Hindawi Publishing Corporation
Mathematical Problems in Engineering
Volume 2015, Article ID 915429, 27 pages
http://dx.doi.org/10.1155/2015/915429

Research Article

KD-ACP: A Software Framework for Social Computing in Emergency Management

Bin Chen, Laobing Zhang, Gang Guo, and Xiaogang Qiu

Research Center of Computational Experiments and Parallel System Technology, College of Information System and Management, National University of Defense Technology, Changsha 410073, China

Correspondence should be addressed to Bin Chen; nudtcb9372@gmail.com

Received 4 June 2014; Accepted 23 August 2014

Academic Editor: Praveen Agarwal

This paper addresses the application of a computational theory and related techniques for studying emergency management in social computing. We propose a novel software framework called KD-ACP. The framework provides a systematic and automatic platform for scientists to study the emergency management problems in three aspects: modelling the society in emergency scenario as the artificial society; investigating the emergency management problems by the repeat computational experiments; parallel execution between artificial society and the actual society managed by the decisions from computational experiments. The software framework is composed of a series of tools. These tools are categorized into three parts corresponding to "A," "C," and "P," respectively. Using H1N1 epidemic in Beijing city as the case study, the modelling and data generating of Beijing city, experiments with settings of H1N1, and intervention measures and parallel execution by situation tool are implemented by KD-ACP. The results output by the software framework shows that the emergency response decisions can be tested to find a more optimal one through the computational experiments. In the end, the advantages of the KD-ACP and the future work are summarized in the conclusion.

1. Introduction

Emergency management attracts the attention of scientists from social computing because the whole process of emergent events is deeply coupled with human society and the emergency response decisions need an approach to testify their effect without the reappearance of emergent events in the society. As a new paradigm of computing and technology development, social computing helps scientists to understand and analyze individual and organizational behavior and facilitate emergency management research and application in many aspects [1].

Based on the fruitful development of computational methodology on emergency management research over the last decade, lots of work has been done to solve the problems in society domain. Both the conceptual frameworks in multiple discipline and the technological platforms developed for the domain requirements are more and more popular in the research on emergency management, especially the agent-based modelling and simulation [2]. The bottom-up technique describes the society in microview by modelling individual behavior, communications in agents, and evolution rules of agent organizations. It is worth notifying that the modelling of agent does not emphasize the intelligence of individual. Large scale, communications and the emergence phenomena are the objects of agent-based modelling and simulation. The agent oriented platforms such as Biowar [3], GASM [4], and EpiSims [5] to study emergency problems have been proposed in many fields. Biowar developed by Carnegie Mellon University is used to study the bioattacks in city with the ability of scalable agent modelling. GASM (Global-Scale Agent Model) by Epstein simulates a global H1N1 epidemic with 6.5 billion people. EpiSims from Los Alamos national lab is used to testify the intervention measures in epidemics of smallpox from United States Department of Health and Human Services.

With the help of agent-based modelling, simulation technique and the concept of artificial society [6], a novel conceptual framework based on artificial systems is introduced in the social computing. The conceptual framework called ACP (Artificial Society, Computational Experiments, Parallel Execution) approach is proposed by Wang in 2004 [7–9]. It is a novel approach in social computing to solve the problems in society domain. ACP approach is categorized into three aspects: representing and modelling society with artificial systems, analysis and evaluation by computational experiments, and control and management of real society by parallel execution. Under the instruction of conceptual framework of ACP approach, a wide spectrum of complex systems, such as transportation, medicine, finance, and environment, can be studied in the computational manner. Actually, many real-world applications using ACP approach have been developed to solve the real problems in domains. For instance, complex socioeconomic system [10] and the research framework for e-commerce system [11] are the good applications of ACP approach in the economic area. The ACP-based framework for integrative medicine [12] is proposed to solve the problems in medicine. An overall framework of emergency rescue decision support system of petrochemical plant [13] is proposed based on ACP theory to study environment risk accidents of petrochemical plant. Parallel BRT operation management system [14] based on ACP approach has been constructed to detect the quantity of passengers on stations real-time, traffic flow on stations or at intersections, and queuing length of vehicles on the road. A novel parallel system for Urban Rail Transportation (URT) [15] based on ACP approach is proposed to address issues on safety efficiency and reliability of the operation of URT. An artificial power system [16] is set up on the models of power systems and complex power grids to provide a feasible approach for the control and management of the modern power system.

Although a lot of work has been done on the concepts and theory framework to study problems by social computing, the following problems of computational experiments in emergency management are still not solved from the perspective of modelling for social systems theory and software framework of platform implementation.

(i) The modelling and simulation of emergency management are not given the special consideration. The representation of society focuses on the generic modelling of agent (represented by Repast [17, 18]). The description of environments is too simple to meet the requirements from research on emergency problems, such as the building size, the place related agent contact frequency.

(ii) The existing tools and platforms cannot support the design of experiment. Computational experiments cannot be done systematically and automatically.

(iii) The existing applications of ACP-based frameworks are still domain specific. A generic workflow and integrated toolkit are needed to implement ACP approach, especially in the application of emergency management.

Therefore, it is necessary to develop an ACP-based software framework for the research on emergency management. The artificial system is the projection of real world in the emergent scenario. The modelling of the system including the emergent events modelling and intervention measures modelling, the design of computational experiments considering the settings of emergency parameters, the settings of large samples experiments, and the parallel execution with loose connection of real society should be considered inside the software framework.

As a result, the purpose of this paper is to propose a software framework called KD-ACP applying the ACP-based computational theory and corresponding methods in studying emergency management problems. KD is short for the Chinese phonetic alphabets of China National University of Defense Technology. KD-ACP means the software framework is developed by National University of Defense Technology to implement ACP approach. The remainder of this paper is organized as follows. Section 2 summarizes the existing agent-based modelling and simulation platforms. Section 3 introduces ACP approach first and proposes the KD-ACP platform. Section 4 illustrates the modelling of Beijing city with KD-ACP; both the agent models and initial data are considered. Section 5 shows how to do experiments with KD-ACP using the H1N1 case study in artificial Beijing. In the end, the paper is concluded in Section 6.

2. Related Works

There have been many efforts on social computing, especially on the emergency management. Agent-based modelling and simulation are popular in the implementation of social computation. The related works are mainly categorized into SWARM-like agent-based modelling and simulation platforms and agent-based platforms for emergency management.

2.1. SWARM-Like Agent-Based Modelling and Simulation Platforms. SWARM [19] originally proposed by Santa Fe institute is widely used in many research areas such as biology, ecology, and society. The tool provides a simulation environment for simulating agent with the support of a series of class libraries. It is worth noting that SWARM is the precursor of multiagent simulation tool; it influences lots of multiagent simulation platforms such as NetLogo [20], RePast (REursive Porus Agent Simulation Toolkit), MASON [21], and SOARS (Spot Oriented Agent Role Simulator) [22].

NetLogo is a multiagent programming language and modelling environment for simulating natural and social phenomena. It is particularly well suited for modelling complex systems evolving over time. The language is easy to study and the agent-based complex systems could be built rapidly; RePast is a software framework for agent-based simulation created at the University of Chicago. An extensible simulation package makes RePast become a generic multiagent simulation platform in social science research computing; MASON designed by George Mason University is used to serve as the basis for a wide range of multiagent simulation tasks ranging from swarm robotics to machine learning to social complexity environments. The tool is a fast discrete-event multiagent simulation toolkit in Java; SOARS is designed by Tokyo Institute of Technology to describe agent activities under the roles of social and organizational structure. Decomposition of multiagent interaction is the most important characteristics in this framework.

All the SWARM-like agent-based modelling and simulation platforms provide a portable, lightweight, and easily extensible environment for simulating agents in arbitrary research areas. However, the heterogeneity in specific social computing domain is not considered. Furthermore, most of the platforms cannot well support large scale agent simulation because of the lightweight engine. The engine cannot afford the simulation of super cities like Beijing and New York which have millions of people.

2.2. Agent-Based Platforms for Emergency Management. Biowar proposed by Carnegie Mellon University simulates the impact of background diseases, bioterrorism attacks within a city. 62 diseases are modeled in this platform to simulate the outbreaks on the population's behavior. GASM (Global-Scale Agent Model) is designed to study the spreading of H1N1; 6.5 billion population is modeled with the support of official statistical data. A global H1N1 spread from Tokyo is simulated in GASM. EpiSimS proposed by Los Alamos lab simulates the spread of disease in regions such as cities, allowing for the assessment of disease prevention, intervention, and response strategies. The daily movements and interactions of synthetic individuals are represented explicitly. Burke and Epstein propose a computational model of smallpox epidemic transmission and control [23]. The agents in this model interact locally with one another in social units such as homes, workplaces, schools, and hospitals.

However, these platforms cannot provide a generic software framework to study emergency problems. Biowar only focuses on social networks; individuals are all modeled as the nodes of social networks. Agents in GASM and EpisimS are isomorphic, without considering the heterogeneity in specific domains.

To sum up, this section briefly reviews the existing multiagent simulation platforms including SWARM-like agent-based modelling and simulation platforms and agent-based platforms for emergency management. However, they cannot satisfy the requirements of simulation performance, adaptability of software framework, and heterogeneity of individuals in research of different emergency scenarios.

3. KD-ACP

KD-ACP is an integrated software framework designed and implemented based on the principle of the ACP approach shown in Figure 1.

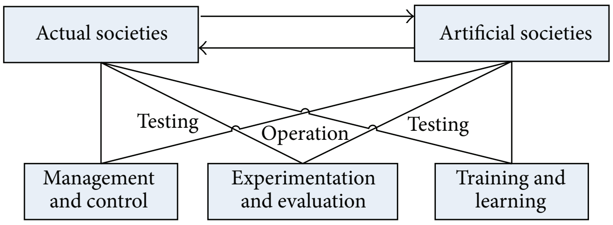


Figure 1: The parallel execution of ACP approach [8, 33].

3.1. The ACP Approach. ACP approach is a social computing-based research paradigm. It is composed of three components as its name: artificial society for A, computational experiments for C, and parallel execution for P. The basic idea of ACP approach is listed as follows.

(i) Model the complex societies involving human behavior and social organizations as artificial societies using multiagent modelling techniques in a "bottom-up" fashion. Artificial societies are regarded as a research platform to study emergency management.

(ii) Utilize innovative computing technologies to evaluate and analyze various factors in emergency management quantitatively; the computers are regarded as the experimental social laboratories for investigating emergency management problems.

(iii) Provide an effective mechanism for the control and management of complex actual social society through comparison, evaluation, and interaction with artificial society.

It is worth notifying that "P" here is not the "parallel" in "parallel simulation" but the representation of "parallel execution." The idea of parallel execution is to build the parallel scenarios by paralleling the actual societies and artificial societies. Consequently, parallel control and management of actual societies are implemented with the help of interactions between parallel scenarios. The goal of parallel execution is to find the best plans to adjust the methods of control and management based on the comparison and analysis of differences between actual and artificial societies. Artificial societies provide possible simulated results of evolutions by repeated computational experiments. The simulated results provide evidences for the adjustment plans. These plans are used in the control and management of actual societies, such as emergency management. After the application of these plans, the observations from actual societies are collected for the comparison with expectation. The differences are used to feedback to artificial societies. The new turn of comparison and analysis to find best adjustments of control and management is repeated.

The mechanism of "parallel execution" has been proved to be effective for use in networked complex traffic systems and is closely related to emerging technologies in cloud computing, social computing, and cyber-physical-social systems [24]. In order to promote the development of parallel control and management in emergency management, the artificial society is proposed in ACP approach which is the expansion of "artificial traffic systems."

Instructed by the ACP approach, KD-ACP is also composed by three components. The details of the architecture and implementation of KD-ACP are discussed below.

3.2. The Software Architecture of KD-ACP. The architecture of KD-ACP is shown in Figure 2; the software framework is composed of a series of tools. These tools are grouped into three parts to support artificial society modelling, computational experiments, and parallel execution.

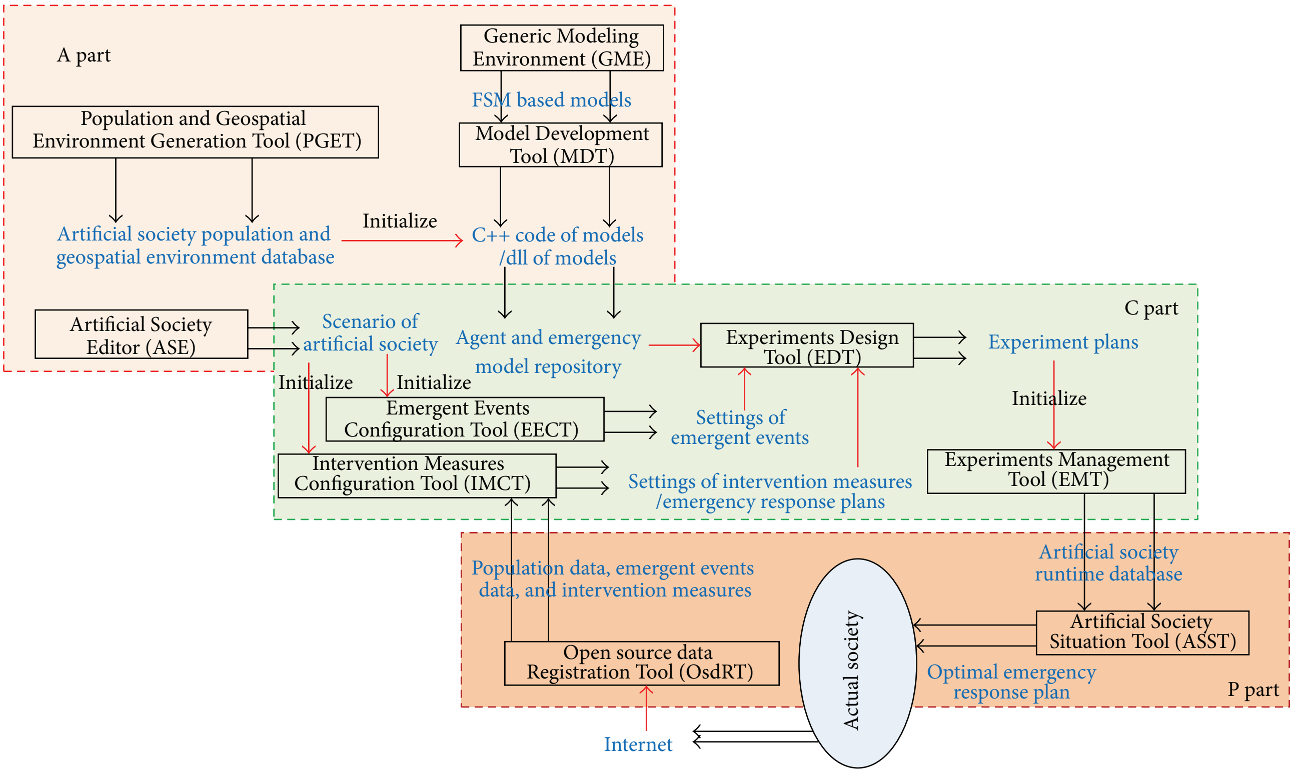

A part
Generic Modeling Environment (GME)
FSM based models
Model Development Tool (MDT)
Population and Geospatial Environment Generation Tool (PGET)
Artificial society population and geospatial environment database
Initialize
C++ code of models /dll of models
Artificial Society Editor (ASE)
Scenario of artificial society
Agent and emergency model repository
Experiments Design Tool (EDT)
Experiment plans
C part
Initialize
Initialize
Initialize
Emergent Events Configuration Tool (EECT)
Settings of emergent events
Intervention Measures Configuration Tool (IMCT)
Settings of intervention measures /emergency response plans
Experiments Management Tool (EMT)
Population data, emergent events data, and intervention measures
Actual society
Artificial society runtime database
Artificial Society Situation Tool (ASST)
Open source data Registration Tool (OsdRT)
Optimal emergency response plan
P part
Internet

Figure 2: The software architecture of KD-ACP.

In the “A” part, Generic Modeling Environment (GME) [25] and Model Development Tool (MDT) are the kernel tools in the modelling of artificial society. GME is an open source modelling tool which supports domain-specific modelling. The domains of artificial society are created by GME in our work. Models such as agent, environment, emergent event, and intervention measure are described in specific domains first in GME. With the help of model transformation, these models are all transformed to the Finite State Machine (FSM) models. Meanwhile, code generations are supported by MDT, and these models are all implemented in C++. Artificial Society Editor (ASE) is used to describe the concrete scenario of actual society, which defines the scope of models set for artificial society; Population and Geospatial Environment generation Tool (PGET) generates the population and geospatial environment data with the support of statistical data from actual society.

In the “C” part, Emergency Events Configuration Tool (EECT) initializes the models of emergent events while Intervention Measures Configuration Tool (IMCT) initializes the models of intervention measures. Experiments plans are generated by Experiments Design Tool (EDT). Based on these plans, Experiments Management Tool (EMT) is used to run and manage the computational experiments to study the emergency problems.

In the “P” part, Artificial Society Situation Tool (ASST) seemed as the monitor of running artificial society. The statistical data and situation are shown by ASST at runtime. In the meantime, the emergency response plans are made by emergency decision organizations. Parts of the influences of emergency plans are reflected on Internet. Open source data Registration Tool (OsdRT) is used to register the open source data from Internet to artificial society.

KD-ACP is developed using the Browser/Server architecture, the tools are integrated in the home page of KD-ACP as shown in Figure 3. Each tool is activated by the click on the link. For example, Artificial Society Editor is started when the link of ASE is clicked. The working environment and programming languages of tools in KD-ACP are listed in Table 1.

Moreover, the implementation of KD-ACP is mainly composed of modelling phase and computational experiments phase. It will be discussed in the next section.

3.3. The Modelling of Artificial Society in KD-ACP.

It is a critical problem to focus on the key parts of society in social computing. Based on the ACP approach, the bottom-up modelling is used to build the artificial society. As a result, modelling of artificial society is composed of three basic elements: agents, environments, and rules for interactions. However, we still meet the problem that specific features should be supported in artificial society. For example, emergent events and intervention measures are required in artificial society for emergency management. The modelling of only basic elements cannot cover the specific features in domains. Therefore, domain-specific modelling [26] is introduced to solve the problems in modelling artificial society.

