Integrated Product and Process Modelling for Construction Robot Control

Ronald Krom and Frits Tolman,
TNO-Building and Construction Research together with the faculty of Civil
Engineering of the Delft University of Technology,
PO Box 49, 2600 AA Delft ,The Netherlands

This paper discusses the role of the product modelling technology in construction robotics control. Information models for the representation of construction product and process information enable more intelligent behaviour of construction robot tools, because construction knowledge can be transferred into the robots. In this paper an information model is presented for product as well as process information. The power of such a model is explained with help of an example of a tiling robot which is simulated in a robot simulation system. With the simulation system we also demonstrate robot independant control via application interfaces which allow one process plan to be interpreted by different robot configurations.

1   Introduction

The special character of the building and construction industry has prevented the use of the "state of the art" robot technology as used currently in a growing number of industries such as for instance the automotive industry. Current robots are not yet advanced enough to cope with the requirements of the building industry. For example they do not have sensors to cope with the unstructured and constantly changing construction environment; their weight is often too large to be shipped easily from site to site, and their flexibility is not high enough to allow easy re-programming in each new situation. This last problem is the problem we focus on in this paper. The paper describes our research on the realization of highly flexible, intelligent or perhaps autonomous behaviour of construction robots with help of product modelling technology. We believe that high flexibility is an essential factor in the wide scale success of construction robotics.

The way in which the current state of the art robots are programmed is not very suitable for the construction industry. In the "traditional" robot applications, robots will perform tens of thousands or even millions of the same operations. In such situations it does not always matter how much time is spent on robot programming because the programming is only done a few times. But in the building industry, where mostly "one of kind" products are built, the effort needed to program a robot must be in proportion with the operation to be performed by the robot. There is no sense in spending half an hour on robot programming if the work can be done in half an hour by hand. Operations in the building industry will only be performed 1, 2 or perhaps 50 times, depending on the application. We would like construction robots to have some knowledge, intelligence or even autonomous behaviour. In order to realize this we need advanced Computer Aided (or perhaps Automated) Robot Process Planning (CARPP) systems which can prepare all information required to enable construction robots to behave adequately. When efficient CARPP systems can be realized, the flexibility and the effective usage of construction robots can be increased significantly. Higher flexibility and higher efficiency will increase the number of (sub-)contractors which can afford construction robot

Wim Gielingh
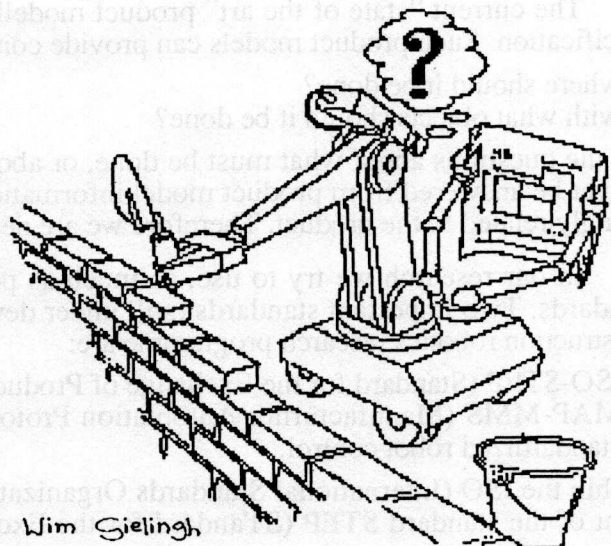
technology and thus stimulate construction robotics.

The question remains: "How can we realize CARPP?". We believe that product modelling technology plays an important role. Using product models extended with process information, we can formalize *construction knowledge*. The choice to describe construction knowledge in terms of product information is based on the idea that it makes much more sense to program a robot in terms like "pick up tile" or even "place door" than in terms of "move gripper x,y,z,a,b,c; close gripper". We have called our integrated product and process model a *production model*.

In chapter 2 we investigate what kind of information is needed for construction robot process planning and what information needs are supported by product modelling. Chapter 3 describes the structure of our production model and how we can these production models in our CARPP system. Chapter 4 describes a test-case of a tiling robot in the simulation system Robcad. Chapter 4 gives our main conclusions.

