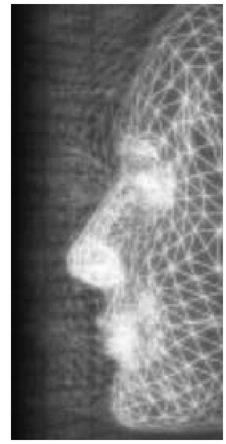
COMPUTER ANIMATION AND VIRTUAL WORLDS
Comp. Anim. Virtual Worlds (in press)
Published online in Wiley InterScience
(www.interscience.wiley.com) DOI: 10.1002/cav.196

Quartz: an autonomous navigation system for MOUT simulations



By Shang-Ping Ting and Suiping Zhou*

Autonomous navigation systems are important to Military Operations on Urbanized Terrain (MOUT) simulations for generating realistic tactical behaviours for the non-player characters (or bots). In this paper, we describe our work on Quartz, an autonomous navigation system for MOUT simulations. Novel features of Quartz include qualitative spatial representation and hierarchical spatial reasoning which enables fast situation analysis and human-like path planning in a dynamic environment. To assess the effectiveness of Quartz, we have integrated it into Twilight City, a virtual environment for MOUT simulations. Experimental results show that Quartz is very effective for quick tactical path generation in dynamic MOUT environments. Copyright © 2007 John Wiley & Sons, Ltd.

Received: 14 May 2007; Accepted: 15 May 2007

KEY WORDS: qualitative spatial reasoning; virtual environments; military operations on urbanized terrain (MOUT)

Introduction

> ''The contour of the land is an aid to the army; sizing up opponents to determine victory, assessing dangers and distances... those who do battle without knowing these will lose.''

The ancient statement of Sun Tzu[1] is especially true for Military Operations on Urbanized Terrain (MOUT).[2,3] Even modern armies may lose their advantage in superior air power and technology in hand-to-hand, street-by-street urban warfare in a foreign city.[4] Going into MOUT operations without extensive preparations may result in substantial military and civilian casualties.[5] As live MOUT training are expensive and are often restricted to the physical mock-ups of the buildings and other urbanized terrain features, building virtual environments for MOUT is becoming increasingly important.[6]

In MOUT simulations, the virtual urban environments are populated with characters and buildings. While human players control some characters, most of the characters are non-player characters (also known as *bots*) which demonstrate some level of human-like tactical behaviours, e.g., tactical path planning and movement. Therefore, autonomous navigation systems are crucial to MOUT simulations.[7]

The basic requirements of an autonomous navigation system in MOUT simulations are effectiveness, efficiency, scalability and adaptability. Effectiveness requires the navigation system to be able to generate realistic tactical movements for the bots in a complex environment. Efficiency requires the paths to be computed in real-time. Scalability requires the system to be able to accommodate a large number of bots. Adaptability requires the navigation system to be able to deal with the dynamic environment.

To address the above requirements, we have designed *Quartz*, an autonomous navigation system for MOUT simulations. The novel features of *Quartz* include the qualitative spatial representation and the hierarchical spatial reasoning.

Qualitative spatial representation describes the virtual environment by a number of regions, connections between different regions, key points, etc. As humans rely more on the qualitative relations among various objects rather than the exact quantitative descriptions of the world in their daily-life spatial reasoning, qualitative

*Correspondence to: S.-P. Ting, Modelling & Simulation, Defence Science & Technology Agency, Singapore 109679, Singapore. E-mail: tshangpi@dsta.gov.sg

Figure 1. Squad engagement formation.

spatial representation is believed to be more suitable to efficiently generate human-like movements for the bots.[8] For MOUT simulations, it is crucial to enable tactical awareness and fast terrain reasoning when planning paths for the bots. This is a challenging task which often involves many factors, e.g., the 3D environment, limited field of views, team formations, cover and concealment.[9–11] As an example, Figure 1 shows the formation of a squad that is firing at the enemy from different directions.

In general, path planning is a time-consuming task, especially for MOUT simulations as the environments are usually complex and highly dynamic. A pre-computed path may become invalid as the environment changes. For example, as shown in Figure 2, the path $A \rightarrow B \rightarrow C \rightarrow D$ that is deemed to be the best path at time

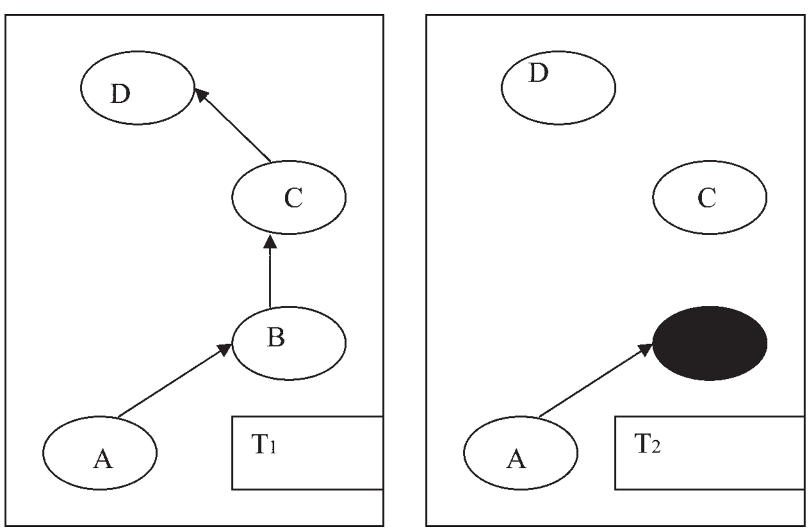


Figure 2. Path affected by dynamic obstacles.

