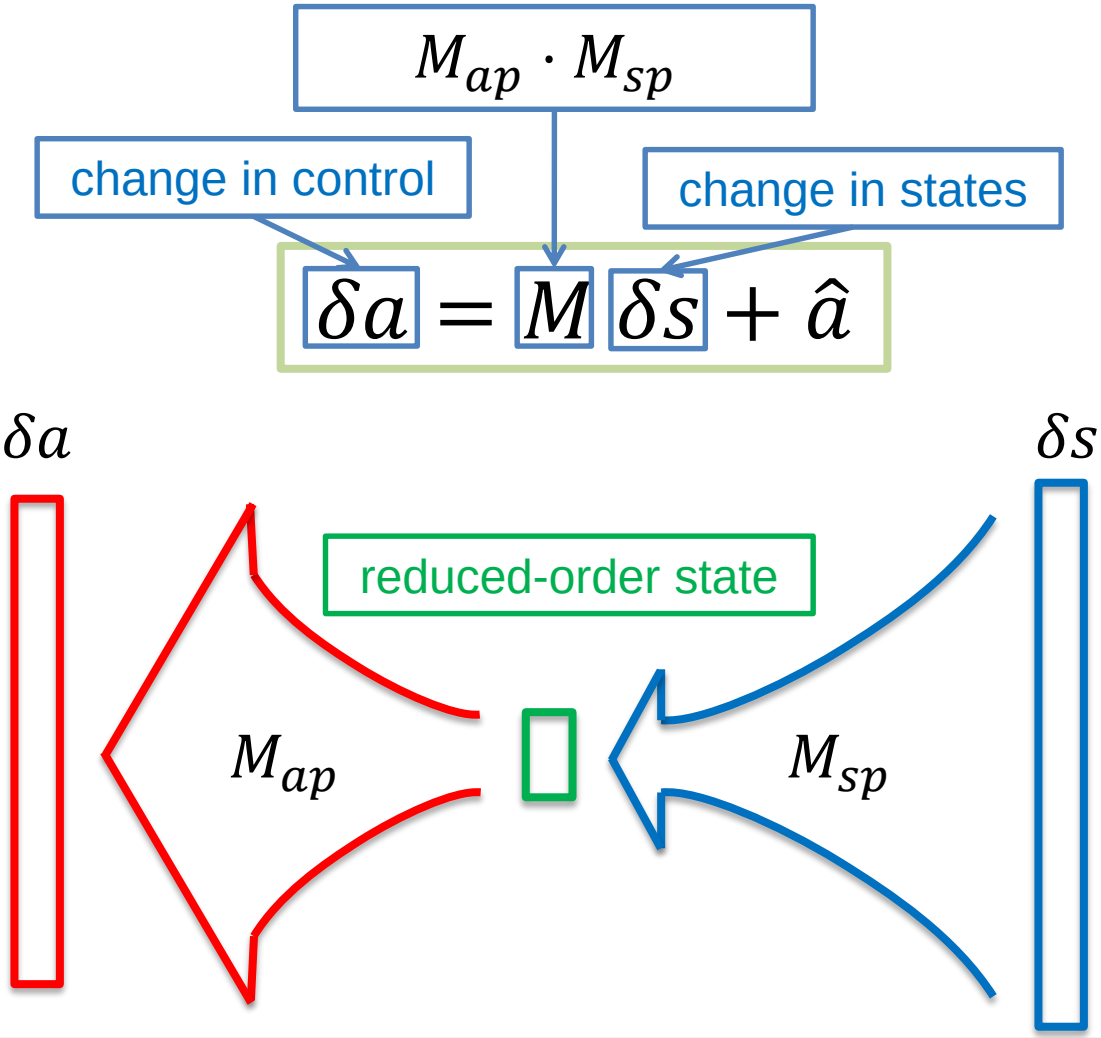
change in control

change in states

$$\delta a = M \, \delta s + \hat{a}$$



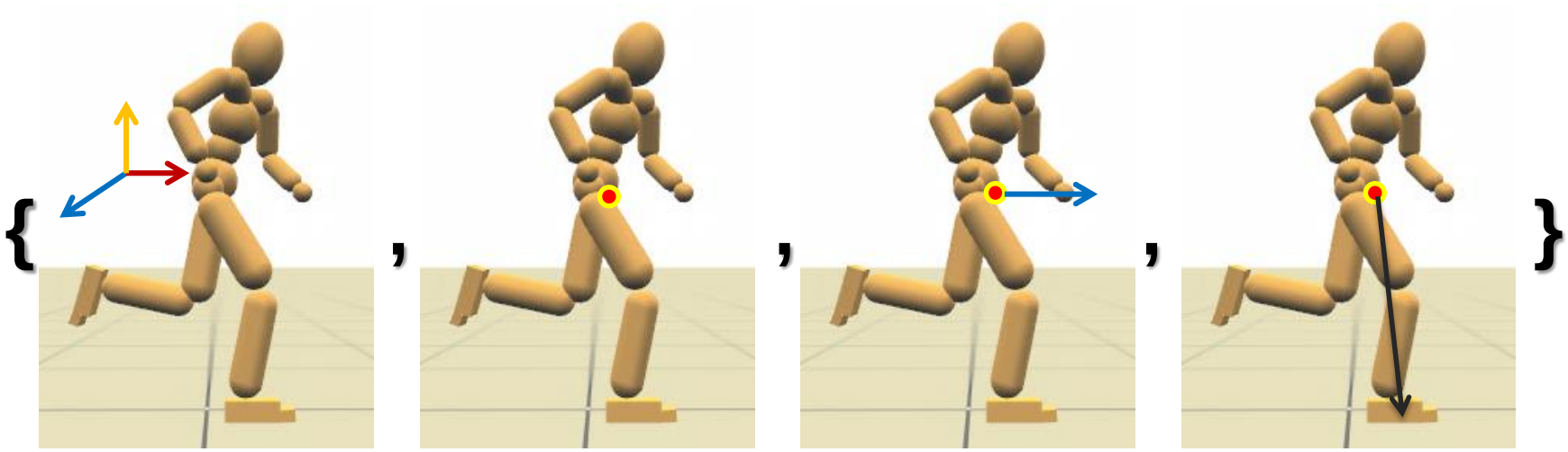
Stage 1b: Reduced-order Closed-loop Policy



