Hybrid Modelling and Simulation of Huge Crowd over a Hierarchical Grid Architecture

DAN CHEN[1], LIZHE WANG[1,2*], XIAOMIN WU[1], JINGYING CHEN[3], SAMEE U. KHAN[3], JOANNA KOŁODZIEJ[5], MINGWEI TIAN[1], FANG HUANG[6]

[1]China University of Geoscience, Wuhan, China.

[2]Center for Earth Observation and Digital Earth, Chinese Academy of Sciences, China

[3]Central China Normal University, Wuhan, China

[4]North Dakota State University, ND, USA

[5]Department of Mathematics and Computer Science, University of Bielsko-Biała, Poland

[6]University of Electronic Science and Technology of China, Chengdu, China

* Corresponding author's email address: LZWang@ceode.ac.cn

Abstract: The last decade has witnessed an explosion of the interest in technologies of large simulation with the rapid growth of both the complexity and the scale of problem domains. Modelling & simulation of crowd is a typical paradigm, especially when dealing with large crowd. On top of a hierarchical Grid simulation infrastructure, a simulation of evacuating tens of thousands of pedestrians in an urban area has been constructed. The simulation infrastructure can facilitate a large crowd simulation comprising models of different grains and various types in nature. A number of agent-based and computational models residing at two distinctive administrative domains operate together, which successfully presents the dynamics of the complex scenario at scales of both individual and crowd levels. Experimental results indicate that the proposed hybrid modelling & simulation approach can effectively cope with the size and complexity of a scenario involving a huge crowd.

Keywords: Hybrid Simulation, Cluster Computing, Grid Computing, Crowd Modelling & Simulation

---

1. INTRODUCTION

.

Modelling and simulation (M&S) are at the very core in many areas of science and engineering. With the rapid growth of both the complexity and the scale of problem domains, it has become a key requirement to create efficient and ever more complex simulations of large scale and/or high resolution for research, industry, and management. M&S of crowd is a typical paradigm. As a collective and highly dynamic social group, a human crowd is a fascinating phenomenon, which has been constantly concerned by experts from various areas. Recently computer-based modelling and simulation technologies have emerged to support investigation of the dynamics of crowds. Crowd M&S has now become a key design issue in many fields including military simulation, safety engineering, architectural design, and digital entertainment.

To represent the behaviour of a crowd, many behaviour models have been proposed with various types of modelling approaches [33], such as particle system models [13], flow-based models [14] and agent-based models [6]. Despite the many existing research efforts and applications in crowd M&S, it is still a young and emerging area. Work on modelling and simulation of large crowds (consisting of thousands of individuals or even more), especially at individual level, is still rare. Large agent-based systems, such as simulation of large crowds at individual level, have long been placed in the highly compute intensive world [4][6][21]. Study on crowd phenomenon still suffers from a lack of (1) an effective modelling approach to cope with the size and complexity of a scenario involving a huge crowd and (2) an appropriate platform to sustain such large crowd simulation systems.

In the last few years, there have been a lot of successful attempts of incorporating Grid [23][24][32] and Cloud technologies [1][2][3] [18][22][25][27] to foster large simulations over the Internet, such as the Grid-aware Time Warp Kernel [16] and the HLA_Grid_RePast framework [5][6]. However, these simulation systems are only suited for executing coarse-grained models due to the limited network bandwidth between different administrative domains. Another issue is that only few nodes of an administrative domain are accessible to the external users due to the existing security rules of most administrative domains. The third issue concerns the reusability: Large simulation developers often already have a set of simulation models/components situating over their intranet. Nevertheless, the advent of the hierarchical Grid infrastructure [4] paves the way to address the above problems. Existing individual bundles of

simulation models be linked to form a dedicated large simulation crossing the boundaries of previously independent simulation groups and the boundaries of the administrative domains. Fine-grained models still benefit from the advantages of Grid technology. Users can have a flexible solution to the reusability issue.

This study employs the hierarchical Grid infrastructure as an effective approach to addressing the two pending issues in large crowd modelling and simulation. The resulted simulation system is hybrid in nature: (1) to capture the individuality of pedestrians, agent-based models have been developed with each agent representing a pedestrian; (2) to describe the global dynamics of the whole (or a part of interest) crowd, computational models have been used; and (3) these heterogeneous models are linked together to allow studying the interaction dynamics of a crowd at both individual and global scales.

The large crowd simulation system had been deployed over two distinctive administrative domains locating at China. The simulated crowd has a size, which is prohibitively large for traditional simulation techniques even using a cutting-edge office desktop. Our approach successfully alleviates the bottleneck in the design and analysis of particularly large and complex scenarios like huge crowd. This study is one of the first to provide a solution to simulation of a crowd with models of multiple scales and types.

The rest of this paper is organized as follows: Section 2 briefs the background and related work of crowd modelling & simulation. Section 3 recaps the hierarchical Grid simulation infrastructure. Section 4 describes the case study of exploring the dynamics of a huge crowd in an evacuation procedure. Section 5 concludes this work with a summary and proposals on future work.

2. CROWD MODELLING AND SIMULATION

As a collective and highly dynamic social group, a human crowd is a fascinating phenomenon, which has been constantly concerned by experts from various areas. A crowd may exhibit highly complex dynamics, in general pure mathematical approaches or analytic models are not adequate in characterizing the dynamics of a crowd.

Recently modelling and simulation technologies have been gaining tremendous momentum in investigating crowd dynamics. Various simulation architectures have been developed [20]. To represent the

behaviour of a crowd, a number of behaviour models have been proposed with different types of modelling approaches, such as flow-based models and agent-based models. To study or mimic the dynamics of a crowd, modellers have considered a number of physical factors, social factors, and psychological factors when characterizing crowds in their models. Crowd models may also concern different aspects of a crowd. Some work aims at the "external characteristics" of a crowd, such as appearance, poses or movement patterns, coordinated positions of individuals; and some other works focus on how a crowd's social behaviours evolve over time upon some events.

