Simulation Models for Virtual Reality Applications

Grigore Albeanu

Spiru Haret University,
13, Ion Ghica Str, RO-30045, ROMANIA
E-mail: galbeanu@gmail.com

Abstract

The paper describes some simulation models used to implement virtual reality applications, addressing the presentation of the architecture of VR systems, VR applications in different fields, including medicine, an introduction to simulation techniques and a set of mathematical models for creating virtual scenes. The material represents a significant development of the presentation given at the workshop VRRM 2007: Virtual Reality in Rehabilitation Medicine, with details on mathematical aspects.

Keywords: Modelling, Simulation, VR applications

1. Introduction

During last decades, many modelling methods were proposed not only for computer-aided design, multi-queueing systems design, scientific visualization, e-learning, but also for entertainment. Recently, more research was dedicated to modelling virtual worlds, to model the behaviour of objects belonging to virtual environments (Dimitropoulos *et al*, 2008; Pasc *et al*, 2007; □arcă *et al*, 2008; Jung *et al*, 2005; Popovici, 2005; etc), and to simulate such a behaviour using computer graphics tools (Falcidieno & Kunii, 1993; Hagen *et al*, 1993; etc), virtual reality interfaces (Fuchs & Moreau, 2003; etc.) and languages, and augmented reality (□arcă *et al*, 2008).

This paper describes some of architectures suitable for VR applications (the second section) and illustrates appropriate simulation techniques (in the third section). In order to implement such techniques, not only information technologies are required but also a strong background in mathematical modelling. Some mathematical models based on recent developments are described in the fourth section.

Examples from medical applications, computer-aided design, scientific-visualization, e-learning and computer games are provided along the presentation.

The material represents a significant modification of the presentation given at the workshop VRRM (Albeanu, 2007), with details on mathematical aspects and the current state of the art.

2. Some architectures of VR applications

Simulated VR applications can be developed for important fields, according to (Albeanu, 2006): virtual current activities (e-learning, training in different subjects, games), virtual "teleportation" (virtual tourism, the study of micro and nano-structures, fluid flow visualization, volume visualization in medicine), virtual collaborative activities (network based games, teleconferences, virtual communities), virtual design (CAD, architecture, fashion), virtual management (urban management, workplace management, workstation usability, environmental protection), virtual exhibitions (antiquities, restoration, …), and virtual events (the study of different civilizations, old sites visiting, police investigation by replay, …).

Mainly, all applications need the participation of humans. Only some of them are off-line simulations. Not only real humans, but also virtual characters will be parts of some VR applications. This is why human modelling and animation is an important topic. Hence such a conceptual model includes a human model and an environment model. Of course, an interaction model will be also considered.

