Smart Manufacturing Control with Cloud-embedded Digital Twins

Theodor Borangiu
Dept. of Automatic Control and Industrial Informatics
University Politehnica of Bucharest
Bucharest, Romania
theodor.borangiu@cimr.pub.ro

Silviu Răileanu
Dept. of Automatic Control and Industrial Informatics
University Politehnica of Bucharest
Bucharest, Romania
silviu.raileanu@cimr.pub.ro

Andrei Silişteanu
Dept. of Automatic Control and Industrial Informatics
University Politehnica of Bucharest
Bucharest, Romania
andrei.silisteanu@cimr.pub.ro

Silvia Anton
Dept. of Automatic Control and Industrial Informatics
University Politehnica of Bucharest
Bucharest, Romania
silvia.anton@cimr.pub.ro

Florin Anton
Dept. of Automatic Control and Industrial Informatics
University Politehnica of Bucharest
Bucharest, Romania
florin.anton@cimr.pub.ro

***Abstract* — The paper presents a model for smart control of large scale manufacturing systems, in which pools of shop floor and computing resources are shared in a dual cloud pattern. The proposed architecture uses the holonic manufacturing paradigm by decoupling the decision layer from the control one. The decision layer uses intelligent agents that reconfigure optimally in real time the resource allocation and scheduling of operations on products at batch level; also, the resource health is monitored continuously. Decisions are taken in the high layer of the MES based on real time machine learning algorithms that predict resource performances and QoS influencing usage costs, classify and cluster resource states to predict anomalies in behaviours and prevent resource failures. The distributed control layer keeps reality awareness during production by using digital twins replicated for all resources. Data is collected in real time streams from physical resource and process twins, aggregated in time series and sent to the intelligent agents in the cloud without delaying production. Experiments discuss the forecast of abnormal pick-and-place robot operations.**

Keywords — intelligent manufacturing; digital twin; machine learning; predictive control; cloud manufacturing.

I. Introduction

The digitalization of production and logistics control, main operations in manufacturing, and of after-sales services in the networked enterprise value chain is recognized nowadays as the priority of industry to build up the 'Factory of the Future' (FoF) in the 'Industry of the Future' vision. FoF labels a new generation of manufacturing structures aimed to be smart, safe, hence sustainable [1]. In smart manufacturing structures production is optimized at batch level considering global cost functions such as makespan, energy consumption, resource usage, zero defect parts, and reality-awareness: fault tolerant and robust controls, highly available resources and process resilience [2]. On global markets smart manufacturing firms must be agile to product changes, allow mass customization through flexibility, in-depth interoperability, and rapid shop floor re-engineering through resource reconfiguring [3, 4].

Artificial Intelligence (AI) is used in smart manufacturing to develop algorithms and techniques that give an enhanced insight on data from resources, processes and products and improve decisions to: optimize batch costs, inventory levels and energy consumptions, automate manufacturing processes and material flow transport / handling, improve asset usage and risk management. AI can be implemented in production planning and control / in process and product design by four technologies [5]: 1) *big data*: smart sensors pervasively instrument resources, products, processes and orders as 'plug-and-produce' modules [6]; 2) *platform*: hardware aggregation nodes and middleware aligning data streams in normalized time intervals and transferring map/reduce data in the cloud; 3) *analytics*: the application of statistics and machine learning to detect deviations in process and resource parameters, state and behaviour patterns, and to predict the quality of their services; 4) *operations*: batch production control, global cost optimization, energy saving, predictive resource maintenance using analytics tools; product-service extension; product redesign from lifecycle feedback [7], [8].

Although big data processing is considered in this case as a technology enabling the management of the contextual enterprise on the business layer, its usage in manufacturing shop floor control of resources and processes was until now less evident, mainly due to the few real time industrial big data streaming techniques and the prohibitive cost of high performance computing (HPC) resources. Related to this aspects, [9] offers a review of big data analytics throughout product lifecycle that includes the production stage.

Machine learning (ML) and big data technologies have got greater importance due to the outbreak in the data points available in the shop floor through edge computing and the HPC power on hand by applying the cloud computing model to manufacturing control. They started to be used at present for predictive resource maintenance [10].