Stage 1b: Feedback Policy

Manually-selected States: *s*

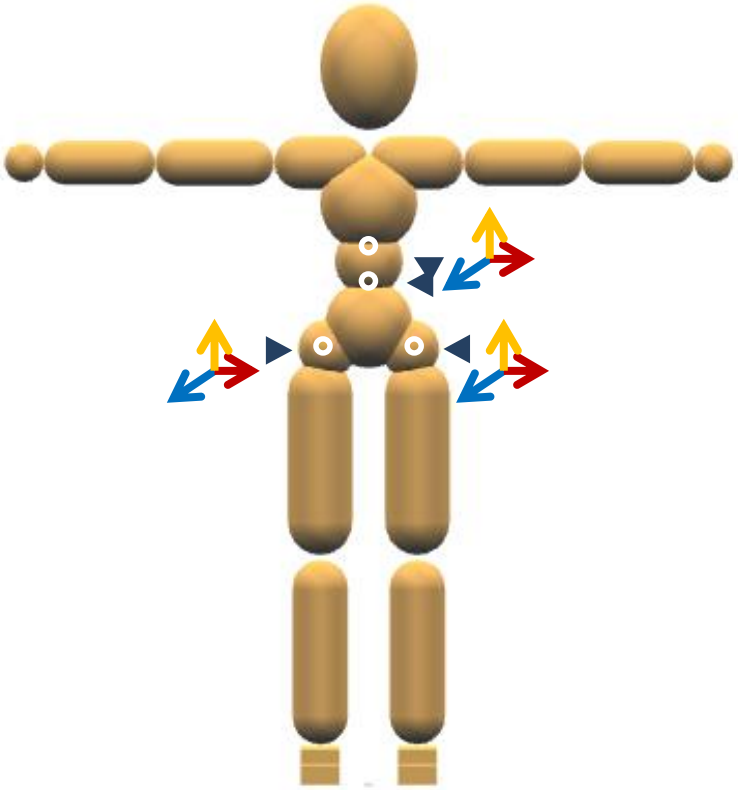
- Running: 12 dimensions



Stage 1b: Feedback Policy

Manually-selected Controls: *a*

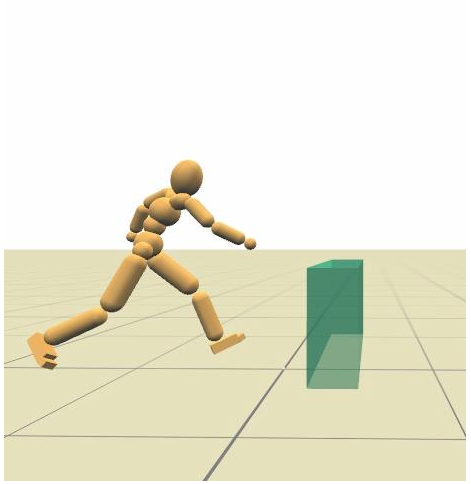
- for all skills: 9 dimensions



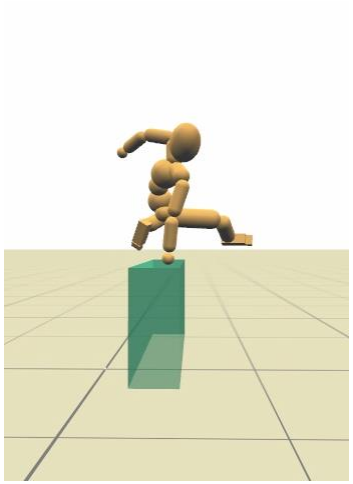
Stage 1b: Feedback Policy

Multi-phase Skills

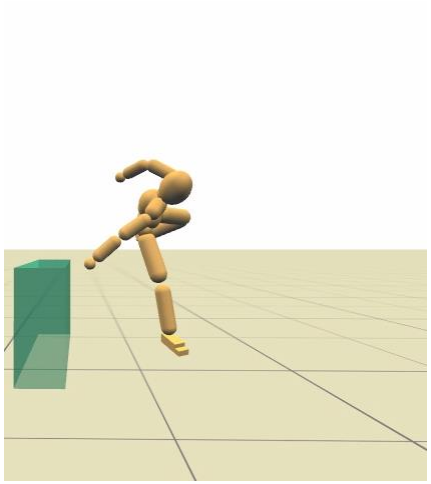
- Vaulting



phase 1: raising



phase 2: dropping

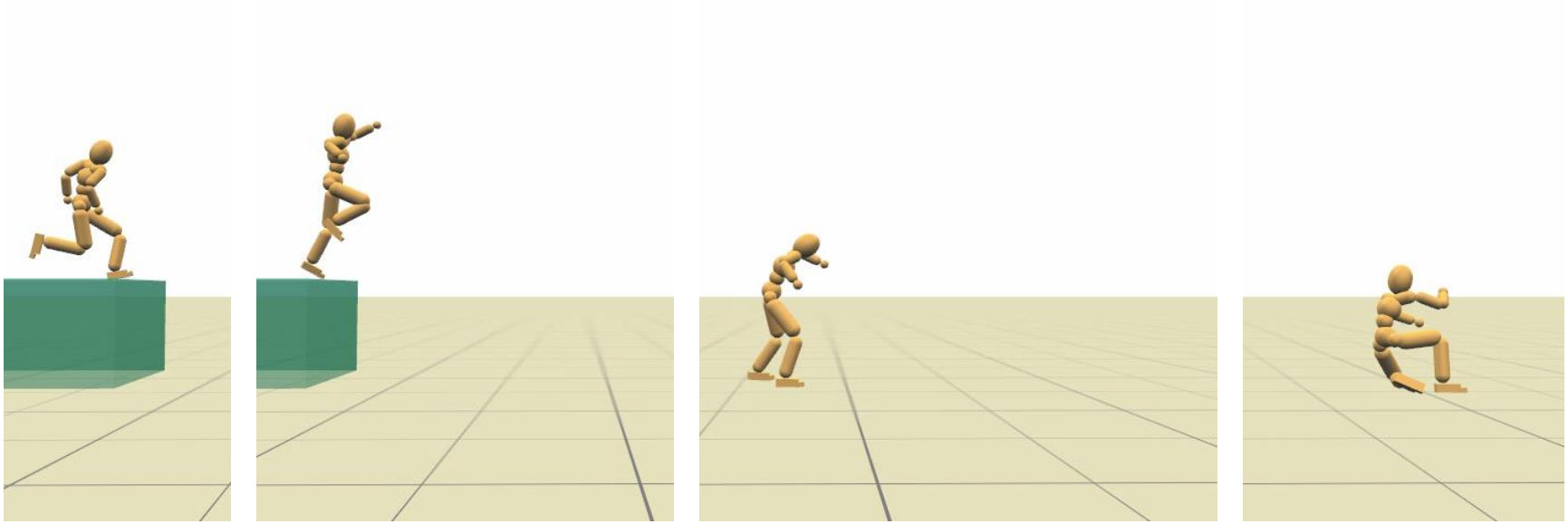


phase 3: landing

Stage 1b: Feedback Policy

Multi-phase Skills

- Drop-rolling

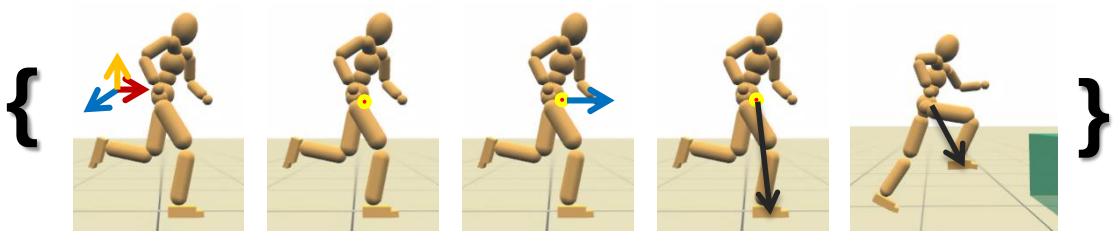


phase 1: jumping phase 2: dropping phase 3: rolling phase 4: standing-up

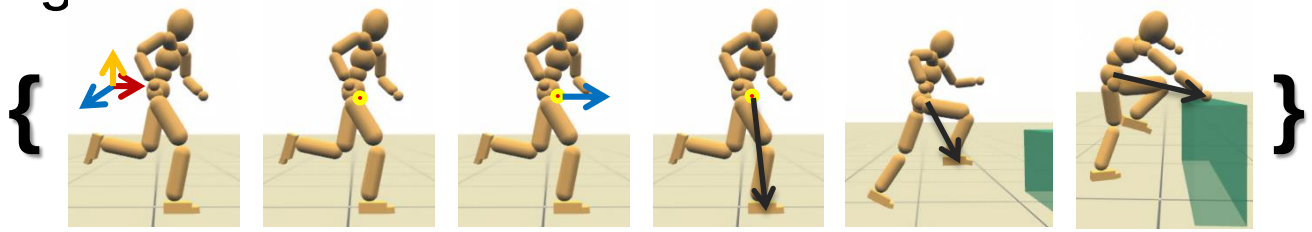
Stage 1b: Feedback Policy

Manually-selected States: *s*

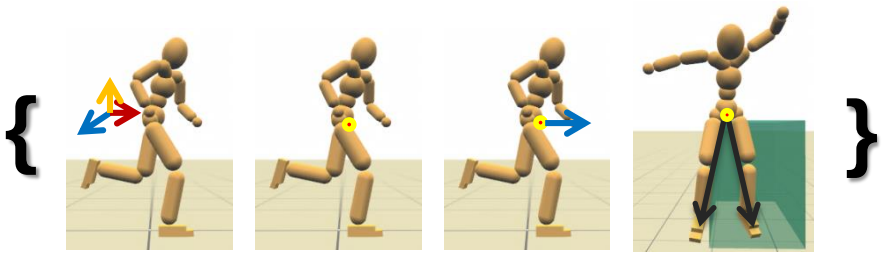
- Jumping



- Vaulting



- Drop-rolling



Stage 1b: Feedback Policy

Optimization

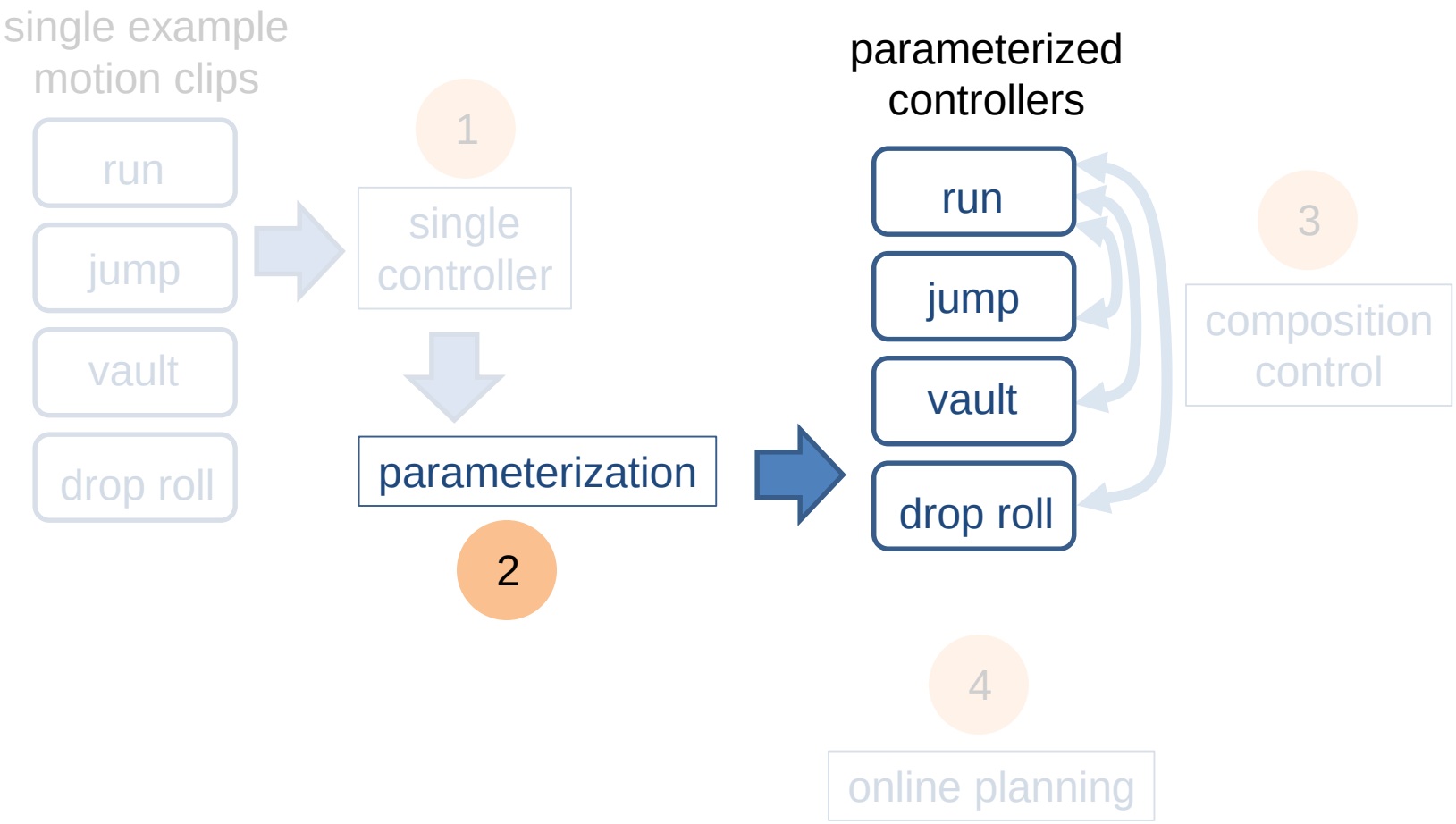
$$\delta a = M\delta s + \hat{a}$$

- Optimize M
  - CMA, Covariance Matrix Adaption ([Hansen 2006])
  - Running:
    - Objective function

      $$E = w_t(N_d T_c - T_s) + w_s E_s + w_p E_p + w_\tau E_\tau$$

    - 12 minutes on 24 cores
    - more details in paper and [Ding et al. 2012]

