Chapter 11
A Symbolic Approach to Self-optimisation in Production System Analysis and Control

Christopher M. Schlick, Marco Faber, Sinem Kuz and Jennifer Bützler

11.1 Introduction

With steadily increasing customer requirements on quality of both products and processes, companies are faced with increasing organisational and technical challenges. The market is characterised by individualised customer wishes which result in individual adaptations of the products. In order to manage this rapidly growing variety of products, the production system has to become much more flexible with respect to the product structure to be manufactured and the corresponding production and assembly processes. Especially in the field of assembly systems the increasing variety of products adds new complexities to the planning process and increases the costs, because (re-)planning efforts tend to grow exponentially to the number of variants.

One approach to overcome these limitations is to design production systems that are able to autonomously adjust to market needs. If the automatic control systems of machines, robots and technical processes could flexibly adjust themselves to the environmental conditions and autonomously find solutions through a goal-oriented forward and backward chaining of production rules, the efforts of developing the control programmes would be reduced significantly. This would cut down the non-value-adding activities, thereby yielding a higher productivity for the company. Following the seminal work of Adelt et al. (2009) we speak of self-optimisation.

Besides flexibility, companies also have to integrate the working person into the production process. The human operator will always be involved either by directly taking over assembly tasks (e.g. for limp components) or by supervising the assembly process. Furthermore, unique human skills such as sensorimotor coordination and creative problem solving cannot be automated. To establish a safe,

C.M. Schlick (✉) · M. Faber · S. Kuz · J. Bützler
Institute of Industrial Engineering and Ergonomics (IAW), RWTH Aachen University,
Bergdriesch 27, 52052 Aachen, Germany
e-mail: c.schlick@iaw.rwth-aachen.de

C. Brecher (ed.), *Advances in Production Technology*,
Lecture Notes in Production Engineering, DOI 10.1007/978-3-319-12304-2_11

effective and efficient integration of the working person into the production process, ergonomic aspects have to be considered. New technologies such as lightweight robots or electro-optical sensors open up new possibilities in the area of ergonomic human-robot cooperation. For the first time, it is now possible to abolish the strict separation between the work areas of the human and the robot (e.g. Bascetta et al. 2011; Fryman and Matthias 2012; Matthias et al. 2011). Light detection and ranging sensors in particular enable the robot to recognise the human early enough to adjust or even stop its movement. The action forces of lightweight robots are also considerably lower than those of conventional industry robots, minimising the risk of injury and ensuring the safety of the cooperating working person.

In this regard a cognitive control unit has been developed that can cognitively control a robotic assembly cell. It is embedded into a general architecture for self-optimising production systems.

11.2 Cognitive Automation

In order to cope with the cited challenges for assembly systems a novel approach to cognitive automation was developed (Mayer 2012; Faber et al. 2013). To support the human operator effectively and efficiently, he/she has to be able to understand the system's functions and behaviour. A simplified compatible representation of the mental model of the operator on assembly processes in a dynamic production environment based on production rules has therefore been developed and integrated into the knowledge base of the cognitively automated system. By explicitly considering ergonomic criteria (e.g. feasibility, occupational risks, freedom of impairment, promotion of personality development (Luczak and Volpert 1987) the system is also capable of improving the working conditions for the human operator interacting, for instance, with the robot or supervising its functions. Figure 11.1 depicts the architecture of the cognitively automated system. The central element is the Cognitive Control Unit (CCU) which is based on the three layer architecture for robotic applications consisting of a planning, a coordination and a reactive layer according to Russel and Norvig (2003). The architecture has been extended with a presentation layer for ergonomic human-machine interaction and a technical layer that includes the sensors, automatic control algorithms and actuators (Hauk et al. 2008).