Many open research issues are still very much in flux due to the complexity of individual and crowd behaviours [33]. Modelling of a crowd first needs to determine the scale (level of detail) of the model. The existing models are largely at two extreme levels: either modelling each individual as an autonomous agent equipped with some human-like behaviour modules such as locomotion, perception, and decision making, or treating the crowd as a whole or a collection of homogeneous particles with limited or no cognitive features. With the fast development of computing technology, there seems to be a trend in crowd simulation to model each individual as some kind of intelligent agent with attempts to incorporate more and more social and psychological factors into the agent behaviour model. However, we believe that the common shortcoming of the existing models is the absence of modelling the social group process and its impact on human behaviour.

The interoperability issue is largely ignored by the existing research. Although there already exist a number of crowd simulation models and modelling approaches, these models can hardly work with each other: they may operate at different levels of abstraction. This makes the communication between different models difficult. In a typical crowd simulation, the crowd model is required to pass the information about individuals' actions and understand the events in the simulated world, so that the individuals can determine how to respond to these events. However, research in interoperable crowd simulation models is still rare.

This study attempts to address the two pending issues via the hierarchical Grid simulation architecture. We aim to explore an approach to constructing simulations of huge crowd constituted by models of various scales and types, thus the dynamics of a huge crowd may be

investigated at different levels in a manner more comprehensive than existing approaches do.

3. THE HIERARCHICAL GRID ARCHITECTURE FOR LARGE HYBRID SIMULATION

In [4], we presented the hierarchical Grid computing architecture for large-scale simulation. The architecture serves as a simulation infrastructure, which can (1) across distributed administrative domains, (2) link multiple distributed simulation models into a large-scale simulation over Grids, and (3) reduce the communication overhead among simulation bundles. This section recaps basics of the Grid system as well as the design and functionalities of the architecture.

3.1. Grid System Architecture

As shown in Figure 1, a production Grid [26][27][28] typically contains a number of compute centres, which are linked by high-speed networking. A compute centre in general is organized at two levels. A head node, which hosts several access services and resource management functions for a compute centre, accepts incoming computing jobs and schedules them to local computing and data resources, which are termed here Compute Element (CE) and Storage Element (SE) respectively. Inside a data centre, the head node, CEs and SEs are interconnected by high performance local area network (LAN) [29][30].
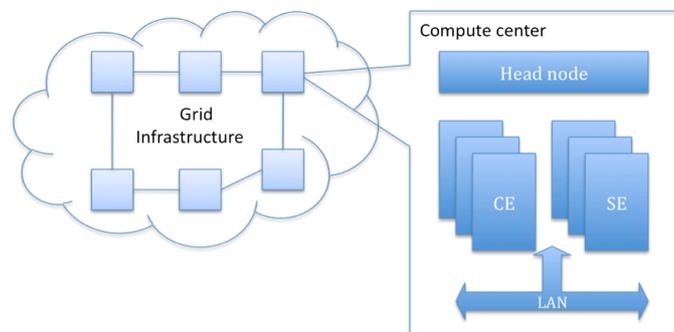

Figure 1. Grid computing system architecture

3.2. HLA-based Simulation Model

The High Level Architecture (HLA) is a technology for simulator interoperability and the de facto standard [15] for simulations over distributed computing platforms. The HLA defines software architecture

for modelling and simulation, which is designed to provide reuse and interoperability of simulation models/components, namely federates. A collection of federates interacting with each other for a common purpose formed a federation: in the context of this study, a simulation bundle is referred to as an independent federation running within a compute centre.

As the HLA defines the specification, it is the Runtime Infrastructure (RTI) that services federates for data interchange and synchronization in a coordinated fashion (see Figure 2). The RTI services are provided to each federate through its Local RTI Component (LRC) [5][7]. The RTI can be viewed as a distributed operating system providing services to support interoperable simulations executing in distributed computing environments [11].
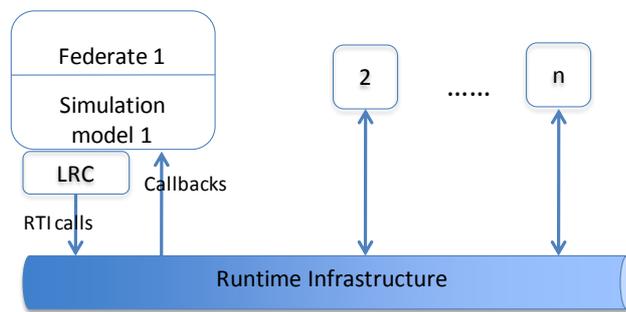

Figure 2. Model of HLA-based simulation

3.3. Hierarchical Grid Simulation Architecture: Overview

The design of the hierarchical Grid Simulation Architecture adopts a concept model as shown in Figure 3. A number of federates are grouped into multiple simulation bundles. Users can freely allocate federates to the same bundle for their own needs, e.g., to ensure closely relevant federates are together. Inside a simulation bundle, multiple federates communicate with each other via the local RTI. Each simulation bundle hosts a gateway service, which coordinates the communication of federates from different simulation bundles. The concept model of hierarchical Grid simulation infrastructure was designed conforming to the Grid system defined in Section 3.1.

As shown in Figure 4, a compute centre hosts a simulation bundle: the gateway service is implemented in the head node, and federates of the simulation bundle are computed in CEs of the compute centre. The gateway service communicates with gateways of other simulation bundles over the Grid infrastructure.
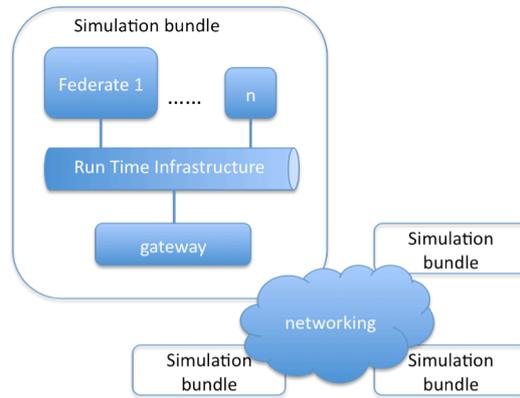
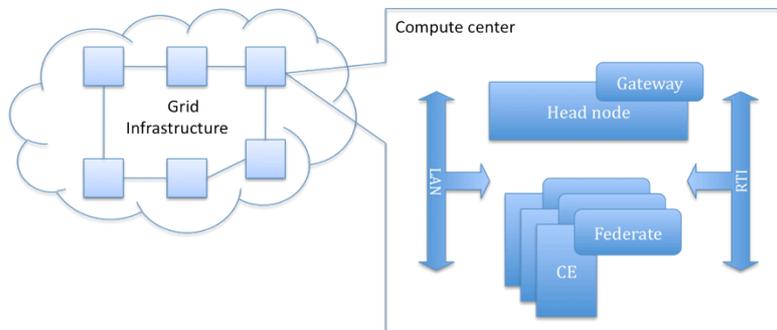Figure 3. Concept model of the hierarchical Grid simulation architecture