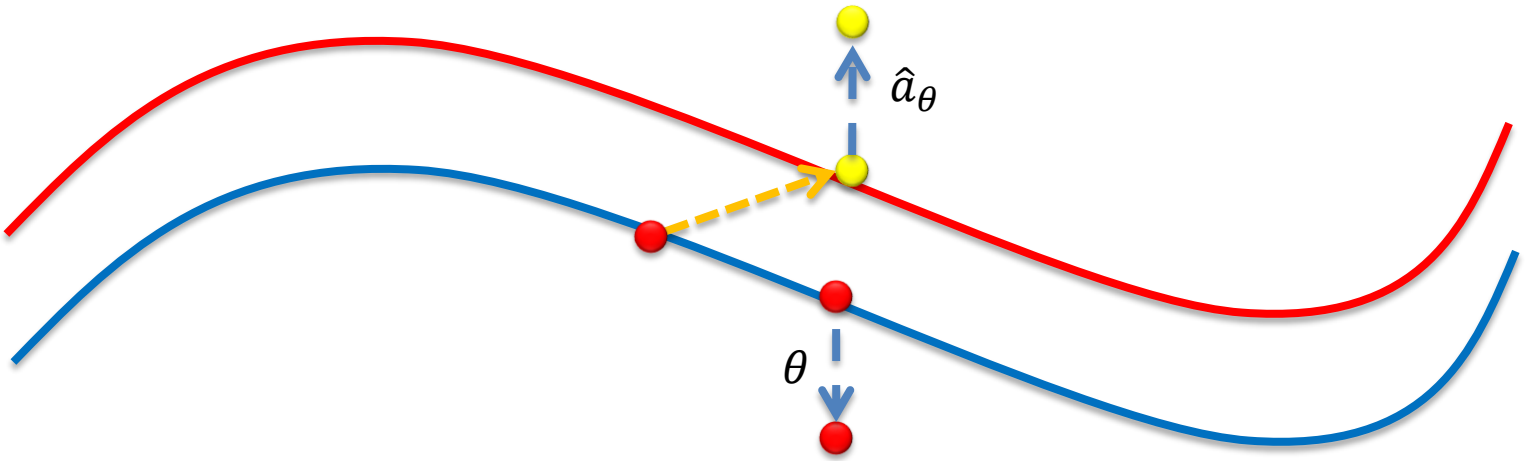
System Overview



Stage 2: Parameterization

$$\delta a = M_\theta \, \delta s + \hat{a}_\theta$$

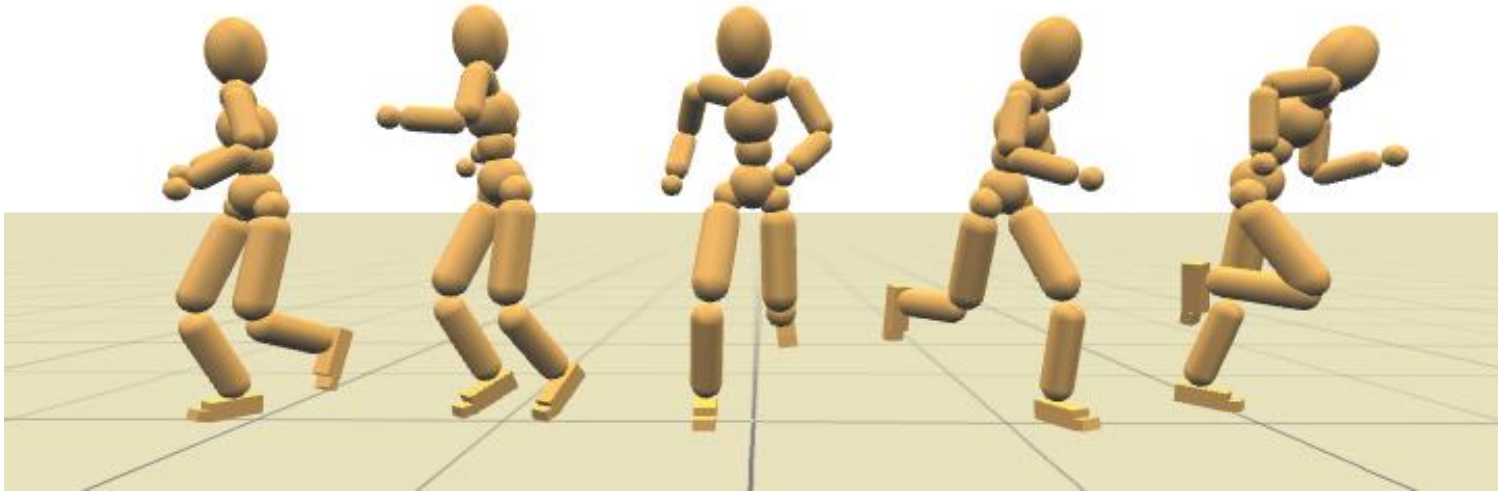
$$\delta a = M \, \delta s + \hat{a}$$



Stage 2: Parameterization

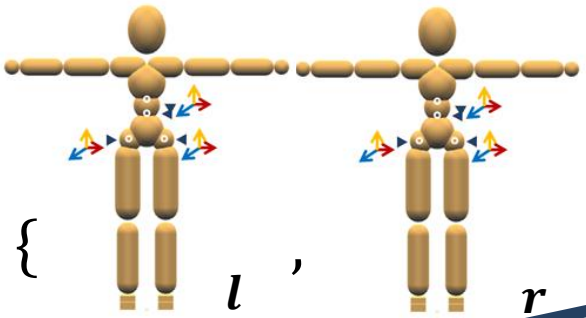
Running: parameter space

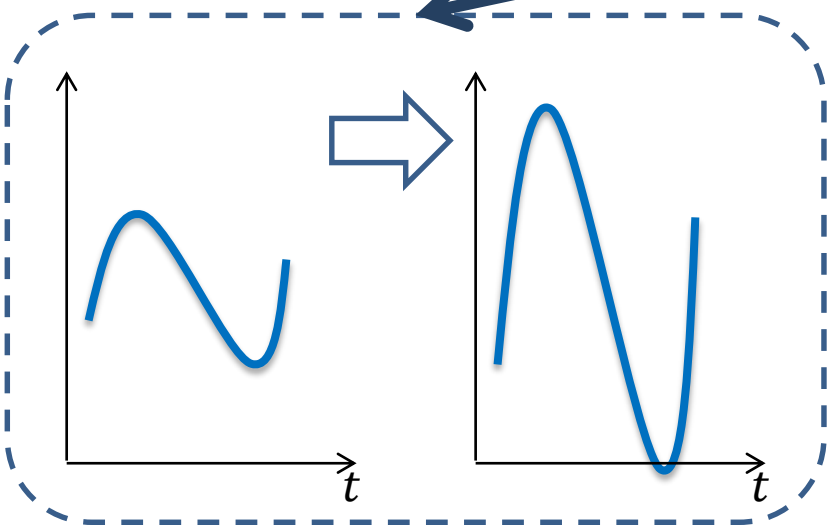
- $\theta = (\mathrm{v}, \phi)$
  - speed, turning rate
  - [2m/s, 5m/s] × $[-6°, 6°]$/step



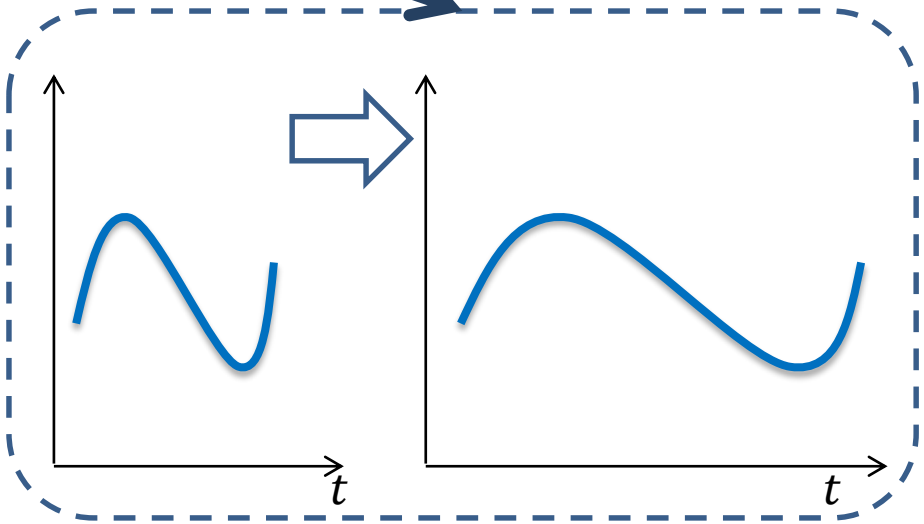
Stage 2: Parameterization

Running: Action Set Augmentation

- $\hat{a}_\theta = \{$  $_l$, $_r$, α , β $\}$



space scaling



time scaling

Stage 2: Parameterization

Running: optimization

- $M_\theta, \hat{a}_\theta$

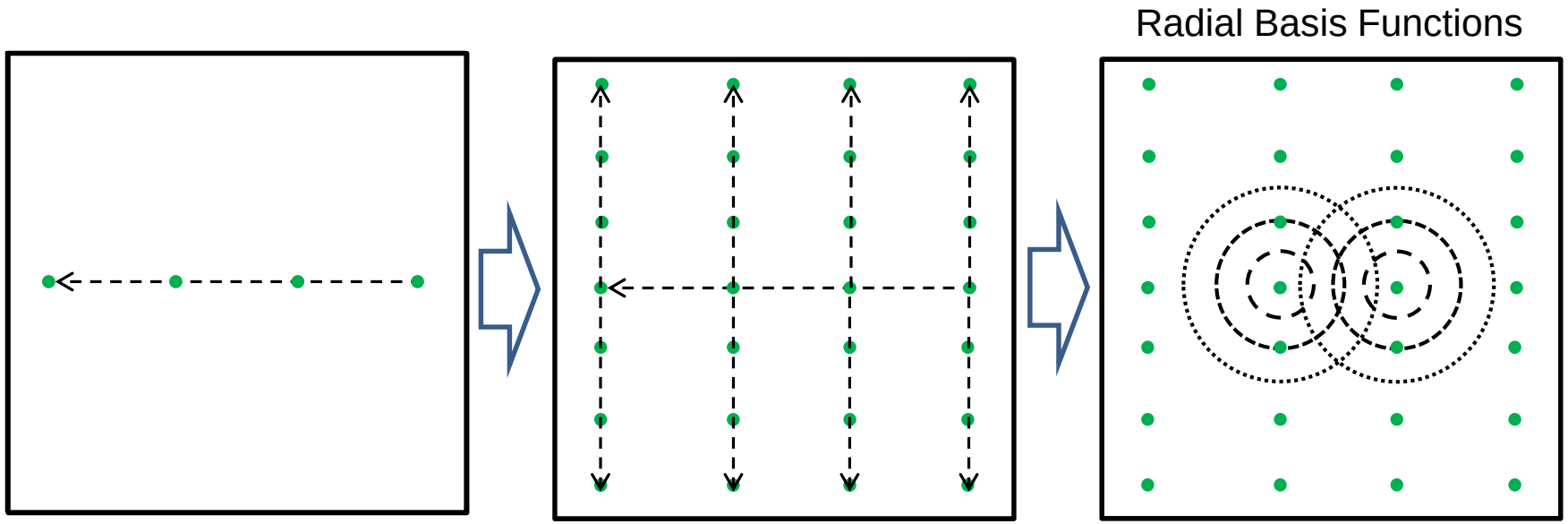
$$E = \overbrace{w_t(N_d T_c - T_s)}^{\text{success}} + \overbrace{\frac{w_h}{T_s} \int ||\mathbf{d}_h - \bar{\mathbf{d}}_h|| dt}^{\text{head's stability}}$$

$$+ \underbrace{\frac{1}{N_s} \left(w_\phi |\phi_i - \phi^*| + w_v |v_i - v^*| + w_f |f_i - f^*|\right)}_{\text{desired parameters}}$$

Stage 2: Parameterization

Continuation

- [Yin et al. 2008]: Continuation methods for adapting simulated skills. SIGGRAPH 2008

