



Development of a plastic injection molding training system using Petri nets and virtual reality

MA Bin (马斌)[†], GUO Zhi-ying (郭志英), ZHOU Hua-min (周华民)

(National Die and Mold CAD Engineering Research Center, Shanghai Jiao Tong University, Shanghai 200030, China)

[†]E-mail: mabin@sjtu.edu.cn

Received May 20, 2005; revision accepted Sept. 1, 2005

Abstract: In this study, Virtual Reality (VR)-based plastic injection molding training system (VPIMTS), which can be modeled as an integrated system with a task planning module, an intelligent instruction module, a simulation module, and virtual environment (VE) module, was developed. Presented in this paper are an architecture of VPIMTS, a practical knowledge modelling approach for modelling the training scenarios of the system by using Petri nets formalism and key techniques (FEM, injection molding procedure modelling) which have been developed independently. The utilization of the Petri net model realized the environment where the trainee can behave freely, and also made it possible to equip the system with the function of showing the next action of the trainee whenever he wants. The overall system is a powerful approach for highly improving the trainee's comprehension and injection molding study-efficiency by building digital, intelligent, knowledgeable, and visual aids.

Key words: Virtual reality, Petri nets, Plastic injection molding, Virtual training, Finite element methods

doi:10.1631/jzus.2006.A0302

Document code: A

CLC number: TP391.9

INTRODUCTION

At present, plastic as a substitute for metals in products is the most important mainstay in industrial raw materials. Furthermore, plastic injection molding is one of the most common and versatile methods for mass production of complex plastic parts with excellent dimensional tolerance. Since the technology of injection molding has been widely applied in manufacturing, training on injection molding is of great importance. The traditional approach is to consult some technical books on injection molding and an operating manual and follow their instructions. It is not only time-consuming but sometimes, important information is not retained. A trainee may also look for advice from experienced workers. But lengthy explanations to trainees are mostly ineffective. Training in injection molding can possibly be done in a real factory where the trainee practices operating a real injection machine, but it is expensive and dangerous because mishandling during the process of training may damage the expensive injection machine.

In view of the above findings, the development of an intelligent, knowledgeable and digital training platform for the training in injection molding is critically important. The learning platform for virtual injection molding developed by Zhang (2002) uses C-Mould software to calculate analytical data by using finite element methods, and integrated with the animation of Flash, simulates the injection molding machine motions, helps the trainee to realize the operating condition of the injection molding machine and is a good tool for the training of novices in injection molding. But, the main shortcoming of this platform is that it is unattractive because of the use of 2D images, which makes this platform lack the "feeling" of reality, more importantly, the human-computer interaction.

Virtual Reality (VR) technology is applied as a solution to the training system for injection molding. VR technology makes it possible for anyone to experience various kinds of operations in a virtual environment just like in the real world. VR-based training is considered to be one of the fields to which

VR technology can be applied effectively (van Dam *et al.*, 2002; Vora *et al.*, 2002; Wang *et al.*, 2004). But, although the availability of VR in the field of training has been recognized, one remaining problem of a training system in a virtual environment (VE) is that the behavior of the trainee is very limited and the workload of the instructors is heavy.

The aim of this study is to develop a new type of injection molding training system in VE in which the trainee can behave freely and can get advice on what to do next whenever he wants in order to reduce the workload of the instructors. In this study, the formalism of Petri nets is applied for representing the operating and injection molding procedures.

Petri nets constitute a general tool for describing the behavior of concurrent discrete events (Sorensen and Janssens, 2004; Kiritsis *et al.*, 1999). The use of a Petri net model realizes the environment in which the trainee can behave freely, and is also the function that shows the next action to implement in a virtual environment to the trainee. Furthermore, in this study, an immersive and interactive VPIMTS is created on an Onyx2 workstation, a toolkit WTK and OpenGL interface by using VR technology and finite element methods (FEMs). A practical knowledge modelling approach is applied to model the training scenarios of the system by using Petri nets formalism. The overall system is a powerful approach to improve the trainee's comprehension and study-efficiency on injection molding by developing digital, intelligent, knowledgeable, and visual aids.