Figure 4. Large scale distributed simulation in a hierarchical Grid architecture

As illustrated by Figure 5, a gateway mainly consists of (1) a Grid federate module (GFM), (2) a Local federate module (LFM), (3) a Routing module, and (4) a Synchronization module. GFM is implemented as a Grid service and communicates with other gateways. The GFM retrieves events and runtime simulation data generated by simulation models from other simulation bundles. LFM operates an individual LRC to directly interact with federates in the simulation bundle. LFM receives time stamped ordered (TSO) events and other runtime simulation data from its local federation and generates events from other federations (remote events) the local federation. The Routing module deals with delivering an event to the right destination gateway and relays remote events to LFM. The synchronization module is responsible for delivering these events in a correct order and coordinating time advances of all simulation models. More details about the synchronization algorithm are available at [8].
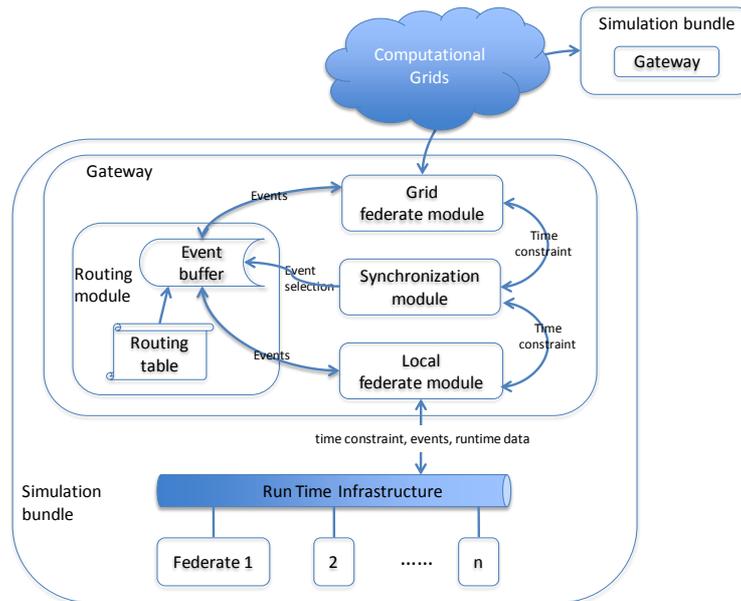
Figure 5. Gateway service architecture

4. HYBRID MODELLING & SIMULATION OF HUGE CROWD: A CASE STUDY

We have developed a large simulation system of a huge crowd's evacuation and its interaction with vehicles in an urban area. The objectives include: (1) to examine the feasibility of composing models of various types to characterize a crowd, (2) to explore the suitability of the hierarchical Grid simulation architecture for crowd simulation, and (3) to address the performance bottleneck of simulating a huge crowd.

4.1 The Huge Crowd Scenario

We first identified a typical scenario of huge crowd phenomenon: an evacuation procedure of a crowd after a National Flag Raising Ceremony at Tiananmen Square (see Figure 6(A)). The square is 440,000 square meters in area with a rectangle shape. The main body of the square is shown as the area circled by the dashed line. Pedestrians may freely traverse in this area except the zones restricted from access. Thousands of pedestrians can gather around the zone of interest (flagged) and remain still in the assembly area with the ongoing event. Figure 6(B) is a snapshot of a crowd gathered at the assembly area in real world.

On the completion of the event, pedestrians start to disperse away from the assembly area and leave the square via six passages, referred to as P1~P6. P1 and P2 denote the entrances of two pedestrian underpasses, and

P3~6 represent four pedestrian crosswalks. As soon as pedestrians enter P1 or P2, they will be considered out of the scenario. Two one-way roads locate on the top and beneath the circled area respectively, and pedestrians
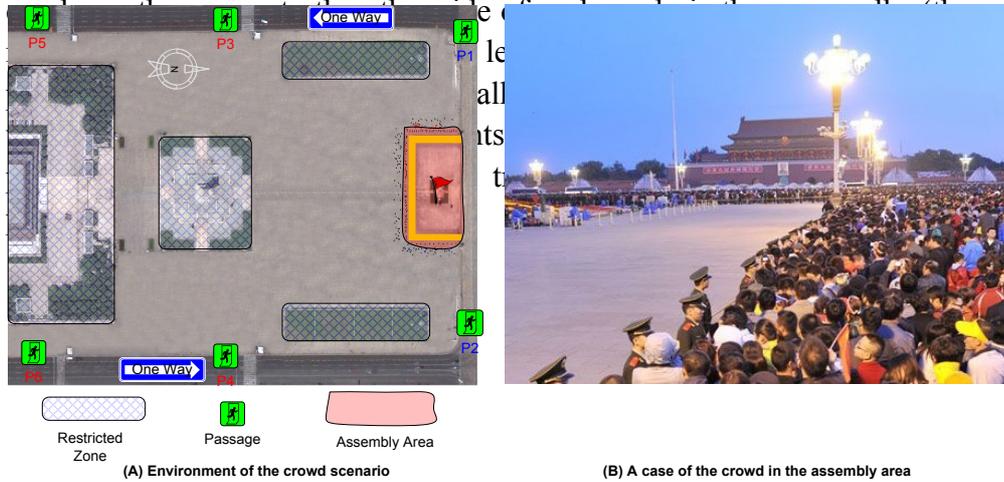

(A) Environment of the crowd scenario (B) A case of the crowd in the assembly area

Figure 6. Illustrating the huge crowd scenario

4.2 Simulation Models

To adapt to the size and complexity of the scenario, we developed three significantly different types of models with various scales and composed the huge crowd simulation with them. Those include (1) an agent-based model for pedestrians in the square (except the assembly area) and the crosswalks, (2) a macroscopic crowd model for the pedestrians aggregated only in the assembly area, and (3) another agent-based model for the vehicles on the roads.