DOI: 10.1002/cav

T1 is not the best path at time $T2 = T1 + \Delta T$, as node B becomes invalid at time $T2$. Thus the whole path from A to D is affected and re-planning is necessary. In this case, the computation of the whole path from A to D is wasted. Note that the likelihood of encountering dynamic obstacles along the path increases as the length of the path increases.

Quartz adopts a hierarchical spatial reasoning framework. At the higher level, only some key points connecting different regions are determined using the qualitative description of the map. At the lower level, the detailed path within a region is determined as the bot moves into the region with the heuristic A^* search algorithm.[12]

With this hierarchical spatial reasoning framework, *Quartz* is able to cope with dynamic environments efficiently. For example, if a path node is blocked along the detailed path in a region, only this detailed path needs to be re-computed; the general direction regarding how to move region by region from the starting point to the destination remains unchanged. Thus, the computational cost of path planning can be greatly reduced, which helps to improve the scalability of the navigation system. With its current implementation, *Quartz* can support up to a few hundred bots on a high performance personal computer.

The remainder of this paper is organized as follows. Firstly, we will provide an overview on the related work done on this topic. Subsequently, we will highlight the design considerations. After determining the design considerations, we shall describe how the *Quartz* system is integrated into *Twilight City*. Experimental results gathered from our performance tests will be discussed in the paper as well. Finally, our conclusions and discussion on future work will be covered in the last section.

Related Work

Autonomous navigation system has been studied extensively by researchers in the entertainment industry and the academia. Existing path finding algorithms such as the A^* algorithm[12] and the Dijkstra's algorithm[13] essentially aim at finding the shortest path in static environments. However these algorithms are not adequate to generate realistic movements for the bots in MOUT simulations mainly due to their high computational cost in large and dynamic environments.

Penn and Turner proposed an exosomatic visual architecture to model pedestrian walking behaviours.[14,15] However, we feel that their model is not adequate for tactical movements. In reference[16] Cohn suggested that it is relatively easy to translate common sense knowledge into some qualitative terms as compared to quantitative terms. This implies that qualitative spatial reasoning may be more suitable for generating realistic and human-like behaviours. How to efficiently generate human-like tactical behaviours for the bots attracts many researchers particularly in the game industry as an interesting and challenging problem. Reece, Kraus and Dumanoir proposed a multi-step path-finding process for tactical movement planning.[17] However, their approach is inadequate to deal with dynamic environments.

Voronoi diagrams have been proposed for spatial analysis in urban environments.[18] Schussman and Bertram further refined Voronoi diagrams to construct a hierarchical data structure for gridless, scattered data.[19] However, their approach was not used for path planning algorithms. In *Quartz*, layered Voronoi diagrams are used for spatial representation.

Design of *Quartz*

Quartz is designed to be an autonomous navigation system that is able to quickly generate human-like tactical movements for a large number of bots in a complex and dynamic virtual urban warfare environment. The two major features of *Quartz* are the qualitative spatial representation and the hierarchical spatial reasoning framework.

Why Qualitative Spatial Representation and Hierarchical Spatial Reasoning?

It has been suggested by some researchers that humans rely more on the qualitative description than the quantitative description of their environment when doing path planning in daily life.[8,20] Typical qualitative information includes location names, relative sizes, regions, relative relations among objects and relative distances, etc.[21] Although there are a number of path-finding algorithms for robots and agents, they are generally based on the accurate quantitative description of the environment. We believe it is necessary to model the human cognitive process in spatial reasoning in order to generate human-like behaviours for the bots in a complex virtual environment. In addition to using

DOI: 10.1002/cav

qualitative spatial descriptions, it seems to us that humans plan their paths in an incremental and hierarchical manner.

As an example, consider how a person describes the way to office from home. It is unlikely that the person knows exactly how many meters to move and along which exact directions. In fact, the person is more likely to describe the way in terms of a number of key points and their relations, e.g., ''first, I will go to the tall building at street S_1, then from there, I will go to the subway station at street S_2, . . ., finally I reach my office at location X!''.

The key spatial points and their spatial relationships can be easily remembered and analysed by humans. They essentially inform a person the *rough* path between two locations. However, when a person actually moves from his home to the office, he/she still needs to plan his/her *detailed* path along the way, e.g., how to move from home to the tall building at street S_1, etc. Note that a detailed path is often planned *only* when the person is on that specific section of the rough path. This incremental and hierarchical path-planning approach helps to reduce the amount of information that a person needs to remember, it also helps to deal with the dynamic environment.

Qualitative Spatial Representation in Quartz

In *Quartz*, we model the human cognitive process in spatial reasoning with a hierarchical multi-resolution spatial representation. As shown in Figure 3, different resolution levels are used to store the key points between A and B. A point at a higher level represents a group of points at the level below it, i.e., the resolution of the spatial description decreases as the level increases.

As path-finding within a dynamic environment is a computation-intensive task especially when the two locations are far apart, a hierarchical approach helps to divide the complicated problem into some manageable sub-problems, thus allows *Quartz* to handle long paths efficiently and elegantly. In *Quartz*, a virtual environment is described with a detailed map and a hierarchy of qualitative maps. Different representation and spatial reasoning techniques are used for these two kinds of maps. The detailed map uses visibility graph to link different points with edges, and it is used for low-level path planning. The qualitative maps use Voronoi graphs to segment the environment into regions, and they are used for high-level spatial reasoning.

