Paper presented at the 22nd Asian Conference on Remote Sensing, 5 - 9 November 2001, Singapore.

CONCEPTUAL FRAMEWORK OF HUMAN SPATIAL BEHAVIOR SIMULATION BASED ON HLA

Rong XIE, Ryosuke SHIBASAKI, Professor
Center for Spatial Information Science
University of Tokyo
4-6-1 Komaba, Meguro-ku, Tokyo 153-8508
Tel: (81)-3-5452-6412 Fax: (81)-3-5452-6414
E-mail: {xierong, shiba}@skl.iis.u-tokyo.ac.jp
JAPAN

KEY WORDS: Human Behavior, HLA, Human Requirements, Conceptual Modeling, Simulation

ABSTRACT: GIS, essentially discussing how people interact with their environment, is now becoming a natural spatial process and the human interaction with the environment in space and time. Abstraction of real world and simulation of human behavior are just two questions of how spatial reality and human experience are linked. To address the problems, the paper introduces and explores the potential of using High Level Architecture (HLA) technology based modeling and simulation as a tool for expressing the complex interactions between human and environment. For undertaking such complex human action analysis, the paper prescribes the development of representation requirements within simulation, which are necessary to ensure the model and simulation represents exactly what the human wants it to represent. A conceptual model is then presented as the linkage between simulation requirements and simulation implementation. Object model abstracts common human spatial characteristics in the real world, their typical spatial relations and spatial interactions. Interface specification provides a specification for the functional interfaces between human and environment and services of communication and coordination. Behavior rules describe the behavior of the overall distributed simulation and their members. The above three parts are focused to discuss and design in the paper. Taking the example of disaster simulation system, the proposed specification method is finally applied to demonstrate its effectiveness and usefulness. Using HLA in studying human spatial behavior can be expected to yield progress in two aspects: in the understanding of human behavior and human interaction, and further, in the guidance on how to derive rules in decision making for human.

1. INTRODUCTION

GIS, essentially discussing how people interact with their environment, is now becoming spatial natural processes and the interaction of human with the environment in space and time. Abstraction of real world and simulation of human behavior are just two questions of how spatial reality and human experience are linked.

The study of human spatial behavior covers wide range of topics, including way finding, human migration, choice and decision make, as well as spatial cognition and environmental perception. Several difficulties in the methods used in the past research of human spatial behavior are met (Kwan, 2000): (1) Lack of effective analytical tools for representing spatial complexity of a realistic environment; (2) Limited means of spatial data acquisition and lack of suitable computational algorithms, which leading to, behavioral complexities in real world were greatly simplified or ignored. (3) Few methods are suitable to represent mobile characteristics of human, that is, how human behavior changes with space and time. To address these problems, modeling and simulation is now put forward as suitable technique and method for analysis of human spatial behavior, and to establish a human behavioral modeling framework that would satisfy the widest range of simulation applications. Further, it would provide guidelines on how to derive rules in decision making for human.

As a paradigm, High Level Architecture (HLA) (Kuhl, 1999) is meant to become a new way of implementation of modeling and simulation, and has now become IEEE standard technical framework. Therefore, the paper introduces and explores the potential of using High Level Architecture (HLA) technology based modeling and simulation as a tool for examining the complex interactions between human and environment.

2. HUMAN BEHAVIOR REQUIREMENTS (HBR)

Human spatial behavior can be described in Figure 1 as the cognitive outcome of human decision-making process based on human own characteristics, constraints from environment, situation, and their response to various policies. Requirements analysis is to understand the problems and human's needs, to identifying all possible constraints on a solution, and to reflect expected behavior to be built to solve the now-understood problem.

