ARTICLE IN PRESS

Journal of Manufacturing Systems xxx (xxxx) xxx–xxx

Contents lists available at ScienceDirect

Journal of Manufacturing Systems

journal homepage: www.elsevier.com/locate/jmansys

Technical Paper

Dynamic production system identification for smart manufacturing systems

Peter Denno[a,*], Charles Dickerson[b], Jennifer Anne Harding[b]

[a] National Institute of Standards and Technology, Gaithersburg, MD, USA
[b] Loughborough University, Loughborough, UK

ARTICLE INFO

Keywords:
System identification
Production systems
Genetic programming

ABSTRACT

This paper presents a methodology, called production system identification, to produce a model of a manufacturing system from logs of the system's operation. The model produced is intended to aid in making production scheduling decisions. Production system identification is similar to machine-learning methods of process mining in that they both use logs of operations. However, process mining falls short of addressing important requirements; process mining does not (1) account for infrequent exceptional events that may provide insight into system capabilities and reliability, (2) offer means to validate the model relative to an understanding of causes, and (3) updated the model as the situation on the production floor changes. The paper describes a genetic programming (GP) methodology that uses Petri nets, probabilistic neural nets, and a causal model of production system dynamics to address these shortcomings. A coloured Petri net formalism appropriate to GP is developed and used to interpret the log. Interpreted logs provide a relation between Petri net states and exceptional system states that can be learned by means of novel formulation of probabilistic neural nets (PNNs). A generalized stochastic Petri net and the PNNs are used to validate the GP-generated solutions. The methodology is evaluated with an example based on an automotive assembly system.

1. Introduction

Knowledge of process requirements, system capacities, and system reliability are the premises on which control policies are formulated. In dynamic manufacturing environments, engineering change to the product, the process, and the production equipment can cause these premises to be violated and thereby make control policies less effective. An accurate, up-to-date model of the production system is essential to production control, but a challenge to maintain.

Both the need for a production system model and the challenge of maintaining it are more intense in smart manufacturing settings. The need is more intense because a key goal of smart manufacturing is to automated decision making [1]. Decisions concerning sequencing [2], line balancing [3,4], and production system engineering [5] are sensitive to changes in process requirements, system structure, capacities, and reliability expressed in production system models. The challenge is more intense because smart manufacturing can make manufacturing more agile [1], and the changes brought on by increased agility must be reflected in the production system model. Change in process requirements is commonplace in manufacturing environments where products are evolving rapidly. Changes in system structure, capacities, and reliability are less common; but control policies are affected as much by changes in these dimensions as they are by changes in product and process.

Dynamic production system identification is a methodology that develops and updates a production system model that can provide information essential to performance analysis and control. The methodology (1) identifies a model that, like traditional statistical system identification [6] responds to stimulus accurately, (2) identifies system components, their properties, and interconnection, (3) identifies normative process for multiple job types, and (4) continually updates the model.

The production system model is a process model. Machine-learning methods of process mining typically develop such models using an analysis of frequently occurring events described in system logs. These methods fall short of addressing the challenge of dynamic production system identification in three important respects: (1) Rather than frequently occurring events, it is the infrequent, exceptional events that typically provide insight into system capacities and reliability. (2) Production system behaviour, especially machine blocking and starvation, are well-understood phenomena; an analysis of cause and effects could be used to guide search to an accurate system model. (3) Process mining lacks inherent means to update the model as the modelled system changes.

The production system model describes processes associated with International Society of Automation (ISA) Level 3 control problems [7].

* Corresponding author.
E-mail address: peter.denno@nist.gov (P. Denno).

https://doi.org/10.1016/j.jmsy.2018.04.006
Received 30 November 2017; Received in revised form 31 March 2018; Accepted 11 April 2018

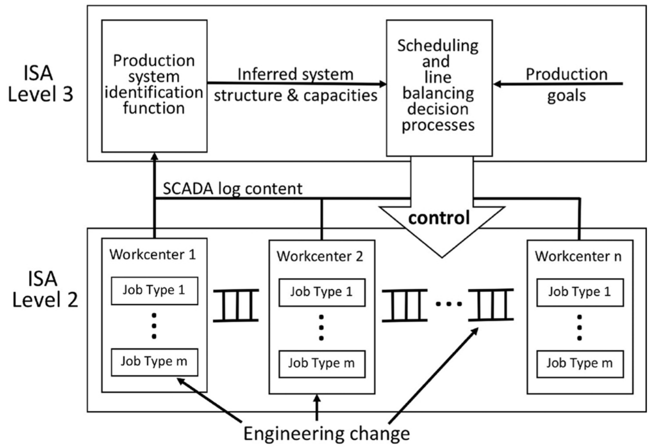

Fig. 1. Production system identification in context.

