AUTMOD3: The Integration of Design and Planning Tools for Automatic Modular Construction

Ramiro Diez, Victor M. Padrón, Mohamed Abderrahim, Carlos Balaguer
Department of Systems Engineering and Automation, Carlos III University of Madrid, Av. Universidad, 30 28911 Leganés (Madrid), SPAIN
E-mail: ramiro.diez@uc3m.es; mohamed.abderrahim@uc3m.es

Abstract: Europe is facing a fast growing need for affordable, high quality housing, which can not be solved by means of traditional or conventional building technology and its associated methods of construction. One of the solutions to this problem is the application of high quality modular construction, which defines a new building process as the assembly of pre-fabricated modules in the construction site. As part of this philosophy, AUTMOD3 is an automatic modular construction software environment, developed in the frame of the EU project FutureHome. The system seamlessly integrates architectural design, planning and simulation tools in a well-known CAD program commonly used by designers. Architectural design using previously defined 3D and 2D modules or modularisation of the traditional building design are the main objectives of the AUTMOD3 design tool. The production planning and the on-site modular assembly planning are the main task to be performed by the planning tool. The AUTMOD3 assembly planning tool generates the trajectories, which can be applied directly to a robotized crane develop in the project FutureHome.

Keywords: Robotics, CAD/CAM Integration, CIC, automatic modular construction, design, planning

1. Introduction

Europe is facing a fast growing need for affordable, high quality housing, which can not be solved by means of traditional or conventional building technology and its associated methods of construction (Atkin, 2000). One of the solutions to this problem is the application of high quality modular construction (Feldman, 2000, Furuya, 2000, Atkin, 1999). This construction technology defines a building process as the assembly of pre-fabricated modules: initially a modular design of the construction project is obtained, then these modules are produced in a factory, and finally, they are assembled on the construction site. The use of pre-fabrication leads to the following advantages: a) structuring of the construction process, which allows the application of automation and robotics and the application of computer integrated construction methods (computer-aided design, planning, control, supervision tools and automation systems), b) reduction of health risks of workers and improving their working conditions, c) increasing the building process predictability (the building process is less affected by weather conditions and there is a greater control on materials use and supply) and d) decreasing waste and increasing productivity.

FutureHome is an European Union funded research project, which is aimed at developing of high quality modular construction technique and its required Information Technology infrastructure for the European construction industry. The participants of the projects are steel and building companies, research centres and universities from Finland, Sweden, Germany, The Netherlands, United Kingdom and Spain (Balaguer, 2002). In the frame of the FutureHome project Carlos III University of Madrid has developed an automated system, AUTMOD3, implemented in the Computer Integrated Construction concept (Takada, 2000, Adeli, 2000, Aouad, 1997, Peñín, 1998). The system consists of a design tool, a planning tool, a simulation tool and a set of databases.

The FutureHome processes are illustrated in Fig. 1. The process “Building and construction process definition” produces the modular design of the building and specifies how the modules should be fabricated. The developed design, planning and simulation tools define the modules and the sequence in which they will be produced, transported and assembled on the construction site. This information is stored in a central database, which allows its access from other processes in the system. The process “Economical planning and evaluation” estimates the cost and time of the project tasks, produces schedules and also, controls and supervises the overall execution of the project. All information produced by this process (estimates, schedules, etc.) is also stored in the central database for further use. During the “Execution Process” the designed modules are produced in the previously planned sequence, transported to the construction site and
assembled there. During this process, all required information is taken from the central database and modified to reflect the current status of the execution: what modules are being produced, transported and assembled in each instant of time.

Fig. 1. The FutureHome processes

This paper describes the AUTMOD3 design and planning tools and their application to the 1:3 scale demonstrator consisting in the design, planning and assembly of several 3D house modules by the robotized crane developed at the Carlos III University of Madrid as part of FutureHome project. Several of the developed techniques and tools have been also used in the full scale residential house built by Corus in their Ijmuiden center (The Netherlands).

2. AUTMOD3 design tool

There are two methods of modular design in AUTMOD3 (see Fig. 2). The first method is based on a traditional architectural design. Architects design and draw architectural plans in a CAD system, and these drawings are processed using a modular division procedure to obtain the modules needed for the house. If there is any problem during the modular division due to the complexity of the design, the system requires architect to make appropriate corrections. In the second method, the design is performed using a catalogue of 3D parametric modules. The library of modules is updated taking into account customer and architect requirements and also development of new materials, accessories and fabrication techniques.