2 Robot Process Planning

The ideal construction robot would be at least as flexible and autonomous as a labourer. Unfortunately the state of the art technology is not advanced enough to realise this on a large scale. However we would like construction robots to have a certain level of intelligent behaviour in order to have some autonomy. Therefore we need to transfer *construction knowledge* into construction robots, enabling robots can make their own decisions. The more decisions can be made by the robots, the more intelligent and autonomous their behaviour can be.

In this paper we look at the term 'construction knowledge' from the construction robot's point of view. From this point of view we can define construction knowledge as information needed to install, assemble or create an object. Construction knowledge includes the answers to questions such as:

- what must be done?
- where should it be done?
- with what objects should it be done?
- in which order should it be done?
- using which tools?
- using which materials?

These questions can be divided into two groups:

- product related construction knowledge
- construction robot related construction knowledge

The last two of the above listed six questions depend on the robot which is used. We will look at the *product* related construction knowledge. Our aim is to describe the product related construction knowledge in such a way that robots of different vendors can perform the tasks.

The current "state of the art" product modelling theory mainly focus on the product design specification. Such product models can provide construction robots with answers to the questions:

- where should it be done?
- with what objects should it be done?

But the questions about what must be done, or about the order in which things must be done, can not yet be answered from product model information. However the answers to these questions are strongly related to the product. Therefore we are developing *production models*.

In our research we try to use, as much as possible, the results of internationally accepted standards. Two important standards (still under development) which play an important role in our construction robotics research programme are:

- ISO-STEP (Standard for the Exchange of Product model data) for product modelling
- MAP-MMS (Manufacturing Automation Protocol-Manufacturing Message Specification) for standardized robot control.

Within the ISO (International Standards Organization) a group has been working on the development of the standard STEP (STandard for the Exchange of Product model data) for a number of

years. The aim of STEP is to provide a standard for the exchange of product model data. The need for a standard such as STEP evolved from the problems with information exchange between CAD systems.

A more global overview of the complete construction robotics research at TNO, and especially the use of MAP-MMS for vendor independent robot control is given in [Krom'90a].

3 Production models: Integrated product and process modelling

As mentioned in the previous chapter, product models can provide a construction robot with some of the information which is needed to do their jobs. But product models do not provide robots with a process plan containing the activities to be performed and the sequential relations between these activities. To provide this information we extended a product model with process information. We have called this model a *production model*. In this chapter we explain how such a production model is structured. Paragraph 1 explains the product modelling technology on which our production models are based, paragraph 2 explains our process modelling. Finally in paragraphs 3 and 4 it is explained how product and process modelling are integrated.

Product modelling

In our research we use the GARM (General AEC Reference Model) [Gielingh'88] as a reference model for product definition GARM product models are specified as a set of entities of which the most important are called Functional Units (FUs) and Technical Solutions (TSs). Between FUs and TSs, different (kinds of) relations can exist. A FU is an entity in which requirements of a (part of) a (to be designed) product are specified. Every FU is fulfilled by a TS. A TS has characteristics which fulfil the requirements of a FU. A TS can be a commercially available article. A GARM product model of a house will for instance probably have a FU "heating". With this FU several requirements for the heating system are specified: fuel, required room temperatures, heat distribution method etc. A possible TS (depending on exact requirements) for the FU "heating" can be a fire place or a central heating system.

One of the most important kinds of entity relations within a GARM product model is the product (design) decomposition structure. The decomposition structure is modelled with the so called FU-TS decomposition. A FU can be fulfilled by a TS which decomposes into a number of lower order FUs. In other words: solve a problem by decomposing it into smaller problems. This decomposition process is repeated until standard solutions can be found. Figure 1 shows a NIAM diagram (data modelling with NIAM is explained in several books including [Nijssen'89]) of the FU-TS decomposition where: a TS fulfils one or more FUs and on the other hand, a TS can decompose into lower order FUs. Figure 1 also shows that TSs corresponds to Realized Units (RU) when the TS has bean realized (Built, Created or Assembled etc). The generalization of FU, TS and Realized Unit is called a Product Definition Unit (PDU). A PDU is an entity in which product properties (either required properties, expected properties or measured properties) are specified. A number of other papers describe (aspects of) the GARM in more detail: [Gielingh'88] and [Tolman'90].
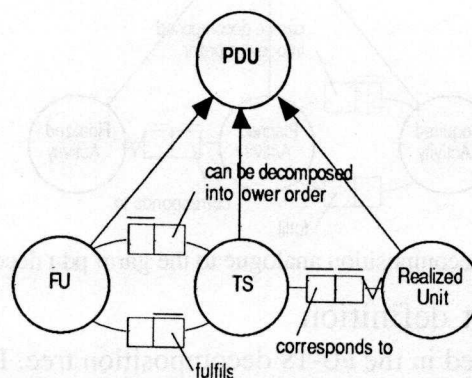

