

# Humanoid controller for walking and jumping motions based on mc\_rtc framework

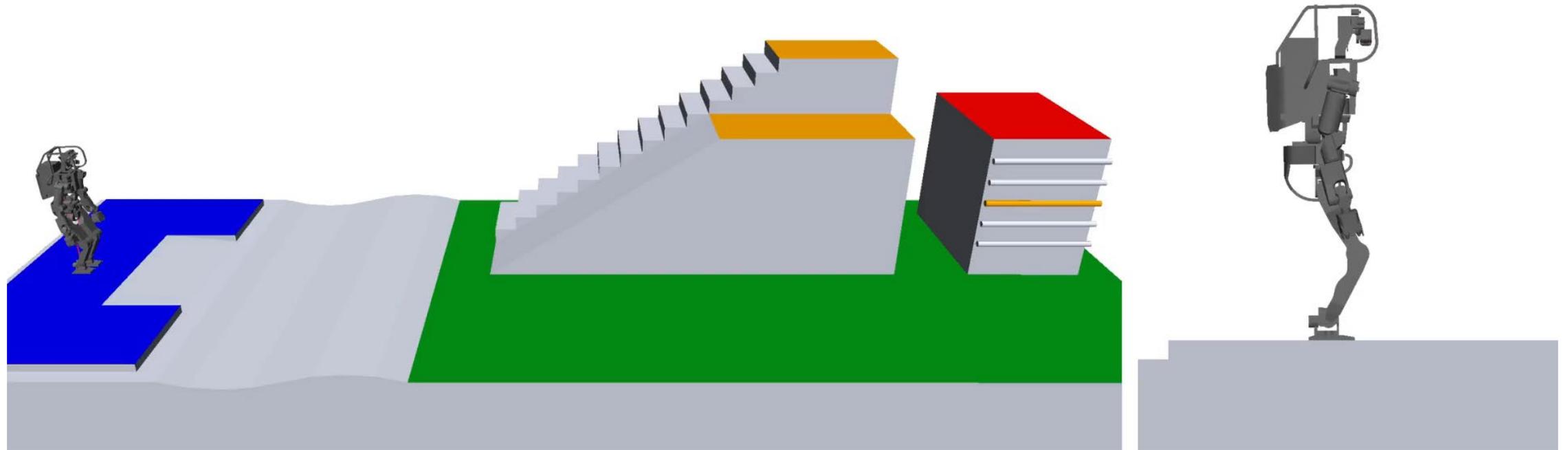
Masaki Murooka  
CNRS-AIST JRL, IRL

November 28<sup>th</sup>, 2022

# Contents

1. HVAC 2021 Results
2. Walking and Jumping Control Methods
3. Control Framework `mc_rtc`
4. Open-source Walking Controller `BaselineWalkingController`

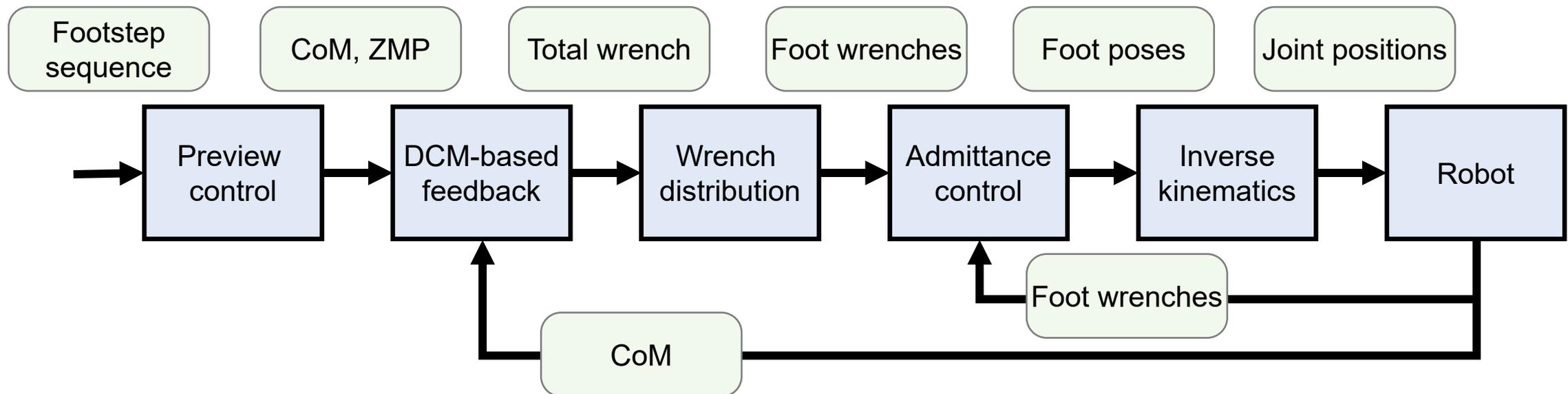
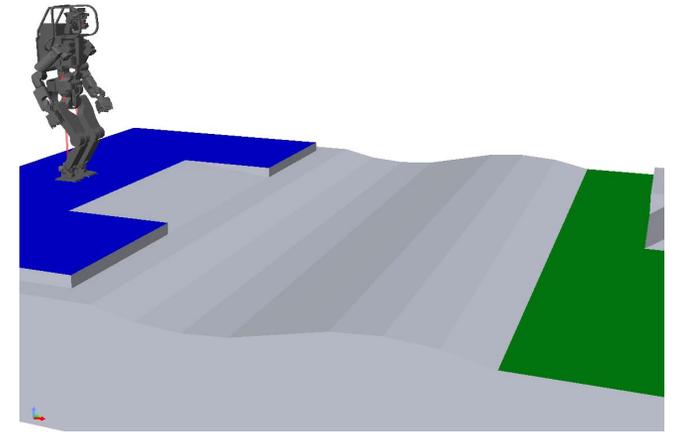
# HVAC 2021 Results