4.2.1 Pedestrian Agent Model

In this study, we adopted an agent model to characterize a pedestrian's behaviours. An agent represents a pedestrian, who can be an independent individual or a member of (1) a tourist group, (2) several friends or (3) a group of relatives. In normal condition the model assumes that pedestrians would follow the rule of proximity, e.g., pedestrians with their initial

locations at the upper part of the square tends to selects the upper exits as their final goals. Figure 7 depicts an agent's state transition process spanning from its start state (initial location) to the end state (an exit reached) where it is removed from the simulation scenario.
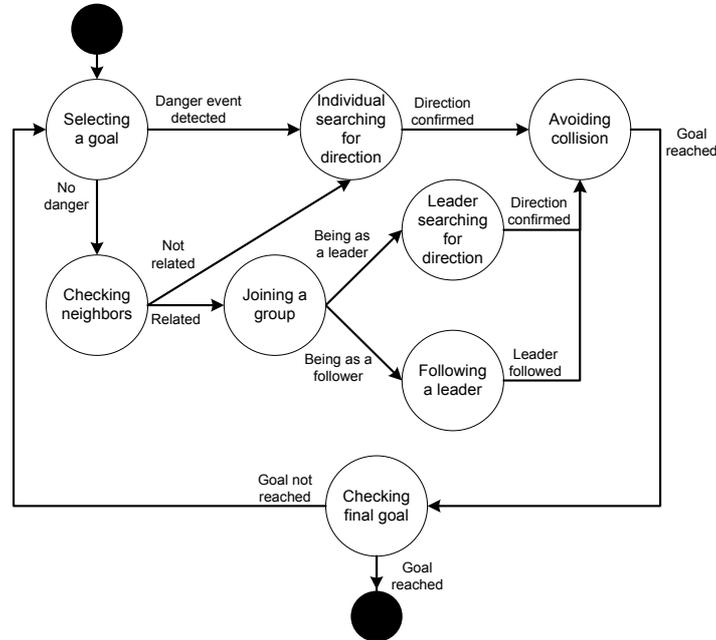

Figure 7. State transition of a pedestrian agent in terms of behaviours

An agent always tends to reach a passage to exist the scenario, and the passage will be selected as its "final goal". We define a human-like rule for routing the agent to fulfil its final goal. The agent's route area is always divided into four connected areas each associated with a sub-goal, which can be any location of the area. In a normal situation, i.e., no danger event detected, the agent needs to check whether it is related to the neighbours. If yes, the agent moves together with the related agent(s) as a group, the leader agent in the group searches for direction while other agents follow the leader. In contrast, an independent agent searches for direction individually.

Once any agent identifies the direction, it switches to the behaviour of avoiding neighbours. The agent checks whether the goal in the area is reached from time to time, if yes, it will move to the next area and do the check again; otherwise it continues searching for direction and avoiding behaviours until it reaches the final goal.

If a danger event is detected, the agent directly searches for direction without considering the grouping option. Another possibility not covered

by Figure 7 is that an agent can perish when encounters danger. The agent will reach end state immediately and be removed from the scenario.

4.2.2 Computational Model of the Crowd Aggregated in the Assembly Area

When dealing with a crowd of high density and large size, a macroscopic model often applies. Such macroscopic (computational) model treats a crowd as a whole to characterize some common features of the individuals in the crowd without distinguishing their individualities, e.g., the flow model using the continuum theory [14]. A computational model can obtain results at the scale of a whole crowd at a much lower computational expense in comparison with using agent-based models. In the study, we compose a computational model, which defines the mechanism for the pedestrians in the assembly area (referred to as *the crowd* in this discussion) choosing the route to the destination and implementing the choice of routing:

The crowd's speed is a dynamic three-dimensional function [31]. It defines the average speed of individuals located within unit area of floor space at given time point *t* and location *(x, y)*. The expected number of these individuals is the pedestrian density, written as ρ *(x, y, t)*; The chosen speed and moving direction of these individuals, i.e., the expected velocity is written as *v(x, y, t)*. The crowd's speed is subject to the density. The speed monotonically decreases from a "preferred speed" down to zero with the density varying from zero to a present maximum value.

The model always chooses the route via which requires the shortest time to reach the destination., We defined a cost function and a potential function for this purpose: The cost function, written as *c(x, y, t)*, represents the minimal time cost for the pedestrians at a given location to move a unit distance, which is determined by the density of pedestrians at this location; The potential function, written as ϕ *(x, y, t)*, defines the time to reach the final destination. The potential function can be calculated given that the neighbouring points with the minimal value of the cost function are always chosen from the current location to the destination. The cost function and the potential function can be quantified using an Eikonal Equation:

$$|\nabla \phi (x, y, t)| = c(x, y, t)$$

After the potential function is resolved from the above equation, the minimal time from any point to the destination can be obtained. The

pedestrians in the crowd will choose the route with the direction opposite that of the gradient of the potential function, $\phi(x, y, t)$. Eventually, the crowd movement is governed by a conservative equation:

$$\partial \rho(x, y, t)/\partial t + \nabla \cdot (v(x, y, t) \, \rho(x, y, t)) = 0$$

4.2.3 Vehicle Agent Model

The vehicles running on the two one-way roads are modelled using another type of vehicle agents with each agent representing an individual vehicle. A road consists of four lanes, and each agent only moves along one lane and will not switch to another. All vehicle agents tend to maintain the maxim speed allowed by the traffic control. A "safety distance" is defined to regulate vehicle agents according to their speeds. When two adjacent agents become too close, the one at the behind will attempt to slow down to maintain a safe distance between them. When an agent encounters a red traffic light, it stops to leave the crosswalk to the pedestrians. When a vehicle agent approaches the end of a road, it slows down for turning.

