The Knowledge Gap: Providing Situation-Aware Information Assistance on the Shop Floor

Mario Aehnelt(✉) and Bodo Urban

Fraunhofer IGD, Joachim-Jungius-Str. 11, 18059 Rostock, Germany
{mario.aehnelt,bodo.urban}@igd-r.fraunhofer.de

Abstract. Situation-aware information assistance strongly depends on the quality of available contextual background knowledge for an application domain and on its automatic processing. In this paper we present a conceptual approach towards using cognitive architectures to provide information assistance and allow complex decision making based on expert knowledge. We transfer our approach into a technical concept which was finally implemented as part of the *Plant@Hand* assembly assistance system within a mobile workshop trolley. The paper gives insights into our work on formalizing knowledge and providing ad-hoc mechanisms for planning, assisting and controlling assembly tasks on the manufacturing shop floor.

1 Introduction

Evaluations show that people with a detailed work plan complete their tasks faster than without it, even if they did not carry out the planning themselves [6]. This is a key motivation for intelligent systems which assist the worker by creating work plans autonomously and guide through single tasks aiming to improve both efficiency and effectiveness of work. Such intelligent systems will help in manufacturing to ensure a high product quality even when working with insufficiently qualified personnel. They need to provide the missing knowledge which is required to fulfill even complex work tasks on the shop floor. Although, manufacturing industries already use powerful data management systems, there is still a lack of methods and technologies which allow an automated processing of the lion's share of domain dependent background knowledge. It is still hidden in unstructured information sources (e.g. standards, guidelines) or simply maintained by experts, thus not available for the average worker.

The paper introduces an abstract model of work related background knowledge. It shows how cognitive architectures are used to bridge the knowledge gap by modeling, applying, and learning domain dependent context knowledge in order to provide situation-aware information assistance at the workplace. Finally, the theoretical approach is implemented in and illustrated with the mobile *Plant@Hand smart assembly trolley*, which guides manufacturing workers through their daily work tasks and assists them at the assembly workplace with detailed situation-dependent task knowledge.

F.F.-H. Nah and C.-H. Tan (Eds.): HCIB 2015, LNCS 9191, pp. 232–243, 2015.
DOI: 10.1007/978-3-319-20895-4_22

2 Related Work

Although, smart factories establish digitalization and automation to streamline manufacturing processes and quality, there is still the need for manual assembly operations. However, we find there a majority of specialized and single task solutions focusing on quality assurance and information transfer. Mayer et al. introduce the usage of intelligent systems to resemble human decision making and problem solving for complex assembly tasks [10]. They use a *cognitive control unit* which ensures the numerical planning of robot behavior backed by a cognitive architecture. Cognitive architectures can be understood as a mean to implement intelligent and autonomous behavior in assistance applications. They have proven capable of supporting even complex problem solving tasks, e.g. for the mission management for unmanned aircrafts [5].

Our own work contributes to the growing demand of information assistance for manual work operations in manufacturing which underlies human flexibility and failures as motivated in [1,3]. In particular, we focus on assistance that can be generated automatically from existing knowledge sources in order to manage todays growing complexity and heterogeneity of extremely small lot sizes.

3 Approach

Similar to formal education processes, information assistance can be understood as an informal way of mediating and *learning facts* (what), *procedures* (how) and *concepts* (why) required for a specific work task. In [2] we showed how assistance artifacts - e.g. embodied by an interactive smartwatch - are used to mediate this work related knowledge depending on a specific *assistance goal* (e.g. remember, understand, apply). However, a technological system which is able to operate the contextual background knowledge in a situation-aware manner was still missing. In our approach (see Fig. 1) we address this knowledge related gap now on three levels:

- on *conceptual level* we establish a relationship between abstract work situations, assistance goals and required knowledge types along the generalized information process,
- on *information level* we map domain dependent context knowledge into formal information rules which control the collection, filtering, and provision of work related knowledge in specific work situations, and
- on *technical level* we use a cognitive architecture to integrate intelligent and situation-aware information assistance with the already existing information infrastructure at the workplace.

We will explain each level carefully in the next sections by introducing the main rationale illustrated with examples from the assembly work domain.