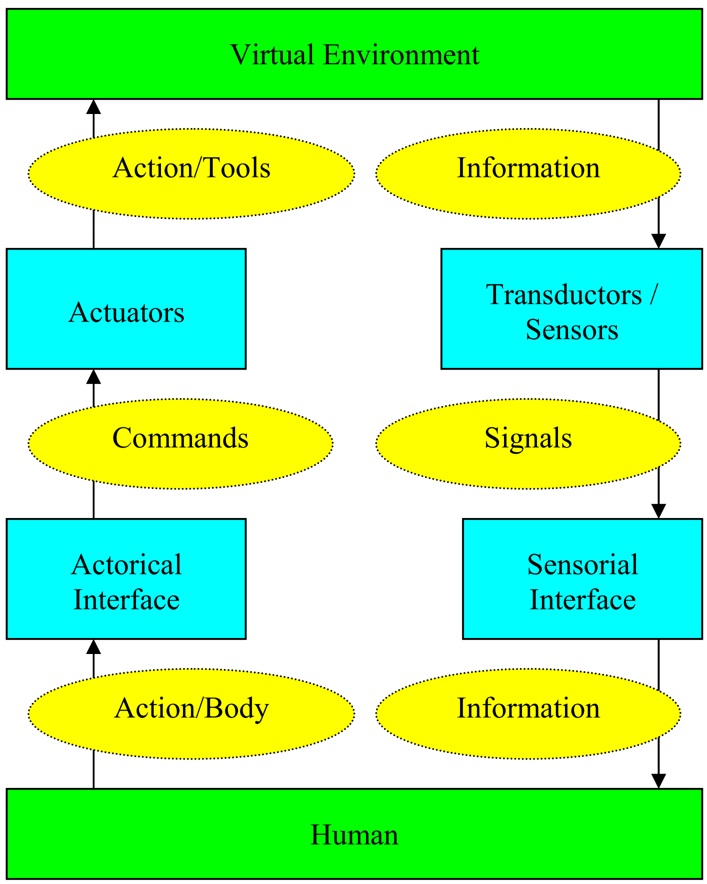

Figure 1. Virtual laboratory

Some architecture of a VR application is based on hierarchical decision graph evaluated repeatedly during simulation or normal running. Other VR applications use state machine transitions (different kind of automata, including cellular), cybernetic architectures based on feedback, etc. However, all VR applications have a modular structure. Let us show that IMHAP platform (Liang *et al*, 2007) is divided into three components: Model, Viewer, and Controller. Of course, the controller implements the interactions between user and the model. The architecture of a remote mechatronic virtual laboratory (useful for virtual training in robotics) is shown in *Figure 1*, based on the presentation from (Pasc *et al*, 2007) where the interaction is assured by commands and signals. Such architecture is suitable also for medical surgery at distance, or operation at distance, according to (Albeanu, 2007).

Another kind of architecture is the one proposed by (Boulic *et al*, 2003) which consists in layers. For instance, the H-Anim architecture contains an walking engine having three layers: generic walk pattern, gait personification, and walking trajectory controllers. All of them are designed in order to maintain the coherence of the model.

The applications designed in order to model some body parts for virtual bodies use different ideas based on: volume imaging technologies, surface rendering and hybrids models, and volume rendering, as (Waterworth, 1999) reviewed.

Other architectures are behaviour oriented like those proposed by (Popovici, 2005), or component oriented like for VR based applications developed by (Haller, 2001). Anyway, our biographical research establish a large collection of contributions (not all mentioned in references), but the basic ideas are those already presented above.

3. Simulation techniques for VR applications

Simulation is the second stage of every VR application, the first stage being the model development. The simulation techniques depend on mathematical models associated to the virtual model. The simulation models used for implementing the behaviour of different real/virtual actors/systems are based on discrete or continuous mathematical model.

When considering the simulation technology, the VR project manager will consider both virtual and physical systems architectures and their integration. For some VR applications, like those of collaborative nature, are necessary distributed simulation methods. The final stage deals with the validation of the simulation model and comparison of different simulation areas (such as vehicle, weather, medical, industrial, and entertainment).

Various mathematical methods are required in different simulation scenarios (matrix transformations, algebra, trig, complex numbers), as well as open-loop and closed-loop system theory, discrete versus continuous simulation, the use of databases in simulations, and the necessary real-world physics/biology/chemistry etc. The references (Bell and Fogler, 1997), (Dimitropulos *et al*, 2008), (Jung *et al*, 2005), (Metze *et al*, 2005), (Souza *et al*, 2007) and (Thelen and Anderson, 2006) are only some of a huge scientific literature dedicated to different aspects on simulation for different VR applications.

In the following we establish the main steps of a any simulation scenario: (1) establish the unit of time or/and distance depending on application; (2) establish the simulation time (how long?); (3) simulation start clock and uniform/variable time-step length; (4) setting the objects behaviour (movement, collision avoidance, …), and (5) generation, analysis and storage of new information.