FRAMEWORK OF THE VPIMTS

In this approach, a VPIMTS is composed of four interactive modules: a task planning module, a simulation module, an intelligent instruction module, and VPIMTS environment. Fig.1 illustrates the modules and the message flow among them. Briefly, the functionality of the modules is as follows:

The responsibility of the task planning module is to adaptively generate appropriate training task plans based on a Petri net according to training goals. It is composed of a knowledge base, a goal decomposition, an action analysis and Petri net modelling. The final training goal of the VPIMTS can be formulated as "make the trainee familiar with the manual operation of an injection machine and know the procedure of

injection molding". Based on the knowledge of injection molding, first, decompose the final complex training goal into simple sub-goals and then analyze plausible actions of the trainee. Then construct goal-task decision tables from which to extract the corresponding training task according to the training goal inputted by trainee. Finally create the training plan using Petri net.

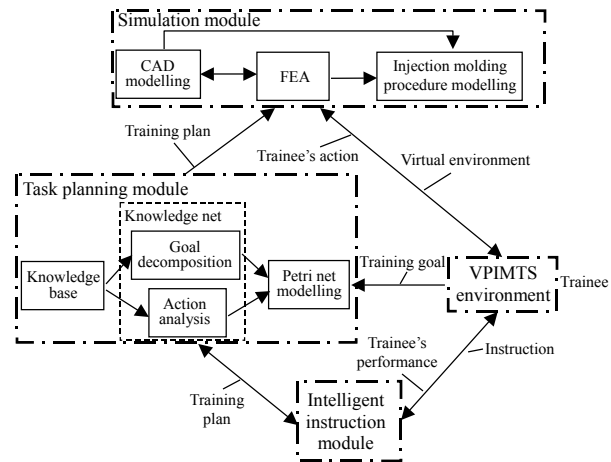


Fig.1 Program architecture

The simulation module receives the training plan from the task planning module, and initializes the model including CAD modelling, FEM and injection molding procedure modelling. The shape and structure of an injection mould and a machine are different (for example a double-plate or three-plate injection mould and a horizontal or vertical injection machine). Also 3D geometric modelling of manufacturing equipments including injection mould and machine is fussy and complicated work by using CAD software. Because of these reasons, with integration of OpenGL interface, a geometric model of an injection mould and an injection machine based on the boundary representation is created by using Delaunay triangulation technique (Escobar and Montenegro, 1996). In this system, the database of geometrical model of injection mould and injection machine is created.

The intelligent instruction module continuously receives messages from the task planning module and provides timely instruction and recommendations to the trainees according to their performance.

Finally, the VPIMTS environment sends training goals inputted by a trainee to the task planning module, receives the instructions from the intelligent in-

struction module, measures the trainee's actions and produces various simulations in the virtual environment. Collision detection (Su *et al.*, 1999) is applied on the interaction between trainees and VPIMTS environment.

The workstation used for a prototype implementation of VPIMTS consists of an SGI Onyx2 system which produces graphical simulations using WorldToolKit as a programming library, Datagloves with a hand tracker, which measure the gestures of the trainee's hands, a 3D sound server, and three high power video projectors. The images are projected in frame sequential stereo, and Head Mounted Display (HMD) is used to decode the display into a true 3D image. Position and orientation tracking is done by Ascension's Flock of Birds with an Extend Range Transmitter (ERT) which uses a pulsed, d.c. magnetic field and can determine 6 degree of freedom information from each of the possible 29 receivers. Measurements can be made at a rate of 10~144 Hz and range of ± 2.4 m from the ERT. Trainees in the effective range of the tracking unit have an optically correct view, while other users perceive visual distortion proportional to their distance from the tracking unit. In this study, based on the virtual scenery (shown in Fig.2) the method that created the VPIMTS will be discussed in detail.

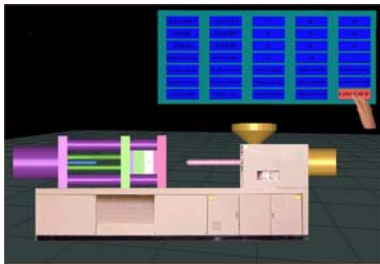


Fig.2 Schematic model of an injection machine

DEVELOPMENT OF THE TASK PLANNING MODULE

Training tasks form the basis of a training scenario and are constructed and organized toward some training goals. Sets of relevant training tasks are called training plans (TPs), which are modelled using Petri nets, and then stored in the knowledge base of VPIMTS. The goal of developing the VPIMTS is "make the trainee familiar with the manual operation

of an injection machine and know the procedure in injection molding". The next sections elaborate on the final training goal decomposition, the action analysis made by trainee and TP-net modelling using Petri nets.

