DETERMINISTIC6G

6G Emulation for Exoskeleton Control

Emilio Trigili, PhD

Assistant Professor

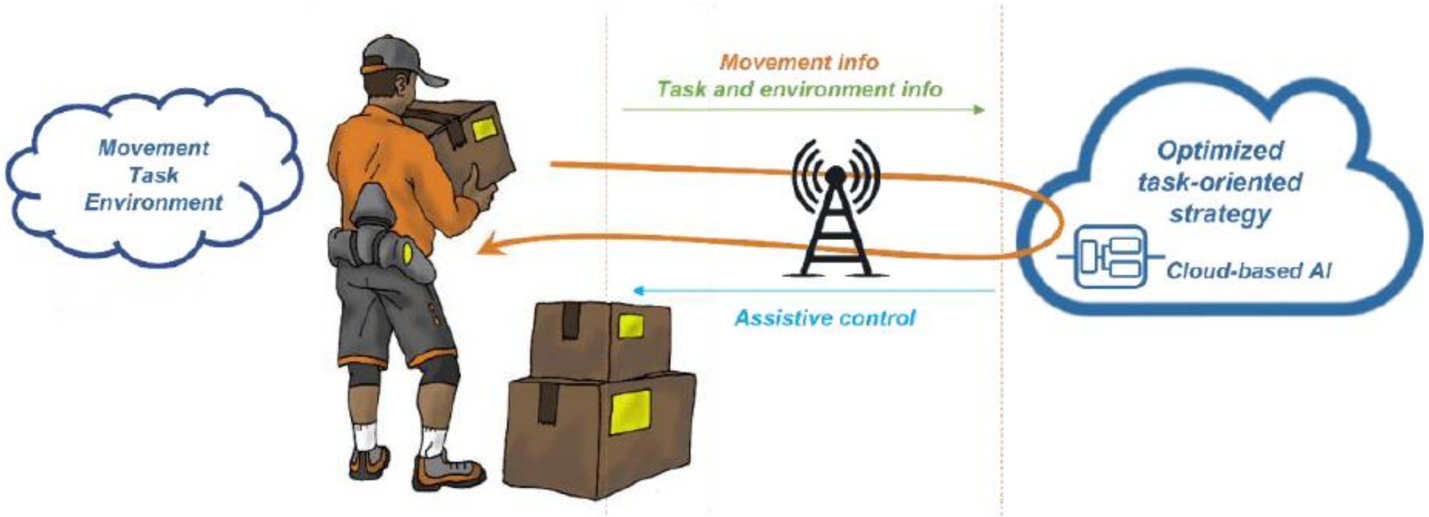
emilio.trigili@santannapisa.it

Sant'Anna
Scuola Universitaria Superiore Pisa

DETERMINISTIC6G has received funding from the European Union's Horizon Europe programme under grant agreement No 101096504.

«Cloud is the limit»: the target

- The target user of the use case is a worker of a logistic warehouse with a lumbar active occupational exoskeleton

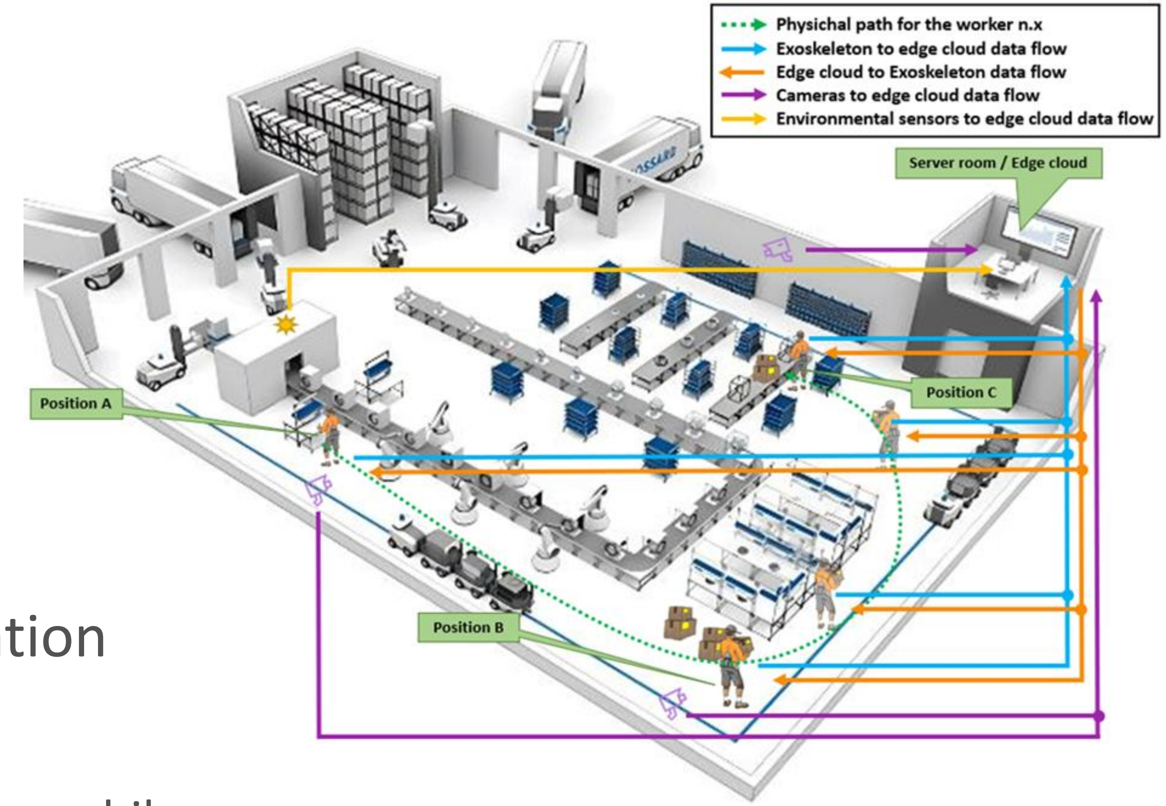


- 6G network: Allows the user to log into the cloud-based system and to receive customized exoskeleton configurations

- Cloud AI-based assistant: collect information from different sensors and to determine task-oriented assistive strategies.

