Whole Body Interface and Simulink Toolbox

Francesco Romano

VVV15 - Sestri Levante

Why WholeBody Interface?

- Abstract whole body controller from implementation details
- Simplify code

Examples: interface with the robot

WBI

```
Vector positions =  // …
robot.setControlMode(wbi::CTRL_MODE_POS);
robot.setControlReference(positions);
```

YARP

```
Vector positions =  // …
IPositionControl *positionControl; //one for each robot part
IControlMode *controlMode; //one for each robot part
//for each robot part
controlMode->setPositionMode()
positionControl->positionMove(positions);
```

Examples: interface with the robot

Another robot platform

??? => learn it and change the code!!

Examples: dynamic library

WBI

```
robot.computeMassMatrix(jointPositions, w_H_b,
massMatrix);

robot.computeGeneralizedBiasForces(jointPositions,
w_H_b, jointVelocities, baseVelocity,
gravityUnitVector, generalizedBiasForces);
```

Other dynamic libraries?

iDynTree, Pinocchio, RBDL, Robotran

Main elements of the WBI

- Actuators: write commands to the robot.
- State: obtain information on the robot
- Model: dynamic and kinematic model of the robot.

Available implementations: YARP-iDynTree

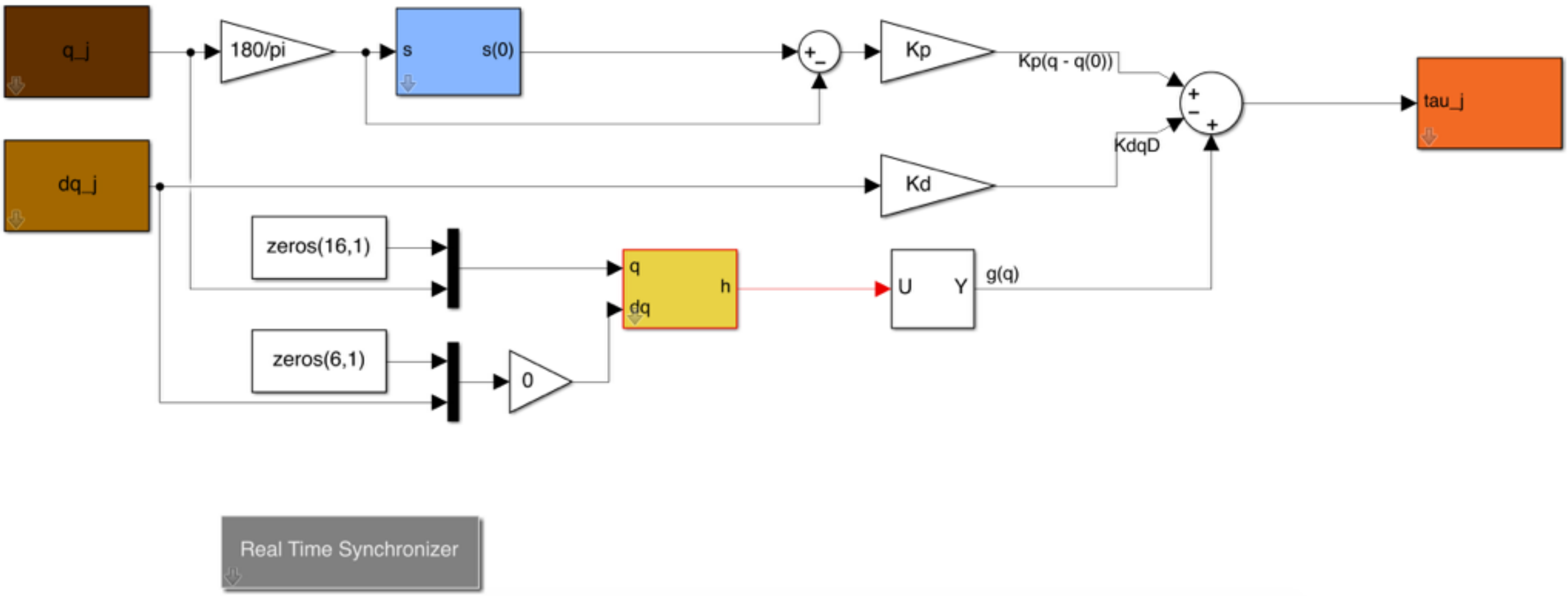
Simulink WBI-Toolbox

- “Rapid prototyping” of controllers to speed up final implementation on the real platform.

- Exploitation of Simulink and MATLAB toolboxes, e.g. System Identification, Optimization, Simulink Control Design, Stateflow, etc.

- Higher level of abstraction to the Whole-Body Interface.

Simulink WBI-Toolbox



Links

- *WBI:* https://github.com/robotology-playground/wholebodyinterface

- *YARP-iDynTree implementation:* https://github.com/robotology-playground/yarp-wholebodyinterface

- *WBI-Toolbox:* https://github.com/robotology-playground/WBI-Toolbox