Goal decomposition knowledge

Before going on to the development of the VR-simulation and the related Petri net model for the injection molding procedure, we make a detailed task analysis by referring to the training materials (manual and video tape). Based on the results of task analysis, using the hierarchical task analysis (HTA) approach, the final training goal is decomposed into three sub-goals (as shown in Fig.3): SG00-01, SG00-02, and SG00-03. The sub-goal SG00-01 is further decomposed into two sub-goals: SG00-11 and SG00-12. SG00-02 is further decomposed into four sub-goals: SG01-11, SG01-12, SG01-13, and SG01-14. The interpretation of each goal is listed in Table 1.

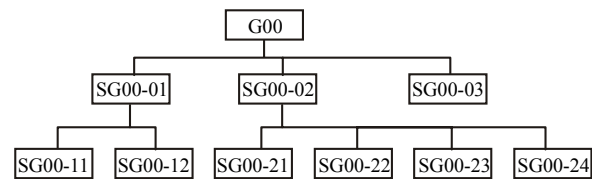


Fig.3 Goal trees in the VPIMTS

Table 1 The interpretation of goal in the VPIMTS

Goal	Interpretation
G00	To familiarize trainees with the manual operation of injection molding
SG00-01	To learn how to initialize the injection equipment
SG00-02	To familiarize trainees with the process of injection molding
SG00-03	To know how to finalize the task and shut down the operation
SG00-11	To familiarize trainees with the initialization of injection machine
SG00-12	To master the operation setting
SG01-11	To learn how to test the processing parameters in manual mode
SG01-12	To learn how to set the machining parameters for automatization
SG01-13	To familiarize trainees with the process of injection molding in auto mode
SG01-14	To know how to take away the product in manual mode

To generate suitable and sufficient training plans that can achieve the training goals, we use decision

tables to represent the knowledge for mapping goals (sub-goals) into training tasks. All training goals with the serial number of corresponding training tasks and corresponding training tasks with training goal's number form a set of GOAL-TASK decision tables. Table 2 shows an example of such a decision table.

Table 2 Shows an example of such a decision table

No.	GOAL-TASK
1	SG00-11: To familiarize trainees with the initialization of injection machine (8)
2	SG00-12: To master operation setting (9)
3	SG01-11: To learn how to test the processing parameters in manual mode (10)
...	...
8	T1: Initialization (1)
...	...
10	T3: Test the processing parameters in manual mode (3)
...	...
12	T5: Familiarize trainees with the process of injection molding (5)
...	...
14	T7: Shut down (7)

Action analysis

The aim of this study is to develop a new type of injection molding training system in VE where the trainee can behave freely and can get advice on what to do next. But action in the virtual environment may be impossible in the real environment. So action violation occurs. In order to overcome action violation, it is important to forecast the various possible action violations and effective dilemma resolution plane should be given. So, an action constraints model is created and added to the knowledge base. Based on training materials (manuals and video tapes) the trainee's actions during the training course must be analyzed and classified into four types: (1) Normal action, the same action as the one designated as the correct action in the manual; (2) Inhibited action, action strongly inhibited or warned by the manual, although possible to implement; (3) Possible action, action not specified in the manual, although possible to implement; (4) Impossible action, physically impossible action.

Although Types (1) and (2) should be considered in designing the VR-based training system, the setting

of the remaining Types (3) and (4) is a difficult problem. If we want to set the more coverage by Type (3), the system should have the capacity to cope with more versatile situation and so, is difficult to design, and Type (4) will be difficult to realize. We decided to set the range of Type (3) as wide as possible, and to use the information or warning to instruct the trainees undergoing training.

The behavior of trainees, a virtual injection machine and a mould in the VPIMTS can be classified into two parts: discrete behavior and continuous behavior. Discrete behavior that seems to be reactive can be described by the triplet (*input_action*, *internal_action*, *output_action*). By adopting a black box approach, the triplet can be rewritten by the pair (*input_action*, *output_action*). Since the input/output action pair alters the internal states, a discrete transition can be denoted by the quad (*input_action*(*i*), *previous_state*(*s*), *output_action*(*o*), *next_state*(*s*)). We call this possible transition a repertoire fragment. A repertoire fragment is then constructed from a finite sequence $i_1s_0o_1s_1, \dots, i_k s_{k-1} o_k s_k$ of alternating states and actions beginning with a start input action such that each quad ($i_{j+1}; s_j; o_{j+1}; s_{j+1}$) is a repertoire fragment.