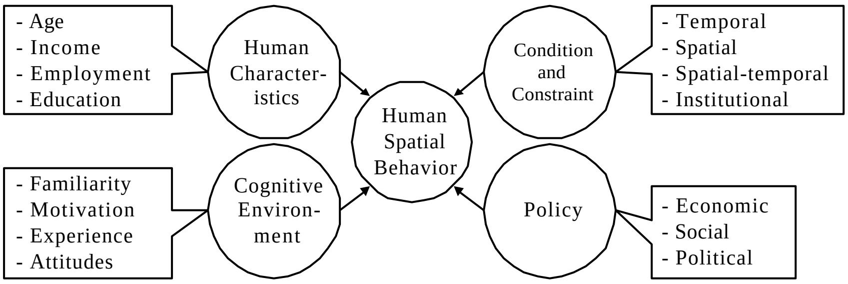


Figure 1 Human Spatial Behavior Description

In the representation of HBR (DMSO, 2000), the first step requires to collect authoritative information about the intended application domain that will comprise the simulation context. Then, the second step is the HBR representation in system and model. These requirements are necessary to ensure the model and simulation represents exactly what the user wants it to represent. The system user, with the assistance of the model and simulation developer, is responsible for developing these requirements.

HBR are comprised of (Robert, 2000): (1) User Requirements, which define the purpose of the model or simulation. They describe what the user wants the model/simulation to accomplish. (2) Derived User Requirements, detailed analysis of the User Requirement. They bound a higher-level requirement with an explicit explanation of what is actually needed. They define specified, implied, and essential tasks within the broadly, generally written User Requirement. (3) High Level Functional Requirements, detailed descriptions of what must be accounted for by the system. They identify what a system must be capable of doing to meet a Derived User Requirement and link the system functionality to the user needs. (4) Allocated Functional Requirements which is the assignment of high-level functional requirements to the components in the system. Critical decisions are made at this time that impacts the simulation design. (5) M&S Representation Requirements, the requirements that define what and how entities and processes must be represented in the system to meet the functional requirements. They specify what elements, attributes, and factors concerning High Level Functional Requirements will be represented in the simulation.

3. SIMULATION CONCEPTUAL MODELING

As the basis of logical model and physical model, conceptual model is the understanding and abstraction of real world. A simulation conceptual modeling is a simulation developer’s way of translating modeling requirements into a detailed design framework (Dale, 2000). It describes how the simulation developer understands what is to be represented by the simulation (object, attribute, interaction, action, task, process etc.) and how that representation will satisfy the requirements to which the simulation responds. The more perspicuous and precise the conceptual model, the more likely the simulation development is to both fully satisfy requirements and allow demonstration that the requirements are satisfied.

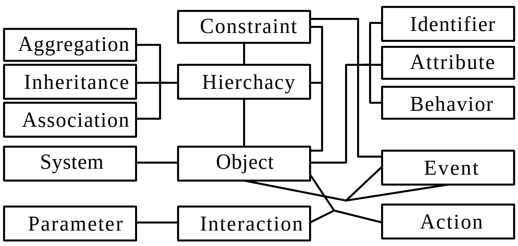


Figure 2 Basic Framework of Conceptual Model

Figure 2 describes basic framework of conceptual model. Such framework is established for us to analyze, represent and foresee structure and contents of system.