Using cloud services for big data analytics and intelligent decision making through machine learning in Manufacturing Execution Systems (MES) is based on the virtualization of shop floor devices and a new control and computing mode that operates in real time in the global Cloud Manufacturing model (CC-CMfg). The CC-CMfg services orchestrate a dual OT (control) and IT (computing) model that:

- Transposes pools of shop floor resources (robots, CNC machines), products (recipes, client specs.), orders (work plans, task sequences) into *on-demand* making services;

- Enables pervasive, *on-demand network access* to a shared pool of configurable HPC resources (servers, storage, applications) that can be rapidly provisioned and released as services to various high level MES tasks with minimal management effort or interaction [11]. Hence, CC-CMfg may use cloud computing facilities for smart control.

In the scope of smart manufacturing control, the CC-CMfg model takes over and adapts some cloud computing features: a) the product-making services are provisioned automatically by a MES optimal resource allocation program run in real time; b) the cloud computing (CC) component offers network access to HPC services through a distributed message platform / manufacturing service bus (MSB); c) The shop floor resources of the CMfg component are placed in clusters with known location relative to the material flow, and dynamically assigned at batch run time; this location is one input parameter weighting the optimal resource allocation; d) the CC services can scale rapidly in order to sustain the variable real time computing demand for order rescheduling respectively anomaly detection, the resources being assigned or released elastically; e) the assigned CMfg resources are monitored and controlled and both the MES (service consumer) and the Cloud (service provider) are notified about the usage within the smart control application; the cost model 'pay as you go' is used to establish the cost offers for client orders in the service level agreements.

The paper proposes a framework for smart manufacturing control systems capable of *cost optimization* at batch level and *reality-awareness* of: resource QoS decline, unexpected events and workspace disturbances. Machine learning (ML) techniques are used: prediction of resource performances based on history and real-time KPI computing from shop floor measurements to optimize batch costs, classification and clustering to detect anomalies in resource behaviour. This framework uses digital twins embedded in production control and resource maintenance tasks. Big data is collected in real time streams at shop-floor edge, pre-processed and sent to provisioned cloud resources for global MES tasks.

The remainder of the paper is organised as follows: Section 2 defines the smart holonic control model, specifying generic entities and their relationships. Section 3 describes the 6-layer, hierarchical architecture of the digital twin and its distribution on IoT data processing and decision making areas in the cloud. Section 4 presents the ML approach and the connectivity of the distributed MES with its centralized cloud layer. Experimental results and conclusions are given in Sections 5 and 6.

II. SMART HOLONIC MANUFACTURING CONTROL MODEL

The smart control model is applied to a holonic MES which is part of the layered manufacturing framework, Fig. 1.

The Service Oriented Architecture (SOA) of the enterprise Holonic Manufacturing System (HMS) shown in Fig. 1 aligns the business information flow on the *business layer* (relating the offer request management module with the customer order management module) with the technology-specific information flow - the latter being partitioned in:

- The *MES layer* is responsible for global, batch level actions like: production planning (schedule operations and allocate resources), production monitoring (dispatch orders, track job execution, monitor resource health and work in process) and quality control using the product knowledge base.
- The *shop floor layer* is in charge with the control of parallel tasks assigned by the MES to the active resources of an a priori configured CMfg pool, and with the real time update of resources' state, performances and quality of services.

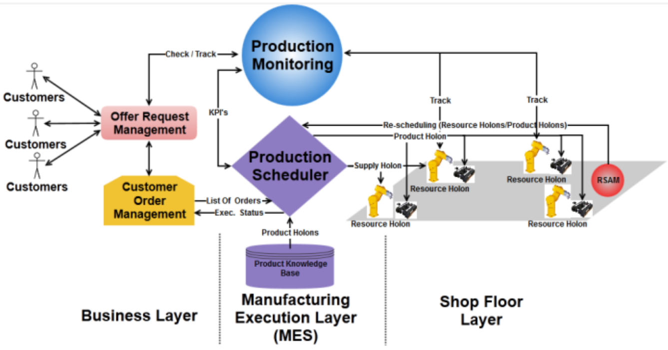

Fig. 1. The MES layer – part of the Manufacturing Integration Framework

The holonic approach enforced on the MES and shop floor layers is transposed in a distributed control architecture based on a set of abstract entities: resources (technology, humans - reflect the producer's capabilities, skills), products (reflect the client's needs) and orders (reflect the business solutions and constraints), represented by autonomous holons that cooperate for a common production-related goal at batch level [12].