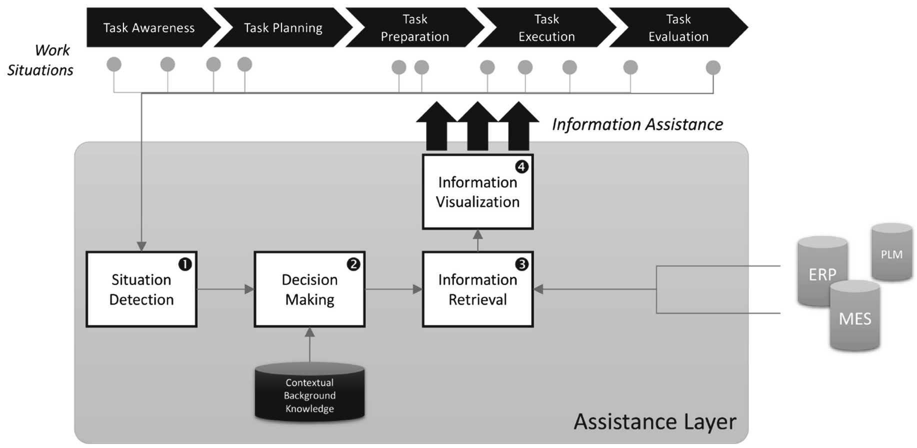

Fig. 1. Conceptual architecture of situation-aware information assistance on the shop floor

3.1 Conceptual Level

Task related information assistance aims at providing the required information needed to understand, carry out, evaluate, or reflect on a specific work task. This means to establish a continuously information process in parallel to the work process as described in [2]. Finally, it is an upcoming information demand during planning, preparing, executing, or evaluating a task which connects work and information processes. Such a demand arises from work situations in which provided information does not satisfy the worker's expectations as well as needs, or the information given is incomplete and even contradictory. What we require here is an ad-hoc mechanism to analyze work situations with respect to the knowledge needed by the worker to carry out tasks without interruptions. We can achieve this by conceptually connecting the work with the information process. This requires the modeling:

- of *work situations* in which knowledge related information demands arise,
- of *knowledge*, which satisfies the information demand,
- of an *information assistance strategy*, which connects the required knowledge with situation dependent assistance objectives to be achieved, and
- of a technical *relationship between knowledge and information sources* which finally maintain and provide the required information pieces in form of structured or unstructured data.

The basis for our further exploration on the conceptual relationship between work situations, assistance goals and knowledge types will be the worker's knowledge-based *control of action* process [12]. Summarizing this process, a conscious perception of information leads to its' interpretation and cognition based remembering in order to actively plan own actions. By the observation and evaluation of action outcomes a learning sub process is triggered.

For our approach we basically simplify the control of action into three phases: action planning, acting, and reflecting. The *planning phase* includes all cognitive steps to get aware of the current situation and plan own actions accordingly. During the *acting phase* the beforehand developed action plan is realized. Afterwards, during the *reflecting phase* the outcomes of own actions are compared against expectations which lead to further learning. The worker requires for each phase specific knowledge in order to plan, execute, and evaluate work tasks efficiently. In Fig. 2 we propose an information assistance strategy which considers this individual knowledge demand by providing small information pieces along the workers own control of action respectively work process. The order and depths of information to be provided to the worker follows an iterative sequence of assistance goals as described in [2] which consequently addresses the increasing knowledge demands of the work process.

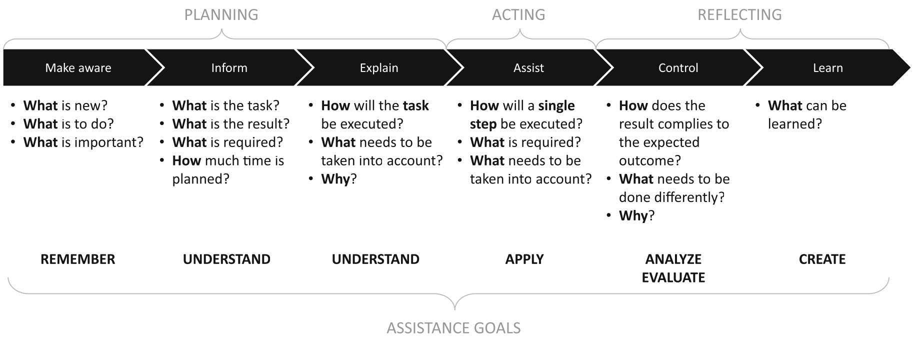

Fig. 2. Conceptual relationship between assistance process, knowledge types and process dependent assistance goals

