Computers in Industry 64 (2013) 1282–1289

Contents lists available at SciVerse ScienceDirect

Computers in Industry

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

Some aspects of dynamic 3D representation and control of industrial processes via the Internet

Sándor Kopácsi *, George L. Kovács, János Nacsa

Institute for Computer Science and Control, Hungarian Academy of Sciences, MTA SZTAKI, Kende u. 13-17, H-1111 Budapest, Hungary

ARTICLE INFO

Article history:
Received 1 September 2012
Received in revised form 11 March 2013
Accepted 5 June 2013
Available online 30 July 2013

Keywords:
3D modeling
3D Internet
Virtual reality
Industrial processes
Incremental Sheet Forming
Rubik's Cube

ABSTRACT

In our research we examine and use 3D representation of industrial processes, for example the novel methods of Incremental Sheet Forming. We also test 3D imaging methods on our industrial robot solving the Rubik's Cube. We have created 3D models of our robots and their environment in our laboratory to examine the behavior of different industrial processes both in the real, and in the 3D virtual environment. We have connected the 3D model with the real system with which we could extend the features of our robots with some services that exits in the virtual space. We have also established synchronized connections of the real and virtual systems, which enables us to control the real robots and machines from its 3D model via the Internet.

1. Introduction

The importance of modeling and simulation is rather high in most industrial processes. Simulation and animation can provide not only numerical, but visual information about the behavior and performance of existing systems, with which experiments can be expensive or dangerous (e.g., nuclear power plants), and about systems which do not yet exist (just under planning). Traditional 2D animation systems cannot provide the accurate and detailed information that customers expect today, therefore 3D simulation and animation methods in virtual and augmented reality systems [1,2] became more and more popular, besides regular simulation.

Due to the rapid development of the graphical modeling tools, accurate modeling of complex systems became feasible. More advanced models require more sophisticated three dimensional (3D) rendering, that can supply the most realistic feeling in virtual environment. As a consequence of the advanced 3D technology considering both input and output devices, there is an increasing need for creating 3D models.

Powerful PCs, graphic cards, and software tools make spatial representation of 3D models possible, as well as taking the process time of the model into consideration, the operation of dynamic 3D models is available, too. Using the information highway for accessing these three dimensional services, state of the art applications can be created on the Internet, too.

Utilizing the above mentioned trends we are going to create three-dimensional dynamic representation of some industrial processes in a virtual reality environment, where the real and visualized components have direct and synchronized connection with each other, in order to extend the features of the real components with the ability and services of the virtual elements. In this way we can connect virtual elements with their counterpart in the real system, as well as we can add new features to the real components that exist only in the cyberspace (like control with special methods).

In this sense the problem that we are going to solve in this research concerns on one hand three-dimensional modeling, and on the other hand concerns distributed collaborative systems. We do not only want to create three-dimensional models, and place them in the cyberspace, but we want to operate the virtual elements in the collaborative environment synchronized with the real system.

2. Some studies on virtual 3D environments and collaborative systems

One of the most wide spread state of the art 3D simulation software is *RoboLogix* [3], that can be used for teaching and testing the functions of the modeled robot. The robot and the surrounding environment can be modeled together. Running the RoboLogix program we can examine the robot, we can walk around it, and we can test different control algorithms. The simulation program takes

* Corresponding author. Tel.: +36 1 2796268.
E-mail addresses: kopacsi.sandor@sztaki.mta.hu (S. Kopácsi), kovacs.gyorgy@sztaki.mta.hu (G.L. Kovács), nacsa.janos@sztaki.mta.hu (J. Nacsa).

0166-3615/$ – see front matter © 2013 Elsevier B.V. All rights reserved.
http://dx.doi.org/10.1016/j.compind.2013.06.007

the working zone of the robot into account to check and avoid collisions. RoboLogix is effective in testing and teaching, but there is no physical connection between the real and the modeled system. Therefore the control of the real system via the model is not possible.

Another interesting effort is the *Simbad Project* [4], that is a 3D robot simulator for scientific and educational purposes. It provides 3D visualization, single- or multi-robot simulation, and many different built-in sensors (like vision, range, contact) are available, too. Simbad enables programmers to design the robot and its environment, to write the robot controller, and to use the available sensors. In Simbad controlling the real system from its model is not possible.

Several other systems are devoted to 3D visualization by loosely or fundamentally integrating 3D visualization components, as e.g., *RT-Middleware* [5], and *ROS* (*Robot Operating System*) [6].

The above discussed tools can be used for visualization of sensual and environmental information, kinematic, dynamic and sensor simulation. However none of them utilizes the possibilities of the virtual 3D space in an extensive manner. If we want to use more features (like cooperation of the components) of the virtual three-dimensional space, we have to take into account the classical collaborative environments, too.

One of the most popular collaborative environments is *Open Cobalt* [7], which is a multi-platform fully customizable advanced simulation system. The user can setup virtual workspaces for custom animation, where the models can be created by open source authoring tools (e.g., Google SketchUp, Blender) and can be shared with other users. The applied three-dimensional multimedia wiki technology makes it easy to create deeply collaborative and hyperlinked multi-user virtual workspaces, where the users can interact with others using voice or text chat. Cobalt supports collaborative data visualization, virtual learning and problem solving environments.