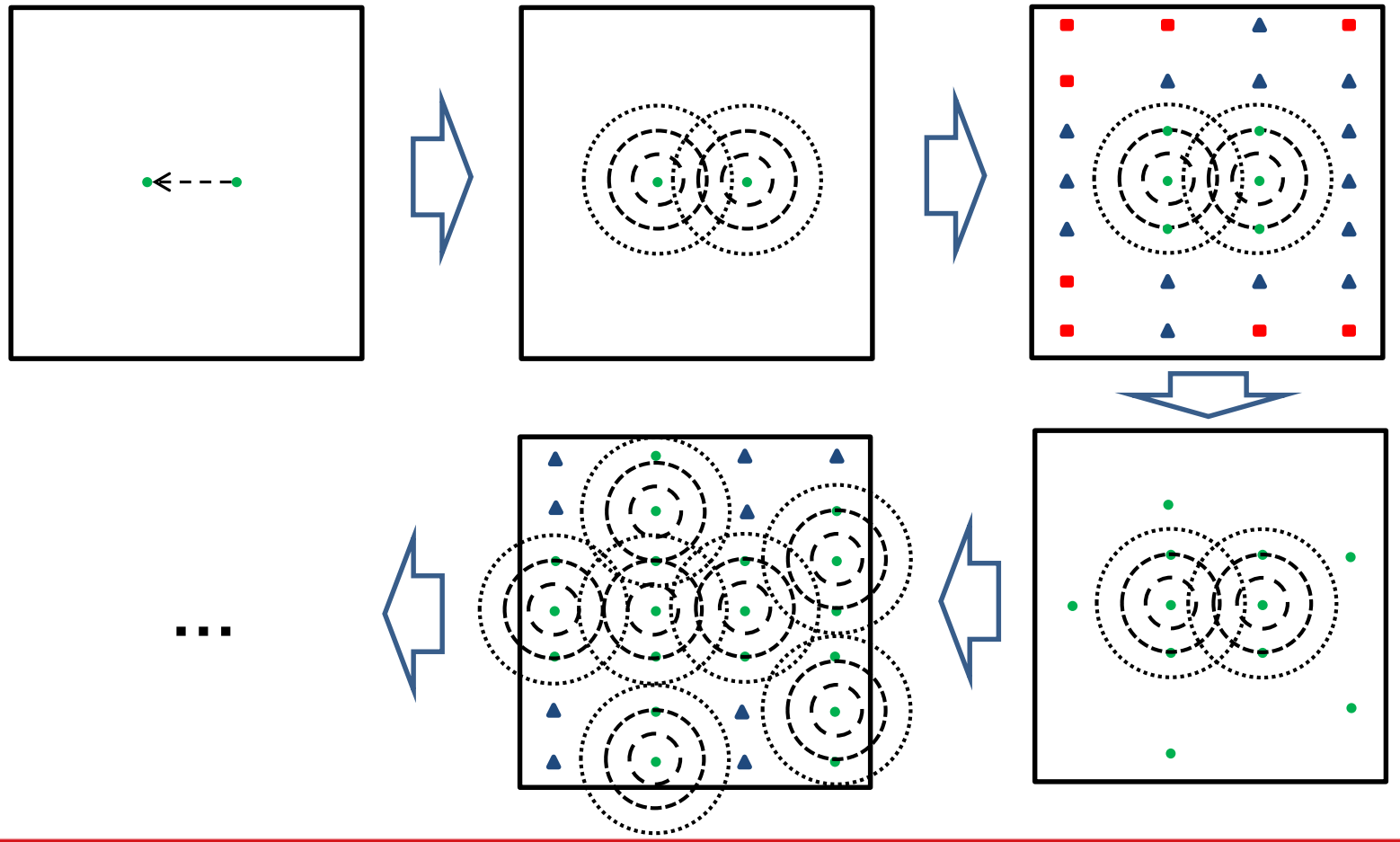


$$f(\boldsymbol{\theta}) = p(\boldsymbol{\theta}) + \sum_{i=1}^{N} \lambda_i \varphi(|\boldsymbol{\theta} - \boldsymbol{\theta}_i|)$$

Stage 2: Parameterization

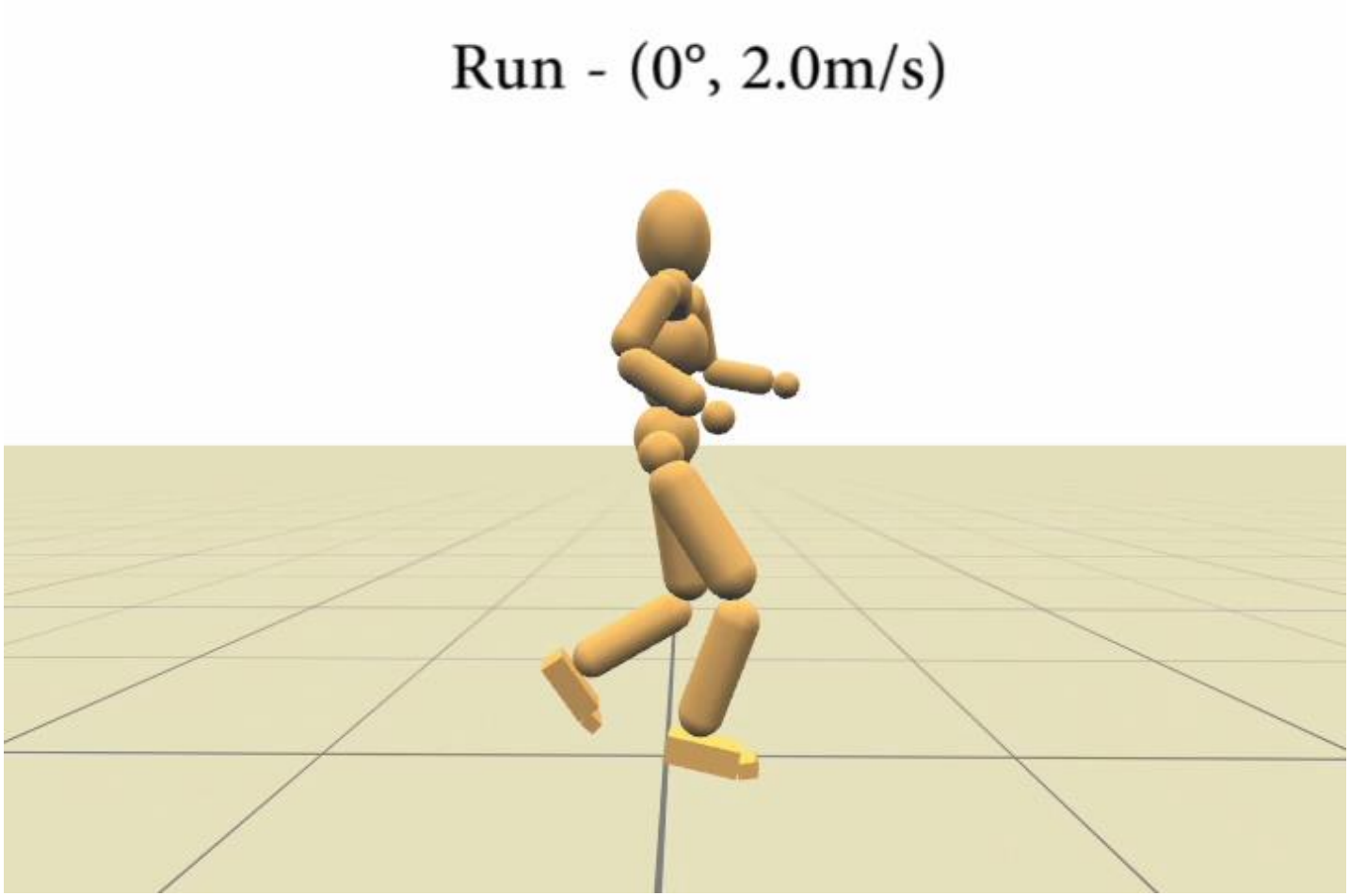
Continuation

- Predictor-corrector



Stage 2: Parameterization

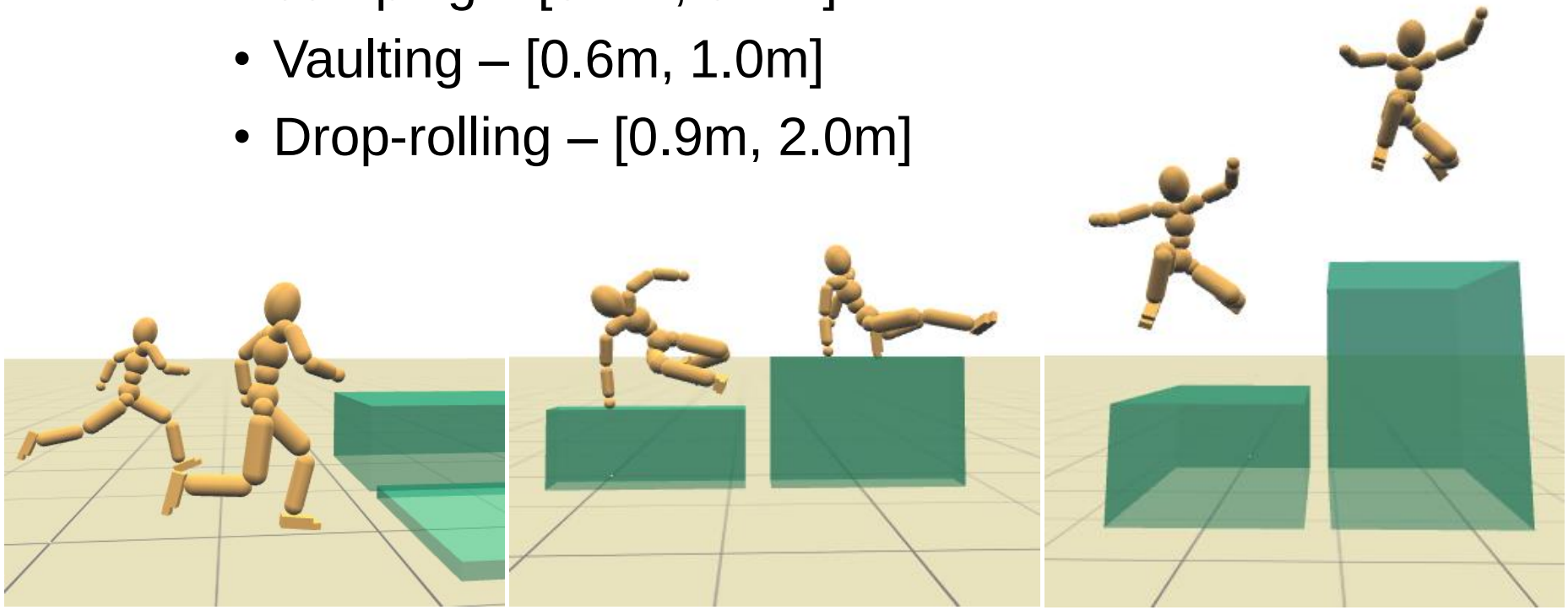
Running Results



Stage 2: Parameterization

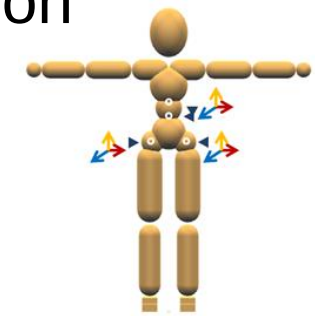
Obstacle Clearing Maneuvers

- $\theta = h$
  - Obstacle height
    - Jumping – [0.1m, 0.7m]
    - Vaulting – [0.6m, 1.0m]
    - Drop-rolling – [0.9m, 2.0m]