Once a new situation occurs, e.g. a new work task which was scheduled from the manufacturing execution system shortly before, the information assistance makes the worker *aware* of this new task and provides help in remembering this information. Before executing the new task the worker requires in depths knowledge to understand what and how to do. We propose here a two-staged information assistance based on *informing* first in general and *explaining* details on demand. Once the task was understood, information assistance comes into the role of assisting the task execution, e.g. by monitoring and visualizing task parameters, showing step-by-step guidance, or similar [1], to ease the application of provided information. Finally, it is again the information assistance which provides required information pieces when the work result is controlled. It then supports the *analysis* and *evaluation* as well as *creation* of new insights and knowledge.

On conceptual level we link with this approach work situations, which require a new action strategy, with an accompanying information process delivering the required knowledge in parallel to the progressing control of action process of the worker. However, information sources in manufacturing, such as manufacturing execution systems or product lifecycle management systems, normally do not provide all information as required for systematically assisting the worker. A lion's share of *contextual background knowledge* which is needed to plan, execute and evaluate work tasks are domain dependent and subject to individual expertise. The next section addresses this very specific challenge when providing ad-hoc and situation-aware information assistance.

3.2 Information Level

The previous section explains our conceptual considerations with respect to acquiring knowledge on actual work situations as well as to providing information assistance at the workplace. Looking at the manufacturing shop floor, a vast amount of required information can already be found in manufacturing data management systems. However, the major share of procedural and conceptual knowledge is not yet formalized in systems which allow their automated processing (see Fig. 3). We find work instructions, standards, or assembly guidelines normally written in natural language within accompanying documents. There has already been research to distinguish between the semantic meaning of instructions and their visual representation [9] based on controlled vocabularies which allows for automation. They still require a manual authoring of instructions for each work process individually, e.g. for the assembly of parts and components. But, what we need for ad-hoc information assistance is a dynamic mechanism which continuously observes the current work situation to decide which information is provided according to the situational demand.

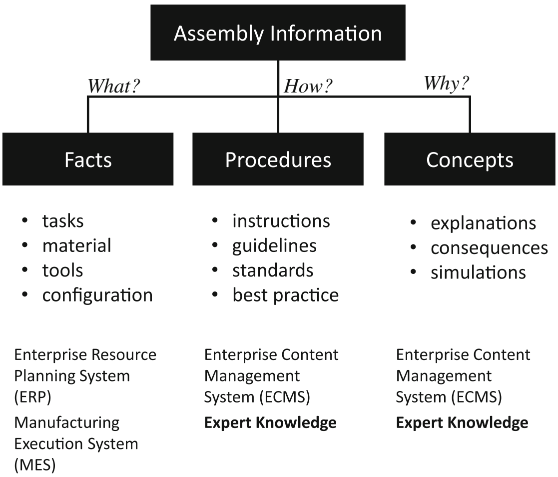

Fig. 3. Required information is contained in different enterprise resources. Partly it is individual expert knowledge which is not externalized in any management system [1].

On information level, we distinguish three major types of work related contextual background knowledge which needs to be provided by information assistance (see Fig. 3). *Facts* contain objective data with respect to tasks, material or tools to be used or a machine configuration for example. Specific work orders or step-by-step instructions are encoded in *procedures* while further explanations are contained in *concepts*. But how can we enable the machine processing of such contextual background knowledge in order to improve the quality of information assistance?

