POLITECNICO MILANO 1863

SCUOLA DI INGEGNERIA INDUSTRIALE E DELL'INFORMAZIONE

EXECUTIVE SUMMARY OF THE THESIS

A Reachability-Based Decoupling Solution to the Routing Problem in Smart Manufacturing

LAUREA MAGISTRALE IN AUTOMATION AND CONTROL ENGINEERING - INGEGNERIA DELL'AUTOMAZIONE E DEL CONTROLLO

Author: ALFREDO STAMA

Advisor: PROF. MARIA PRANDINI

Co-advisor: ANDREA CATALDO, ALESSANDRO FALSONE, LUCREZIA MANIERI

Academic year: 2022-2023

---

1. Introduction

Manufacturing industries are typically characterized by a sequence of operations that need to be performed in a specific order. Processing times and throughput of a manufacturing plant can be improved by designing appropriate strategies to operate the transport lines that are driving the goods through the different machines for subsequent processing, avoiding bottlenecks, starvation and congestion problems [6].

In this work, we address optimal routing in smart manufacturing according to a model predictive control (MPC) approach that was proposed in the PhD thesis [3] based on a mixed logical dynamical (MLD) model of the transport line in a plant and using a cost function that favors those actions which move pallets towards their destinations through the shortest path. Conflicts are avoided by introducing state-dependent constraints on the admissible control actions.

MLD systems ([1]) are discrete time systems described by linear equations subject to linear mixed-integer inequalities, i.e. inequalities involving both *continuous* and *binary* variables. These include physical/discrete states, continuous/integer inputs, and continuous/binary auxiliary variables. As a result, the obtained MPC optimization problem is admittedly computationally intensive due to its combinatorial nature, since the control inputs are discrete, which can slow down the transport line operation and reduce the manufacturing plant throughput as the number of pallets and the MPC prediction horizon grow. Indeed, the MPC algorithm proposed in the PhD thesis [3] is characterized by the online computation times that become larger and larger as the number of pallets and prediction horizon grows to the point that most of the time of the pallets on the transport line is spent waiting for the next control action, thus deteriorating the performance of the manufacturing plant.

Here, we propose a methodology to defeat such a computational complexity by decoupling the MPC optimization program into multiple smaller dimensional programs that can be solved in parallel. This is achieved using a graph representation of the transport line and partitioning it in sub-graphs associated to the different pallets by resorting to reachability analysis over the MPC prediction horizon. A reduced MLD

model is determined for each sub-graph by pruning state and input variables, and eliminating redundant equations and constraints from the complete MLD model. If the MPC solutions computed in parallel on the sub-graphs are not conflicting, then, the planned actions are applied. If instead some conflict is detected, sub-graphs are joined together and the reduced MLD model and associated MPC solution of the joint sub-graph are computed. The same procedure is repeated until all conflicts are solved.
Although of general applicability, the proposed strategy is developed with reference to the manufacturing plant located in the laboratory at IT IA - CNR, Gorgonzola (MI), Italy studied in the PhD thesis [3] and also in ([2] and [5]).
Extensive simulations show the effectiveness of the reachability-based decoupling approach.

2. Modeling and MPC Formulation

In this section we shall describe the graph and MLD system modeling the considered manufacturing plant together with the adopted MPC finite horizon cost.

2.1. Manufacturing plant description

The plant is equipped with four machines for de-manufacturing of electronic boards:

- Machine M_1 is the Load/unload robot cell, which load/unload an electronic board onto/from a pallet;
- Machine M_2 is the Testing machine, which identifies the failure mode of the board;
- Machine M_3 is the Reworking machine, which repairs the board;
- Machine M_4 is the Discharge machine, which discharges and destroys non repairable boards.

In order to move the boards from one machine to another, transport modules are employed according to the plant layout in Figure 1.
Operations that can be actuated are listed next:

1. The board is loaded on the pallet by M_1;
2. The transport line moves the pallet to M_2 where the board is tested;
3. The pallet is then moved to M_3 where the failure is repaired, if possible;
4. The pallet is moved back to M_2 and the test is repeated. If the board works properly, it's sent back to be unloaded by M_1; otherwise is sent to M4 where the board is discharged and destroyed.