HRP-5P is walking on uneven ground, stairs, and jumping over gaps

# Walking Control Method

- Integration of existing typical components

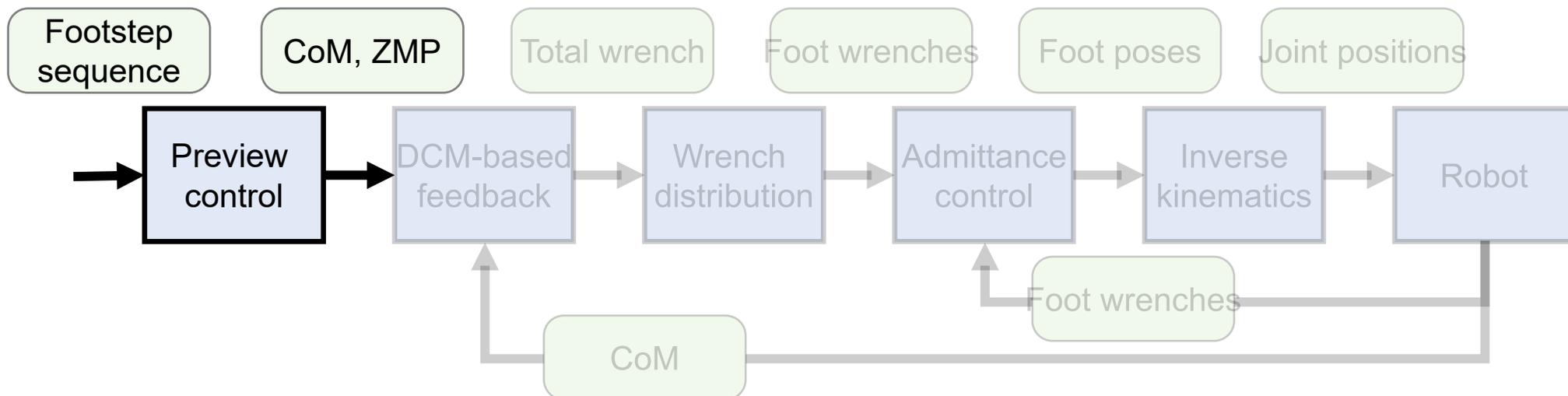


# Preview Control

- Preview control provides closed-form solutions to the following problem [Kajita, ICRA'03]

$$\min_{\ddot{c}oM} \sum_0^{\infty} \| ZMP - ZMP^{ref} \|^2 + \| \ddot{c}oM \|^2 \quad \text{s.t. } LIPM \text{ dynamics}$$

- $ZMP^{ref}$  is determined from a predefined footstep sequence
- Integrating the optimal  $\ddot{c}oM$  yields the  $CoM^{plan}$  and  $ZMP^{plan}$



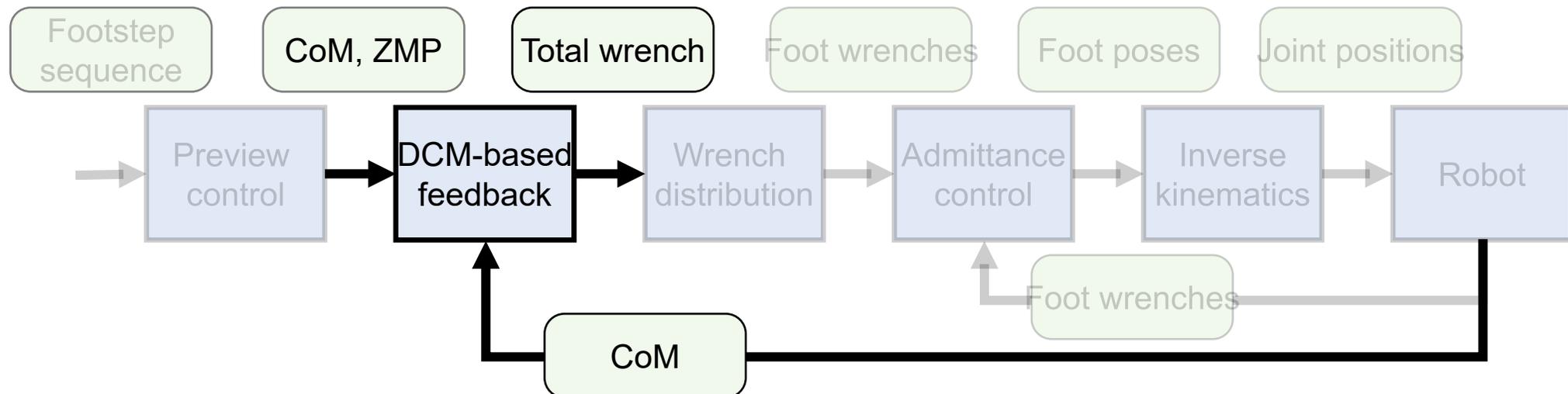
# DCM-based feedback

- Modify ZMP to reduce errors due to disturbances

$$ZMP^{target} = ZMP^{plan} + K (DCM^{actual} - DCM^{plan}) \quad \text{where } DCM = CoM + \dot{CoM} / \omega$$

