Details About Agent Response in "Cooperative Product Agents to Improve Manufacturing System Flexibility: A Model-Based Decision Framework"

Ilya Kovalenko, Efe C. Balta, Dawn M. Tilbury, Kira Barton

February 28, 2022

1 Introduction

This document overviews a deliberation methodology for product agents (PAs) and resource agents (RAs) to respond to violation requests from the direct, actively cooperating product agent (DA-PAs) described in:

> I. Kovalenko, E. C. Balta, D. M. Tilbury, and K. Barton, "Cooperative Product Agents to Improve Manufacturing System Flexibility: A Model-Based Decision Framework," submitted to *IEEE Transactions on Automation Science and Engineering*.

In addition, this document talks about DA-PA response to unexpected faults and disturbances in the system.

2 Coordination with a product agent

The direct, actively cooperating product agent (DA-PA) can send a re-planning or re-scheduling request to other PAs in the system. As part of this request, the DA-PA provides the desired delay transitions that are in its optimal path (for more information, see Section IV of [1]). To make a decision whether to authorize the constraint violation, the contacted PA, $ag_{pa} \in Agents$, must reason about the effect the constraint violation will have on its planned sequence of actions (i.e. path). If the decision making of ag_{pa} is rule-based, then ag_{pa} must have an appropriate rule that allows it to search for alternate paths in the system. Similarly, if ag_{pa} is a model-based PA, the the new scheduling constraints should be encoded in the environment model of the PA. If an alternate path is not found, then the rule-based or the model-based PA rejects the request from the DA-PA.

Since a DA-PA is classified as a model-based PA, the decision making framework proposed in this work can be used to find alternate paths in the system. The sequence diagram in Figure 1 shows what happens when one DA-PA, ag^2_{pa}, sends a violation request to another DA-PA, ag^1_{pa}. In this example, ag^1_{pa} receives a request with new scheduling constraints from ag^2_{pa} and progresses through the model creation, path planning, and coordination phases shown in Figure 3 of [1].

The rejected violated constraints are added as hard constraints to the decision making model of ag^1_{pa} during the model creation phase. Then, during the path planning phase, ag^1_{pa} solves the optimization problem shown in (7) in [1]. Note that to prevent the violation request from cascading to other agents in the system, ag^1_{pa} is not allowed to request constraint violations from other agents in the system, i.e., set ζ to 0 for (7d) in [1]. Future work will determine how to allow for relaxing this cascading limitation by allowing ag^1_{pa} to request constraint violations during this path planning phase. The coordination phase of ag^1_{pa} consists of a

This work is supported in part by the National Science Foundation (including CNS Grant #1544678 and DGE #1256260), a gift from Rockwell Automation, and NIST Award No.70NANB19H090.

I. Kovalenko is with Department of Mechanical Engineering and Industrial & Manufacturing Engineering, Pennsylvania State University, University Park, PA. {iqk5135}@psu.edu D. M. Tilbury and K. Barton are with the Department of Mechanical Engineering, University of Michigan, Ann Arbor, MI, USA. {tilbury, bartonkl}@umich.edu. E. C. Balta is with the Automatic Control Laboratory, ETH Zurich, 8092 Zurich, Switzerland. ebalta@ethz.ch

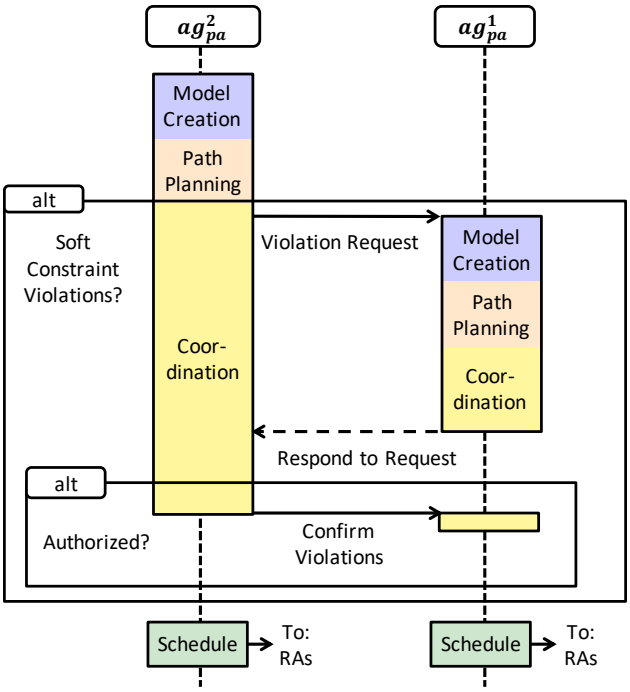

Figure 1: A sequence diagram for negotiation between two direct, actively cooperating product agents, ag^1_{pa} and ag^2_{pa}.

reply to ag^2_{pa} to authorize or deny constraints based on whether or not a path was found in the path planning phase.

