

SYNCHRONIZED ROBOT CONTROL FOR INCREMENTAL SHEET FORMING IN THE CYBERSPACE

Sándor Kopácsi – János Nacsa – Ádám Kisari

1. SUMMARY

More and more advanced manufacturing technologies require synchronized cooperation of machines or robots. These complex operations can be modelled in a virtual environment, where the cyber objects are deeply connected and synchronized with the real objects to provide closer view and deeper understanding of the given processes.

In this paper – as an example – the basics of Incremental Sheet Forming will be introduced, and then the realisation in a virtual environment of the machines and its processes will be discussed. After that the creation and operation of 3D models will be reviewed.

2. INTRODUCTION

Due to the rapid development of the graphical modelling tools, accurate modelling of complex system became feasible. As a consequence of the advanced three dimensional (3D) technology considering both input and output devices, there is higher and higher need for creating 3D models.

Personal computers, graphic cards, and rendering systems make the spatial operation of 3D models possible. Using the information high way for accessing these three dimensional services we can create state of the art applications.

In the Cognitive Informatics Group of the Computer and Automation Research Institute of the Hungarian Academy of Sciences (MTA SZTAKI) has been developed a Virtual Collaboration Arena, called VirCA, that has been used for the modelling and control of an Incremental Sheet Forming (ISF) application in the robot test site of Computer Integrated Manufacturing Laboratory.

We have not just developed the model of the tools used for Incremental Sheet Forming, and made it alive in the Cyber Space, but the real manufacturing activity has been synchronized with the virtual processes.

3. INCREMENTAL SHEET FORMING

Incremental Sheet Forming (ISF) is an encouraging research area of manufacturing, where a metal or polymer sheet is formed with a round or flat forming tool into the final shape by a series of small localized deformations. The forming tool, carried by a CNC machine or by an industrial robot, depresses into the sheet and follows a tool path. The unique automotive, aeronautic and space industry [1] are interested in this new sheet forming technique, and experimental applications in medical aid manufacturing are also fairly promising [2].

ISF can be divided into two groups, depending on the number of contact points between sheet, tool and die [3]. The term Single Point Incremental Forming (SPIF) is used when one side of the part is supported by a faceplate (Figure 1. [4]).

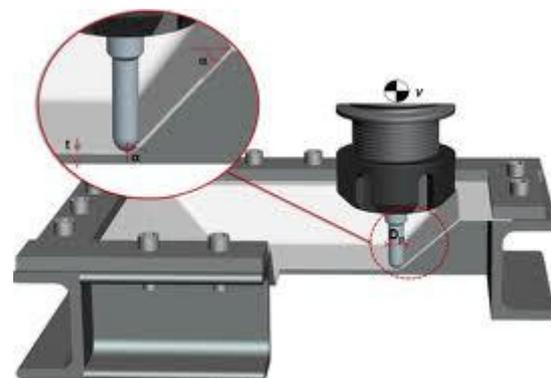


Figure 1.
Set-up of Single Point Incremental Forming.

The process is called Two Point Incremental Forming (TPIF) when a partial or full die is present to support the sheet. It is worth mentioning that a new and flexible variant with two synchronized forming tools [5] can make almost arbitrary free form surfaces. However, in all of these examples the optimization of the process parameters needs to be solved.

4. ISF EXPERIMENTS IN MTA SZTAKI

In the recently established robot laboratory in MTA SZTAKI an experimental environment has been set up for testing the principles of single point incremental sheet forming [6]. Figure 2. shows the experimental environment for hot SPIF tests, where warm air heating equipment (hot air gun) fixed on an industrial robot (MITSUBISHI RM 501) can synchronously be moved along with the incremental forming tool mounted on a small milling machine.

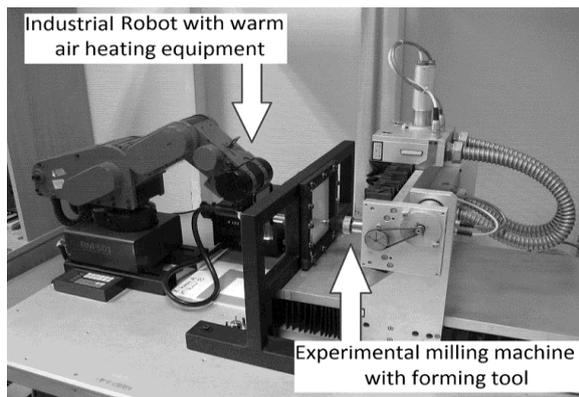


Figure 2.
Set-up of Single Point Incremental Forming

Both machines have EMC2 machine controller [7] that is a Real-time Open Control for Machine Tools and Robots. The controllers together with its Ubuntu Linux based operating systems were set up for both machines. The real time features of the controller are solved using real time patch of the normal Linux operating system. All the parts together were installed from a live CD solution.