Map Representation for Low-Level Path Finding

As shown in Figure 4, the detailed map consists of a large number of points that represent various positions in the virtual environment. An edge will be formed linking two points if there exists a line-of-sight between the two points and they are within a reasonable distance. This map is used by some low-level path-finding algorithms such as A^* to equip the bots with the low level navigation capability, such as moving from point A to point B.

Map Representation for High-Level Path Planning

At the high-level, the map is organized as a hierarchy of spatial regions using Voronoi graphs, as shown in Figure 5 and Figure 6. A point known as the region node represents a region in the map.

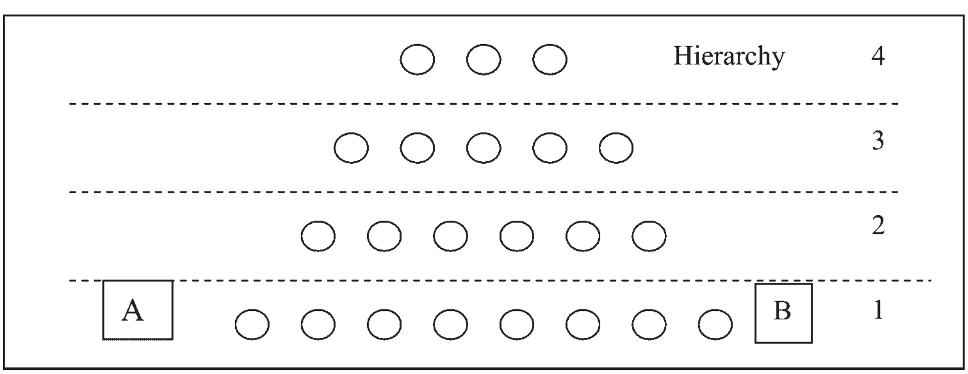


Figure 3. Hierarchical spatial representation.

DOI: 10.1002/cav

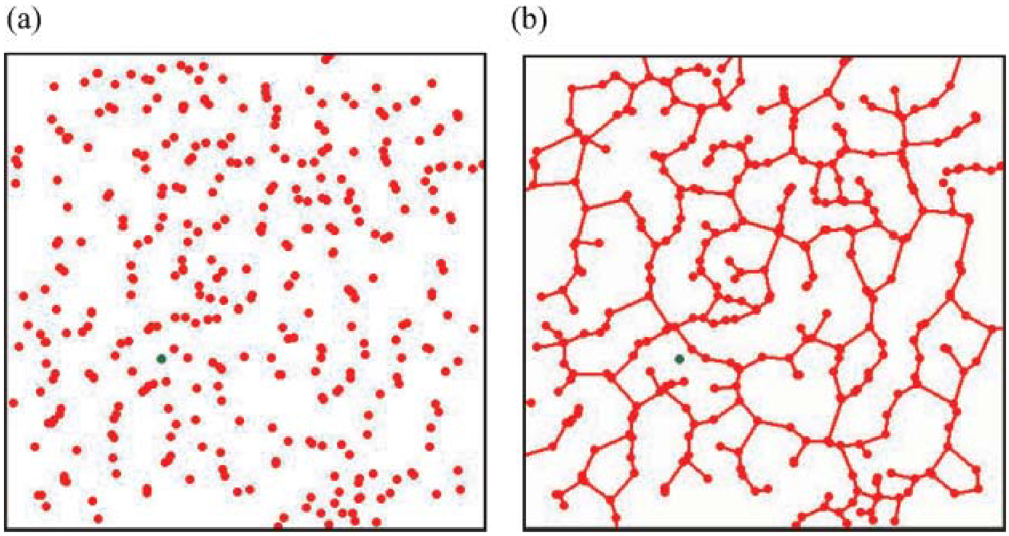


Figure 4. Visibility graph. (a) Points; (b) Points with edges.

In *Quartz*, the hierarchy of regions is built in a bottom-up manner. The Voronoi graph-based regions form the base layer (level 1) of the hierarchy. The next layer (level 2) in the hierarchy is built on the base layer by grouping the base regions into region clusters. Each region cluster is then treated as a new region and is represented by its region node. Figure 6 demonstrates the aggregation of the lower level regions to form higher-level regions. In figure 6, regions 3A and 3B are aggregated to form their *parent* region 4A. To represent this parent–child relation between regions, a hierarchical data structure is adopted in *Quartz*. As shown in Figure 7, the dashed line connecting two region nodes (at different levels) means the parent–child relation between the two regions. A solid line connecting two region nodes means that the two regions are adjacent to each other and they belong to the same parent region. A region node contains the information about how to move across its child regions. This hierarchical data structure provides important information for qualitative spatial reasoning.

Note that while the higher level regions are suitable for analysing long paths rapidly, the lower level regions can find short paths more accurately.