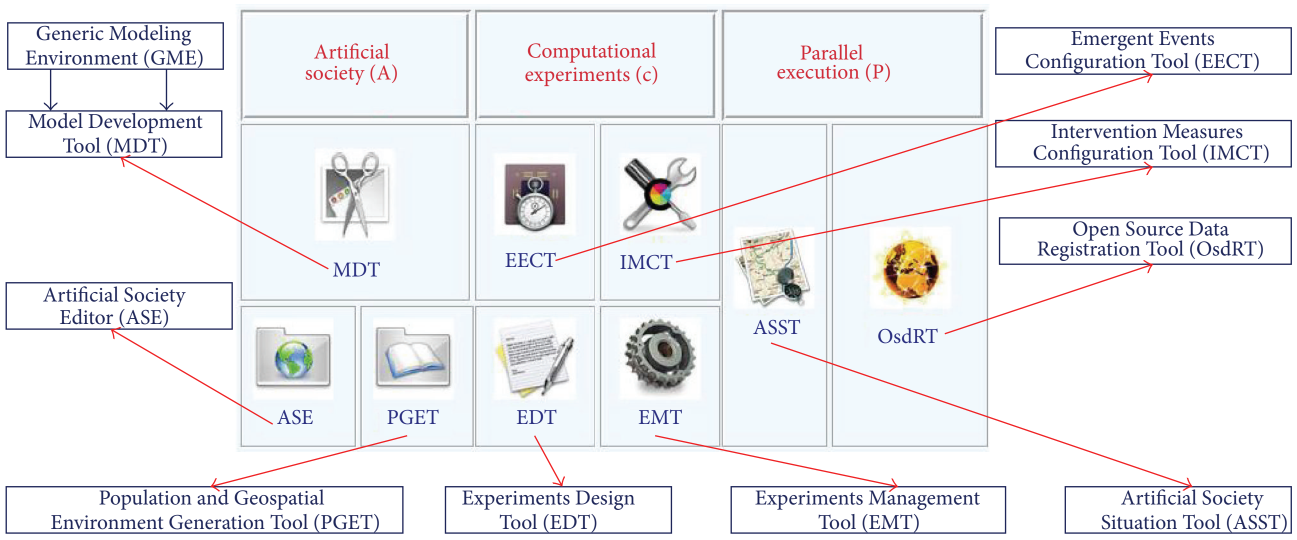

Generic Modeling Environment (GME)
Model Development Tool (MDT)
Artificial Society Editor (ASE)
Artificial society (A)
Computational experiments (c)
Parallel execution (P)
MDT
EECT
IMCT
ASST
OsdRT
ASE
PGET
EDT
EMT
Emergent Events Configuration Tool (EECT)
Intervention Measures Configuration Tool (IMCT)
Open Source Data Registration Tool (OsdRT)
Population and Geospatial Environment Generation Tool (PGET)
Experiments Design Tool (EDT)
Experiments Management Tool (EMT)
Artificial Society Situation Tool (ASST)

FIGURE 3: The implementation of KD-ACP.

TABLE 1: The working environment and programming languages of tools.

Tools	Working environment	Programming languages	Development platform	User type
Generic Modeling Environment (GME)	General computer (desktop application)	NULL	NULL	Domain experts
Model Development Tool (MDT)	General computer (desktop application)	C++	Visual Studio	Model developers
Artificial Society Editor (ASE)	General computer (desktop application)	C#	Visual Studio	Domain experts
Population and Geospatial Environment generation Tool	General computer (desktop application)	C#	Visual Studio	Domain experts
Emergency Events Configuration Tool (EECT)	General computer (with Internet Explorer, client side)	C#	ASP.NET	Domain experts
Intervention Measures Configuration Tool (IMCT)	General computer (with Internet Explorer, client side)	C#	ASP.NET	Domain experts
Experiments Management Tool (EMT)	General computer (with Internet Explorer, client side)	C#	ASP.NET	Users of computation experiment
Server of EMT	Nodes in supercomputer (Console Program, server side)	C++	Visual Studio	
Runtime Infrastructure of EMT	Nodes in supercomputer (Console Program, server side)	C++	Visual Studio	
Artificial Society Situation Tool (ASST)	General computer (desktop application)	C++	Visual Studio	Domain experts
Open Source Data Registration Tool (OsdRT)	General computer (with Internet Explorer, client side)	Java	JSP	Domain experts

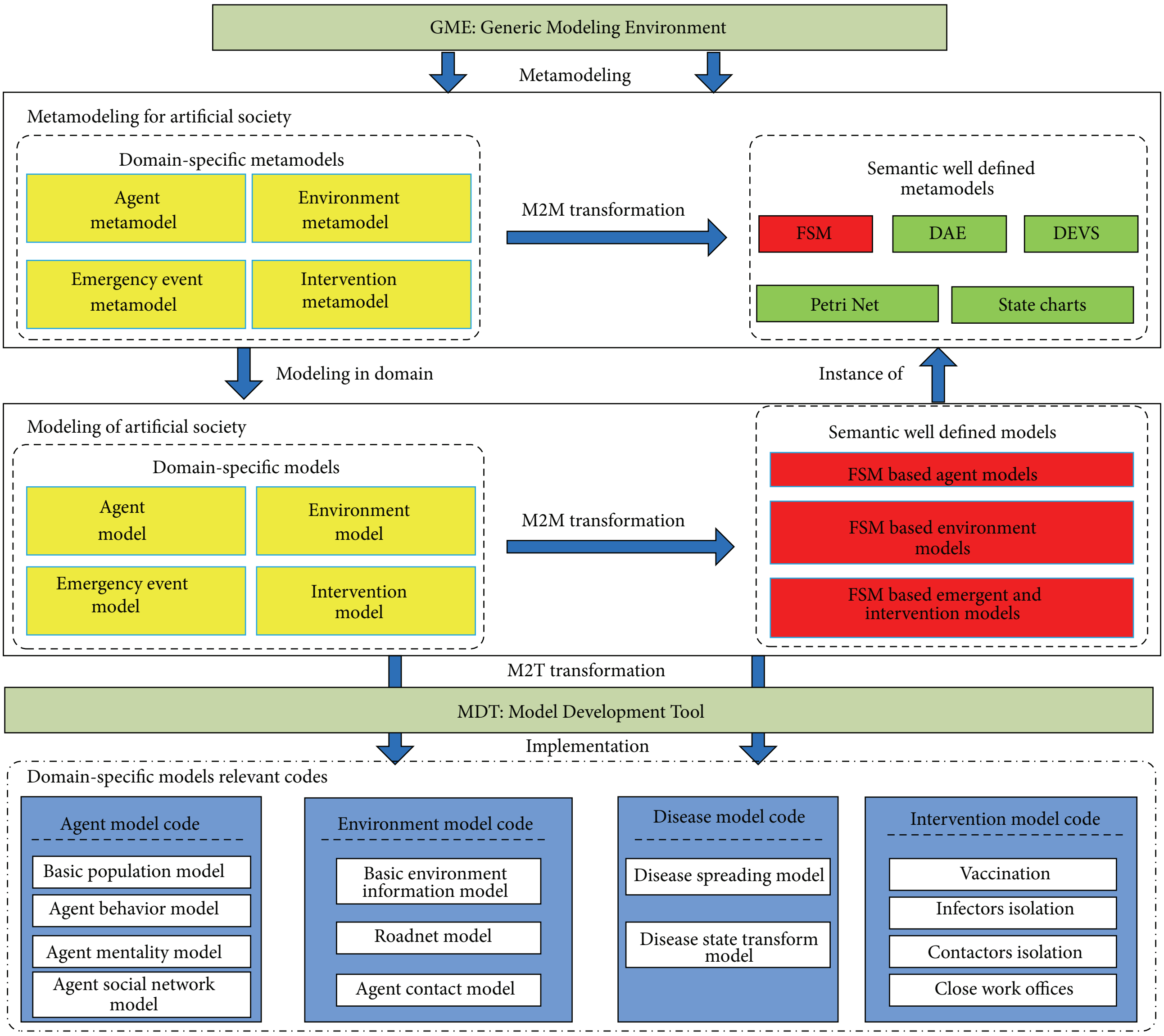
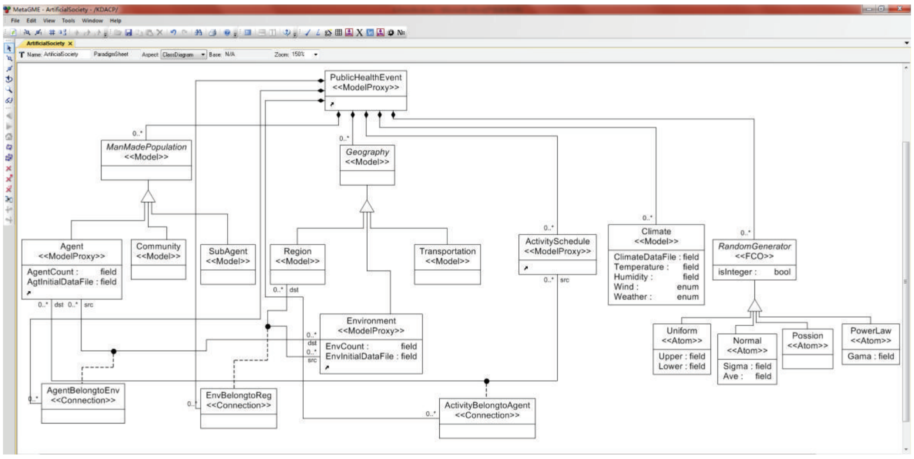


Figure 4: From metamodelling and modelling by GME to implementation by MDT of artificial society.

3.3.1. The Principle of the Modelling of Artificial Society. According to the principle of domain-specific modelling, the modelling of artificial society contains the following steps: first, metamodelling the basic elements of artificial society; second, modelling the specific features in domain of emergency management; third, implementing the models of artificial society in codes. The whole process is illustrated in Figure 4. The first and second steps are implemented in GME while the third step is implemented in MDT.

The first step is metamodelling, which mainly focuses on constructing the metamodels of artificial society. Metamodelling tries to study the common patterns of artificial society. The outputs of metamodelling are metamodels, which represents the abstraction of the whole system. The basic elements of artificial society are described in metamodel. The process of metamodelling is divided into four phases. The first is the construction of the domain-specific metamodels. As shown in Figure 4, agent metamodel, environment metamodel, emergent event metamodel, and intervention metamodel compose the metamodel of artificial society. The second is the construction of the metamodels described by typical modelling formalisms such as FSM, DAE, DEVS, and Petri Net [27]. These formalisms are all semantically well defined. The third is the definition of the model transformation from domain-specific metamodels to metamodels of typical modelling formalisms. The transformation standardizes the metamodels of artificial society by typical modelling specifications. The fourth is the definition of the transformation templates from metamodels to code framework. The templates list the basic abstract interfaces of metamodels of artificial society. These abstract interfaces are implemented in the specific-domain modelling and code generations.

The second step is modelling; the models of artificial society such as agent model, environment model, emergent



MetaGME - ArtificialSociety - /KDACP/

Name: ArtificialSociety ParadigmSheet Aspect: ClassDiagram Base: N/A Zoom: 150%

Figure 5: The metamodels of artificial society in GME.

event model, and intervention model are built. Actually, the models are the instantiation of metamodels in the last step. Different from the general modelling environment like UML [28], the domain-specific modelling provides a familiar modelling environment for the domain experts in artificial society. For example, emergency response experts only concern emergent event model and intervention model inherited from metamodels. After constructing the domain-specific models based on domain-specific metamodels, domain users execute the model transformation defined in the first step. All the models of artificial society are transformed into FSM models. As a result, the models are implemented in this unified modelling formalism (FSM). The model transformation makes the simulation of the models possible.

The third step is the generation of executable codes of models. The executable code framework is generated by mapping template from metamodels to code framework defined in the first step. Moreover, domain developers also add necessary codes to the framework to integrate the dynamic semantics of the models. The code framework outputs the dynamic link libraries by compiling. The dynamic link libraries are loaded in the large scale artificial society runtime infrastructure in computational experiments.

3.3.2. The Metamodelling and Modelling of Artificial Society by GME. GME is used to build metamodels and models in our work. As mentioned before, the abstraction and common patterns of society are represented in metamodels. According to the bottom-up modelling style, metamodels of agent, environment, and communications are described in GME. Figure 5 shows part of metamodels of artificial society. The features of an agent metamodel are extracted from the census figures and statistical data. Environment metamodel simulates the geospatial places for the behaviors of agents. The metamodel of communications among agents is modeled to simulate the interactions such as infection in epidemics and rumor propagation in public opinion formation events. It is worth notifying that the metamodel of communications includes both the emergent event metamodel and intervention metamodel.

From the perspective of modelling, the details from specific domains are considered in the models by the instantiation from metamodels of artificial society. For example, social relationships based on complex networks are added in agent model to support the communications. Agent activity is also used to quantify the agent activity under different scenario. Environment models are linked with the help of transportation services; subways and roads are modeled while the path search is encapsulated in the services. Emergent event model and intervention model are also the domain-specific models. The modelling of artificial Beijing in GME will be discussed in detail in next section.

3.3.3. The Implementation of Models of Artificial Society by MDT. As mentioned before, MDT is used to implement models such as agent, environment, emergent event, and intervention. According to the template of code framework, the implementations of models are generated by MDT. The implementations are classified into two categories: FSM models and services. FSM models such as agents and environments are built under the specification of Finite State Machine (FSM) [29] in MDT, while all the services such as transportation are encapsulated under the Public Service Standard. This standard provides a generic interface specification for modelers to encapsulate public common services in artificial society. FSM models like agent are built statistically from the quantitatively analyzed characteristics, such as demographic attributes, social behaviors, emergency behaviors, and social networks. Social behaviors describe the daily behaviors of individuals while emergency behaviors describe the individual behaviors in emergent events. For example, infected individuals are all isolated in hospital in SARS. Isolation is modeled as a typical emergency behavior in our work. Correspondingly, services are used to simulate the macroactual society. Take transportation service for instance; the path search is needed by almost every agent during moving from spot to spot.

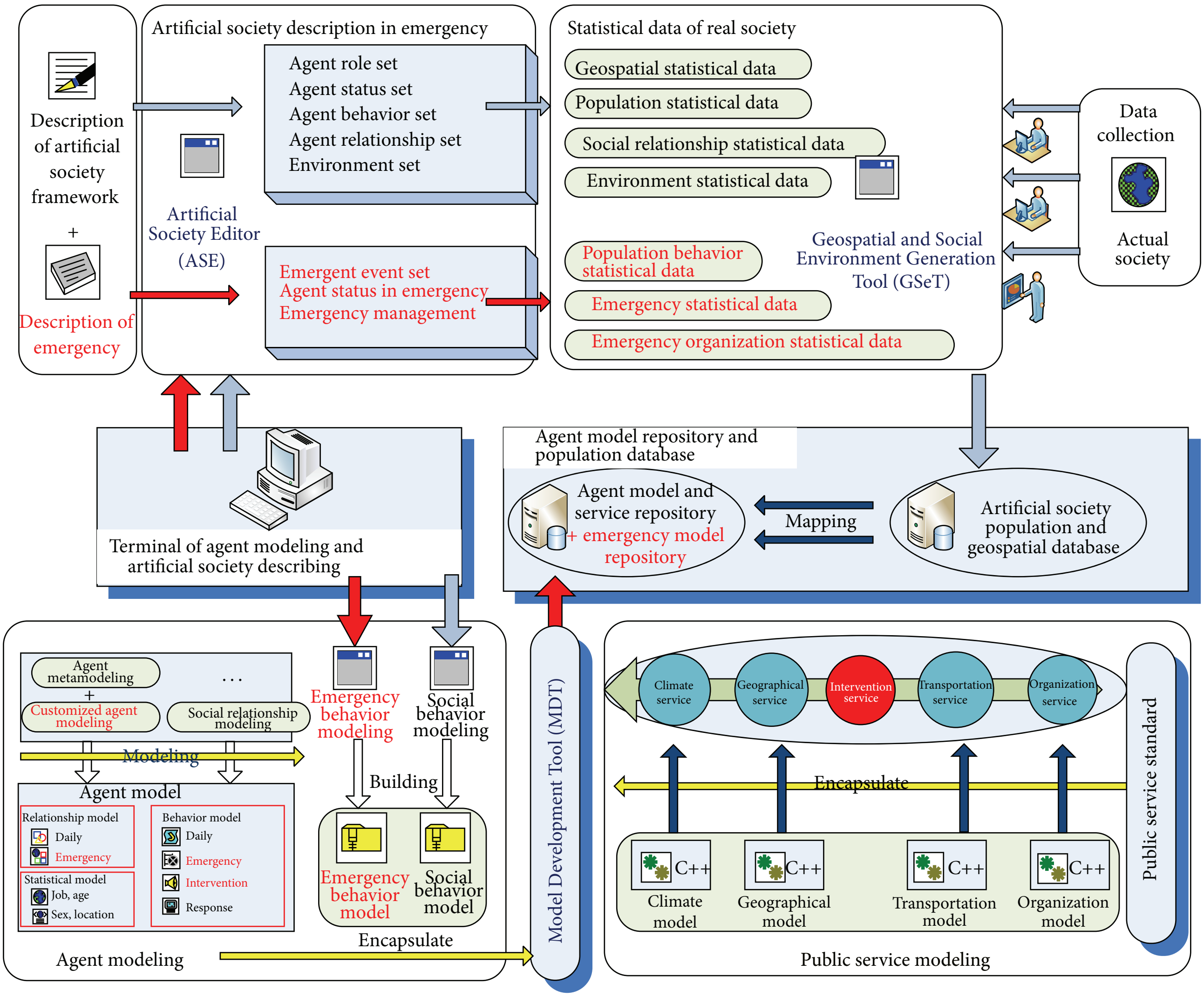

Description of artificial society framework
+
Description of emergency
Artificial society description in emergency
Artificial Society Editor (ASE)
Agent role set
Agent status set
Agent behavior set
Agent relationship set
Environment set
Emergent event set
Agent status in emergency
Emergency management
Statistical data of real society
Geospatial statistical data
Population statistical data
Social relationship statistical data
Environment statistical data
Population behavior statistical data
Emergency statistical data
Emergency organization statistical data
Geospatial and Social Environment Generation Tool (GSeT)
Data collection
Actual society
Terminal of agent modeling and artificial society describing
Agent model repository and population database
Agent model and service repository + emergency model repository
Mapping
Artificial society population and geospatial database
Agent metamodeling
+
Customized agent modeling
Social relationship modeling
. . .
Modeling
Agent model
Relationship model
Daily
Emergency
Statistical model
Job, age
Sex, location
Behavior model
Daily
Emergency
Intervention
Response
Emergency behavior modeling
Social behavior modeling
Building
Emergency behavior model
Social behavior model
Encapsulate
Agent modeling
Model Development Tool (MDT)
Climate service
Geographical service
Intervention service
Transportation service
Organization service
Encapsulate
C++
Climate model
C++
Geographical model
C++
Transportation model
C++
Organization model
Public service standard
Public service modeling

Figure 6: The editing and initialization of KD-ACP.

