EPJ Web of Conferences **365**, 06004 (2026) https://doi.org/10.1051/epjconf/202636506004
BPU12 Congress

Digital Twins in Mechatronics: A Symmetry-Based Approach to System Modelling and Analysis

Gabriel Ion Manescu[1*], *Andrei-Costin* Trasculescu[2], *Florin-Alexandru* DITA[3], *Daniela* Coman[4],and *Nicu-George* Bizdoaca[5]

[1]University of Craiova, Mechatronics and Robotics Department, Craiova, Romania
[2]University of Craiova, Mechatronics and Robotics Department, Craiova, Romania
[3]University of Craiova, Mechatronics and Robotics Department, Craiova, Romania
[4]University of Craiova, Engineering and Management of Technological Systems Department, Drobeta Turnu Severin, Romania
[5]University of Craiova, Mechatronics and Robotics Department, Craiova, Romania

Abstract. This paper explores the integration of symmetry principles into digital twin models for complex mechatronic systems, highlighting their potential to enhance modelling efficiency, anomaly detection, and system scalability. By leveraging structural and functional symmetries, the proposed approach supports modular system design, reduces computational overhead, and improves predictive maintenance. A comprehensive showcase is presented, featuring a modular automated production line composed of a conveyor with 90 FIFO buffer positions, multiple CNC machining centres, and two 7-axis robots controlled via Siemens Sinumerik RunMyRobot/Direct Control. The simulation framework employs a combination of Software-in-the-Loop (SiL) and Model-in-the-Loop (MiL) methodologies using a CMVM digital twin of the CNC controller, Simit for communication simulation, and Mechatronic Concept Designer (MCD) for behaviour modelling. Hierarchical physical and functional decomposition, aligned with Weiss and Qiao's methodology, is applied to facilitate the development of a Component Mapping Matrix and ensure simulation fidelity. The presented approach demonstrates how symmetry and modularity in digital twins can accelerate deployment, enhance robustness, and improve decision-making in cyber-physical manufacturing environments.

1 Introduction

Digital twins have emerged as powerful tools in the modelling, simulation, and analysis of complex mechatronic systems. By creating virtual replicas of physical components, digital twins enable real-time monitoring, predictive maintenance, and performance optimization. In mechatronics, where mechanical, electrical, and software components interact

---

* Corresponding author: manescu.gabriel.w5t@student.ucv.ro

dynamically, accurate digital representation is crucial. Symmetry, a fundamental principle in physics and engineering, can significantly enhance digital twin models by reducing complexity and improving computational efficiency. Structural and functional symmetries within a system allow for modularization, where mirrored components can share behaviours and design rules. This symmetry-based approach simplifies system modelling by identifying invariant properties and reusing simulation patterns. It enables engineers to detect anomalies by comparing symmetric counterparts, improving fault detection accuracy. Additionally, symmetrical modelling supports scalable digital twin architectures, facilitating multi-domain integration in mechatronic design. Control strategies can be optimized by leveraging symmetrical dynamics, reducing controller design time and computational burden. In robotics, symmetrical limbs or actuators can be mirrored in the digital environment to accelerate testing and validation. When incorporated into digital twin platforms, symmetry principles help in achieving more robust and adaptable systems. They also contribute to better visualization and interpretation of system behaviours. Real-time synchronization between physical and virtual twins becomes more efficient when symmetrical relationships are maintained. As digital twin technology evolves, incorporating symmetry will be key to handling the increasing complexity of cyber-physical systems. This approach not only enhances model fidelity but also supports intelligent decision-making in automated mechatronic environments

2 Show Case

We will be looking at an production line consisting of several sub section as follows.

2.1 Conveyor

One conveyor (made of 10 identical standard sections). The conveyor rollers are actuated by one AC Motor VFD driven with the same prescribed velocity for all the sections. When the entire line is filled out with pallets (a total 90 pallets could be present on the entire conveyor), then automatically the speed is decreased with 10% until there will be max 72 pallets present on the conveyor. At this time the prescribed velocity will be re imposed to all VFDs. A 90 position FIFO buffer will track the part types at the Conveyor’s PLC.

At the end of this conveyor (after the 90 positions) if a pallet is detected to be empty or the part which is on the pallet still has “InProcessOK” or “InProcessRework” code for the current operation, those pallets will be recirculated back to the beginning of the conveyor. The adaptive AI strategy will need to be determined in a future work for optimal insertion of those recirculated pallets together with the method of the recirculation (e.g. via a parallel conveyor or with one of the robots, etc.).