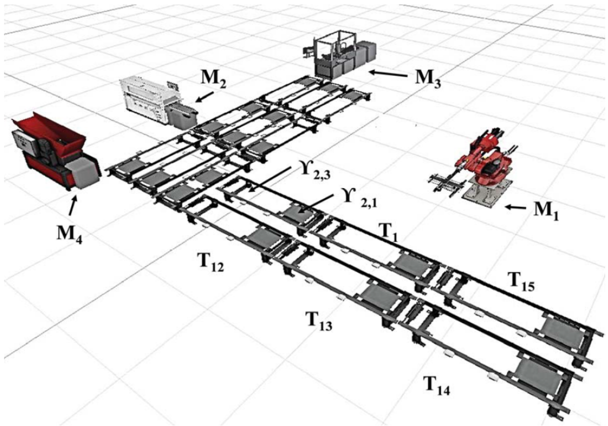

Figure 1: Manufacturing plant layout

The transport line is composed of modules and in each module up to three pallets can lay in three adjacent positions called Buffer Zones *(BZ)*. The actual number of BZs available on each transport module depends on its specific layout configuration: some modules will only use one BZ, some modules will use two BZs and some other modules will use all of their three BZs available.
The transport line can be modeled through a directed graph where active BZs are represented as the circular nodes whereas the four machines are represented by squared nodes. This sums up to a total of 35 nodes, which are progressively numbered and labeled $N_1 - N_{35}$. More specifically, the BZs are labelled $N_1 - N_{31}$ while the machines $M_1 - M_4$ are labelled $N_{32} - N_{35}$. The action of moving the pallet from a BZ to an adjacent one through the control action $u_{i,j}$ is represented in the directed graph by an arrow with label $u_{i,j}$. The graph of the plant is shown in Figure 2.

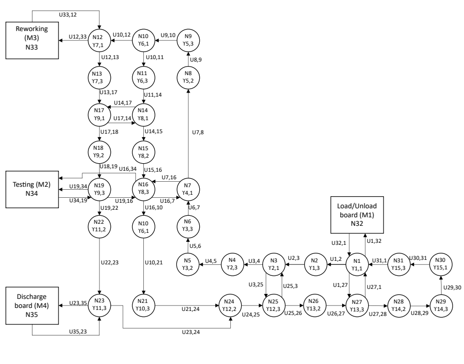

Figure 2: Direct graph of the plant

The event-based evolution of the system over the directed graph representing the plant can be described in terms of the *target state* variables Γ_i, $i = 1\ldots,35$, associated to the nodes N_i, $i = 1\ldots,35$, which take values in $\{0,1,\ldots,5\}$:

1. $\Gamma_i(k) = 0$ if the BZ or the machine corresponding to node N_i is empty at k.
2. $\Gamma_i(k) = j$, with $j = 1,2,3,4$ if the BZ or the machine corresponding to the node N_i contains a pallet with a board to be sent, respectively, to the machines M_1 - M_4.
3. $\Gamma_i(k) = 5$ if the BZ or the machine corresponding to node N_i contains a pallet without any target to be reached.

where k denotes the time step associated with the k-th event occurrence.
Commands $u_{i,j}$ take values in $\{0,1\}$:

$$u_{i,j}(k) = \begin{cases} 1 & \text{if the command is active at } k \\ 0 & \text{if the command is not active at } k \end{cases}$$