In continuous behavior, the virtual object can independently update its internal states and external states over intervals of time, without being invoked externally. That is, continuous behavior is proactive, rather than reactive (or interactive) as in discrete behavior. All actions of continuous behavior are locally controlled. Continuous behavior is based on asynchronous message-based interaction. Therefore, continuous behavior can be described by the triplet (*previous_state*, *action*, *next_state*) not by the input/internal/output action triplet. The repertoire fragment for continuous behavior is slightly different from that of the discrete one. A repertoire fragment is described as a finite sequence $t_0s_0a_1s_1t_1, \dots, t_{k-1}s_{k-1}a_k s_k t_k$ of alternating states and actions beginning with a start state (s_0) at time t_0 , ending with a state (s_k) at time t_k , and returning to the start state (s_0), which can be simulated by a single-loop program.

TP-net modelling

In a VPIMTS environment, in order to reach a certain training goal, a set of training tasks is organized into a plan called a training plan (TP). Then, the TP is specified as a specialized PN (TP-net) and is

stored in the knowledge base of the system. A TP-net represents a non-sequential causality among tasks and is used to personalize a TP-net by adapting to the trainer’s training status.

A TP-net is a special kind of PN defined with the following restrictions:

- (1) Only one transition can be “fired” at a time;
- (2) A TP-net is a safe PN;
- (3) All arcs have weight equal to 1.

In addition, the following rules are applied to represent a virtual training plan using the PN model:

(1) Each machining operation or a well-established group of machining operations is represented by a transition T_i ($i=1,2,\dots,n$);

(2) A common input-output place is defined called control place (represented by ‘double circle’)

(3) A transition called automatic transition is defined, which represents that the state of the objects sometimes changes without the trainee’s intention and is represented by two parallel bars.

(4) One token is assigned to a “virtual object”;

(5) Different plausible states of a “virtual object” plausible in the simulation are represented by an “object place” and the realization of that state is represented by the existence of a token in the object place.

Fig.4 shows an example of initialization injection machine using Petri net. The interpretation of places and transitions are listed in Table 3 and Table 4.

All TP-nets of a VPIMTS are stored in a TP-nets base. To retrieve a suitable TP-net from the base, for each training goal (or sub-goals) G , according to the serial number of corresponding training tasks, a list of tasks: $\{T(G)_1, T(G)_2, \dots, T(G)_i, \dots, T(G)_m\}$ is found by using the goal-task mapping decision table. For each training task $T(G)_i$ ($1 \leq i \leq m$), a corresponding TP-net is retrieved and kept in the retrieved list $\{T(G)\}$.

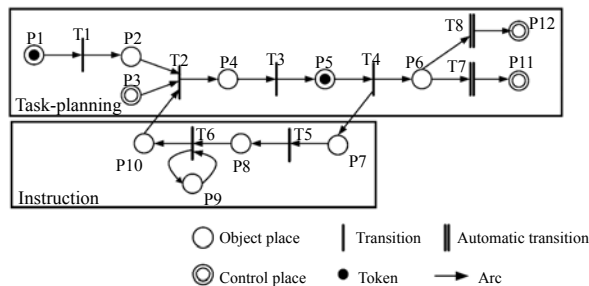


Fig.4 Initialization injection machine using Petri net

Table 3 The interpretation of places in “Initialization” mode

Place	Interpretation
P1	Start
P2	The injection equipment geometric model is loaded
P3	Action of trainee
P4	Collision between virtual hand and a button
P5	Button is pressed
P6	‘On’ button is pressed
P7	One of the other button is pressed
P8	“Power on first” appeared
P9	One of the other button is pressed
P10	‘Yes’ button pressed
P11	‘On’ button is red
P12	Machining sound is played

Table 4 The interpretation of transitions in “Initialization” mode

Transition	Interpretation
T1	Load injection machine model
T2	Touch a button (detect collision)
T3	Send response
T4	Press power ‘On’ button
T5	Instruct trainee to “Power on first”
T6	Press ‘Yes’ button
T7	Display the ‘On’ button red
T8	Play machining sound

SIMULATION MODELLING

Finite element methods of injection moulding

The traditional approach in simulation of plastic injection molding (PIM) applies the middle-plane technique or double surface-plane technique, based on 2D models, and uses finite element/finite difference method calculations.