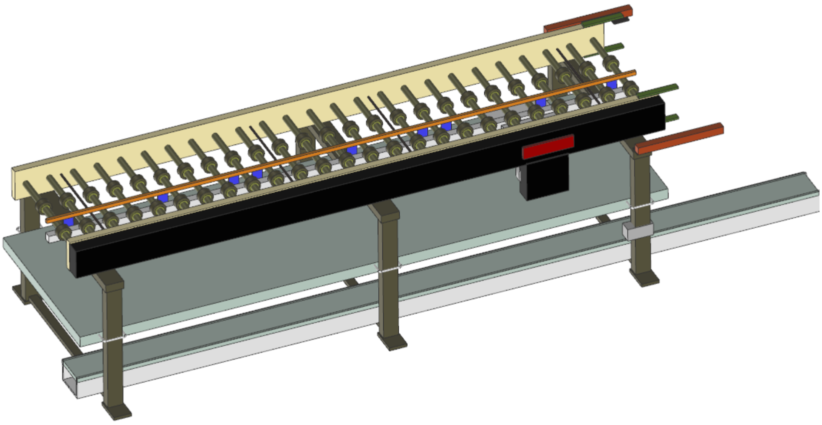


Fig.1. Conveyor Section

2.2 Machine Tool

Up to 9 machine tools. Depending of the overall application production balance, capital expenditure, etc, the machining centres could be identical (possible with different tools and fixtures) or highly different, however from the system modelling perspective that is irrelevant for as long as they will fit in the allocated floor space. Each unit machining can be enabled / disabled either by the operator or the tools or fixture status.

The part is handled via automated loading front door which moves up/down in order to accommodate the part transfer.

The fixture will lock (clamp) the workpiece on the machine for the duration of the machining and machine handling. There shall be no motion applied to the workpiece without the fixture being clamped, in such unwanted event the part might rock off the fixture due to high values of the axes accelerations.

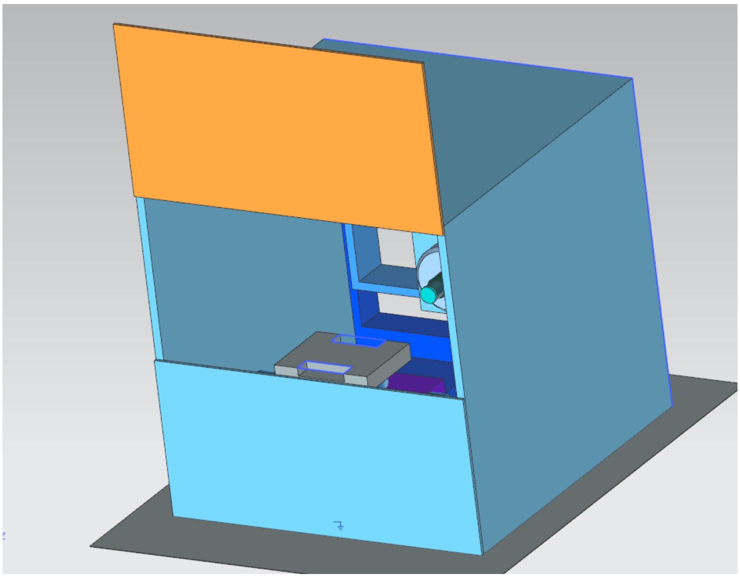


Fig.2. Machining Centre

2.3 Robot

Up to 4 robots 7 Axes (6 regular axes of SCARA with additional linear 7th axis). Those are any standard 7Axis robot which would be compatible with RMR/Direct Option for Siemens SinumerikOne controllers, which have the required payload to handle the workpiece as well as the work envelope to cover the motion from the robot rail to/from a target machining centre. During the analysis various sizes of the robots coming from the RMR database could be tested for the best fit for the application. For the first iteration, arbitrarily, covering the payload and reachability, the COMAU NJ420-3.0 was selected.
The robot will pick the part from the pallet on the fly and place it in the closest available machining centre to be machined as the RFID tag which is attached to the workpiece dictates as the next operation for as long as the status of the workpiece is marked as "InProcessOK" or "InProcessRework" for the operation number current active for this line.
Same robot can also go for a machine pickup (unload the machine centre) on demand, when a workpiece will be unloaded from machine which requires it and will be placed on the first available empty pallet which comes down the line.
There can be up to 4 robots on the line, but for this study we will limit to just 2.
Both robots could simultaneously have same or different tasks(loading and/or unloading) as the situation requires it. However since such decision requires further analysis, arbitrarily will decide that first robot will perform only loading tasks (pickup from the conveyor) while the second robot will perform only unloading tasks (drop off to the conveyor)
As soon as a pallet from the FIFO buffer is flagged for a task, the pallet status gets marked as "InProcessOccupied".
The entire details of the part are recorded on the accompanying RFID tag which travels with the part on all the stations.