Some VR applications use backward simulation. The backward simulation asks to start from a current or final state and to move backwards in time to an initial state in order to determine the sequence of actions (the trajectory, the path) for moving the system from the initial state to the current final state. Other VR applications use forward simulation based approaches that usually execute a single forward pass through time based on some dispatching rules to develop the sequence.

These techniques can be used also for motion generation required by some VR applications by automated or interactive control. Both keyframing and procedural methods can benefit from backward/forward simulations. Keyframing asks for key positions of objects to be animated and then interpolation is necessary to identify the positions in-between frames. Inverse kinematics (Zhao & Badler, 1994; Badler *et al*, 1999; etc) and different kind of interpolation procedures (Albeanu, 1999; Badler *et al*, 1999; Magnenat-Thalmann & Thalmann, 2004); etc) are implemented during simulation and motion generation. For particle systems, procedural methods can be developed based on laws of physics to generate motion. Not only individual objects but also groups of objects can be moved together during motion generation. Capturing or graphical design can also be used as INMEDEA uses in its web-based medical simulator.

If considering human walking, the simulator divides the simulation time in succession of phases: right takeoff (RT), right footstrike (RF), left takeoff (LT), left footstrike (LF), and its implementation will ensure that the body parts motion and contact with terrain or stairs looks realistic, and will solve pushing/collisions and other kind of interventions (Figure 2). (Thelen & Anderson, 2006) developed a powerful methodology based on forward dynamic musculoskeletal simulation model. According to (Multon *et al*, 1999) some methods will be mixed depending on the VR application type.

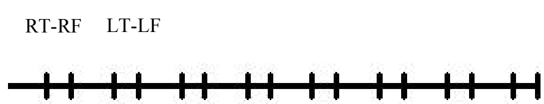

Figura 2. Time rule during RT-RF-LT-LF cicle

Of course, simulations models used for VR applications in chemistry (Bell & Fogler, 1997) or environment protection (Albeanu, 2007) are completely different from human walking simulation, but the main ideas about controlling the sequence of events remain.

4. Some mathematical models

Mathematical modelling is important not only for scientific aspects in industrial and business applications, but also for virtual learning and entertainment. Not only geometric transformations (2D, 3D), viewing transformations (parallel or perspective projection), clipping and hidden line or surface removal, well known in computer graphics, but mathematical models dependent on the concrete applications are required in order to create realistic behaviour of the objects belonging to a simulated environment.

In the following some models concerning trajectory generation, surface-terrain generation and object generation will be detailed.

4.1. Trajectory/terrain modelling

For the modeling of a walking/evolutionary trajectory, curves interpolated from data, or approximation curves can be used. Special curves could be obtained using trigonometric interpolation as described in (Albeanu, 1999). However, interproximation can be used as a mixed interpolation-approximation methodology as described in (Falcidieno & Kunii, 1993) by Cheng and Barsky (pag. 359). This methodology can also be used to model closed shapes.

Let $D = \{D_i \mid i = 1, 2, \ldots, n\}$ be a set of 2D data (points described as $D_{i[j]} = P_j\ (x_j, y_j)$, or rectangles described as $D_{i[j]} = [a_j, b_j]x[c_j, d_j]$, j = 1, 2, …, n. A common interproximation scheme uses cubic B-splines to fit the data set *D*.

The cubic B-spline curve is a piecewise curve of n+m-1 segments, requiring n+m+6 interpolating knots denoted by $T = \{t_{-2}, t_{-1}, t_0, t_1, \ldots, t_{n+m+3}\}$, where $t_{-2} = t_{-1} = t_0 = t_1 = 0$, $t_{n+m} = t_{n+m+1} = t_{n+m+2} = t_{n+m+3} = 1$, and $t_i = t_{i-1} + d_i$, where d_i is given according to the centripetal model of Lee (cited in (Falcidieno & Kunii, 1993)):

$$d_i = \frac{\left|A_i - A_{i-1}\right|^{1/2}}{\sum_{j=2}^{m+n}\left|A_j - A_{j-1}\right|^{1/2}}, 2 \le \mathrm{i} \le \mathrm{n} + \mathrm{m} - 1,$$

with A_i being $D_{i[j]}$ (the point or the centre of the rectangle), j = 1, 2, …, n.