MDT provides domain experts with a tool to obtain the code implementations of models. With the help of compile environment like visual studio, MDT also supports the further programming development of the specific domain details which cannot be described in modelling step.

Both the FSM models and services are developed by the MDT first and then stored in the agent model and service repository. The repository manages the models according to the requirements from emergency problems and provides the models for EDT to make the experiment plans.

3.4. The Editing and Initialization of Artificial Society by ASE and PGET.

As shown in Figure 6, ASE is used to edit the scenarios of artificial society within emergent events. The editing is composed of two parts: (1) the statistical information of artificial society in daily life, such as the roles of agents, the relationships of agents, and the types of environments, and (2) the statistical information of artificial society in emergency, including the statistical data of emergent events, the emergency organization, and emergency related behaviors of agents.

According to the requirements of the editing, these statistical data are collected from actual society manually by the domain experts. Based on these statistical data, PGET generates the artificial society population and geospatial environment database. The database supports the instantiation of artificial society at individual level. For example, the attributes such as age and gender of each agent can be found in the database. With the support of the database, FSM models, and service repository discussed before, it is sufficient for domain experts to study the emergency problems by computational experiments.

3.5. The Computational Experiments and Parallel Execution in KD-ACP.

The tools of "C" part and "P" part in KD-ACP are used to support the process of computational experiments and parallel execution. The working process is shown in Figure 7; EECT and IMCT are both the starting and ending

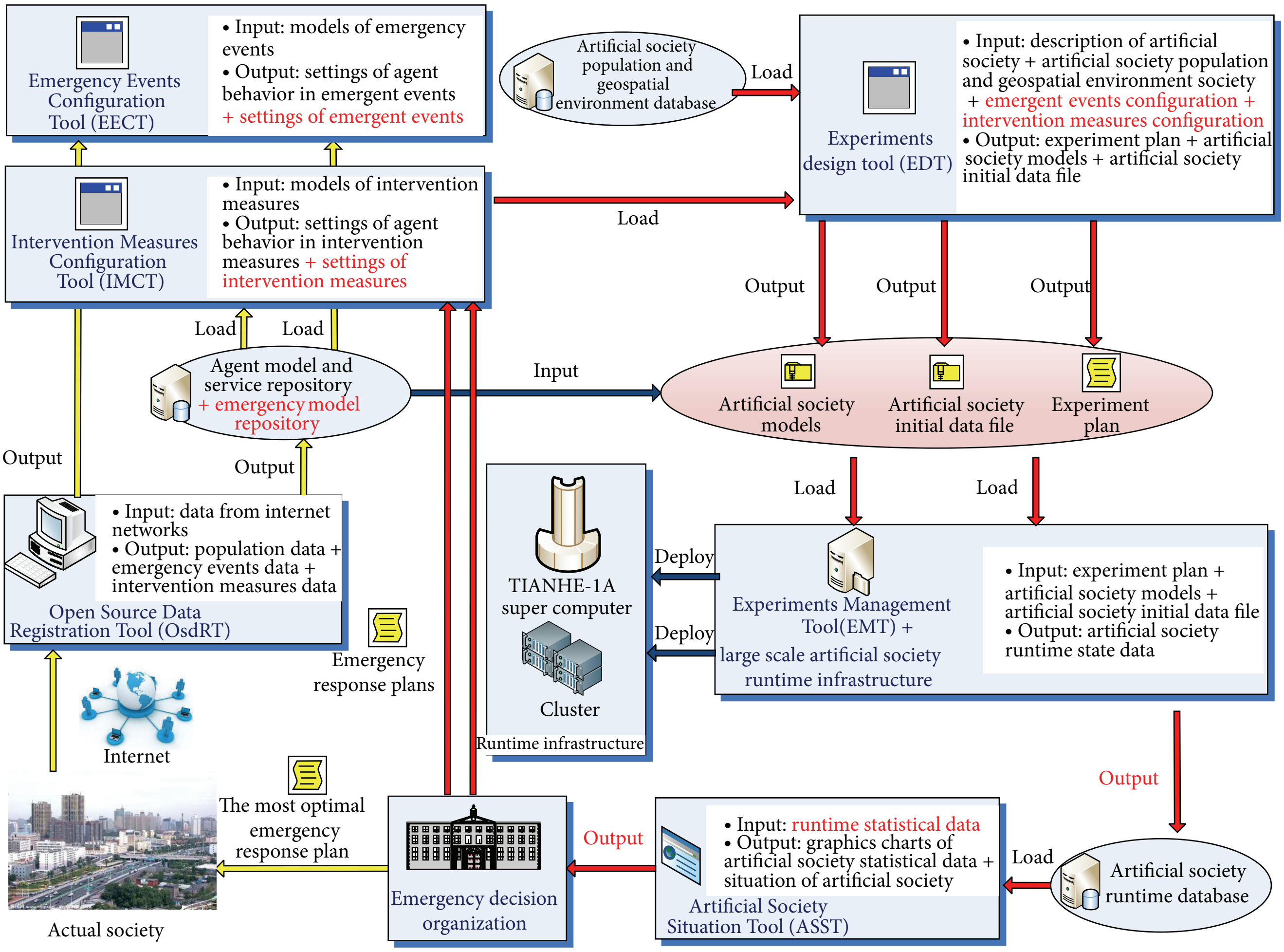

Emergency Events Configuration Tool (EECT)
• Input: models of emergency events
• Output: settings of agent behavior in emergent events + settings of emergent events
Artificial society population and geospatial environment database
Load
Experiments design tool (EDT)
• Input: description of artificial society + artificial society population and geospatial environment society + emergent events configuration + intervention measures configuration
• Output: experiment plan + artificial society models + artificial society initial data file
Intervention Measures Configuration Tool (IMCT)
• Input: models of intervention measures
• Output: settings of agent behavior in intervention measures + settings of intervention measures
Load
Output
Output
Output
Load
Load
Agent model and service repository + emergency model repository
Input
Artificial society models
Artificial society initial data file
Experiment plan
Output
Output
Load
Load
• Input: data from internet networks
• Output: population data + emergency events data + intervention measures data
Open Source Data Registration Tool (OsdRT)
TIANHE-1A super computer
Deploy
Deploy
Cluster
Runtime infrastructure
Experiments Management Tool(EMT) + large scale artificial society runtime infrastructure
• Input: experiment plan + artificial society models + artificial society initial data file
• Output: artificial society runtime state data
Emergency response plans
Internet
Output
The most optimal emergency response plan
Output
• Input: runtime statistical data
• Output: graphics charts of artificial society statistical data + situation of artificial society
Load
Artificial society runtime database
Artificial Society Situation Tool (ASST)
Emergency decision organization
Actual society

Figure 7: The computational experiments and parallel execution of KD-ACP.

point. The emergent events and intervention measures are configured by EECT and IMCT, respectively. The configurations of emergent events are used to simulate both the real emergencies like SARS and H1N1 and the supposed emergencies for experiments. Similarly, the configurations of intervention measures are also used to reproduce the real one and simulate the supposed one. The repeat of the emergency is used to verify the models while the supposed configurations are used to obtain the optimized decision plan to the response of the possible emergencies.

With the input of artificial society model and service repository, artificial society population and geospatial environment database, and the configurations discussed before, EDT generates the experiment plans to meet the requirements of research on emergency management. The output of EDT includes artificial society models, artificial society initial data files, and experiment plan. The models are downloaded from the repository while the data file is the collection of data from the database to initialize the models. When the models and data files are ready, EMT loads the experiment plan and deploys the models and data to the cluster or TIANHE-1A supercomputer [30] which was the world's fastest supercomputer built by National University of Defense Technology (NUDT) in China in 2010. According to the plan, the experiments are done repeatedly on the large scale artificial society runtime infrastructure [31, 32] by the multisample settings. The work process is the implementation of computational experiments in ACP approach.

Traditionally, emergency response plans are made by emergency management theories and experiences. The only way to test the effective of plans is the feedback results of real world. ACP approach provides a novel method to support emergency response plan making by parallel execution. As shown in Figure 7, the work process of KD-ACP is composed of two loops. The inner loop composed by red arrows describes the process of computational experiments while the outer loop of yellow arrows illustrates the process of parallel execution. During the runtime of computational experiments, the statistical data of artificial society is collected and stored in the artificial society runtime database. Based on the database, ASST outputs the customized situation of running artificial society by graphics charts and situation maps. The information is sent to the organizations of emergency decision to support making the emergency response plans. With the help of computational experiments loop, these plans are simulated repeatedly to find the most optimal one.

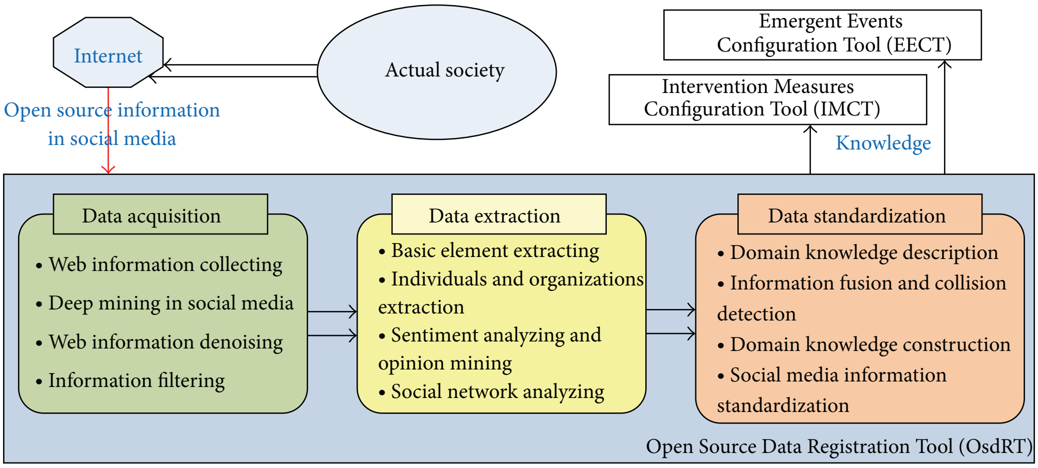

Internet
Actual society
Emergent Events Configuration Tool (EECT)
Intervention Measures Configuration Tool (IMCT)
Open source information in social media
Knowledge
Data acquisition
• Web information collecting
• Deep mining in social media
• Web information denoising
• Information filtering
Data extraction
• Basic element extracting
• Individuals and organizations extraction
• Sentiment analyzing and opinion mining
• Social network analyzing
Data standardization
• Domain knowledge description
• Information fusion and collision detection
• Domain knowledge construction
• Social media information standardization
Open Source Data Registration Tool (OsdRT)

Figure 8: The compositions of OsdRT [34].

Moreover, the most optimal plan is used to the response of emergency in actual society. According to the idea of parallel control in [24], the feedback of actual society is partly collected from Internet networks by OsdRT. As shown in Figure 8, OsdRT is composed of three components: data acquisition, data extraction, and data standardization. Data acquisition collects, mines, and filters information from social sensing networks. Data extraction includes basic element extracting, individual and organization extracting, sentiment analyzing, and social networks analyzing. Data standardization specifies the useful knowledge and sent it to configuration tools in KD-ACP.

By processing in OsdRT, the knowledge about emergent events and intervention measures are analyzed first and registered in the EECT and IMCT. The registration updates the settings of emergent events and intervention measures. This loop composed of yellow arrows implements the parallel execution in ACP approach. The implementation of parallel control and management provides a data-driven approach that considers both the engineering and social complexity for modelling, analysis, and decision making in emergency management.

4. Modelling Beijing City with KD-ACP

4.1. How to Build the Artificial Beijing. According to the modelling of artificial society in KD-ACP discussed before, the Beijing city is modeled as follows.

To meet the requirements of emergency managements, the basic elements of artificial society are extended. As shown in Figure 9, six elements are required to simulate the city: agents, environments, transportation, activity schedule, communication, and agent activity.

4.1.1. Modelling Artificial Beijing. Figure 10 shows the main GUI of GME for the modelling of artificial modelling in public health events. Metamodels are listed in the left area of Figure 10; the list provides basic syntax elements for domain experts to model artificial society. Domain experts build models based on the knowledge of their own. Meanwhile, GME supports hierarchy for building large scale systems. The syntax symbol listed in GUI can be extended in new tab by double clicking. Take agent for example; the model of agent can be detailed by edition in another tab page of agent.

As shown in the center of Figure 10, the models of artificial Beijing consist of five parts: models of agent and environment, domain models of public health events, intervention models, controller models, and services. Agent model describes individuals in society; it is composed of basic population information, action, social relationships, activity schedule, and disease related information. Activity schedule represents individual's physical action model, focusing on the daily action of agents. Environment model includes physical entities such as buildings, playground, transportations, and agents contained in environment. Domain models of public health events are composed of the propagation model of disease, disease state transition model, and so on. Intervention models include the settings of vaccination, isolation, and so on. The models mentioned before are all FSM models. The mechanisms of these models will be detailed in next sections. Controller models and services are the public service modules; they are implemented in the development in MDT.

4.1.2. Modelling Agents and Environments. Under the specification of FSM, agent and environment models are implemented in two parts: the state space and state transitions. The state space is composed of the demographic attributes and behavior related attributes. The transitions are triggered when the conditions of states are satisfied. As shown in Figure 11(a), the action of agent is changed when the "next time" condition is satisfied in agent model while the agents list is changed when the agent arrival condition is satisfied in the environment model.

4.1.3. Modelling Activities. Agent activities come from the agent state transitions of actions such as movements and communications. The actions of agents are instructed by the activity schedule shown in Table 2. Activity schedule lists

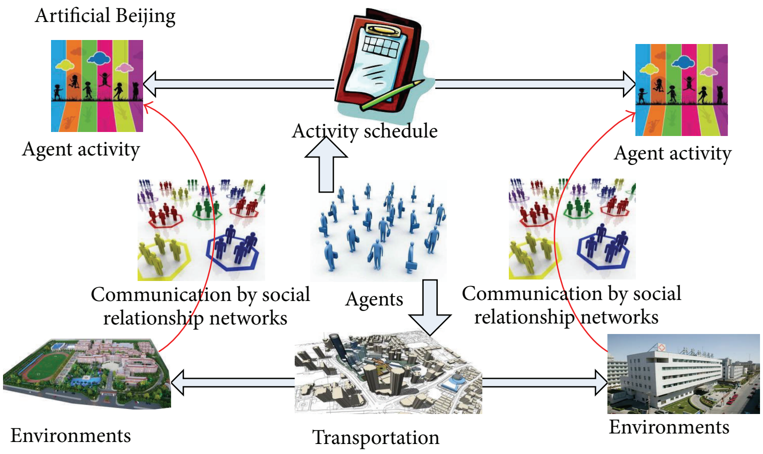


Figure 9: The basic elements of artificial society.

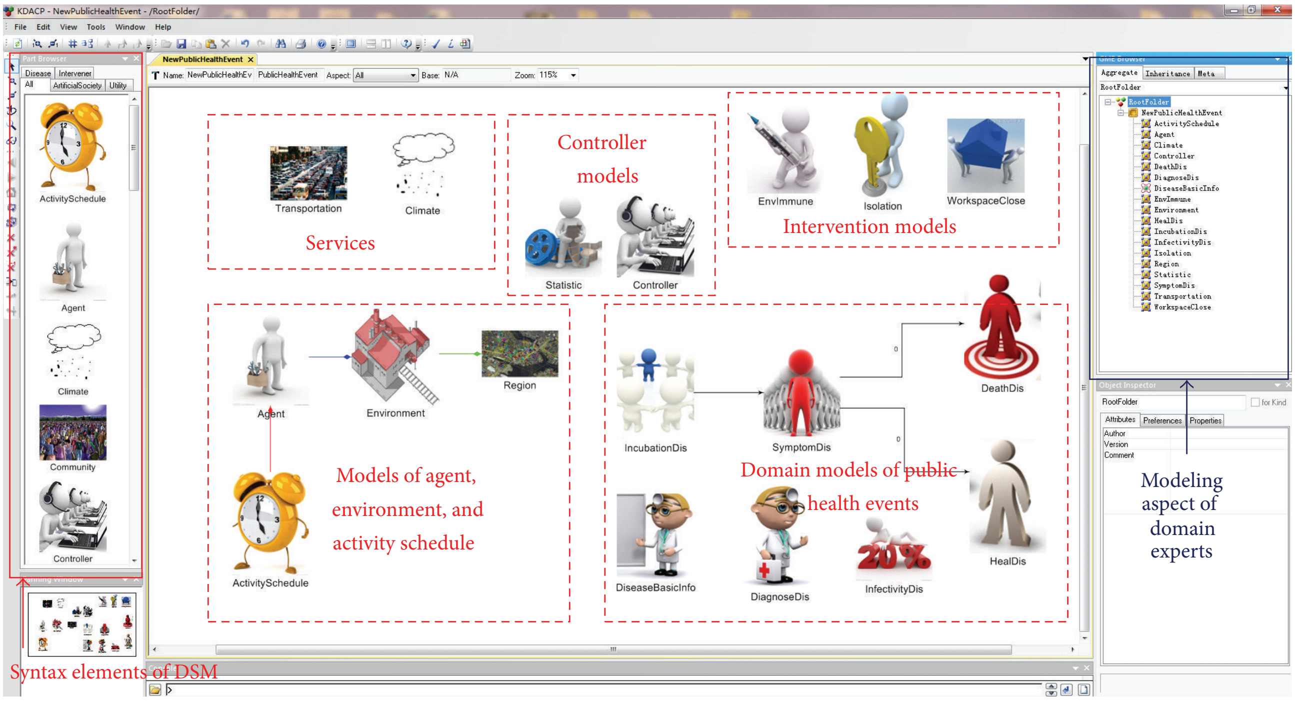
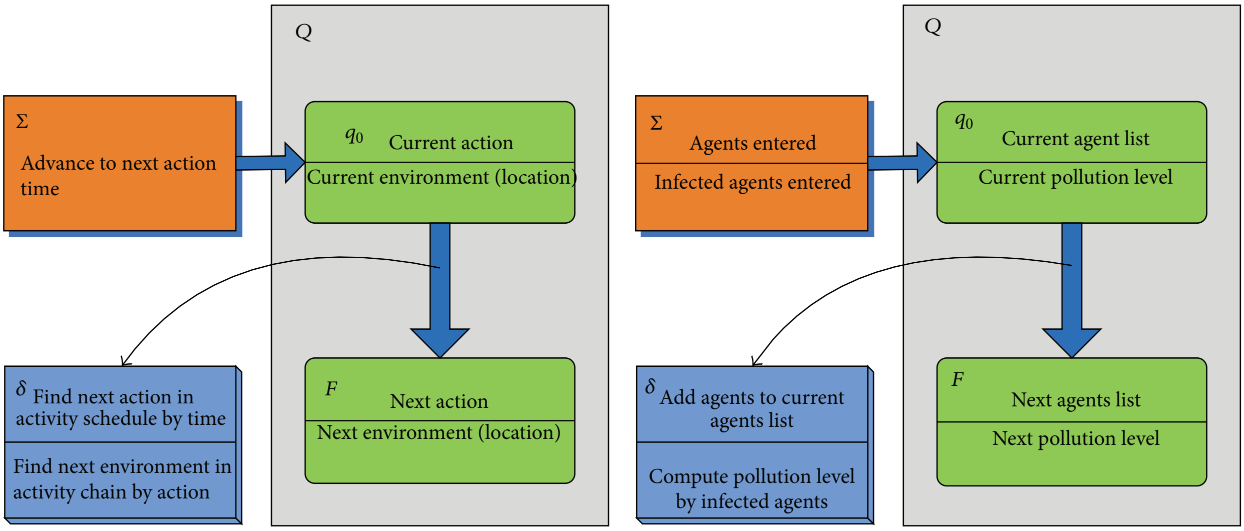


Figure 10: The models of artificial Beijing in GME.

all the actions with probability in one day for agents in both normal and emergent situation [38]. There are several types of activity schedule in artificial Beijing: student agent activity schedule, worker agent activity schedule, emergent agent schedule, and so on. For example, Table 2 gives an agent activity schedule. Upon the instruction of activity schedule, student agent changes the actions by p_i after state transitions. The p_i in the table means the action probability in the relevant period. In the duration from 08:00 to 12:00, a student agent either goes to classroom to have class or goes to library to study. The probability of class action is p_2 while the probability of study action is $1 - p_2$.