A microprocessor controlled SMD Rework Station, with temperature and airflow adjustment possibilities, is used to apply local heat flux into thermoplastic materials. The experiments were carried out with commercial polyvinylchloride (PVC) sheets.

5. MODELLING ISF IN THE CYBERSPACE

5.1 VIRCA environment

The VirCA (Virtual Collaboration Arena) is a component based, distributed, interactive virtual reality manager system for connecting virtual and real rooms, displaying and manipulating the devices in them, realizing collaboration that way [8].

VirCA handles real, physicals as well as virtual spaces in order to realize distributed, interactive collaboration. To the virtual space different real

rooms can be connected, providing the relationship to the real applications. 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 virtual and real devices can refresh their representations through so-called trackers and can retrieve information from the virtual room through virtual sensors.

The novelty of the VirCA system is on the one hand the component based, distributed build-up. The single components communicate according to the RT-Middleware [9] standard through typified ports with each other. So the real and virtual resources represented by them can be used easily for the realization of the given task. Devices and services 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 innovation of the VirCA system on the other hand is the connection of virtual and real physical rooms. The user can see and manipulate the different, usually distant devices together, on a standard surface. With the help of the virtual objects, the cooperation of such real devices far from each other can be tested without the development of a simulator, the transport or purchase of which would be expensive otherwise, so VirCA is an efficient test environment.

5.2 System architecture

The current information system consists of the VirCA environment itself, the different independent objects in the environment called cyber devices and those real objects that have IT parts.

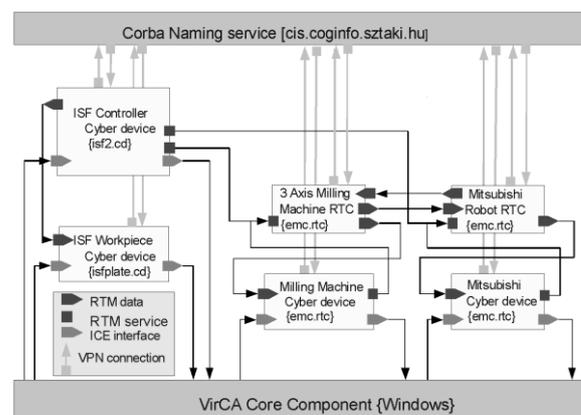


Figure 3.
ISF2 system architecture

Figure 3. explains the structure of the system. Each box represents a separate program. They are connected to each other based on the Corba naming service (the upper box on the figure) that manages them (gives them identical addresses, checks the consistency of the interfaces and activates/deactivates them).

The bottom box represents the VirCA system that has connection with four different cyber devices using ICE interface [10] (four different boxes in the figure). They are the following: two manufacturing machines (the milling machine and the robot), the cyber device representing the workpiece (*isfplate.cd*) that is manufactured by the two machines, and finally the cyber device of the *isf2.cd* process itself. They send visual data to the VirCA 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, they are using the 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.

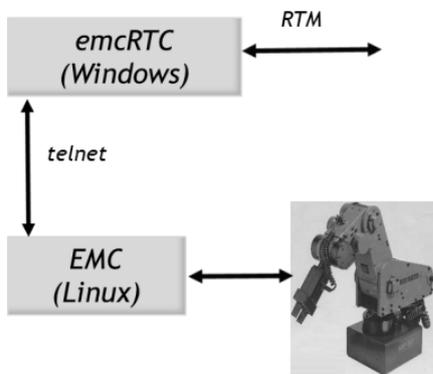


Figure 4.
Coupling EMC into the RT Middleware

5.3 Network interface units of the machines

As both machine controller is based on EMC in the experimental environment a common solution had to be developed to couple the controllers into the VirCA. As EMC2 has some complex, but poorly documented key elements, mainly the real-time Linux patch, called RTAI [11], it was decided not to modify the EMC2 Live CD that was used in case of both machines. So the telnet-based *emcrsh* [12] was chosen to provide a network interface to the EMC controller. In the

emc.rtc program the RT Middleware interface is gatewayed towards the EMC telnet commands (Fig. 4)

5.4 Cyber Devices

The cyber devices of the system have very similar software structure. The main class is based on the so-called cyber device API, that realizes the ICE-based connection to the VirCA, while a subclass is used for the RT Middleware-based interface to communicate with the other programs of the application.