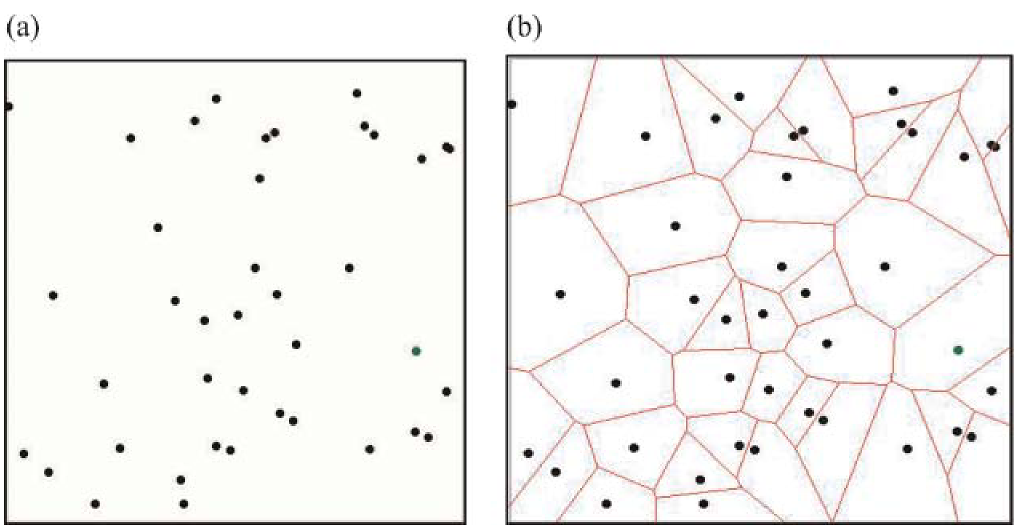


Figure 5. Voronoi graph and regions. (a) Region Nodes; (b) Regions.

DOI: 10.1002/cav

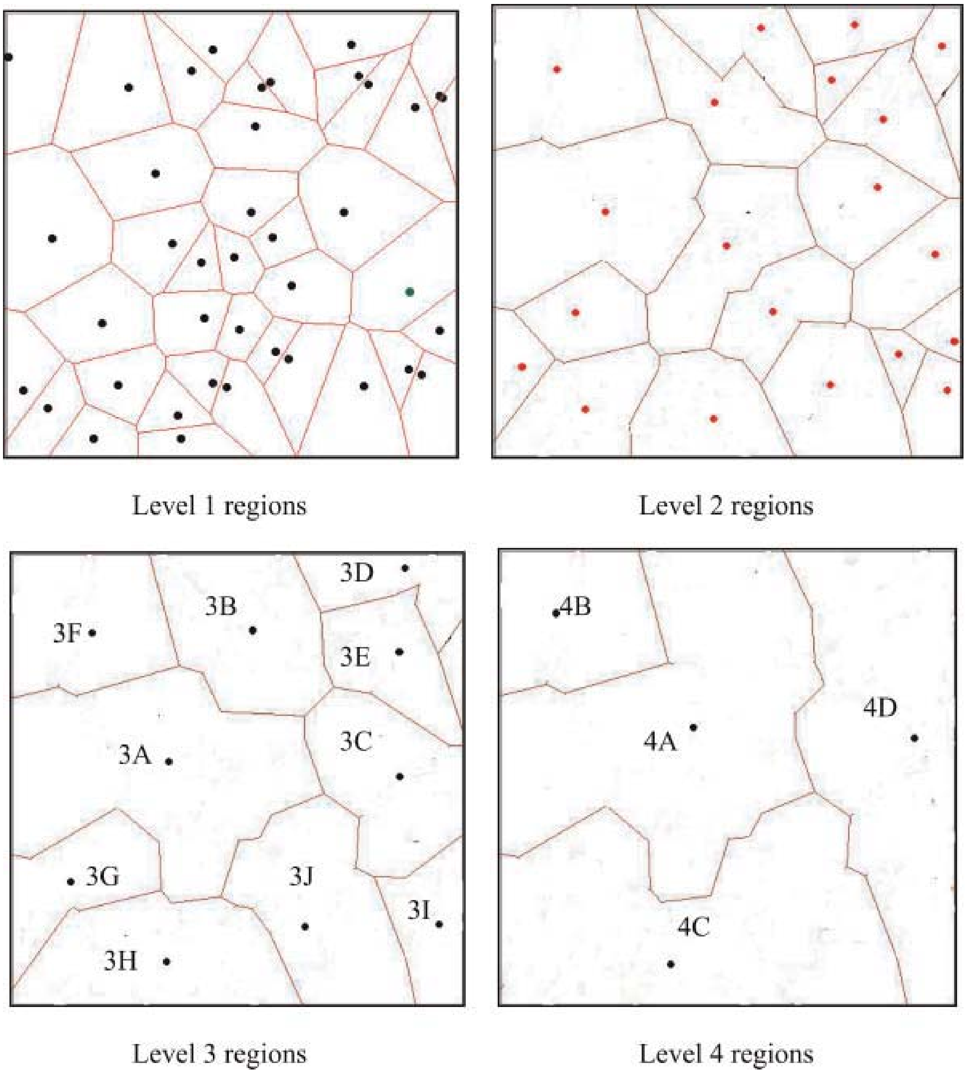


Figure 6. Hierarchy of qualitative spatial layers.

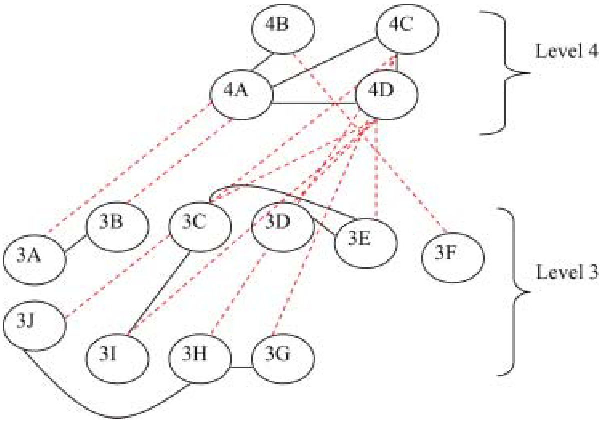


Figure 7. Hierarchical data structure in Quartz.

Qualitative Spatial Reasoning in Quartz

Quartz has a hierarchical qualitative spatial reasoning framework based on the hierarchical spatial representation as described in Section 3.2.