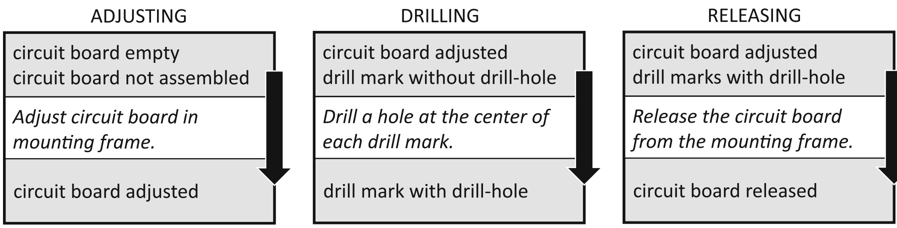

Fig. 4. Contextual background knowledge is modeled in information rules. The example describes very simple the order of assembly steps combined with a textual information for the worker. At the top of each rule are characteristic states of the work environment and at the bottom a target state which will be reached once the assembly step was successfully executed.

In our approach we model knowledge with formal *information rules*. An information rule encodes required knowledge as processable information in dependency of a specific work situation. The situation is identified by characteristic conditions or states of the work environment, e.g. materials, parts, or tools. This helps us for example to formalize even the hitherto missing procedural or conceptual knowledge. Thus, knowledge related to specific work procedures and routines can be expressed within modular rules. In our example (see Fig. 4), we formalized parts of the German industrial standard VDI 2860:1990 [4] in very simple information rules which declare the structure and sequence of assembly steps in case of a drilling operation. We can now use the same rules to model more complex knowledge, e.g. possible solution strategies in case of assembly issues or even the selection and filtering of information sources which provide further operational data (planned figures, construction details, etc.). The modularity of information rules guarantees knowledge modeling in very different granularities. New knowledge can simply be added in a new rule declaring the situational dependencies and thus extending the existing knowledge base.

But, the more information rules with the same or even similar situational conditions are modeled, the more complex grows the decision making for valid and applicable rules in a specific work situation. Therefore, the next section will take a look at the technical perspective of situation aware information assistance.

3.3 Technical Level

In Fig. 1 we developed a conceptual view onto the integration of intelligent and situation-aware information assistance with the existing technological infrastructure on a shop floor. Basically, we have to deal here with a heterogeneity of available information sources and technical systems including their native interfaces and protocols. *Enterprise service bus* technologies reduce the effort of connecting all system components on technical level by providing an universal messaging bus which interconnects each system on data and procedural level as well. This component can be understood as functional equivalent of the *information retrieval* component (3) in our conceptual approach.

The observation of work situations requires a multitude of different *sensors* within the work environment. Previous work [1,3] showed that a basic instrumentation of material and tools already provides us with a sufficient quality of sensor data for following analysis and interpretation steps.

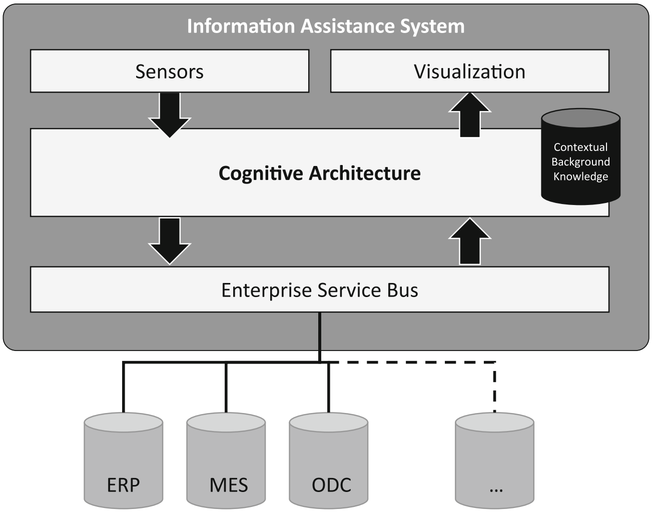

Fig. 5. The technical architecture of an information assistance systems shows a layered layout of interaction, logical and data related components. The system uses an enterprise service bus approach to connect with existing shop floor management systems, such as enterprise resource planning (ERP), manufacturing execution (MES), operational data collection (ODC), as well as other information sources (...).