In summary, table I lists the basic features of the three types of models:

Table I: Features of simulation models

Models	Pedestrians	Crowd in the assembly area	Vehicles
Type	Agent-based	Computational	Agent-based
Scale	Individual	Crowd (macroscopic)	Individual
Outputs	Each pedestrian's behaviours	A crowd's dynamic distribution pattern	Each vehicle's motion pattern
Environment	2D space	2D space	1D space

4.3. The Crowd Simulation over the Hybrid Grid Simulation Infrastructure

The overall scenario consists of two main regions, i.e., the square and the surrounding roads involving entities with very different dynamics. Accordingly, two sets of simulation models were developed to represent the evolving activities in the two regions. Each set of models forms an individual simulation federation. The "square" federation and the "roads" federation operated on two administrative domains correspondingly (marked as domains "A" and "B" in the rest of this paper). The former federation (referred to as federation A) simulates how aggregated pedestrians disperse from the assembly area (see Figure 6) and "permeate" over the whole square. The later federation (referred to as federation B) describes the pedestrian flows on the crosswalks and the motion patterns

of the vehicles on the roads. In particular, the combined influence from the dynamics of the crowd on the square and from the traffic control on each pedestrian flow has been considered. Figure 8 illustrates the structure of the overall crowd simulation sustained by the infrastructure. The major event or data flows amongst individual federates are presented.

Twenty thousand pedestrians were simulated in the scenario, and this requires the same large number of deliberative agents which inevitably results in performance bottleneck for a desktop computer. Aiming at the potential, the square was divided into seven partitions as shown in Figure 9. Each dashed line represents a border between two neighbouring partitions. Dark grey blocks denote the areas restrict from access. Six pedestrian federates (denoted by A1~A6) were constructed with each corresponding to a partition linked to a crosswalk or passage to simulate the activities within the partition. A pedestrian federate consists of a number of pedestrian agents. Another crowd federate (A7) simulates the crowd initially aggregated in the assembly area using the macroscopic model. Each partition of the virtual environment was maintained by a federate respectively, as illustrated by Figure 8 which highlights the interactions amongst federates A4, A6 and A7 (The interactions between entities inside a federate are detailed in Section 4.2):

(1) A pedestrian federate to another

If a pedestrian moves to an adjacent partition, the ownership of the corresponding agent will be transferred to the federate maintaining this partition. This partition-crossing activity will be signalled by an event sent from the source federate to the destination federate, which carries the information of agents' states, the locations and the time point this movement occurs.

(2) The crowd federate to an adjacent pedestrian federate

The crowd federate provides statistical information about the pattern of the crowd dispersing from the assembly area. The information drives each adjacent pedestrian to create agents at certain locations and times as well as to initialise the state of each agent. This results in a gradual de-aggregation of the crowd in the assembly area, which was originally in an ultra high density.
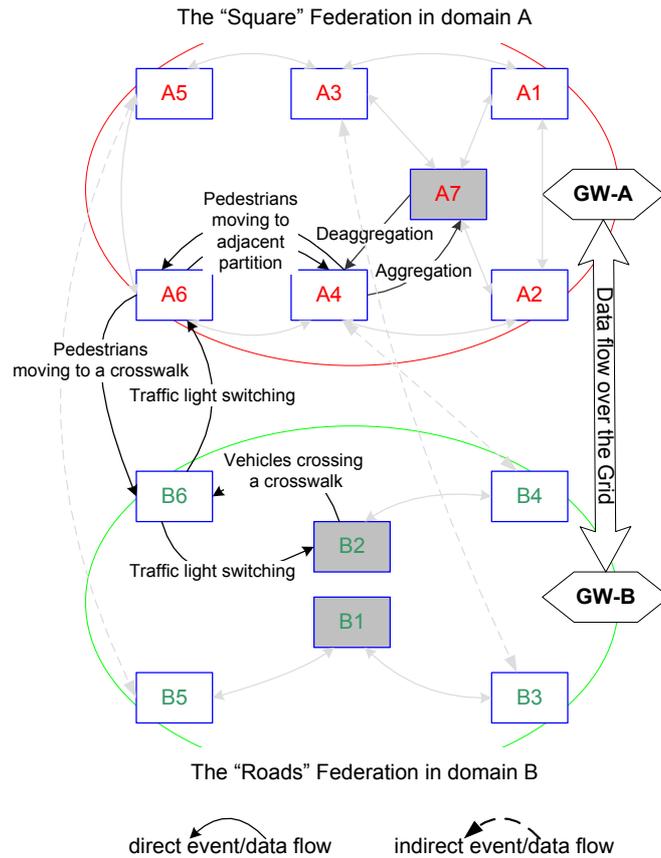
The "Square" Federation in domain A

Pedestrians moving to adjacent partition

Deaggregation

Aggregation

Pedestrians moving to a crosswalk

Traffic light switching

Data flow over the Grid

Vehicles crossing a crosswalk

Traffic light switching

The "Roads" Federation in domain B

direct event/data flow   indirect event/data flow

Figure 8. Structure of the crowd evacuation simulation system

(3) A pedestrian federate to the adjacent crowd federate

A pedestrian may move back to the assembly area from an adjacent partition, thus the departing agent will aggregate to the crowd simulated by the crowd federate. The macroscopic model needs to adapt to this event.

Federation B simulates vehicles and pedestrians' activities on the roads. Each road and the vehicles running on its top are simulated by an individual road federate (denoted by B1 and B2). A road federate maintains a number of vehicle agents. The activities of pedestrians on each crosswalk are simulated by an individual pedestrian federate (denoted by B3~6 corresponding to the crosswalks illustrated in Figures 6 and 8).
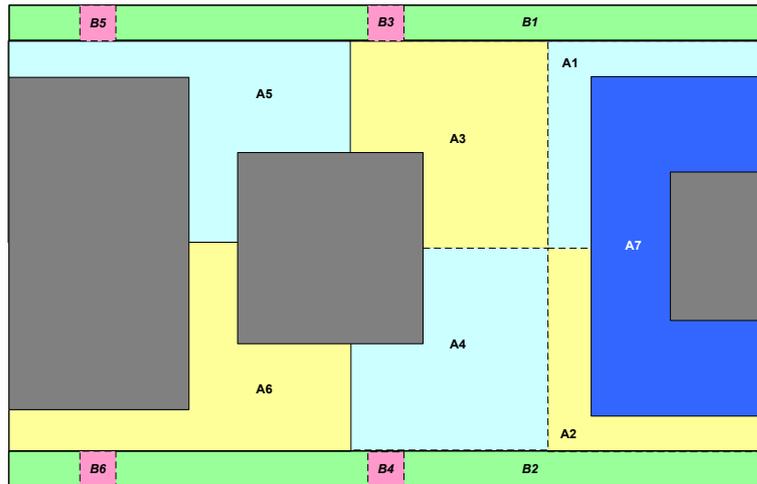
Figure 9. Conceptual View of the Partitions of the Scenario

(4) Between a pedestrian federate (in "square") and an adjacent pedestrian federate

Taking federate A6 and federate B6 for example, at a time point on which traffic light switches, federate B6 signals federate A6 with an event. If the light becomes green to pedestrians waiting in front of the crosswalk, federate A6 responds with an event of "pedestrian moving to the crosswalk". Federate B6 will be fed with the information of the pedestrians, and federate B6 accordingly initiates a procedure of simulating agent movement on the crosswalk.