- The optimal ratio of  $CoM$  and  $\dot{CoM}$  feedback gains leads to  $DCM$  [Sugihara, ICRA'09]
- Convert target ZMP to total wrench

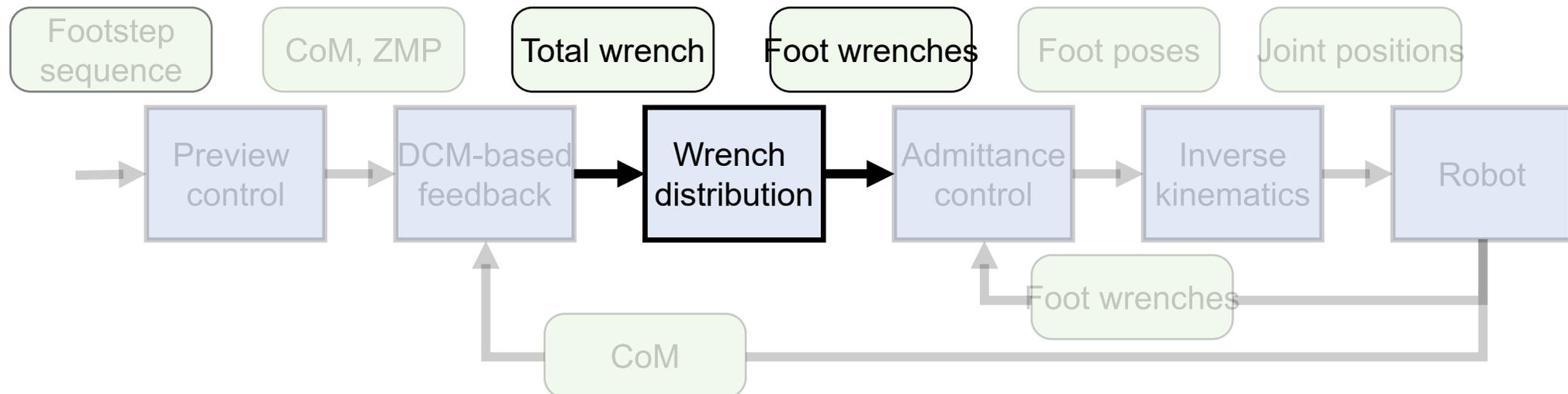
$$Force^{total} = m \omega^2 (CoM - ZMP^{target})$$



# Wrench Distribution to Feet

- Distribute total wrench to each foot under friction constraints
- Formulated as a quadratic programming problem

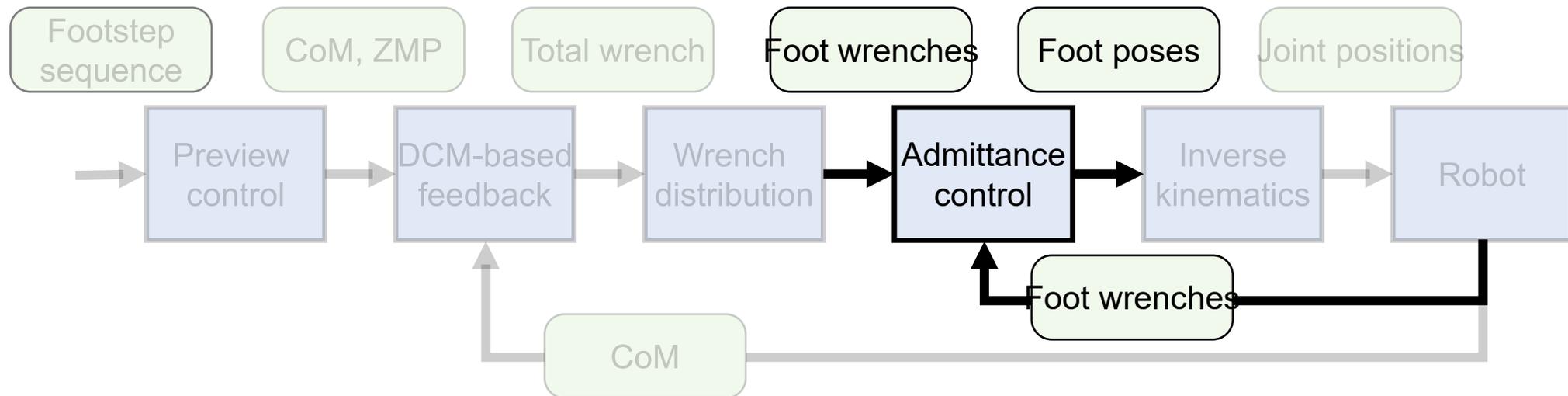
$$\min_{Wrench^{foot}} \left\| Wrench^{total} - \sum_{foot} Wrench^{foot} \right\|^2 \quad \text{s. t.} \quad Wrench^{foot} \in \text{Friction constraint}$$