Open Wonderland [8] is another collaborative environment with which we can create interactive, multi-user simulation system, or collaborative business applications. It is an open source Java toolkit for developing collaborative three-dimensional virtual worlds. In the virtual environments users can communicate with each other using immersive audio, or shared desktop applications. Open Wonderland is fully extensible, developers and graphic designers can extend its functionality to create entirely new worlds and add new features to existing worlds. Virtual content can be created with open source graphical tools.

The *Virtual Collaboration Arena* (*VirCA*) [9], developed by the Cognitive Informatics Group of our Institute is a component based, distributed, interactive virtual reality manager system for connecting virtual and real environments, displaying and manipulating their devices, and realizing distributed, interactive collaboration. In this virtual reality environment, there are representations of virtual and real devices, as well as static and dynamic virtual objects, to which simulation and animation can be applied. The single components of VirCA communicate according to the RT-Middleware [5] standard through typified ports with each other. Devices and services in VirCA that are far from each other can be connected through the Internet, hereby the costs and time of the development of the given application can be reduced. That way, VirCA provides the realization of knowledge plug'n'play, the boost of the connection between different researches and developments. The user of VirCA can observe and manipulate the different, usually distant virtual and real physical devices together. With the help of the virtual objects, the cooperation of such real devices far from each other can be tested without development of a simulator, the transport or purchase of which would be complicated and expensive otherwise, so VirCA is an efficient test environment, too. VirCA is suitable not only to model and control but also to monitor the different industrial processes from different places in 3D space. The first applications of VirCA proved its capabilities (see [10]).

Since our goal is not just to model and visualize our robots in a 3D virtual reality environment, but to synchronize the model with the real, operating robot, as well as control real robotic systems from the virtual world of their model, even by means of remote control via the Internet, the VirCA system seems to be the best selection among the studied 3D visualization and collaborative environments. This way our 3D models become dynamic and alive in the cyberspace, and we can synchronize the real manufacturing activity with the virtual processes for test, training and remote control purposes. None of the above discussed environments, except VirCA can provide all the functions we need.

3. Experiments with real processes

We have selected some industrial processes with which we can make experiments in a virtual world and in a real environment. In our first tests we have examined the different parameters (e.g., size and form of the tool, depth, speed) of Incremental Sheet Forming (see below). The accuracy of the model could be tested by the parallel and synchronized operation of the virtual and the real environment.

Our second test bed was the solving of the Rubik's Cube by our industrial robot. It is not a typical industrial process, but the methods of synchronized control are rather similar, with which we could test our system, especially the precise movements of the robot.

3.1. Incremental Sheet Forming (ISF)

Recent rapid prototyping techniques, as Stereo Lithography, Laminated Object Manufacturing, Selective Laser Sintering, Fused Deposition Modeling, Solid Ground Curing and 3-D Ink-Jet Printing can hardly be used to produce metal or plastic sheet parts.

Incremental Sheet Forming (ISF) [11] is a 3D shaping process of sheet metals based on a series of small localized deformations, created by a forming tool with a spherical head which is mounted on a CNC milling machine or on an industrial robot. In the recent years the process is tested also on polymer sheets and sandwich structures. In all cases the starting point of the process is the 3D CAD model from which the tool path calculation (post processing) is done by a CAM program.

ISF is a relatively cheap and fast procedure to make one-o-f-a-kind production (OKP), or rapid prototyping. These applications would be very expensive with the traditional pressing die-sheet-die structure, as die production requires a lot of time, energy and money. On the other hand fast sheet production or OKP is often necessary. The automotive, aeronautic and space industry [12] are interested in this new sheet forming technique, and experimental applications in medical aid manufacturing are also promising [13].

Dealing with ISF we have tested two different configurations. In the first configuration we used a small robot and a small milling machine for ISF, while in the second configuration we used a real industrial robot. Another interesting application was when our industrial robot solved the Rubik's Cube.

3.1.1. ISF experiments with a small robot and milling machine

We have set up an experimental two-machine-based environment for testing the principles of Single Point Incremental Forming (see Fig. 1), where a hot air gun fixed on a small (Mitsubishi RM 501) robot can synchronously be moved along with the incremental forming tool mounted on a 2.5D Rieckhoff milling machine [14]. Heat is applied on the opposite side of the plastic sheet.

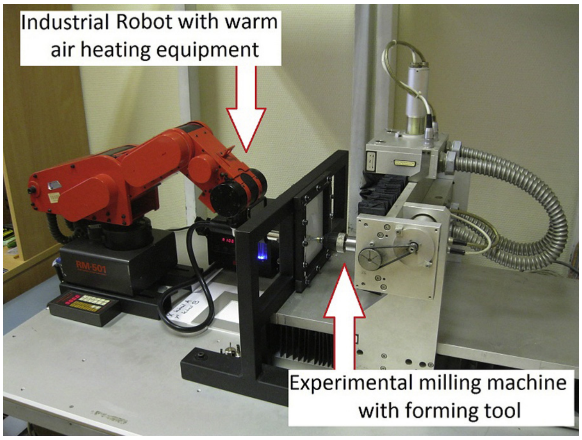

Fig. 1. Set-up of Single Point Incremental Forming (see movie in Multimedia Component 1).