3 Coordination with resource agents

The DA-PA may request constraint violations from the RAs in the system. In practice, the RAs often have a desired, optimal time to complete various resource actions based on their own resource models and collected production data. However, resources can often accomplish tasks faster if prompted. While this might be sub-optimal to the individual RA, it might help PAs in the system to accomplish their goals. For example, if a DA-PA requires a faster-than-usual transfer between two locations in the system, a material handling robot can accomplish this task at the cost of higher energy expenditure.

A soft constraint that is associated with an RA represents the RA's desired, optimal time required to finish the associated resource actions. However, when necessary and if beneficial to the DA-PA in the system, the constraint can be violated during a DA-PAs cooperation step. Therefore, once queried, an RA must decide whether to authorize or deny a violation request as the request. While RA decision-making is not within the scope of this work, the RA must weigh the benefits and disadvantages of violating a constraint with respect to its associated resource. Then, the RA makes decisions whether to allow the DA-PA to violate a constraint and communicates this decision with the DA-PA.

4 DA-PA Response to Disturbances, Faults, and Failures

The DA-PA will re-plan and re-schedule when there are disturbances, faults, and failures in the system that affect the associated physical part in the system. Examples of physical disturbances and faults include resources not finishing the requested action within the requested amount of time or resources breaking down [2]. Example of software failures include if an optimization problem cannot be solved by the DA-PA or if the DA-PA does not solve the problem in a specified bounded time.

As described in Section V of [1], the DA-PA will restart the proposed decision-making methodology when a fault, disturbance, or failure is identified. The DA-PA will re-explore the system to re-create the environment model, and proceed with the steps in the cooperation framework. The process of restarting each of the steps in the cooperation framework allows the DA-PA to capture and respond to faults and unexpected disturbances in the system. For example, if a resource goes down in the system, a resource takes longer to perform a requested task, or a new resource is added to the system, an RA will inform the DA-PA of this disturbance. Using this information, the DA-PA will capture this information during a new model creation stage [2, 3]. Then, the DA-PA will re-plan and re-schedule based on the new environment model. During the new model creation phase, the DA-PA can also obtain the schedules of new parts in the system and use this information to re-plan and re-schedule through communication with RAs in the system. Another possible unexpected disturbance is if a customer wants to change the process plan. Similar to the previous disturbances, this request can be handled by creating a new environment model [3] and changing the desired states (marked states) during the path planning process. An example of an unexpected failure in the DA-PA decision making process is if the DA-PA cannot find a plan or cannot find a plan within a bounded time. If this occurs, the DA-PA will also try to build a new environment model and re-try running the optimization solver.

Note that if the DA-PA restarts the process and cannot find a solution within a bounded time or after a specific number of iterations, the DA-PA will use its exit plan to call an agent to help with the decision making and/or leave the manufacturing system. Future work will incorporate building a flexible scheduling policy that will be used prior to calling the exit plan.

References

[1] I. Kovalenko, E. C. Balta, D. M. Tilbury, and K. Barton, "Cooperative Product Agents toImprove Manufacturing System Flexibility: A Model-Based Decision Framework," *IEEE Transactions on Automation Science and Engineering*, In Submission, 2021.

[2] I. Kovalenko, D. Tilbury, and K. Barton, "The model-based product agent: A control oriented architecture for intelligent products in multi-agent manufacturing systems," *Control Engineering Practice*, vol. 86, no. March, pp. 105–117, may 2019.

[3] I. Kovalenko, D. Ryashentseva, B. Vogel-Heuser, D. Tilbury, and K. Barton, "Dynamic Resource Task Negotiation to Enable Product Agent Exploration in Multi-Agent Manufacturing Systems," *Robotics and Automation Letters*, vol. 4, no. 3, pp. 2854–2861, jun 2019.