Our methodology infers the production-system structure and capacities specifically for use in line scheduling and balancing processes (see Fig. 1). In the methodology, genetic programming, default causal knowledge, and probabilistic classification of exceptional conditions are used to evolve a population of individuals each representing a candidate model. The fitness of an individual is assessed with respect to its ability to (1) reproduce the content of logs describing typical Supervisory Control and Data Acquisition (SCADA) events, (2) in comparative steady-state analyses, respond to perturbations in workstation capacity with plausible differences in buffer occupancy and state sojourn times, and (3) detect critical job-type distinctions (e.g. that one job type requires significantly more processing time at some workstation than does another job type).

The main contribution of this paper is a robust methodology for dynamic production system identification. The paper investigates the value of genetic programming (GP) of Petri nets (PNs) in meeting its goals. GP on PNs is intended to facilitate adaptation of the methodology to diverse production system architectures and logging scenarios. The paper provides novel methods to interpret logs, validate the model, and learn from exceptional events.

Section 2 of the paper describes related work. Section 3 presents a Petri net model, the Augmented Queueing Petri Net (AQPN) which provides the model of process used in GP evolution. Section 4 describes how exceptional conditions, causal validation, and model updating are handled. Section 5 describes a case study that uses the methodology. Section 6 concludes the paper with an assessment of the methodology's limitations and a discussion of future work.

2. Related work

Process mining [8,9], and advanced system identification methods [10,11] provide semi-automated means to produce process and system models for various purposes including process conformance (i.e., determining whether or not the actual process being practiced conforms to the normative process). Typically, these methods have the goal of capturing the most frequent process patterns and exhibiting robustness to noise [12].

van der Aalst et al. [13] describe a process mining algorithm known as the α-miner. The algorithm produces structured workflow nets (SWF-nets) from process logs. SWF-nets are untimed safe Petri nets constrained to avoid two forms of so-called “confusion” in the composed use of choice and synchronization in Petri nets.

Alves de Medieros [12] describes a genetic algorithm approach using SWF-nets to address some of the limitations of the α-miner. Specifically, it solves the choice/synchronisation confusion problem and addresses invisible and duplicate tasks. It is robust to noise by ignoring infrequent events.

Rozinat et al. [8] describe a methodology for constructing simulation models that involves four perspectives on process: control-flow, data, performance, and resource. The work uses coloured Petri nets. The simulation models produced do not make a distinction between normative and exceptional events.

Some relevant work associates more closely with system identification than process mining. Several of these, including [11,10,14] use integer linear programming (ILP). Ould El Mehdi et al. [11] uses ILP to produce deterministic and stochastic Petri net (DSPN) models of systems. The work is targeted to reliability analysis of repairable systems. DSPNs are of limited use in modelling production systems because an analytical solution of steady-state can only be had with DSPNs if no more than one deterministic transition is enabled in any marking [15].

Basile et al. [10] describes a mixed integer linear programming method of system identification that produces timed PNs. The underlying algorithm assumes a bijective relationship between event-log entries and PN transitions. The work does not use a coloured Petri net (CPN) model. Colours in CPNs can be used to represent differing job types, which is necessary in models of production lines.

Turner et al. [16] is the only work the authors are aware of that uses genetic programming for process mining. This short paper asserts that genetic programming provides greater flexibility in problem formulation and the possibility of mining complex and problematic event logs. The systems described do not use buffers nor does the methodology address exceptional conditions.

Compared to the work cited, our methodology emphasizes a means to establish a relationship between the information generated in production and the system's components. The identified model is not designed for use as a simulation directly but as a means to infer, organize, and update information needed when building simulations and decision support tools that need to be responsive to change.

3. Dynamic production system identification

The goal of any process modelling effort is to produce models fit for purpose [17]. Knowledge of system capacities is essential to the purpose of modelling production scheduling. For complex system engineering generally, and production system engineering particularly, capturing the most frequent process patterns will not be sufficient to create such a model. There are three interrelated reasons for this. First, the behaviour of complex systems under unforeseen circumstances cannot be predicted from the study of its response to seen circumstances. Hence models based only on frequent events (seen circumstances) are not in themselves very good simulations of the actual system. Second, a system response (e.g. blocking) can be a consequence of earlier interactions between the system and it environment. That environment might reflect exceptional circumstances. For example, while a machine is inoperative, work builds up at its input buffer. A model useful to scheduling must be capable of carrying this information forward to reflect a new state. The new state reflects exceptional circumstances and a capacity. Conversely, a model fit to data from only frequent and normative events would have no basis for doing this. Third, many analytical methods in production control require a specification that separates system description (e.g. capabilities, capacities, and system topology) from problem specification (e.g. demand, product mix). Unfortunately, state-of-the-art process-mining methods do not address these issues.

A sketch of the methodology is provided in Fig. 2. To test the methodology, a discrete event simulation system for mixed-model