2.1. Structure of the AUTMOD3 design tool

The AUTMOD3 design tool is formed by a set of applications or modules and a group of data libraries (Diez 2000, 2003), with a graphical interface that integrates the different applications (see Fig. 3). AutoCAD© has been selected as the main software package, due to its widespread use and because it incorporates several development tools. The chosen development tool for the implementation is the AutoCAD Runtime Extension (ObjectARX©) (Ransen, 1997). A group of new functions has been created in order to enlarge and adapt AutoCAD to the problem specification. To facilitate the design, these additional software utilities are included in the AutoCAD menu bar. The structure calculations are achieved by a finite element program (CYPE), which is compatible with the AutoCAD data files. Another module carries out the building installations calculation: heating, ventilation and air conditioning (HVAC). The modular division is calculated with an application developed in C++ with ObjectARX. This module is executed from within AutoCAD. It takes data from the drawings performed in AutoCAD with the modifications made by the other modules.

Fig. 2. AUTMOD3 design methods

Fig. 3. Architecture of the AUTMOD3 design tool

The system includes a group of libraries, which contain the module characteristics, the construction criteria or the modular division specifications’ information. These libraries consist of the following:

1. Modules database contains the parametric modules data, their dimension parameters, their accessories, and other characteristics like type of assembly connectors, weight, etc.
2. Structural specifications contain information about the building structure.
3. Division specifications contain criteria to calculate the number and type of modules required to form a building, having as input the 2D architectural design.

2.2. Automatic modularisation of traditional designs
This section presents the automatic modularisation of traditional designs. Fig. 4 shows an example of the apartment block, whose design has been modularised. The real building, upper figure, has been designed and constructed employing traditional methods by the Spanish company Dragados, S.A. The bottom figure is a 3D model representation of the building.

2.2.1. Traditional design
The design process can be grouped in a small number of stages (see Fig. 5). With the specifications and requirements of the client a first design is drawn. The details and accessories are added in the CAD system. The next step is the structural calculation, in order to check that the structure resists and comply with the existent regulations. If the structure does not resist, a return to the design stage is required to perform the necessary changes. This process is repeated until the structure is acceptable. The following stage carried out is the housing installation calculation: heating, ventilation, air conditioning, etc. If any of these installations could not be placed appropriately or it does not comply with the regulations, another return to the design module is required to modify the drawing and repeat the process.

Once this phase is completed, the traditional design is finished.

2.2.2. Automatic modular division
With the data from the 2D drawing and some additional dimensions (mainly heights of different elements), the application calculates the modules, which will constitute the building. In this calculation one must keep in mind a series of restrictions imposed by: the customer specifications, urban regulations (maximum number of floors, height, etc.), characteristics of the ground/land, economical factors, etc. Following the selection of the adequate division criteria the system is instructed to carry out the necessary calculation. The modularisation can be divided in two parts: on the one hand the calculation of the interior structure with three-dimensional modules and on the other, the external walls division in two-dimensional panels. In both cases a library of modules is provided. A series of criteria for the modular calculation and a group of restrictions should be taken in consideration upon applying these criteria.

2.2.3. Types of modules
Automation is related to the repetition of tasks, with big product series, while the architectural design carries implicitly the idea of exclusivity. One could not make a group of different modules for each building. However, the available modules’ catalogue should be extensive enough to not limit the designers’ creativity. The best solution should be a compromise between both ideas. The built modules’ library consists of three-dimensional modules and two-dimensional panels. A distinction is made between structural and not structural modules. The modules can be classified in 3 types: 3D modules, 2D panels and roofs. A fourth type, including special modules could be added to cover any constructive detail that could not be carried out with the basic group.
The three-dimensional modules are structural. Their length will vary from 1.2 to 10.8 m, while the width ranges from 1.2 to 2.4 m (with a 30 cm. grid), all with a typical height of 2.7 m. These modules constitute the volumetric spaces such as kitchens, bath, stairways, etc.
and they should incorporate all the necessary installations (Fig. 6a and 6b). The two-dimensional panels are divided into structural and not structural, where the former is employed to complete the building structure. Their dimensions are 2.7 m height and up to 6 m of length. The floor system panels have a maximum dimension of 6 x 2.7 m. The facades are non-structural panels and of maximum dimensions of 6 x 3 m. (Fig. 6c). The roof modules are three-dimensional modules assembled from two-dimensional panels. In this group other special modules like chimneys are included.