These three classes of basic holons are created by the MES and operate on production control layer by exchanging knowledge about the production processes and resource sharing. Thus, the *product holons* (PH), one for each product type in a batch, receive data and schemes on how to perform the product making processes; this information is transferred to the *resource holons* (RH) that represent those resources selected to perform operations based on their capabilities, time-varying performances and QoS. The resources' state and behaviour are updated online in a Resource Service Access Model (RSAM in Fig. 1). The process knowledge encapsulated in one PH is thus related to the process execution knowledge held by several RHs included in the CMfg pool that qualify for allocation: making resource reservations; starting, interrupting and resuming application programs; monitoring tasks and processes. *Order holons* (OH) transpose process knowledge acquired from a PH in batch production data: order of product entry, operations schedule / product and assignment of resources / operations.

The MES layer includes *staff holons* (SH) that assist the basic holons by optimizing at batch horizon the order schedule and resource allocation [13]. The SH principle uses centralized HPC elements that are accessed in CC mode; the smart control solution proposed assumes that SHs deliver recommendations that are used as long as no significant deviations in resource QoS are computed from measurements and predictions or no anomalies are detected. When such changes are estimated, the production planning is relocated in heterarchical mode to the basic holons.

The smart manufacturing control model proposed is shown in Fig. 2. This model uses the holonic reference architecture ARTI (Activity-Resource-Type-Instance) [14] recently defined for industrial applications; it operates with digital twins that are mirroring reality in the control strategy represented as a cube.

The basic and staff holon types are originated from highly-abstracted, generic entities: ‘activity’, ‘resource’, ‘instance’ and ‘type’ of the ARTI holonic reference architecture that focuses on the relationship between production (material) reality and its virtual representation, mirroring and smart control. A PH stems from the category *activity type* while OHs, as *activity instances*, are computed from the PH by a System Scheduler according to a pattern of interest (cost efficiency, energy saving, balanced resource usage – at product / batch level). RHs represent *resource instances* (SCARA robots, CNC machines, cloud blade server) of *resource types* (technology, people) assigned to OH tasks, respectively to PH operations.

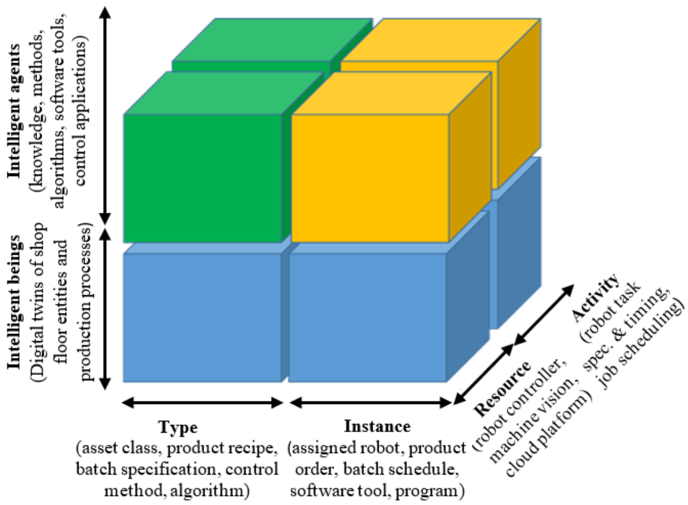

Fig. 2. The smart manufacturing control model uses the ARTI architecture

The smart control mode proposed aims at: 1) *optimizing* the ensemble of activities to make all products in an ordered batch; 2) operating resources in *reality-aware* mode: track their states, performance and QoS, detect anomalies and unforeseen events.

Optimization refers to global cost functions calculated from batch execution time and electrical energy consumed. Energy is computed from electrical power absorbed in time; the instantaneous power of resources is measured and integrated on time periods in which operations on products are completed. In this smart control model, optimization is based on a semi-heterarchical computational, 3-tier scheme that: a) plans the order of product entry; b) schedules operations on products; c) assigns resources to operations [15]. Two resource types are intertwined in this scheme:

- A *centralized* System Scheduler (SS) that deploys cloud services on demand on the high-level MES layer with HPC capabilities for global, batch OH (re)-scheduling.
- A *heterarchical* cooperation mechanism run by the ad hoc created multi-agent system (MAS) that includes the OHs of all products currently in execution in the shop floor; the set of all the temporarily created OH clusters with context-dependent composition build up the *distributed* MES layer (dMES).