Analytical results are shown on the surface of geometrical models, and the realistic flow behavior including the flow form and push mode of the molten polymer front cannot be visualized during the filling stage in injection molding. So, it cannot be appropriately applied in the VPIMTS. In view of this limitation, 3D solid models instead of middle-plane or surface-plane models were adopted, and a 3D control volume (shown in Fig.5) method was used to track the flow front in order to present the “fountain flow” phenomena enhancing the authentic feeling of the actual injection molding scene, the detailed algorithm of which is described in (Geng et al., 2003).

Sequential control using Petri nets

The whole process of injection molding includes

four procedures: closing die, filling mould, opening die, and ejecting product. The procedure of filling a mould consists of three sub-procedures: filling, packing and cooling. During the process of injection molding, motional mode can be classified into two parts: the running of molten polymer and the mechanical movement of parts of an injection mould and a machine. The running of molten polymer is created by visualizing the data, which are calculated by finite element method (EFM) based on the different time steps. Compared to the flow behavior during the process of filling, the motional behavior of injection mould parts is complex, so the characteristics of injection mould parts motion must be explained.

If there is no mechanical motion in the injection mould side core, the mechanical motion of parts of an injection mould is linear along one direction. So, motional mode of parts of an injection mould is simpler. The basic principle of the motion of parts of an injection mould is that motions are grouped to be uniform in terms of grouping by static mould, grouping by injection mechanism, grouping by moving mould, grouping by ejecting mechanism, and grouping by inclined slide. Each group has a uniform characteristic of motion. If there is side core mechanical motion, the pace of motion of the angle pin is calculated. The equation for the motional pace of the angle pin can be expressed as Eq.(1):

$$x = STEP / \cos \alpha, \tag{1}$$

where α is gradient angle of the angle pin, $STEP$ is a primary motional pace, and x is the motional pace of the angle pin. Based on the above ideas, the procedure of injection molding can be explained by using a Petri net, and is illustrated in Fig.6. The interpretation of places and transitions are listed in Table 5 and Table 6. The example of an injection mould in Fig.6 is a double-plate mould. Fig.7 shows the process of virtual injection molding based on the Petri net.

CONCLUSION

In this study, a VR-based plastic injection molding training system VPIMTS, developed on the basis of Petri nets, is outlined. The Petri net theory has been used as a tool for representing training task plans of injection molding. The overall system is a powerful approach for improving the trainee’s comprehension and study-efficiency on plastic injection molding by building digital, intelligent, knowledgeable, and visual aids. Future work includes the extension of the architecture to networked virtual environments.

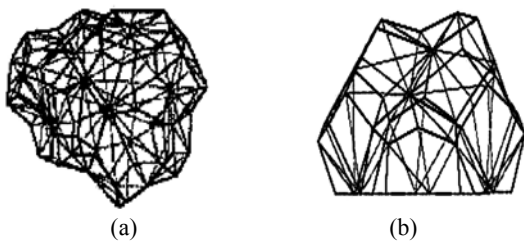


Fig.5 Graphical results obtained by FEM. (a) Control volume based on internal node; (b) Control volume based on boundary node