Agent behaviors are decided by the settings of activity schedule. In addition to the daily activity schedule mentioned before, emergent activity schedules are also considered in our work. Take public health events for instance; an infected agent changes schedule from a normal one to emergent one. The workflow of a susceptible agent is illustrated in Figure 12 to show the change of behaviors. After the infection, the agent is set in incubation phase. Not all the incubation agents will become symptomatic. Some of them turn back to being susceptible and some of them become symptomatic. The symptomatic agents change their activity schedule from normal to emergent. In the emergent case,


Q
Σ
Advance to next action time
q_0 Current action
Current environment (location)
δ Find next action in activity schedule by time
Find next environment in activity chain by action
F Next action
Next environment (location)
Q
Σ
Agents entered
Infected agents entered
q_0 Current agent list
Current pollution level
δ Add agents to current agents list
Compute pollution level by infected agents
F Next agents list
Next pollution level

(a) The state transitions of agent

(b) The state transitions of environment

Figure 11: The state transitions of models.

Table 2: The activity schedule of a student agent.

Duration (Δt)	Activity	Location	Probability	T_{Location} (minute)
00:00–06:00	Sleep	Dormitory	P_0 (1.00)	360
06:00–08:00	Breakfast	Dormitory/restaurant	P_1 (0.68)	120
	Sports-breakfast-travel	Playground-dormitory/restaurant	$1 - P_1$ (0.32)	
08:00–12:00	Class	Classroom	P_2 (0.77)	240
	Study	Library	$1 - P_2$ (0.23)	
12:00–14:00	Lunch	Restaurant	P_3 (0.90)	120
	Lunch-shopping	Restaurant-convenience store	$1 - P_3$ (0.10)	
14:00–18:00	Class	Classroom	P_4 (0.70)	240
	Study	Library	$1 - P_4$ (0.30)	
18:00-19:00	Dinner	Dormitory/restaurant	P_5 (0.63)	60
	Sports-shopping	Playground/convenience store	P_6 (0.25)	
	Study	Classroom/library	$1 - (P_5 + P_6)$ (0.12)	
19:00–22:00	Rest	Dormitory	P_7 (0.45)	180
	Dinner-rest	Dormitory/restaurant-dormitory	P_8 (0.20)	
	Study	Classroom/library	$1 - (P_7 + P_8)$ (0.35)	
22:00–24:00	Rest	Dormitory	P_9 (0.72)	120
	Sleep	Dormitory	$1 - P_9$ (0.18)	

infected agents go to hospital according to the treatment schedule or stay in dormitory according to isolation schedule. After the treatment in hospital or self-immunoprocess, agents become healthy and immune of disease. If the agents are treated in the hospital, they are not allowed to get out until recovered. Moreover, activity schedules are also influenced by the emergency response plans. For example, in the case of isolation in emergency response plans, the activity schedule of agents who had contacted with infected is changed. Only the locations including home and dormitory compose this emergent schedule. Likewise, based on the statistical data from emergency response plans, the additional possibilities are added into activity schedules. The behaviors of agents also change with activity schedules. In the view of ACP approach, the injection of new emergency response plans implements the parallel execution of microagent behaviors.

In another aspect, contact frequency is also another crucial element to determine the infected rate in public health event. It is important to model the contact frequency and contact time of individuals. Based on the studies on the contact behavior of human being by questionnaire survey, Edmunds found that the contact frequency of individual could be fitted approximately into a normal distribution, and the mean and the standard deviation distributions are

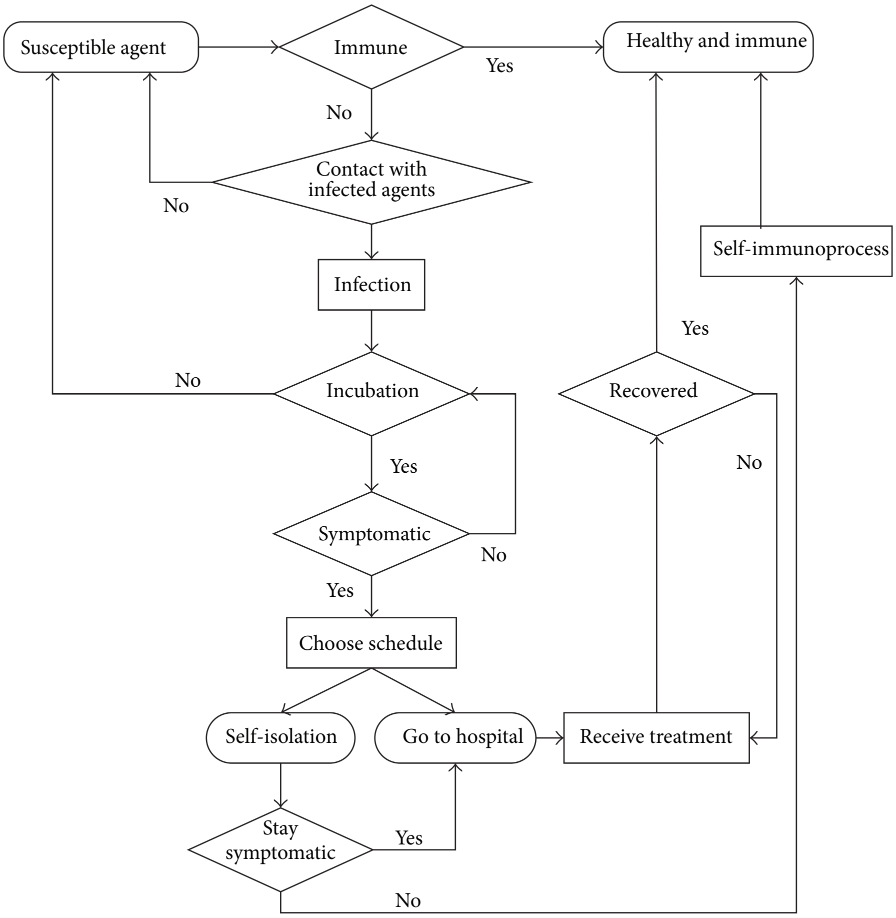


Figure 12: The workflow of a susceptible agent.

16.8 and 8.5 [39]. So it is possible to apply normal random variable to model the contact frequency. In our work, Box-Muller method [40] is used to generate the random variable of contact frequency, shown in the following:

$$F_C = \mu_F + \sigma_F\left(-2\ln\left(\gamma_1\right)\right)^{1/2}\cos\left(2\pi\gamma_2\right), \quad (1)$$

in which F_C is the random variable of contact frequency, μ_F is the mean value of normal distribution, σ_F is the standard deviation of normal distribution, and γ_1 and γ_2 are the uniform random variables distributed in the interval $[0,1]$. Based on (1) and survey data [41], the contact frequencies of agent $F_{\text{contact}}(A_i)$ are discretized as in Table 3 within the consideration of activity differences.

Similarly, the duration per contact between individuals is another key factor of AHC transmission. The duration of contact could also be modeled by the normal random variable in the following [42]:

$$T_C = \mu_T + \sigma_T\left(-2\ln\left(\gamma_1\right)\right)^{1/2}\cos\left(2\pi\gamma_2\right), \quad (2)$$

in which T_C is the random variable of duration per contact, μ_T is the mean value of normal distribution, σ_T is the standard deviation of normal distribution, and γ_1 and γ_2 are the uniform random variables distributed in the interval $[0,1]$. In addition, because the time spent in specific location $T_{\text{Location}}(A_i)$ also follows the random distribution like (3) in our work. In order to make the duration of contact $T_{\text{contact}}(A_i)$ shorter than the $T_{\text{Location}}(A_i)$, the mean and standard deviation of (2) is set up in (3) and (4) [42]:

$$\mu_T = \frac{T_{\text{Location}}}{\left(\mu_F + \sigma_F\right)}, \quad (3)$$

$$\sigma_T = \frac{T_{\text{Location}}}{\left(10\mu_F + 10\sigma_F\right)}, \quad (4)$$

in which T_{Location} is the time of an activity in a specific location set in Table 2. μ_T and σ_T are the mean and standard deviation of contact frequency. As a result, the duration of agent $T_{\text{contact}}(A_i)$ in specific locations (listed in Table 2) is also discretized in Table 3 within the consideration of activity differences.

4.1.4. Modelling Transportation. In artificial Beijing, three kinds of travel models are considered: walking inside the district, travelling by road networks, and travelling by subway networks. According to the traveling models, the agent movements are implemented by the compositions of traveling models before. So it is important to build the basic road

Table 3: The contact frequency and duration in different locations and activities by social relationships in case of student and teacher.

Social relationship	Activity	Location	$T_{\text{contact}}(A_i)$ (minute)	$F_{\text{contact}}(A_i)$
Student-student	Sleep	Dormitory	0	0
	Breakfast	Restaurant	$N(5,2)$	$N(3,2)$
	Breakfast	Dormitory	$N(5,2)$	$N(2,1)$
	Sports	Playground	$N(2,1)$	$N(5,3)$
	Class	Classroom	$N(2,1)$	$N(2,1)$
	Study	Library	$N(2,1)$	$N(2,1)$
	Lunch	Restaurant	$N(15,8)$	$N(3,2)$
	Shopping	Convenience Store	$N(20,3)$	$N(2,1)$
	Dinner	Restaurant	$N(10,8)$	$N(2,1)$
	Dinner	Dormitory	$N(15,8)$	$N(3,2)$
	Rest	Dormitory	$N(6,3)$	$N(5,2)$
Teacher-student	Class	Classroom	$N(2,1)$	$N(2,1)$

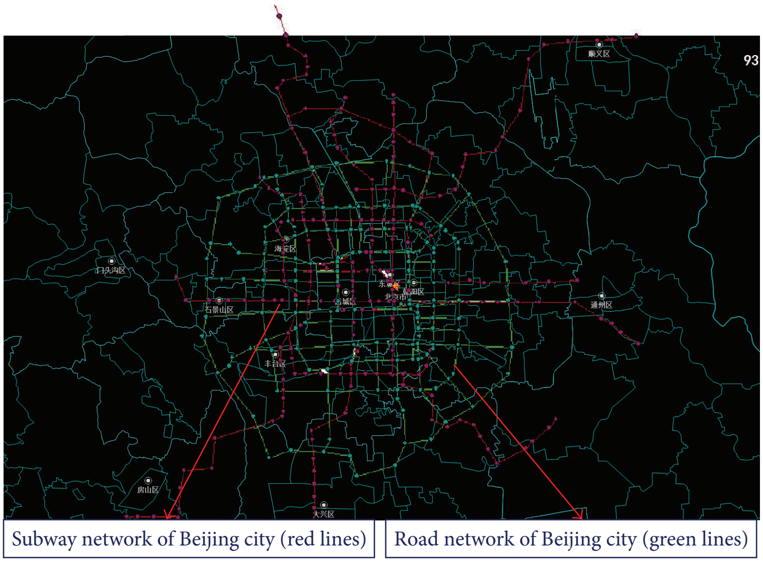

顺义区
93
海淀区
门头沟区
石景山区
西城区
东城区
北京市
朝阳区
通州区
丰台区
房山区
大兴区
Subway network of Beijing city (red lines)
Road network of Beijing city (green lines)

Figure 13: The implementation of transportation for artificial Beijing.

and subway networks. With the help of Google map, the key points of road are sampled first and the road networks are generated by the links among point. Similarly, subway networks are generated by the links between subway stations mapped from the real stations in Google map. Figure 13 shows the road and subway networks in artificial Beijing, with the road networks in green and the subway networks in red. Most of the districts in Beijing city are covered by these two networks. The algorithm of hierarchical route planning is proposed as follows.

(i) Find the next position where agent will be located in next action.

(ii) Obtain the traveling models from the statistical transportation data.

(iii) Search the path from the starting position to the nearest road entry or subway stations.

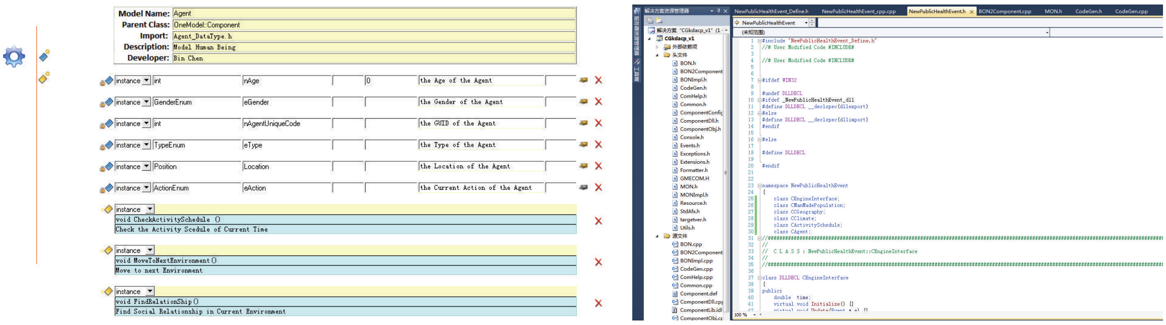
(iv) Search the optimal path from the starting station or road entry to the nearest station or road exit of target position.

(v) Search the path from the target subway station or road exit to the target position.

(vi) Attach the subway train number or the bus number including the transfer information to the path according to the timetable of subway or bus.

(vii) Connect all the paths obtained before. Generate the path from the starting position to the target position.

Based on the algorithm for each agent in artificial Beijing, the transportation is simulated during the computational experiments. It is worth noting that the activities inside the buses or train cars are considered, because the agent communication during travelling is also necessary to be simulated.



Model Name: Agent
Parent Class: OneModel::Component
Import: Agent_DataType.h
Description: Model Human Being
Developer: Bin Chen

instance	int	nAge		0	the Age of the Agent
instance	GenderEnum	eGender			the Gender of the Agent
instance	int	nAgentUniqueCode			the GUID of the Agent
instance	TypeEnum	eType			the Type of the Agent
instance	Position	Location			the Location of the Agent
instance	ActionEnum	eAction			the Current Action of the Agent

instance
void CheckActivitySchedule ()
Check the Activity Scedule of Current Time

instance
void MoveToNextEnvironment()
Move to next Environment

instance
void FindRelationShip()
Find Social Relationship in Current Environment

(a) The implementation of agent in MDT

```
#include "NewPublicHealthEvent_Define.h"
//# User Modified Code #INCLUDE#

//# User Modified Code #INCLUDE#


#ifdef WIN32

#undef DLLDECL
#ifdef _NewPublicHealthEvent_dll
#define DLLDECL __declspec(dllexport)
#else
#define DLLDECL __declspec(dllimport)
#endif

#else

#define DLLDECL

#endif


namespace NewPublicHealthEvent
{
    class CEngineInterface;
    class CManMadePopulation;
    class CCGeography;
    class CClimate;
    class CActivitySchedule;
    class CAgent;
//##########################################################################################
//
//  C L A S S : NewPublicHealthEvent::CEngineInterface
//
//##########################################################################################

class DLLDECL CEngineInterface
{
public:
    double  time;
    virtual void Initialize() {}
```

(b) The code framework of agent

Figure 14: The agent model implementation in MDT [35].

4.1.5. The Implementation of Models by MDT. All the models are developed in the MDT; Figure 14 shows the GUI of the tool. MDT is used to implement the state space and state transitions of FSM models. The code framework is shown in Figure 14(b). Using the C++ inheritance features and the techniques of dynamic link library (dll), the models are encapsulated in the dll components. The flexible composition mechanism is designed to support the models evolution in computational experiments.

4.1.6. The Description of Artificial Beijing by ASE. As mentioned in Section 3.4, ASE describes the artificial Beijing in macroview. The statistical information of Beijing city is collected by ASE. Table 4 lists part of the description of artificial Beijing generated by ASE.

4.2. How to Obtain the Data of Beijing City

4.2.1. Population and Geospatial Environment Database of Beijing City. Based on the models developed for Artificial Beijing, it is necessary to generate the initial data to do the computational experiments. Because the individual level data are not available, it is necessary to construct an individual-based population database for both accurate computational experiment and determining optimal decisions.

According to the state spaces of models discussed in Section 4.1.2, the geospatial and population database of Artificial Beijing is designed as shown in Figure 15. The kernel part of database consists of the tables such as agent list table, geospatial environment list table, household list table, and agent distribution table. Agent list table is an individual level table used to store the data to initialize the state space of agent model such as id and gender age. Geospatial environment list table is also an individual level table which stores the data of environment attributes such as id, street id, and type. Household list table and agent distribution table store the statistical data for the data generation for agent and environment models. With the help of the geospatial and population database, the data generating of artificial Beijing is proposed in the next section.