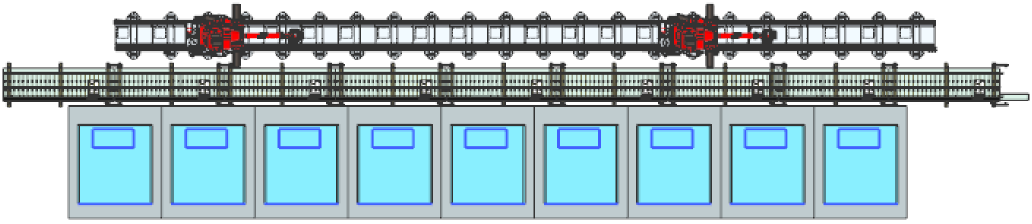


Fig.3. The entire showcase for 2 robots and 9 machining centres

3 Methodology

As our aim is to really use a modular approach, this will be best fitted if the controlling solution will stay the same regardless of the mechanical robotic arm used (from the available database), therefore a Siemens Sinumerik RunMyRobot /Direct Control will be used. Moreover for efficiency reasons we can use one CNC unit to control up to 4 7 axes robot (since the Sinumerik is good for 31 axes and we need all together 28 axes for 4 robots). Individual path control is ensured by assigning each robot to a unique CNC processing channel.

That means that one Sinumerik One controller with up to 4 channels with 28 Sinamics Servodrives will be implemented for the real solution.

As for the simulation we want to use a complete Digital Twin of the controller driving the robots in a full behavioural based simulation, therefore a possible topology of such simulation will involve:

- 1 x Create My Virtual Machine (CMVM) model as Digital Twin of the 4 Channel CNC controller (having therefore a Software In The Loop SiL setup), which includes the HMI-Operate, the NC controller, the PLC controller and the Machine Control Panel (MCP).
- 1 x Simit Framework which will simulate all RFID antennas and modules as well as connecting the 9 maching centres Profinet communication as Model In The Loop
- 1 x MCD (with RTB behaviours) to emulate the flow of data accompanying every workpiece.

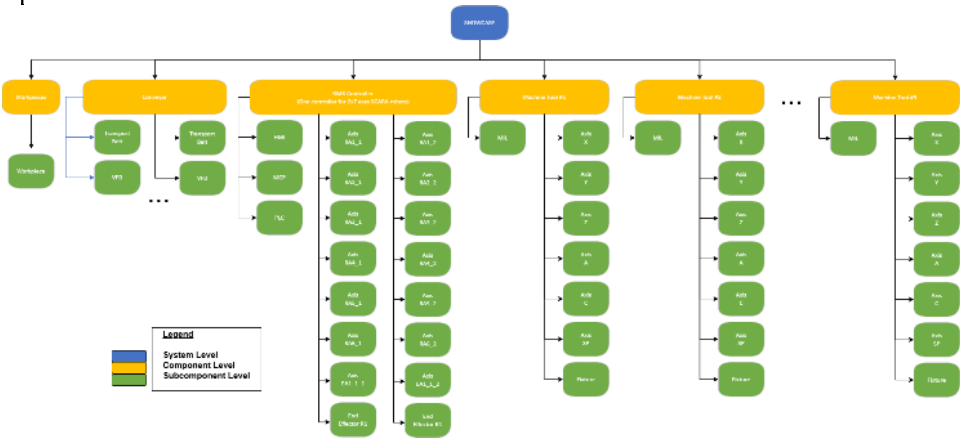


Fig.4. Physical Hierarchical Decomposition

From this figure (Fig.4), we can observe the modular approach for various sections of the Show Case.

With all those steps executed, we can create the desired simulation.

4 Simulation Environment

Everything starts with the engineering stage. During that stage the machining centres were selected to match the manufacturing needs for the operation which is assigned to the Show Case line. As we mentioned the machining centres were selected from a library of available machines.

As we bring the machine assembly on the CAD world, the machine components are as a standard for a 5 axis X/Y/Z/A/B/SP model machine with a hatch, front loading door and fixture. This represents a simplified simulation model which excludes the tool mechanisms as well as the tool magazine(s) if any present in the real machine, this however being ideal for a MiL simulation setup.