As both the robot and the milling machine used in this experiment are more than 10 years old, their controllers were changed recently. They have the same LinuxCNC, which is a free and powerful machine controller based on the NIST Enhanced Machine Controller software [15]. In both controllers the default LinuxCNC configuration (liveCD) was used together with their Ubuntu Linux based operating system and the real time Linux patch called RTAI [16].

3.1.2. Incremental Sheet Forming with an industrial robot

In the recently established robot laboratory in MTA SZTAKI [17] an industrial robot (Fanuc S-430iF) has been set up, which has 6 degrees of freedom, it is able to lift up 130 kg, and has a ~2.5 m horizontal reach. With this industrial robot we have set and proved some of the process parameters (forming speed: 100 mm/s, tool diameter: 10 mm, incremental depth with Z-level tool path: 1 mm) of the ISF (see Fig. 2). We allowed a maximum forming force of ~700 N with a JR3 force-torque sensor (type 45E15A-U760-D) mounted between the robot and the tool holder by forming a half hard 0.5 mm thick AA 1070 Aluminum sheet with a size of 500 mm × 500 mm. The desired geometry was a truncated-pyramid with wall angle 45° in the first experiments, now we can produce more complex objects due to software purchase and development.

3.2. Rubik's Cube

3.2.1. Theoretical background

Rubik's Cube is a three dimensional mechanical puzzle game, where each six faces of the cube is divided by nine squares (3 × 3)

Fig. 2. ISF with Fanuc robot (see 2D movie in Multimedia Component 2 or side by side 3D movie in Multimedia Component 3).

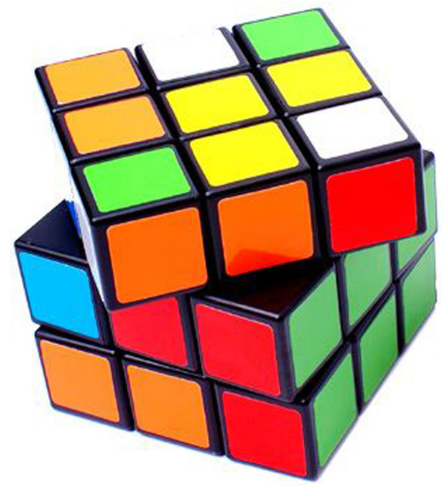

Fig. 3. Rubik's Cube.

and covered by six different colors – one color per side (see Fig. 3). These cubes are fixed to each other in a way, that one or two layers of cubes can be turned around a central spindle by 90, 180 or 270 degrees. There are 8 cubes (on the corners) with 3-3 colored surfaces, 12 have 2 colors each, there is one little cube in the very middle with no visible surfaces, and the rest 6 (on the middle of the 6 surfaces) have one-one different colored surface. A special pivot mechanism makes possible to turn the sides of the cube independently.

The start position of the game is a mixture of colors on the sides, while in the goal position each side must consist of one of the six colors. Solving the cube needs 3D vision and comprehension not only on the surface of the cube, but in the inside of it, as well.

Theoretically we can place the first corner of the cube to 8 places in 3 positions, the second corner to 7 places in 3 positions, etc. That is $8!*3^8$. Considering the edges, the first edge of the cube can be placed to 12 different points in 2 positions, the second one into 11 places in 2 positions, etc. That is $12!*2^{12}$. If we take into account only those combinations, from which the cube can be solved, then the last corner should not be turned around (divide by 3), and the last two edges should be sited in certain place and certain position (divide by 2*2). This way the total number of possible patterns from which the cube can be solved is $8!*3^7*12!*2^{10}$ = 43 252 003 274 489 856 000 ~ $4.3*10^{19}$ [18].

3.2.2. Solving the Rubik's Cube with industrial robot

Solving the Rubik's Cube with our industrial robot was a great challenge, in spite of the fact that there are several similar applications. Most of the Rubik's Cube solver robots are dedicated to this special task, with special gripping and turning methods (see e.g., [19]) or on the other hand there are general, but even very efficient applications, like the world's fastest cube solver using Lego's Mindstorm NXT powered by a Samsung Galaxy S II (see [20]). There are existing solutions with industrial robots, too, see e.g., [21] or [22].

We wanted to test the precise movements, acceleration, retardation, change of direction of our robot, so for these purposes solving the Rubik's Cube seemed to be an ideal application. The solving algorithm was developed by the author of [19], while the mechanical extension and actual moves of the robot were developed in our laboratory. Our industrial robot can solve the Rubik's Cube from any start position in maximum 21 steps (see Fig. 4).

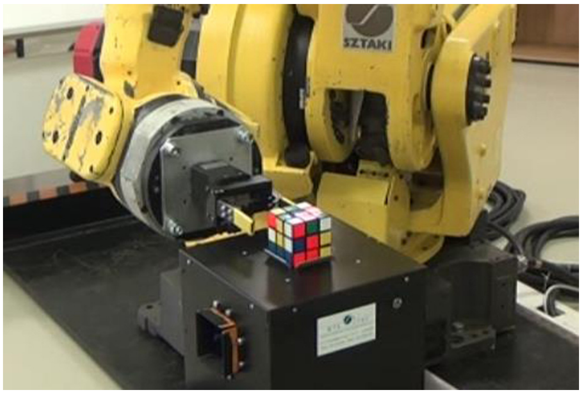

Fig. 4. Solving the Rubik's Cube (see 2D movie in Multimedia Component 4 or side by side 3D movie in Multimedia Component 5).