«Cloud is the limit»: the target

- In the cloud is possible to create a digital twin of the system to remotely monitor movements and provide real-time feedback and suggestions.
- This allows to have multiple exoskeletons and multiple digital twins sharing the perceived data acting independently.
- As well, it is possible to integrate data coming from other sensors (i.e. cameras) in the "Smart Factory"



Enabling technology: Offload computation

↓

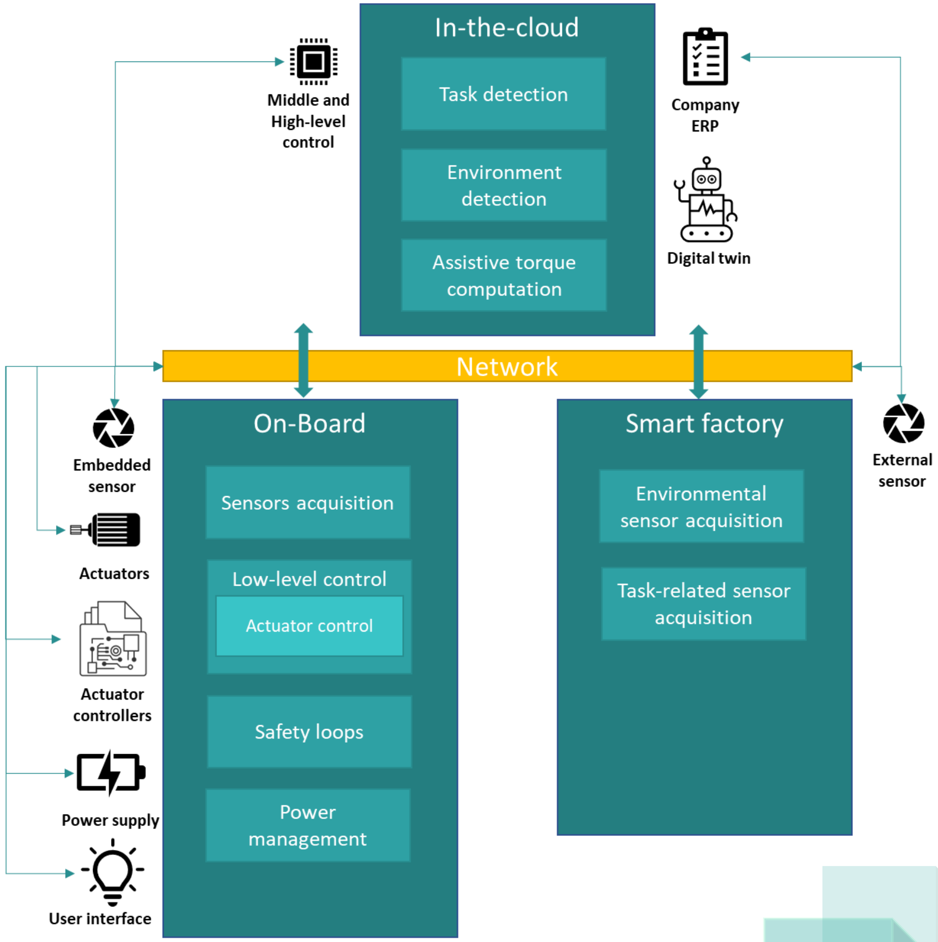
The low-level control is embedded in the exoskeleton, while the middle and high-level control are delocalized in the cloud.

Why Offload Computation?

- The exoskeleton includes an embedded processor that runs a low-level controller, which translates the desired joint assistance torque into current and voltage setpoints to drive the motors accordingly.

- The middle-level and high-level controllers are offloaded to the cloud relying on 6G-based wireless network, as well as the hardware components on-board the exoskeleton, thus reducing power consumption.

- A delocalized controller, monitoring, and real-time processing of a huge amount of information gathered from multiple subsystems will be made possible by enabling off-board complex AI assistive strategies.



Why Offload Computation?

Offloading computation offers several advantages:

- Reduced cost (one/few powerful machines in the cloud replace a lot of powerful hardware on many exoskeletons)
- Reduced size and weight (smaller batteries)
- Reduced power consumption
- Full integration with the smart factory digital ecosystem
- Adaptive support based on the user’s or environment’s inputs

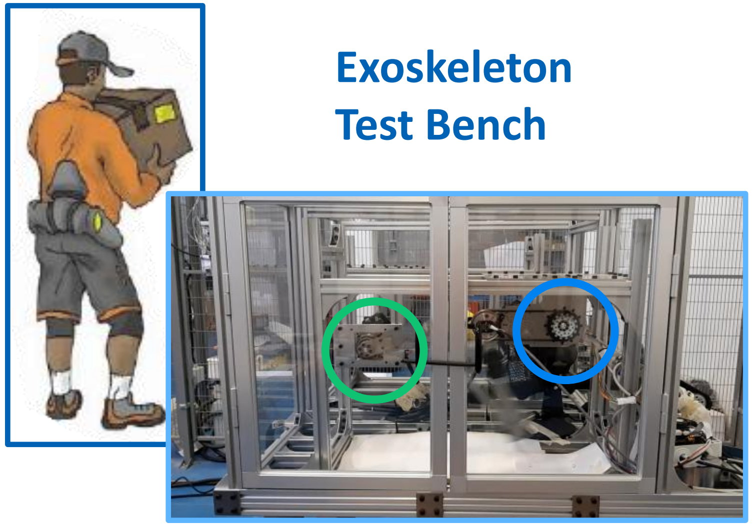
- However, as the device architecture becomes less embedded and based on external computation, it must rely on highly dependable communication to ensure safety and consistency of operation