11.2.1 Cognitive Automation of Assembly Tasks

The cognitive automation functions for self-optimising assembly processes are realised in the planning layer, the central element of the CCU. This layer is responsible for planning and optimising the assembly sequence and for deriving high-level action commands according to the generated assembly steps. In contrast, the reactive layer is responsible for the direct communication with the actuators and

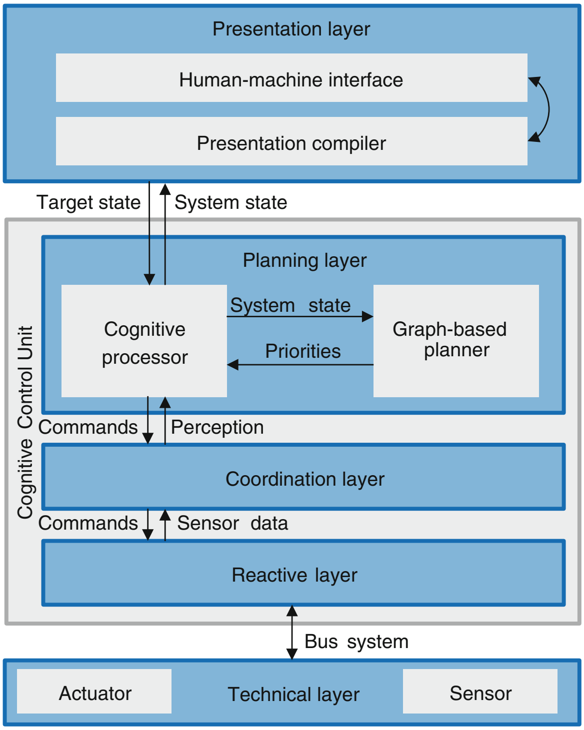

Fig. 11.1 Architecture of the Cognitive Control Unit (adapted from Mayer (2012))

sensors. The coordination layer in between translates between the planning and reactive layer. A detailed description of all three layers can be found, for example, in Hauk et al. (2008), Mayer et al. (2012) and Faber et al. (2013). The following section will focus on the planning layer. To evaluate the functions of the CCU a cognitively automated assembly cell has been developed (Brecher et al. 2012). A six axes articulated robot (KUKA KR30 Jet) is used with a three finger gripper with haptic sensors (SCHUNK SDH2) to handle parts and components. The work area of the assembly cell is divided into three sub-areas: Parts and components are fed into the system through a circular conveyor belt. In addition, two sub-areas are used to assemble the final product and to buffer parts and components that cannot be assembled directly. The parts and components on the conveyor belt do not have to be in a predefined sequence. The sequence can be completely random and may also include parts that are not needed for the current product being assembled.

The final product is specified by the human operator through the human-machine interface in the presentation layer and represents the goal state of the cognitive controller. The goal state contains only the geometric information about the final product including the type, position and orientation of the individual components in terms of CAD data. This data is forwarded in combination with the planning knowledge to the cognitive controller. Based on the goal state and the current system state that is propagated by the coordination layer, the cognitive processor is able to derive the assembly sequence autonomously. The optimal next assembly step is transferred as a high-level command to the coordination layer where it is translated to machine commands for the articulated robot used.

The decision-making process of the cognitive processor is based on the cognitive architecture Soar (Laird 2012), a symbolic computational system that is able to simulate the human cognition. The knowledge that is necessary for planning the assembly steps is solely specified in terms of if-then production rules (Faber et al. 2014). The CCU is able to adjust flexibly to changes in the part sequence, because there is no need to (re-)estimate parameters as there is with other methods such as dynamic Bayesian networks. The planning knowledge includes procedural knowledge of experienced operators and is therefore represented in a way that makes the assembly process more transparent and conforms to the expectations of the human operator supervising the system (Mayer and Schlick 2012; Faber et al. 2014). In addition, it is designed as generically as possible so that it can handle changes in the product structure as well.

To keep the complexity of the production rules and the planning process within the cognitive processor low, the processor has a very limited planning depth. In fact, it is only able to plan one assembly step in advance. However, this is not enough to deal with complex planning criteria that need to take information about the whole assembly sequence into account in order to ensure that safety-critical situations do not occur in the sequence. This is essential for ergonomic working conditions, because the safety of the human operator has to be ensured at all times during the production process. To satisfy this requirement the cognitive processor was extended by a graph-based planner. This planner is described in detail in the next section. In this way, the ergonomic risk can be minimised and, if some risk is unavoidable, reduced to an acceptable level. In this case a warning message could be given to the human operator at specific points in time to alert him/her to the types and sources of risks.