(5) Between a pedestrian federate (in "roads") and a related vehicle federate

Federate B2 simulates the vehicles running on a road (B2). When the traffic light of the crosswalk associated with B6 remains green (from vehicles' perspective), federate B2 sends events to federate B6 about the vehicles crossing the crosswalk. When the traffic light switches, federate B6 signals federate B2 with an event. If the light turns to red or green, vehicle agents on federate B2 will respond to this event as described in Section 4.1.

In the whole crowd simulation, one virtual time unit in the simulation represents 0.1 second wall clock time. Sixty minutes' activities in the scenario were simulated, which corresponds to 36,000 virtual time units. The pedestrian federates' timesteps and lookaheads [17] are set to 2.0, and the vehicle models' timesteps and lookaheads are set to 1.0. As each traffic light signals every 60 seconds, the lookahead of both gateways are

set to 600. This value appropriately reflects the relation of the two federations. One virtual space unit represents 1.0 meter in the scenario. The resolution of the simulated square and crosswalks are set to 0.5 meter, while the resolution of the roads is set to 1.0 meter.

5. EXPERIMENTS AND RESULTS

The test bed is built upon two compute centres at Yanshan University and at China University of Geosciences, China. Each site contains a compute cluster. Table II describes their resource configuration. Grid services are provided by Globus Toolkit 5.0.3 [12].

Table II. Test bed description

Compute Centre	Resource Description
Yanshan Univ., China	Compute cluster: 12 nodes Node: Intel® Core™ i5-2520M Processor 2.50 GHz, 4G RAM Network:1Gbps Ethernet OS: Ubuntu Linux 10.04 LTS
CUG, China	Compute cluster: 10 nodes Node: Intel® Xeon® Processor E5603, 2G RAM Network: 2Gbps Myrinet OS: SuSE Linux Enterprise 11sp1

5.1. Communication Latency

We first performed a latency benchmark to investigate the communication overhead incurred by the underlying infrastructure. The benchmark measures the latency of federate communications as follows: One federate (on A1) sends an attribute update [7][15][17], and upon receiving this update, the other federate (on B1) sends it back to the sending federate. The elapsed time of this communication is calculated using the real-time taken at the sending and reflecting federates [5][6]. The averaged result indicates that the latency is about 373 milliseconds. Such communication between different compute centres over Grid is quite costly. It is important to have a design which can minimize the number and frequency of Grid communications thus to ensure a satisfactory runtime performance of the whole system.

5.2. Crowd Simulation Outputs

The experimental data recorded from one run were collected and the evolvement of the whole scenario is reconstructed. Snapshots of the

virtual scenario's states of the two regions at different stages are presented as follows:

(1) Initial Stage of the Activities in the Square

There are in total 30,000 agents with their locations set conforming to uniform distribution in the assembly area. Those represent the state of the crowd that initialized by federate A7 (the computational crowd model) according to the density function. The speed (see Section 4.2.2 for definition) of the crowd remains zero until the occurrence of the event "*End_of_the_ceremony*".

(2) Middle Stage of the Activities in the Square

After the triggering of the "*End_of_the_ceremony*" event, agents at federates A1~4 were initialized and driven by the inputs from A7 (see Figures 8 and 9) to represent the scattering pedestrians. With the simulation progressing, agents at models A5~6 were eventually driven to simulate the activities of pedestrians on partitions A5 and A6.

We developed a visualization end for displaying the pedestrians in the square area, and Figure 10(A) presents the overall distribution of pedestrians at simulation time 9,000. The coordinates of the rest pedestrians are exactly the agents' current positions logged by A1~6. Agents are marked with three different colours: red (a member of a tourist group), blue (a member of a group of friends), and black (an independent individual)

We selected two areas as shown in Figure 10(A) to have a zoom-in view. The zoom-in view explicitly gives the each pedestrian's position and moving direction. Figure 10(B) presents a number of agents in the middle of the square in which exist two tourist groups. We can see the agents in a group are close to each other and moving towards the similar directions. Figure 10(C) highlights a selected area at the entrance of passage P1. It can be observed that pedestrians are swarming towards the entrance of the crosswalk.

(3) Activities over Crosswalks and Roads

Federates B3~6 generate the activities of the pedestrians moving over the crosswalks, and federates B1~2 simulate the activities on the roads. We log the status data of walking pedestrians and vehicles simulated by these federates. Figure 11 reconstructs a 2D snapshot of the activities over crosswalk P4 and the status of nearby vehicles at simulation time 9,030. Each small circle and black block represents a pedestrian and vehicle correspondingly with an arrow marking its direction and rate.

