Flexible scheduling for human-robot assembly

Riccardo Maderna
DEIB, Politecnico di Milano
Milano, Italy
riccardo.maderna@polimi.it

Andrea Maria Zanchettin
DEIB, Politecnico di Milano
Milano, Italy
andreamaria.zanchettin@polimi.it

Paolo Rocco
DEIB, Politecnico di Milano
Milano, Italy
paolo.rocco@polimi.it

***Abstract*—This paper presents a complete framework for dynamic task scheduling in the context of flexible human-robot assembly. A Digital Twin tracks the state of the process in real-time by monitoring the activity of humans and robots. Scheduling decisions are taken online based on the current state and the predicted evolution of the system.**

***Index Terms*—Human-robot collaboration, flexible manufacturing, task scheduling, human monitoring**

I. Introduction

As manufacturing shifts from mass production to mass customization, human-robot collaboration is expected to add flexibility to production lines [1]. Versatile scheduling algorithms are needed to organize the increasingly complex workflow and exploit the gained flexibility, ensuring optimal use of resources and smart management of failures [2]–[4].

In the proposed approach (Fig. 1), a Digital Twin (DT) tracks the state of the collaborative workspace in real-time based on data coming from the robot controllers and a human monitoring unit. Monitoring and predicting the evolution of the ongoing human activity is beneficial for task scheduling [5]–[7]. The DT is then used to simulate the future evolution of the system and determine the optimal instructions for humans and robots with a receding horizon approach. This allows dynamically adapting the schedule to the variability of human behaviour and the occurrence of robot faults.

In the following, the main elements of the system architecture are presented: the DT, the human monitoring unit, and the dynamic scheduling algorithm.

II. Digital twin

The DT describes both the physical structure of the workspace and the assembly task, including situations that originate from human variability and robot failures. The process consists of a set of agents (humans and robots) devoted to the assembly of a set of products according to a time-varying production mix. Each product type may be completed following different assembly sequences, which are encoded in modified AND/OR graphs, along with data on resource requirements and workspace layout.

Given the AND/OR graphs, the DT of the process is automatically built as a partially controllable Timed Petri Net, whose evolution is tracked in real-time through subsequent transitions firings. Transitions are either controllable, whose firing is decided by the scheduler and model the start of new tasks, or uncontrollable, which are triggered by exogenous events and model task completion and fault occurrence.

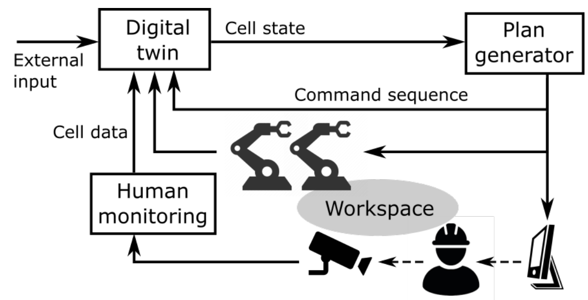


Fig. 1: Control system architecture.

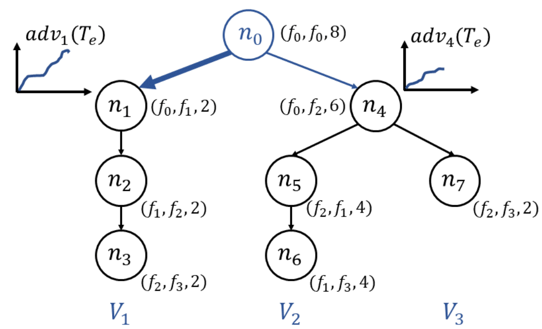


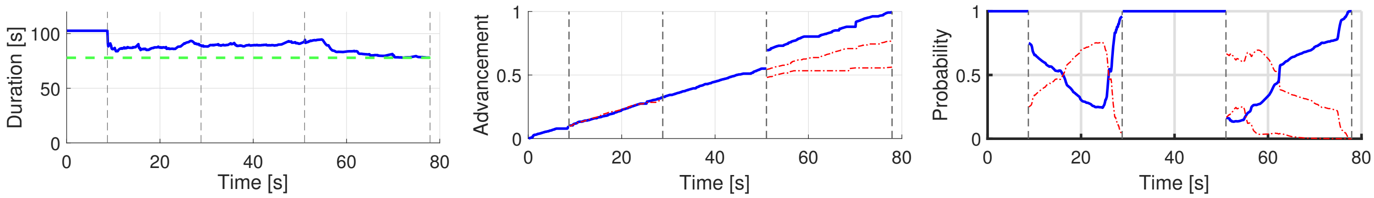
Fig. 2: Template tree of a task with three known variants. After n_0, each one of the two branches is monitored as if it was the correct one. For each branch, the algorithm outputs in real time an estimate of task progress and the likelihood of being the one performed by the human (represented by the width of the blue arrows).

If production changes, one can modify the descriptions of product assembly accordingly and generate the new DT.

III. Real-time monitoring of human tasks

One of the information that defines the DT state is the remaining time to completion of the ongoing tasks. Estimating the duration of human activities is a tough challenge: at each repetition, he/she will complete the same task with different speeds and movements. Also, tasks may be performed following different sequences of actions and the possibility of errors and pauses must be considered. Thus, data from past executions are insufficient, but additional information on the current activity is needed. If a real-time estimate $adv(T_e)$ of the task progress is available, an estimate of its duration is:

$$\widehat{T}(T_e) = \frac{T_e}{adv(T_e)} \tag{1}$$



(a) Expected (blue) and actual (green) duration. (b) adv of correct (blue) and wrong (red) segments. (c) P_i of correct (blue) and wrong (red) segments.