11.2.2 Adaptive Planning for Human-Robot Interaction

As mentioned in the previous section, the originally developed cognitive processor is purely reactive and is not able to consider complex optimisation criteria in the planning process. Extending its planning process to a higher planning depth (or even to a full planning process considering the complete assembly sequence) would inevitably result in a much more complex planning procedure. An exponentially growing number of achievable goal states have to be simulated and compared against each other in order to find the optimal alternative. To make the cognitive processor more efficient it has been extended by a graph-based planner (Faber et al. 2014) whose operation mode follows a hybrid planning approach including an offline and an online phase (Ewert et al. 2012). It interacts with the cognitive processor and provides additional information about the future assembly sequence for the decision phase of the cognitive processor.

In preparation of the assembly process, the structure of the product is transferred into a directed state graph including all valid assembly sequences. In particular, each state represents an achievable intermediate goal state in the assembly process. The intermediate states are identified by recursively decomposing the final product according to the “assembly by disassembly” strategy (see e.g. Thomas and Wahl 2001). Consequently, each edge of the resulting graph can be considered as a feasible assembly step which modifies the intermediate product state by adding exactly one part or component. The generation of the assembly graph can be done offline because, despite changes in the product structure, the same dependencies can be used in every cycle without losing the flexibility of the cognitive processor to react to changes in the assembly environment. Figure 11.2 shows an exemplary assembly graph of a simple product consisting of five cubic parts.

In order to be able to compare the alternatives for the next assembly step, each edge is weighted with a set of costs indicating how “costly” it is to perform the corresponding assembly step. In its simplest form, each feasible assembly step induces costs c_b representing the basic effort of the assembly action. In addition, rule-based planning knowledge can be formulated and applied to consider additional optimisation criteria. These rules can be activated individually and refer to the state transitions of the assembly graph. If the condition of a rule is satisfied, its costs are added to c_b yielding a set of costs per edge depending on the activated planning criteria. Figure 11.2 demonstrates a simple scenario where a two finger gripper is

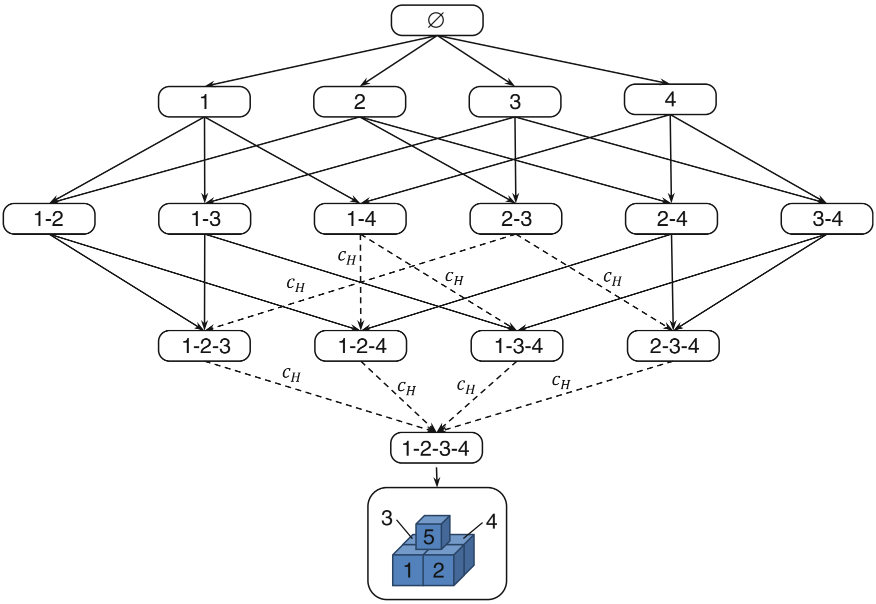

Fig. 11.2 Exemplary state graph of a simple product consisting of cubic parts. The *dotted edges* indicate assembly steps that have to be carried out by the human operator due to the technical restrictions of a two finger gripper (Faber et al. 2014)

used by an articulated robot to assemble the product. As the gripper requires two freely accessible parallel sides, the gripper cannot handle all components. At some points in the assembly process the human operator has to take over assembly tasks to assemble the final product (indicated with costs c_H). However, as one of the main objectives of the CCU is to assemble the product as autonomously as possible and to let the human operator take over an assembly task only if necessary, the number and sequence of manual interventions should be optimised. These manual interventions should be chosen in a way that the operator can effectively and safely use and develop his/her skills in the assembly process and has a complete work process. As can be seen in the graph, the interventions by the operator cannot be avoided completely, but can be optimised by selecting an assembly sequence in advance that leads to low physiological costs due to few and grouped interventions. Therefore, the right decision already has to be made on the second level of the presented graph, at which the cognitive processor itself does not have enough information in order to reliably choose the optimal path.