4.2.2. Generating Geospatial and Social Environment Data for Artificial Beijing by PGET. The algorithm of generating geospatial and social environment data for artificial Beijing is used to quantify the spatial distribution of population and formalize geospatial behavior of each agent. The algorithm shown in Figure 16 is capable of generating a synthetic artificial society, which allows multiresolution statistical data, social interactive behavior, and multilayer social networks to be integrated together. The synthetic population can represent individual agents in the form of households and household members, and the synthetic population is statistically equivalent to a real population. For each household, characteristics, such as address, household size, family types and relationships, are generated. Each person is described by characteristics such as age, gender, social role, and correlated locations. The algorithm provides an effective methodology to reconstruct the computing environment in high resolution by using statistical data in low resolution, leading to better prediction and management of emergencies. The details of algorithm are illustrated in [36].

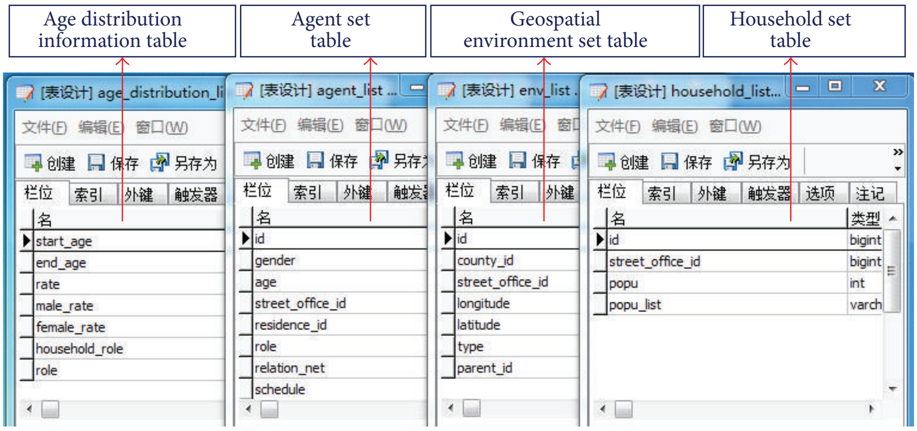
As shown in Figure 17, the implementation of the algorithm is embedded inside the GEPT. The collective input data such as the data files of population distribution is collected in the tool first; then the algorithm is activated to generate the database. The generation lasts almost twelve hours one time with the specified parameters settings. When the generation is finished, the data of artificial Beijing is obtained in the database with 19,610,000 agents and 16,000 environments.

5. H1N1 Experiments in Artificial Beijing with KD-ACP

With the support of the artificial Beijing modeled before, the computational experiments and parallel execution are also implemented by KD-ACP. The experiments are designed to find the most optimal emergency decision response plans. Epidemic in city is a typical scenario in emergency management. H1N1 epidemic in Beijing in 2009 is used to be the used case to test KD-ACP. According to Section 3.4, it is necessary to model emergency management first in order to support the computational experiments.

Table 4: The description of artificial Beijing generated by ASE.

Statistical features of artificial Beijing	Descriptions of statistical features	Data source/default values
Agent roles	{Baby, child, primary student, middle school student, college student, worker, . . ., retired}	Demographic data file of Beijing
Environment types	{Super market, store, restaurant, park, stadium, hospital, school, clinic, commercial building, dormitory building, . . ., residential building}	Demographic data file of Beijing
Demographic data	{Age distribution of population, sex distribution of population, household distribution of population, retired age distribution, . . ., children distribution}	Demographic data file of Beijing
Environment data	{Distribution of all types of environments}	Environment distribution data file of Beijing
Relationships	{Classmate, relative, college, friendship, . . ., family}	Demographic data file of Beijing
Varying parameters	Age difference of couples $N(\mu,\sigma)$	$N(0,3) \in [0, 12]$
	Minimum marriage age (male, female)	(22, 20)
	Age difference of children $N(\mu,\sigma)$	$N(1,3) \in [1, 10]$
	Enrollment rate of kindergarten α	0.8
	Enrollment rate of primary school β	1
	Enrollment rate of middle school γ	1
	Enrollment rate of college δ	0.8
	Entrance age of kindergarten α'	3
	Entrance age of primary school β'	7
	Entrance age of middle school γ'	13
	Entrance age of middle school δ'	19


Age distribution information table
Agent set table
Geospatial environment set table
Household set table
文件(F) 编辑(E) 窗口(W)
创建 保存 另存为
栏位 索引 外键 触发器 选项 注记

[表设计] age_distribution_li

名
start_age
end_age
rate
male_rate
female_rate
household_role
role

[表设计] agent_list ...

名
id
gender
age
street_office_id
residence_id
role
relation_net
schedule

[表设计] env_list .

名
id
county_id
street_office_id
longitude
latitude
type
parent_id

[表设计] household_list...

名	类型
id	bigint
street_office_id	bigint
popu	int
popu_list	varch

Figure 15: The geospatial and population database of Beijing city.

5.1. Modelling Emergency Management. The modelling of emergency management consists of two parts: the modelling of emergent events and the modelling of intervention measures. H1N1 and the intervention measures for H1N1 are modeled in our case.

5.1.1. Modelling H1N1 by MDT. H1N1 Model in agent simulates the states transitions of health status and the relevant actions. Referring to SIR (Susceptible\Infective\Recovered) [43, 44] models, agent has three health statuses: susceptible, infected, and healthy with immunity. Only susceptible agent can be infected by the contact with infected agents. After the three infected phases (incubation, being symptomatic, and recovery) of H1N1 model, agent is healthy again. It is worthy mentioning that the recovered agents can be infected iteratively when the immunity disappears gradually by time. The state transitions are illustrated in Figure 18.

According to [45], the latent period of H1N1 was in a Weibull distribution within people, from one to seven days [46, 47]. The distribution was usually centered in the range between one and three days. A Weibull random variable [40] is used to model the latent period, as described in

$$T_{\text{Lat}} = \alpha \times [-\ln(1-\gamma)]^{1/\beta} + \upsilon \, (T_{\text{Lat}} \geq \upsilon) . \tag{5}$$

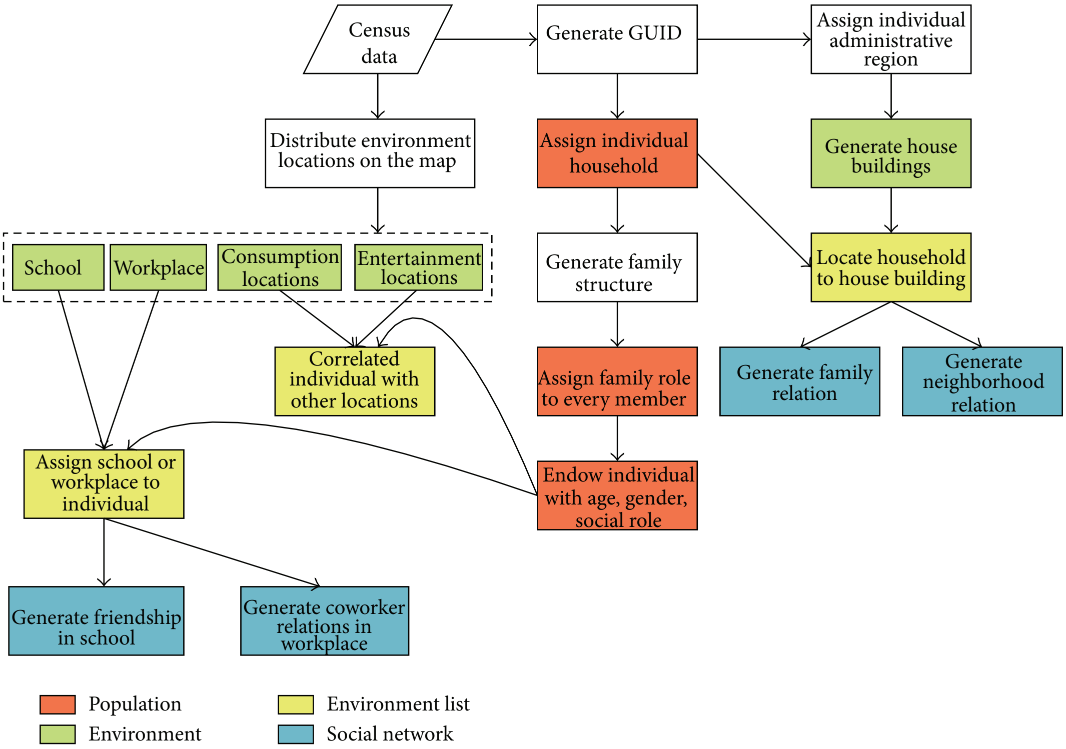
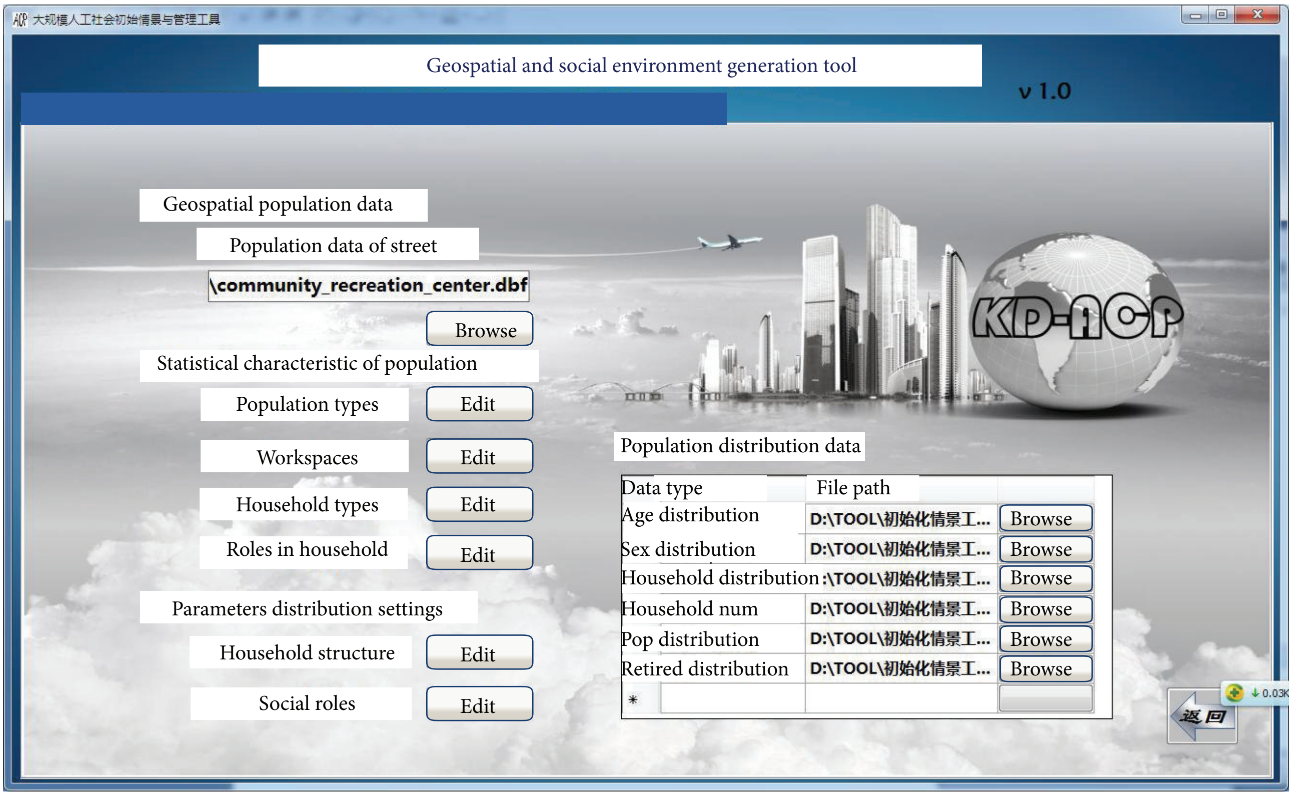

Census data
Generate GUID
Assign individual administrative region
Distribute environment locations on the map
Assign individual household
Generate house buildings
School
Workplace
Consumption locations
Entertainment locations
Generate family structure
Locate household to house building
Correlated individual with other locations
Assign family role to every member
Generate family relation
Generate neighborhood relation
Assign school or workplace to individual
Endow individual with age, gender, social role
Generate friendship in school
Generate coworker relations in workplace
🟧 Population
🟩 Environment
🟨 Environment list
🟦 Social network

Figure 16: The algorithm of generating geospatial and social environment data [36].



ACP 大规模人工社会初始情景与管理工具

Geospatial and social environment generation tool

v 1.0

Geospatial population data

Population data of street

\community_recreation_center.dbf

Browse

Statistical characteristic of population

Population types Edit

Workspaces Edit

Household types Edit

Roles in household Edit

Parameters distribution settings

Household structure Edit

Social roles Edit

Population distribution data

Data type	File path	
Age distribution	D:\TOOL\初始化情景工...	Browse
Sex distribution	D:\TOOL\初始化情景工...	Browse
Household distribution	:\TOOL\初始化情景工...	Browse
Household num	D:\TOOL\初始化情景工...	Browse
Pop distribution	D:\TOOL\初始化情景工...	Browse
Retired distribution	D:\TOOL\初始化情景工...	Browse
*		

Figure 17: The GUI of Population and Geospatial Environment generation Tool (PGET).

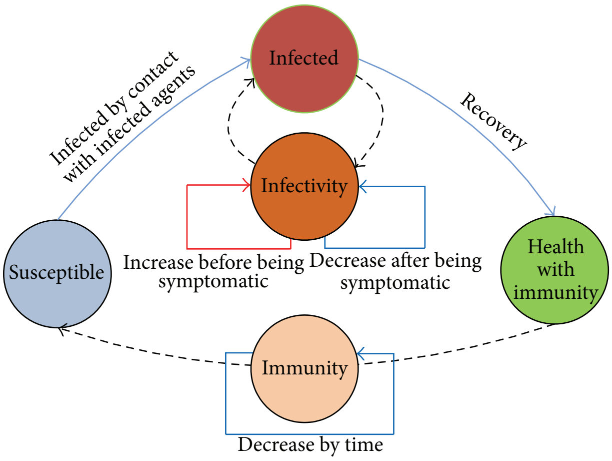

Infected
Infected by contact with infected agents
Recovery
Infectivity
Susceptible
Increase before being symptomatic
Decrease after being symptomatic
Health with immunity
Immunity
Decrease by time

FIGURE 18: The transitions of agent health status.

T_{Lat} denotes the duration of latent period. υ, α, and β are the location parameter, scale parameter, and shape parameter of Weibull distribution, respectively. γ is a uniform random number in the range $[0, 1]$. According to statistics [47], υ, α, and β are set as 0, 1.8, and 1.21, respectively. Then the mean (standard deviation) of latent period is calculated as 1.59 days (0.58 day^2). Furthermore, infectious period is set as 7 days.

A susceptible student's probability of getting infected depended on the infectivity of infectious agents, his own immunity level, the duration of the contact action, and so on. The infectivity of an infected agent evolved with time during the infectious period. According to the statistics of infectivity in the chart with red bars in Figure 19, an infected student had the highest level of infectivity in the second day after he had the first symptom. The infectivity levels in other days in comparison with the second day are in Table 5. The day labeled as "–1" means the day before the starting of the symptomatic period.

Based on the algorithm discussed before, the H1N1 model is developed with the help of MDT. The settings of influenza model will be discussed in Section 5.2.1.

5.1.2. Modelling Intervention Measures by MDT. In correspondence, intervention measures model is used to cease the outbreak of H1N1. As discussed in [42, 48, 49], if the appropriate measures are taken when the infectious disease emerges, the transmission of the disease could be slowed down and the damage could be decreased. Therefore, it is necessary and important to design emergency intervention measures in the artificial Beijing. According to the Ministry of Health in China, the intervention policies [50] are designed to control the spread of influenza such as H1N1. The intervention measures of these policies including are interventions activated time, vaccination rate, antibiotic, delay to hospital, isolation duration, close workspace duration, and limitation of activities listed in Table 6. Similarly, the models of these intervention measures are also developed with the help of MDT. Using the reasonable data ranges listed in Table 6, intervention models can be initialized to do the computational experiments. These experiments are used to show how to restrain the H1N1 transmission. The settings of these intervention measures in IMCT are introduced in Section 5.2.2.

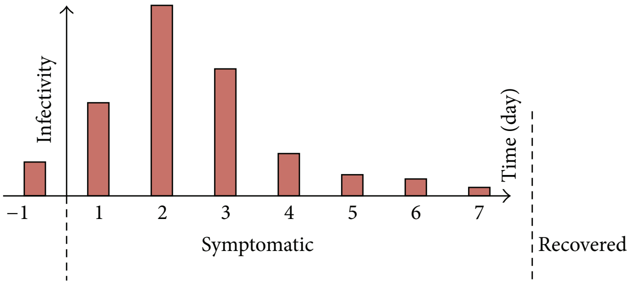


FIGURE 19: The distribution of infectivity rate of an infected agent [37].

TABLE 5: The proportion of infectivity.

Days (i)	−1	1	2	3	4	5	6	6
Proportion of infectivity	0.3	0.5	1	0.8	0.4	0.2	0.1	0.05

5.2. Experiment Settings

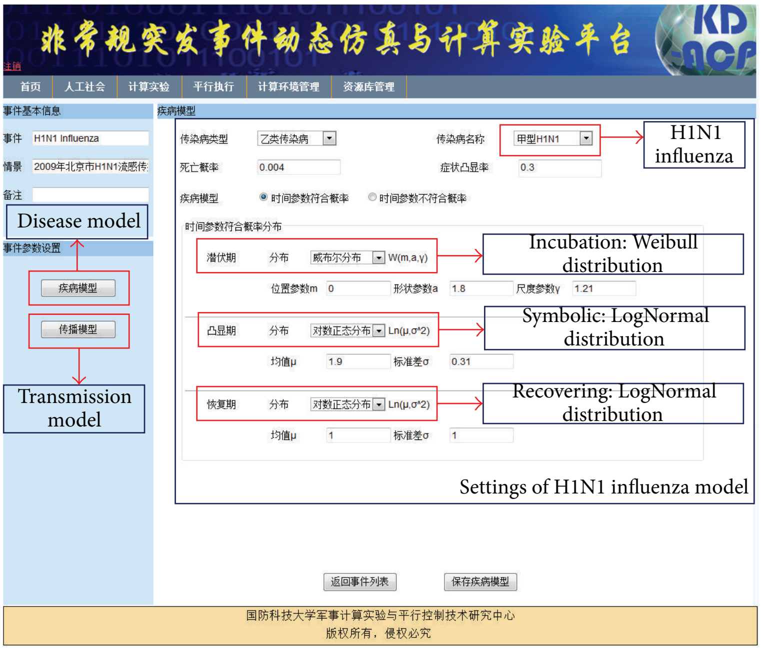
5.2.1. H1N1 Settings by EECT. EECT is designed to initialize the emergent events models such as H1N1. The settings come from the specified emergent event models. As shown in Figure 20, in the H1N1 used case, the settings are composed of the distribution of infection source, infection rate, infection period, and so on. The infection rate changes with environments and infection periods. The settings are initialized from both the research of medicine on H1N1 and the data of H1N1 outbreak in Beijing in 2009. The process of influenza in artificial Beijing is simulated to repeat the possible scenarios of H1N1 break. During the computational experiments of artificial Beijing, the H1N1 settings such as the period of being symptomatic are updated by the dynamic data registrations from OsdRT. The updating corrects the H1N1 model during the experiments. With the help of OsdRT, artificial Beijing is able to approach actual society. It is also an implementation of parallel execution.