Fig. 3: Execution a shorter-than-average variant: as the correct variant is recognized the prediction converges to the actual task duration.

The task advancement is obtained as the output of a Dynamic Time Warping-based algorithm that receives human motion data as input. The developed method is robust against nonlinear speed variations and occlusions. Also, it does not require any offline learning phase, but compares the input to a reference template of the activity, which is learnt online from past repetitions of the same task [8].

To account for the presence of multiple variants of the same task, the reference template has a tree structure (Fig. 2). Each time the human executes the task, the ongoing operation is compared to the known variants. The algorithm identifies which is the one being performed and evaluates the progress and the expected duration of the task accordingly. When a new variant is detected, it is added to the template [9].

The method has been tested on a complex assembly task with six variants, including two error cases. The algorithm was always able to recognize when a new variant was performed. Fig. 3 shows one execution of a known variant of the task. Overall, the proposed method accurately predicts the duration, regardless of the presence of several variants of the task.

IV. Flexible Scheduling Algorithm

The scheduling algorithm assigns tasks to all agents, which assemble multiple products concurrently. For each product, the assembly sequence is not fixed a priori, but the scheduler dynamically makes this choice for higher flexibility. Also, the scheduler decides when and which product to start next.

Starting from the current DT state, feasible system evolutions are found by exploring the Reachability Tree of the Petri Net (Fig. 4) over a planning horizon. The optimal schedule is obtained as the sequence of controllable transition firings along the branch with minimum cost. The cost function penalises the distance from the target mix, the takt time, the agents' idle time, and WIP storage. The plan is updated with a receding horizon approach: commands are sent to the free resources that start the new operations, the DT state is updated, and the schedule recomputed. This allows adapting to the natural variability of the process and unforeseen events, such as faults.

During the experimental campaign, involving a human and two robots, the dynamic scheduler was able to reduce the cycle time by adapting to the actual duration of tasks and by optimally handling the occurrence of robot faults (Fig. 5).

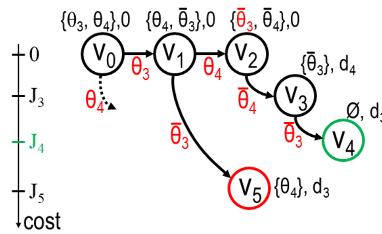


Fig. 4: Reachability tree example. Nodes represent states of the DT, labels show enabled transitions and arrival time. Node v_4 is the current best leaf.

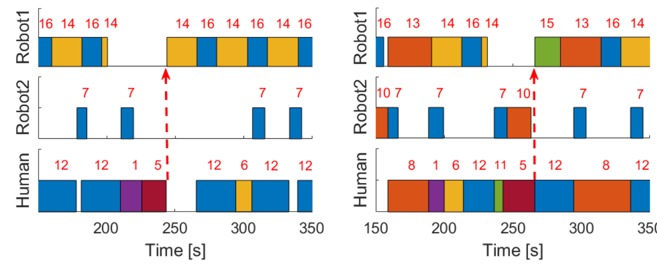


Fig. 5: Handling of a Robot1 failure (red crosses). Left: planning the recovery action as soon as possible leads to high idle times. Right: the proposed scheduler delays the recovery action to prepare a WIP for Robot2, avoiding a complete stop of the production.

References

[1] A. Ajoudani, A. M. Zanchettin, S. Ivaldi, A. Albu-Schäffer, K. Kosuge, and O. Khatib, "Progress and prospects of the human–robot collaboration," *Autonomous Robots*, vol. 42, no. 5, pp. 957–975, 2018.

[2] Y. Chen, X. Mao, F. Hou, Q. Wang, and S. Yang, "Combining re-allocating and re-scheduling for dynamic multi-robot task allocation," in *2016 IEEE International Conference on Systems, Man, and Cybernetics (SMC)*, 2016, pp. 395–400.

[3] L. Johannsmeier and S. Haddadin, "A hierarchical human-robot interaction-planning framework for task allocation in collaborative industrial assembly processes," *IEEE Robtics and Automation Letters*, vol. 2, no. 1, pp. 41–48, 2017.

[4] A. Casalino, A. M. Zanchettin, L. Piroddi, and P. Rocco, "Optimal scheduling of human-robot collaborative assembly operations with time petri nets," *IEEE Transactions on Automation Science and Engineering*, pp. 1–15, 2019.

[5] V. V. Unhelkar, P. Lasota, Q. Tyroller, R. Buhai, L. Marceau, B. Deml, and J. A. Shah, "Human-aware robotic assistant for collaborative assembly: Integrating human motion prediction with planning in time," *IEEE Robotics and Autom. Letters*, vol. 3, no. 3, pp. 2394–2401, July 2018.

[6] D. Riedelbauch and D. Henrich, "Exploiting a human-aware world model for dynamic task allocation in flexible human-robot teams," in *2019 Intern. Conf. on Robotics and Automation*, 2019, pp. 6511–6517.

[7] A. M. Zanchettin, A. Casalino, L. Piroddi, and P. Rocco, "Prediction of human activity patterns for human–robot collaborative assembly tasks," *IEEE Trans. Industrial Informatics*, vol. 15, no. 7, pp. 3934–3942, 2019.

[8] R. Maderna, P. Lanfredini, A. M. Zanchettin, and P. Rocco, "Real-time monitoring of human task advancement," in *2019 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2019.

[9] R. Maderna, M. Ciliberto, A. M. Zanchettin, and P. Rocco, "Robust real-time monitoring of human task advancement for collaborative robotics applications," in *2020 IEEE/RSJ International Conference on Intelligent Robots and Systems*, 2020.