# Foot Admittance Control

- Modify the foot pose to track target foot wrench

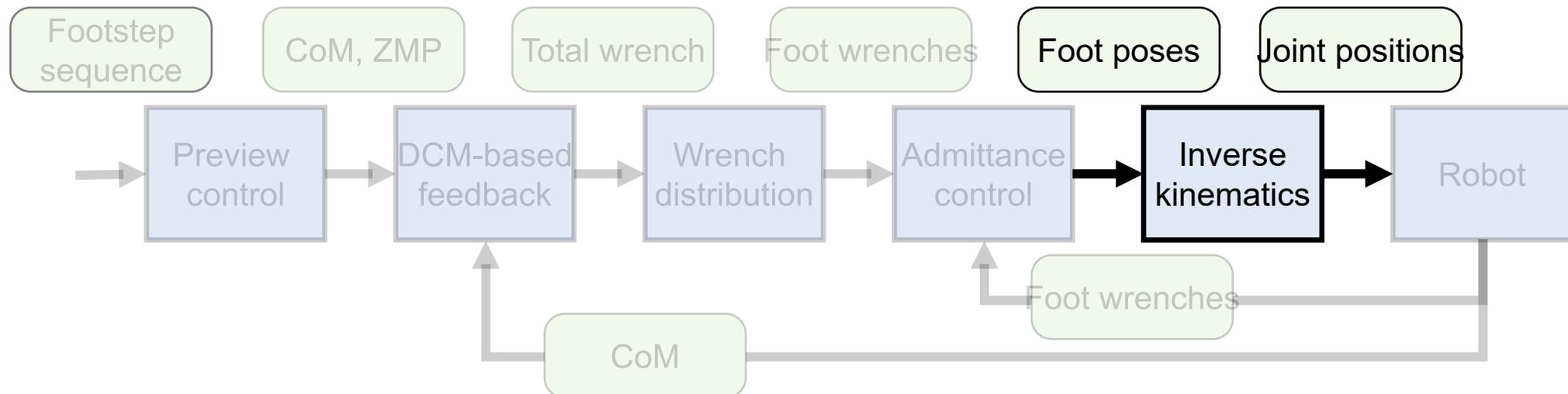
$$Pose^{target} \leftarrow Pose^{target} + \Delta Pose \quad \text{where} \quad \Delta Pose = D (Wrench^{actual} - Wrench^{target})$$



# Inverse Kinematics

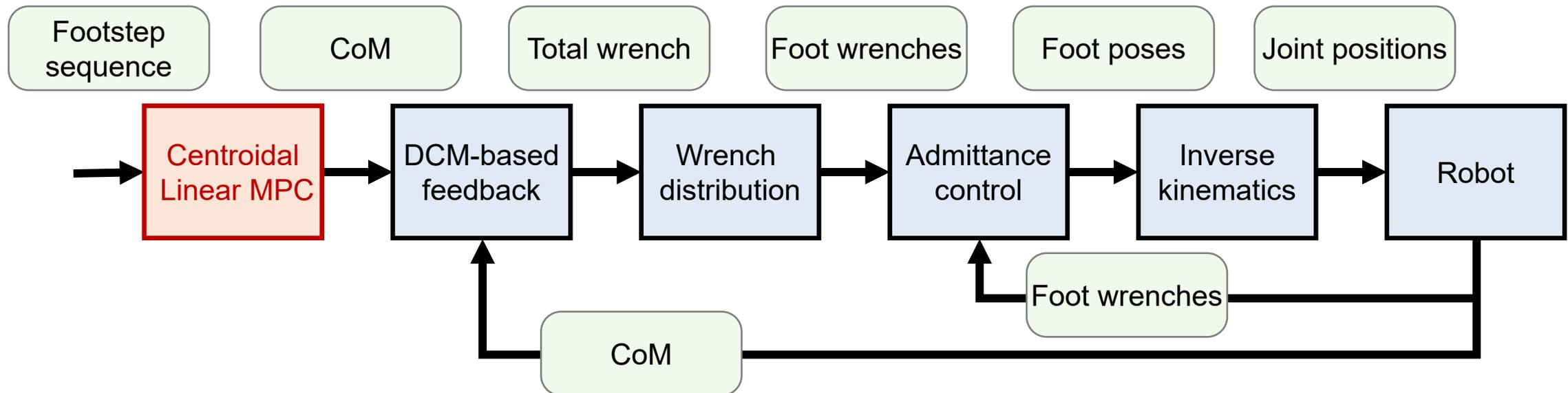
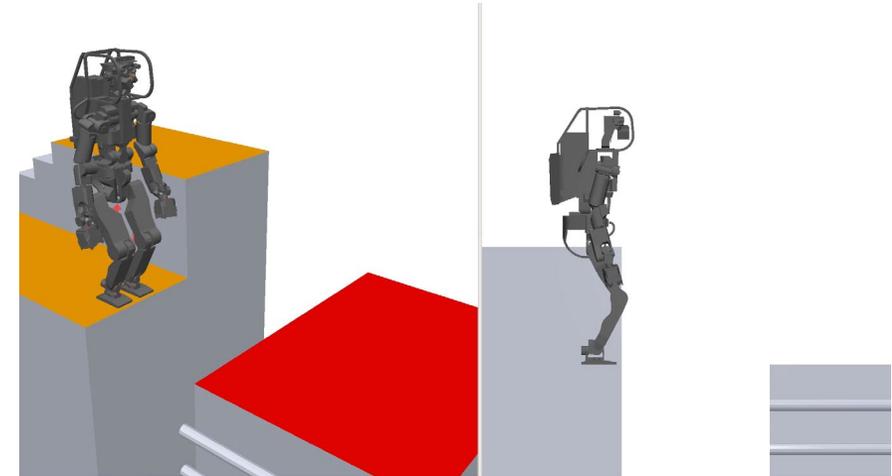
- Calculate joint positions that satisfy target CoM and foot poses
- Formulated as a quadratic programming problem