4. 3D modeling of industrial processes in the cyber space

4.1. 3D representation

3D rendering is available today not only for special devices, but for TV sets, for desktop and notebook computers, too. There are several methods for 3D representation, let us have a look at the most wide-spread techniques, especially at those that we used in our research.

Most of the techniques are based on producing two offset images, one for the left and one for the right eye. The two images represent two perspectives of the same view with the deviation as the two eyes can see it. The viewer technology of these images can be basically passive or active. The cheapest passive technology is the so called anaglyph method [23], where special glasses – in most of the cases with red and cyan filters in front of the left and right eyes respectively should be used, and the color modulated content will arrive dedicated to the left or right eye, which results 3D image in the brain. It does not require special monitor, or active devices, just very cheap, usually paper framed glasses. Viewing side-by-side images on active or passive monitors [24] requires special software that puts the images together and either filters the image to the left and right eye, or controls the active 3D glasses that separate the images and causing 3D feeling.

Animated GIF image can also produce 3D-like feeling, where the image to the left eye, and to the right eye alters each other in a certain speed. The human brain can put the image together and see it as a slightly vibrating 3D object. The main advantage of this method is that it does not require any special equipment for displaying the 3D image. In the rendering of our 3D models we use both the red-cyan and active stereoscopic technique supported by VirCA while in our 3D webpages (http://3dweb.hu) [25], where we have published some results of this research and development, all the above mentioned methods of 3D rendering are applied.

4.2. Elements of the 3D environment

The 3D elements (see for example Figs. 5 and 6) and their environment in our models have been created and finalized in Google SketchUp 8 [26]. Some of the original 3D visual components have been created in other graphic modeler programs (e.g., SolidWorks, Solid Edge) using different file formats (e.g., IGES) that have been imported to Google SketchUp. In some cases Autodesk 3ds Max has been used for transforming original files to 3DS format that has been imported to Google SketchUp, too.

Since the graphic engine of the applied collaborative environment (VirCA) is Ogre [27], therefore the visual components of the models have been supplied in Ogre (mesh and material files) format. The mesh and material files have been exported from Google SketchUp, using "SketchUp to Ogre Exporter" plug-in.

For all the components an individual cyber device object has been created keeping their original scale, as they were exported from Google SketchUp. In most of the cases the original orientations of the components were also kept, but in the case of the robot, the orientation of its components has been set up to the required angles.

5. Synchronized operation of the real system and its 3D model

In order to establish the synchronized operation of the real and virtual system we had to set up an information system that consists of three different elements:

- the 3D virtual collaborative environment implemented in VirCA,
- so-called cyber devices in the virtual environment that are representing the real objects,
- and those real objects that have IT parts in the real world (like controller of the robot).

As the VirCA environment uses the RT-Middleware [5] to communicate with the external real devices, a special Network Interface Unit (NIU) has to be developed to communicate with the machine controller.

The controller sends its status and – when it is moving regularly – the actual joint coordinates and program line number to the

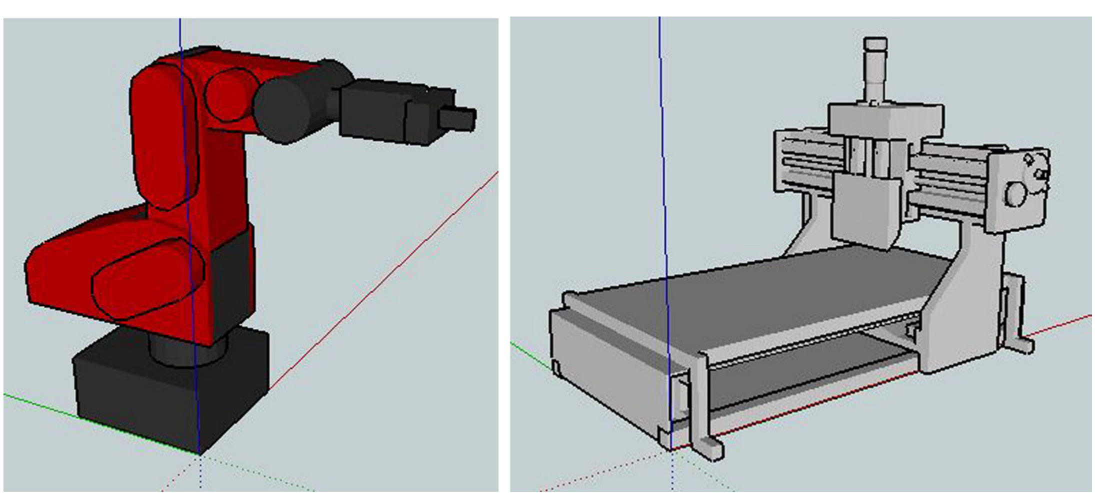

Fig. 5. 3D Model of the Mitsubishi Robot and Rieckhoff milling machine (see 3D files in Multimedia Component 10).

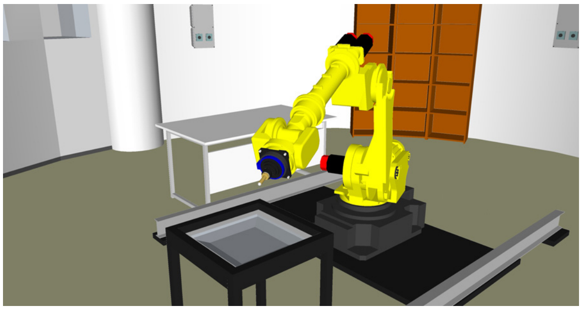

Fig. 6. Model of the Fanuc Robot and its environment (see 3D files in Multimedia Component 11).