(800,300)

(800, 0)
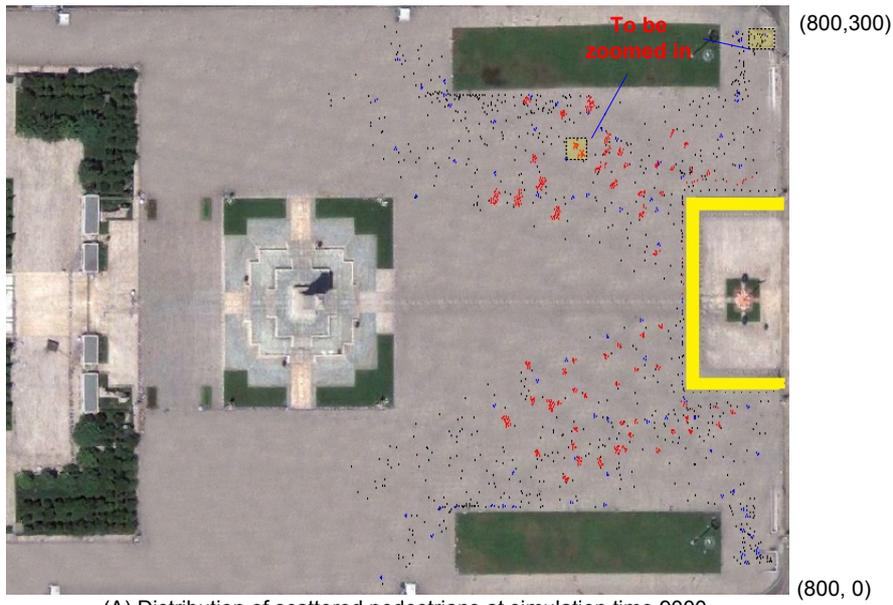
(A) Distribution of scattered pedestrians at simulation time 9000

(B) Zoom-in view of agents in the middle of the square

(C) Zoom-in view of agents at the entrance of a passage

Figure 10. Snapshot of the pedestrians in the square at simulation time 9,000
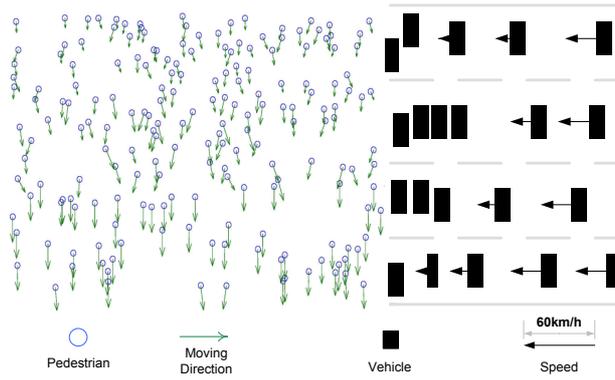

Pedestrian   Moving Direction   Vehicle   60km/h   Speed

Figure 11. Snapshot of the Pedestrians and Vehicles on a Road at Simulation Time 9,030

5.3. Performance Evaluation

In order to investigate the runtime performance of the hierarchical Grid architecture, we compare the crowd simulation over the architecture (referred to as the Grid-enabled crowd simulation, GCS) versus a standalone compute node at the CUG side (see Table II for specification). We developed a pedestrian agent model which covers the crowd activities of the square (i.e., the combination of federates A1~6) and uses the log of federate A7 as inputs to initialise the agents originating from the assembly area and the logs of federates B3~6 as inputs to throttle the actions of agents which finally exist the scenario via passages P3~6. This model is referred to as the sequential crowd simulation, SCS, in this section.

We performed a series of experiments which focuses on the execution times of GCS and SCS. We trust that the complexity of the two simulation systems is basically controlled by the number of pedestrians, i.e., the number of the deliberative agents.

Table 2 gives the execution times of the two types of crowd simulation systems with increasing crowd sizes. When there are 3000 pedestrians, the execution time of GCS is 1318s which longer than that of SCE. This is due to the high communication overhead of the Grid services. With the complexity increases significantly, GCS performs better when the communication overhead is covered by the benefits gained by distributing computation load of agents to multiple compute nodes. A single compute node can sustain about 8500 agents to the maxim.

The results indicate that (1) the hierarchical Grid architecture can significantly improve the runtime performance of executing complex simulation scenarios and (2) it scales well with the scenario's complexity. Therefore a fast-than-real-time method has been enabled to study the simulated scenario and the potential bottleneck of executing large number of deliberative agents has been successfully resolved.

Table III: Performance of sequential and parallelized COS systems

Number of agents	Execution time (Sec)	
	SCS	GCS
3000	897	1318
5000	1965	1571
8000	2923	1943
10000	N/A	2080
20000	N/A	2117
30000	N/A	2518

6. CONCLUSIONS AND FUTURE WORK

This paper presents a hybrid approach to modelling & simulation of huge crowd upon a hierarchical Grid simulation architecture. We successfully developed a simulation system of the evacuation procedure of a huge crowd in an urban area, which initially has a high density. Three heterogeneous types of models, namely a computational microscopic crowd model, pedestrian agent models and vehicle agent models, have been constructed to characterise different portions and aspects of the large and complex simulated scenario.

A simulation of huge crowd at the scales of both individual and crowd has been successfully developed with the support of the hierarchical Grid simulation architecture. The simulation outputs indicate that the proposed approach is effective in dealing with the size and complexity of scenarios involving huge crowd. The performance issue in connection with executing a large number of deliberative agents has been properly addressed as well.

These achievements are based on the underlying simulation architecture empowered by Grid computing. Coarse-grained and fine-grained models can be seamlessly interplayed to describe the global dynamics of the whole when being able to capture the individuality of pedestrians. The experimental results also indicated that even the execution efficiency of fine-grained models can be guaranteed when the asymmetry of the simulation models in different domains are properly mapped to the infrastructure. The infrastructure is well suited for compute intensive and large-scale simulation.

For future work, we will extend the current design to support large simulation with models using hybrid computing platforms, e.g., incorporating many-core architectures [9][10]. In particular, we are interested in adapting multi-scale ecosystem simulations and interacting neuronal networks [18] to the infrastructure.

REFERENCES

[1] Buyya, R., R. Ranjan, and R. N. Calheiros. 2010. InterCloud: Utility-Oriented Federation of Cloud Computing Environments for Scaling of Application Services. In Proceedings of the 10th International Conference on Algorihtms and Architectures for Parallel Processing, pp. 13-31, Busan, Korea, March, 2010.

[2] Calheiros, R. N., R. Ranjan, A. Beloglazov, C. A. F. De Rose, and R.Buyya. 2010. CloudSim: A Toolkit for the Modeling and Simulation of Cloud Resource Management and Application Provisioning Techniques. Journal of Software: Practice and Experience, Published online, DOI: 10.1002/spe.995.

[3] Calheiros, R. N., R. Ranjan, and R. Buyya. 2011. Virtual Machine Provisioning Based on Analytical Performance and QoS in Cloud Computing Environments. In Proceedings of the 40th International Conference on Parallel Processing (ICPP 2011), pp. 295-304, Taipei, Taiwan, September 13-16, 2011.