3.1 Object Model

An Object Model (U. S. DoD, 1998) is composed of a group of interrelated components specifying information about classes of objects, their attributes and their interactions. A standardized structural framework for specifying object models is an essential component, such as objects, attributes, object class structure, interaction class and parameters. Object is defined as a fundamental element of a conceptual representation that reflects the real world at levels of abstraction and resolution appropriate for data interoperability. Attribute specifies a property or characteristic of one or more objects. Object class structure provides a description of a group of objects with similar properties, common behavior, common relationships and common semantics. Interaction class records an explicit action taken by an object, that can optionally (within the bounds of the object model) be directed toward other objects, including geographical area etc. Interaction Parameter associated with an interaction which objects potentially affected by the interaction must receive in order to calculate the effects of that interaction on its current state. They are described by BNF as follows.

```
<Object>::= " (Object (Name " <<OBJ_Name>> " ) "
        [" (Type " <<OBJ_Type>> ") "]
        [" (Purpose " <<OBJ_Purpose>> ") "]
        [" (ApplicationDomain " <<OBJ_ ApplicationDomain>> ") "]
        [" (ModificaionDate" <<OBJ_ ModificaionDate >> ") "]
        [" (SponsorOrgName " <<OBJ_ SponsorOrgName>> ") "] ") ";
<Attribute>::= " (Attribute (Name " <<ATT_Name>> [<NoteRef>] " ) "
        [" (DataType " <<ATT_DataType>> [<NoteRef>] ") "]
        [" (Units " <<ATT_Units>> [<NoteRef>] ") "]
        [" (Resolution " <<ATT_Resolution>> [<NoteRef>] ") "]
        [" (Accuracy " <<ATT_Accuracy>> [<NoteRef>] ") "]
        [" (AccuracyCondition " <<ATT_AccuracyCondition>> [<NoteRef>] " ") "]
        [" (UpdateType " <<ATT_UpdateType>> [<NoteRef>] ") "]
        [" (UpdateCondition " <<ATT_UpdateCondition>> [<NoteRef>] ") "]
        [" (UpdateReflect " <<ATT_UpdateReflect>> [<NoteRef>] ") "]
        [" (Description " <<ATT_Description>> ") "]") ";
<Class>::= " (Class (ID " <<CLS_ID>> " ) "
        " (Name " <<CLS_Name>> [<NoteRef>] ")"
        [" (MOMClass " <<CLS_IsMOMClass>> ") "]
        [" (PSCabilities " <<CLS_ PSCabilities>> ") "]
        [" (Description " <<CLS_Description>> ") "]
        {<ClassComponent>}* ") ";
<Interaction>::= " (Interaction (ID " <<INT_ID>> " ) "
        "Name" <<INT_Name>> [ <Noteref> ]  ")"
        [" (MOMInteraction " <<INT_MOMInteraction>> ") "]
        [" (Description " <<INT_Descriprion>> ") "]
        [" (RouteSpace " <<INT_RouteSpace>> ") "]
        {<InteractionComponent>}* ") ";
<Parameter>::= " (Parameter (Name " <<PRM_Name>> [<NoteRef>] " ) "
        [" (DataType " <<PRM_DataType>> [<NoteRef>] ") "]
        [" (Cardinality " <<PRM_Cardinality>> [<NoteRef>] ") "]
        [" (Units " <<PRM_Units>> [<NoteRef>] ") "]
        [" (Resolution " <<PRM_Resolution>> [<NoteRef>] ") "]
        [" (Accuracy " <<ATT_Accuracy>> [<NoteRef>] ") "]
        [" (AccuracyCondition " <<ATT_AccuracyCondition>> [<NoteRef>] " ") "]
        [" (Description " <<ATT_Description>> ") "]") ";
```

3.2 Functional Model

The purpose of functional model is to model the functions, processes and activities of human, which can be supported by information technology applications, thus providing a high description of application opportunities and showing logical dependencies among opportunity areas. It includes: (1) Documents the functions or tasks for objects with behaviors; (2) Provides a means to decompose the functions into more detailed functions or processes; (3) Supports the establishment of system architecture; (4) Models the interactions among objects; (6) Defines boundaries and information of system and the interactions across the boundaries.

Every Individual/Organization has a general objective, which can be decomposed into several tasks to implement other objectives. These tasks can also be decomposed into several sub-tasks. Such decomposition continues to do until a group of basic tasks obtained which can be implemented easily. The following objective tree $P = \{P_1, P_2, \ldots, P_n\}$ is finally established shown in Figure 3.