Only one control input and/or output can be active at a time; moreover, when a node is empty, no control output can be actuated.
The target state variables associated with the BZs evolve according to the discrete time equation:

$$\begin{aligned} \Gamma_i(k+1) = \Gamma_i(k) + \sum_{j\in I_{i,in}} \Gamma_j(k)u_{j,i}(k) - \\ - \sum_{j\in I_{i,out}} \Gamma_i(k)u_{i,j}(k), \quad (1) \\ i = 1,...,31. \end{aligned}$$

where $I_{i,in}$ or $I_{i,out}$ are the set of indices associated with the input or output commands to/from the node N_i.
A distance is associated to each node N_i, $i = 1,...,35$, through the following function:

$$\begin{aligned} \gamma_i(\Gamma_i) = \zeta_{i,1}(\Gamma_i)\phi_{i,32} + \zeta_{i,2}(\Gamma_i)\phi_{i,34} + \\ + \zeta_{i,3}(\Gamma_i)\phi_{i,33} + \zeta_{i,4}(\Gamma_i)\phi_{i,35} + \\ + (\zeta_{i,0}(\Gamma_i) + \zeta_{i,5}(\Gamma_i))\phi_{i,0}, \end{aligned} \quad (2)$$

where $\zeta_{i,s}(\Gamma_i), s = 0,...,5$ is a binary variable defined as follows:

$$\zeta_{i,s}(\Gamma_i) = \begin{cases} 1 & \text{if } s = \Gamma_i \\ 0 & \text{otherwise} \end{cases}$$

and $\phi_{i,j}$, $j = 32,\ldots,35$, is the minimum distance between the node N_i and the target machine in node N_j and $\phi_{i,0} = 0$.

The permanence of the pallet on the transport line must be penalized. This is done by introducing a counter η_i, $i = 1,...,31$, for each BZ. Its dynamic equation is given by:

$$\begin{aligned} \eta_i(k+1) = \eta_i(k) + \delta_i(k)\vartheta_i(k) + \\ + \sum_{j\in I_{i,in}} [\eta_j(k)+1]\vartheta_j(k)u_{j,i}(k) - \\ - \sum_{j\in I_{i,out}} \eta_i(k)\vartheta_i(k)u_{i,j}(k). \end{aligned} \quad (3)$$

The four machines M_1, M_2, M_3 and M_4, corresponding to nodes N_i, $i = 32,\ldots,35$, are described by a Finite State Machine (FSM) (see Figure 3) with the following three Boolean-valued states:

- $x_{i,1}$: idle and empty machine;
- $x_{i,2}$: manufacturing;
- $x_{i,3}$: end manufacturing, loaded machine;

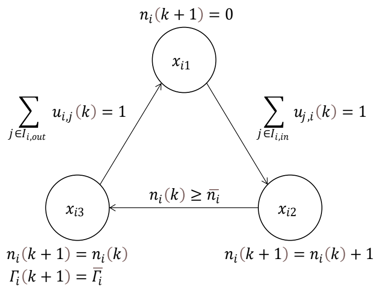

Figure 3: FSM model of a machine.

Letting $\delta_{i,23}(k)$ be a Boolean variable representing the logic condition associated to the transition from $x_{i,2}$ to $x_{i,3}$, i.e.:

$$x_{i,2}(k) \wedge (n_i(k) \geq \bar{n}_i) \leftrightarrow \delta_{i,23}(k) \quad (4)$$

we can express the dynamic equations regarding the pallet target $\Gamma_i(k)$ and the counter $n_i(k)$ associated with the generic machine $N_i, i = 32,...,35$, as follows:

$$\begin{aligned} \Gamma_i(k+1) = \Gamma_i(k) + \sum_{j\in I_{i,in}} \Gamma_j(k)u_{j,i}(k) - \\ - \sum_{j\in I_{i,out}} \Gamma_i(k)u_{i,j}(k) + \delta_{i,23}[\bar{\Gamma}_i - \Gamma_i(k)] \end{aligned} \quad (5)$$

$$n_i(k+1) = [n_i(k) + x_{i2}(k)][1 - x_{i1}(k)]. \quad (6)$$

By using the HYSDEL tool [7], the dynamic model of the plant described in this section can be translated into a MLD system of the following form

$$\begin{cases} x(k+1) = Ax(k) + B_u u(k) + B_\delta \delta(k) + B_z z(k) \\ y(k) = Cx(k) + D_u u(k) + D_\delta \delta(k) + D_z z(k) \\ E_x x(k) + E_u u(k) + E_{aux} w(k) \leq E_{aff} \end{cases} \tag{7}$$