Stage 2: Parameterization

Obstacle Clearing Maneuvers

- Optimization
  - $\hat{a}_h = \{$  $, \alpha, \beta\}$

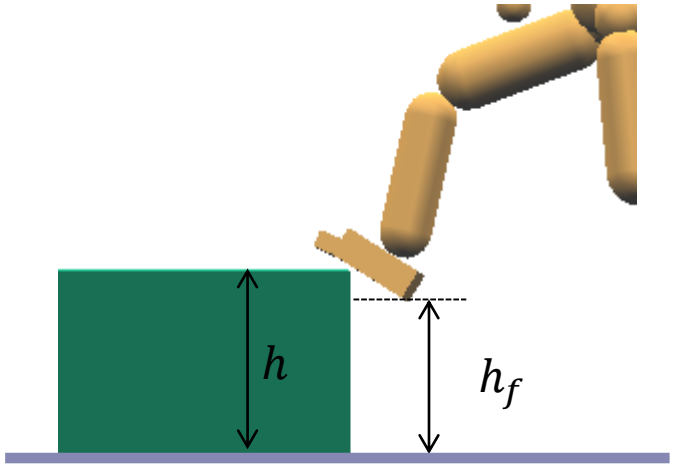
contact balance pose

$$E_h = w_c E_c + w_b E_b + w_p E_p$$

Stage 2: Parameterization

Obstacle Clearing Maneuvers

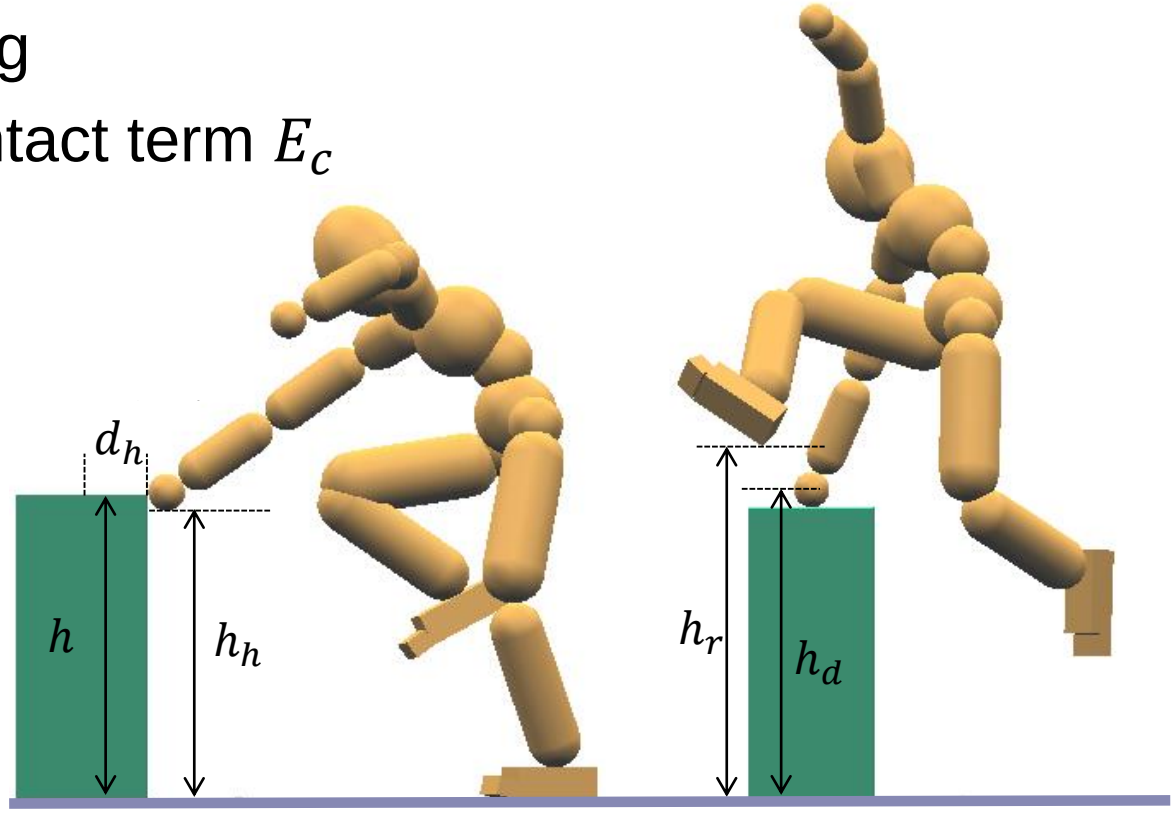
- Jumping
  - Contact term E_c



Stage 2: Parameterization

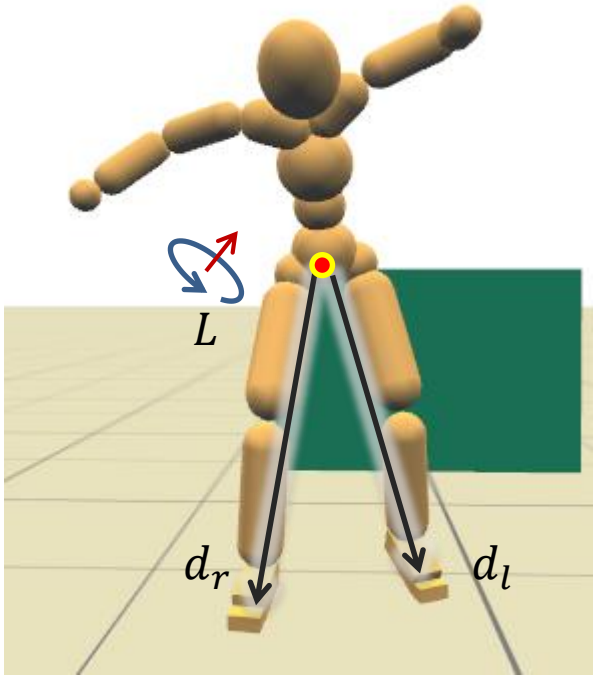
Obstacle Clearing Maneuvers

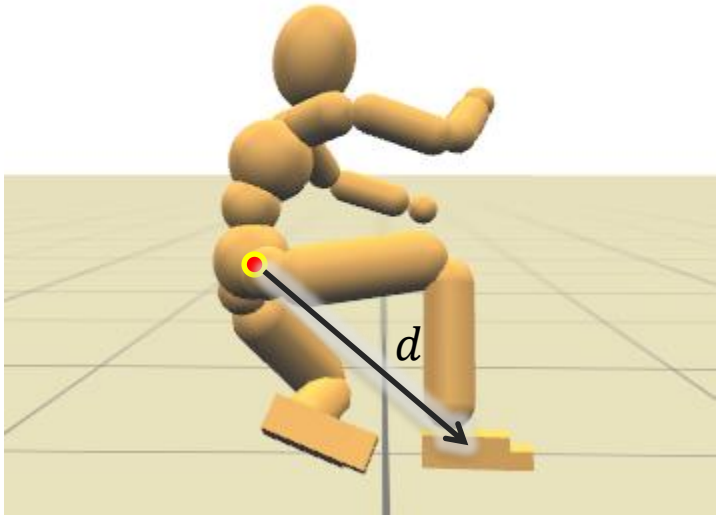