It can be seen, that the cyber devices of the robot and the milling machine have the same functionality. The basic difference is their visual representation (Fig. 5) and the type of their movement. While the robot has rotation movements, the mill has translation movements of their joints. It worked out in the realization, that the same cyber device program (*emcCD*) is running in both cases, and it is decided during the initialization whether a certain issue belongs to the robot or to the mill.

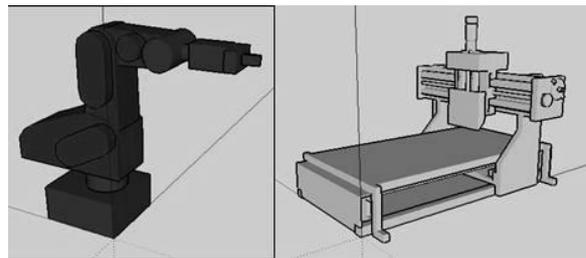


Figure 5.
Virtual model of the robot and the mill

Both *emcCD* programs send remote commands (e.g. product select, setup, start, home etc.) to their controller, get actual positions of their joints and visualize them. It knows from the setup which joint represents translation or rotation movements.

The *isf2* cyber device is the representation of an abstraction, the forming process. Visually the forming frame (Fig. 8.) that stands between the two machines and holds the workpiece has been selected to incarnate the abstraction in the virtual arena.

The *isf2* cyber device remotely controls both machines together. Commands like product selection, setup, and start are sent to both machines and their works are prepared for the common task, where their running is synchronized. The *isf2* also controls the appearance of the product as it has a lookup table, that maps

the g-code lines with the visual state (mesh files) of the product.

The *isfplate* cyber device is the virtual representation of the workpiece. It has no active role, but as the production is progressing, the view of the product is changing (Fig. 7)

6. ELEMENTS OF 3D ENVIRONMENT

If we want the user to immerse in the virtual reality space, we have to provide realistic feeling, so the model should be as accurate as possible. The most popular 3D modeller programs, like Google Sketchup, Autodesk 3ds Max, Autodesk Maya, Blender, Solid Works, etc. offer a several features with which we can create detailed and accurate models.

The 3D environment of ISF2 for VirCA has been created and finalized in Google Sketchup 8 [13]. Some of the original 3D visual components have been created in other graphic modeller 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 VirCA is Ogre [14], therefore the visual components of ISF2 have been supplied in Ogre (mesh and material files) format. The Ogre mesh and Ogre material files have been exported from Google Sketchup, using "Sketchup To Ogre Exporter" plug-in. The direct conversion from Autodesk 3ds Max format (3ds) to Ogre mesh did not succeed, because the converter, that we tried (3ds2mesh) hasn't created material files.

Converting the 3D components from Google Sketchup format to Ogre mesh, we had to realize, that the coordinate systems are different. After exporting the models from Google Sketchup to Ogre the coordinate system has turned 90 degrees around the X axes, which means that +Z in Sketchup will be +Y in, and +Y will be -Z in Ogre (Fig. 6).

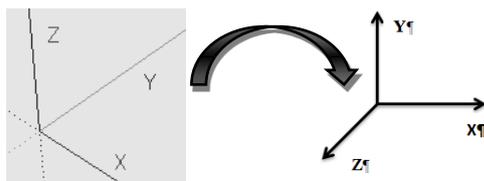


Figure 6.
Coordinates in Google Sketchup and in Ogre

The Ogre mesh and material files have been built together for test purposes in a sample application. The most important goal of this test was to set the positions of the individual components, to see how the complete scene can be assembled. Some of the components have been registered as simple objects, while others have been registered as hierarchical objects. The hierarchical objects have parents, which means very simply, that they move and rotate together.

The initial positions of the individual components have been calculated in the Ogre coordinate system. The values are the position of the origin of an element relative to its parent. If there is no parent (i.e. not a hierarchical object), then the coordinate values are relative to the corner of the table on which all the components are sitting.