To be able to provide sufficient information to the cognitive processor, the graph-based planner has to evaluate the costs of the remaining assembly sequence in each assembly cycle starting at the current system state. Therefore a modified version of the algorithm A*Prune (Liu and Ramakrishnan 2001) is applied to the graph. The modifications refer to the modality of comparing two alternative assembly sequences in order to adjust the algorithm to the given application scenario (Faber et al. 2014). Once a set of k potential assembly steps fitting best to the current system state is found, this set is transferred to the cognitive processor. The processor is then able to make its decision based on its own information as well as external information. If conflicts arise between goals due to the wider planning horizon, the information of the graph-based planner is always weighted higher than that of the cognitive processor.

11.3 Embedding the Cognitive Control Unit into an Architecture for Self-optimising Production Systems

A promising approach to design more flexible production systems is to take architectures of self-optimising systems into account (Adelt et al. 2009). These kinds of systems are sensitive to environmental changes and can therefore make goal-oriented decisions or adjust their internal goal system. Figure 11.3 depicts a self-developed architecture of a cognitively automated self-optimising production system. The model is based on the cascading quality control circuits after Schmitt et al. (2012) and differentiates the levels segment, cell, machine and process. Each layer follows its own decision cycle according to its own cognitive controller. Every subordinated layer can be considered as a cognitively controlled system of the next higher level. The resulting cascade control leads to a self-similar structure of the

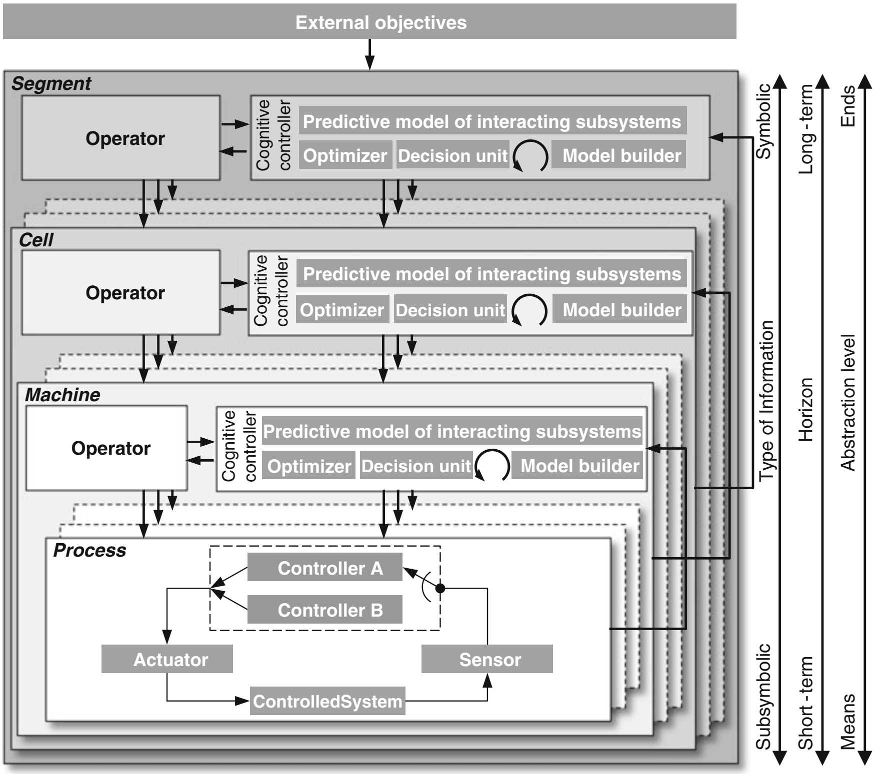

Fig. 11.3 Architecture for cognitively automated self-optimising production systems (adapted from Mayer (2012))

overall architecture that is comparable to hierarchically controlled software systems (e.g. Litoiu et al. 2005).