Understand the effect of offload computation and network delay into exoskeleton control

Objective and overall architecture

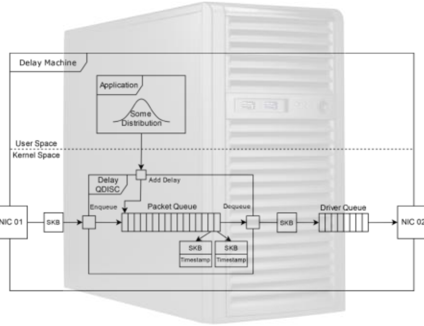
- Emulate the effect of a 6G network in an occupational exoskeleton

Exoskeleton Test Bench



Wired Ethernet

Linux Machine
6G Network emulator



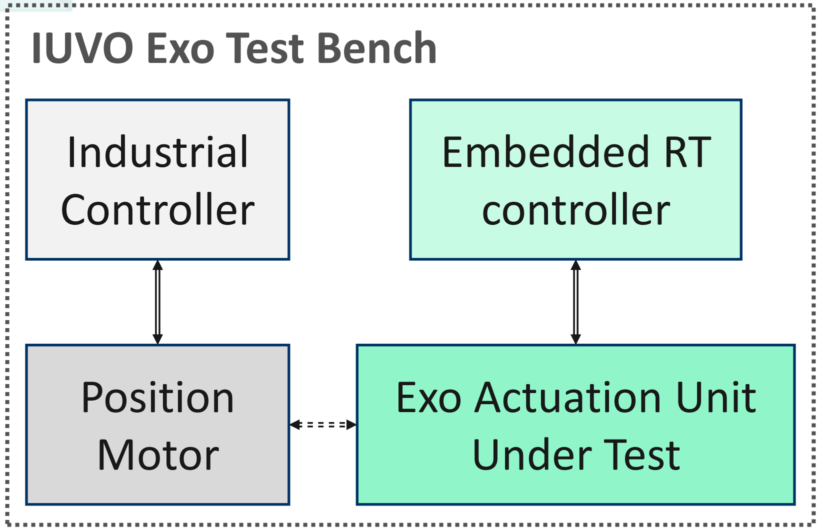
Wired Ethernet

Edge Cloud Server & App

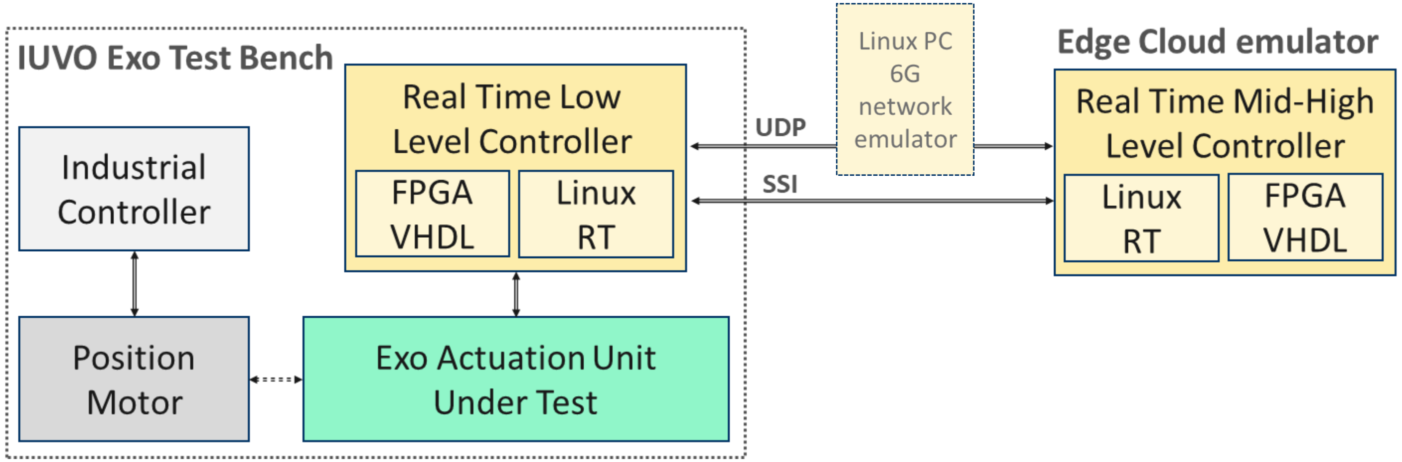
- IUVO test bench to evaluate reconfigurable and repetitive work cycle performances for exoskeleton actuation unit
- University of Stuttgart 6G Network emulator tool (more details tomorrow!)