For all the components a new 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, as they were exported, but in the case of the robot, the orientation of its components has been set up to the required angles.

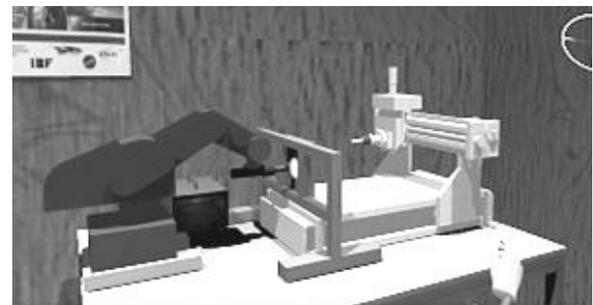


Figure 7.
Incremental sheet forming in VirCA

One of the components of the ISF2 environment, the table (*isf_table*), on which other components are located, has been placed as a static object directly in the room. All the other components have been registered to Ogre in the *emc.cd* and *isf2.cd* programmes.

The MITSUBISHI RM 501 robot (Fig. 5) has been built up by five components representing its non-moving part and its four moving parts. The robot is sitting on a base consisting of two individual bars that are fixed to the table.

On the fifth components of the robot a hot air booster has been fitted. The source of the hot air is modelled by a black box, with a jpg file as a

texture on it. We had to use the mirror image of the original picture.

The experimental milling machine (Fig. 5) used in this project has been modelled by four individual components. The first component has been fixed on the table, other parts remain movable (either horizontally or vertically).

The parts to be formed by the Incremental Sheet Forming technique have been modelled by 11 different mesh files, for two different types, i.e. a cone (Fig. 8) and a pyramid. For both types a faceplate has been applied, which helps in forming the parts (Fig. 9).

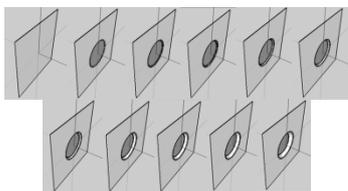


Figure 8.
Different phases of cone type workpiece

A frame has been placed on the table between the robot and the milling machine, which holds a faceplate that helps in forming the parts (Fig. 9). In our virtual environment this part, as it has been discussed earlier, represents the abstraction, since the frame should be selected, if we want to control the system.

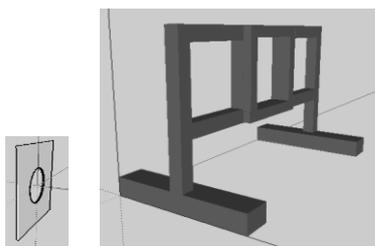


Figure 9.
Faceplate of a workpiece and the ISF frame

7. OPERATION OF THE SYSTEM

Currently a special web based tool [15] (Fig. 10) supports the connection of the different programs with the proper interfaces and it starts (activates) the whole system. It controls the states and connections of all the affected programs.

All the connections described above can be seen on the Fig. 10. Every cyber device is connected to the central VirCA program with two different type of connections (they get commands

from it and send their actual state including their positions). The other connections are using RT-Middleware to allow data exchange among the cyber devices and the two further programs that are the controllers of the real devices.

After the activation of the programs, both machines and the ISF frame will appear on the previously empty table and the system is ready for the collaboration. It also means, that there is living connection between the real machine controllers and their cyber devices.

The user can select both machines and the ISF process, and give them commands. To reach the process itself, the frame between the machines are used. It incarnates the process itself even it is a passive piece of metal. So the user can learn naturally, that the commands to the frame mean the commands to the ISF process. If the user want an ISF production, then first he or she chooses, which workpiece to produce, then setups the process and finally starts it. The commands passed only to the robot or to the mill will not result ISF production because they have no effects to the other machine.

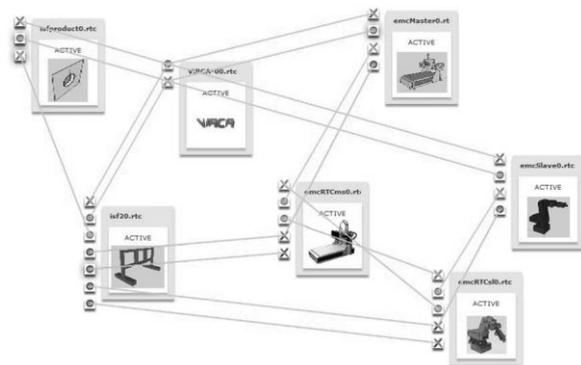


Figure 10.
Connection of the ISF2 programs

8. CONCLUSIONS

A challenging manufacturing process and its application in a virtual environment were introduced in this paper. The two-machine-based operation has been managed within the VirCA environment that helped in the integration of the components.