The cubic B-spline defined on [0, 1] can be represented as

$$S(u) = \sum_{i=-2}^{m-n-1} C_{i-2} N_{i,3}(u),\ u \in [0,1],$$

where C_i are control points and $N_{i,3}(.)$ are normalised cubic B-splines defined related to the sequence of knots T.

Other trajectories can be obtained using trigonometric piecewise functions. If $F_i(.)$, i = 1, 2, 3, 4 are Hermite trigonometric polynomials having parameters α and β, the trigonometric curve with endpoints P_α and P_β, and derivatives P'_α and P'_β, has the representation: $h_{\alpha,\beta}(t) = P_\alpha F_1(t) + P_\beta F_2(t) + P'_\alpha F_3(t) + P'_\beta F_4(t)$, $t \in [\alpha, \beta]$.

A suitable representation of terrain regions in order to apply collision detection, or finding contact points with some objects is based on Bézier rectangles.

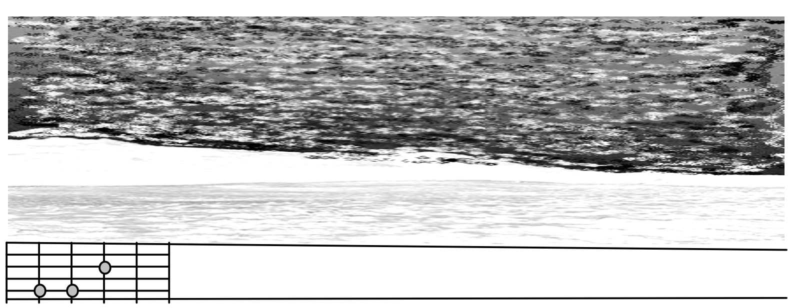

Figure 3. Terrain generation by Bézier rectangles with texturing

A Bézier rectangle of degree *m*x*n* has the form

$$S(u,v)=\sum_{i=0}^{m}\sum_{j=0}^{n}B_{i,m}(u)B_{j,n}(v)P_{i,j},\, 0\le \text{u, v} \le 1.$$

where $B_{i,m}(.)$, respectiv $B_{j,n}(.)$ are univariate Bernstein polynomials of degree *m*, respectiv, *n*, and $P_{i,j}$ are the control points of the Bézier rectangle.

For some applications a texture is applied in order to obtain a realistic view (Figure 3), but from many algorithmic tasks, only the skeletal of the terrain is required.

4.2. Object modelling

Solid physical objects can be represented as a combination (union, intersection, difference) of primitives like cubes, spheres, cones, cylinders, tetrahedrons or quadratic pyramids, etc. according to the Constructive Solid Geometry (CSG) based methodology. Other methods are: spatial enumeration, cell decomposition, boundary representation, and primitive instancing. Some useful models consider super-primitives like super-elipsoid and super-toroid objects and other entities obtained by mathematical transformations (translation along, rotations, …, etc) or sweep methods.

The most natural way to represent a CSG model is the *CSG tree*:

<*CSG tree*> ::= <primitive> |
<*CSG tree*> <set operation> <*CSG tree*> |
<*CSG tree*> <rigid motion>

where <primitive> is an instance of one of the primitives of the primitive data base, <rigid motion> is either a translation or a rotation, and <set operation> is either $\cup$, $\cap$, or — (setminus).

To animate the scene, both a static description and information about object movement is required. The information about the movement is described in an articulated model (the objects are connected by joints in a hierarchical structure). For some objects the motion is determined by rules (for instance, the laws of physics), specified also for deformable entities.