The bottom level of the architecture represents the sub-symbolic information processing of the automatic control systems. In the next higher levels, the adaptation process is based on symbolic "cognitive controllers". Their decision-making process is based on the current system state in conjunction with the pursued goal. In particular, they generate and update a model of the controlled process in conjunction with the environment within the model builder. This model contains the execution conditions of the production process as well as the information of the interacting subsystems in the appropriate granularity. Based on the generated model, the optimiser and decision unit are able to make context-sensitive decisions. At the machine level, for instance, functionalities of a model-based self-optimisation (Schmidt et al. 2012) are realised whereas the cell level aggregates several machines to higher level production units following coordinated actions. Finally, the segment can be considered as a macro structure combining several cells for the overall production process. The level of abstraction correspondingly increases from process level to segment level. The type of information that is processed also changes. The automatic control is based on continuous spatiotemporal signals whereas the controllers at machine, cell and segment levels use a symbolic representation of the state information.

At each of the higher levels, a human operator interacts with the cognitive controller (Mayer 2012). This can be a physical interaction, such as at machine level, but are more usually supervisory control tasks processed in order to monitor the system behaviour. The system therefore requires ergonomic human-machine interfaces to display information, enable the operator to recognise the current state of the system, to understand its functional state and behaviour and to be able to intervene if necessary.

The optimisation criteria of the production system are determined by both external and internal objectives. External objectives, such as constraints regarding the lead time or costs, are processed at each level and propagated to the next lower system. Each subsystem on the individual levels generates additionally its own internal objectives. At the machine level, this could be constraints regarding wear and tear or energy consumption whereas at higher levels the objectives could relate to, for instance, throughput and utilisation. On account of the self-optimising functions, the systems are able to adjust their internal objectives to adapt to environmental changes in the production process (Schmitt et al. 2012). As long as the internal objectives do not contradict the external objectives or objectives generated by higher order systems, they can be adjusted and altered by the corresponding cognitive controllers. In this way, systems can generate additional constraints for their subordinated systems.

The cognitively automated robotic cell, and in particular the CCU presented above, can be embedded into this architecture. Obviously, the machine elements such as the robot and the conveyor belt are located at the machine level. With the self-developed cognitive controller, the robot is capable of managing the pick and place process of individual parts and components in line with its own internal objectives. As the CCU focuses on automating the whole assembly cell, it is located at the cell level. The interacting subsystems of the cognitive controller are accordingly the assembly robot, the conveyor belt and the work areas. The main external objective is the assembly of the final product with respect to the given constraints (e.g. the part supply). To achieve this goal, a predictive model is used that contains the description of the final product in terms of CAD data and the knowledge about assembling the product. The knowledge comprises the production rules of the cognitive processor and the planning rules of the graph-based planner and forms the basis for the joint decision-making process.

The predictive model of the interacting subsystems is built by the model builder. The knowledge base required for the assembly process is formulated manually by production experts. This task has to be done with care as the knowledge affects not only the assembly process itself but also safety aspects of the human-robot interaction. Introducing erroneous production rules can lead to wrong decisions and non-acceptable risks for the human operator. In addition to the knowledge base, the internal representation of the final product is generated in the model builder for planning purposes. This representation also includes the automatically extracted neighbourhood relationships of the individual parts and components. Within the model builder it is also possible to generate new intermediate goals in order to divide the current task into smaller subtasks. Such subtasks could include managing the buffer area or removing erroneous components that have been misplaced (e.g. due to erroneous sensor readings).

Finally, the fusion of the data takes place in the optimiser and decision unit. In the optimiser the machine states are evaluated, including the available components. Based on the environmental model of the cognitive controller, the preferences in the material flow are set and alternatives in the action sequence are compared by the graph-based planner. The main goal of the optimiser is to reduce the solution space for the decision cycle of the cognitive software architecture Soar by providing action-oriented planning information. The decision for one of the possible actions is made in consideration of the preferences that have been set and the internal and external objectives of the subsystems involved.

11.4 System Validation

The function of the planning layer of the presented architecture has been validated by means of a simulation study. This study validated both the correctness of the generated assembly sequences and the support of human-robot interaction. Based on the developed architecture, the following hypotheses were formulated:

- The assembly process should be as autonomous as possible, so that the number of manual interventions within the human-robot cooperation is reduced to a minimum. Additionally, the type of manual tasks should let the human operator focus on his/her unique sensorimotor skills.
- To achieve a complete work process for the human operator, the manual work steps should be placed within the shortest possible time interval. The working person then has more flexibility in designing and organising his/her remaining work (supervisory control, quality control, etc.).
- If the product consists of several assembly groups, there should be as few changes between those groups during the assembly process as possible. This maximises the transparency of the assembly process and makes it easier to intervene if errors occur.