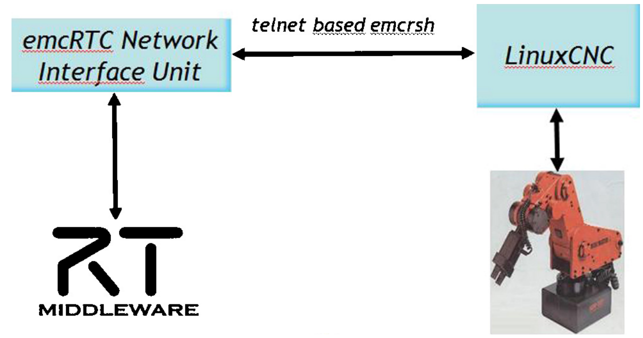

Fig. 7. Connecting LinuxCNC to the VirCA through emcRTC NIU.

virtual world, while it gets commands (start, stop, program selection, reset etc.) from the cyberspace.

In the two-machine-based environment (see chapter 3.1.1) both machines have LinuxCNC controllers. The simplest solution was chosen that ensured that no controller software modification was needed while it has some rather complex, but poorly documented key elements. The LinuxCNC can be configured to run a telnet-based server that allows external clients to remote control the machine via a so-called *emcrsh* protocol. So the emcRTC NIU has an RTM connection toward the VirCA world, and an emcrsh connection to reach the controller (Fig. 7). Via this protocol a wide range cooperation could be defined between the controller and the cyber device. The solution is general so the same NIU is used for the robot and for the milling machine.

Fig. 8 shows the structure of the system of the two-machine-based ISF application, since the industrial solution is rather similar. In this structure each box represents a separate program. They are connected to each other based on the Corba naming service that manages them. The bottom box represents the VirCA system that has connections with four different cyber devices (two manufacturing machines, the actual workpiece, and the process itself) using Internet Communications Engine (ICE) interface [28]. They send visual data to the VirCA system, and get the commands of the user. The other two boxes without direct VirCA connections represent the network interface units that couple the two physical machine controllers to the system.

When these application specific programs want to exchange data with each other and not with the VirCA system, they are using