where x is the vector of the state variables, u is the vector of the control actions, δ is the vector of Boolean auxiliary variables and z is a vector of continuous auxiliary variables (see PhD thesis [3] for more details).
According to the MPC approach proposed in the PhD thesis [3] to automatize the transport line, the following optimization problem is solved at every time step k:

$$\text{minimize } J_k \tag{8}$$
$$\text{subject to: (7)} \tag{9}$$

where J_k is a finite-horizon cost over some prediction horizon of length N_{RH}:

$$J_k = \sum_{h=1}^{N_{RH}} \{ \underbrace{\sum_{i=1}^{35} \gamma_i(\Gamma_i(k+h))}_{(a)} + \underbrace{\sum_{i=32}^{35} q_{xi} x_{i3}(k+h)}_{(b)}$$
$$+ \underbrace{\sum_{i=1}^{31} q_{\eta_i} \eta_i(k+h)}_{(c)} + \underbrace{\sum_{(i,j)\in I_u} q_{u_{i,j}} u_{i,j}(k+h-1)}_{(d)}$$
$$+ \underbrace{\sum_{(m,r,i,j)\in\Psi} \lambda_{m,r} \sigma_m(k+h-1) u_{i,j}(k+h-1)}_{(e)} \}.$$

The 5 contributions to the cost J_k represent:
(a) the distance of the pallets from their targets;
(b) the permanence of the pallets in the states $x_{i,3}, i = 1, ..., 4$ of the machines;
(c) the permanence of the pallets on the transport line;
(d) the control actions;
(e) the permanence of a pallet in the nodes adjacent to $M_1 - M_4$.

The last term (e) is introduced to allow the manufactured pallets to exit the machine. It includes the variables σ_m, $m = 32, \ldots, 35$, which are defined as:

$$\begin{cases} \sigma_{32} = 1 \leftrightarrow (\Gamma_1(k) = 1 \vee \Gamma_{32}(k) = 5 \vee \\ \qquad \vee\, \vartheta_{32}(k) = 1) \\ \sigma_{33} = 1 \leftrightarrow (\Gamma_{12}(k) = 3 \vee \vartheta_{33}(k) = 1) \\ \sigma_{34} = 1 \leftrightarrow (\Gamma_{19}(k) = 2 \vee \vartheta_{34}(k) = 1) \\ \sigma_{35} = 1 \leftrightarrow (\Gamma_2 3(k) = 4 \vee \Gamma_{35}(k) = 5 \vee \\ \qquad \vee\, \vartheta_{35}(k) = 1) \end{cases}$$

Summation is taken with indices ranging in the set
$\Psi = \{(32, 1, 27, 1), (32, 2, 31, 1), (34, 1, 7, 16), (34, 2, 15, 16), (34, 3, 8, 19), (33, 1, 10, 12), (35, 1, 22, 23)\}$.

3. Proposed reachability-based decoupling strategy

Starting from the observation that when the pallets are occupying different areas of the plant, they will not interfere, we introduce a reachability-based method to decouple the MPC optimization problem in multiple lower-dimensional problems that can be solved in parallel, each one defined on an appropriate reduced model of the MLD system modeling the overall plant.
This involves the following steps:

1. perform reachability analysis on the directed graph representing the plant so as to identify the nodes that each pallet can reach within the prediction horizon;
2. determine the reduced MLD systems modeling the sub-graphs by pruning variables and constraints from the complete MLD system;
3. compute the MPC solution for every reduced MLD system;
4. check whether or not the optimal control actions to be applied at the current time step create any conflict. If no conflict is found, then the computed actions are applied according to the receding horizon strategy. Otherwise, the sub-graphs of the conflicting pallets are joined together and a new MPC problem is defined for the resulting MLD model. This step is repeated until there are no conflicts.