When we do simulate such structure we need to focus on the features which matters for the simulation (observe the hierarchical decomposition as well). In our case we would need just to simulate:

- Front Door Opening and Closing as part of machine function conditioned by the signal interface RMR controller to the individual machine controller (see table xx, Signal map)
- Workpiece present
- Fixture Open
- Fixture Closed

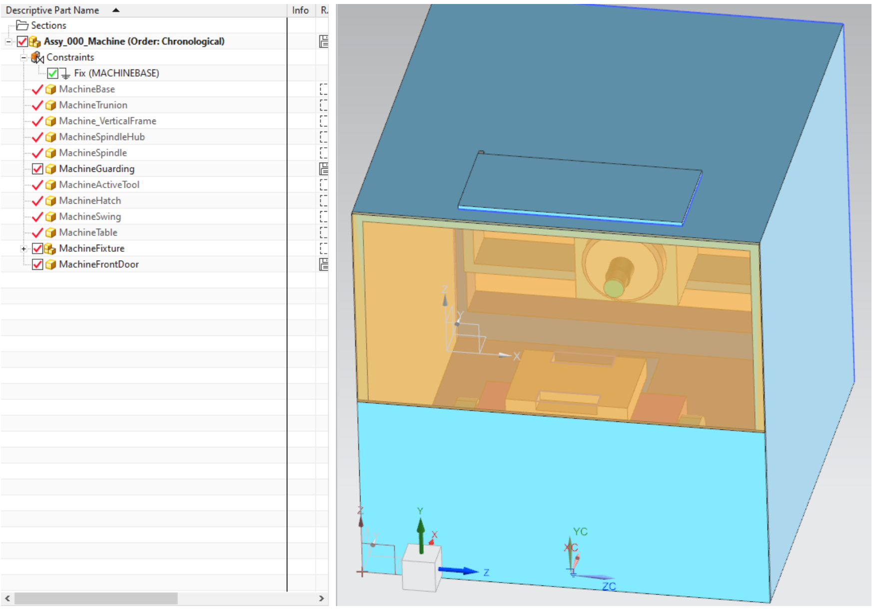


Fig.5. Machining Centre CAD Assembly

Any other behaviour is irrelevant for the MiL station, if more exists doesn't affect the simulation (excepting usage of simulation resources, e.g. computing power).

Now, is desirable that as we were able to select our machines from the CAD library, to be able to have those behaviours automatically follow the CAD work. Now, depending on the platform CAD/CAE used, the behaviours could be stored in the upper assembly file or another superior level might have to be created, which will just take the CAD full assembly without any CAD modification and will link the simulation related entities to the CAD world. The authors recommend the latter since in most cases such approach allows quite easy to increase the library to new models applying a standard set of simulation entities. In our test case using Mechatronic Concept Designer (MCD).

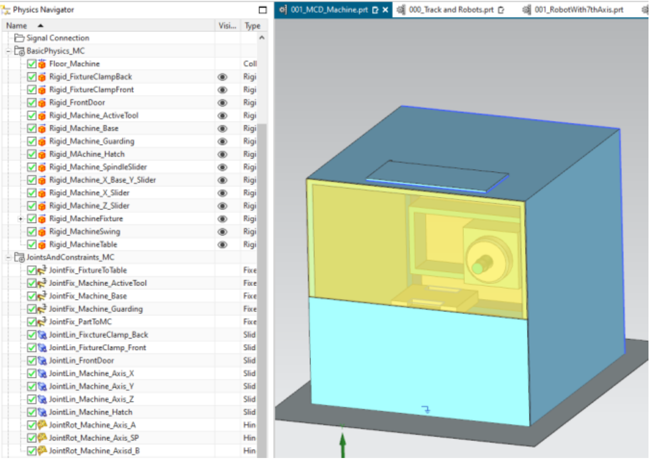
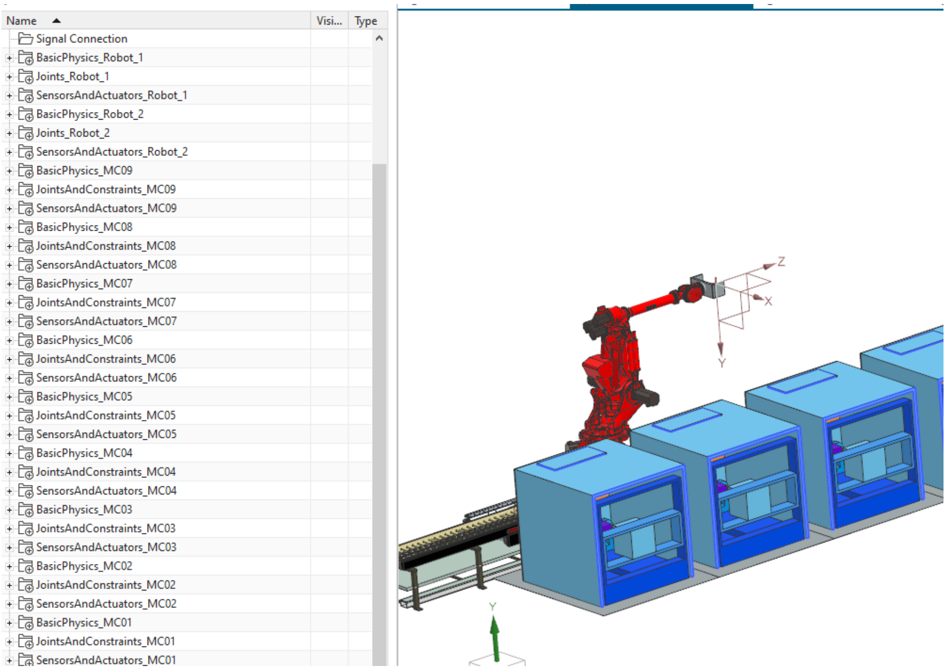