5.2.2. Intervention Measures Settings by IMCT. In the H1N1 case, the settings of intervention measures are shown in Figure 21 in IMCT. These measures are designed under the instruction of Emergency Decision Organizations like China Disease Center (CDC). The measures such as vaccination rate, the rate of closed workspace, and the starting time can be initialized in the tool. Emergency response plans are implemented in these settings. As illustrated in Figure 7, all the possible compositions within value settings in reasonable intervals are tested to find the most optimal emergency response plan. For example, different compositions of measures such as vaccination rates and rates of workspace closed are used to design the plans of experiments. By the analysis of the results, the best composition of intervention measures can be obtained. The analysis will be discussed in case study in detail.

Table 6: The intervention measures of influenza.

Intervention measures	Values
Interventions activated Time	$T_{\text{Intervention}} \in [0, 80\text{th day}]$
Vaccination rate	$R_{\text{Vaccination}} \in [0, 1]$
Antibiotic rate	$R_{\text{Antibiotic}} \in [0, 1]$
Delay to hospital	$H_{\text{ospital}}(A_i), T_{\text{infected}} < t < T_{\text{recovered}} + T_{\text{HDelay}}, T_{\text{HDelay}} \in (0, 3\text{th day}]$
Isolation duration	$I_{\text{solation}}(A_i), T_{\text{Intervention}} < t < T_{\text{Intervention}} + T_{\text{IDuration}}, T_{\text{IDuration}} \in (0, 14\text{th day}]$
Close workspace duration	$C_{\text{lose}}(E_i), T_{\text{Intervention}} < t < T_{\text{Intervention}} + T_{\text{CWDuration}}, E_i \in \text{Workspace}, T_{\text{CWDuration}} \in (0, 7\text{th day}]$
Limitation of agent activity duration	$L_{\text{imitation}}(A_i), T_{\text{Intervention}} < t < T_{\text{Intervention}} + T_{\text{LAADuration}}, T_{\text{LAADuration}} \in (0, 7\text{th day}]$

$R_{\text{Vaccination}}$ means the vaccination rate of artificial Beijing while $R_{\text{Antibiotic}}$ means the antibiotic rate used for infected agents in hospital.
H_{ospital} (A_i) means that the infected agent A_i is sent to hospital; he or she will receive the retreatment immediately.
I_{solation} (A_i) means that the infected agent A_i is isolated at home; he or she can only contact with the other agents until isolation duration is over.
C_{lose} (E_i) means that the environment E_i typed of workspace is closed until the close workspace duration is over.
$L_{\text{imitation}}$ (A_i) means that the activities of agent A_i are limited, only the workspace and home are allowed until limitation duration is over.



事件 H1N1 Influenza
情景 2009年北京市H1N1流感传
备注

传染病类型 乙类传染病 传染病名称 甲型H1N1
死亡概率 0.004 症状凸显率 0.3
疾病模型 ● 时间参数符合概率 ○ 时间参数不符合概率

时间参数符合概率分布
潜伏期 分布 威布尔分布 W(m,a,γ)
位置参数m 0 形状参数a 1.8 尺度参数γ 1.21
凸显期 分布 对数正态分布 Ln(μ,σ^2)
均值μ 1.9 标准差σ 0.31
恢复期 分布 对数正态分布 Ln(μ,σ^2)
均值μ 1 标准差σ 1

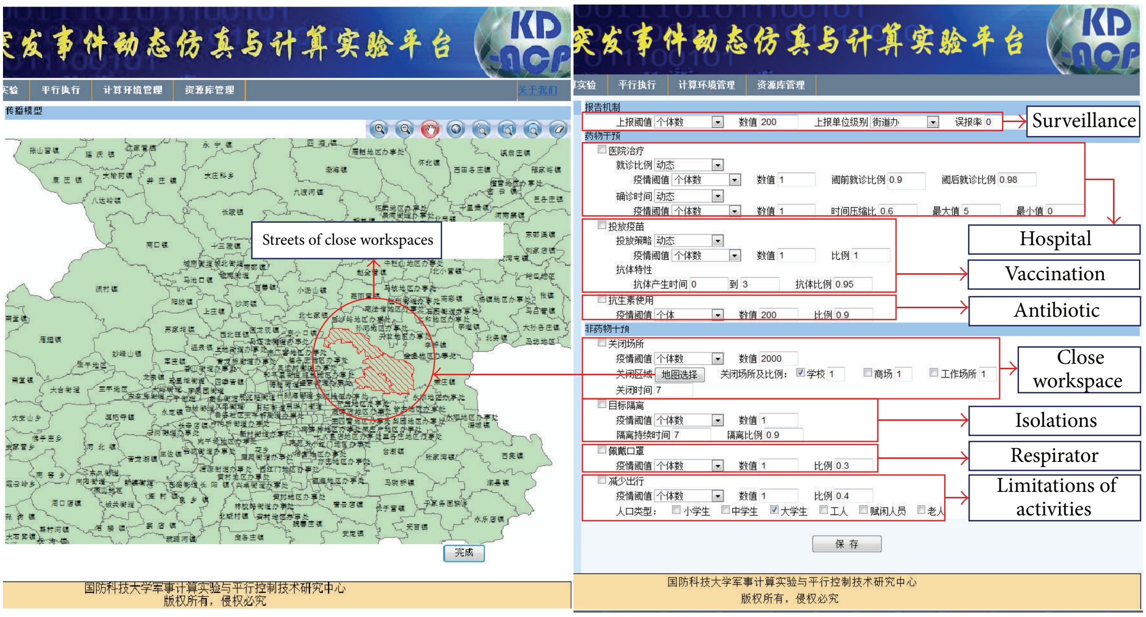
国防科技大学军事计算实验与平行控制技术研究中心
版权所有，侵权必究

Figure 20: The settings of H1N1 in EECT.

It is worth notifying that the settings of intervention measures are also influenced by OsdRT. For example, the H1N1 outbreak areas are notified by OsdRT. The intervention measures such as close workspace are used only in the outbreak areas. Therefore, the unnecessary computation of intervention measures is avoided. So the data injection by OsdRT not only makes artificial society approach actual society, but also increases the performance of computational experiments.

5.3. Experiments by EMT. When the initialization of artificial Beijing is ready, EMT is used to do the computational experiments. EMT is composed of two parts: the central controller in the central node and the residential service deployed in the computational nodes. Central controller loads the experiment plans and controls the whole experiments process; the GUI of controller is shown in Figure 22; EMT manages the computational experiments in four steps: cluster configuration, load experiment plans, models deployment, and running experiments.

5.3.1. Cluster Configuration. Before the configuration, cluster information is collected from the residential services inside the cluster. The information includes nodes amount, node names, node IPs, and CPU occupations. It is used to quantify the computing power of the cluster. The cluster configuration is based on this information. The nodes are selected for experiments. Then the LP (local process) [51] number for each node is set according to the number of the CPU kernels. After the configuration, the experiment plans are loaded to customize the models and data.


突发事件动态仿真与计算实验平台
KD-ACP
实验 | 平行执行 | 计算环境管理 | 资源库管理
关于我们
传播模型
Streets of close workspaces
完成
Surveillance
Hospital
Vaccination
Antibiotic
Close workspace
Isolations
Respirator
Limitations of activities
保 存

报告机制

上报阈值 个体数 数值 200 上报单位级别 街道办 误报率 0

药物干预

□ 医院治疗
就诊比例 动态
疫情阈值 个体数 数值 1 阈前就诊比例 0.9 阈后就诊比例 0.98
确诊时间 动态
疫情阈值 个体数 数值 1 时间压缩比 0.6 最大值 5 最小值 0

□ 投放疫苗
投放策略 动态
疫情阈值 个体数 数值 1 比例 1
抗体特性
抗体产生时间 0 到 3 抗体比例 0.95

□ 抗生素使用
疫情阈值 个体 数值 200 比例 0.9

非药物干预

□ 关闭场所
疫情阈值 个体数 数值 2000
关闭区域 地图选择 关闭场所及比例：☑ 学校 1 □ 商场 1 □ 工作场所 1
关闭时间 7

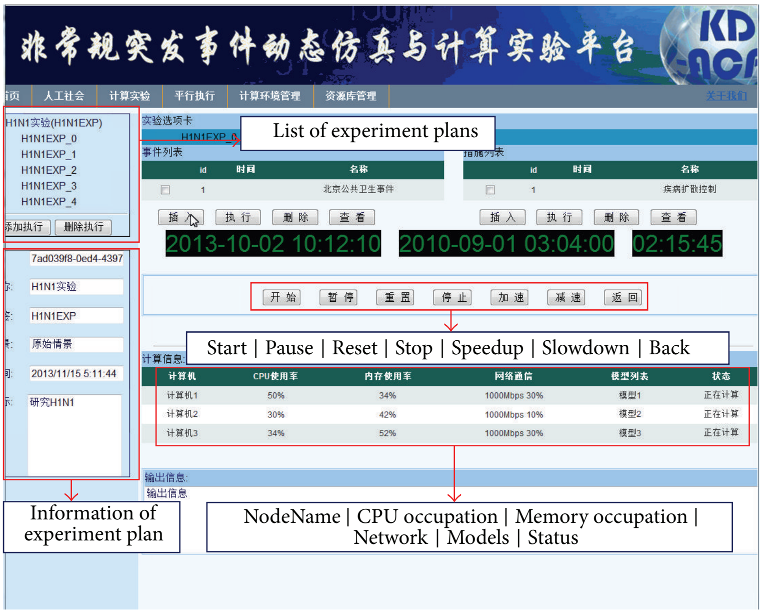
□ 目标隔离
疫情阈值 个体数 数值 1
隔离持续时间 7 隔离比例 0.9

□ 佩戴口罩
疫情阈值 个体数 数值 1 比例 0.3

□ 减少出行
疫情阈值 个体数 数值 1 比例 0.4
人口类型：□ 小学生 □ 中学生 ☑ 大学生 □ 工人 □ 赋闲人员 □ 老人

国防科技大学军事计算实验与平行控制技术研究中心
版权所有，侵权必究

Figure 21: The settings of intervention measures in IMCT.


非常规突发事件动态仿真与计算实验平台
KD-ACP
首页 | 人工社会 | 计算实验 | 平行执行 | 计算环境管理 | 资源库管理
关于我们
List of experiment plans
插 入 | 执 行 | 删 除 | 查 看
开 始 | 暂 停 | 重 置 | 停 止 | 加 速 | 减 速 | 返 回
Start | Pause | Reset | Stop | Speedup | Slowdown | Back
Information of experiment plan
NodeName | CPU occupation | Memory occupation | Network | Models | Status

H1N1实验(H1N1EXP)
- H1N1EXP_0
- H1N1EXP_1
- H1N1EXP_2
- H1N1EXP_3
- H1N1EXP_4

添加执行 删除执行

7ad039f8-0ed4-4397

H1N1实验

H1N1EXP

原始情景

2013/11/15 5:11:44

研究H1N1

实验选项卡

H1N1EXP_0

事件列表

	id	时间	名称
□	1		北京公共卫生事件

措施列表

	id	时间	名称
□	1		疾病扩散控制

2013-10-02 10:12:10 2010-09-01 03:04:00 02:15:45

计算信息:

计算机	CPU使用率	内存使用率	网络通信	模型列表	状态
计算机1	50%	34%	1000Mbps 30%	模型1	正在计算
计算机2	30%	42%	1000Mbps 10%	模型2	正在计算
计算机3	34%	52%	1000Mbps 30%	模型3	正在计算

输出信息:

输出信息

Figure 22: The deployment and running of experiments in EMT.

5.3.2. Load Experiment Plan. As discussed in Section 3.5, the description of experiment plans is composed of the information in three aspects: models and the initial data for the experiments, the deployment mapping tables from the models to the nodes, and the plan of experiment execution. When the experiment plan is loaded by the EMT, the file names and model descriptions of models and data are listed. With the support of cluster configuration, models can be mapped to the nodes in the GUI according to the computation requirements. EMT also provides the setting of experiments itself, including the running times, start time, and end time.

5.3.3. Models and Data Deployment. Models are stored in the agent population and services repository. According to the settings in experiment plan, the customized models are downloaded from the repository first; then the models such as agent model, environment model, emergent events model, and intervention measures model are integrated to be the composition models of artificial Beijing. The models and data locations in the nodes are configured in the "model path" and "data path" in the GUI of central controller. Finally the models are uploaded to the nodes in cluster with the consideration of partition. The partition of artificial society models is different from the multiagent system in [52]. Models in artificial society are divided into three levels: agents and environments, grids, and city. Agents and environments are encapsulated in grids which compose the whole city. Thus the partition of artificial society becomes the partition of grids. The parallelism degrees are found to support the partition of models: agent independence and grid independence. According to the parallelisms, agents are partitioned to LPs according to the population distribution in grids. Agent movements among grids are simulated by the movement events among grids in different LPs. As a result, a two-tier parallel architecture is proposed to support the simulation of artificial society. The architecture is detailed in [53].

5.3.4. Running of Experiments. When the preparation is ready, the experiments can be done by the central controller of EMT. The running and control commands of the experiments are grouped into three aspects: start, pause, or stop the experiments; get the running information of the nodes such as the snapshot of the desktop of node. With the help of these commands, users can run the computational experiments to do the research on the emergency management. It is worth notifying that the simulation engine is optimized in order to support the ten million agent simulation. Using GPU as coprocessor, a two-tier parallel simulation engine is designed with support of MPI and OpenCL through phased synchronization [54]. One-sided communication is used for reflection of remote simulation objects and message passing between processes. A general kernel function prototype is elaborately designed and conditionally compiled for execution on both CPU and GPU. Moreover, the optimization operations like load balance is developed under the instruction of activity-based simulation [55]. The densely populated grids are given more CPU and memory resource according to the activity predictions.

5.4. Results of H1N1 Experiments with KD-ACP. On the basis of artificial society models and the intervention measures mentioned in Section 5.1.2, a series of computational experiments are performed to study the H1N1 influenza in Beijing. In our case of artificial Beijing, 19,610,000 agents are simulated in the cluster within 48 CPU cores and 128 G memories, the nodes are connected by kilomega networks. It takes 18 hours to simulate a 250-day disease spreading.

In order to illustrate how the optimal plans are obtained by KD-ACP, the experiments are divided into four groups: the experiments of model validation, the experiments of sensitivity analysis of vaccination rate, the experiments of sensitivity analysis of isolation, and the experiments of the combination of intervention measures. These four groups show the general process of the research on emergency management in H1N1 influenza. Firstly, the models of H1N1 are validated with the support of historical data. Secondly, the traditional medical interventions of pandemic such as vaccination are analyzed to find the optimal plan. Thirdly, the traditional nonmedical interventions of pandemic such as isolation are analyzed to find the optimal plan. Finally, the combination of interventions is tested to show the combined effect.

It is worth mentioning that all the experiments in our work are performed 100 times. These experiments are initialized with different settings of interventions in IMCT. The results shown in the figures are the mean values.

5.4.1. The Experiments of Model Validation. In our artificial Beijing, the models of H1N1 influenza are built based on the existing researches. Wang and her colleagues had studied the H1N1 influenza by SEIR (Susceptible\Exposed\Infective\Recovered) models [56]. With the support of their work, we build the models of artificial Beijing in the manner of multiagents.

In Figure 23, the control group shown in red line is drawn based on the realistic statistical data collected in H1N1 influenza in Beijing in 2009. The influenza lasts more than six months, and more than 170,000 people are infected. Comparing with the control group, the simulated results without interventions are drawn in blue. It is obvious that the simulated case fits the control group well before 70 days. But the ascending of control group is slowed after 70 days, and the peak value (17,883) is much less than the simulated results (25,135). It is because the vaccination is activated at day 67 in control group according to the intervention measures used in 2009. The restrain by vaccination becomes effective eight days later. The change of infected number after vaccination is shown by red spotted line. The trend of control group is different from the simulation data after 75 days.

However, the simulated results (blue line) in computational experiments fit the control group (as shown in Figure 23) in the first 70 days when no intervention measures are executed. The key features such as effective basic reproduction number (R_0) are also almost the same in both cases. Therefore, it can be concluded that the models of artificial society built in microview are validated by the realistic statistical data in influenza in Beijing in 2009.

In this group of experiments, no interventions are executed in order to validate the models. Therefore, the simulated results of this extreme case which cannot happen today could be obtained. As shown by blue line, the number of infected agents without interventions reaches peak value at day 91 with 25,135, consisting of 8,052 in symptomatic phase and 28,950 in recovery phase. It is obvious that the total number of infected agents grows slowly at the beginning time. Then the number increases quickly when the influenza breaks, and it reaches its maximum value with 383,870 in the end. This phenomenon can be explained as in the following. At the beginning of influenza, due to the limited number of infected agents, the propagation of the disease remains slow but many agents become susceptible. When the number of infected agents increases rapidly, the spreading becomes significant. However, the number of infected agents decreases after day 91. It is because the disease spreading relies on the social networks and spatial contact networks [49]. It means that the infected agents can only infect agents either in the social relationship networks or in the spatially contacted.

5.4.2. The Experiments of Sensitivity Analysis of Vaccination Rate. According to Section 5.1.2, vaccination is a typical

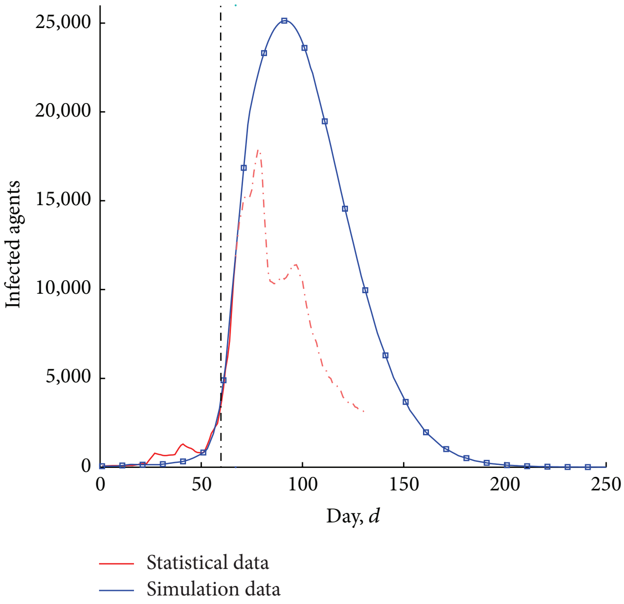


Figure 23: The results with no intervention executed, and the comparison with realistic statistical data in Beijing in 2009.

medical intervention in influenza. Traditionally, the effectiveness of vaccination can only be evaluated after the influenza, and the key settings such as vaccination rate are determined by the experience. It is always difficult for emergency management organizations to find the most optimal vaccination rate in advance. Therefore, the sensitivity of vaccination rate is analyzed in this group of experiments. The settings are listed in Table 7; average results are shown in Figure 24.