- Vaulting
  - Contact term E_c



Obstacle Clearing Maneuvers

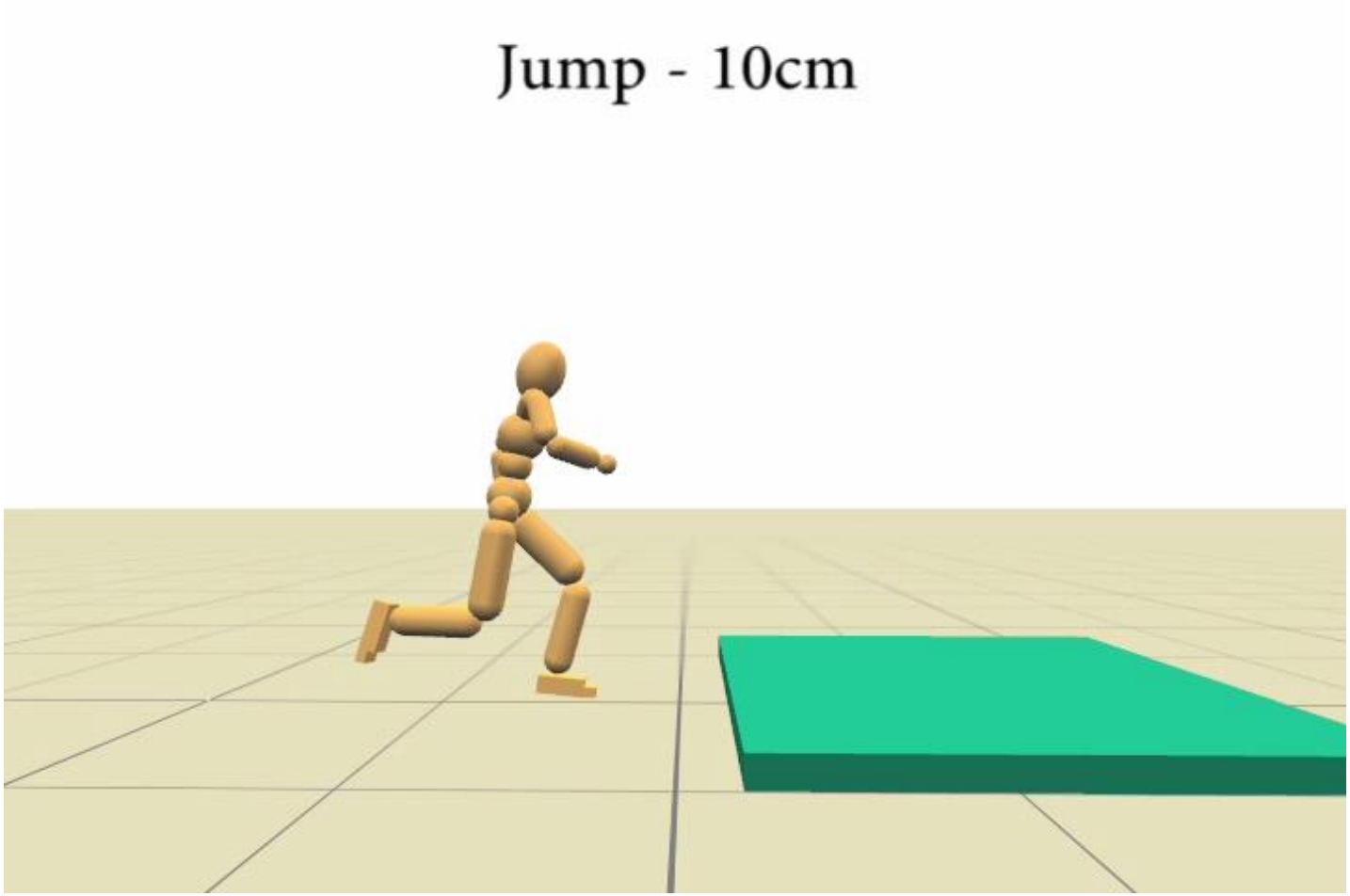
- Drop-rolling
  - Balance term E_b



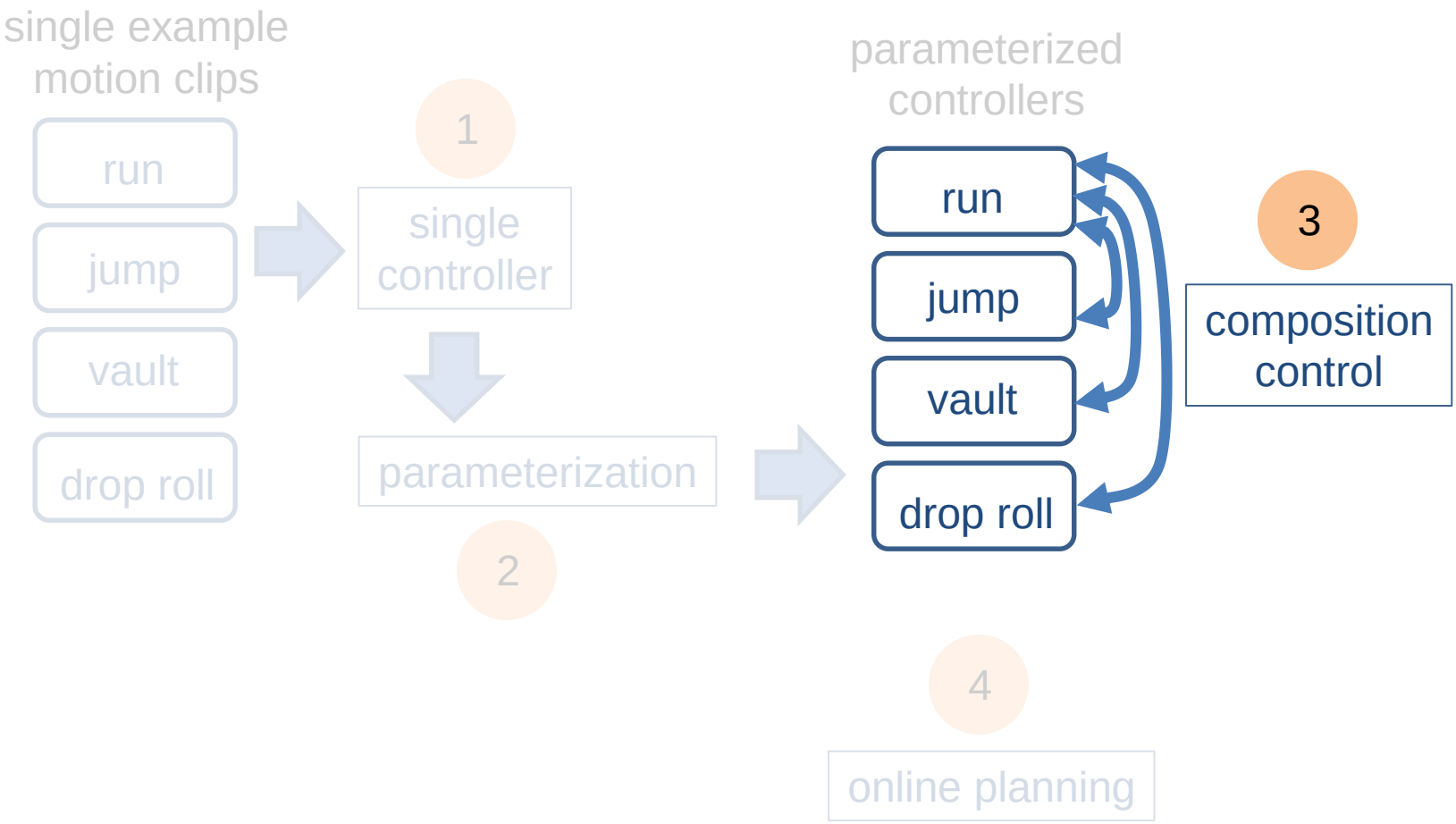


Stage 2: Parameterization

Obstacle Clearing Results

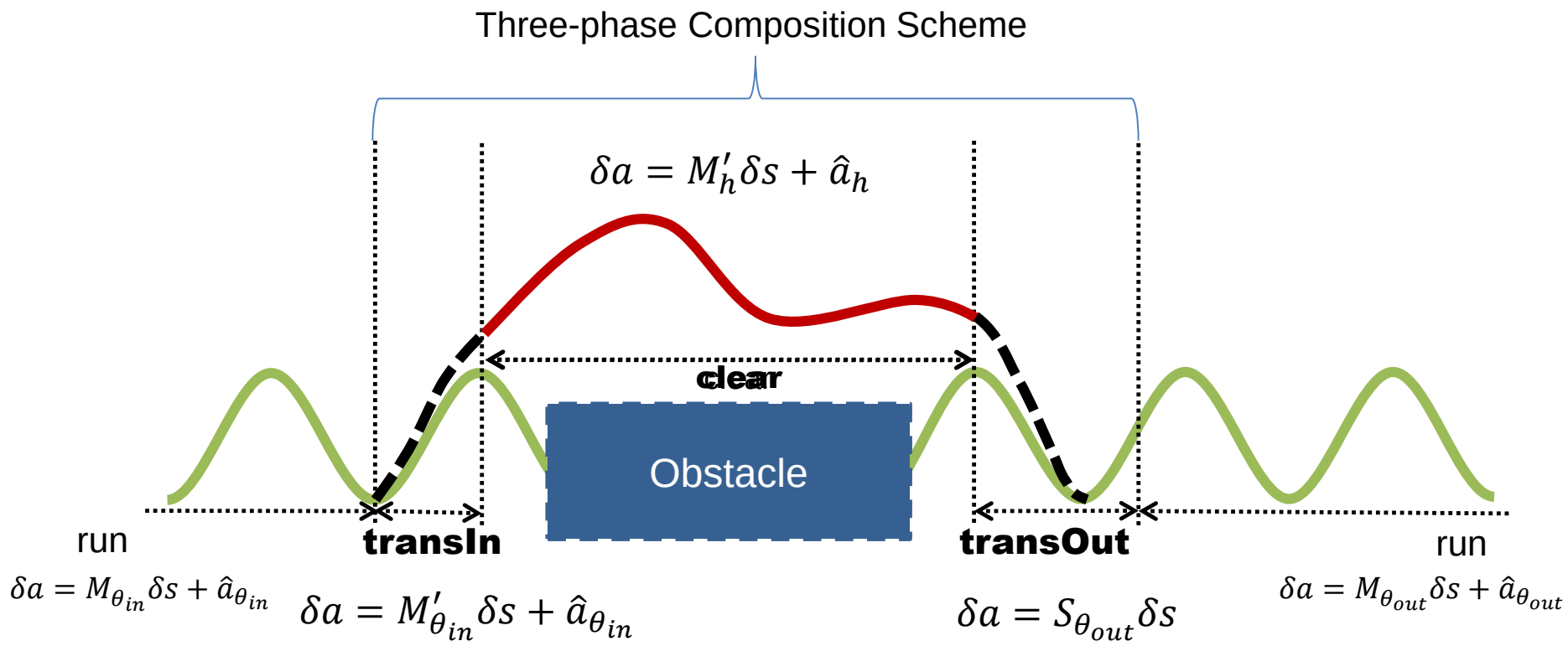


System Overview



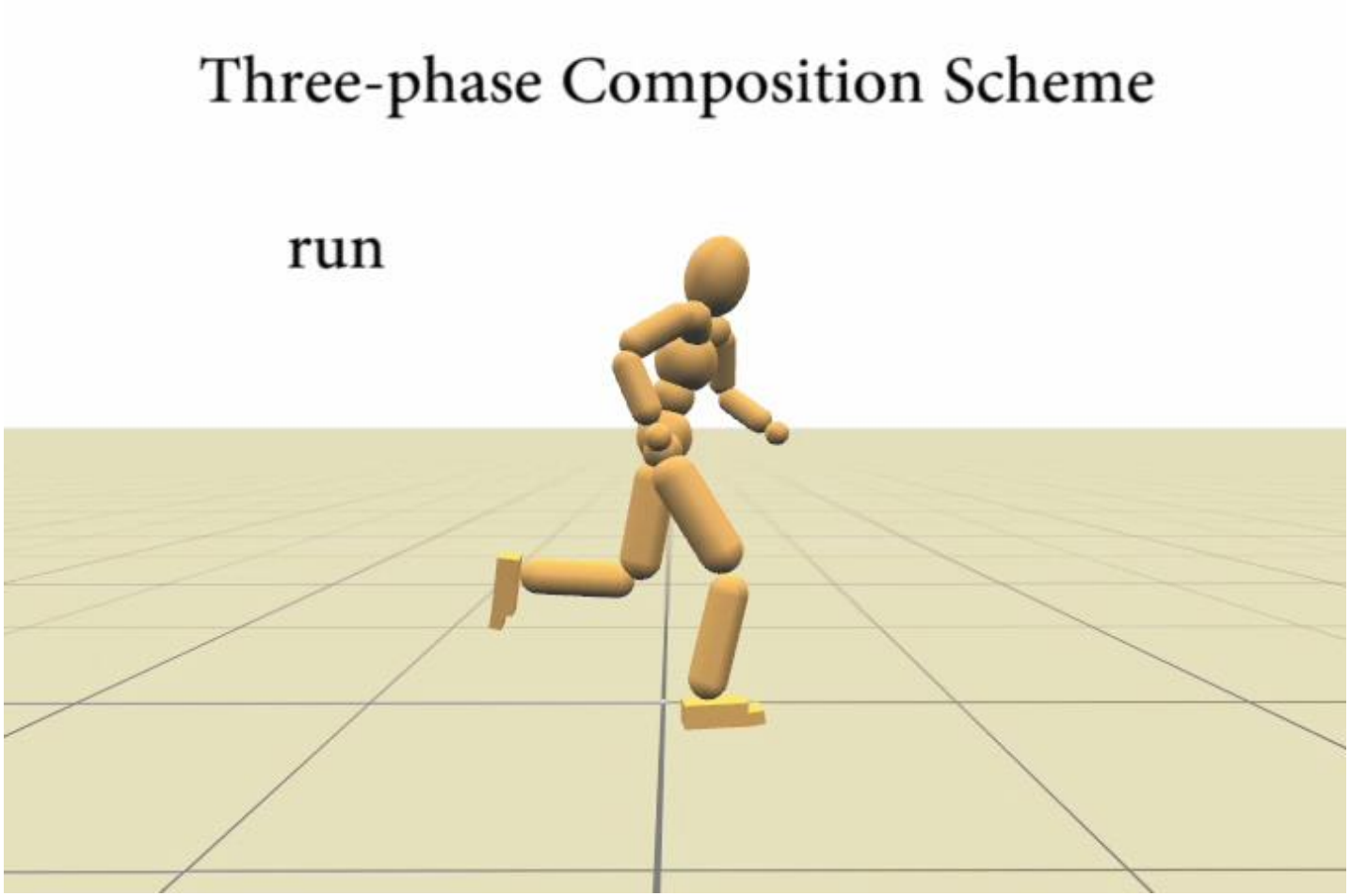
Stage 3: Composition

Three-phase Composition Scheme



Stage 3: Composition

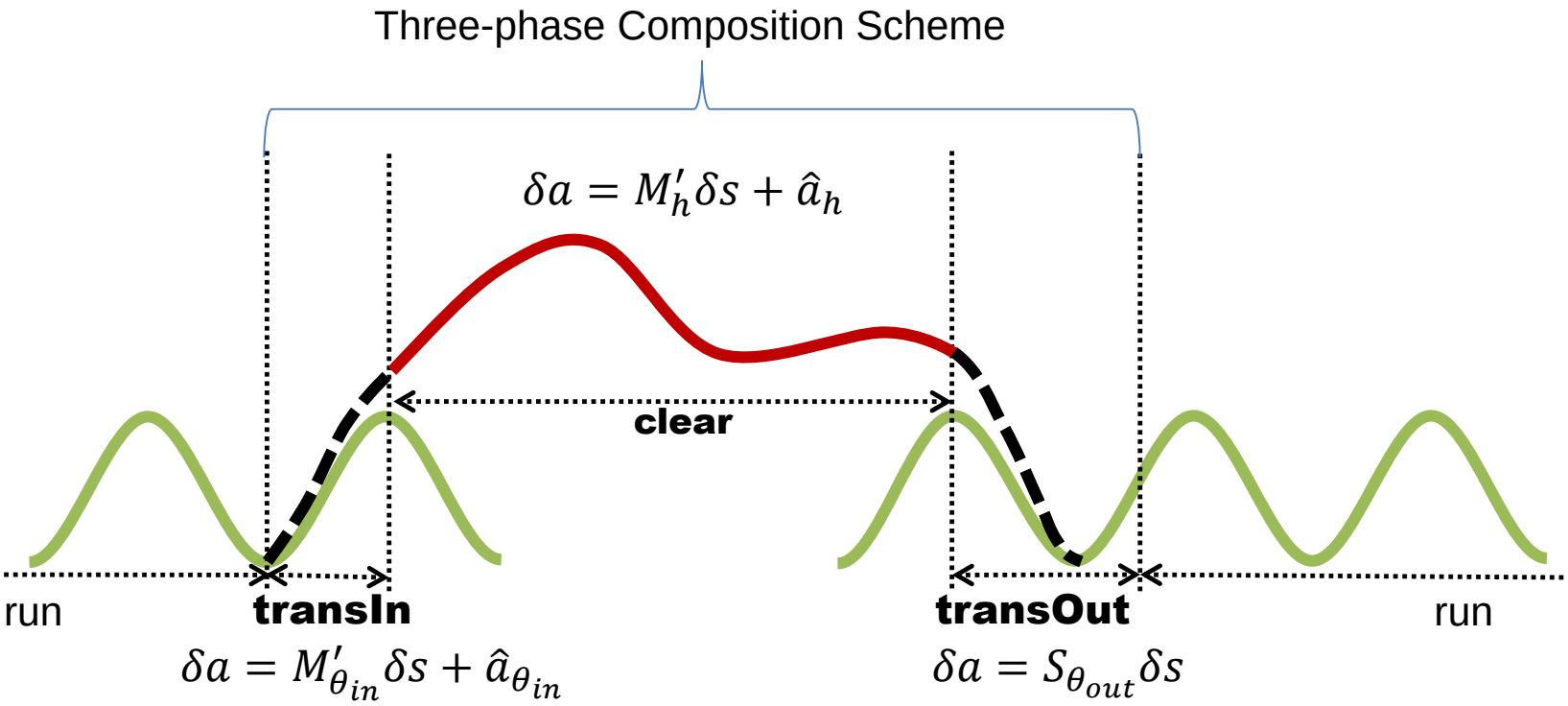