When doing path planning, *Quartz* first considers those paths between key spatial points if these points are connected by either the dashed lines or solid lines as shown in Figure 7. As a parent region node stores the information about how to move across its child regions, path planning is straightforward if the current region and the destination region belong to the same parent region. As an example, consider the path planning between a current location in regions 3C and a

DOI: 10.1002/cav

```
path QuartzPath(region a, region b){

//Let parent_a  & parent_b  be parent region of a and b;

if  (parent_a = parent_b)
    return a + parent_a + b;
return a + QuartzPath(parent_a , parent_b) + b;
}
```

Figure 8. QuartzPath algorithm.

destination in region 3D (see Figure 6). The path information on the possible routes between region 3C and 3D can be found from their parent region node 4D. To move across the child regions (at level 2 in this case) of region 3D, the path information on the possible routes leading towards the level 2 region containing the destination can be found from region node 3D. This process is repeated along the hierarchical levels (in descending order) until the destination point is found.

However, if the current and destination regions are not in the same parent regions, the path planning needs to move to the next higher level. Figure 8 illustrates the *QuartzPath* path-planning algorithm in *Quartz*. The recursive algorithm first checks if the two child regions have the same parent region, and adds the two child regions into the path plan if no common parent region is found. The *QuartzPath* algorithm terminates when a common parent region is found and returns the path plan.

With the hierarchical qualitative spatial reasoning framework, bots can incrementally plan their paths. This helps to reduce the computational cost of path planning. In addition, as bots do not commit to long paths, the impact of dynamic obstacles is greatly reduced.

Integrating Quartz Into *Twilight City*

In this section, we first give a brief overview of *Twilight City*, then describe the integration of *Quartz* into *Twilight City*.

Overview of *Twilight City*

Twilight City aims to provide a high fidelity simulation platform for MOUT simulations.[6] A typical scenario in *Twilight City* is the special squad operation to save hostage held in a building by some terrorists. The squad may consist of several human controlled characters and some AI-driven bots. The terrorists may also be some human-controlled characters and bots. *Twilight City* is built on top of the UT engine[22] with various modifications.

Twilight City simulates urban warfare in an area of approximately 20 km × 20 km. There are more than 30 buildings in the virtual environment. We have made various modifications to the UT engine to enhance the effectiveness of *Twilight City* for MOUT simulation. Figure 9 shows some screen shots of *Twilight City*.

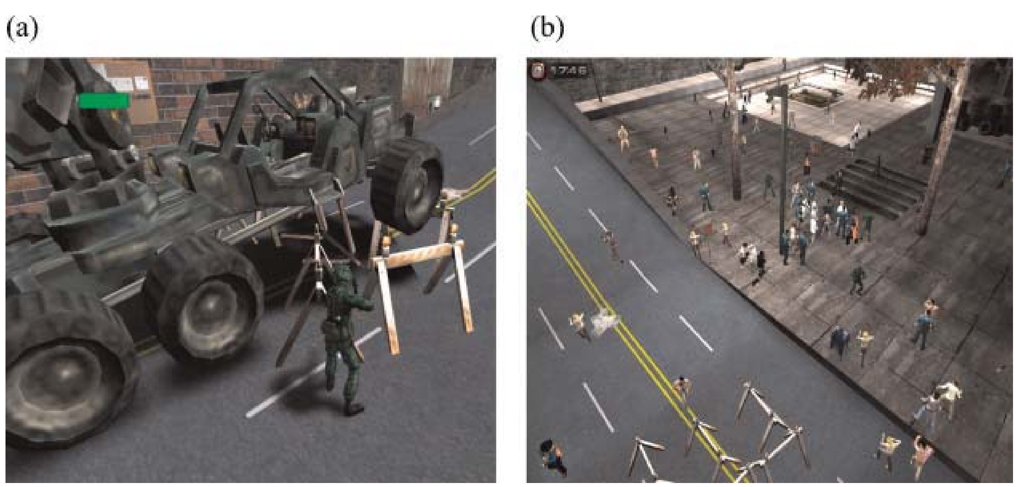


Figure 9. Twilight City. (a) Soldier with vehicle; (b) Bots.

DOI: 10.1002/cav

Spatial Representation in *Twilight City*

Quartz is used as the navigation system for the bots in *Twilight City*. As discussed in Section 3.2, a hierarchy of qualitative maps is used to describe the virtual environment. Voronoi graphs are used to form the base layer of the qualitative spatial regions. There are more than 200 spatial regions at the base layer of the *Twilight City* map. Each parent region holds between two to four child regions, thus five layers are used in the hierarchical map structure. It is noticed from Figure 10(a) that the centre of the map has the highest concentration of regions. This corresponds to the large number of the buildings at the center of *Twilight City* as shown in Figure 10(b). Figure 10(c) shows the visibility graph network that consists of nodes and edges. *Twilight City* contains more than 650 qualitative spatial points that act as nodes in the visibility graph network. These maps are used by *Quartz* for spatial reasoning and path planning.