Table 7: The initial settings of experiments of vaccination.

Parameters	Value
Agent count	19,610,000
Initial infected count	40
Interventions activated time ($T_{\text{Intervention}}$)	0
Vaccination rate ($R_{\text{Vaccination}}$)	{0%, 10%, 30%, 50%, 70%}

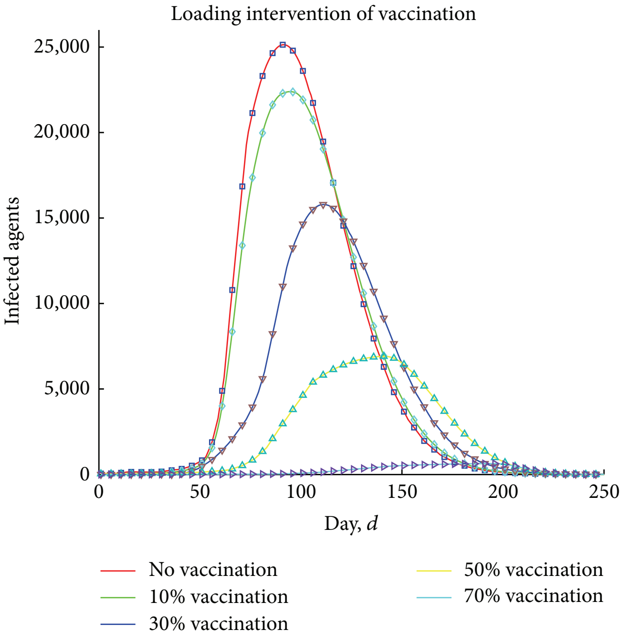


Figure 24: The results of loading intervention of vaccination.

Vaccination rates are set linearly (10%, 30%, 50%, and 70%) to test the changes of influenza. The peak value of infected agents decreases from 25,135, 22,392, 15,787, 6,907 to 638, respectively. It shows that the more the agents are vaccinated, the smaller the peak value of the infected agents will be. In the case of 70% vaccination, the total infected agent is only 13,799. It means that the influenza in this case does not happen due to the high vaccination rate.

The change of peak values also shows an interesting phenomenon. Vaccination rates increase linearly; the peak values decrease slower than the ascending of vaccination rates at beginning, but the descending becomes more and more faster with the linear increase of vaccination rate. It means that the small scale of vaccination (less than 30%) does not make sense. With the increase of vaccination rate, the number of susceptible agents decreases significantly. In sequence, the possibility of infections is lowered greatly. Additionally, it could be found that the decrease of peak values comes later than the increase of the vaccination rates. This is quite reasonable according to Mei et al.'s work in [57].

As a result, it is concluded that high vaccination rate is effective in the control of H1N1 influenza, though costly.

5.4.3. The Experiments of Sensitivity Analysis of Isolation. According to Section 5.1.2, isolation is a typical nonmedical intervention. It is the most used intervention measure in the emergency management in influenza. As mentioned before, a problem also exists that the measure cannot be validated before the intervention. Therefore the sensitivity of isolation is analyzed in this group of experiments. It is worth mentioning that the isolation is usually combined with the hospital policy. It means the infected individual is sent to hospital; meanwhile the people who contact with the infected are all isolated. As a result, parameters delay to hospital and isolation duration are designed together in the settings of experiment. Delay to hospital stands for the period from infection time to hospital time while Isolation duration represents the period when an agent cannot contact others if he has contact with an infected agent. The settings of these parameters are listed in Table 8; average results are shown in Figure 25.

Firstly, the parameter delay to hospital is simulated alone to test the sensitivity. The simulated results are shown in solid line in Figure 25. The peak value decreases from 6,708 to 2,218 when the delay to hospital decreases from 1.5 days to 1 day. The half day change brings the 67% reduction. Moreover, the peak value almost approximates to zero in the case of 0.5 days. The influenza does not happen in this case. It can be concluded that the earlier the infected agents

Table 8: The initial settings of experiments of isolation.

Parameters	Value
Agent count	19,610,000
Initial infected count	40
Interventions activated time ($T_{\text{Intervention}}$)	80
Delay to hospital (T_{HDelay}), isolation duration ($T_{\text{IDuration}}$)	{(1.5, 0), (1, 0), (0.5, 0), (1.5, 3), (1, 3), (0.5, 3)}

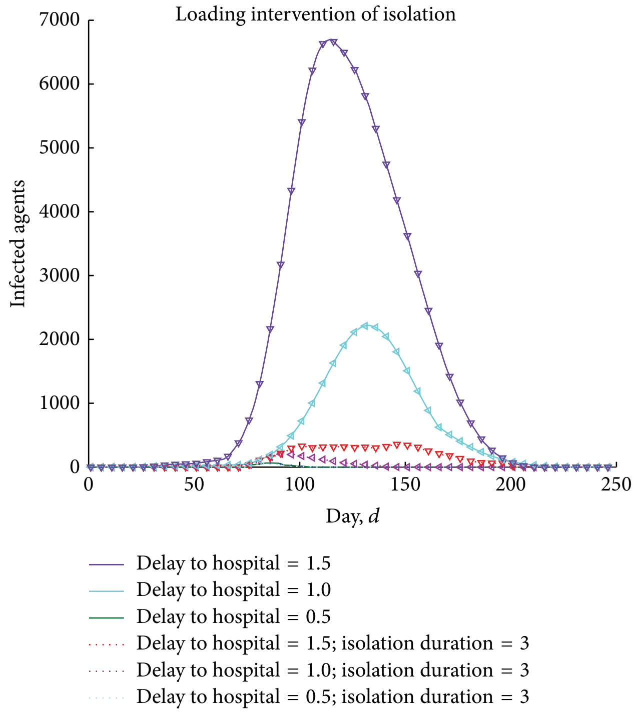


Figure 25: The results of loading intervention of isolation.

are sent to hospital, the less the agents will be infected. So the construction of effective mechanism to find infected individuals is important in the emergency management of influenza.

Secondly, the parameter isolation duration is added to simulate the isolation of intervention. Obviously, the isolation measure decreases the infected agents greatly. Compared to the parameter delay to hospital of 1.5-day case, the peak value decreases from 6,708 to 362 when the isolation is added in the intervention. The reduction of 6346 agents comes from the 3-day isolation of agents who had contacted with the infected agents. However, the decrease of parameter delay to hospital cannot bring the similar reduction in the 3-day isolation cases. According to the figure, the peak value decreases from 362 to 51; from 1.5-day case to 0.5-day case, only 311 agents are saved from infection. As a result, the sensitivity of simulated results does not change obviously with delay to hospital in the case of isolation. It can be concluded that isolation is a more effective measure. Though it is not easy for emergency management organizations to decrease the delay to hospital due to the current monitoring mechanism, the influenza can still be controlled by isolation.

5.4.4. The Experiments of Combination of Intervention Measures. Isolation is not the only nonmedical intervention in influenza, close workspace is another common measure. When close workspace is activated, the workspaces are all closed, agents can only stay home at the closed time. Luckily, isolation and close workspace are independent from one another. It is possible for emergency management organizations to use these two measures together. So this group of experiments is designed to obtain the effectiveness of the combination of intervention measures. The parameters are initialized with different settings listed in Table 9.

Parameters delay to hospital and isolation duration is set in the middle values (1 day, 3 days) according to the last section. According to the experience, close duration is set as seven days. It is known that close workspace is a really cost intervention measure; the activated time is a key parameter. With the fixed parameters mentioned before, the change of interventions activated time is analyzed in this group of computational experiments. The average results are shown in Figure 26.

Compared with the simulated results of experiments discussed above, Figure 26 strongly illustrates that influenza is ceased when the intervention measures are combined together. The peak value of infected agents is 141 in the "80th days of close-workspace and isolation with 1 day delay to hospital" case while 25,135 in "no intervention" case, 2,218 in "1 day delay to hospital" case, 203 in "1 day delay to hospital and isolation" case. The interventions bring a great reduction of infection; the peak values decrease from ten thousands level to hundred level.

On the other hand, the peak value even decreases to 108 when the execution time of close workspace is 10 days earlier. The date of maximum value of infected agents is advanced from the 135th day to the 108th day. The whole circle of H1N1 influenza is reduced. The results of this group of experiments show that intervention measures should be activated as soon as possible, and the compositions of interventions are more effective in emergency management of H1N1 influenza.

In summary, based on the general process, lots of experiments could be done by KD-ACP to study emergency management in H1N1 influenza. Many conclusions are obtained based on the analysis of the experiments in this section. According to these conclusions, the most optimal response in our case study is the case of "70th days of close-workspace and isolation with 1 day delay to hospital." It is because this case has the fewest infected agents relatively. But the modelling of H1N1 influenza is not sufficient. Economic and social factors of interventions, such as the money cost by vaccination and isolation, the influence of the close-workspace in society,

Table 9: The initial settings of experiments of isolation and close workspace.

Parameters	Value
Agent count	19,610,000
Initial infected count	40
Interventions activated time ($T_{\text{Intervention}}$)	70th day, 80th day
Delay to hospital (T_{HDelay}),	1 day
Isolation duration ($T_{\text{IDuration}}$)	3 days
Close workspace duration ($T_{\text{CWDuration}}$)	7 days

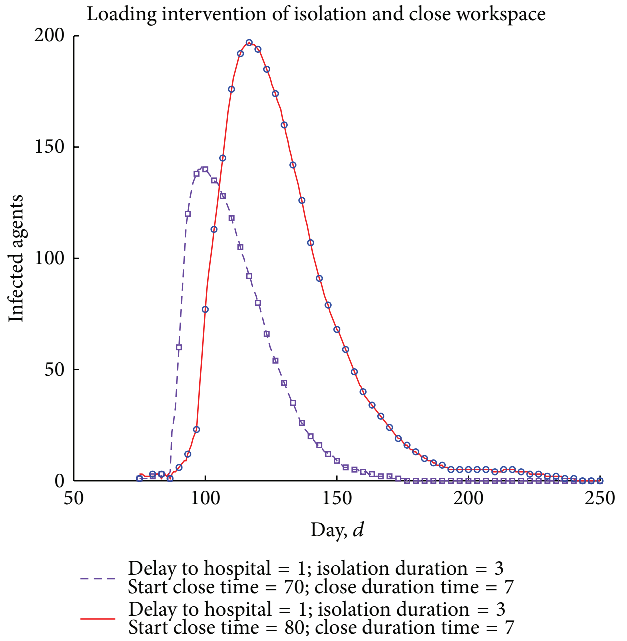


Figure 26: The results of loading interventions of isolation and close workspace.

and even opinion transmission of H1N1 in internet, are not considered in our case study. Without the analysis of these factors, it does not make sense to talk about the most optimal intervention measures. However, KD-ACP had provided a software framework and general process of the research on emergency management. Supported by KD-ACP, the experts of emergency management are able to study emergent events such as H1N1 influenza as detailed as possible. Based on the experiments and analysis again and again, the most optimal intervention measures will be found ultimately.

5.5. Parallel Execution by ASST and OsdRT. As discussed in Section 3.5, ASST and OsdRT are used to support the parallel execution of artificial Beijing. The statistical data is collected and stored in artificial society runtime database at runtime. ASST downloads the infection data from the database and displays it in the map. As shown in Figure 27, the infected agents are distributed in the districts in Beijing. The snapshot gives the geospatial distribution of the H1N1 epidemic. The legends of the agents in different status are listed in the left right corner in Figure 27. Five statuses are shown in the figure: health without vaccination, health with vaccination, incubation, being symptomatic, and convalescence. ASST provides a very direct feeling for the emergency experts. Partly with the situation from ASST, Emergency Decision Organization makes the emergency response plans to the response of the emergency scenario. With the help of plans, the epidemic in actual society can be restrained. In the meantime, the influenza information is reflected in the Internet. According to Figure 8, the data acquisition, data extraction, and data standardization of H1N1 influenza are processed in OsdRT. OsdRT is implemented by Dr. Cao's team from Institute of Automation from Chinese Academy of Sciences [58]. The tool provides not only the reports of H1N1, SARS, hand-foot-mouth disease, and intestinal diseases are all monitored. The web page is shown in Figure 28; H1N1 reports from Internet are collected first. Based on the data extraction of the reports, H1N1 case of distribution is obtained and shown in the incidence map. As a result, H1N1 incidence data is regarded as the knowledge which is sent to EECT and IMCT. The settings of emergent events and intervention measures are modified in order to make artificial society approach actual society in next turn of computational experiments. With the help of OsdRT, the loop of parallel execution process integrates ACP approach.

6. Conclusion

This work provides an integrated software framework for social Computing in emergency management. From a system perspective, KD-ACP provides a reliable, flexible, low cost, and effective platform for the scientists to do the research on emergency response problems. The analysis and prediction of these problems with inherent complexity can be solved by the repetition of possible alternative experiments on the software framework. Therefore, KD-ACP which is an attempt to implement ACP approach seemed as an effective computational framework to support the decision making in emergency response.

Currently, KD-ACP is used to study the H1N1 epidemic in Beijing in 2009. The most optimal compositions of intervention measures are testified to support the response to next influenza. However, it is still a long way ahead before the automation, systematization, and practicalization of KD-ACP. A lot of work will be carried out along several directions as in the following.

(i) The parallel execution process in KD-ACP is actually not strongly connected with the emergency response organization. Traditional experience and theory still play a dominant role in the decision of emergency management. It is necessary to propose a mechanism of hall for workshop to facilitate the parallel execution loop in actual society.

(ii) Section 3.3 introduces the principle and process of modelling artificial society. Metamodels and models of artificial Beijing are built by GME. Model transformation and code generation are used to implement codes from FSM based models. But many details

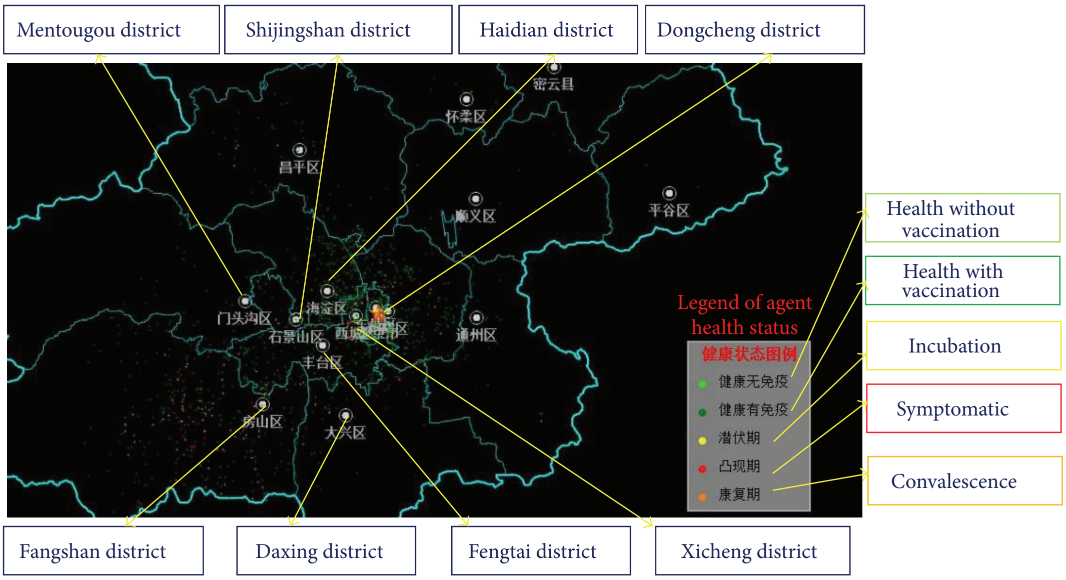
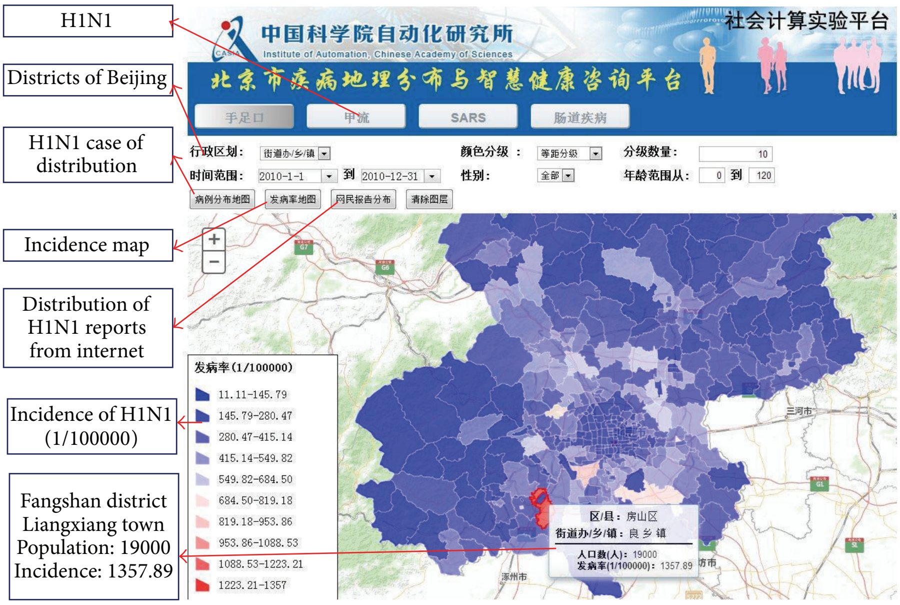


Figure 27: The situation of artificial Beijing by ASST.