figure 1: The GARM FU-TS decomposition

Process modelling

We have chosen to model the process plan as a *network plan*. Network planning is a universal planning method which allows the determination of, for instance, the *critical path*. In network planning, activities are related to each other in a network structure, where sequences and dependencies of activities are described. Large activities can be decomposed into a number of smaller activities which in their turn form small networks on their own. The structure of a network plan, represented as a NIAM model, is shown in figure 2.
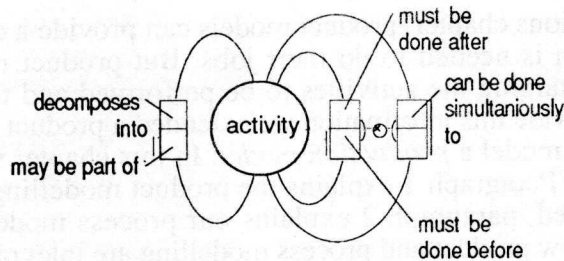

figure 2: NIAM diagram of network plan

The model shows three relations between activities:

- activities can decompose into a number of smaller activities, or an activity may be a part of a larger activity
- some activities must be performed before, or after other activities
- some activities can be performed simultaneously with other activities

So far nothing new. However we would very much like to integrate this network plan model with a product model containing a product design.

Figure 3 shows new entities *required activity*, *planned activity* and *realized activity*, subtypes of a generic entity called ADU (Activity Definition Unit) analogue to the PDU.

A Planned Activity (a process plan) can fulfil a Required Activity, or alternatively a Planned Activity can decompose into lower order Required Activities, for which lower order Planned Activities can be sought. Finally it shows that a Planned Activity can actually be performed and then results in a Realized Activity.

A second analogy between the decomposition of PDUs and ADUs is that relations between lower order Required ADUs and Required PDU's can be modelled in networks. The Required PDU-network is mainly an adjacency network [Willems'88]. The Required ADU-network is the precedence network as given in figure 2.
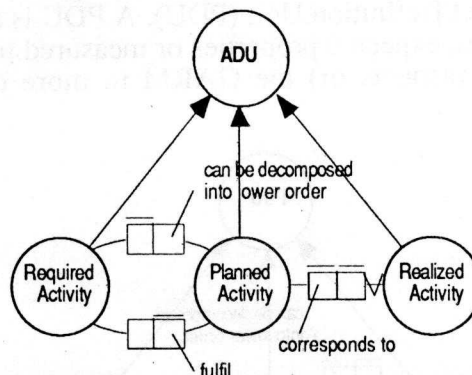

figure 3: ADU decomposition analogue to the garm pdu decomposition

Production oriented product definition

A product design is specified in the FU-TS decomposition tree. In the process plan we would like to plan all activities related to the realization of the technical solutions in the PDU decomposition.

Therefore we decided to categorise all PDU's into categories which need similar kinds of activities for their realization. We decided to follow the terminology introduced by Turner [Turner'88], who distinguishes between:

- system
- sub system
- component

Besides those terms we introduce the term

- feature

Systems are composed of sub systems, sub systems are assembled of other sub systems or components. Components are prepared by realising all features of the components. Features can be created by activities (or operations). The term feature is used in manufacturing industries for all geometric shape properties (of components) which can be realised by shape transformation activities such as for instance milling, drilling, bending etc.