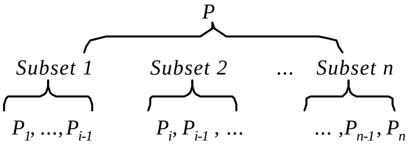


Figure 3 Objective Tree

In HLA, Interface Specification provides a specification for the functional interfaces between member of simulation and Runtime Infrastructure (RTI). The RTI provides services to members in a way that is analogous to how a distributed operating system provides to applications. These interfaces are arranged in six basic functions (U. S.

DoD, 2000): (1) Federation Management; (2) Declaration Management; (3) Object Management; (4) Ownership Management; (5) Time Management; (6) Data Distribution Management.

3.3 Behavioral Descriptions

The behavior of distributed systems is a combination of continuous time, physical phenomena and events occurring at discrete space and time coordinates. According to state and its change in a simulation, there are three different time models: continuous time model, discrete time model and discrete-event model (see Figure 4).

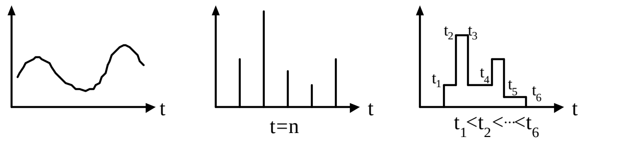


Figure 4 Three Types of Time Model

MODELING IN CONTINUOUS TIME

Many physical phenomena, such as building collapse, body movement, evolving as continuous functions of time can be described by a set of differential equations (Hu, 2000).

If define input as $\{u(t)\}$, output as $\{y(t)\}$, to describe behavior in continuous time, the following differential equation is discussed.

$$\sum_{i=0}^{n} a_i \frac{d^i y(t)}{dt^i} = \sum_{j=0}^{m} b_j \frac{d^j u(t)}{dt^j} \qquad (1)$$

Respectively, if system has random input $\{e(t)\}$, the randomicity differential equation discussed will be:

$$\sum_{i=0}^{n} a_i \frac{d^i y(t)}{dt^i} = \sum_{j=0}^{m} b_j \frac{d^j u(t)}{dt^j} + \sum_{k=0}^{l} C_k \frac{d^k e(t)}{dt^k} \qquad (2)$$

Model (1) and (2) are the important means for the study of system reflection and object behavior. The whole problem will be how to solve the two differential equations. The resolution of $\{y(t)\}$ is decided by input $\{u(t)\}$ and $\{e(t)\}$. From model (1), high rank of integral can be simplified to linear equation:

$$y(t) = f(u(t), t) \qquad (3)$$

Similarly, model (2) will be solved as the following linear equation:

$$y(t) = g(u(t), e(t), t) \quad (4)$$

MODELING IN DISCRETE TIME

In discrete time model (Ghosh, 1999), examples include train timetable and traffic lights etc, time is organized into a discrete set of monotonically increasing time steps where the choice of the duration of the time step interval reflects the desired accuracy of the model. Where the state of the process may undergo a change as a result of an external action, any change between two subsequent time steps must occur atomically within the corresponding time step interval. Regardless of whether its state incurs any changes, a process and all its parameters may be examined at every time step.

If the above continuous time model (1) or (2) is conducted on a set of discrete time steps, considering, $u_k = u(kT)$, $y_k = y(kT)$, here T is time step, a discrete model will be established by successive equation:

$$y_k = f(u_k, T) \qquad (5)$$

Similarly, randomicity model (2) can be also established by

$$y_k = g(u_k, e_k, T) \quad (6)$$

MODELING IN DISCRETE-EVENT TIME

Discrete-event systems are dynamic systems, which evolve in time by the occurrence of events at possibly irregular time intervals. Many physical phenomena, including traffic flow, decision-making etc., whose state evolved on a series of discrete points in time, which are difficult to foresee, are therefore best modeled by petri-nets or data-flow graphs. In Petri Net approaches, for example, the basic Petri net definition contains no information regarding the temporal properties of the model behavior. Timing information may be added to an ordinary Petri net to produce a Timed Petri net in the following manner: (1) Time may be associated with transitions. Each transition T has a time *d*, called the execution time of *T*. The usual interpretation of this construct is that if *T* begins firing at time *t*, the input token(s) are removed at *t* and the output token(s) appear at time $t+d$. (2) Time may be associated with places. Each place P has an associated time *d*. In this case, if a token appears in in P at time *t*, then any associated output transition may not fire until time $t+d$.