In a first simulation study, simple products consisting of single-type cubic parts were assembled (Faber et al. 2014). Both size and structure of the product were varied to yield assembly graphs of different complexity with respect to the average node degree. Products of type 1 consist of a single layer of parts whereas in products of type 5 all parts are mounted one above the other (“tower”). The structures in between describe intermediate complexities of the assembly graph. The part supply through the conveyor belt was completely randomized and included components that were not needed for the current product. The number of parts

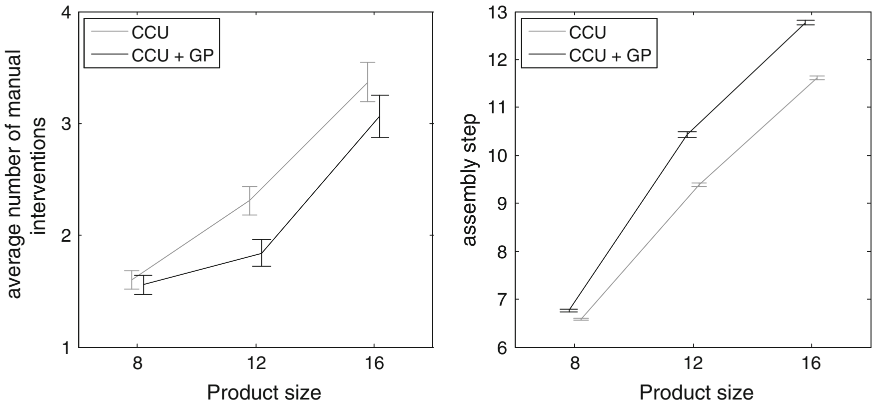

Fig. 11.4 Average number of manual interventions with activated and deactivated graph-based planner (*left*) and positions of the manual intervention in the assembly sequence of product type 2 (*right*)

that are concurrently fed into the system was also varied systematically. For each combination of the aforementioned independent variables, the assembly was simulated by the CCU with the graph-based planner either activated or deactivated. The cognitive planning of the assembly process had to be done under the following constraints: (1) New parts were only allowed to be assembled in the direct neighbourhood to existing parts in order to increase the transparency of the system behaviour (Mayer 2012). (2) The two finger gripper used needs two freely accessible parallel sides. Otherwise, this part has to be assembled manually by the human operator. Dependent variables for all simulation runs were the generated assembly sequence and the resulting number of manual interventions by the human operator.

Figure 11.4 shows the average number of assembly steps that have to be carried out by the human operator on the left side. Products of type 5 ("tower") are not considered here as they do not require human intervention. As shown in Fig. 11.4 (left) the manual interventions can be reduced for all product sizes. For products consisting of 12 parts this reduction is also significant ($p<0.01$) according to a t-test with level of significance $\alpha = 0.05$. The distribution of the manual assembly steps in the assembly sequence could also be improved. On the right-hand side, Fig. 11.4 exemplarily shows the results for products of type 2. In this case, the interventions could be moved to a later point in time for product sizes larger or equal to 12 parts and almost fixed to a single point in time for products consisting of 8 or 16 parts.

To evaluate the third hypothesis and to transfer the approach to a real product, a second simulation study was carried out (Schlick et al. 2014). In this second study, a model of a Stromberg carburetor consisting of three independent assembly groups was assembled. A new planning rule was introduced in the graph-based planner prohibiting a new assembly group from being started while other assembly groups were still not finished. The simulation study covered three scenarios: (1) planning

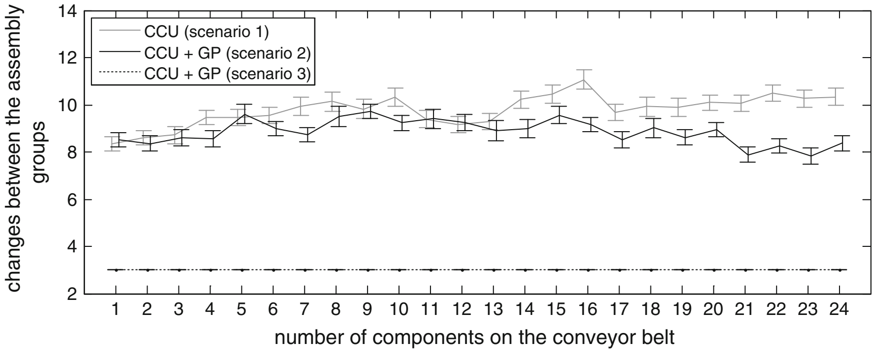

Fig. 11.5 Average number of changes between assembly groups of the Stromberg carburettor depending on the number of components that are fed into the system at the same time (adapted from Schlick et al. (2014))