Even the VirCA provides a continuously 3D animation rendering the running processes, the tests proved that an additional video stream is very effective in the sense of understanding.

The different operation parameters (e.g. the temperature of the hot air, the depth and the speed of the forming, and the strategy of the

forming) were examined using this VirCA based experimental system.

The VirCA environment allows us to be in a virtual environment and interact with it knowing that all the actions will be also done somewhere in a real environment. So real process can be easier examined and understood in this virtual environment. Furthermore the system provides the possibility to extend the communication with the user in a more effective way, for example the monitoring of the temperature (in case a temperature sensor is attached to the system) can be realized with the variation of a noise or with different colours of the workpiece in the virtual environment.

This ISF approach is the first one, where real industrial processes were introduced into the VirCA environment.

9. LIST OF REFERENCES

- [1] Rauch, M., Hascoet, J. Y., Hamann, J. C., Plenel, Y. 2009. Tool path programming optimization for incremental sheet forming applications. *Computer-Aided Design* Vol. 41, No. 12, 877–885.
- [2] Duflou, J.R., Lauwers, B., Verbert, J., Gelaude, F., Tunckol, Y. 2005. Medical Application of Single Point Incremental Forming: Cranial Plate Manufacturing. In *Proc. of the 2nd Internat. Conf. on Advanced Research in Virtual and Rapid Prototyping VRAP*, Leiria, 161–166.
- [3] Hirt, G., Ames, J., Bambach, M., Kopp, R. 2004. Modeling and experimental evaluation of the incremental CNC sheet metal forming process. *CIRP Annals – Manufacturing Technology*, Vol. 53, No. 1, 203–206.
- [4] G. Ambrogio, L. Filice, F. Gagliardi: Formability of lightweight alloys by hot incremental sheet forming, *Materials & Design*, Volume 34, February 2012, Pages 501–508
- [5] Tisza, M., Paniti, I., Kovács, P. Z. 2010. Experimental and numerical study of a milling machine-based dieless incremental sheet forming. *International Journal of Material Forming*, Volume 3, Supplement 1, 441-446.
- [6] Imre Paniti, CAD API based tool path control for novel incremental sheet, forming, *POLLACK PERIODICA* 5:(2) pp. 81-90.(2010), DOI:10.1556/Pollack.5.2010.2.8
- [7] EMC – Enhanced Machine Controller, 2011, www.linuxcnc.org/
- [8] P. Galambos, B. Reskó, and P. Baranyi, 2010. Introduction of Virtual Collaboration Arena (VirCA), *The 7th International Conference on Ubiquitous Robots and Ambient Intelligence*, Busan, Korea, 2010, pp. 575-576.
- [9] N. Ando, T. Suehiro, K. Kitagaki, T. Kotoku, W-K Yoon: “RT-Middleware: Distributed Component Middleware for RT (Robot Technology)”, 2005 *IEEE/RSJ International Conference on Intelligent Robots and Systems (IROS2005)*, pp.3555-3560, 2005.08, Edmonton, Canada
- [10]The Internet Communications Engine (Ice), 2011, zeroc.com/ice.html
- [11]RTAI – the RealTime Application Interface for Linux from DIAPM, <https://www.rtai.org/>
- [12]emcrsh – text-mode interface for commanding emc2 over the network, 2011, linuxcnc.org/docs/html/man/man1/emcrsh.1.html
- [13]What's New in Google SketchUp 8, 2011, sketchup.google.com/intl/en/product/newin8.html
- [14]Gregory Junker: *Pro Ogre 3D Programming, Expert's Voice in Open Source*, Apress Publisher, ISBN:9781590597101, 2006
- [15]P. Galambos and P. Baranyi, “VirCA as Virtual Intelligent Space for RT-Middleware,” in 2011 *IEEE/ASME International Conference on Advanced Intelligent Mechatronics (AIM)*, 2011, pp. 140-145.