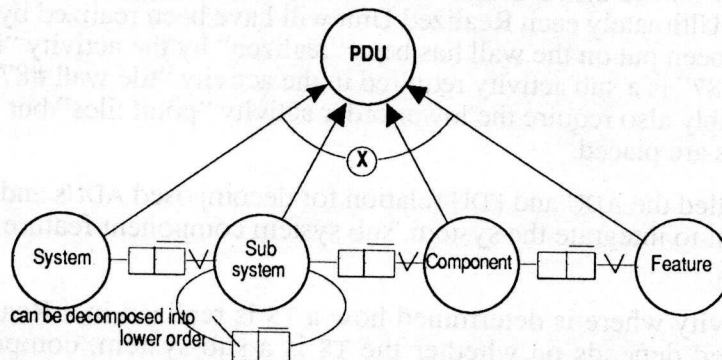The NIAM model of this categorisation is shown in figure 4.

figure 4: NIAM diagram of relations between system, sub system, component and feature

Every PDU is either a system, sub system, component or feature. Systems consist of one or more sub systems, sub systems can consists of components which can have features.

The relation between product model and process model

So far we have presented a information structure for a (network) process plan of activities and an activity related PDU specialisation. The next step is to model the relation between PDU's and ADUs. The global relation between ADU and PDU is very simple as is shown in figure 4. With every PDU there exists an ADU which realizes the PDU.
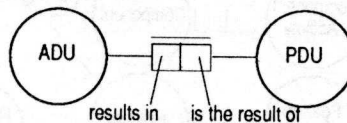

figure 5: Relation between ADUs and PDUs

This simple model give the connection of ADUs and PDUs and is therefore the basic structure of our production model.

As the next step we substitute the PDU decomposition and the ADU decomposition of figure 3 in he model of figure 5. The result of this substitution is shown in figure 6.
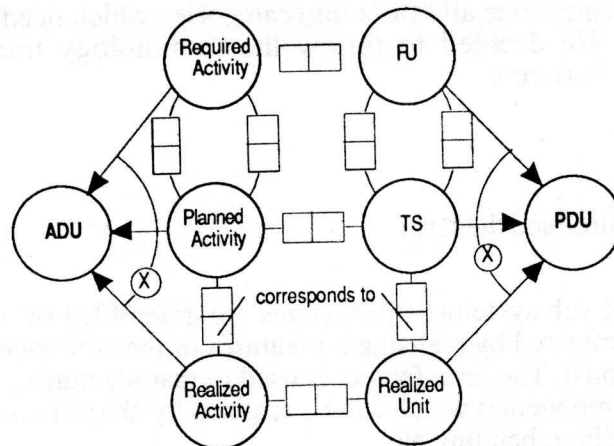
figure 6: Decomposition of ADUs

The model in figure 6 shows the ADU and PDU decomposition together with the relations between the two hierarchies. Ultimately each Realized Unit will have been realized by a Realized Activity. A tile which has been put on the wall has been "realized" by the activity "install tile #187". The activity "install tile #187" is a sub activity required in the activity "tile wall #873". The activity "tile wall #873" will probably also require the lower order activity "point tiles" but his activity can only be done when all tiles are placed.

We have now modelled the ADU and PDU relation for decomposed ADUs and decomposed PDUs. We would now want to integrate the system, sub system component feature specialisation into the ADU-PDU relation.

Planning is the activity where is determined how a TS is realized in a Realized Unit. What kind of activity is required depends on whether the TS is a sub-system, component or feature. Substituting the categories given in figure 4, leads to the model shown in figure 6.
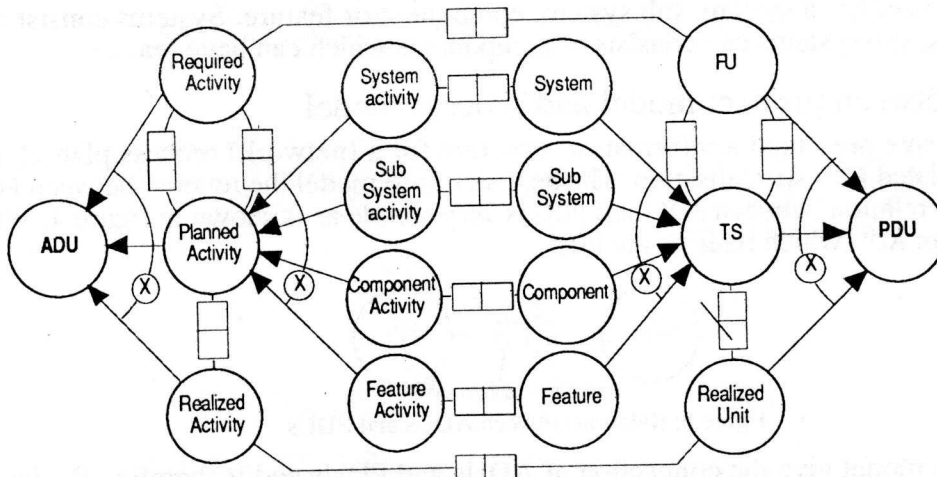