without the graph-based planner, (2) planning with the graph-based planner, whereby it was allowed to ignore the cited planning rule and (3) planning with the graph-based planner, whereby the rule had to be obeyed. The part supply was again completely randomised. The number of supplied parts was varied systematically between 1 and 24. In all cases, the central part, on which the other parts are assembled, was supplied first.

The simulation results show that the new planning rule has an impact on the number of changes between the assembly groups (Fig. 11.5). Using the CCU with deactivated graph-based planner (scenario 1) yields an average number of changes of 9.78 ($SD = 0.65$). In scenario 2 the graph-based planner is activated and consequently the assembly of an assembly group should preferably be finished before starting a new one (but this is not obligatory). This effect can be reproduced by the simulation. It was possible to significantly reduce the number of changes between the assembly groups according to the Wilcoxon signed-rank test at a level of significance of $\alpha = 0.05\,(mean = 8.86,\ SD = 0.53,\ p < 0.001)$. In contrast to scenario 2, the third scenario requires one assembly group to be finished before starting a new one. In this case, it was always possible to reach the minimum number of three changes. In summary, the simulation study shows that using the graph-based planner significantly reduces the number of changes between the assembly groups and thereby improves the transparency of the system behaviour for the human operator.

However, the scenarios require different efforts for managing the component flow, because supplied components that are not allowed to be assembled directly have to be stored in a buffer. Consequently, more motion cycles (pick and place) are required yielding a higher assembly time for the product. In scenario 2 there was only an average increase of the pick and place operations of 0.84 % compared to scenario 1, whereas in scenario 3 the increase was 63.66 %. The reason behind this significant increase is the fixed rule of prohibiting alternating between assembly groups. So both scenarios have to be traded off against each other with respect to the improvement of working conditions on the one hand and the additional efforts required on the other.

11.5 Summary and Outlook

The increasing changeover to customised production imposes new requirements on companies, which want to remain competitive on the market. They have to redesign their production systems to be flexible enough to produce a huge variety of products in product space under changing conditions of the manufacturing environment. One approach to cope with this kind of complexity is to design self-optimising production systems according to a hierarchical system model. Each level can be considered as a self-optimising system in itself that controls the interacting subsystems. The cognitive controller on each level adjusts its predictive model accordingly and makes goal-oriented decisions on the basis of an optimiser and a decision unit.

The architecture was successfully validated by developing a cognitive control unit (CCU) for a robotic assembly cell. The CCU is able to cope with a large number of product variants, changes in the product structure and variability in the part supply. Its cognitive processor is based on the cognitive software architecture Soar. In order to be able to consider complex planning criteria such as ergonomic aspects, the cognitive processor is enhanced by a graph-based planner. It works on a dynamic state graph that contains all valid assembly sequences and whose edges are weighted according to the planning knowledge. Two simulation studies have shown that the CCU could successfully assemble products under completely randomised part supply and at the same time significantly improve the working conditions for the human operator. In future, the planning knowledge has to be enriched with further ergonomic knowledge in order to further improve human posture, movements and action forces in direct human-robot interaction.

The presented architecture could also be successfully applied to a sub-symbolic level of self-adaptive milling processes based on an adaptive model predictive control algorithm. First approaches concerning the prediction of parameters such as the dead time and the system matrix have produced promising results for future research.

Acknowledgments The authors would like to thank the German Research Foundation DFG for its kind support within the Cluster of Excellence “Integrative Production Technology for High-Wage Countries”.