[4] Chen, D., Lizhe Wang, Congcong Bian, Xuguang Zhang. 2011. A Grid Infrastructure for Hybrid Simulations. International Journal of Computer Systems Science & Engineering 13(3): 197-206.

[5] Chen, D., S. J. Turner, W. Cai, M. Xiong. 2008. A Decoupled Federate Architecture for High Level Architecture-based Distributed Simulation. Journal of Parallel and Distributed Computing 68(11): 1487-1503.

[6] Chen, D., G. K. Theodoropoulos, S. J. Turner, W. Cai, R. Minson, Y. Zhang. 2008. Large scale agent-based simulation on the grid. Future Generation Computer Systems 24(7): pp. 658-671.

[7] Chen, D., S. J. Turner, W. Cai. 2008. Towards Fault-tolerant HLA-based Distributed Simulations. Simulation: Transactions of the Society for Modeling and Simulation International 84(10/11): 493-509.

[8] Chen, D., S. J. Turner, W. Cai, G. K. Theodoropoulos, M. Xiong, M. Lees. 2010. Synchronization in Federation Community Networks, Journal of Parallel and Distributed Computing 70(2):144-159.

[9] Chen, D., L. Wang, M. Tian, J. Tian, S. Wang, C. Bian, X. Li. 2011. Massively Parallel Modelling & Simulation of Large Crowd with GPGPU, Journal of Supercomputing (Springer), Online Publication, DOI: 10.1007/s11227-011-0675-4.

[10] Chen, D., L. Wang, G. Ouyang, X. Li. 2011. Massively Parallel Neural Signal Processing on a Many-Core Platform. Computing in Science and Engineering 13(6): 42-51.

[11] Fujimoto, R. M. 1998. Time Management in the High Level Architecture. Simulation: Transactions of the Society for Modeling and Simulation International 71(6): 388-400.

[12] Globus Toolkits. 2011. http://www.globus.org/

[13] Helbing, D., Farkas, I., and Vicsek, T. 2000. Simulating Dynamical Features of Escape Panic. Letters to Nature 407:487-490.

[14] Hughes, R.L. 2002. A Continuum Theory for the Flow of Pedestrians. Transportation Research Part B 36(6):507-535.

[15] IEEE 1516. 2000. IEEE Standard for High Level Architecture.

[16] Iskra, K., G. Albada, P. Sloot. 2005. Towards Grid-aware Time Warp. Simulation: Transactions of The Society for Modeling and Simulation International 81(4):293-306.

[17] Kuhl, F., R. Weatherly, J. Dahmann. 1999. Creating Computer Simulation Systems: An Introduction to HLA. ISBN 13-022511-8, Prentice Hall, USA.

[18] Laszewski, G., J. Dayal, Lizhe Wang. 2011. eMOLST: A Documentation Flow for Distributed Health Informatics. Concurrency and Computation: Practice and Experience 23(16): 1857-1867.

[19] Morrison, A., C. Mehring, T. Geisel, A.D. Aertsen, M. Diesmann. 2005. Advancing the Boundaries of High-connectivity Network Simulation with Distributed Computing. Neural Computing 17(8):1776-1801.

[20] Musse, S. R. and Thalmann, D. 2001. Hierarchical Model for Real Time Simulation of Virtual Human Crowds. IEEE Transactions on Visualization and Computer Graphics 7(2):152–164.

[21] Vigueras, G., Juan M. Orduña and Miguel Lozano. 2010. A GPU-Based Multi-agent System for Real-Time Simulations. Advances in Intelligent and Soft Computing 70/2010:15-24.

[22] Wang, L., Cheng Fu. 2010. Research Advances in Modern Cyberinfrastructure. New Generation Computing 28(2):111-112.

[23] Wang, L., Gregor von Laszewski, Jie Tao, Marcel Kunze. 2009. Grid Virtualization Engine: Design, Implementation, and Evaluation. IEEE Systems Journal 3(4):477-488.

[24] Wang, L., D. Chen, Fang Huang. 2011. Virtual Workflow System for Distributed Collaborative Scientific Applications on Grids. Computers & Electrical Engineering 37(3):300-310.

[25] Wang, L., G. von Laszewski, A. J. Younge, X. He, M. Kunze, J. Tao, C. Fu. 2010. Cloud Computing: a Perspective Study. New Generation Computing 28(2):137-146.

[26] Wang, L., G. von Laszewski, Jie Tao, Marcel Kunze. 2010. Virtual Data System on Distributed Virtual Machines in Computational Grids. International Journal of Ad Hoc and Ubiquitous Computing 6(4): 194-204.

[27] Wang, L., M. Kunze, J. Tao, G. von Laszewski. 2011. Towards building a cloud for scientific applications. Advances in Engineering Software 42(9): 714-722.

[28] Wang, L., G. Laszewski, D. Chen, J. Tao, M. Kunze. 2010. Provide Virtual Machine Information for Grid Computing. IEEE Transactions on Systems, Man, and Cybernetics, Part A 40(6): 1362-1374.

[29] Wang, L., Implementation and performance evaluation of the parallel CORBA application on computational grids. Advances in Engineering Software 39(3): 211-218, 2008.

[30] Wang, L., Kunze M., Tao J., Performance evaluation of virtual machine-based Grid workflow system. Concurrency and Computation: Practice and Experience 20(15): 1759-1771, 2008.

[31] Xiong, M., W. Cai, S. Zhou, M. Y. H. Low, F. Tian, D. Chen, Daren Wee Sze Ong and Benjamin D. Hamilton. A Case Study of Multi-Resolution Modeling for Crowd Simulation. In Proceedings of the Agent-Directed Simulation Symposium (ADS'09), March 22-27, 2009, San Diago, CA, USA.

[32] Yang, X., L. Wang, G. von Laszewski. 2009. Recent Research Advances in e-Science. Cluster Computing 12(4): 353-356.

[33] Zhou, S., D. Chen, W. Cai, L. Luo, M. Low, F. Tian, V. Tay, D. Ong and B. D. Hamilton. 2010. Crowd Modeling and Simulation Technologies. ACM Transactions on Modeling and Computer Simulation 20(4): Article No. 20.