figure 6: Conceptual schema of the kernel of the production model

The conceptual schema of the production model of figure 6 gives the proposed relations between PDU's (with known shape and material properties) and ADUs, without specifying how the ADUs should be performed; e.g. which tools to use etc. Every robot activity operates on one or more objects that fall into the PDU classification of figure 3. Each PDU category has a group of related activities attached. Feature activities relate to simple manufacturing operations on components. Component activities relate to assembly operations etc.

In the next section we will discuss the role of production models for construction robotics control.

4 Production models in construction robot control

In (nearly) every building project (at least in Europe) several different project partners are involved. These partners are mostly the architect, a contractor and a number of subcontractors carrying out different tasks in the building process. When we want intelligent robots, the robot system must be able to acquire knowledge; knowledge about the building design from the architect, knowledge about the materials and parts to handle from the suppliers and perhaps knowledge about the project planning of the contractor. Of course each subcontractor can have several different construction robots. The construction industry needs CARPP systems which is able to communicate with all computer systems of the other project partners, and control different kinds of construction robots intelligently.

In the CARPP system information from different sources is merged. Information can only be merged when the different information models contain common elements. For instance in a product model from the architect it is indicated that the walls in the bathroom are to be finished with tiles up to a height of 2 meters above the floor. A CA-tiling-RPP system can only understand this product model if it recognizes the TS "tiled surface". When the CARPP system has knowledge about tiling it can evaluate the area to be tiled and generate –depending on the tiles used– a process plan for the tiling.

In the situation described in the previous paragraph a CARPP system needs communication with:

- the architect; the architect describes what areas are to be tiled and perhaps what tiles are to be used.
- the tile supplier; the supplier of the tiles has the exact information about the characteristics of the tiles: size, the quality, tolerances, best method to apply them to the wall etc.
- the contractor; the contractor arranges the "time slot" in which the tiling can take place. This means that he knows what will be finished in the robot work environment and what not.

Because each of the project partners is in principle independent, standardized communication is required in order to be able to exchange product and process information. When standardized production models (such as for example we are developing) exist, computer communication of construction knowledge between all project partners is possible.

With extended versions of STEP which include the exchange of information such as can be stored in our production models, CARPP can be integrated in a Computer Integrated Construction (CIC) infrastructure. Of course the availability of standardized medium for exchange of production knowledge is not only convenient for construction robot control but opens many other possibilities within CIC. In [Krom'90b] more is said about production modelling.

In the next chapter we explain an example of a production model and show the control of our simulated prototype of a tiling robot.

5 Implementation and simulation

To test our ideas we use the robot simulation system Robcad™. Within Robcad we have modelled a number of simulations of construction robot applications. One of those simulated prototypes is a tiling robot. In this chapter we present this tiling robot and show what kind of information is needed to control the robot. We emphasize the fact that these simulated prototypes are ment for investigation of the *control* and not for their mechanical design!

Our simulated prototype of a tiling robot is shown in figure 7. The robot itself is a standard robot, a Puma 652, on top of a simple AGV like vehicle. The robot tool is a vacuum gripper which can pick up tiles from a crate on top of the AGV. Besides the crate is a glueing device which can apply the glue to the back of the tiles.

To make this tiling robot to work autonomously we need:

- what position to drive to, to be in reach of all places where tiles are to be placed.
- a route for the AGV to drive to its desired positions.
- information about the order in which the tiles should be placed
- information about the size and duration of the force to be applied to the tile before it sticks to the wall
- information when the AGV must drive on a little so the robot does not work out of its reach.