References

Adelt P, Donoth J, Gausemeier J et al (2009) Selbstoptimierende Systeme des Maschinenbaus – Definitionen, Anwendungen, Konzepte. In: Gausemeier J, Raming FJ, Schäfer W (ed) HNI-Verlagsreihe, Volume 234. Heinz Nixdorf Institut, Paderborn

Bascetta L, Ferretti G, Rocco P, Ardö H, Bruyininckx H, Demeester E & Di Lello E (2011) Towards safe human-robot interaction in robotic cells: an approach based on visual tracking and intention estimation. In: IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS), p 2971–2978

Brecher C, Müller S, Faber M & Herfs W (2012) Design and Implementation of a Comprehensible Cognitive Assembly System. In: Conference Proceedings of the 4th International Conference on Applied Human Factors and Ergonomics (AHFE), USA Publishing, p 1253–1262

Ewert D, Mayer M Ph, Schilberg D & Jeschke S (2012) Adaptive assembly planning for a nondeterministic domain. In: Conference Proceedings of the 4th International Conference on Applied Human Factors and Ergonomics (AHFE), p 2720–2729

Faber M, Kuz S, Mayer M Ph & Schlick C M (2013) Design and Implementation of a Cognitive Simulation Model for Robotic Assembly Cells. In: Engineering Psychology and Cognitive Ergonomics. Understanding Human Cognition. Springer Berlin Heidelberg, p 205–214

Faber M, Petruck H, Kuz S, Bützler J, Mayer M Ph & Schlick, C M (2014) Flexible and Adaptive Planning for Human-Robot Interaction in Self-Optimizing Assembly Cells. In: Advances in The Ergonomics in Manufacturing: Managing the Enterprise of the Future, CRC Press, p 273–283

Frymann J & Matthias B (2012) Safety of Industrial Robots: From Conventional to Collaborative Applications. In: Proceedings of 7th German Conference on ROBOTIK 2012

Hauk E, Gramatke A & Henning K (2008) Cognitive Technical Systems in a Production Environment. In: Proceedings of the Fifth International Conference on Informatics in Control, Automation and Robotics. ICINCO: Madeira, Portugal

Laird J E (2012) The Soar Cognitive Architecture. MIT Press

Liu G & Ramakrishnan K (2001) A*Prune: an algorithm for finding K shortest paths subject to multiple constraints. In: Proceedings of the 20th Annual Joint Conference of the IEEE Computer and Communications Societies, p 743–749

Litoiu M, Woodside M, Zheng T (2005) Hierarchical Model-based Autonomic Control of Software Systems. In: Proceedings of the 2005 workshop on Design and evolution of autonomic application software (DEAS). St. Louis, Missouri, USA

Luczak H and Volpert W (1987) Arbeitswissenschaft. Kerndefinition – Gegenstandskatalog – Forschungsgebiete. RKW-Verlag, Eschborn

Matthias B, Kock S, Jerregard H, Kallman M, Lundberg I & Mellander R (2011) Safety of collaborative industrial robots: Certification possibilities for a collaborative assembly robot concept. In: IEEE International Symposium on Assembly and Manufacturing (ISAM)

Mayer M Ph (2012) Entwicklung eines kognitionsergonomischen Konzeptes und eines Simulationssystems für die robotergestützte Montage. Shaker Verlag, Aachen.

Mayer M Ph & Schlick C M (2012) Improving operator's conformity with expectations in a cognitively automated assembly cell using human heuristics. In: Conference Proceedings of the 4th International Conference on Applied Human Factors and Ergonomics (AHFE), USA Publishing, p 1263–1272

Mayer M Ph, Odenthal B, Ewert D et al. (2012) Self-optimising Assembly Systems Based on Cognitive Technologies. In: Brecher C (ed) Integrative Production Technology for High-Wage Countries. Springer, Berlin

Russel S J & Norvig P (2003) Artificial Intelligence: A Modern Approach. 2nd edition. Prentice Hall

Schlick C M, Faber M, Kuz S & Bützler J (2014) Erweiterung einer kognitiven Architektur zur Unterstützung der Mensch-Roboter-Kooperation in der Montage. In: Industrie 4.0 - Wie intelligente Vernetzung und kognitive Systeme unsere Arbeit verändern, Schriftenreihe der Hochschulgruppe für Arbeits- und Betriebsorganisation e.V. (HAB), p 239–263

Schmitt R, Brecher C, Corves B et al. (2012) Self-optimising Production Systems. In: Brecher C (ed) Integrative Production Technology for High-Wage Countries. Springer, Berlin

Thomas U & Wahl F (2001) A System for Automatic Planning, Evaluation and Execution of Assembly Sequences for Industrial Robots. In: Proceedings of International Conference on Intelligent Robots and Systems. p 1458–1464