The cornerstone of our technical architecture (see Fig. 5) is represented by a logical component, the *cognitive architecture*. It is responsible for the functional elements *situation detection* (1) and *decision making* (2) in Fig. 1. As of today, a cognitive architecture transfers the mental structure for human information processing into a technical system, which specifically includes the representation and organization of knowledge within these structures as well as the functional processing required to acquire, use, and modify knowledge [8]. Cognitive architectures can be traced back to Newell's early hypothesis that any artificial intelligence is based on a symbol system and related rules [11]. Hence, they establish technical equivalents for long-term and short-term memories as well as for cognitive processes, such as learning or remembering which are required for an intelligent information assistance system. Cognitive architectures represent and interpret knowledge in a similar formal way as introduced in Sect. 3.2. Within our approach we use it specifically for:

- *situation detection* based on sensorical observations of the physical work environment,
- the *formalization and processing* of contextual background knowledge (e.g. procedures, explanations), as well as for
- *decision making* in order to structure and control the work process itself based on knowledge and data from the work domain, and for
- *learning* new knowledge and practices from observation.

Finally, a *visualization* component communicates the previously selected knowledge as visual information embedded in the ongoing work process.

The following section explains how this still abstract conceptual and technical approach has been implemented in an industrial application which supports assembly workers on the shop floor.

4 Industrial Application

The proposed approach has been implemented within the *Plant@Hand* assembly assistance system which was already introduced in our previous work [1–3]. We used *Soar* as cognitive architecture which enables us now to analyze the specific work situation and to make decisions on missing information which an assembly worker needs to plan, execute, and evaluate his work accordingly.

The whole system was developed to provide mobile information assistance for the assembly part of the manufacturing shop floor. It is required to support there assembly work on complex special units which require a great amount of worker flexibility and mobility. The next sections describe the technological setup as well as our use of Soar for automatic knowledge processing.

4.1 Technological Setup

The assembly of partly large and complex special units requires from the worker a high degree of flexibility and mobility. Components need to be assembled in varying complexities at different locations of the special unit. This makes it difficult to instrument the work environment with activity recognizing sensors. An instrumentation of the worker is also limited due to safety reasons. Because of this challenging conditions we use a standard mobile workshop trolley (see Fig. 6) as technical basis for our assembly assistance application. Such a unit is normally used to store and transport tools as well as material during an assembly.

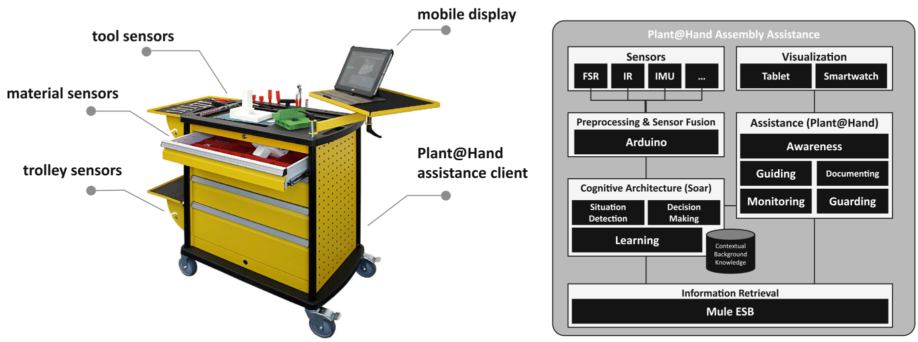

Fig. 6. Hardware and software components of the industrial *Plant@Hand smart assembly trolley* prototype.

The trolley provides shelves and drawers for different sorts of assembly tools or small to medium sized work materials. All hardware and software components of the *Plant@Hand* assembly assistance system are built into the mobile workshop trolley:

– **Sensors:** We use different sensor types to monitor the ongoing assembly activities of the worker. *Force sensitive resistors* (FSR) and *infrared sensors* (IR) provide data on material and tool usage, e.g. the removal of screws from a material container. *Inertial measurement units* (IMU) give us information on the trolley movements.
– **Preprocessing and Sensor Fusion:** An *Arduino Uno* board is used to make a first preprocessing (filtering) and fusion of incoming sensor data. It generates activity events for further interpretation in the Soar cognitive architecture.
– **Cognitive Architecture:** The cognitive architecture *Soar* provides the functional subcomponents for situation detection out of the incoming trolley events, decision making on required assistance actions and information to be provided as well as learning from observed activities.
– **Information Retrieval:** For the connection with external manufacturing data systems the enterprise service bus system *Mule ESB* is used. Based on information flows, data is continuously exchanged which guarantees a provision of the latest information.
– **Assistance:** The main implementation of assistance functions can still be found within the *Plant@Hand* assembly assistance system. Functional blocks, such as raising the workers awareness, the step-by-step guiding of assembly works, or the documentation of work results are part of the *Plant@Hand* assistance client.
– **Visualization:** The mobile workplace requires also a mobile visualization of information for the worker. We use here mobile displays which are still available during the assembly task execution. Provided displays are tablets and even smartwatches [2].