By continuously collecting resource data (execution times, work parameters, instantaneous power) and integrating it at operation level, the efficiency of the current resource allocation for product making can be evaluated in real time, and eventually changed. Two types of changes in resource- and product-KPI computed from the measured parameters have been defined: hard changes and soft changes. Their occurrence will induce different updates in the smart control strategy for cost optimization.

Hard changes of state (e.g., resource breakdown, failure of an operation) may occur at any time; they invalidate the current 3-tier production scheduling scheme, since affected resources are discharged from the CMfg pool. In this case it is mandatory to launch two actions: 1) delegate the agent cluster representing the OHs in execution to negotiate the assignment of remaining valid resources to jobs; 2) query the cloud SS to compute a new optimal schedule for the products waiting to be introduced in the shop floor and processed by the valid resources. These actions are executed concurrently.

Soft changes of resource KPIs (small variances in the operation duration or energy consumption) are most frequent unforeseen changes; they are prone to occur more often than hard changes and alter the resource usage cost in operations on products as part of the total batch cost (e.g., due to bigger execution time or more energy consumed). In order to directly relate resource use to production costs, these changes are evaluated by integrating measured data on operation execution periods. Small increases of operation execution time or energy consumption will cause most likely minimal increase of the makespan; the *decision* for a new run of the optimization algorithm is taken only at changes that exceed a threshold set up experimentally. Orders in the waiting queue for insertion are scheduled in the cloud by an optimization algorithm run by the MES SS whose input is the cost of resource usage updated per operation in the RSAM. The costs weight down the resource selection in the CMfg model.

In order to prevent the system to become too nervous through frequent updates of the global schedule, the *decision* to apply a new global schedule will be applied when a product is completed and only if the cloud optimization algorithm yields much better cost results than its actual instance [16]. The SS is part of the centralized staff holon in the cloud; the output of the optimization program is a new set of OHs that exert in parallel the *distributed control* of the production process.

Decoupling decision making from the shared CMfg control is the most important feature of smart manufacturing control, dividing *intelligent* scheme entities in *agents* and *beings*. The first are classified into *types* (optimization algorithms, learning schemes, prediction patterns) and *instances* (software tools and programs, languages, APIs, controls) that apply concepts.

The batch optimization problem is solved in real time with the CPLEX mathematical optimization algorithm (the SH *type*) implemented with IBM ILOG mathematical optimization soft-ware and formulated in OPL programming environment (the SH *instance*). ILOG CPLEX constructs a solution sequentially and improves it continuously [17]; this allows assessing the time in which a feasible solution is computed for a given search space (shop floor layout, order complexity, batch size). This interval is used in conjunction with the moment when an order (product) will be completed (output of last program run) to decide when the consistency of soft changes must be evaluated, the optimization model’s input updated and the CC program started. The timed smart decision making procedure is given in the pseudo-code:

s0. *Off-line run the 3-tier optimization program for new batch with model input from history; record the running time (tr)*

s1. *Select the next optimally planned order, start its execution and remove it from the batch*

s2. **If** *there are more orders available and the shop floor layout still allows introducing new ones to be executed* **GO TO** *s1*

s3. *Compute when next order is completed (to); Wait to – tr*

s4. *Wait event: completion of a new operation; update KPIs of resource for this type of operation*

s5. **If** *there are still orders to be executed and the input model has consistent changes* **then** *run the optimization program*

s6. *Compare the global cost results of the new run with actual ones;* **if** *the new ones are better* **then** *update schedule*

s7. *GO TO s1*

The blue layer of the smart holonic control cube complies with the theory of flexibility (design for reality awareness). The real time picture of states and QoS of 'activity' and 'resource' instances including the working context, operating modes and process models is captured by digital twins that embody blue ARTI cubes – the *intelligent beings*. This second objective of the smart control is reached embedding *digital twins* in prediction-based detection of anomalies and unforeseen events by help of machine learning techniques.

III. Digital Twins for Predictive Control

Digital Twins (DT) of production assets (resources, products, orders) and system (control, maintenance and tracking) are key elements of the smart control model; they build up the middleware layer linking the physical (shop floor) twin with its virtual control applications, and offer a complete view of past, actual and future states and behaviours of resources, processes and outcomes. In-depth interoperability of the system's elements is enabled by the aggregate DT [18].