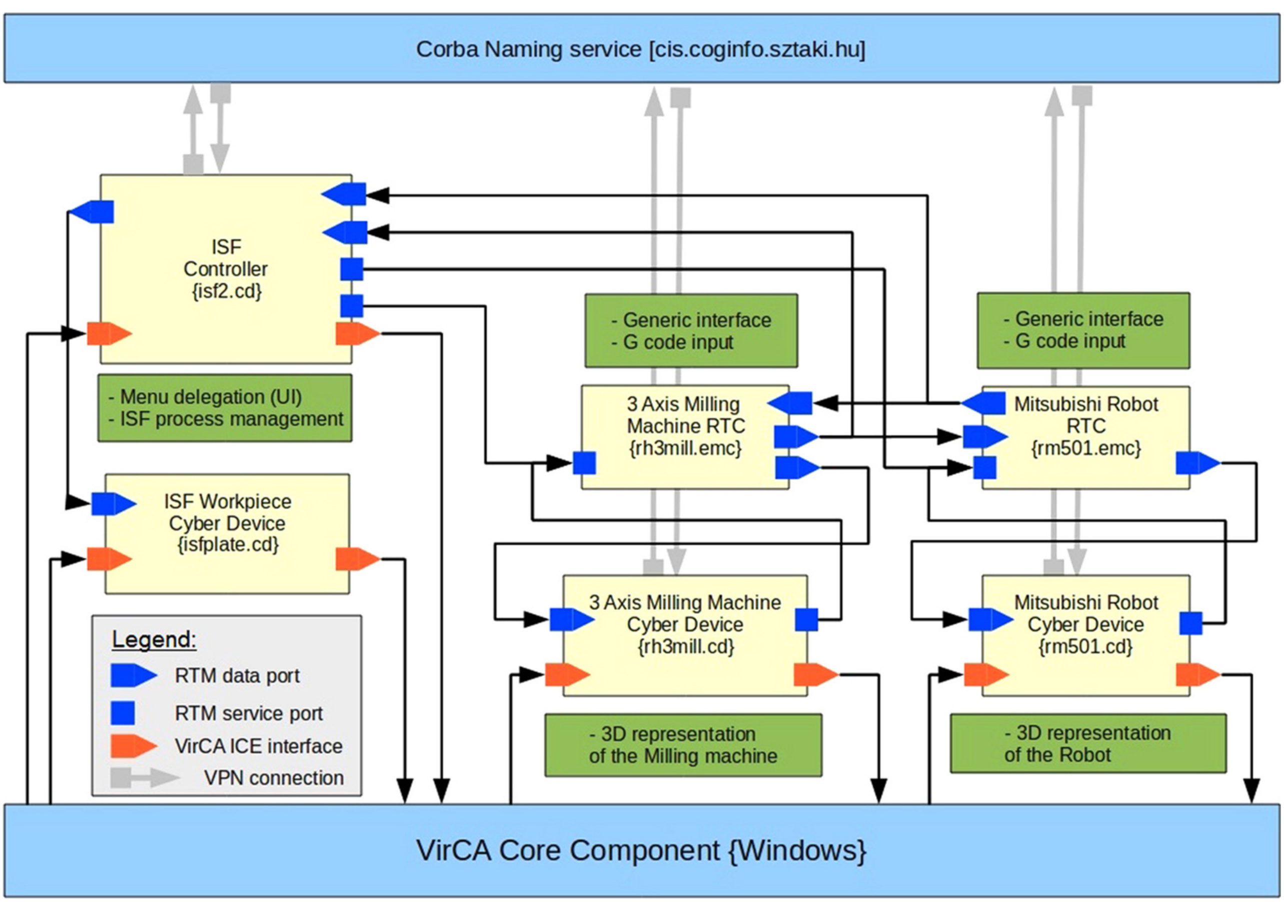

Fig. 8. System architecture.

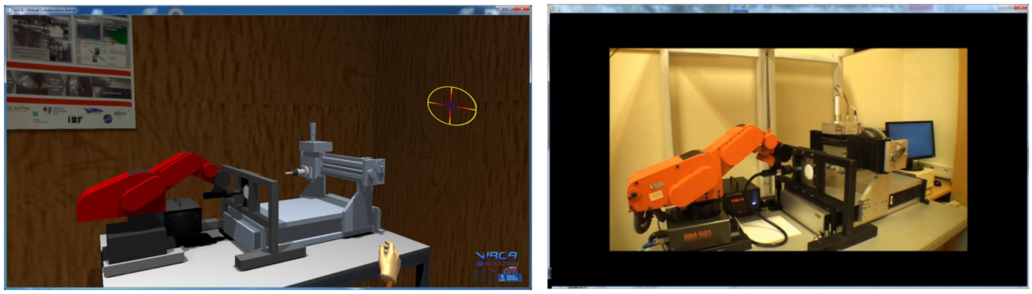

Fig. 9. The operation of the model synchronized with the real system (see full video in Multimedia Component 6).

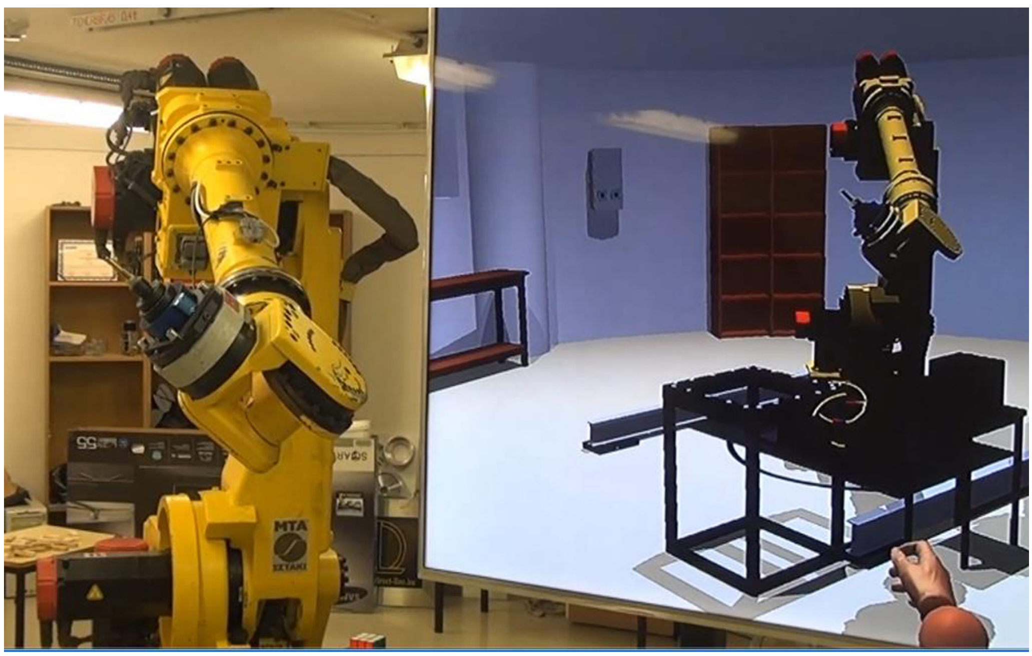

Fig. 10. Synchronized operation of the Fanuc robot and its model (see video in Multimedia Component 7).

different ports of the RT-Middleware communication. The nature of the communication assigns which data and service ports are used for a certain information sharing.

The second ISF application (see chapter 3.1.1) using a Fanuc robot has been realized with the same structure by leaving out the RTC (RT-Middleware Component) and the Cyber Device of the milling machine.

A special web based tool [29] supports the connection of the different programs with the proper interfaces, and starts (activates) the whole system via the Internet. It controls the states and connections of all the affected components, which might be available even from different parts of the world. There is a living connection between the real machine controllers and their cyber devices.

The VirCA environment allows us to be in a virtual space, and interact with it knowing that all the actions will be also done in the real environment. This way several parameters, as well as the model of the industrial process can be examined both in the virtual and in the real system (Fig. 9).