A special case for VR applications deals with trivariate data modelling, that means the construction of a function F(x, y, z) which interaproximates the relationship implied by the data (x_i, y_i, z_i; F_i). If (x_i, y_i, z_i) are interior points of some object, the model will provide information about attributes of other points belonging to the object.

This information is useful for a realistic rendering of the animated scene. A common method to identify a model uses the distance function approach (least square). Piecewise Hermite form of the spline models can also be used, and the coefficients will be identified by solving the obtained linear system of equation.

5. Conclusions

This paper described the main principles of the simulation models used to implement virtual reality applications. There are presented the architecture of VR systems, VR applications in different fields, including medicine, an introduction to simulation techniques and a set of mathematical models for creating virtual scenes.

The complexity of the subject is large and a strong mathematical background is necessary. Also, implementing VR applications asks for recent information technologies resources including virtual reality hardware and software tools.

REFERENCES

ALBEANU, G. (1999), On the Geometric Modeling of Curves by Trigonometric Polynomials, *Annals of Bucharest University, Mathematics-Informatics Series 48*, 1, 55-60.

ALBEANU, G. (2007), Lecture on Simulation Models for VR Applications, *VRRM-2007: International Workshop on Virtual Reality in Rehabilitation Medicine*, University of Medicine and Pharmacy "Carol Davila", Bucharest, Romania, September 24-25, 2007, Manuscript.

ALBEANU, G. (2006), Modelling and Programming VR Applications – an ICT undergraduate course, *Proceedings of ICVL 2006,* Bucharest, 83-90, http://fmi.unibuc.ro/cniv/2006/disc/icvl/documente/pdf/met/2_albeanu.pdf.

ANDERSON, F. C. and PANDY, M. G. (2001), Dynamic Optimization of Human Walking. *Journal of Biomechanical Engineering 123*, 381-390.

ANDERSON, F. C., ARNOLD, A. S., PANDY, M. G., GOLDBERG, S. R. and DELP S. L. (2006), Simulation of Walking. In *Human Walking*, Williams and Wilkins, 3rd Edition, Baltimore, http://nmbl.stanford.edu/publications/pdf/Anderson2006.pdf.

BADLER, N. A., PHILIPS, C. B. and WEBBER B. L. (1999), *Simulating Humans: Computer Graphics, Animation, and Control*. Oxford University Press, New York.

BELL, J. T. and FOGLER, H. S. (1997), The application of virtual reality to chemical engineering education, *Proceedings of the 1997 International Conference on Simulation in Engineering Education (ICSEE '97),* January 12-15, 1997, Simulation Series 29(2), Society for Computer Simulation, San Diego.

BOULIC, R., GLARDON, P. and THALMANN, D. (2003), From measurements to model: the walk engine, Proc. of 6th Conf on Optical 3D Measurement Techniques. *The 6th Conference on Optical 3D measurement Techniques*, http://infoscience.epfl.ch/getfile.py?docid=13105&name=Boulic_Glardon_Thalmann_CO3DMT_03&format=pdf&version=1.

DIMITROPOULOS, K., MANITSARIS, A. and MAVRIDIS, I. (2008), Building Virtual Reality Environments for Distance Education on the Web: A Case Study in Medical Education. *International Journal of Social Sciences 2*, 1, 62-70.

FALCIDIENO, B. and KUNII, T. L. (eds.) (1993), *Modeling in computer Graphics*, Springer Verlag.

FUCHS, P. and MOREAU, G. (2003), Le traité de la réalité virtuelle, Vol. 2: Création des environments virtuels & applications, Ecole des mines de Paris, Paris.

HAGGEN, H., MÜLLER, H. and NIELSON, G. M. (eds.) (1993), *Focus on Scientific Visualization*, Springer Verlag.

HALLER, M. (2001), A component oriented design for a VR based application. *International Workshop on Structured Design of Virtual Environments and 3D-Components at the Web3D 2001 Conference*, Paderborn, Germany, February 19th, 2001.