The DTs constitute the blue cubes of the smart control model. They are connected to their physical counterparts in a direct, one-to-one relation; they are used to predict resource KPIs for usage cost, to weight the participation in bids for jobs, to detect anomalies and predict maintenance. Thus, the control software can be brought closer to shop floor reality, leading to safe, stop-free production. Fig. 3 shows the 6-layer digital twin embedded in the smart control tasks - batch optimization and resource health maintenance. The tasks are performed using HPC CC manufacturing cloud services [19].

Raising reality awareness for shop floor resources needs to collect online data from their physical parts and served process (layer I), aggregate and analyse the data streams (layer II) and classify states, predict QoS and usage costs / product operation (layer III). Forecasting unexpected events, operating anomalies and decisions for customized predictive maintenance are performed in layer IV. These layers are replicated for all active resources; the forecasted KPIs are transferred to layer V for predictions on product topic for real time batch optimization computed in layer VI.

The first two lower level Digital Twin layers are related to the structure of 'Resource' and 'Activity' (tasks performed by resources) blue cubes defined in ARTI as 'intelligent beings'. The high level DT III and DT V layers embody the green cubes 'intelligent agents' with ML algorithms to forecast resources' behaviour, classify health states (DT III is replicated for all the members of the CMfg pool of active resources) respectively to predict the cost evolution of batch schedules (DT V reuses the forecasts of resource usage costs made in DT III). The high-rank DT IV, DT VI incorporate decision-making tasks respectively for predictive resource maintenance and batch optimization. DT features are listed:

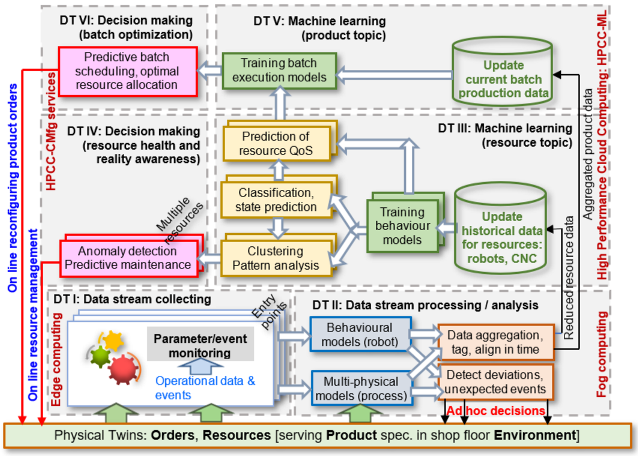

Fig. 3. 6-layer digital twin embedded in the decision-making process of context-aware, predictive resource maintenance and allocation

DT I: Collecting data streams - is linked to the physical twins: resources' components, served processes and operating environment, and collects data (parameters and events) from multiple entry points. The variable timing for acquisition and message filtering at the shop floor edge of the manufacturing control scheme induces important software constraints in DT I.

DT II: Processing and analysis of data streams - has the role to separate and align the data streams produced in DT I in standard time slices, and to group them using covariance rules. DT II is organized in four modules:

- *Repository of resource performance models*: includes data about the way in which resources operate; these patterns are created to manage properly different working modes (e.g. automated, dry-run). Some behaviour parameters should be defined off-line and eventually updated on line.
- *Repository of multi-physical activity models*: holds patterns of the multiple processes served by resources.
- *Aggregating data aligned in time*: merging raw data streams labelled in conformity to the two patterns above in task-specific streams. This assumes the distribution of big data join/merge operations for parallel runs on multiple nodes in map-reduce clusters. The runs are done in series of micro-batches in time slices set up by the rate of stream messages. Multiple aggregations use various map-reduce keys, for example labelling messages with their position in the shop floor, energy consumer place or heat circulation.
- *Detection of anomalies and unforeseen events*: produces immediate decisions to stop and isolate faulty resources.