With this setup we are independent from specific or stationary workplaces. The workshop trolley is positioned close to the assembly site and by observing the worker's activities with respect to material, tools and trolley, we obtain a sufficient overview on the ongoing work process to provide required knowledge with the P*lant@Hand* assistance system.

4.2 Knowledge Processing with Soar

In Sect. 3.3 we described the cognitive architecture as cornerstone component of our technical approach. It provides us with the functional abilities to add intelligent behavior to a static assistance system. In the following, we will explain the role of Soar for situation-aware knowledge processing in relationship to assisting the assembly worker. We use the generation of step-by-step instructions for novel assembly units as illustrating example. In general, processes in Soar are related to the gradual alternation of information and states in working or long-term memory [7]. Here, a situation is formalized as a state in working memory, which is modified by evaluating and applying *operators* until an intended final state is reached. The operator definition consists of required conditions and actions on the working memory. It inherits procedural and conceptual knowledge from the corresponding knowledge domain. New operators can also be derived by observation of decision making and through learning processes (*chunking*).

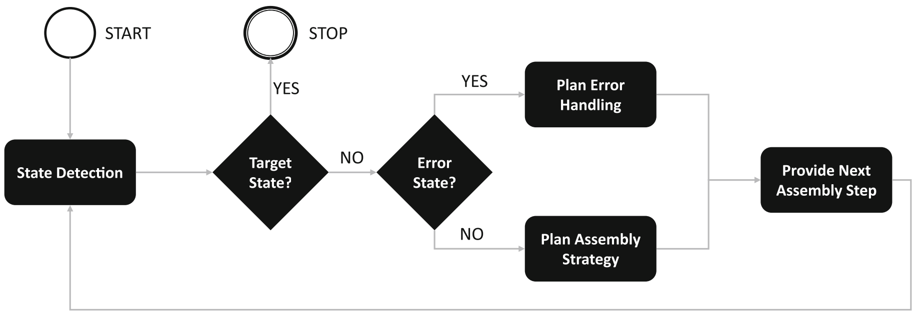

Fig. 7. Working cycle of Soar for the assembly sequence problem space.

Traditionally, the work instructions for an assembly are manually prepared by an expert. Based on the geometric model of the unit to be assembled, predefined production sequences and expert knowledge, each assembly step is textually and pictorially described including dependencies to a bill of materials as well as tools to be used. Soar helps us now to automate this time-consuming authoring process. Explaining our approach briefly, we first store expert knowledge related to the structuring and sequencing of assembly tasks in Soar's procedural memory (long-term memory). Additionally, we define a target state (unit assembled) as well as error states (wrong material, wrong tool, etc.) and leave it up to

Soar finding a valid solution for the *assembly sequence problem* (see Fig. 7). Soar uses now its operational decision making based on the previously stored assembly knowledge in order to plan an assembly strategy which finally leads to an assembled unit. Each assembly step is then visualized to the worker including all information required to understand, execute and control the step. In parallel Soar learns new knowledge through observing the decision making itself as well as the work environment. This knowledge is used in a similar situation to improve the decision quality. This has the positive effect, that during decision making the working memory complexity of Soar dramatically decreases in comparison to the same situation without the previously learned knowledge chunks.

Table 1. Comparison of two experiments without additionally learned production rules (*chunks*) and with using this additional knowledge for solving a small subassembly problem.