Fig. 6. a) 3D stairs module, b) 3D bathroom module, c) 2D facade panels

Modules data are stored with an object oriented approach. For this purpose several types of classes had been developed and structured in the way to simplify their interfacing with the AutoCAD Data Base. Fig. 7 shows the hierarchical structure of the building called C3Bi which includes 3D modules classes and their corresponding C++ structure.

Fig. 7. Hierarchical structure of the building module classes

2.2.4. Stages in the process of modular division

For the modular division, a two-dimensional traditional architectural drawing with the building plant is taken as start. This drawing should contain, in different layers, additional information like the different elements height (roofs, doors, windows, etc.). The division process is performed in stages as follow:

- Data reading from the drawing. During this step the identities of the drawing in DXF format are accessed and data is read layer by layer and stored in groups according to their type (bath rooms, kitchens, bed rooms, etc.).
- Data preparation. To extract the borders of the rooms, corridors, stairs, façades, etc. the wall contours must be extracted. There are two steps in the algorithm: a) identification of the walls and transforming their representation to poli-line format which will be make easier the next steps. The objective is to obtain as much as possible closed poli-lines and identify the drawings continuous lines. The most complex step is the second one, described in Fig. 8. In order to facilitate these two first steps a set of recommendations should be followed during the design. For example, draw the building by structured layers (façade, walls, etc.).

b) identifies the walls and transforming their representation to poli-line format which will be make easier the next steps. The objective is to obtain as much as possible closed poli-lines and identify the drawings continuous lines. The most complex step is the second one, described in Fig. 8. In order to facilitate these two first steps a set of recommendations should be followed during the design. For example, draw the building by structured layers (façade, walls, etc.).

Fig. 8. Algorithm to search contiguous lines in the building drawing

- Building rooms extraction. The boundary of each room is calculated from the data stored in layers: lines stored in layer wall, doors in layer carpentry, windows in layer windows, etc. Figure 9 shows the sequence followed to calculate inner walls of each room. The process starts from an opening in a wall, which represent a door. The algorithm determines the elements (openings, walls, columns) that form the room’s perimeter by searching contiguous elements. This application determines the building characteristics: elements (beams, walls, doors,
windows, etc.), rooms distribution (kitchens, bath, stairways, etc.).

Fig. 9. Algorithm to search the room’s boundaries

- Boundary analysis. The room’s boundary shape obtained can be irregular as shown in figure 10a. To simplify the modules calculation, these irregularities are eliminated in order to obtain a boundary like in figure 10b. This simplification is realized following the sequence described in figure 11. The algorithm detects small lines in the room boundary like those named shape f1 and shape f2 in figure 10a. Then, it replaces or modifies the lines to produce a simplified contour (see figure 10b).

Fig. 10. Boundary analysis: a) examination, b) rectification

- Rooms division. The result of the previous steps is a list containing the room’s boundary. Nevertheless, some rooms cannot be placed in a module due to their irregular shape (see figure 10b for corridor and 12b for bedrooms). In these cases, the room has to be divided in modules to allow transportation and delivery to the site. Figure 12a shows the division for a corridor and figure 12b presents the modules that form some bedrooms.

- Modular division. The modular division is performed according a set of selected criteria, consisting in a group of rules and a series of restrictions. The aim is to place rooms into modules. There are two ways to calculate the needed modules to erect a building. The first one is based on structural components like columns and load bearing walls. The application reads structural data and performs the calculations with these data (see figure 13, left branch). The second method attempts to place one room in each module. In some cases the modules are too small to form a module, and therefore, they are joined to other modules to obtain a module that follows the minimum dimension criterion. Then other criteria selected by the users can be applied: to place two rooms in a module, to allow big modules, to allow special modules (with irregular shape), to minimize the number of modules, etc. (see figure 13, right branch).