DT III, V: Machine learning - extracts on line insights from the aggregated data streams originating from the shop floor resources, processes and workplace environment. The computation is parallel for all resource channels and product orders. DT III uses three ML techniques: *prediction* with neural networks: deep learning patterns and measurement variations (e.g. energy), and then forecasting the evolution of values; *classification*, when DT III finds classes for feature vectors received; *clustering*, which searches and identifies similarities in non-tagged, multiple-dimension data and tags suitably each feature vector. Recording history data storage is needed to train ML patterns. Based on continuous update of product data (operations started / finished, durations, changed precedencies, quality tests passed / failed) and of predicted QoS of resources, batch cost models are generated in terms of resource allocation and operation scheduling.

DT IV, VI: Making intelligent decisions - the decision support made on these centralized, high level DT layers allows embedding the digital twin for smart production control in two roles: i) smooth batch rescheduling based on the forecasted cost of resource usage for global optimization; ii) resource health management and predictive maintenance.

IV. Machine Learning for Resource Health Managing

ML methods are used by the DT III MES layer in the cloud. Detecting resource anomalies needs classification at multiple levels, for example the multi-perspective health of resources, obtained by combining a number of KPIs in the map-reduce phase. When all KPIs exist in a merged stream, the Principal Component Analysis (PCA) reduces the size of the problem from NK (the dimension of the KPI space) to 2, after which an algorithm detecting major deviations (anomalies) is applied to classify the resource (e.g. robot) state. A binary sorting of the state is considered, where the robot system can be either in a healthy or unhealthy state. In this context, the goal is to identify outliner states; if the robot state is far-off the centre of good known states after PCA size reduction it is counted as anomaly.

The building block of the predictive functionality in DT III, V is the long short-term memory (LSTM) model of recurrent neural network (RRN) in univariate mode, used to learn the pattern of a single parameter and predict in time the value one or more steps ahead [20]. Resource-based predictors are bound to each shop floor device and distinct for each operation it is performing. They are implemented with sklearn LSTM and run centralized in HPCC clusters of virtual machines; they hold an own time-based representation used to establish the current outcome and to predict future values of learned patterns [21].

Fig. 4 shows the LSTM time sequence of energy prediction with in-between step estimate. The layout has t chained cells, each one responsible for one step in the time series pattern. The inputs are instant energy measurements at each step (X: input feature vector), and the intermediate outputs (Y: output feature vector) represent the predicted energy consumption at the next step in the sequence.

Resource-based predictors are identified by the resource they belong to and the operation type. For example, for n shop floor resources R_i, $1 \leq i \leq n$, each of them capable of executing O_i distinct operations there would be (O1+...+On) independent LSTM neural networks allocated. Each neural network learns a time-based pattern of cost (e.g., energy consumption) for a type of operation in unsupervised mode by processing the relevant stream resulting from the map-reduce operation.

The design of DT V, DT VI layers for large-scale CMfg pools of resources and big production orders considers a combined input model for the optimization of batch schedules that uses resource cost usage KPIs obtained both from real time measurements (data received from DT II) and from predictions on a number of steps ahead (data computed in DT III) by help of ML techniques. The batch scheduling algorithm is therefore implemented in two stages. In the first stage the scheduling and allocation are computed based on historical data and initial estimations; in the second stage one LSTM is trained on each resource / operation pair during live execution of scheduled operations. The LSTM models are then used as input for real-time rescheduling, whenever the actual optimal instance differs significantly from the new, predicted one.

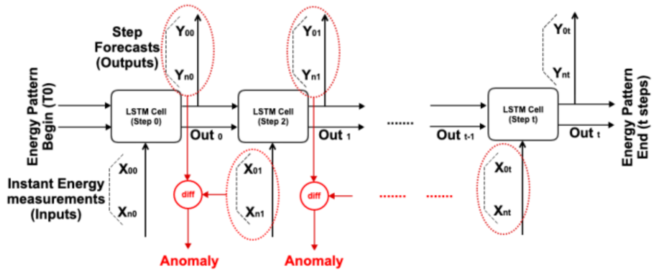

Fig. 4. Prediction of resource energy consumption in LSTM time sequence

Multi-dimension feature vectors can be set up to customize the resource's usage cost for operations or tasks on products; for example, joining robot motion speed, energy consumption and temperature of the 1st and 2nd axes' drives differently weighted would allow configuring multi-objective global cost KPIs for products, groups of products or batches: primary target - execution cycle, secondary indicators – energy saving and resource protection. Depending on the dimension of the feature vector considered other ML algorithms can be used, such as isolation forest and local outliner factor for high and medium dimensions respectively. Another important decision factor for the classification algorithm selection is how clean the training data set is.