MODELING IN HYBRID SYSTEM

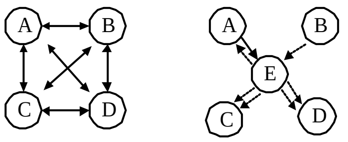
For high-fidelity simulation, hybrid modeling and simulation is required in which both continuous and discrete phenomena can be represented (Refer to Grimm).

4. DISTRIBUTED MANAGEMENT

Under communication network, every independent simulation distributing on different geographical locations is connected within distributed system. Multiple users interact with each other in a shared 3D environment and implement tasks cooperatively. Each independent simulation is defined as "node", inside, user is described in the system as an entity, or called object. State of object is controlled by user's input and update messages are reflected on other nodes by message passing.

Two communication modes in distributed system are used: peer-to-peer and client-server (Zhao, 2000)
For peer-to-peer, each node communicates with other nodes in network. For client-server, client should send messages to server first, and then server returns messages to client. It is clear that each node can self-government thoroughly in peer-to-peer. However, in client-server mode, very heavy computing can be shifted to server, so large data can be managed easily.

Message passing can be either point mode or group mode as shown in Figure 5. Node should be connected with other nodes to satisfy the requirement of point mode. While in group mode, one message is sent to partial nodes, not all nodes, and node receives only messages, what it is interested in. Group mode is now used widely in distributed applications.



(a) point mode (b) group mode

Figure 5 Message Communication modes

A Client-Server mode is used in HLA Architecture as shown in Figure 6, inside, federates interact with other federates by RTI services. So relationship between RTI and federates forms C/S structure logically. HLA integrates various federate in one system and support data transmission with multi-protocols. User Datagram Protocol (UDP) communicates a large quantity simulation data, including attribute update or interactive information.

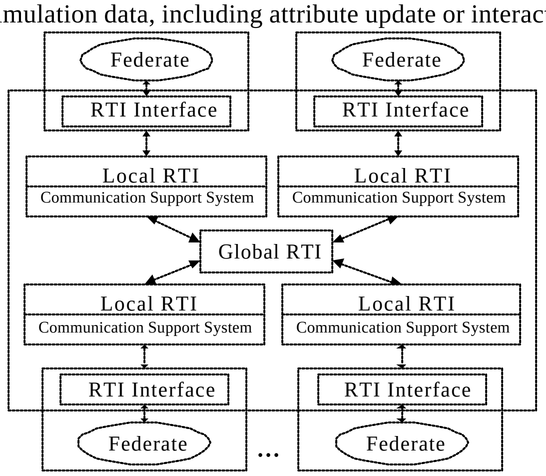


Figure 6 HLA RTI Architecture

5. EXAMPLE: A DISASTER SIMULATION SYSTEM

A Disaster Simulation System is developed using HLA based on the two objectives: (1) Analyze awareness factors (traffic accident/traffic jam, road block, building damage) when in emergency situation, which affect the behavior and movement of human in cognitive, space-time and constraints. (2) Provides public services for human decision-making, for instance, route planning decision, person evacuation, and implementation of tactics.