$$\theta \leftarrow \theta + \Delta\theta^* \quad \text{where} \quad \Delta\theta^* = \underset{\Delta\theta}{\operatorname{argmin}} \sum_{\text{task}} \left\| J_{\text{task}} \Delta\theta - \left( v_{\text{task}}^{\text{target}} - v_{\text{task}}^{\text{current}} \right) \right\|^2$$



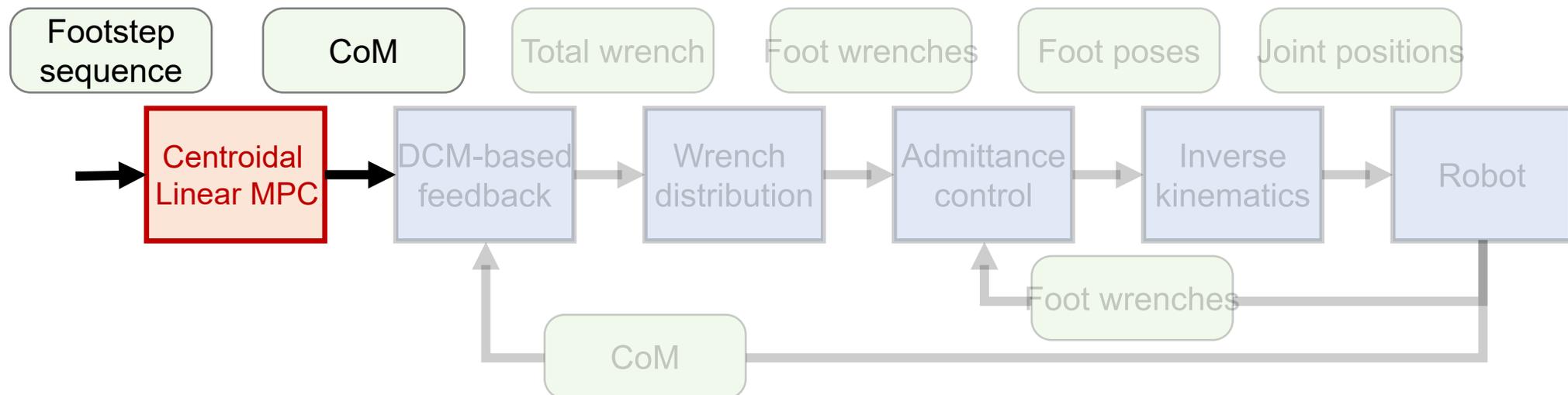
# Jumping Control Method

- The only difference from the walking control is the CoM planning
- Apply linear MPC based on centroidal dynamics



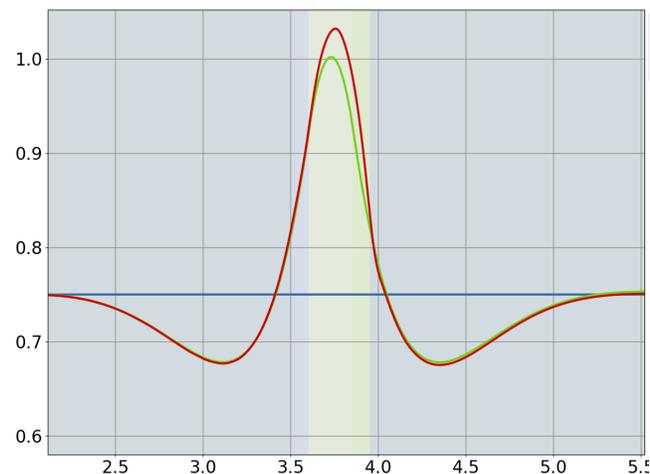
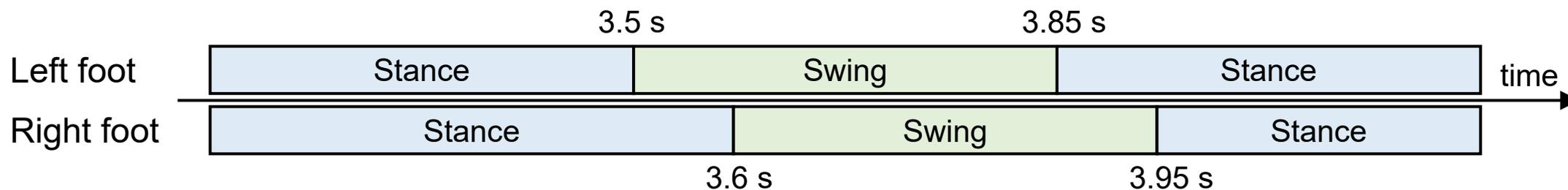
# Centroidal Linear MPC