Fig.6. Machining Centre MCD assembly navigator

In this case as we will assemble on the CAD world all 9 machining centers as intended, for as long the CAE environment does have a "inheritance" capability, automatically all the desired behaviors will become available automatically as instances, one for every machining center.



Name | Visi... | Type

- Signal Connection
- BasicPhysics_Robot_1
- Joints_Robot_1
- SensorsAndActuators_Robot_1
- BasicPhysics_Robot_2
- Joints_Robot_2
- SensorsAndActuators_Robot_2
- BasicPhysics_MC09
- JointsAndConstraints_MC09
- SensorsAndActuators_MC09
- BasicPhysics_MC08
- JointsAndConstraints_MC08
- SensorsAndActuators_MC08
- BasicPhysics_MC07
- JointsAndConstraints_MC07
- SensorsAndActuators_MC07
- BasicPhysics_MC06
- JointsAndConstraints_MC06
- SensorsAndActuators_MC06
- BasicPhysics_MC05
- JointsAndConstraints_MC05
- SensorsAndActuators_MC05
- BasicPhysics_MC04
- JointsAndConstraints_MC04
- SensorsAndActuators_MC04
- BasicPhysics_MC03
- JointsAndConstraints_MC03
- SensorsAndActuators_MC03
- BasicPhysics_MC02
- JointsAndConstraints_MC02
- SensorsAndActuators_MC02
- BasicPhysics_MC01
- JointsAndConstraints_MC01
- SensorsAndActuators_MC01

Fig.7. Entire line MCD navigator

The above Fig.7 shows us exactly such circumstance when using MCD, where every machining centre (and to that extent every robot as well) creates all those behaviours automatically, reducing tremendously the time required to built such simulation

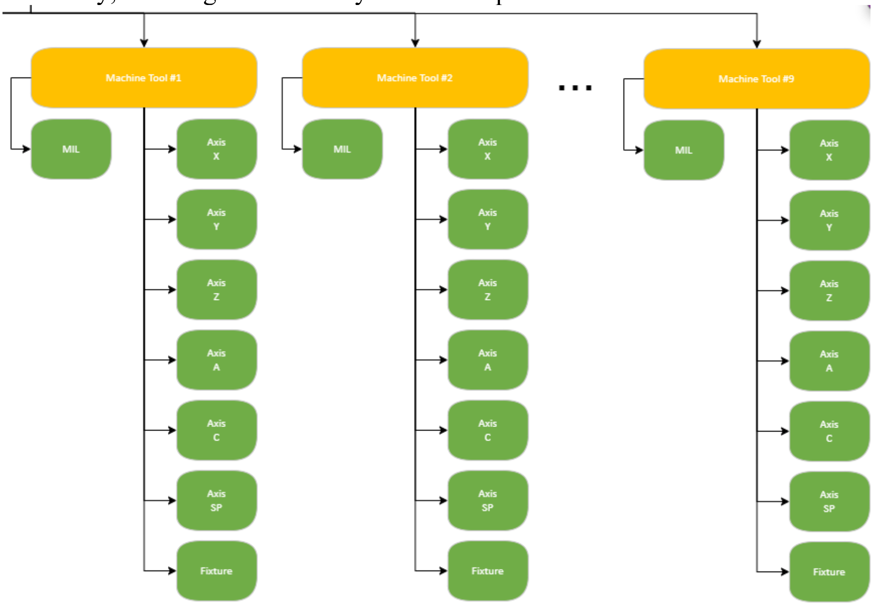


Fig.8. Physical hierarchical decomposition detail, machining centre

That is the direct result by implementing the subset of the physical hierarchical decomposition related to the 9 machining centers as shown in Fig.8.

Same solution can be observed for the robots, all required behaviours are inherited from the library as one can observe it in Fig.9. for one SCARA robot with 7 axes.