Spatial Reasoning in *Twilight City*

Quartz's hierarchical spatial reasoning framework utilizes the hierarchical map structure and the information contained in the maps to build paths for the bots. For tactical path planning, apart from containing spatial information, qualitative spatial points also contain some

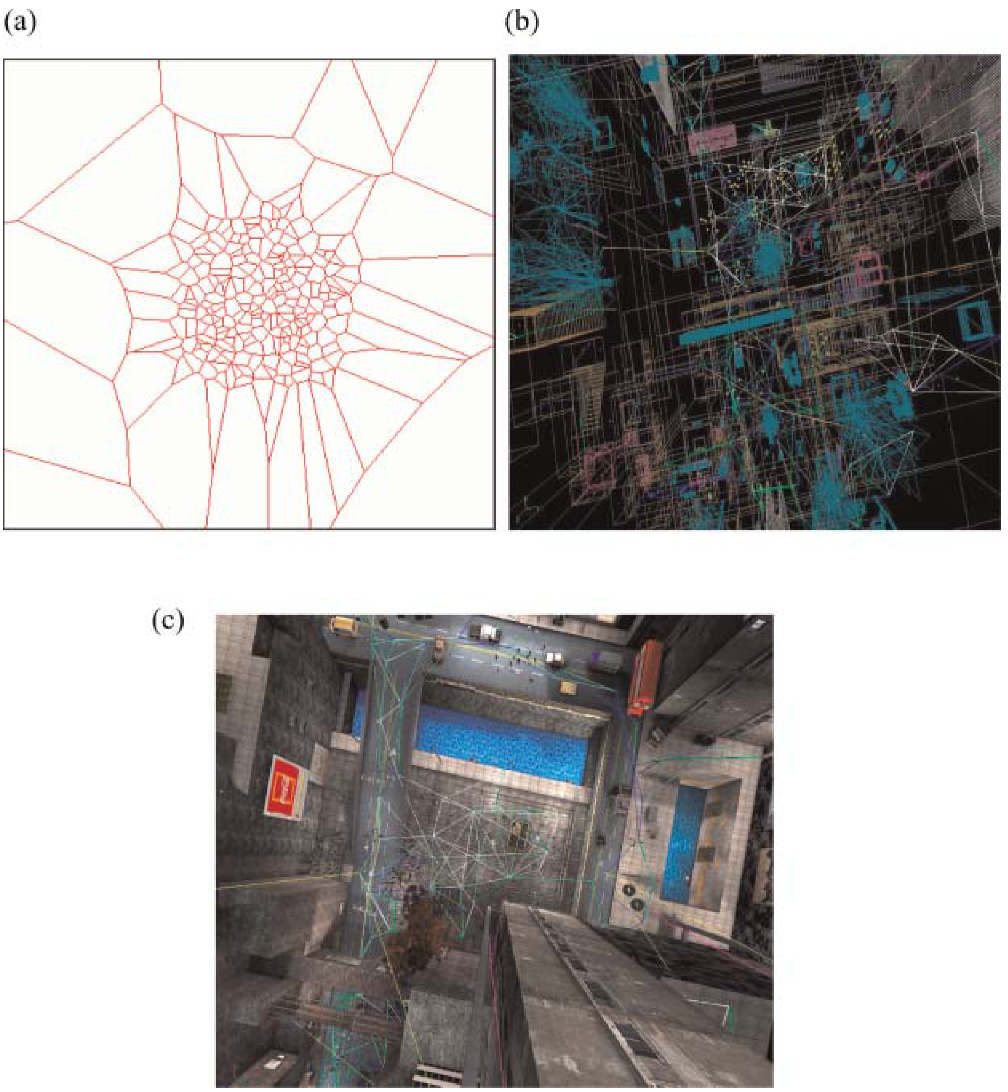


Figure 10. Maps in Twilight City. (a) Regions; (b) Centre of Twilight City; (c) Visibility graphs

DOI: 10.1002/cav

tactical information such as whether the point is a sniping point, ambush zone, friendly zone or enemy zone. Path information stored in the region nodes allows the bots to navigate rapidly towards child regions. These pre-computed data greatly speed up the path planning in *Twilight City*. In the current implementation, the number of levels in the map hierarchy is five. Thus, with the *QuartzPath* algorithm, the maximum number of qualitative spatial points in any path is only nine. Without using the hierarchical map structure, with most existing path finding algorithms, the number of spatial points in a path can easily exceed 30. As we mentioned before, long paths containing a large number of spatial points are computationally expensive and more susceptible to the changes in the environment.

Experimental Results

To evaluate the performance of *Quartz*, we have conducted a series of experiments. In this section, we summarize the major results of these experiments. These experiments were conducted on a computer with Intel T2500@2GHz processor and 2GHz RAM. The data shown in the results are the average of 10 runs.

To evaluate the efficiency and scalability of *Quartz*, we have developed a scenario that consists of more than 200 bots in *Twilight City* with three bots acting as terrorists, 10 bots as members of a special squad and remaining bots as civilians.

The time for 10 soldier bots to reach their destination with and without (i.e., simply using the A* algorithm and the detailed map) the *Quartz* system is recorded. Figure 11 shows the results. In Figure 11, the normal case means the implementation without using *Quartz*. As expected, the time taken for the squad to reach the destination increases as the number of civilian bots (thus the total number of bots) increases. However, it can be clearly seen that the recorded time for the implementation with *Quartz* increases much slower than the implementation without *Quartz*. This indicates that *Quartz* helps to improve the efficiency of the autonomous navigation system, and thus also improves the scalability of *Twilight City*.

To further evaluate *Quartz*'s capability in supporting real-time simulations, we have also conducted the frame rate analysis using the same scenario. Although the frame rate can be affected by a number of factors, the autonomous navigation system can significantly affect the frame rate. The results of the frame rate analysis are shown in Figure 12.