Figure 8a shows the structure of the product design information in a product model for residential buildings [Waard'91]. A product type model is an information model (in this case based on the GARM) for special categories of products. Imagine the design of a house. Within this context a "bathroom" is required. A "bathroom design" is a technical solution for such a (required) "bathroom". This TS

figure 7: simulation of tiling robot

"bathroom design" decomposes into a number of lower order FUs. To these FUs belong the "Elementary space boundaries" of the bathroom . A bathroom requires these "Elementary space boundaries" to be water resistant. "Tiling" is a TS which fulfils this requirement. Figure 7a also shows some adjacency relations between some of the FUs.
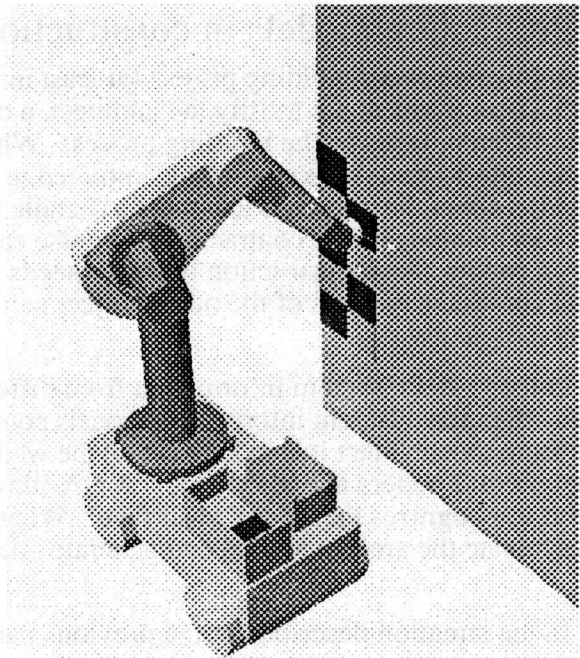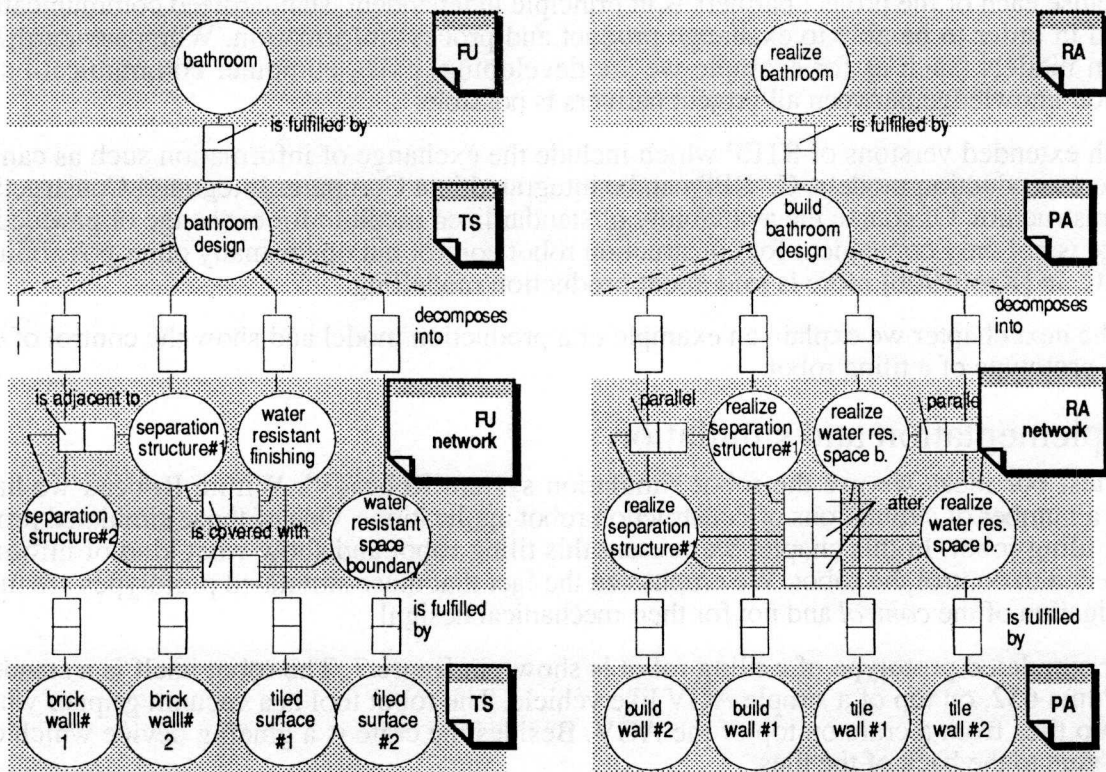