Fig. 11. Algorithm to simplify the room’s boundary

- Data storing in the drawing. The obtain modules are then inserted in the database in an adequate format for their later treatment.

Fig. 12. Room division: a) corridor, b) bedrooms

2.2.5. Criteria of modular division

Several criteria can be applied to perform the modular division. The first one is based on the building structure, where the structural elements are maintained in their position. This criterion only considers structural data to determine the modules that form the building. Figure 14 presents the original 2D architectural drawing (a) and the first criteria modularization (b). The second one is based
on the room distribution; where an attempt is made to fit each room within a module (see Fig. 14 c). Other criteria are based on obtaining the minimum number of modules or the maximum number of identical modules, resulting in a minimum number of different modules. A combination of these methods could be applied to take advantage of the best qualities offered by each. When applying these criteria a series of restrictions should be considered. For instance, the presence of parking basements or other special installations forces the structural elements to maintain their original position. The land characteristics and the location are other factors to be considered, in addition to the transportation to the site, which limits the maximum modules size.

Fig. 13. Modular division criteria application

2.3. Modular design using a catalogue of modules

The basic set of modules described in Section 2.2.3, has been employed to design basically two types of housing: rows of terraced houses and rows of detached houses. The objective is to design a variety of houses using a small number of different modules from the catalogue. The modules of the catalogue are equal in the sense they have the same structure, the same size, the same shape and the same functionality. Nevertheless, these modules can have different accessories, for example, the carpentry elements could be placed in different positions or could have different dimensions, color, qualities, etc. The modules can be manufactured with a higher degree of automation in a flexible production line, because the modular structure is the same for a big number of produced modules.

A specific tool has been developed to assist the designer in the drawing task. In order to facilitate the design, a group of additional functions are included in the menu bar of AutoCAD. These process-oriented utilities are
dialogue boxes that guide the design process in an easy and friendly way. Figure 15 shows the design of a stairs-case 3D module. The user selects the type of module to insert from a menu integrated in AutoCAD. Then a wizard composed by several dialog boxes is displayed. The user selects the module characteristics in these dialog boxes: stairs properties (type, material, etc.), position and orientation, etc. Then, the module is inserted in the drawing and added to the database.

3. AUTMOD3 planning tool

The production planning and the on-site modular assembly planning are the objectives of the AUTMOD3 planning tool. To accomplish this task an assembly planner has been developed (Padrón 2002). This planner defines the sequence, in which pre-fabricated modules should be produced, transported and finally assembled on the construction site. Also, it will generate trajectories and programs for the automated devices involved in the on-site assembly.

Assembly planning is a very complex problem when it is viewed in the context of robots and other automated devices programming (see Fig. 16). Assembly planning defines the assembly operations and their sequence, the fixture design, the tool selection and the workspace layout, satisfying a set of technical and economical requirements. This process produces the assembly sequence, which is executed by an automated device. In order to do this, it is necessary to plan the grasp points and the grasping of each module, the path planning from the module initial position to the destination position and the execution of the fine motion to insert the module, making a suitable use of the device sensors during all these operations (Wolter, 1988).

A key concept to obtain a successful automated assembly system is the integration of the planning process with the design and execution processes (Nevin, 1989, Boothroyd, 1992). This concept has been applied in FutureHome. On one hand, specially developed assembly connectors have been added to the pre-fabricated modules to allow a bigger tolerance of robotized crane movements and to help the insertion of modules. In addition, the crane has been provided with a platform carrying electromagnets and an artificial vision system in order to locate, grasp and insert modules in an automatic way. This approach permits the simplification of the high level programming allowing to decompose assembly planning in two simpler processes: assembly sequence planning and motion planning. Assembly sequence planning searches the order and spatial direction in which the insertion of each module should be done. Motion planning defines the spatial trajectories of the modules during the assembly and the automated devices commands needed for their execution.
3.1. Assembly sequence planning

Assembly sequence planning searches the order and spatial direction in which the insertion of each module should be done. This sequence should satisfy the following types of restrictions and criteria: a) geometrical restrictions, b) physical restrictions, and c) assembly process related restrictions and criteria. Geometrical restrictions are those related to the collision between parts in the assembly process. Physical restrictions are those related to the unwanted changes due to gravity, unwanted motion, etc. (these restrictions are also defined as the assembly stability). The assembly process (available fixtures, tools and workspace) can be specified to the assembly sequence planner by means of a set of restrictions and criteria. A criterion, unlike a restriction, is not a prohibition, but it is a measure for evaluation. This kind of restrictions and criteria allow the generation of different sequences and the selection of the most suitable one.