- Since centroidal dynamics is nonlinear, linear MPC cannot be applied as is
- Plan CoM vertical motion only (linear dynamics) with MPC
- Plan full centroidal motion (linear dynamics with known vertical motion) with MPC [Nagasaka, RoboSym'12 (in Japanese), Audren, IROS'14]
- Linear MPC is formulated as quadratic programming problem

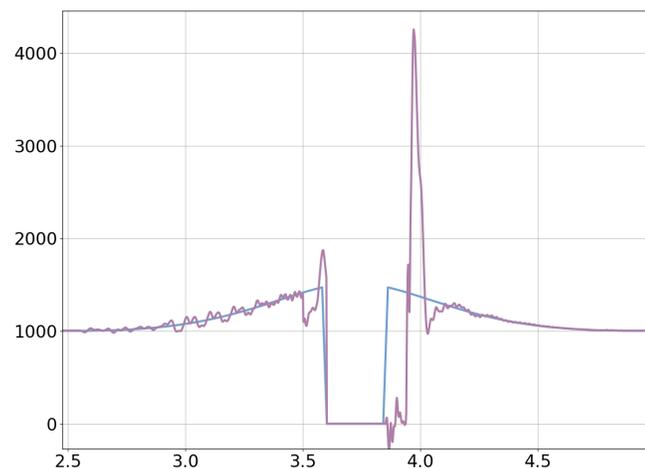


# Centroidal Linear MPC

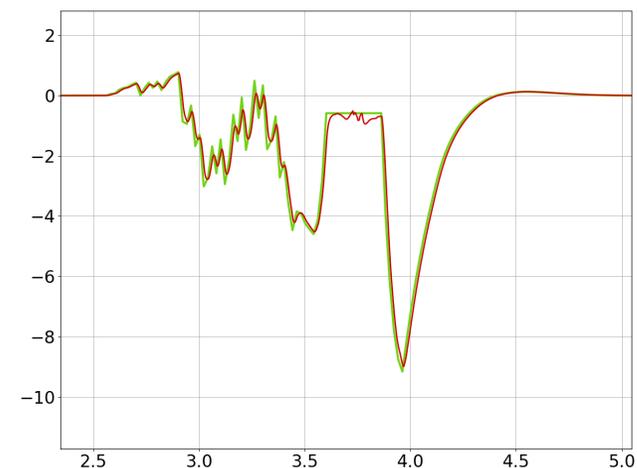
- Landing 1.4m forward with a jump of 0.25 s floating phase
- Allow up to 4 times the limits for velocity and torque at knee and hip pitch joints



Vertical motion



Vertical force



Pitch angular momentum

# Control Framework mc\_rtc

[https://jrl-umi3218.github.io/mc\\_rtc/](https://jrl-umi3218.github.io/mc_rtc/)

- CNRS-AIST JRL develops mc\_rtc
  - Kinematics and dynamics algorithms
  - Robot modeling
  - Logging and utility functions
  - Documentation
  - CI environments

The screenshot shows the 'Robots' page of the mc\_rtc website. The page title is 'Robots' and it includes a navigation menu with links to 'Interfaces', 'Robots', 'Tutorials', 'API documentation', 'JSON/YAML documentation', 'Online demonstration', and language options 'EN | JP'. The main content area lists several robots, each with a small image, a title, a 'Description' link, and a 'Module' link.

Robot	Description Link	Module Link
HRP-2-Kai	<a href="#">hrp2_drc</a>	<a href="#">mc-hrp2</a>
HRP-4-LIRMM	<a href="#">hrp4</a>	<a href="#">mc-hrp4</a>
HRP-4J	<a href="#">hrp4j_description</a>	<a href="#">mc_hrp4j</a>
HRP-4CR	<a href="#">hrp4cr_description</a>	<a href="#">mc_hrp4cr</a>
HRP-5P	<a href="#">hrp5_p_description</a>	<a href="#">mc_hrp5_p</a>
Nao	<a href="#">nao_description</a>	<a href="#">mc_nao</a>
Pepper	<a href="#">pepper_description</a>	
Panda		<a href="#">mc_panda</a>

Tutorial at Humanoids 2022  
 mc\_rtc: An Application Framework for Robotics  
 November 28<sup>th</sup>, 2022. Room B3

# Open-source Walking Controller BaselineWalkingController

- ✓ Tested on real robots
- ✓ Dynamics simulation tests is run on CI environments
- ✓ Docker image is released with the latest version of the controller
- ✓ Easy to switch between various methods for CoM planning
- ✓ Easy to switch between “closed-loop MPC” and “open-loop MPC + stabilizer”