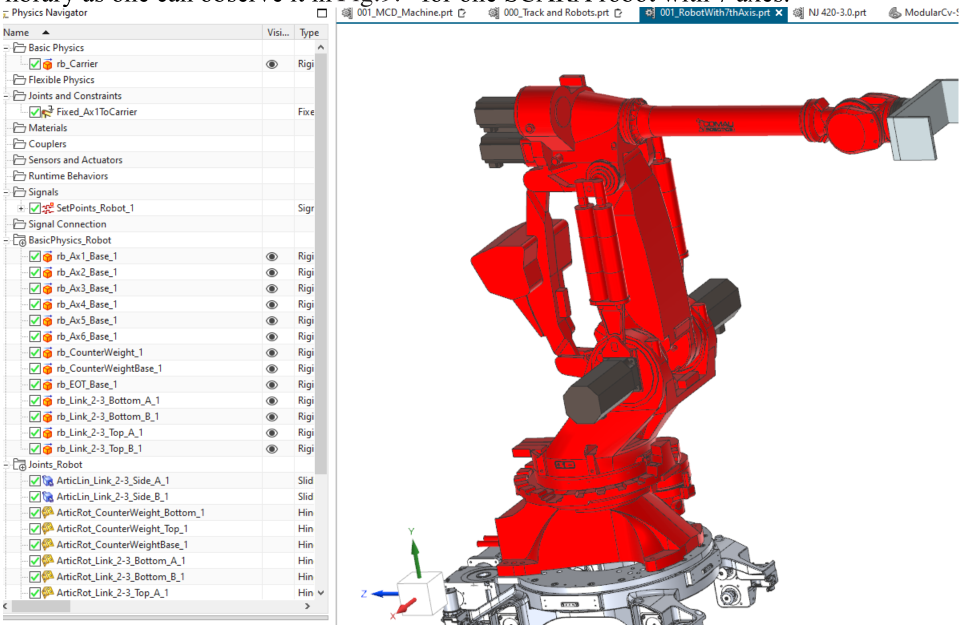


Fig.9. Robot 7 axis simulation behaviour standard from library

This figure (Fig.9) results from the subset of the physical hierarchical decomposition detail related to the subcomponents of the RunMyRobot component as depicted in Fig.10.

In the same manner, any more modules used multiple time would be able to be implemented. The method would further be possible to be applied for behaviors resulting from the Component Mapping matrix and their derived functions (that will be presented in future work).