Emulator integration:
Exo Test Bench, 6G Network Emulator

Test Bench, basic configuration:

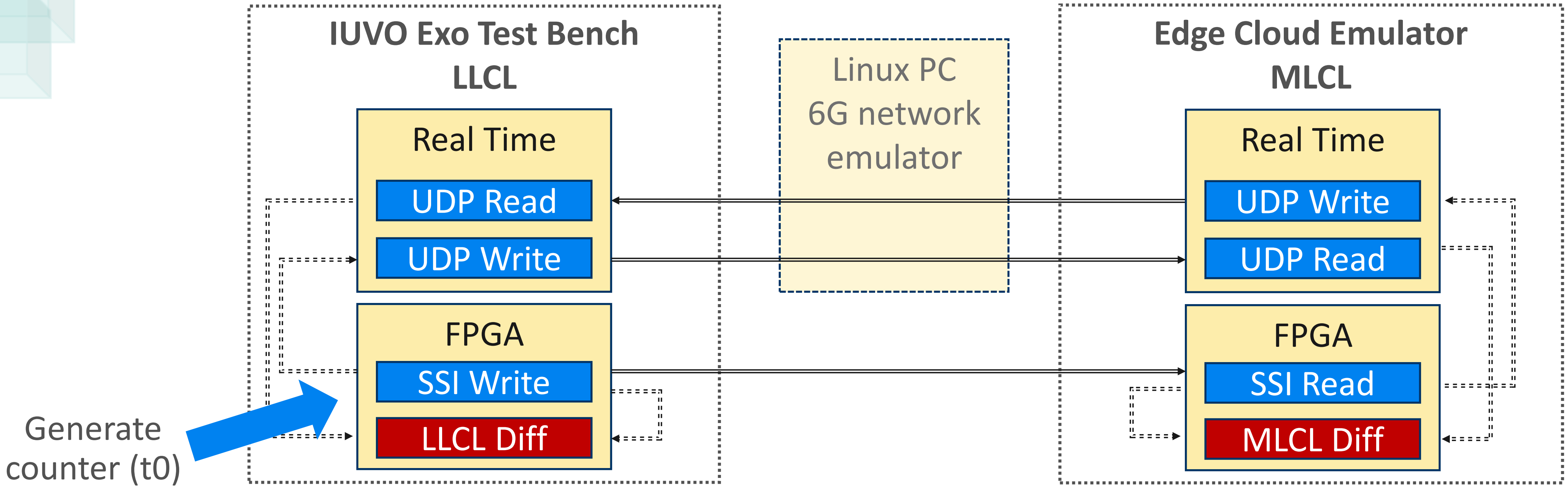


Updated Test Bench with DET6G emulator:



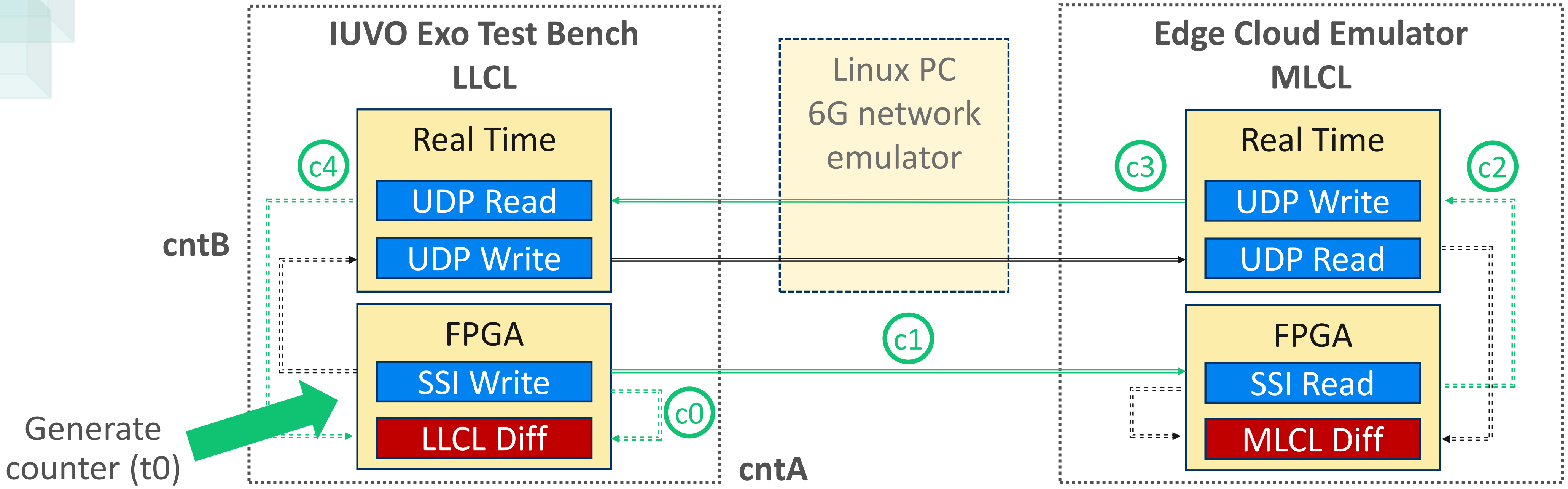
- ❑ UDP connection between the Exoskeleton Test Bench (ETB) and the "Edge Cloud Emulator" (ECE)
- ❑ Linux PC installed on the UDP line and it can be set as "transparent forwarder" or as 6G network emulator
- ❑ Synchronous Serial Interface parallel line works in real-time, and it synchronizes the ETB and ECB, and it permits diagnostic of the overall architecture

Network Architecture



- Investigate how and how much the network introduces misalignments between the low-level control (LLCL) and the high-middle-level control (MLCL)
- Idea: share a counter generated at the FPGA level of the LLCL through two different pathways to see the time needed to complete the roundtrip and highlight the network contribution