Before to proceed further, it is necessary to bring to mind some assembly planning terms (Romney, 1992):

- The number of hands required to perform a given step in an assembly sequence is defined as the number of subassemblies that are moving with respect to one another. The hand is considered a fixture.
- A monotone assembly sequence is one in which each subassembly, once constructed, is final. For example, a sequence to insert a piece with a latch is not always monotone.
- A stack assembly is one in which all the assembling motions occur in a single direction, usually up-and-down. An example of this assembly type is the Printed Circuit Board.

3.1.1. State-of-the-art of assembly planners

Assembly sequences search is a NP-complete computational problem (Kavraki, 1993), that is the time needed to determine the optimal assembly sequence grows exponentially with the number of parts. Due to this complexity most of the existing planners are interactive systems, which generate two-handed monotone assembly sequences in reverse, starting from the more highly constrained, until reaching full assembly state. Some of the most known assembly planners are:

- DFA 8.0 and DFA/Pro. This is a very useful advisory system, widely used in the industry.
- Archimedes 4.0. It is basically an ACIS enabled automatic planner composed of a search module, a geometric reasoning module, a tool module for tool accessibility analysis and an interactive user interface with a library of restrictions, that allows assembly planning in a very pragmatic way (Jones, 1998, Halperin, 2000).

3.1.2. AUTMOD3 assembly sequence planning tool

In the FutureHome project, pre-fabricated modules are vertically assembled using cone shaped connectors (see Fig. 23). This system allows the auto-centering of the modules during the insertion, as well as a bigger tolerance in the precision of the crane movements. The assembly by means of cone connectors can be considered a stack assembly. In a construction work, assembly is always performed on a structural basis. In this way, the modules of bigger “structurality”, i.e. modules that support other modules, are inserted first in the assembly sequence. This ordering is performed on levels, and inside the same level, it is performed on coordinates.

Each component has associated several attributes: an insertion point or reference point, the level in the assembly and its “structurality”. The components are joined to each other and through what surfaces they are joined are specified during the design stage. This allows to create an adjacency graph that describe the assembly. The assembly in each level is realised assembling first the more structural components in parallel lines. The insertion points of the components are sorted by coordinates. This in conjunction with the adjacency graph allows to obtain assemblies like those in Figure 17-a, and avoid those like in Figure 17-b.

![Fig. 17. a) Easy to assemble configuration. b) Hard to assemble configuration](image)

Though in a first approach the assembly of the modules can be considered a stackable assembly, it is convenient to modify the vertical assembly vector and convert it to a diagonal vector in order to minimise the contact of the components during the assembly. To achieve this a 15 cm and 45° oriented assembly vector originated in the insertion point is produced. The horizontal component of this vector is obtained from the normal of the surfaces of the adjacent assembled modules (see Figure 17).

The assembly sequence is obtained sorting components by level, “structurality”, and co-ordinates of the insertion point and using the adjacency graph in order to avoid difficult assembly configurations. In addition, a special attribute (grouping) is provided to take into account custom planning criteria. This attribute has the biggest priority allowing the division of a given work in smaller tasks. The sorting time for a building containing 70 modules using this method is instantaneous. The tool is also provided with manual sorting to face unexpected events during the execution process.
3.2. Motion planning

Motion planning defines the spatial trajectories of the modules during the assembly and the automated device or robot commands needed for their execution. Motion planning is defined as follows: given the initial and the final configuration of the module, find a joining path that avoids collision with the obstacles. Motion planning problems are modeled in two spaces: workspace and C-space. Workspace is the geometrical 2D or 3D space in which a robot works. To freely orient and translate a module in a plane (2D space) three degrees of freedom (DOF) are needed, while to do the same in a 3D space, six DOF are required. Workspace can be transformed into the C-space or configuration space. This mathematical transformation poses motion planning problem as a search in a new space in which each dimension corresponds to a one DOF of the robot and the robot position corresponds to a single point in the C-space.