We had to develop another NIU to connect the Fanuc robot to the VirCA using the serial port of the robot to exchange data. A Karel-based program [30] is running in the robot controller that is able to get motion commands from the NIU, to execute these commands and to send back continuously the actual coordinates of the joints (see Fig. 10).

After the successful tests of the synchronized operation of the model and real Fanuc robot, we performed the Incremental Sheet Forming (*see video* in Multimedia Component 8) and Rubik's Cube solving (*see video* in Multimedia Component 9) in parallel in the virtual and in the real system in a synchronized way.

6. Performance evaluation of the proposed method

Evaluating the performance of the proposed system we can state, that the connection and synchronization of the real components were much easier through the virtual cyber space, than it would have been using direct connection of the available real interfaces.

Theoretically (e.g., [31]) there are exact ways to run robots and machine tools in synchronized mode, however, in practice, it is worth avoiding the modification of the controller kernel. So even in cases of two machines it is better to take into consideration the available external communication interfaces of the controllers and find a way to synchronize the motions via them.

The external software connected to both machine controllers can provide the synchronization. It requires no modification inside the controllers. A simple strategy is that none of them can move further till the other does not reach the corresponding position. The strategy is good enough if the movements are sufficiently small. While the external interfaces are used to connect to the virtual environment, it was an obvious option to establish the synchronization through the virtual environment, while these connections have already existed. According to our qualitative estimation, the synchronization within the virtual world is about 5% slower than without it, using dedicated software.

Comparing our approach with other methods described in Section 2 we can state that using the collaborative VirCA environment we could easier and faster add new features (like voice control) to the real system, that existed in the virtual environment only.

7. Conclusions

A challenging manufacturing process (ISF), a 3D puzzle (Rubik's Cube) and the operation of their dynamic 3D models in the virtual collaborative environments of VirCA have been introduced in this paper. In case of both configurations (two-machine-based and industrial-robot-based) of ISF, as well as of the Rubik's Cube VirCA helped not only in the integration of the components, and in adding new features to them, but in their synchronized operation, as well.

We made our robot systems more intelligent by using sensors coming from the game industry (as e.g., Kinect [32] and Wii controller [33]), with which we can control the real system through its virtual counterpart. By controlling the virtual robot in the cyberspace, that is connected and synchronized with the real one, we can control the real robot via the Internet even from a far place. This way tele-presence and tele-operation tasks can be performed, too. The system provides the possibility to extend the user communication in a more effective way than ever before.

In our solution we realized that industrial processes should be managed logically as normal objects. Applying the 3D representation of the abstract process as a certain component of the environment, the user senses it as other objects in the room. In this way the real machines and the abstract manufacturing processes have similar potential actions that the user can apply (e.g., choose part, setup, release, start, stop).

Based on our results we used our 3D visualization and synchronized control experiments on other fields of industry, as well, like for visualization of multifunctional furniture.

The main conclusion is that the connection of real industrial processes and their three-dimensional virtual system provides faster planning, easier system integration, and more reliable operation and control, than the currently known methods. With our system we could establish a very efficient synchronized operation of a virtual and a real system.

Acknowledgments

This research and the establishment of the robot laboratory were supported by MTA SZTAKI. We are grateful to the Cognitive Informatics Laboratory of our institute for the cooperation in applying VirCA.

Students of the Dennis Gabor College (GDF) and of the Technical University of Budapest (TUB) took active part in the research and development. We also appreciate the financial support of the GDF for the establishment of a Virtual 3D Laboratory which has direct connection with the real Robot Laboratory of MTA SZTAKI.

Appendix A. Supplementary data

Supplementary material related to this article can be found, in the online version, at http://dx.doi.org/10.1016/j.compind.2013.06.007.