It can be seen that the frame rates for both implementations decrease as the number of bots increases. When the number of civilian bots increases beyond 64, the frame rate of the implementation without *Quartz* drops much faster than the implementation with *Quartz*. When the number of civilian bots hits 228, the frame rate of the implementation with *Quartz* is maintained at around 80, while the frame rate of the implementation without *Quartz* drops to around 30. The

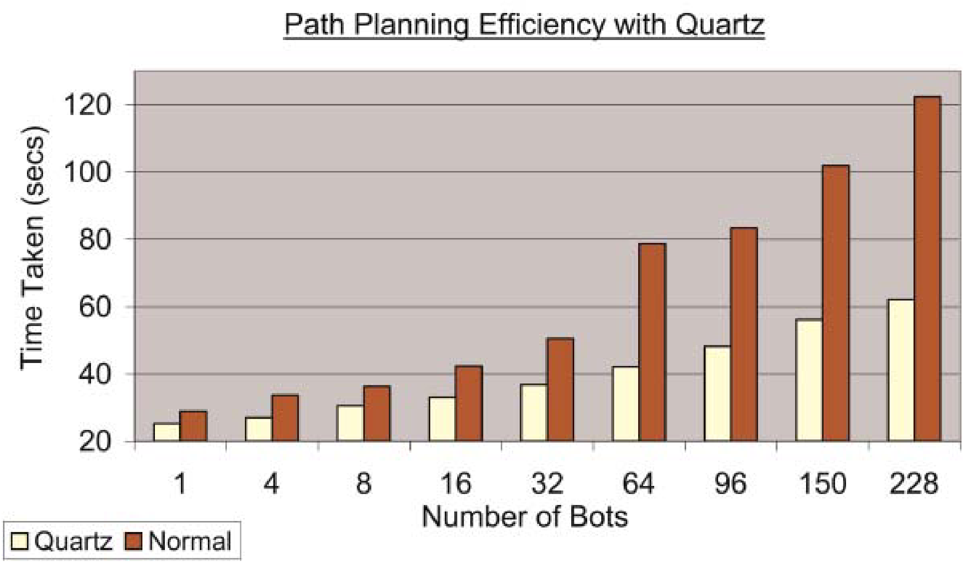


Figure 11. Impact of Quartz on path planning efficiency.

DOI: 10.1002/cav

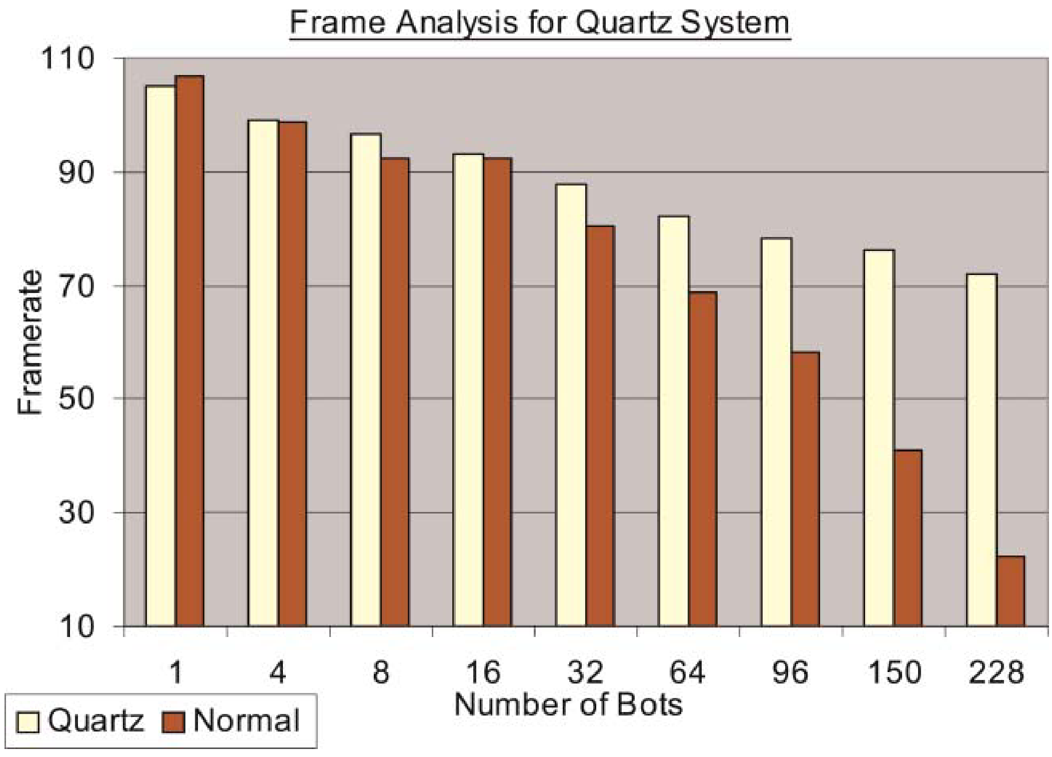


Figure 12. Frame rate analysis of Quartz.

Figure 13. Tactical path planning.

DOI: 10.1002/cav

results show that *Quartz* helps to reduce the computational overheads of the autonomous navigation system, and the benefit of *Quartz* becomes more evident as the number of bots increases. This makes *Quartz* particularly useful for large-scale MOUT simulations such as crisis management involving a large crowd in urban areas.

Figure 13 shows a soldier bot searching for some qualitative spatial points as defined by the maps of *Twilight City*. With the information stored in various spatial points, the bot is able to move to suitable positions to execute various tactical commands such as attack, defend, snipe, etc. It is also interesting to observe that some bots might get stuck at some locations without using *Quartz*, whereas there is no such case when *Quartz* is used. This indicates that the spatial information contained in the hierarchy of maps helps *Quartz* to plan path more effectively.