There are many techniques and algorithms for motion planning, but they are generally classified in three groups: roadmaps, cells decomposition and potential fields (Latombe, 1991). Roadmaps techniques produce a network of trajectories that are used to connect initial and final configurations. The visibility graph, the tangent graph, the Voronoi diagram and the freeway net are examples of these techniques. Cell decomposition techniques are divided into exact and approximated cell decomposition. Exact cell decomposition divides the space free of obstacles in small regions. The union of these regions or cells exactly covers the free space. The adjacency relation between cells is represented by means of an undirected graph, in which a path is searched to join initial and final configurations. Approximate cell decomposition covers the free space with a set of uniform regions, but the union of these cells is equal or bigger than free space. That is, these cells can be empty or mixed depending if they cover only free space or a portion of free space and a portion of an obstacle. The use of approximated cell decomposition divides the space free of obstacles in small regions. The union of these regions or cells exactly covers the free space. The adjacency relation between cells is represented by means of a collision detection engine. Potential fields is a technique which assigns a repulsive artificial potential to each obstacle and an attractive artificial potential to the final configuration. Potential fields are more efficient than other techniques, but they have the drawback that search can be trapped in local minima.

3.2.1. State-of-the-art of motion planners

Though motion planning has been solved exactly (Schwart, 1983, Canny, 1988), the obtained algorithms are slow and complex. The complexity of the motion planning problem grows exponentially with the number of DOF of the robot and in a polynomial way with the number of obstacles. Nevertheless, there are many problems in practice that are not really complex and that can be solved exactly or approximately. For robots with four or less DOF of freedom it is possible to solve motion planning problem, using roadmaps or approximated cells techniques in C-space, within a few seconds or a few minutes, depending upon problem complexity (Gupta, 1998). Motion planning for robots with more than four DOF is an active research area. Some of the outstanding motion planners in this area are: the Probabilistic Roadmap Planner (PRM) – this planner generates a roadmap based on random, but properly selected, collision-free configuration nodes, and uses this roadmap to find paths (Hsu, 1999, Kavraki, 1998), SANDROS – a hierarchical, multi-resolution dynamic-graph search planner (Chen, 1998) and AMROSE—a multi-agents, artificial potential based system applied to off-line high level programming of welding applications (Overgaard, 1998).

3.2.2. AUTMOD3 motion planning tool

In order to reduce the motion planning complexity, AUTMOD3 motion planner basically works on two planes. These planes are called assembly plane and free plane (see Fig. 18). The assembly plane is defined at 0,5 m higher than the current assembled level. Free plane is defined at 0,5 m height with respect to the next (higher) level than the currently being assembled. Each component is provided with a function to generate its projection in the assembly plane, in a such way that the component does not collide during its translation in this plane. The simplest way to generate this projection is to slightly increase the bounding box of an ACIS solid. The sequence of the algorithm is as follows:

1. Create an assembly plane for each level.
2. If the current component is the first to assemble, proceed to translate it in the assembly plane. Otherwise go to step 4.
3. The projection of the component is introduced in the assembly plane.
4. Assembly plane projections of the assembled components are increased taking into account the projection of the component under assembly.
5. Tangent Graph is calculated and searched for a suitable trajectory (Liu, 1994).
6. If there are more components to assemble go to step 1.

Fig. 18. Working planes of the AUTMOD3 motion planning tool

Execution of the planner is realised in an independent thread, whose execution time is supervised by the main thread. If this time reaches a user predefined limit, the planning thread is suspended and a new path is generated using the free plane. The same occurs if the
path does not exist or its distance is very large. In addition, the tool is provided with a manual planning function allowing the modification of the obtained trajectories or the generation of other types of paths. This method trades-off optimality versus complexity producing a simple algorithm. Grasping operations are automatically performed by means of the electromagnets and the artificial vision system installed in the crane platform, while insertion operations are also automatically realised using the conic connectors and the crane platform artificial vision system (see Figs. 22 and 23).

Fig. 19 shows the “View” dialogue box which allows to visualise the assembly following the obtained sequence. This can be done step-by-step, using a cursor or automatically.

Fig. 20 shows the exporting the robotized crane commands to a file and its visualisation. These commands are written in the robotized crane programming language.