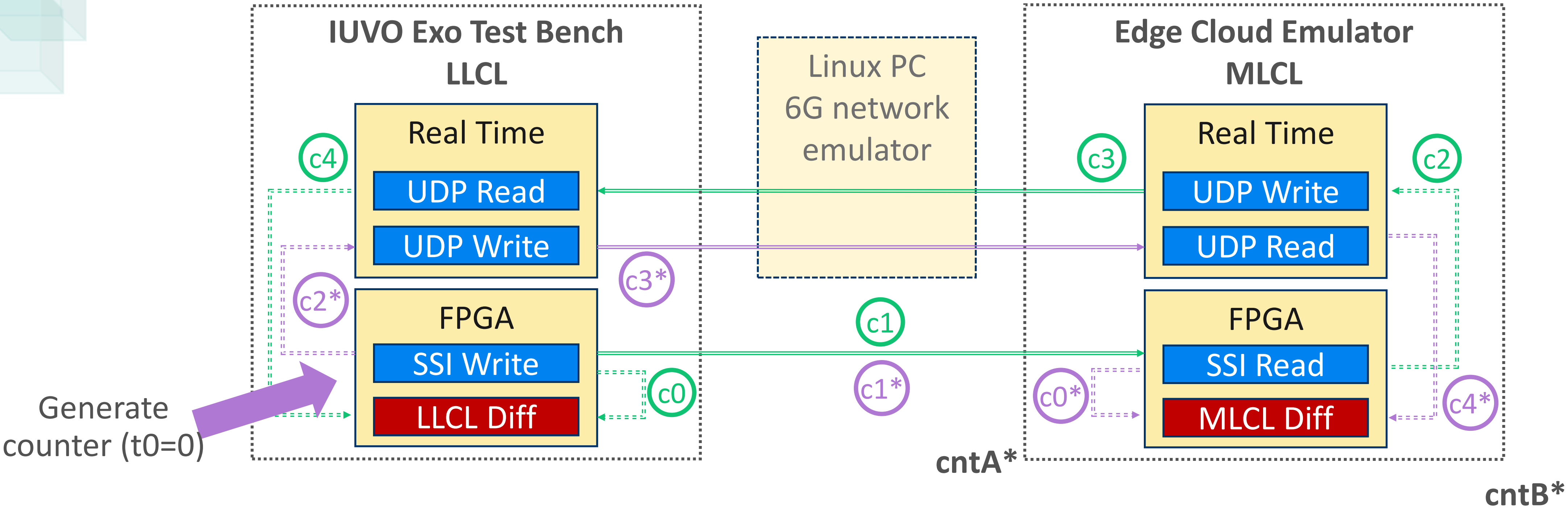
Network Architecture



LLCL Routes

cntA => c0
cntB => c1 + c2 + c3 + c4

Network Architecture



MLCL Routes

cntA* => c1* + c0*
cntB* => c2* + c3* + c4*

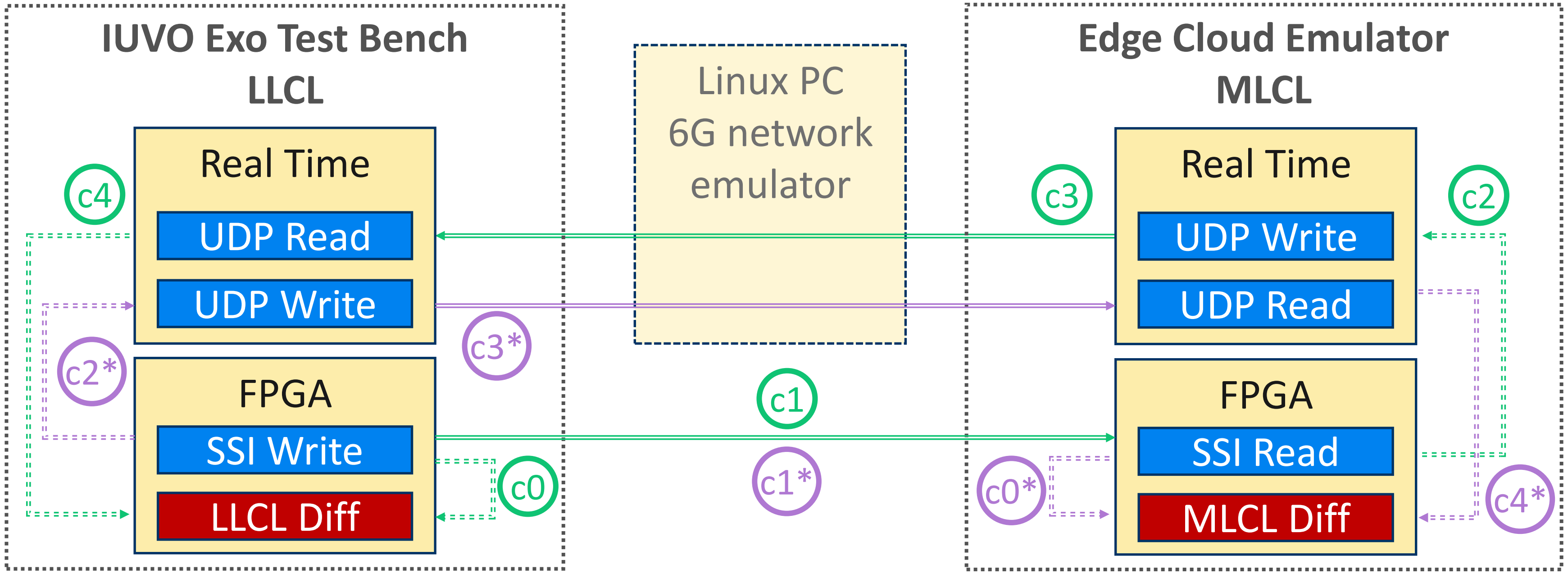
Network Architecture

t0: instant where the FPGA SSI counter begin send.
The variable «Emulated Counter [cnt]» in the LLCL is updated at t0