Three-phase Composition Scheme



Stage 3: Composition

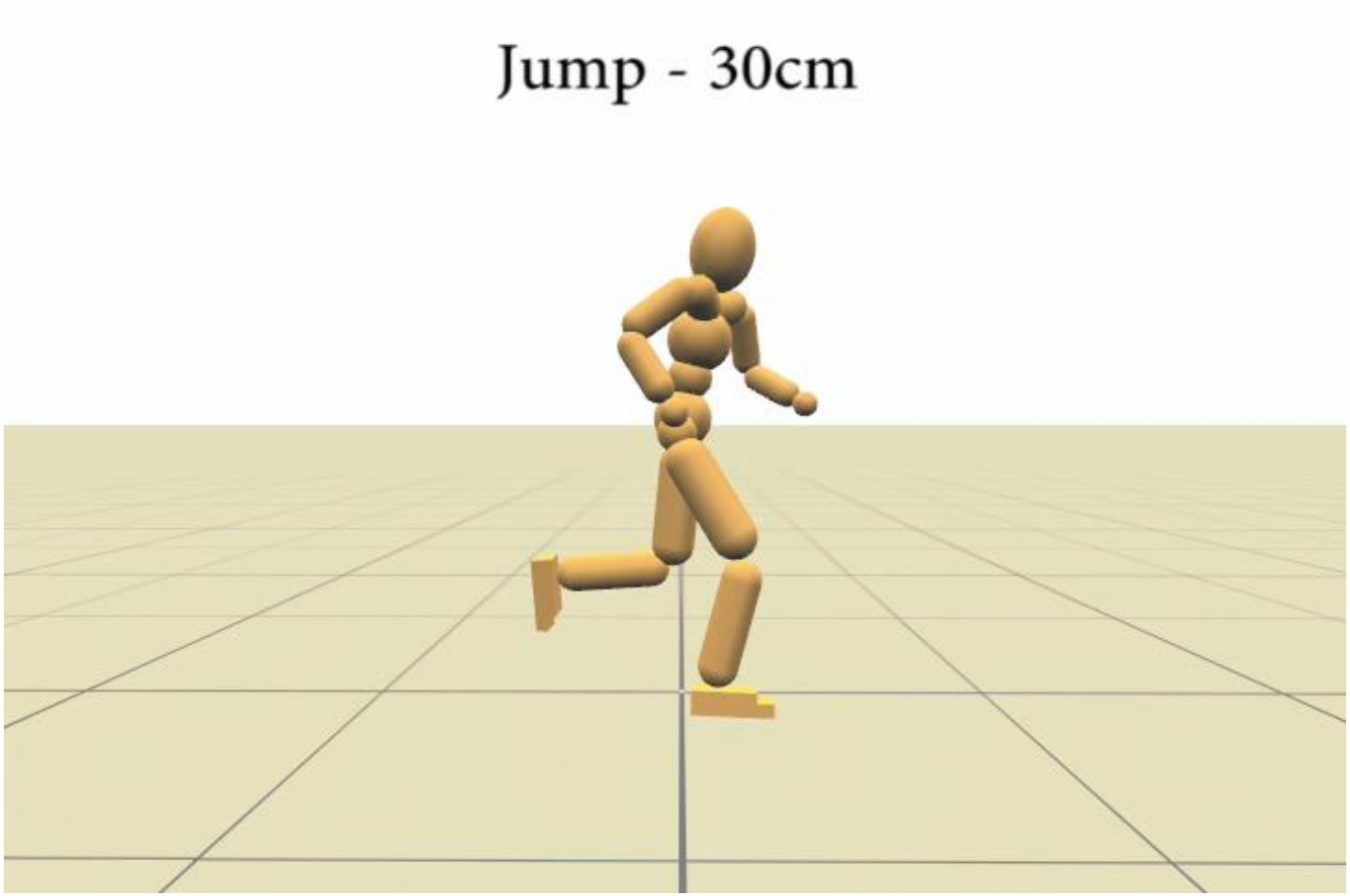
Optimization

- Parameters
  - $\{\theta_{in}, M'_{\theta_{in}}, M'_h, S_{\theta_{out}}\}$

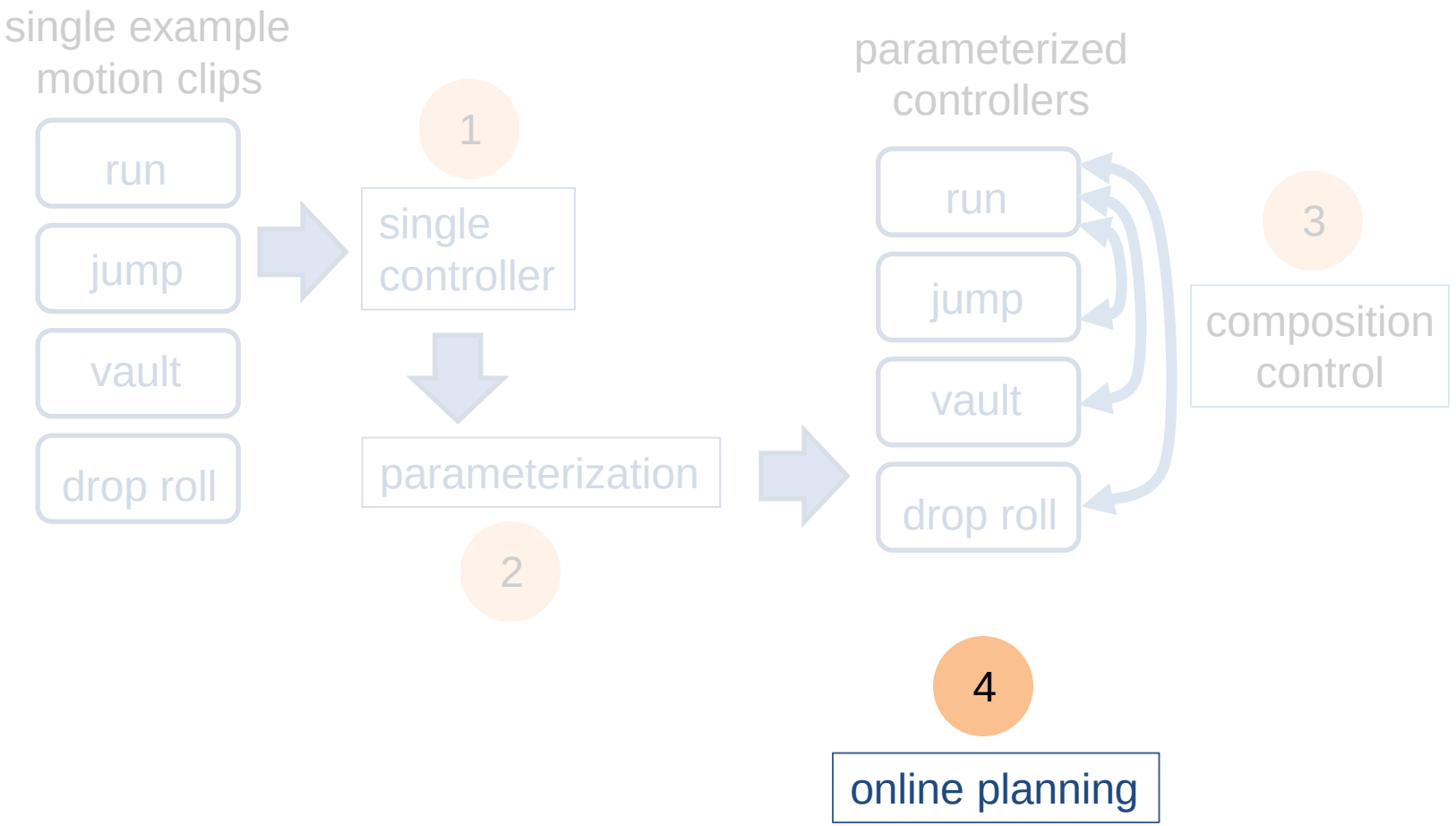


Stage 3: Composition

Results

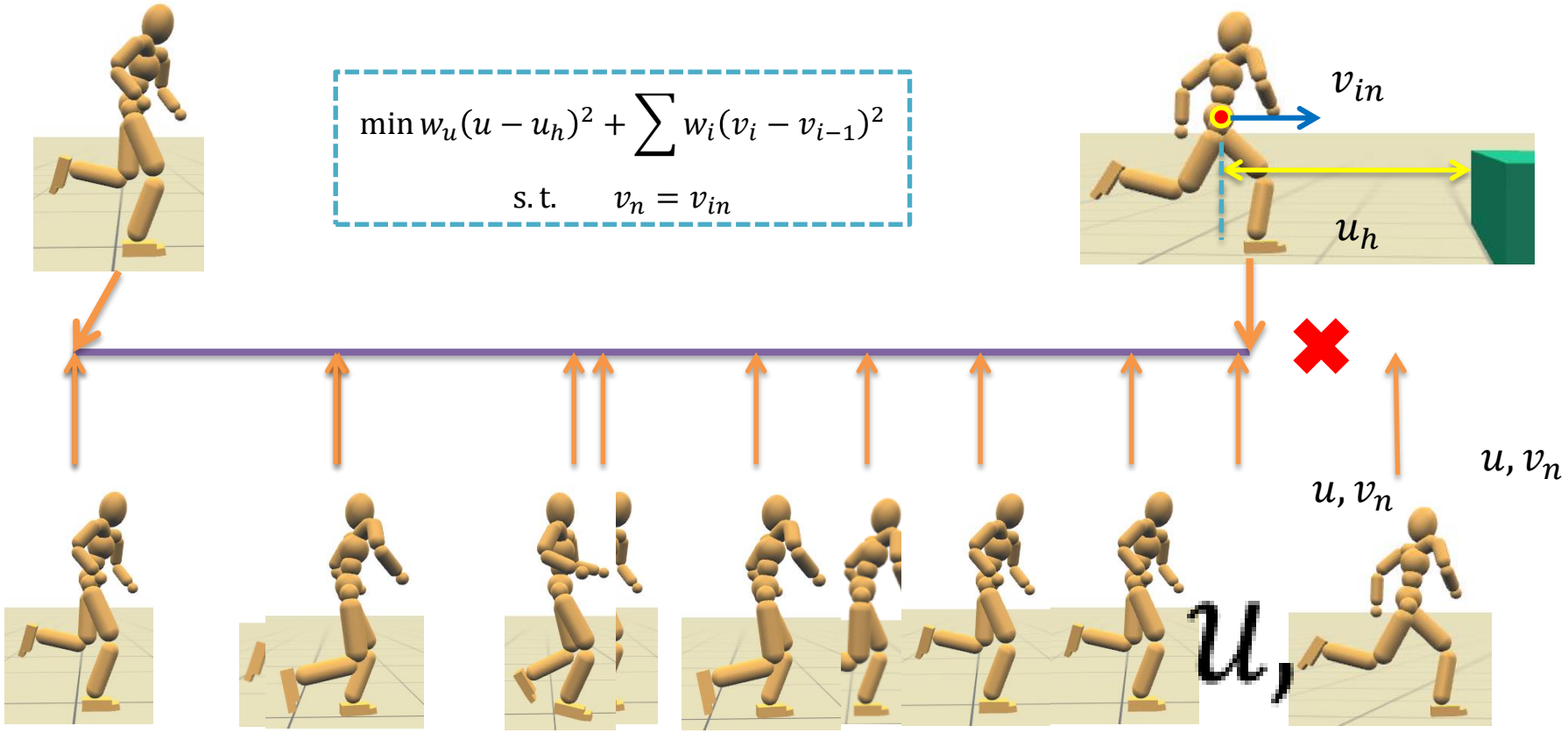


System Overview



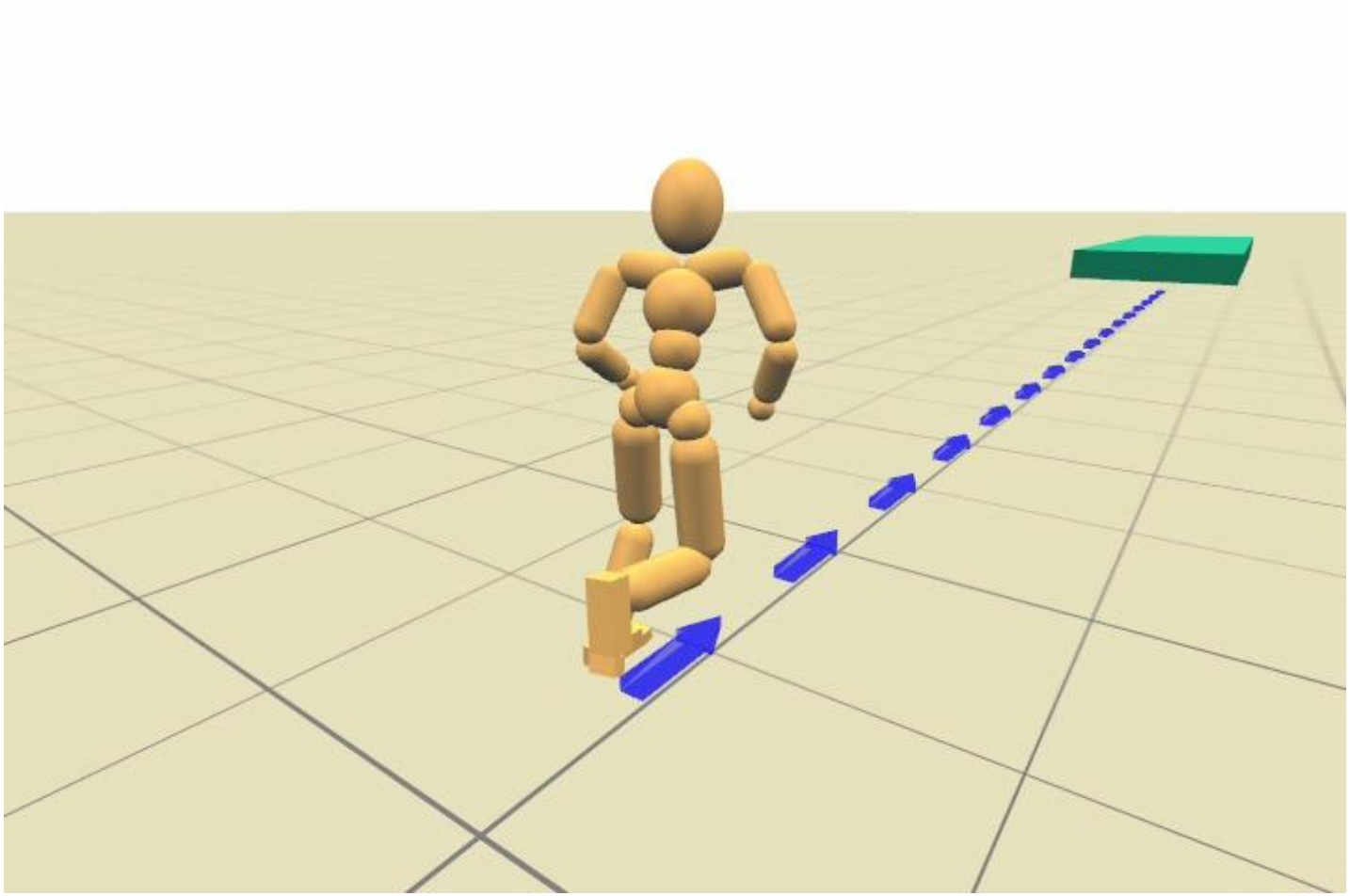
Stage 4: Online Planning

- Step-based kinematic planning



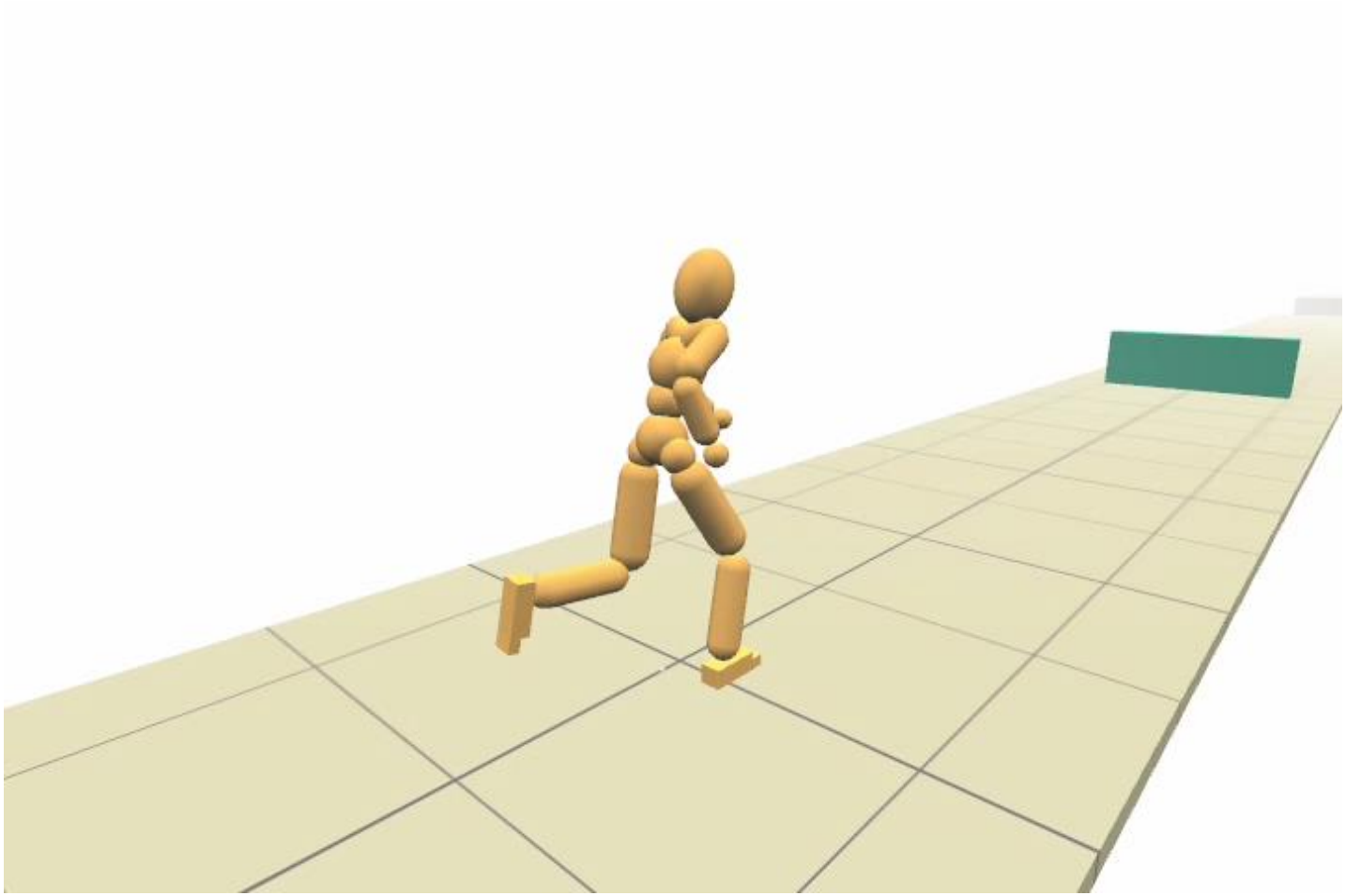
Stage 4: Planning

Results



Results

Terrain Running



Conclusion

- Parkour-style motions
  - Running, jumping, vaulting, drop-rolling
- Complete framework, Realtime synthesis
  - Control construction, parameterization, composition, planning
- Structured optimization scheme

Limitations

- Only partly automated
- Composition can fail
- No arbitrary transitions

Future Work

[*Parkour Memories*, uploaded by 3runTube
http://www.youtube.com/watch?v=24cgnAA6x0I&hd=1]

Thanks