4. Application to the futurehome robotized crane

Once the assembly sequence and the robotized crane commands have been obtained, the assembly can be simulated in the AUTMOD3 simulation tool (see Fig. 21) or the commands can be sent to the robotized crane for its execution.

The prototype robotized crane developed at Carlos III University of Madrid, as part of the FutureHome project (Abderrahim 2001), employed a commercial gantry crane that has been converted into an automated device. For this purpose several modifications have been performed:

1. Each of the crane motor has been replaced by a servomotor with a resolver in order to control its movement from a computer.
2. An anti-swing system has been implemented in order to reduce the oscillation of the modules during the movement.
3. The crane hook has been replaced by a special platform (see Fig. 22) provided with electromagnets to grasp the modules and an artificial vision system for correct positioning during grasping and insertion operations.

This crane is used to assemble modules in a demonstrator with 1:3 scale modules. These modules are assembled using cone shaped connectors as shown in Fig. 23. The AUTMOD3 software environment jointly with the robotized crane control system permit the automatic modular construction of the designed building in the demonstrator.
5. Conclusions

This paper introduces the automatic modular construction software environment AUTMOD3 and shows how the system seamlessly integrates architectural design, planning and simulation tools in a well-known CAD program. The paper shows the modular design methodologies employed in the environment and how the AUTMOD3 tools jointly with the robotized crane and its control system permits the automatic modular construction of the designed building.

6. Acknowledgement

This work has been supported by the EU project FutureHome under BRITe program with reference no. ER4-29671 and was done in close collaboration with the IF7 Japanese program in the frame of the IMS (Intelligent Manufacturing Systems) trans-regional program.

7. References

Abderrahim, M. Giménez, A. Nombela, A. Garrido, S. Diez, R. Padrón, V. M. and Balaguer, C. (2001), The Design and Development of an Automatic Construction Crane. 18th International Symposium on Automation and Robotics in Construction (ISARC’2001), Krakow (Poland), 149–154.

Adeli, H. (2000), Emerging Information and Computing Technologies for Construction Automation, 17th International Symposium on Automation and Robotics in Construction (ISARC’2000), Taipei (Taiwan), K29-K32.

Aouad, G., Marir, F., Child, T., Brandon, P. and Kawaoya, A. (1997), Construction Integrated Databases – Linking Design, Planning and Estimating: the OSCON Approach, International Conference on the Rehabilitation and Development of Civil Engineering Infrastructures, Beirut (Lebanon), 51-60.

Atkin, B. (2000), Housing demand and need in Europe and World Export Potential, FutureHome Internal Report WP1-T1-D2.

Atkin, B. and Wing, R. (1999), FutureHome Manufactured Housing for Europe, 16th International Symposium on Automation and Robotics in Construction (ISARC’1999), Madrid (Spain) 573-578.

Balaguer, C., Abderrahim, M, Navarro, J.M., Boudjaber, S., Aromma, P., Kahkonen, K., Slavenburg, S., Seward, D., Wing, R., Atkin, B. (2002), FutureHome: An Integrated Construction Automation Approach, IEEE Robotics and Automation magazine, 9(1), 55-66.

Boothroyd, G. and Alting, L. (1992), Design for Assembly and Disassembly, Annals CIRP, 41(2), 625-636.

Canny, J. (1988), The complexity of Robot Motion planning, MIT press.

Chen, P.C. and Hwang, Y. K. (1998), SANDROS: A Dynamic Graph Search Algorithm for motion planning, IEEE Transactions on Robotics and Automation, 14(3), 390-403.

Diez, R. Abderrahim, M. Padrón, V. M. Celorrio, L. Pastor, J. M, Balaguer C. (2000). AUTMOD3: an Automatic 3D Modularization System. 17th International Symposium on Automation and Robotics in Construction (ISARC’2000),Taipei (Taiwan), 1033–1038.

Diez, R. (2003), Automatización del Diseño de Edificios Modulares para su Construcción Robotizada (in Spanish). PhD Thesis, Carlos III University of Madrid.

Feldman, K. and Koch M. (2000), Chances, Obstacles and a Possible Trend of Automation in Construction from the Point of View of the German Construction Industry, 17th International Symposium on Automation and Robotics in Construction, (ISARC’2000), Taipei (Taiwan), 965-970.