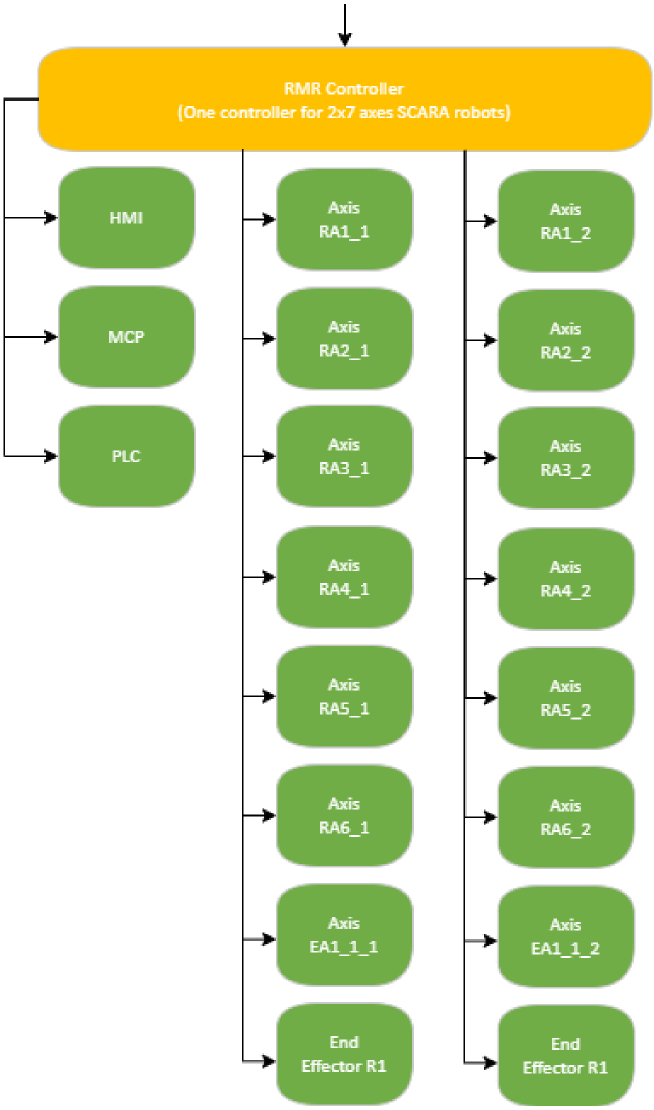


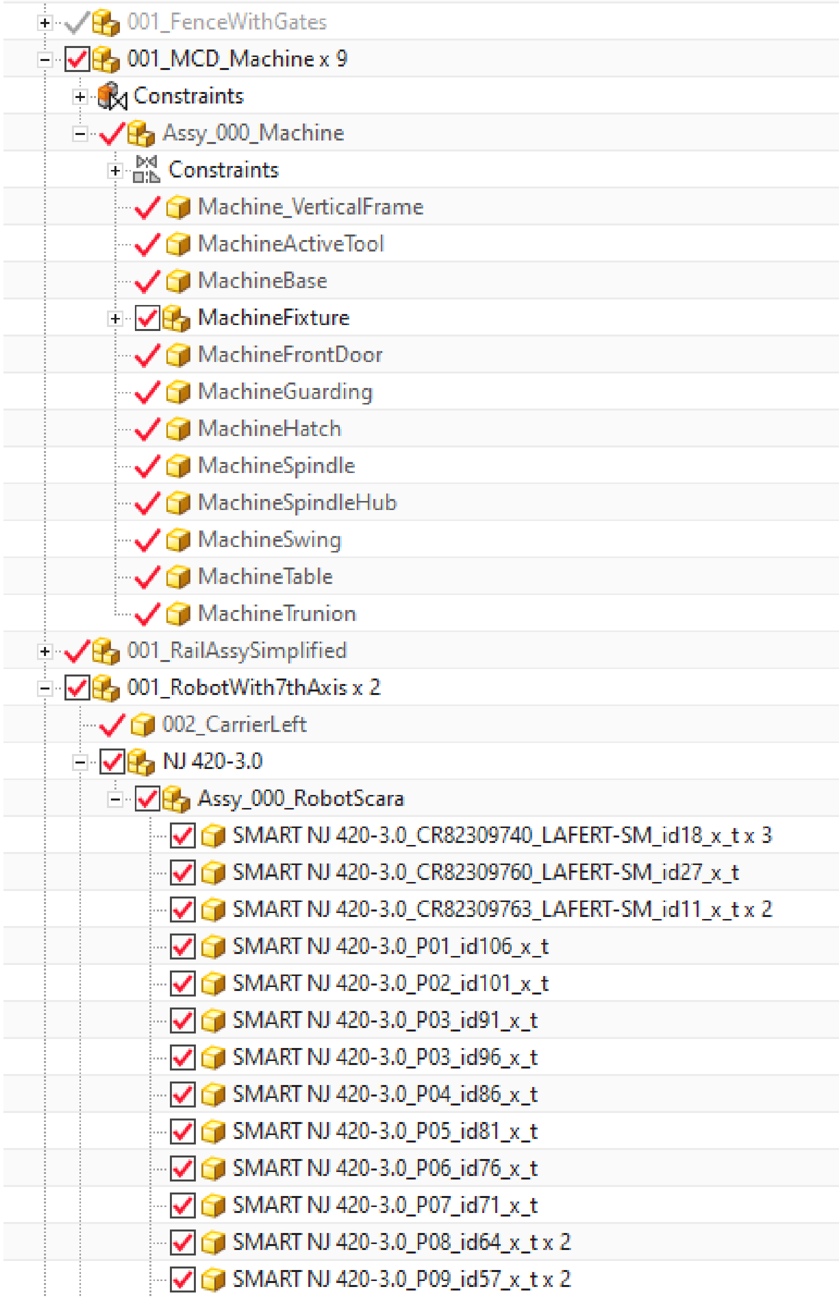
Fig.10. Physical hierarchical decomposition detail. Run My Robot component

5 Conclusions

Based on the details presented into the previous chapter, for as long as the engineering tools utilized for implementation of the simulation accepts and supports such methodology, the behaviours could be inherited by the components (as part of libraries) for as long the following conditions are satisfied:

- The CAD environment will be created neutral, with just CAD elements, in such way the CAD structure being independent of the behaviours. This can be observed in Fig.11 by subcomponents "Assy_000_Machine" for component "001_MCD_Machine" and subcomponent "Assy_000_RobotScara" for component "NJ420-3.0".

- The behaviours (through their behavioural components) are added to the simulation by referencing the standard CAD entities assembled at the previous point. Those are implemented at the "001_MCD_Machine" and "001_RobotWith7thAxis" file level. The behavioural components are shown in the Physics Navigator for the machining centre in Fig.6 and for the 7Axes robot in the Physics Navigator shown in Fig.9.