References

[1] Virtual reality, http://en.wikipedia.org/wiki/Virtual_reality, 2013.
[2] Augmented reality: http://en.wikipedia.org/wiki/Augmented_reality, 2013.
[3] RoboLogix, http://www.robologix.com/, 2011.
[4] Simbad project, http://simbad.sourceforge.net/, 2012.
[5] N. Ando, T. Suehiro, K. Kitagaki, T. Kotoku, W.-K. Yoon, RT-Middleware: distributed component middleware for RT (robot technology), in: 2005 IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2005), Edmonton, Canada, August, (2005), pp. 3555–3560.
[6] Willow Garage, Robot Operating System, 2011 http://www.ros.org.
[7] Open Cobalt, http://www.opencobalt.org/, 2013.
[8] Open Wonderland, http://openwonderland.org/, 2013.
[9] P. Galambos, B. Reskó, P. Baranyi, Introduction of Virtual Collaboration Arena (VirCA), in: The 7th International Conference on Ubiquitous Robots and Ambient Intelligence, Busan, Korea, (2010), pp. 575–576.
[10] Virtual Collaboration Arena, http://www.virca.hu/, 2011.
[11] Kathryn Jackson, Julian Allwood, The mechanics of incremental sheet forming, Journal of Materials Processing Technology (2009) 1158–1174.
[12] M. Rauch, J.Y. Hascoet, J.C. Hamann, Y. Plenel, Tool path programming optimization for incremental sheet forming applications, Computer-Aided Design 41 (12) (2009) 877–885.
[13] J.R. Duflou, B. Lauwers, J. Verbert, F. Gelaude, Y. Tunckol, Medical application of single point incremental forming: cranial plate manufacturing, in: Proceedings of the 2nd International Conference on Advanced Research in Virtual and Rapid Prototyping VRAP, Leiria, (2005), pp. 161–166.
[14] M. Tisza, I. Paniti, P.Z. Kovács, Experimental and numerical study of a milling machine-based dieless incremental sheet forming, International Journal of Material Forming 3 (1) (2010) 441–446.
[15] EMC/Linuxcnc – Enhanced Machine Controller, http://www.linuxcnc.org/, 2011.
[16] RTAI – the Real Time Application Interface for Linux from DIAPM, https://www.rtai.org/.
[17] G. Kovács, I. Paniti, A New Robot Laboratory in SZTAKI, ERCIM News (91) (2012) 33–34.
[18] Rubik Ernő, The Magic Cube, Műszaki Könyvkiadó, 1981.
[19] Boros Miklós, A Robot that can Solve the Rubik's Cube, 2012 http://technik.mindenkilapja.hu/html/18054347/render/robot-vilag.
[20] smartplanet, Robot solves Rubik's Cube in 5s, 2012 http://www.smartplanet.com/blog/thinking-tech/robot-solves-rubiks-cube-in-5-seconds-sets-world-record-video/9033.
[21] Rubiks Cube Robot, https://www.youtube.com/watch?v=4JSBoIbVtSg, 2012.
[22] SRI au salon Micronora, https://www.youtube.com/watch?v=nr_-tww-bTc, 2012.
[23] Anaglyph method used for 3D imaging, http://www.3djournal.com/001/anaglyph.php?lgd=0, 2012.
[24] P. Miller, T. Monynihan, Active 3D vs. Passive 3D, PCWorld, 2011 (April 14).
[25] 3D website, http://3dweb.hu, 2012.
[26] What's New in Google SketchUp 8, http://sketchup.google.com/intl/en/product/newin8.html, 2011.
[27] Gregory Junker, Pro Ogre 3D Programming, Expert's Voice in Open Source, Apress Publisher, 2006 (ISBN 9781590597101).
[28] The Internet Communications Engine, http://zeroc.com/ice.html, 2011.
[29] P. Galambos, P. Baranyi, VirCA as Virtual Intelligent Space for RT-Middleware, in: 2011 IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM), 2011, 140–145.
[30] Fanuc Robotics, SYSTEM R-J3 Controller Karel Reference Manual, FANUC Robotics North America, Inc., 1998.
[31] Soon-Jo Chung, Jean-Jacques E. Slotine, Cooperative robot control and concurrent synchronization of Lagrangian systems, IEEE Transactions on Robotics – Special issue on rehabilitation robotics archive 25 (June (3)) (2009) 686–700.
[32] Kinect, http://en.wikipedia.org/wiki/Kinect, 2012.
[33] Wii controller, http://en.wikipedia.org/wiki/Wii, 2012.

Dr. Sándor Kopácsi received his Electrical Engineering Degree, Specialized Engineering Degree, Ph.D. and MBA at the Technical University of Budapest, as well as, Engineer-Economist Degree at the Budapest University of Economical Sciences. At present he is a senior researcher in the Institute for Computer Science and Control (MTA SZTAKI) and part time professor at the Dennis Gabor College, Budapest. He was a visiting researcher at the Korea Institute of Science and Technology, at the Technical University of Chemnitz-Zwickau and at the Institute for Applied Systems Technology Bremen. He received János Bolyai Research Grant from the Hungarian Academy of Sciences, and Hungarian State Scholarship Eötvös. He has been teaching in English for several years at different Universities in Budapest. He is member of the Public Body of the Hungarian Academy of Sciences. He has more than 70 scientific publications in national and international periodicals and conferences, mostly in English. He has been involved in several EU projects. His professional interest includes knowledge-based simulation, animation, artificial intelligence, graphics systems, virtual reality, web programming and 3D Internet.

Prof. Dr. Ing. Habil. George L. Kovács, head of CIM Res. Laboratory in the Institute for Computer Science and Control, Professor at the University of Pécs and at the Budapest University of Technology and Economics. Dr. Techn., Ph.D. and Dr. Habil. from the Technical University, Dr. of Sciences from the Academy of Sciences. Main interest: Robotics, AI solutions, design and operation problems of manufacturing systems, flexible manufacturing systems. Research in designing, simulation, scheduling, real-time control, evaluation and quality control of flexible manufacturing cells and systems and in general complex information systems, life-cycle management, virtual and extended enterprises, sustainability, reuse, recycling. Manager of several European and Hungarian research projects. Member and position holder in several Hungarian and international organizations. He has more than 400 publications. Founding member of Euron/Eurobotics. Visiting researcher/Professor for at least 6 months in the USA, Soviet Union, West Germany, Mexico and Italy.

Dr. János Nacsa received his M.Sc. in Electrical Engineering and Ph.D. in Mechanical Engineering at the Budapest University of Technology and Economics. He has been working at MTA SZTAKI for 25 years and where he is a senior researcher. He has about 90 scientific publications and knows more than 110 citations on his papers. He has worked in several European research projects (PROARC, OSACA, XPERTS, ManuCyte). He has led for years the development and operation of a complex decision support system in the substation of the Paks Nuclear Power Plant. Main interest: intelligent MES, open controllers, knowledge sharing in manufacturing environment, advanced robotics.