Scan Engine ($c2$, $c4$, $c2^*$, $c4^*$):

- User Defined Variable (UDV-UI16)
- Boolean Trigger

n: number of RT loops (at 100 Hz) to wait to receive the counter



LLCL Diff = cntA – cntB = n (updated every 10 ms)

$c0: C_{LV_{FPGA\to FPGA}} = 500\ us$ because diff is at 2kHz

$c1: C_{SSI_{LL\to ML}} = 18 * \frac{1}{5\ \text{kHz}} = 3,6\ ms$

$c2: C_{SE_{FPGA\to RT}} = 500\ us + 50\ us$

$c3: C_{UDP_{ML\to LL}} = ?$

$c4: C_{SE_{RT\to FPGA}} = 500\ us$

MLCL Diff = cntA* - cntB* = n^* (updated every 10 ms)

$c0^*: C_{LV_{FPGA\to FPGA}} = 500\ us$ because diff is at 2kHz

$c2^*: C_{SE_{FPGA\to RT}} = 500\ us$

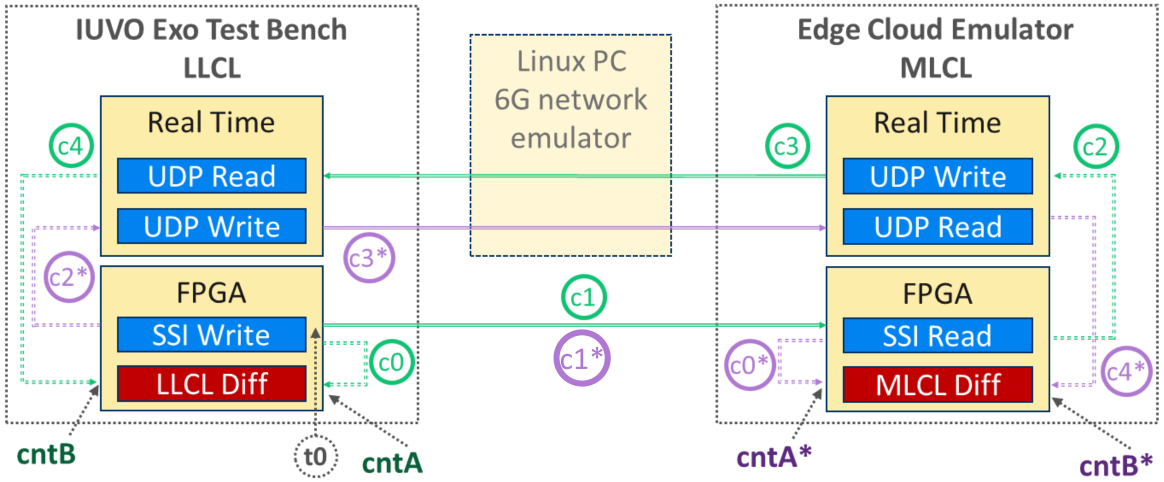
$c3^*: C_{UDP_{LL\to ML}} = ?$

$c4^*: C_{SE_{RT\to FPGA}} = 500\ us$

$c1^* = c1$

Emulator network architecture, delay study

DETERMINISTIC6G



LLCL Diff ➞ $n * 10\ ms = (cnt_A - cnt_B) * 10\ ms\ = -c_0 + (c_1 + c_2 + c_3 + c_4)$

$(4{,}15\ ms + c3) = n * 10\ ms$ ➞ $c3 = (n * 10\ -\ 4{,}15)\ ms$

MLCL Diff ➞ $n^* * 10\ ms = (cnt_{A^*} - cnt_{B^*}) * 10\ ms\ = -(c_{0^*} + c_{1^*}) + (c_{2^*} + c_{3^*} + c_{4^*})$

$-\ 3{,}1\ ms +\ c3^* = n^* * 10\ ms$ ➞ $c3^* = (n^* * 10 + 3{,}1)\ ms$

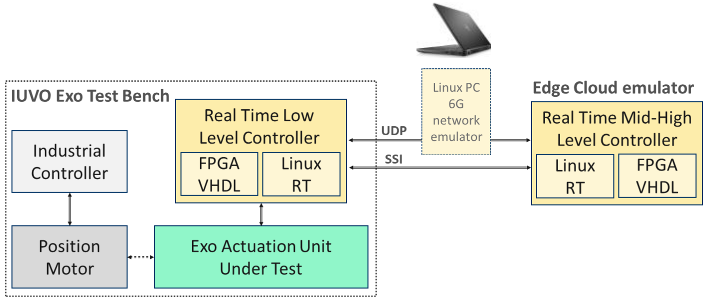
Notes:

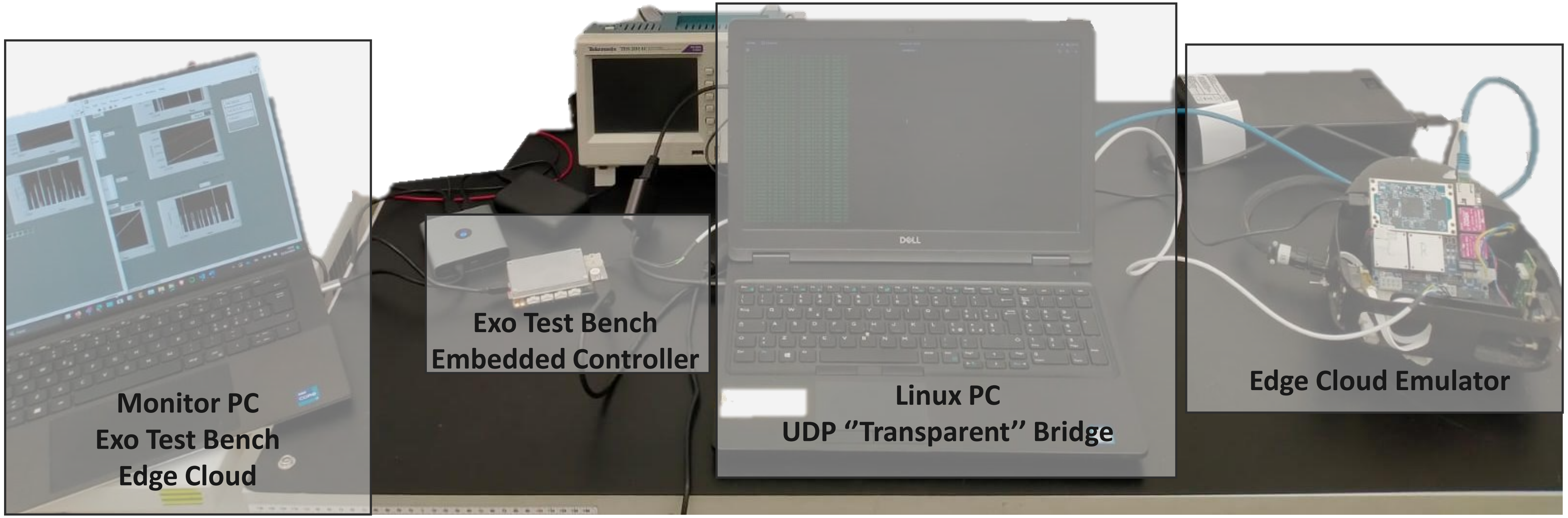
The 6G Network Emulator breaks c3 and c3*.

This introduces some additional delay, even if the emulator operates as transparent forwarder.

Debian PC trasparent bridge verification

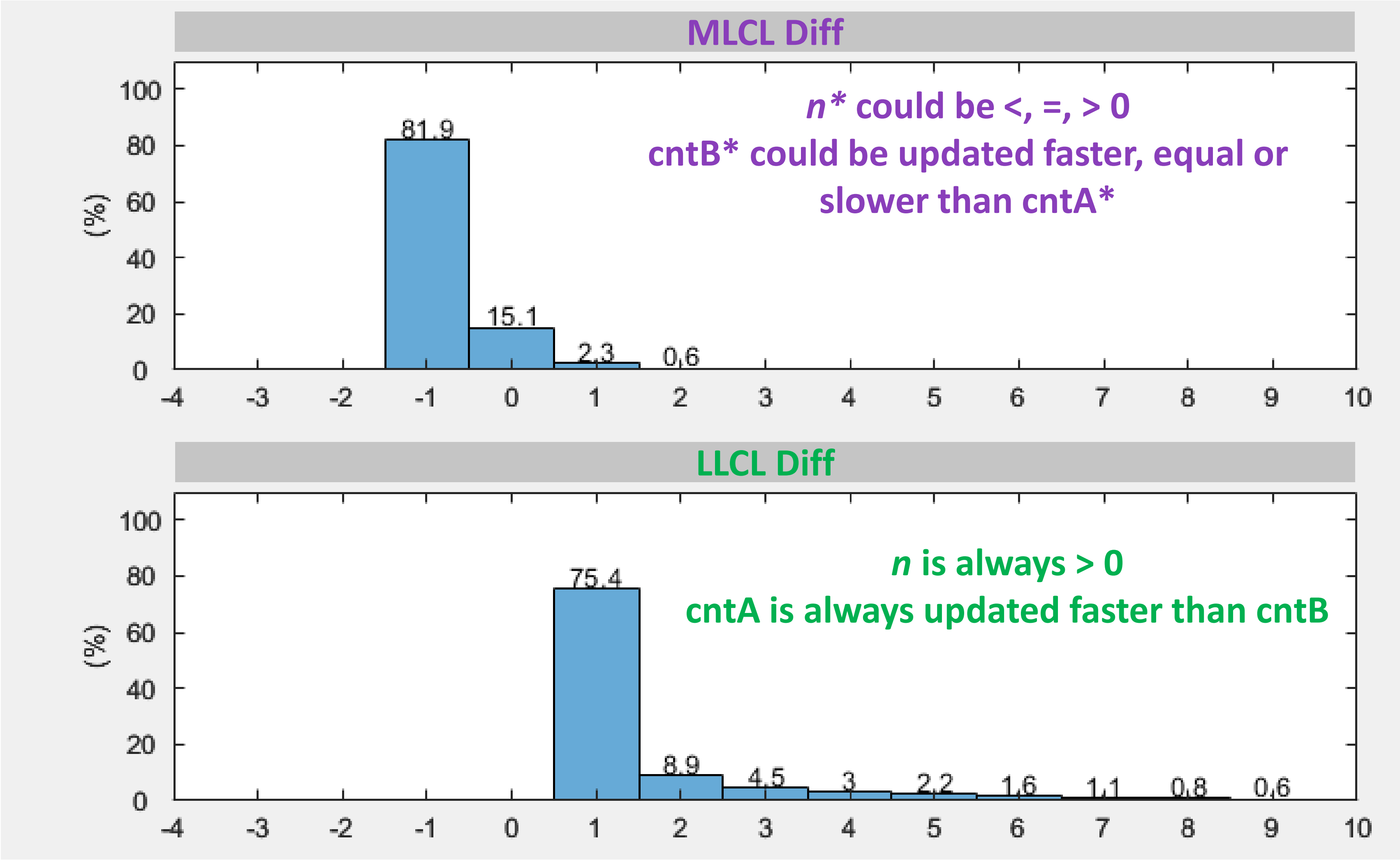
- Performed 30min acquisition test with the emulated counter differentiation performed both in RT or FPGA