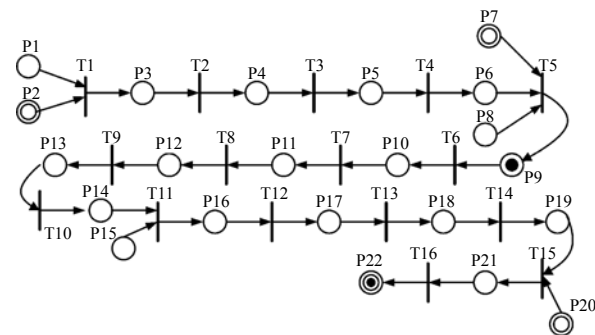


Fig.6 Modelling the process of injection molding using Petri net

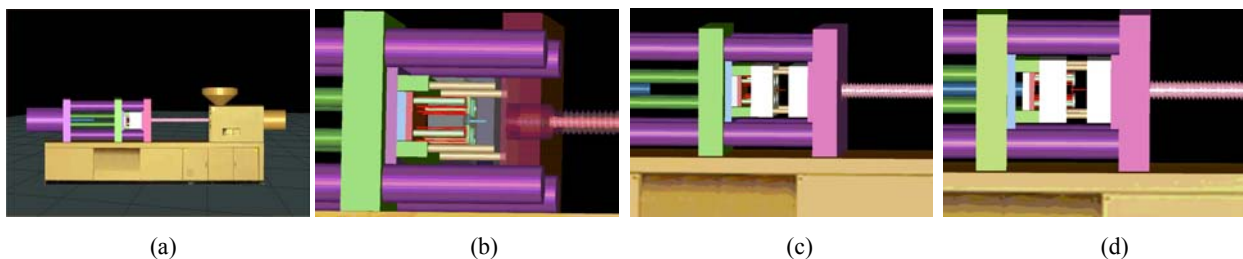


Fig.7 Embodies the Petri net. (a) Closing die; (b) Process of filling; (c) Opening die; (d) Ejecting the product

Table 5 The interpretation of places in the injection molding process

Place	Interpretation
P1	The injection equipment geometric model is loaded
P2	'Start' button is pressed by trainee
P3	Group of moving parts is moving
P4	Movement of parts stopped
P5	Injection bar is moving
P6	Movement of injection bar stopped
P7	Data of FEA is executed
P8	Zoomed up the viewpoints
P9	Showing the flow behavior
P10	Flowing stopped
P11	"Packing" appeared
P12	"Packing" disappeared
P13	"Cooling" appeared
P14	"Cooling" disappeared
P15	Zoomed down the viewpoints
P16	Group of moving parts and product are moving
P17	Movement of mould and product stopped
P18	Group of ejecting mechanism and product are moving
P19	Movement of ejecting mechanism and product stopped
P20	Action by trainee
P21	Collision between virtual hand and product
P22	Trainee took away the product

Table 6 The interpretation of transitions in the injection molding process

Transition	Interpretation
T1	Closing die
T2	Process of closing die is completed
T3	Start the process of injection molding
T4	Movement of injection bar is completed
T5	Start the process of filling mould
T6	Process of filling mould is completed
T7	Start the process of packing
T8	Process of packing is completed
T9	Start the process of cooling
T10	Process of cooling is completed
T11	Start the process of ejecting
T12	Opening die is completed
T13	Ejecting the product
T14	Complete the ejecting
T15	Touch the product (detection of collision)
T16	Send response

References

- Escobar, J.M., Montenegro, R., 1996. Several aspects of three-dimensional Delaunay triangulation. *Advances in Engineering Software*, **27**(1-2):27-39. [doi:10.1016/0965-9978(96)00006-3]
- Geng, T., Li, D.Q., Zhou, H.M., 2003. Numerical simulation of filling stage in injection molding based on a 3D model. *China Plastic*, **7**:78-81 (in Chinese).
- Kiritsis, D., Neuendorf, K.P., Xirouchakis, P., 1999. Petri net techniques for process planning cost estimation. *Advances in Engineering Software*, **30**(6):375-387. [doi:10.1016/S0965-9978(98)00126-4]
- Sorensen, K., Janssens, G.K., 2004. A Petri net model of a continuous flow transfer line with unreliable machines. *European Journal of Operational Research*, **152**(1):248-262. [doi:10.1016/S0377-2217(02)00653-7]
- Su, C., Lin, F., Ye, L., 1999. A new collision detection method for CSG-represented objects in virtual manufacturing. *Computers in Industry*, **40**(1):1-13. [doi:10.1016/S0166-3615(99)00010-X]
- van Dam, A., Laidlaw, D.H., Simpson, R.M., 2002. Experiments in immersive Virtual Reality for scientific visualization. *Computers & Graphics*, **26**(4):535-555. [doi:10.1016/S0097-8493(02)00113-9]
- Vora, J., Nair, S., Gramopadhye, A.K., Duchowski, A.T., Melloy, B.J., Kanki, B., 2002. Using virtual reality technology for aircraft visual inspection training: presence and comparison studies. *Applied Ergonomics*, **33**(6):559-570. [doi:10.1016/S0003-6870(02)00039-X]
- Wang, P., Kreuzer, I.A., Davies, R.C., 2004. A Web-based cost-effective training tool with possible application to brain injury rehabilitation. *Computer Methods and Programs in Biomedicine*, **74**(3):235-243. [doi:10.1016/j.cmpb.2003.08.001]
- Zhang, G.Y., 2002. Establishment of Digital Learning Platform for the Virtual Injection Molding. Master's Thesis, Mechanical Department of Zhongyuan University, Taiwan, China (in Chinese).