figure 8a: product model structure for bathroom   figure 8b:planning model structure for bathroom realization

Figure 8 shows the process planning structure for the realization of the "bathroom". At the top there is a RA "realize bathroom" which is fulfilled by the Proposed Activity (PA) "build bathroom design". This activity decomposes into a number of lower order required activities. These activities all have relations which form a network of sequential dependencies of the different activi-
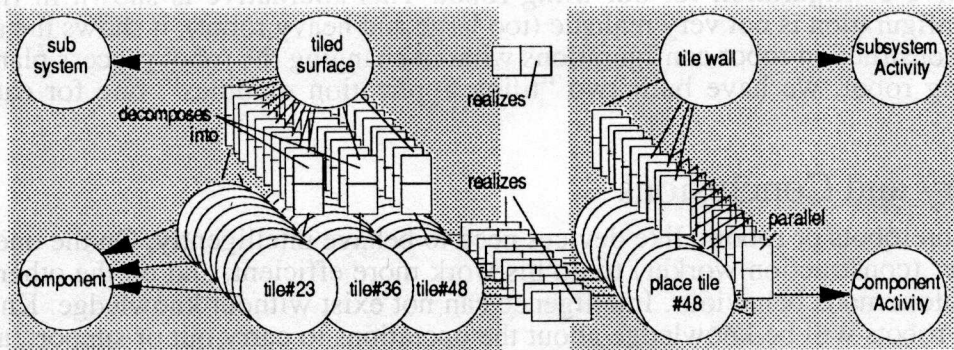
figure 9: Detailed design and planning for tiling robot control

ties. E.g. the tiling can only be done when the walls are realized. This realization process can for instance include masonry work and the drying of the cement.

The level of detail of the product model structure and the process model in figure 8 are not yet precise enough for robot control. Figure 9 shows how the TS "tiled surface" decomposes into a collection of tiles. This design detailing will probably not be done by an architect but by the sub-contractor who will do the tiling. In this design detailing process he can determine where for example the cut-off tiles are to be used (e.g. preferably on the left-side or perhaps on the right side). Such design decisions can simplify the use of robots and is a (simple form of) design for construction (with robots). This design detailing can probably be automated because it can be formulated in a procedure. This procedure can generate a global process plan at the same time. In principle, each tile can be applied to the wall independently of any other tile and thus be done parallel. At this level of detail the process plans must be transferred to the robot. The objective is that the robot can decide on the exact sequence in which the tiles are to be applied. Imagine some futuristic tiling robot which can apply 4 tiles at the same time, such a robots can also use the same planning information. The philosophy is to postpone as much of the decisions to the end and delegate the decisions to the one "who knows what is best". This philosophy allows for instance the robot to decide what to do in case of some kind of exception occurring; the robot can continue tiling at some other place and warn the operator to come and have a look at the problem which triggered the exception. The simple tiling robot such as the one in our simulation system just applies the tiles from left to right and stops in case an exception occurs.

Still each of the required activities at the bottom of the decomposition tree, such as place tile in the process plan shown in figure 9, are to be translated into robot specific instructions. The robot specific instructions are instructions such as "move arm to ..; vacuum on" which depend on the robot hardware configuration being used. To allow robot configuration independent control, each of the configurations must have a standardized control interface. We have called such an interface an *Application Interface* (AI). Behind this AI, all configuration dependent control is done. It is clear that AIs for all different construction robot applications must be developed. However we expect that we can build more advanced and application specific AIs in terms of –one or more– primitive AI instructions. More research is needed to investigate this.