Example of result, 30min acquistion

DETERMINISTIC6G



$n^* = 0$

$c3^* = 3{,}1\ ms$

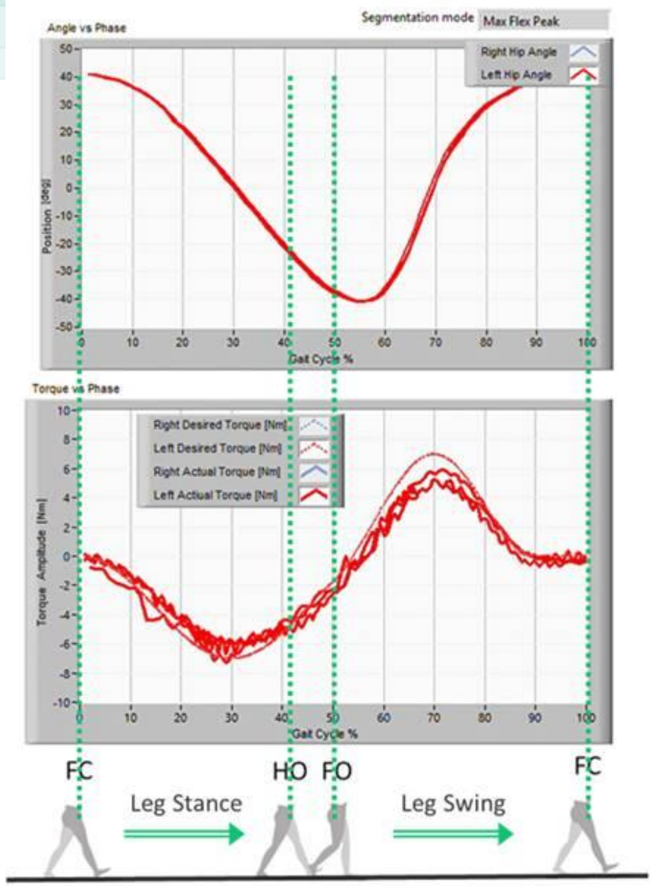
$c3 = 5{,}85\ ms$

$n = 1$

Notes: The results are consistent with bigger computational load for the MLCL RT processor!

Exercise:
Network Differences while Walking

DETERMINISTIC6G



Evaluate the robot performance while providing walking assistance with different network architectures (NA):

- ❑ Fully-integrated
- ❑ Split control architecture, but with the 6GDetCom network delay emulator disabled, and the Debian PC acting as a transparent UDP bridge
- ❑ Split control architecture, with the 6GDetCom network delay emulator enabled*

On the test bench, for each NA:

- ❑ Two different walking velocities (2.5 km/h and 5 km/h)
- ❑ Sinusoidal assistance profiles with ±7 Nm of peak desired torque**
- ❑ A smart algorithm capable of detecting the walking phase and tracking the movement is used in the MLC[1]

* The parameters selected for the 6GDetCom network delay emulator are the default [limit 1000, reorder true] and the imported delay distribution is the PD-Wireless-5G-1.
** The selected assistance profile is representative of a typical worker (body weight: 75 kg, height: 175 cm) walking across the shop floor while carrying a previously lifted small box

[1] "An oscillator-based smooth real-time estimate of gait phase for wearable robotics", T. Yan et all, Autonomous Robots, 2017.

DETERMINISTIC6G

Exoskeleton Use Case Validation

IUVO, USTUTT, SSSA

28/05/2025

DETERMINISTIC6G has received funding from the European Union's Horizon Europe programme under grant agreement No 101096504.

Conclusions

- This work described the development of an emulator designed to study novel paradigms for off-loading the control of an occupational exoskeleton using a 6G-oriented wireless network.

- The observed reduced repeatability in tracking the assistance profiles is particularly critical in real-world use cases involving a human in the loop—especially when the movement is non-periodic or when the high-level controller detects a task transition (e.g., from walking to lifting).

- To achieve the desired assistance specifications and deliver the intended benefits to the user, the described tool can serve as a powerful co-design verification platform for the exoskeleton control layers, algorithms, and communication network.

DETERMINISTIC6G

Emilio Trigili

Assistant Professor

emilio.trigili@santannapisa.it

Sant'Anna Scuola Universitaria Superiore Pisa

Thanks to

Lorenzo Amato

PhD Student

Sant'Anna Scuola Universitaria Superiore Pisa

Filippo Dell'Agnello

Electronic R&D Engineer

IUVO

Visit Us:

THE BIOROBOTICS INSTITUTE

Sant'Anna School of Advanced Studies – Pisa

WRLab

Viale Rinaldo Piaggio 34, 56025, Pontedera, Pisa, Italy

https://www.santannapisa.it/en/institute/biorobotics/wearable-robotics-laboratory

DETERMINISTIC6G Grant Agreement No. 101096504

The DETERMINISTIC6G project has received funding from the European Union’s Horizon Europe research and innovation programme under grant agreement No. 101096504.

If you need further information, please contact the coordinator:

János Harmatos, ERICSSON

E-Mail: coordinator@deterministic6g.eu

or visit: www.deterministic6g.eu

@DETERMINISTIC6G   DETERMINISTIC6G