 [isri-aist / BaselineWalkingController](#) Public

<https://github.com/isri-aist/BaselineWalkingController>

# Real Robot Testing of BaselineWalkingController



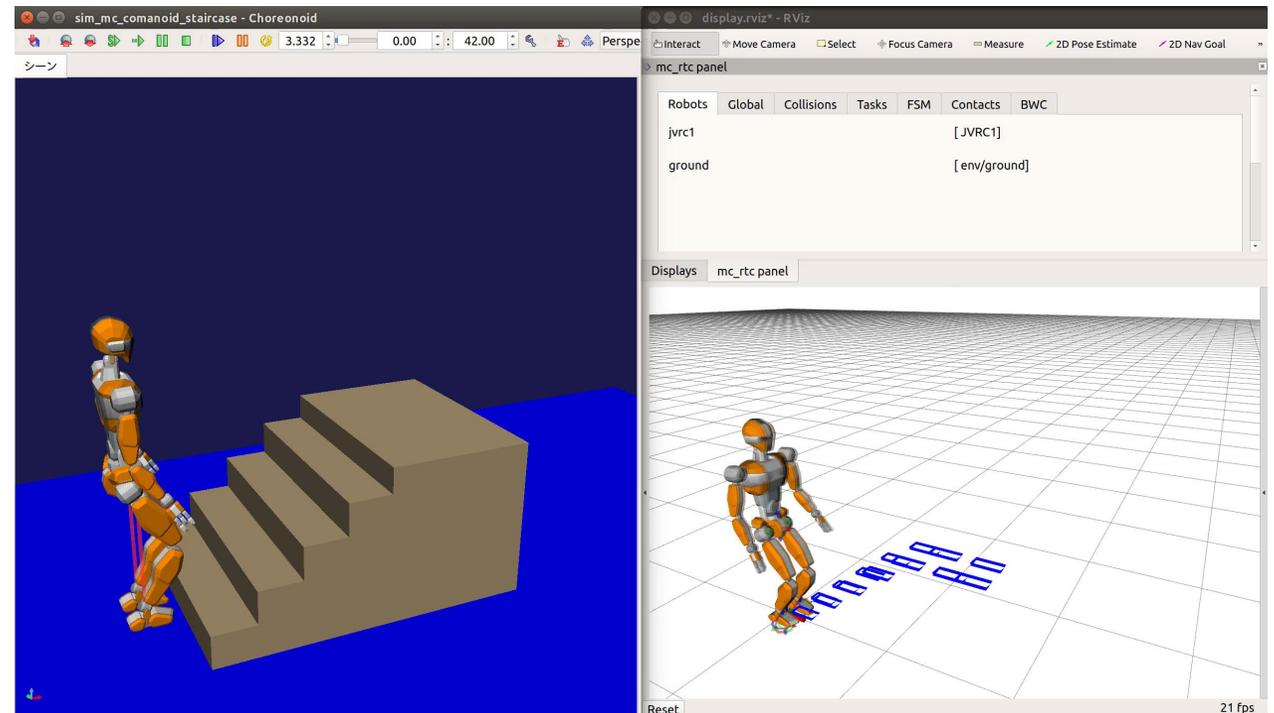
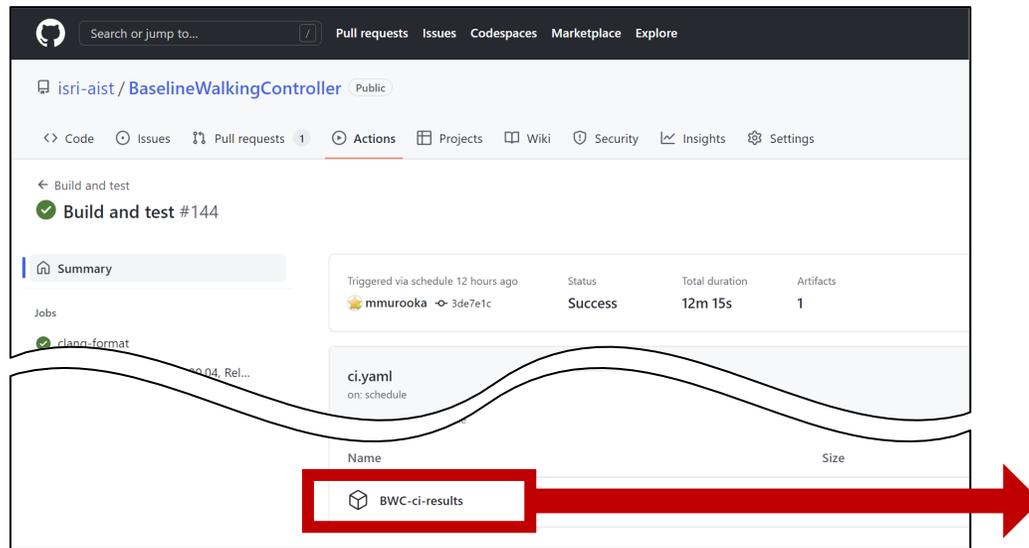
HRP-5P (Test on Nov.11th)



HRP-2KAI (Test on Nov. 7th)

# Simulation Tests on CI environments

- Choreonoid tests are performed with every update on CI (GitHub Actions)



# Automatic Release of Docker Images

- Docker images are released on CI for every update

isri-aist / BaselineWalkingController Public

Code Issues Pull requests 1 Actions Projects Wiki Security Insights Settings

master 3 branches 0 tags

mmurooka Allow to configure deltaTransLimit via GUI. 3de7e1c 2 days ago 104 commits

File	Description	Commit Time
.clang-format-common.sh	Add scripts for clang-format.	4 months ago
.clang-format-fix.sh	Add scripts for clang-format.	4 months ago
.gitignore	Generate mc_rtc.yaml	4 months ago
CMakeLists.txt	Delete unnecessary find_packages in cmake.	3 months ago
LICENSE	Create LICENSE	4 months ago
README.md	[README] add technical details.	4 days ago