Certain zones of the plant are prone to deadlock situations, we shall then group together the pallets that are within these *critical zones* and perform reachability analysis for them jointly since

the very first iteration of the decomposition algorithm.
In order to perform reachability analysis, we shall consider the directed graph representation of the plant in Figure 2.
We shall associate to this graph with $n = 35$ nodes a 35-by-35 *adjacency matrix* A whose elements satisfy

$$a_{i,j} = \begin{cases} 1, & \text{if there is an arrow from } N_j \text{ to } N_i \\ 0, & \text{otherwise.} \end{cases}$$

Due to the self-loops at each node, we then have that $a_{i,i} = 1$, $i = 1, \ldots, 35$.
In order to determine what are the nodes that a subset of m_{sub} pallets out of a total of $m \geq m_{sub}$ can possibly reach over the prediction horizon N_{RH}, we just need to define a column vector X_0 with all elements equal to zero except those corresponding to the position of the pallets that are set equal to 1 and compute

$$X_{N_{RH}} = A^{N_{RH}} X_0$$

The nodes that can be reached by some of the considered pallets in the look-ahead time horizon of length N_{RH} are those corresponding to the elements in $X_{N_{RH}}$ that differ from zero.
Since reachability analysis needs to be performed for multiple subsets of pallets, it can be done in parallel by defining X_0 as a matrix instead of a vector, with as many columns as the number of considered subsets.
Before dividing the problem into smaller ones, a preliminary analysis must be conducted. Indeed, the pallets in the so-called off-limit and axchange zones will need to be grouped together in the first place in order to avoid deadlocks and/or save computational resources.

Off-limit zones

When a machine has completed the processing operations, it should be able to take the pallet carrying the board to its output node, which then must be empty. However, every output node for a machine is also an input node for that same machine. If the pallets move independently, then some pallet could end up being stuck in the output node of a machine while it is busy in its working state; this positioning could prevent the machine from ever dispatching the processed pallet and, at the same time, could prevent the pallet from ever entering the machine. In order to solve this problem, we shall define *off-limit zones* associated to pre-defined sub-graphs within the transport line that need to be preserved in the MLD reduced model. Off-limit zones are highlighted in red in Figure 4.

Exchange zones

An *exchange zone* is defined as a sub-graph containing a couple of adjacent nodes where a pallet is able to move from one node to the other and vice versa. There are several examples of exchange zones in the plant: each machine and their adjacent node constitutes an exchange zone, since it is possible to travel from the node to the machine and vice versa. Most of the exchange zones happen to be inside of the already established off-limits zones, so the exchange zones within an off-limits zone are discarded, since there is already an MPC coordinating the traffic inside an off-limits zone. Two zones happen to be outside of every off-limits zone and are depicted in yellow in Figure 4.

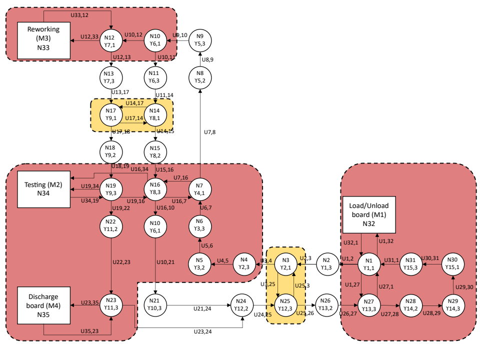

Figure 4: Exchange zones and off-limit zones of the plant.

3.1. Deriving the MLD system of a sub-graph

The goal of this section is to describe how to tie every variable and constraint of the system to the node they refer to, such that when the reachable set of nodes of every sub-graph with at least one pallet is identified, the variables and constraints to be pruned from the MLD system (7) modeling the overall plant will be known as well. This will then lead to the derivation of the reduced MLD system of the sub-graph.
Once the associations with the nodes have been performed, it is necessary to derive the MLD

model of the reduced systems. The process begins by considering the matrices of the MLD system, as defined in (7). Once the reachable nodes are known, the matrices are reduced by removing the rows and the columns of the variables associated to the non-reachable nodes, thus reducing the size of the matrices. Moreover, all of the control actions of the non-reachable nodes are set to zero.