The system architecture of simulation design can be arranged by the following strategies. (1) The main entities in the disaster simulation system could consist of transportation, building, road, and people, which are modeled as simulator. Figure 7a shows the possible structure of a simulation for disaster systems. (2) Distributed disaster simulation models are developed using HLA. A set of interoperable simulators cooperates and communicates via HLA RTI. (3) Every simulation describes a different functional subsystem of the enter disaster system to be modeled. (4) The disaster network can be partitioned into regions (see Figure 7b), along geographical boundaries. Simulations corresponding to entities in the same region are hosted together in the same node in computer, so that they have a high-bandwidth, low latency communication among them. (5) When simulation of a new region is deployed, it will automatically connect and synchronize with the adjoining regions. (6) Intersection simulators are responsible for controlling the information flow from one region to the next.

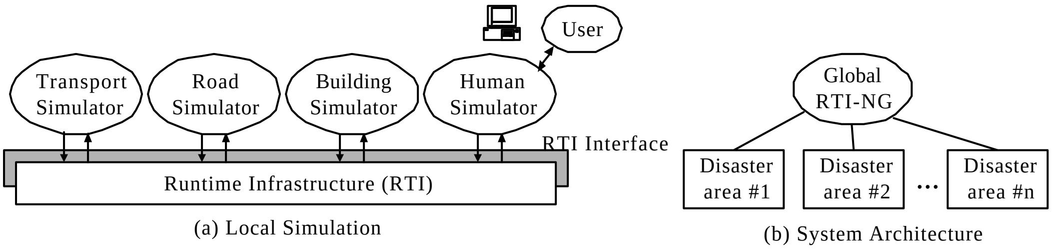


Figure 7 Example: A Disaster Simulation System

6. Conclusions

It can be summarized as follows: (1) Abstraction of real world and simulation of human behavior are two aspects of how spatial reality and human experience are linked. (2) Conceptual model, as the linkage between simulation requirements and simulation implementation, is the understanding and abstraction of real world, which can be established by object model, interface specification and behavior rules. (3) The distributed management presented in the paper implements consistency in space and time for nodes on the network. (4) As a new-generation architecture of distributed interactive simulation, HLA technology can be used as an effective and useful tool for modeling and simulation to express the complex interactions between human and environment. (5) Using HLA in studying human spatial behavior can be expected to yield progress in two aspects: in the understanding of human behavior and human interaction, and further, in the guidance on how to derive rules in decision making for human.

References

1. Mei-Po Kwan, 2000. Analysis of Human Spatial Behavior in a GIS Environment: Recent Developments and Future Prospects. J. Geographical Systems, (2), pp. 85-90

2. Frederick Kuhl, Richard Weatherly, Judith Dahmann, etc. 1999. Creating Computer Simulation Systems: An Introduction to the High Level Architecture. Prentice Hall

3. DMSO, 2000. Conceptual Modeling of Human Behavior. http://38.241.48.9/hbr/home.cfm

4. Robert J. Might, 2000. Developing Requirements for Analytical Models and Simulations. 2000 Fall SIW Papers, http://www.sisostds.org/siw/00fall/view-papers.htm

5. Dale K. Pace, 1999. Conceptual Model Descriptions. 1999 Spring SIW Papers, http://www.sisostds.org/siw/99Spring/view-papers.htm

6. U. S. Department of Defense, 1998. High-Level Architecture Object Model Template Specification Version 1.3. http://hla.dmso.mil

7. U. S. Department of Defense, 2000. HLA RTI 1.3- Next Generation Programmer's Guide Version 3.2. http://hla.dmso.mil

8. Hu Feng, Sun Guoji, 2000. A Review for Computer Simulation of Dynamic System (in Chinese). Computer Simulation, 17(1), pp. 1-7

9. Sumit Ghosh, Tony S. Lee, 1999. Modeling and Asynchronous Distributed Simulation - Analyzing Complex Systems. New York, IEEE Press.

10. Christoph Grimm, Klaus Waldschmidt, KIR- A graphed-based model for description of mixed analog/digital systems

11. Zhao Xinping, He Hongmei, 2000. Distributed Virtual Environment Overview (in Chinese). Computer Science, 27(1), pp. 1-7