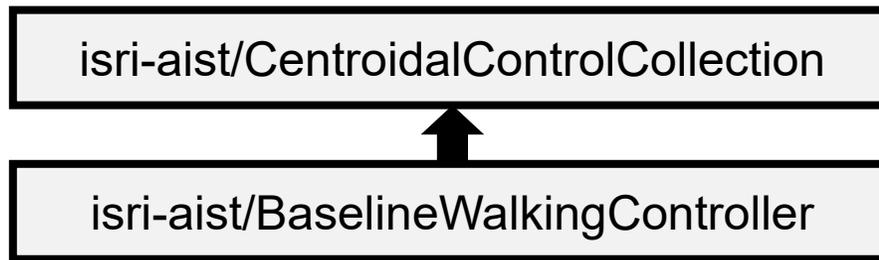
Packages 3

- baseline\_walking\_controller\_ci
- baseline\_walking\_controller
- baseline\_walking\_controller\_hvac2022

```
$ docker pull ghcr.io/isri-aist/baseline_walking_controller:latest
$ docker run --gpus all --rm -it
  --env="DISPLAY" --volume="/tmp/.X11-unix:/tmp/.X11-unix:rw"
  ghcr.io/isri-aist/baseline_walking_controller:latest ./walk_on_stairs.bash
```

# Easy switching of CoM planning methods

- You can try the various CoM trajectory generation methods implemented in CentroidalControlCollection



- List of methods
  - PreviewControlZmp [Kajita, ICRA'03]
  - DdpZmp [Feng, Journal of field robotics '15]
  - DcmTracking [Englsberger, IROS'13]
  - FootGuidedControl [Sugihara, IROS'17, Kojio, IROS'19]
  - LinearMpcZmp [Wieber, Humanoids'06]
  - IntrinsicallyStableMpc [Scianca, Humanoids'16]
  - SingularPreviewControlZmp [Urata, Humanoids'11]

isri-aist / CentroidalControlCollection

Code Issues Pull requests Actions Projects Wiki Security Insights Settings

master 2 branches 0 tags

Go to file Add file Code

PreviewControlZmp

- Shuji Kajita, et al. Biped walking pattern generation by using preview control of zero-moment point. ICRA, 2003.

```

$ rosrn centroidal_control_collection TestPreviewControlZmp
$ rosrn centroidal_control_collection plotTestZmpBasedMethodResults.py --method PreviewControlZmp
    
```

PreviewControlZmp-X

PreviewControlZmp-Y

DdpZmp

- S Feng, et al. Optimization-based full body control for the darpa robotics challenge. Journal of field robotics, 2015.

```

$ rosrn centroidal_control_collection TestDdpZmp
$ rosrn centroidal_control_collection plotTestZmpBasedMethodResults.py --method DdpZmp
    
```

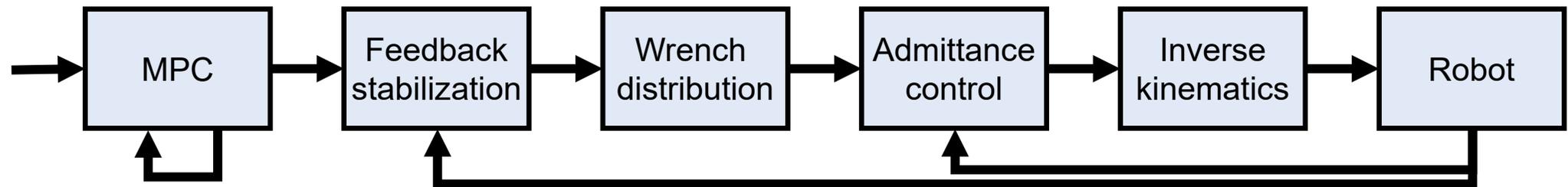
DdpZmp-X

DdpZmp-Y

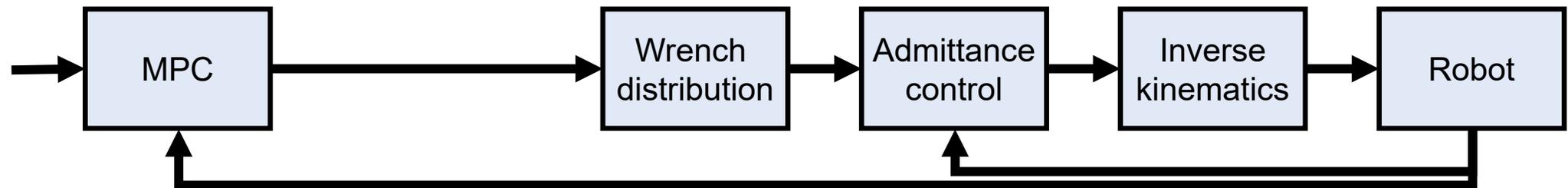
# Easy switching of open/closed-loop MPC

- Open-loop MPC is often used implicitly in bipedal walking
- Differences between the two MPC schemes have not been fully discussed

## Open-loop MPC + Feedback stabilization



## Closed-loop MPC



# Concluding remarks

- For beginners (and experts, of course), it is very helpful to be able to refer to a set of formulas in a paper and source code
- Let's share them in open discussion forums such as Humanoid Virtual Athletics Challenge