Category	1st Run	2nd Run (with learning)
required decisions	169	22
elaboration cycles	650	67
production firings	2.548	163
working memory changes	9.097	761
CPU time	0,047 sec	0,005 sec

In our experiments (see Table 1) we could achieve a twelve times smaller working memory complexity (amount of modified working memory elements) resulting in a nine times faster processing to solve the assembly sequencing problem. Here we used a subassembly of electronic components as testing environment which included the planning of handling, drilling, inserting, and controlling activities for several electronic parts into a circuit board.

5 Conclusions and Future Work

In this paper we introduced a threefold approach of bridging the knowledge gap when providing situation-aware information assistance. Based on the conceptual linking between work situations, assistance goals and knowledge types, we focused on the autonomous processing of contextual background knowledge with a cognitive architecture on information as well as on technical level. Finally, we presented the implementation of our approach within the *Plant@Hand* assembly assistance system which was integrated into a mobile workshop trolley. Here we used Soar for the autonomous planning and problem solving of assembly sequences.

Although, we could achieve with comparatively little modeling efforts good results in solving even complex assembly sequencing problems, the authoring of contextual background knowledge as production rules of a cognitive architecture is still one of the remaining major issues for the automation of information assistance. Here we need to investigate further into alternative authoring as well as learning approaches in the future in order to make it suitable for everyday use.

Acknowledgements. This research has been supported by the German Federal State of Mecklenburg-Western Pomerania and the European Social Fund under grant ESF/IV-BM-B35-0006/12.

References

1. Aehnelt, M., Bader, S.: Information assistance for smart assembly stations. In: Loiseau, S., Filipe, J., Duval, B., van den Herik, J. (eds.) Proceedings of the 7th International Conference on Agents and Artificial Intelligence (ICAART 2015), vol. 2, pp. 143–150. SciTePress, Lisbon (2015). http://dx.doi.org/10.5220/0005216501430150
2. Aehnelt, Mario, Urban, Bodo: Follow-me: smartwatch assistance on the shop floor. In: Nah, Fiona Fui-Hoon (ed.) HCIB 2014. LNCS, vol. 8527, pp. 279–287. Springer, Heidelberg (2014)
3. Bader, S., Aehnelt, M.: Tracking assembly processes and providing assistance in smart factories. In: Duval, B., van den Herik, J., Loiseau, S., Filipe, J. (eds.) Proceedings of the 6th International Conference on Agents and Artificial Intelligence, ESEO, Angers, Loire Valley, France, 6–8 March 2014, vol. 1, pp. 161–168. SciTePress, S.l. (2014). http://dx.doi.org/10.5220/0004822701610168
4. German Engineers' Association: Vdi 2860:1990–05 assembly and handling; handling functions, handling units; terminology, definitions an symbols (1990)
5. Gunetti, P., Dodd, T., Thompson, H.: Simulation of a soar-based autonomous mission management system for unmanned aircraft. J. Aerosp. Inf. Syst. **10**(2), 53–70 (2013)
6. Kokkalis, N., Köhn, T., Huebner, J., Lee, M., Schulze, F., Klemmer, S.R.: Taskgenies: automatically providing action plans helps people complete tasks. ACM Trans. Comput. Hum. Interact. **20**(5), 1–25 (2013)
7. Laird, J.E.: The soar cognitive architecture. Artif. Intell. Simul. Behav. Q. **134**, 1–4 (2012)
8. Langley, P., Laird, J.E., Rogers, S.: Cognitive architectures: research issues and challenges. Cogn. Syst. Res. **10**(2), 141–160 (2008)
9. Mader, S., Urban, B.: Creating instructional content for augmented reality based on controlled natural language concepts. In: Proceedings of 20th International Conference on Artificial Reality and Telexistence (ICAT 2010) (2010)
10. Mayer, Marcel Ph, Odenthal, Barbara, Wagels, Carsten, Kuz, Sinem, Kausch, Bernhard, Schlick, Christopher M.: Cognitive engineering of automated assembly processes. In: Harris, Don (ed.) Engin. Psychol. and Cog. Ergonomics, HCII 2011. LNCS, vol. 6781, pp. 313–321. Springer, Heidelberg (2011)
11. Newell, A.: Physical symbol systems*. Cogn. Sci. **4**(2), 135–183 (1980)
12. Reason, J.T.: Human Error. Cambridge University Press, Cambridge, New York (1990)