Furuya N., Shiokawa, T., Hamada, K., Kurita, K., Kunimoto, I. and Miyakawa H. (2000), Present Circumstances of an Automated Construction System for High-Rise Reinforced Concrete Building, 17th International Symposium on Automation and Robotics in Construction (ISARC’2000), Taipei (Taiwan), 953-958.
Halperin, D., Latombe, J. C. and Wilson R.H. (2000), A General Framework for Assembly Planning: The Motion Space Approach. Algorithmica, 26(3-4), 577-601.

Hiroo, T. and Yagi, J. (2000), Study on Assembly Methods for Large-Scale structure Using an Innovative and Intelligent Field Factory, 17th International Symposium on Automation and Robotics in Construction (ISARC’2000) Taipei (Taiwan), 983-989.

Hsu D., Kavraki, L., Latombe, J.-C., and Motwani, R. (1999), Capturing the Connectivity of High-Dimensional Geometric Spaces by Parallelizable Random Sampling Techniques, Advances in Randomized Parallel Computing, in Pardalos, P.M. and Rajasekaran, S. Eds., Combinatorial Optimization Series, Kluwer Academic Publishers, Boston, MA, 159-182.

Gupta, K. (1998), Overview and State of the Art, in Gupta, K. and del Pobil, P., Eds., Practical Motion Planning in Robotics: Current Approaches and Future Directions, John Wiley, 1-8.

Jones, R.E., Wilson, R.H. and. Calton, T.L (1998), On Constraints in Assembly Planning, IEEE Transactions on Robotics and Automation, 14(6), 849-863.

Kavraki, L., Latombe, J-C. and Wilson R.H. (1993), On the Complexity of Assembly Partitioning, Information Processing Letters, 48(5), 229-235.

Kavraki, L., Latombe, J.-C., Motwani, R., and Raghavan, P. (1998), Randomized Query Processing in Robot Path Planning, Journal of Computer and System Sciences, 57(1), 50-60.

Latombe, J. C. (1991), Robot Motion Planning, Kluwer Academic Publishers.

Liu, Y-H. and Arimoto S. (1994), Computation of the Tangent Graph of Polygonal Obstacles by Moving-Line Processing, IEEE Transactions on Robotics and Automation, 10(6), 823-830.

Nevins, J. L. and Whitney, D.E. (1989), Concurrent Design of Products and Process, McGraw-Hill.

Overgaard, L., Larsen, R. and Jacobsen, N. (1998), Industrial Applications of the AMROSE Motion Planner, in Gupta, K. and del Pobil, P., Eds., Practical Motion Planning in Robotics: Current Approaches and Future Directions, John Wiley, 155-183.

Padrón, V.M., Cárdenas, O., Diez, R., Abderrahim, M. and Balaguer, C. (2002), AUTMOD3: A Planning Tool for Modular Building System. 19th International Symposium on Automation and Robotics in Construction (ISARC’02), Washington (USA), 91–96.

Pastor, J.M., Balaguer, C., Rodriguez, F.J., Diez, R. (2001), Computer-aided Architectural Design Oriented to Robotized Façade Panel Manufacturing, Computer-Aided Civil and Infrastructure Engineering, 16 (2), 216-227.

Perin, L.F., Balaguer, C., Pastor, J.M., Rodriguez, F.J. and Barrientos, A. (1998), Robotized Spraying of Prefabricated Panels, IEEE Robotics and Automation Magazine, 5(3), 18-29.

Ransen, O. (1997), AutoCAD programming in C/C++, John Wiley & Sons.

Romney, B., Godard, C., Goldwasser, M. and Ramkumar, G. (1995), An efficient system for geometric assembly sequence generation and evaluation, Proceeding of the 1995 ASME International Computers in Engineering Conference, 699-712.

Schwartz, J. T. and Sharir, M. (1983), On the ‘Piano Mover’ Problem: II. General Techniques for Computing Topological Properties of Real Algebraic Manifolds, Advances in Applied Mathematics, 4, 298-351.

Wolter, J.D. (1988), On the automatic generation of plans for mechanical assembly, PhD Thesis, The University of Michigan.