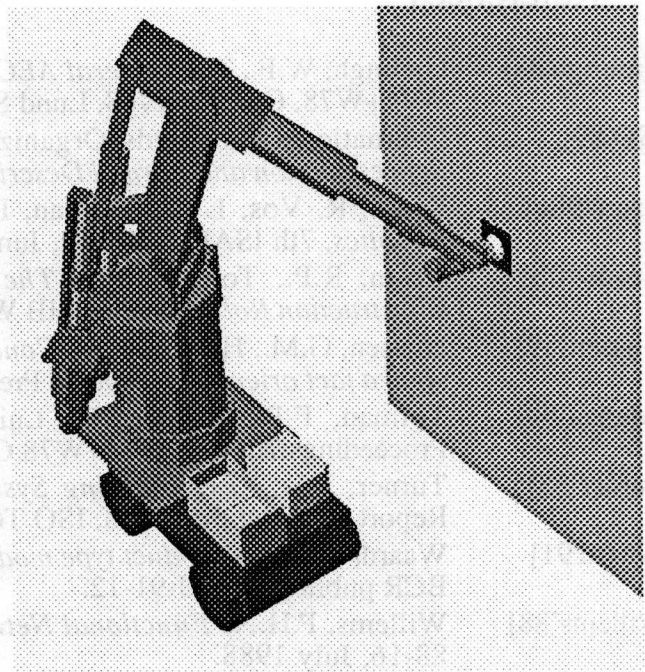To demonstrate the principle of configuration independent control, we have

figure 10: Alternative robot configuration

built an alternative configuration for our tiling robot. This alternative is shown in figure 10. Although the configuration is not very realistic (too large and heavy robot), it shows the ability to use different to construction robot configurations without changing the robot process plans. In the case of our tiling robot, we have built two "tiling application modules", one for each robot configuration.

6   Summary and Conclusions

Construction robots or robotic like devices need to behave intelligently. On the one hand to support their user (construction worker) to do his work more efficient, and on the other hand to realize (semi) autonomous behaviour. Intelligence can not exist without knowledge. Knowledge for construction robots will be knowledge about the operations to carry out or support, about the objects which are dealt with and the environment in which the work must be carried out.

The need for such knowledge requires the process planning systems of construction robots to be integrated with:
- the design systems of the architects
- the databases of the product suppliers
- the planning systems of the contractors

Because each of the project partners use different (computer) systems, standardized media for information exchange are needed. We have designed a production model (integrated product and process model) which enables the formalization of construction to acquire most of the needed information.for construction robot control.

Our production model is based on the GARM and is extended with process planning information. In our simulated construction robotics world we can demonstrate the control of tiling robot using our production model information. In future research we will broaden our production modelling technology by trying to control different kinds of robot applications in our simulation system.

7   Acknowledgements

8   References

[Gielingh'88]   Gielingh, W.F., *The General AEC Reference Model (GARM)*, proceedings CIB W74+W78, October 1988, Lund Sweden.

[ISO'89]   International Standards Organization, *Exchange of Product Model Data – Representation and Formal Description*, ISO DP 10300, February 1989.

[Krom'90a]   Krom, R, Vos, L de, Tolman, F. *Standard components for Construction Robotics*, 7th ISARC, Bristol, June 1990.

[Krom'90b]   Krom, R.P., Tolman, F.P. *The role of standardized Product Models in Construction Robotics*, 2nd CIB W78+74 Seminar, Tokyo, September 1990.

[Nijssen'89]   Nijssen, G.M., Halpin, T.A., *Conceptual Schema and Relational Database design, a fact oriented approach*, Prentice Hall, Sydney, 1989.

[Tolman'90]   Tolman, F.P., Kuiper, P., Luiten, G.T., *Product modelling at Work*, Proceedings 7th CIB W74+W78 Conference, September 1990, Tokyo.

[Turner'88]   Turner, J.A., *AEC Building Systems Model*, IGES/PDES AEC Committee Report, Version 3.6, 1988, ISO TC184/SC3/WG1.

[Waard'91]   Waard, M. de, *A product type model for residentail buildings*, April 1991, TNO-BCR publication PU-91-12.

[Willems'88]   Willems, P.H., *A Functional Network for Product Modelling*, TNO-BCR PU-88-16, July 1988.