V. Experimental Results

Experiments were performed in the 4-robot manufacturing cell of University Politehnica in Bucharest with job-shop working scenarios. The four revolute Adept Viper s650 robots perform bin picking and feed/extract braking disks for multiple machine tools in pick-and-place operations. The robot gripper holds a magnetic tool capable of collision detection; it uses a spring which is compressed when a disk is hit (*pick*). Before tending the CNC machines (*place*) the disks are demagnetized. To restrict to 2 the number of disks grasped in a pick sequence, the gripper weights the payload based on a sensor measuring the spring's extension. During rapid robot movements, the spring vibrations are monitored and limited for proper operating of the gripper mechanism. Such extended robot abilities need greater real time processing power and parallel executions of robot controls. The pick-and-place duration is monitored as cost KPI.

The classifications of robot pick-and-place operations were computed using One Class Support Vector Machine (SVM) [22] with radial basis function non-linear kernel. Models were trained on normal data; the trained models were used then to detect significant increase in pick-and-place time (abnormal robot operation) based on current time measurements and real-time LSTM predictions. The anomaly detection algorithm in DT IV decides whether a new operation belongs to the same distribution of normal operations or is an outliner. The decision border was set up from initial samples on normal operations.

Fig. 5 represents a set of 100 pick-and-place operations, timed at two robot motion speed settings (75% and 100%). The visualisation uses the Principal Component Analysis technique that reduces the feature space to 2D. There were included in the data set four abnormal operations produced by combinations of: rubber damper wear and malfunction of inertial sensors for tool vibration amplitude/analogue sensors for spring extension. The training data for the model is symbolised by green dots, while the yellow ones represent live executions of operations.

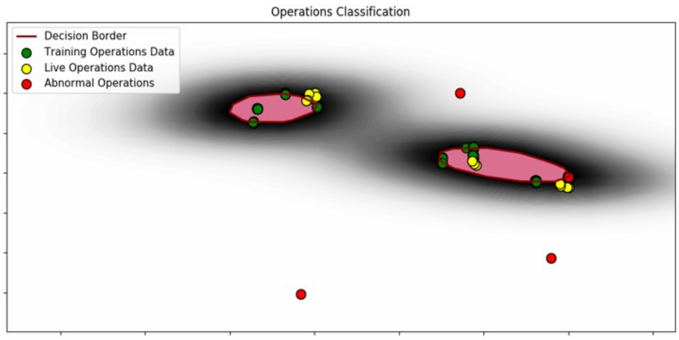

Fig. 5. Visualization of robot operations classification with PCA and SVM

The SVM has identified two distinct clusters representing the two cruise speeds for the robot operations completed and added to the data set. Abnormal executions are pointed in red. A useful feature of this classification approach is that a score can be computed for the location of the operation relative to the decision border (represented by the red line); a threshold can be hence applied for soft changes in operation time and used as decision point for on-line new runs of the optimization engine.

VI. Conclusion

The paper proposes a smart manufacturing control model that ensures cost optimization at global horizon of production batches and reality awareness of resource usage; these objectives are reached with a hybrid holonic control topology that assigns a shared pool of resources to product operations re-scheduled optimally in real time in the cloud. The smart control embeds a 4-layer digital twin replicated for the health monitoring of all shop floor resources, and a 2-layer high level centralized DT based on IA, deciding upon resource re-allocation from measurements of cost/operation KPIs and real-time prediction of QoS evolution.

Acknowledgment *(Heading 5)*

The work has been funded by the Operational Program Human Capital of the Ministry of European Funds through the Financial Agreement 51675/09.07.2019, SMIS code 125125.