State Variables

Since the number of state variables is greater than the number of nodes, the nodes will be associated with more than one state variable. Each node N_i, for $i = 1, \ldots, 31$ is associated to two variables: the Pallet Target $\Gamma_i(k)$ and the counter $\eta_1(k)$; then, each one of remaining nodes, that is N_i for $i = 32, \ldots, 35$, are associated to to the Pallet Target $\Gamma_i(k)$, the machine counter n_i and the three state variables for the target automata modelling $x_{i,1}$, $x_{i,2}$, $x_{i,3}$.

Input Variables

Each node is associated with a subset of the control actions. Since there are a total of 50 input variables for only 35 nodes, each node is associated with one or more control actions. Once the associations are done, all the input associated to the nodes reached within $N_{RH} - 1$ steps will be preserved.

Output Variables

The output variables are defined in Equation 2 as a piecewise affine function of the BZ's distance from the target to be reached. In order to prune the output variables, it is sufficient to know the nodes that will be reached within N_{RH} steps from the initial position. The only output variables that will not be pruned are the ones tied to the reachable nodes.

Auxiliary Variables

Before pruning the auxiliary variables, they must be associated with the node they refer to. By searching in the HYSDEL file for all occurrences of a certain auxiliary variable name within the constraints and equations, it is possible to deduce the corresponding variable in the MLD model and, finally, the node in the graph of the plant transport line to which the variable refers. By following this reasoning, it is possible to create a map associating one or more auxiliary variables to each node of the plant.

Constraints

The previous sections demonstrate how to associate the state, input, output and auxiliary variables to each node of the system. Once the reachable nodes are known, all the variables associated with these nodes must be collected into a single set, called V_r. Then, a set called V_i with $i = 1, \ldots, 2904$ is defined as the set of all the variables involved in the i-th constraint. Whenever $V_r \cap V_i \neq \emptyset$ for $i = 1, \ldots, 2904$, the i-th constraints must not be pruned because it contains and affects one or more of the variables associated to the reachable nodes. Once the intersections have been evaluated, we must select all the indices for which we have a non-zero intersection.

4. Simulation Results

Two indexes are adopted to analyse the performance of the proposed approach and compare it with the original MPC method in the PhD thesis [3] via extensive simulations. The YALMIP toolbox for MATLAB has been used for the MILP problem definition and IBM ILOG CPLEX solver for its solution. The software implementation of the plant is discussed in [4].

4.1. Online optimization computation times

Simulations are run for different values of the prediction horizon N_{RH} and of the number m of pallets on the transport line. For each pair (N_{RH}, m), MPC is applied for 100 steps. Since the input applied at each step is determined by first decomposing the system into smaller ones and computing their MPC solution in parallel, then checking for possible conflicts and possibly joining the small systems to repeat the parallel MPC solution and the conflict check, the computation time per step is given by the sum of the worst time for the MPC solution over all iterations needed to eliminate conflicts.
The time saving of the approach described in this thesis is shown in Figure 5. The four plots represent the average online computation times

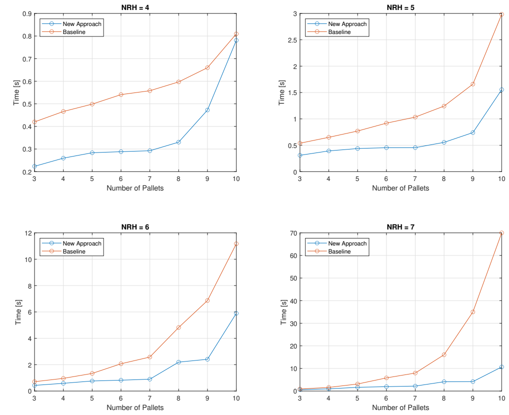

Figure 5: Average computation time as a function of the number of pallets for every tested value of N_{RH} for the MPC baseline approach of the PhD thesis [3] (orange line) and for the new approach in this thesis (blue line).

for the four values of the prediction horizon as a function of the number of pallets.
The approach described in the PhD thesis[3], highlighted in orange in the four images of Figure 5 provides the baseline; the approach developed in this thesis is depicted in blue in the same figure and labeled as 'New Approach'.
The top left plot corresponding to $N_{RH} = 4$, shows a limited difference between the new approach and the baseline approach which decreases as the number of pallets increases. The difference increases progressively in the other three plots, becoming significant in the bottom right figure, showing that the efficacy of the baseline approach is hampered by the excessive computational complexity incurred for a large number of pallets and a large value of the predictive horizon.