- ☑ **000_Track and Robots (Order: Alphanumeric)**
  - ✓ 001_FenceWithGates
  - ☑ 001_MCD_Machine x 9
    - Constraints
    - ✓ Assy_000_Machine
      - Constraints
      - ✓ Machine_VerticalFrame
      - ✓ MachineActiveTool
      - ✓ MachineBase
      - ☑ MachineFixture
      - ✓ MachineFrontDoor
      - ✓ MachineGuarding
      - ✓ MachineHatch
      - ✓ MachineSpindle
      - ✓ MachineSpindleHub
      - ✓ MachineSwing
      - ✓ MachineTable
      - ✓ MachineTrunion
  - ✓ 001_RailAssySimplified
  - ☑ 001_RobotWith7thAxis x 2
    - ✓ 002_CarrierLeft
    - ☑ NJ 420-3.0
      - ☑ Assy_000_RobotScara
        - ☑ SMART NJ 420-3.0_CR82309740_LAFERT-SM_id18_x_t x 3
        - ☑ SMART NJ 420-3.0_CR82309760_LAFERT-SM_id27_x_t
        - ☑ SMART NJ 420-3.0_CR82309763_LAFERT-SM_id11_x_t x 2
        - ☑ SMART NJ 420-3.0_P01_id106_x_t
        - ☑ SMART NJ 420-3.0_P02_id101_x_t
        - ☑ SMART NJ 420-3.0_P03_id91_x_t
        - ☑ SMART NJ 420-3.0_P03_id96_x_t
        - ☑ SMART NJ 420-3.0_P04_id86_x_t
        - ☑ SMART NJ 420-3.0_P05_id81_x_t
        - ☑ SMART NJ 420-3.0_P06_id76_x_t
        - ☑ SMART NJ 420-3.0_P07_id71_x_t
        - ☑ SMART NJ 420-3.0_P08_id64_x_t x 2
        - ☑ SMART NJ 420-3.0_P09_id57_x_t x 2

Fig.11. Object structure CAD and begaviours for the Show Case

With this method, a true symmetry based solution based on modularity can be implemented having multiple advantages, from which we can point to:

- By reusing behaviours in a modular solution, by creating the behaviour once, is multiplied to the number of instances automatically, shortening significant the implementation effort. That can be observed for example by the fact that 9 instances of the "001_MCD_Machine" (as shown in Fig.11) creates automatically the behavioural components for all 9 machining centres (as shown in Fig.7).
- By separating the CAD environment and the behavioural environment, the CAD components could be replaced with similar CAD components and new simulation based on different mechanics (for as long it follows the same kinematic definition) can be effortlessly implemented. For example the 6 axis Comau robot could be replaced by a 6 axis Fanuc robot (as their kinematic is the same) of course taking in consideration the payload and reachability criteria.
- By reusing the same behaviours already proven as part of a library the validation of the simulation becomes implicit.

Disclaimer

The Siemens solution does not represent the unique solution that could cover such a simulation's needs; the authors arbitrarily chose it, while was supporting the methodology proposed

Acknowledgment

This work was supported by the grant "HUB-UCv—Support Center for International CD Projects for the Oltenia region" project code POC/80/1/2/107885, co-financed by the European Social Fund within the Sectorial Operational Program COMPETITIVENESS 2014–2020.

References

1. [1] Executive Ofice of the President – President's Council of Advisors on Science and Technology PCAST 2024, "Report to the President – Supercharging Research: Harnessing Artificial Inteligence to Meet Global Challenges", 2024, https://doi.org/10.2172/2481685
2. [2] Executive Ofice of the President – President's Council of Advisors on Science and Technology PCAST 2014, "Report to the President – Accelerating US Advanced Manufacturing", 2014, https://www.manufacturingusa.com/sites/manufacturingusa.com/files/amp20_report_final.pdf.pdf
3. [3] National Academies of Sciences, Engineering and Medicine, "Foundational Research Gaps and Future Directions for Digital Twins", 2024, https://doi.org/10.17226/26894
4. D. Siegrist, A. Matsikis, S. Dirbach, A/Volz, F. Bellaouna, "A Virtual Commissioning Selection Approach for Machine Automation", 2022, IEEE 27th International Conference on Emerging Technologies and Factory Automation (ETFA), DOI: 10.1109/ ETFA52439. 2022.9921618.

5. D. Shylla, A. Jain, P.Shah, R. Sekhar, "Model in Loop (MIL), Software in Loop (SIL) and Hardware in Loop (HIL) Testing in MBD", 2023, 4th IEEE Global Conference for Advancement in Technology (GCAT), DOI: 10.1109/GCAT59970.2023.10353323.
6. B.A. Weiss, G. Qiao, "Hierarchical decomposition of a manufacturing work cell to promote monitoring, diagnostics, prognostics", 2017, ASME 2017 12th International Manufacturing Science Engineering Conference collocated with the JSME/ASME 2017 6th International Conference on Materials Processing, [online] https://tsapps.nist.gov/ publication/get_pdf.cfm?pub_id=922582.
7. [7] D. Fait, V. Masek, R. Cermak, "Using Digital Twins in mechatronics and manufacturing", 2022, International Symposium on Multidisciplinary Studies and Innovative Technologies (ISMSIT), DOI: 10.1109/ISMSIT56059.2022.9932840