行政区划：街道办/乡/镇　颜色分级：等距分级　分级数量：10

时间范围：2010-1-1 到 2010-12-31　性别：全部　年龄范围从：0 到 120

发病率(1/100000)

▇	11.11-145.79
▇	145.79-280.47
▇	280.47-415.14
▇	415.14-549.82
▇	549.82-684.50
▇	684.50-819.18
▇	819.18-953.86
▇	953.86-1088.53
▇	1088.53-1223.21
▇	1223.21-1357

区/县：房山区
街道办/乡/镇：良乡镇
人口数(人)：19000
发病率(1/100000)：1357.89

Figure 28: The implementation of OsdRT by Institute of Automation from Chinese Academy of Sciences.

are still not illustrated due to the topic of this paper. The key techniques such as template of code framework and model transformations from specific domains to FSM will be focused and introduced in detail in our next paper.

(iii) The performance of computational experiments in KD-ACP is still slower than expectation. The optimization of simulation engine and models structure is necessary to improve the performance. So it is possible to run artificial society with the real world in parallel.

In summary, KD-ACP paves a new way for scientists in both emergency management and computation simulation to collaborate with each other in solving the emergency management problems by social computing.

Research Highlights

(i) KD-ACP is implemented based on ACP approach.

(ii) Artificial Beijing is built with the help of KD-ACP.

(iii) H1N1 is modeled to simulate the influenza in Beijing in 2009.

(iv) Computational experiments testify the effectiveness of intervention measures.

Conflict of Interests

The authors declare that there is no conflict of interests regarding the publication of this paper.

Acknowledgment

This work was supported in part by the National Science Foundation of China (nos. 91024030, 71303252, 61403402, and 91324013).

References

[1] W. Mao and F.-Y. Wang, *Advances in Intelligence and Security Informatics*, Academic Press, Oxford, UK, 2012.

[2] E. Bonabeau, "Agent-based modeling: methods and techniques for simulating human systems," *Proceedings of the National Academy of Sciences of the United States of America*, vol. 99, no. 3, pp. 7280–7287, 2002.

[3] K. M. Carley, D. B. Fridsma, E. Casman et al., "BioWar: scalable agent-based model of bioattacks," *IEEE Transactions on Systems, Man, and Cybernetics A: Systems and Humans*, vol. 36, no. 2, pp. 252–265, 2006.

[4] J. M. Epstein, "Modelling to contain pandemics," *Nature*, vol. 460, no. 7256, p. 687, 2009.

[5] *Los Alamos National Laboratory Reports about EPISIMS*, Los Alamos National Laboratory, 2003.

[6] C. H. Builder and S. C. Bankes, "Artificial societies: a concept for basic research on the societal impacts of information technology," RAND Report 149, RAND Corp., 1991.

[7] F.-Y. Wang, "Computational theory and method on complex system," *China Basic Science*, vol. 6, no. 5, pp. 3–10, 2004.

[8] F.-Y. Wang, "Toward a paradigm shift in social computing: the ACP approach," *IEEE Intelligent Systems*, vol. 22, no. 5, pp. 65–67, 2007.

[9] F.-Y. Wang and S. Tang, "Artificial societies for integrated and sustainable development of metropolitan systems," *IEEE Intelligent Systems*, vol. 19, no. 4, pp. 82–87, 2004.

[10] D. Wen, Y. Yuan, and X.-R. Li, "Artificial societies, computational experiments, and parallel systems: an investigation on a computational theory for complex socioeconomic systems," *IEEE Transactions on Services Computing*, vol. 6, no. 2, pp. 177–185, 2013.

[11] F. Wang, F. Zeng, and Y. Yuan, "An ACP-based approach for complexity analysis of E-commerce system," *Complex Systems and Complexity Science*, vol. 5, no. 3, pp. 1–8, 2008 (Chinese).

[12] F.-Y. Wang and P. K. Wong, "Research commentary: intelligent systems and technology for integrative and predictive medicine: an ACP approach," *ACM Transactions on Intelligent Systems and Technology*, vol. 4, no. 2, pp. 1–6, 2013.

[13] J. Sifeng, X. Gang, F. Dong, and H. Chunpeng, "Study on the emergency rescue decision support system of petrochemical plant based on ACP theory," in *Proceedings of the 6th Management Annual Meeting (MAM '11)*, 2011, (Chinese).

[14] X. Dong, D. Fan, G. Xiong, F. Zhu, Z. Zhang, and Y. Yao, "Parallel Bus Rapid Transit (BRT) operation management system based on ACP approach," in *Proceedings of the 6th Management Annual Meeting (MAM '11)*, Beijing, China, 2011.

[15] B. Ning, F.-Y. Wang, H.-R. Dong, R.-M. Li, D. Wen, and L. Li, "Parallel systems for Urban rail transportation based on ACP approach," *Journal of Transportation Systems Engineering and Information Technology*, vol. 10, no. 6, pp. 23–28, 2010 (Chinese).

[16] F.-Y. Wang, J. Zhao, and S.-X. Lun, "Artificial power systems for the operation and management of complex power grids," *Southern Power System Technology*, vol. 2, no. 3, pp. 1–11, 2008 (Chinese).

[17] M. J. North, T. R. Howe, N. T. Collier, and J. R. Vos, "Repast simphony development environment," in *Proceedings of the Agent 2005 Conference on Generative Social Processes, Models and Mechanisms*, 2005.

[18] M. J. North, T. R. Howe, N. T. Collier, and J. R. Vos, "Repast Simphony runtime system," in *Proceedings of the Agent Conference on Generative Social Processes, Models and Mechanisms*, 2005.

[19] *SWARM Development Group. Swarm 2.1.1 Reference Guide*, 2000, http://www.swarm.org.

[20] S. Tisue and U. Wilensky, "NetLogo: a simple environment for modeling complexity," in *International Conference on Complex Systems*, pp. 16–21, 2004.

[21] S. Luke, C. Cioffi-Revilla, L. Panait, K. Sullivan, and G. Balan, "MASON: a multiagent simulation environment," *Simulation*, vol. 81, no. 7, pp. 517–527, 2005.

[22] M. Ichikawa, H. Tanuma, Y. Koyama, and H. Deguchi, "SOARS: introduction as a social microscope for simulations of social interactions and gaming," in *Organizing and Learning through Gaming and Simulation. Proceedings of the 38th Conference of the International Simulation and Gaming Association (ISAGA '07)*, pp. 149–158, 2007.

[23] D. S. Burke, J. M. Epstein, D. A. T. Cummings et al., "Individual-based computational modeling of smallpox epidemic control strategies," *Academic Emergency Medicine*, vol. 13, no. 11, pp. 1142–1149, 2006.

[24] F.-Y. Wang, "Parallel control and management for intelligent transportation systems: concepts, architectures, and applications," *IEEE Transactions on Intelligent Transportation Systems*, vol. 11, no. 3, pp. 630–638, 2010.

[25] Z. Molnár, D. Balasubramanian, and A. Lédeczi, "An introduction to the generic modeling environment," in *Proceedings of the TOOLS Europe 2007 Workshop on Model-Driven Development Tool Implementers Forum*, Zurich, Switzerland, 2007.

[26] S. Kelly and J.-P. Tolvanen, *Domain Framework, in Domain-Specific Modeling: Enabling Full Code Generation*, John Wiley & Sons, Hoboken, NJ, USA, 2007.

[27] B. P. Zeigler, H. Praehofer, and T. G. Kim, *Theory of Modeling and Simulation: Integrating Discrete Event and Continuous Complex Dynamic Systems*, Academic Press, San Diego, Calif, USA, 2000.

[28] G. Booch, J. Rumbaugh, and I. Jacobson, *The Unified Modeling Language: User Guide*, Addison-Wesley, Reading, Mass, USA, 1999.

[29] F. Wagner, R. Schmuki, T. Wagne, and P. Wolstenholme, *Modeling Software with Finite State Machines: A Practical Approach*, CRC & Taylor & Francis, Boca Raton, Fla, USA, 2006.

[30] T. Espiner, "China builds world's fastest supercomputer," ZDNet UK, October 2010.

[31] B. Chen and G. Guo, "A two-tier parallel architecture for artificial society simulation," in *Proceedings of the ACM/IEEE/SCS 26th Workshop on Principles of Advanced and Distributed Simulation (PADS '12)*, pp. 184–186, July 2012.

[32] G. Guo, B. Chen, X. G. Qiu, and Z. Li, "Parallel simulation of large-scale artificial society on CPU/GPU mixed architecture,"

in *Proceedings of the ACM/IEEE/SCS 26th Workshop on Principles of Advanced and Distributed Simulation (PADS '12)*, pp. 174–177, July 2012.

[33] S.-X. Lun, "Research on the classification of parallel execution modes of ACP theory," *Acta Automatica Sinica*, vol. 38, no. 10, pp. 1602–1608, 2012.

[34] F.-Y. Wang, "A framework for social signal processing and analysis: from social sensing networks to computational dialectical analytics," *Science China*, vol. 43, no. 12, pp. 1598–1611, 2013.

[35] G. Guo, *One Model User's Manual, version 1.2*, National University of Defense University, 2011.

[36] Y. Ge, R. Meng, Z. Cao, X. Qiu, and K. Huang, "Virtual city—an individual-based digital environment for human mobility and interactive behavior," *Simulation: Transactions of the Society for the Modeling and Simulation International*, 2014.

[37] M. Kretzschmar and R. T. Mikolajczyk, "Contact profiles in eight European countries and implications for modelling the spread of airborne infectious diseases," *PLoS ONE*, vol. 4, no. 6, Article ID e5931, 2009.

[38] W. Duan and X. Qiu, "Fostering artificial societies using social learning and social control in parallel emergency management systems," *Frontiers in Computer Science*, vol. 6, no. 5, pp. 604–610, 2012.

[39] W. J. Edmunds, C. J. O'Callaghan, and D. J. Nokes, "Who mixes with whom? A method to determine the contact patterns of adults that may lead to the spread of airborne infections," *Proceedings of the Royal Society B: Biological Sciences*, vol. 264, no. 1384, pp. 949–957, 1997.

[40] J. Banks, J. S. Carson, B. L. Neison, and D. M. Nicol, *Discrete-Event System Simulation*, Prentice Hall, Englewood Cliffs, NJ, USA, 4th edition, 2007.

[41] H. Yongxia, "Investigation and analysis of college students' work and rest," *Value Engineering*, vol. 31, no. 3, 2012 (Chinese).

[42] W. Duan, Z. Cao, Y. Wang et al., "An ACP approach to public health emergency management: using a campus outbreak of h1n1 influenza as a case study," *IEEE Transactions on Systems, Man, and Cybernetics Part A: Systems and Humans*, vol. 43, no. 5, pp. 1028–1041, 2013.

[43] H. W. Hethcote, "The mathematics of infectious diseases," *SIAM Review*, vol. 42, no. 4, pp. 599–653, 2000.

[44] G. Chowell, E. Shim, F. Brauer, and et al, "Modelling the transmission dynamics of acute haemorrhagic conjunctivitis: application to the 2003 outbreak in Mexico," *Statistics in Medicine*, vol. 25, no. 11, pp. 1840–1857, 2006.

[45] Y. Ge, L. Liu, B. Chen, X. Qiu, and K. Huang, "Agent-based modeling for Influenza H1N1 in an artificial classroom," *Systems Engineering Procedia*, vol. 2, pp. 94–104, 2011.

[46] A. R. Tuite, A. L. Greer, M. Whelan et al., "Estimated epidemiologic parameters and morbidity associated with pandemic H1N1 influenza," *CMAJ*, vol. 182, no. 2, pp. 131–136, 2010.

[47] L. Brouwers, B. Cakici, M. Camitz, A. Tegnell, and M. Boman, "Economic consequences to society of pandemic H1N1 influenza 2009—preliminary results for Sweden," *Euro Surveillance*, vol. 14, no. 37, pp. 1–7, 2009.

[48] Y. Ge, L. Liu, X. Qiu, H. Song, Y. Wang, and K. Huang, "A framework of multilayer social networks for communication behavior with agent-based modeling," *Simulation: Transactions of the Society for the Modeling and Simulation International*, vol. 89, no. 7, pp. 810–828, 2013.

[49] W. Duan, X. Qiu, Z. Cao, X. Zheng, and K. Cui, "Heterogeneous and stochastic agent-based models for analyzing infectious diseases' super spreaders," *IEEE Intelligent Systems*, vol. 28, no. 4, pp. 18–25, 2013.

[50] *The Preparation and Emergency Response Plans to Pandemic Published by Ministry of Health*, Ministry of Health, 2005, (Chinese).

[51] R. M. Fujimoto, *Parallel and Distributed Simulation Systems*, Wiley Interscience, 2000.

[52] F. Cicirelli, A. Giordano, and L. Nigro, "Distributed simulation of situated multi-agent systems," in *Proceedings of the 15th IEEE/ACM International Symposium on Distributed Simulation and Real Time Applications (DS-RT '11)*, pp. 28–35, September 2011.

[53] B. Chen and G. Guo, "A two-tier parallel architecture for artificial society simulation," in *Proceedings of the 26th ACM/IEEE/SCS Workshop on Principles of Advanced and Distributed Simulation (PADS '12)*, pp. 184–186, July 2012.

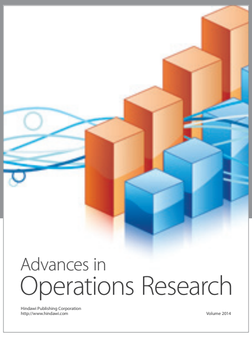
[54] G. Guo, B. Chen, and X. Qiu, "Parallel simulation of large-scale artificial society with GPU as coprocessor," *International Journal of Modeling, Simulation, and Scientific Computing*, vol. 4, no. 2, Article ID 1350005, 2013.

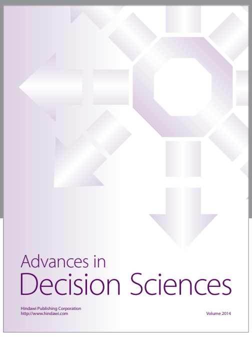
[55] B. Chen, L.-B. Zhang, X.-C. Liu, and H. Vangheluwe, "Activity-based simulation using DEVS: increasing performance by an activity model in parallel DEVS simulation," *Journal of Zhejiang University: Science C*, vol. 15, no. 1, pp. 13–30, 2014.

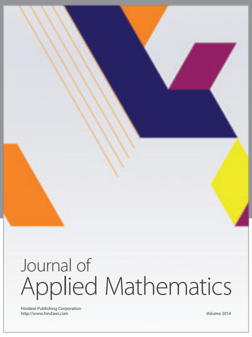
[56] X.-L. Wang, P. Yang, Z.-D. Cao et al., "Quantitative evaluation on the effectiveness of prevention and control measures against pandemic influenza A (H1N1) in Beijing, 2009," *Chinese Journal of Epidemiology*, vol. 31, no. 12, pp. 1374–1378, 2010 (Chinese).

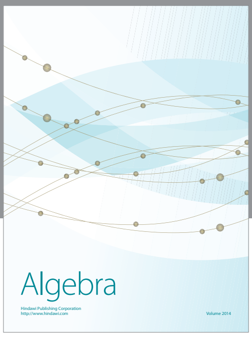
[57] S. Mei, A. D. van de Vijver, L. Xuan, Y. Zhu, and P. M. A. Sloot, "Quantitatively evaluating interventions in the influenza A (H1N1) epidemic on China campus grounded on individual-based simulations," *Procedia Computer Science*, vol. 1, no. 1, pp. 1675–1682, 2010.

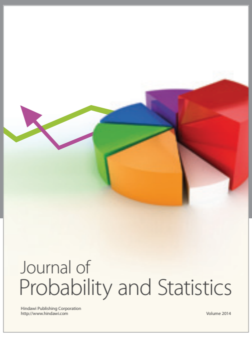
[58] X. Qiu, Z. Cao, and Z. Li, *Mid-Term Reports of China National Science Foundation 91024030*, 2014.


Advances in
Operations Research

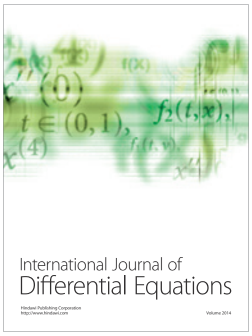

Advances in
Decision Sciences


Journal of
Applied Mathematics

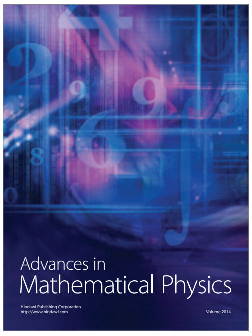

Algebra

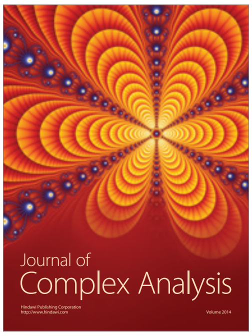

Journal of
Probability and Statistics

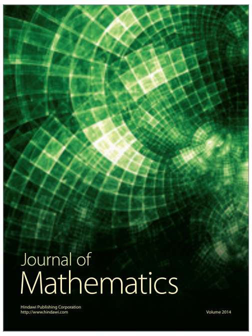
The Scientific
World Journal

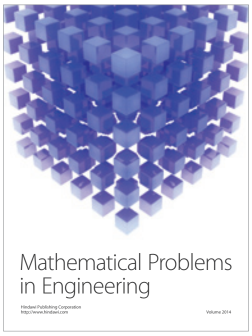

International Journal of
Differential Equations

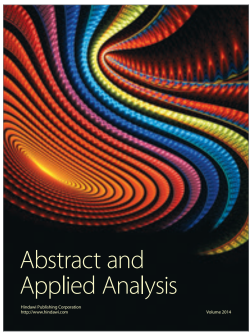
International Journal of
Combinatorics

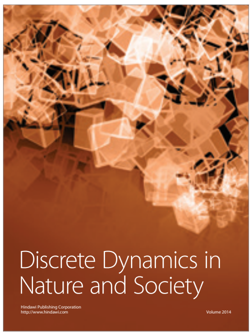

Advances in
Mathematical Physics

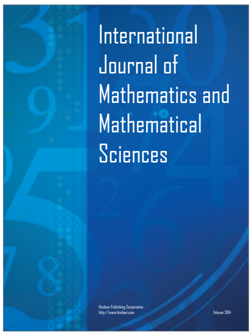

Journal of
Complex Analysis


Journal of
Mathematics


Mathematical Problems
in Engineering

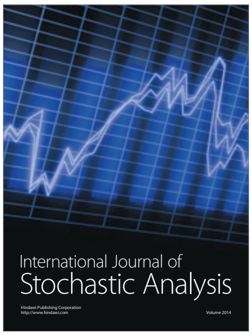

Abstract and
Applied Analysis


Discrete Dynamics in
Nature and Society


International
Journal of
Mathematics and
Mathematical
Sciences

Journal of
Discrete Mathematics

Journal of
Function Spaces


International Journal of
Stochastic Analysis

Journal of
Optimization