Inmedea simulator: http://www.inmedea-simulator.net, *INMEDEA GmbH*, Tübingen.
JUNG, B, AHAD, A. and WEBER, M. (2005), The Affective Virtual Patient: An e-Learning Tool for social Interaction Training within the Medical Field, *Training Education & Education International Conference (TESI 2005).* Proceedings Nexus Media, March, http://isnm.de/~aahad/Downloads/AVP_TESI.pdf.
LIANG, C.-H., TAO P.-C. and LI, T. Y. (2007), IMHAP-An experimental Platform for Humanoid Procedural Animation, *Proceedings of the third International Conference on Intelligent Information Hiding and Multimedia Signal Processing*, Tainan, Taiwan, http://imlab.cs.nccu.edu.tw/paper/2007.11iihmsp2007.pdf.
MAC NAMEE, B., DOBBYN, S., CUNNINGHAM, P. and O'SULLIVAN C. (2003), Simulating Virtual Humans Across Diverse Situations. *Proceedings of Intelligent Virtual Agents '03,* 159-163, http://www.comp.dit.ie/bmacnamee/papers/SimulatingVirtual HumansAcrossDiverseSituations.pdf.
MAGNENAT-THALMANN, N. and THALMANN, D. (editors) (2004), *Handbook of Virtual Humans*, John Wiley & Sons, Ltd.
METZE, J., NEIDHOLD, B. and WACKER, M. (2005), Towards a general concept for distributed visualisation of simulations in Virtual Reality environments, *IPT & EGVE Workshop*, 1–12.
MULTON, F., FRANCE, L., CANI-GASCUEL M.-P. and DEBUNNE, G. (1999), Computer Animation of Human Walking: a Survey. *The Journal of Visualization and Computer Animation 10*, 39-54
NG-THOW-HING, V. (2001), Anatomically-based models for physical and geometric reconstruction of humans and other animals, *PhD Thesis*, University of Toronto.
PASC, I. M., TARCA, R. C., POPENTIU-VLADICESCU, FL. and ALBEANU, G. (2007), On Designing Virtual Environments Based on Intelligent Mechatronic Systems, *ISSAT International Conference on Modeling of Complex Systems and Enviroments*, 16-18 July 2007, Ho Chi Minch City, Vietnam, 126-130.
POPOVICI, D. M. (2005), Behavioral aspects and behavior-oriented architectures in 3D virtual environments, *An. St. Univ. Ovidius Constanta 13*, 2, 61-74.
SILVERT, W. (2000), Modelling as a discipline. *Int. J. General Systems 30*, 261-282.
SOUZA, D. F. L., CUNHA, I. L. L., SOUZA, L. C., MORAES, R. M. and MACHADO, L. S. (2007), Development of a VR Simulator for Medical Training Using Free Tools: A Case Study. *Proc. of Symposium on Virtual and Augmented Reality (SVR'2007),* 100-105.
THELEN, D. G. and ANDERSON, F. C. (2006), Using computed muscle control to generate forward dynamic simulations of human walking from experimental data. *Journal of Biomechanics 39*, 1107 – 1115.
ARCĂ, R. C., IVĂNESCU, A., BARDA, I., ALBEANU, G., PASC, I. and POPENIU VLĂDICESCU FL. (2008): Augmented reality used for a remote robot control, *The 12th World Multi-Conference on Systemics, Cybernetics and Informatics* (WMSCI 2008), Orlando, Florida (Jun 29th -July 2nd), in press.
WATERWORTH, J. A. (1999), *Virtual Reality in Medicine: a survey of the state of the art*, http://www.informatik.umu.se/~jwworth/medpage.html.
ZHAO, J. and BADLER, N. I. (1994), Inverse Kinematics Positioning Using Nonlinear Programming for Highly Articulated Figures. *ACM Transactions on Graphics 13*, 4, 313-336.