4.2. Number of machined pallets

We now compare the number of pallets processed with the two approaches over a long period of time. The approach developed in the PhD thesis [3] and the approach developed in this thesis are run for a total of 2000 time steps. The number of pallets that are processed by each machine after 2000 steps are then compared.
It should be noted that the number of pallets processed after 2000 steps with the new approach is not less than the number of pallets processed in the approach considered as a base-

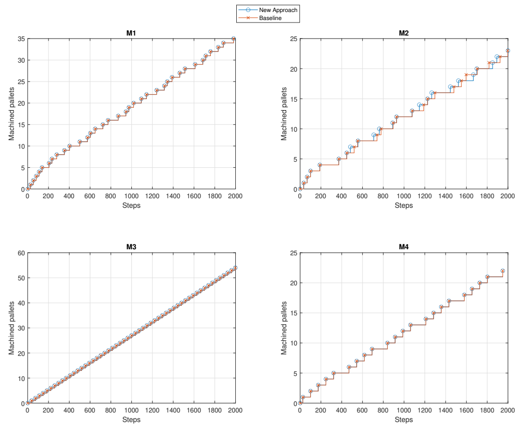

Figure 6: Number of machined pallets performed by M_1 (top left), M_2 (top right), M_3 (bottom left), and M_4 (bottom right).

line, thus demonstrating that the new approach does not suffer from a decrease in throughput in the long run nor it does incur in deadlocks.

5. Conclusions

Motivated by the need of automatic routing in smart manufacturing and inspired by a recent MPC approach proposed in the literature, we developed in this thesis a reachability-based approach for decoupling the MPC optimization problem for pallet routing into multiple smaller ones to be solved in parallel. This reduces the computational complexity of the MPC problem, especially in the cases where both the prediction horizon N_{RH} and the number of pallets m on the transport line assume large values, thus making computing times compatible with the transport system dynamics.
In order to get a more efficient implementation, however, one should find a methodology to automatically identify those critical zones in the plant that call for a grouping of the pallets that are located therein. A learning by doing approach could be adopted by simulating the system and observing possible deadlock conditions or recurrent grouping of pallets located in certain areas.

References

[1] Alberto Bemporad and Manfred Morari. Control of systems integrating logic, dynamics, and constraints. *Automatica*, 35(3):407–427, 1999.

[2] A Brusaferri A, Marcello Colledani, G Copani, M Sacco N Pedrocchi, TULLIO ANTONIO Tolio, et al. Integrated demanufacturing systems as new approach to end-of-life management of mechatronic devices. 2012.

[3] Andrea Cataldo. *Model predictive control in manufacturing plants*. PhD thesis, Politecnico di Milano, Italy, 2017.

[4] Andrea Cataldo and Riccardo Scattolini. Logic control design and discrete event simulation model implementation for a demanufacturing plant. *Automazione-plus online Journal*, 2014.

[5] Marcello Colledani, Giacomo Copani, and Tullio Tolio. De-manufacturing systems. *Procedia CIRP*, 17:14–19, 2014.

[6] A Resano Lázaro and CJ Luis Pérez. Dynamic analysis of an automobile assembly line considering starving and blocking. *Robotics and Computer-Integrated Manufacturing*, 25(2):271–279, 2009.

[7] F.D. Torrisi and A. Bemporad. HYSDEL - a tool for generating computational hybrid models for analysis and synthesis problems. *IEEE Transactions on Control Systems Technology*, 12(2):235–249, 2004.