References

[1] Trentesaux, D., Borangiu, T., Thomas, A. (2016). Emerging ICT concepts for smart, safe & sustainable industrial syst., COMIND, 81, pp. 1-10

[2] IEC (2018). Factory of the future, White paper, ISBN 978-2-8322-2811-1, www.iec.ch/whitepaper/pdf/iecWP-futurefactory-LR-en.pdf

[3] Soldatos, J., Cavadini, F.A., Lazaro, O. (2020), The Digital Shopfloor: Industrial Automation in the Industry 4.0 Era, River Publishers Series in Automation, Control and Robotics, doi: 10.13052/rp-9788770220408

[4] EFFRA (2020). Project FAR-EDGE: Factory Automation Edge Computing Operating System Reference Implementation, EFFRA - EU Factories of the Future Research Association, https://portal.effra.eu/project/1628

[5] Lee, J., Azamfar, M., Siahpour, S. (2020). Integration of digital twin and deep learning in Cyber-Physical Systems: towards smart manufacturing, IET Collaborative Intelligent Manufacturing, doi:10.1049/iet-cim.2020.0009

[6] Derigent, W., Cardin, O., Trentesaux, D. (2020). Industry 4.0: contributions of holonic manufacturing control architectures and future challenges, J. Intell. Manufacturing, doi: 10.1007/s10845-020-01532-x

[7] Arasu, A. et al. (2016). STREAM: The Stanford data stream management system, Data Stream Management, pp. 317-336, Springer

[8] Wang, J., Ma, Y., Zhang, L., Gao, R.X., Wu, D. (2018). Deep learning for smart manufacturing: Methods and applications, J. of Manufacturing Systems, Vol. 48, part C, pp. 144-156

[9] Ren, S., Zhang, Y., Liu, Y., Almeida, C. (2019). A comprehensive review of big data analytics throughout product lifecycle to support sustainable smart manufacturing, J. Cleaner Production, vol. 210, pp. 1343-65

[10] Alwis, R., Perera, S., Penchicala, S. (2017). Machine Learning Techniques for Predictive Maintenance InfoQ, https://www.infoq.com/articles/machine-learning-techniques-predictive-maintenance/

[11] Coullon, H., Noyé, J. (2018). Reconsidering the relationship between Cloud Computing and Cloud Manufacturing, Proc.of SOHOMA 2017, Studies in Computational Intelligence, Vol. 762, Ch. 6, pp. 217-228, Springer

[12] Gamboa Quintanilla, F., Cardin, O., L'Anton, A., Castagna, P. (2016). A modeling framework for manufacturing services in Service-oriented Holonic Manufacturing Systems, Engineering Applications of Artificial Intelligence, 55, pp. 26-36

[13] Van Brussel, H., Wyns J., Valckenaers, P., Bongaerts, L., Peeters, P. (1998). Reference Architecture for Holonic Manufacturing Systems: PROSA, Computers In Industry, vol. 37(3), pp. 255-276

[14] Valckenaers, P. (2019). ARTI Reference Architecture-PROSA revisited, SOHOMA'18, Studies in Computational Intelligence, vol. 803, pp. 1-19, Springer

[15] Răileanu S., Anton F., Iatan A., Borangiu T., Anton S. (2017). Resource scheduling based on energy consumption for sustainable manufacturing, Journal of Intelligent Manufacturing, Vol. 28, Issue 7, pp. 1519-1530

[16] Novas, J., Bahtiar, R., Van Belle, J, Valckenaers, P. (2012). An approach for the integration of a scheduling system and a multi-agent manufacturing execution system. Towards a collaborative framework, Proceedings 14th IFAC Symposium INCOM'12, Bucharest, pp. 728-33, IFAC PapersOnLine

[17] IBM Analytics (2020). CPLEX Optimization Studio 12.10 performance improvements, https://www.ibm.com/downloads/cas/KEVDB4NZ

[18] Andre, P., Azzi, F., Cardin, O. (2020). Heterogeneous communication middleware for Digital Twin-based cyber manufacturing systems, SOHOMA 2019, Stud. in Comput. Intell., vol. 853, pp. 146-57, Springer

[19] Redelinghuys, A.J.H., Kruger, K., Basson, A.H. (2019). A Six-Layer Architecture for Digital Twins with Aggregation, Proceedings SOHOMA'18, Studies in Computational Intelligence, Vol. 803, Chapter 32, pp. 412-423, Springer

[20] Olah, C. (2015). Understanding LSTM Networks, Towards Data Science, http://colah.github.io/posts/2015-08-Understanding-LSTMs/

[21] Williams, J.D. (2016). End-to-end LSTM-based dialog control optimized with supervised an reinforced learning, https://arxiv.org/pdf/1606.01269v1.pdf

[22] Microsoft Azure, One-Class Support Vector Machine, June 2019, https://docs.microsoft.com/en-us/azure/machine-learning/studio-module-reference/one-class-support-vector-machine