Conclusions and Future Work

Autonomous navigation systems are important to generate realistic behaviours for the non-player characters in virtual training systems and computer games. Our *Quartz* navigation system explores the human cognitive process in spatial reasoning. *Quartz* adopts qualitative spatial representations and hierarchical spatial reasoning. These two features make *Quartz* closer to the way that humans handle daily-life spatial problems. As humans rely more on the qualitative relations among various objects rather than the exact quantitative descriptions of the world in their daily-life spatial reasoning, we believe that qualitative spatial representation and reasoning are more suitable for generating human-like movements for the bots.

We will continue to work on the proposed qualitative spatial reasoning framework. In particular, qualitative reasoning techniques will be developed to induce common sense and situation awareness to the non-player characters.

References

1. The Art of War, translated by Lionel Giles, 1910.
2. *FM 90-10, Military Operations on Urbanized Terrain (MOUT)*. Headquarters, Department of the Army: Washington, D.C., 1979.
3. *FM 3-06, Urban Operations*. Headquarters, Department of the Army: Washington, D.C., 2003.
4. Stacey CP. *Official History of the Canadian Army in the Second World War Volume III: The Victory Campaign: The Operations in North West Europe 1944–1945*, 1960.
5. Ballard JR. *Fighting For Fallujah, A New Dawn for Iraq*, 2006.
6. Zhou SP, Ting SP, Shen ZQ, Luo LB. Twilight city — a virtual environment for MOUT. *International Journal of Computer and Applications (accepted)*.
7. Wray RE, Laird JE, Nuxoll A, Jones RM. Intelligent opponents for virtual reality trainers. In *Proceedings of I/ITSEC*, Orlando, Florida, 2002.
8. Forbus K, Mahoney JV, Dill K. How qualitative spatial reasoning can improve strategy game AIs. In *15th International Workshop on Qualitative Reasoning*, San Antonio, Texas, 2001.
9. Kamphuis A, Rook M, Overmars MH. Tactical path finding in urban environments. In *International workshop on crowd simulation,EPFL*, Lausanne, Switzerland, November 24–25, 2005.
10. Rook M, Kamphuis A. Eurographics/ACM SIGGRAPH *Symposium on Computer Animation*, Utrecht, 2005.
11. Crino ST. In *Winter Simulation Conference, Proceedings of the 33rd Conference on Winter Simulation*, Arlington, Virginia, 715–720, 2001.
12. Nilsson N. *Problem-Solving Methods in Artificial Intelligence*. McGraw-Hill: New York, 1971.
13. Cormen TH, Leiserson CE, Rivest RL, Stein C. Introduction to Algorithms, 2nd edition, 2001.
14. Turner A, Penn A. Encoding natural movement as an agent-based system: an investigation into human pedestrian behaviour in the built environment. *Environment and Planning B: Planning and Design 2002;* **29**: 473–490.
15. Penn A, Turner A. Space syntax based agent simulation. In *Proceedings of the 1st International Conference on Pedestrian and Evacuation Dynamics*, University of Duisburg: Germany, 2002. pp. 99–114.
16. Cohn AG. Calculi for qualitative spatial reasoning. In *International Conference on Artificial Intelligence and Symbolic Mathematical Computation*, Steyr, Austria, September 23–25, 1996.
17. Reece D, Kraus M, Dumanoir P. Tactical movement planning for individual combatants. In *Proceedings of the 9th Conference on Computer Generated Forces and Behavior Representation*, Orlando, Florida, 2000.
18. Chaimowicz L, *et al.* Deploying air-ground multi-robot teams in urban environments. In *Proceedings of the 2005 International Workshop on Multi-Robot Systems*, pp. 223–234, Washington DC, March 2005.
19. Schussman S, Bertram M, Hamann B, Joy KI. Hierarchical data representations based on planar Voronoi diagrams. In *Proceedings of VisSym'00*, Amsterdam, Springer, 2000.
20. Knauff M, Rauh R, Schlieder C. Preferred mental models in qualitative spatial reasoning: a cognitive assessment of Allen's calculus. In *Proceedings of the 17th Annual Conference of the Cognitive Science Society* (pp. 200–205), 1995.
21. Zimmermann K, Freksa C. Qualitative spatial reasoning using orientation, distance, and path knowledge. *Applied Intelligence* 1996; **6**: 49–58.
22. Unreal Engine, http://www.unrealtechnology.com/html/technology/ue2.shtm, May 15, 2007.

DOI: 10.1002/cav

Authors' biographies:

ShangPing Ting is currently an Engineer in the Modelling and Simulation Department at Defence Science Technology Agency (Singapore). He received his B.Eng. in Computer Engineering from Nanyang Technological University (Singapore). Currently, he is reading his Ph.D. in Nanyang Technology University. His current research interests include: qualitative physics, reusable software architectures, large scale virtual environments and human behaviour representation in modelling and simulation.

Suiping Zhou is currently an Assistant Professor in the School of Computer Engineering at Nanyang Technological University (Singapore). Previously, he worked as an engineer in Beijing Simulation Center, China Aerospace Corporation, and then joined Weizmann Institute of Science (Israel) as a Post-doctoral fellow. He received his B.Eng., M.Eng. and Ph.D. in Electrical Engineering from Beijing University of Aeronautics and Astronautics (P.R. China). His current research interests include: large-scale distributed interactive applications (e.g., MMOGs), parallel/distributed systems, and human behaviour representation in modelling and simulation. He has published over 50 technical papers in these areas, and has served as technical program committee member of several international conferences and workshops in computer games and virtual environments.

DOI: 10.1002/cav