

多人虛擬環境中虛擬演員的即時運動計劃 Real-Time Path Planning for Virtual Actors in a Multi-user Virtual Environment

雷嘉駿
國立政治大學資訊科學系
s9144@cs.nccu.edu.tw

李蔡彥
國立政治大學資訊科學系
li@nccu.edu.tw

摘要

在多人虛擬環境中，多數虛擬替身 (Avatars) 的運動都是由真實使用者所控制，其運動並非電腦所能掌控。因此，如何為電腦所控制的虛擬替身 (稱為虛擬演員)，規劃其不與環境發生碰撞的運動，是一個具挑戰性的計算問題。在本論文中，我們提出一種機率模型，用以評估不確定性障礙物 (如其他虛擬替身) 與虛擬演員碰撞的風險，進而以此機率模型為基礎，設計一個路徑規劃器，為虛擬演員規劃出免於碰撞的運動路徑。在虛擬演員移動的過程中，路徑規劃器會不斷地根據其他虛擬替身的位置，更新對不確定性障礙物的評估，進而調整路徑以降低碰撞風險。我們以實驗模擬方式，說明此計畫器的有效性，未來並將與 3D 多人虛擬環境系統結合，提高虛擬演員運動控制的真實性。

關鍵字: 不確定性障礙物、虛擬演員、路徑規劃器、動態虛擬環境、多人虛擬環境

1. 簡介

在多人虛擬世界中，使用者透過虛擬替身 (Avatar) 進入虛擬世界，進行各項與其他虛擬替身的互動。隨著應用需求的增加，虛擬世界的內容也越來越豐富，而使用者對虛擬環境界面的要求也越來越高，以期獲得身歷其境的感受。因此，多數的虛擬環境系統必須能模擬真實世界中的物理性質或滿足幾何限制。其中，避免與環境中的物體發生碰撞便是一項基本需求。

虛擬替身對虛擬世界的認知，可能是全部的、局部的或動態改變的。一般而言，虛擬環境中的障礙物可分為三類：一、場景中固定的障礙物 (static obstacles)，如花草、建築物等；二、電腦控制的動態確定障礙物 (certain dynamic obstacles)，如解說員等其他虛擬演員；三、其他使用者即時控制的動態障礙物 (uncertain dynamic obstacles)。上述三種物件的特性各有不相同；例如靜止不動的障礙物是沒有移動性、長時間固定在一個位置上；而電腦模擬的虛擬替身，如果是由同一個程式所控

制，則其虛擬替身的運動是可以掌握的，因此我們稱之為確定的可移動障礙物；最後，由代表真實使用者的替身，由於是由使用者所即時控制的，因此其未來的移動軌跡是無法明確掌握的，我們稱這類障礙物為不確定的可移動障礙物。

在各種障礙物皆存在的情況下，為電腦控制的數位演員規劃出一條免於碰撞的路徑，以到達目的地，就是本研究的目標。雖然我們無法確知不確定障礙物的未來位置，但我們假設這些障礙物過去的歷史軌跡為已知。在此不確定障礙物存在的動態環境中，我們將設計有效的即時運動計畫器，根據不確定障礙物的軌跡，評估未來路徑的風險，以計算出安全性較高的路徑。在執行此路徑的過程中，計畫器會根據不確定障礙物新的軌跡，局部更新路徑以降低執行風險，進而產生品質較佳和較符合真人運動的動畫。

2. 相關研究

在電腦動畫、機械人學以及人工智慧的領域中，行動機械人 (mobile robot) 廣泛地被應用在各種情境及產業裡；例如工廠的生產線、倉庫搬運、旅遊指南、視訊教學、水底探索、醫療設備等。在以上的應用中，行動機械人的運動規劃能力，對其是否能自主的完成任務，扮演十分重要的角色。

文獻中對運動計劃的研究，已經提出相當多的有效方法[8][11]。組態空間 (Configuration Space, C-Space) 是運動計畫演算法經常使用的觀念。所謂的組態空間是指描述機器人自由度的參數所構成的抽象空間。在此空間中，機械人與障礙物碰撞的組態，被紀錄為 C-obstacle 區域。如果我們可以計算此一空間，將可以把機器人視為一個點，在此空間中進行可行路徑的搜尋。然而，並非所有的 C-obstacle 區域皆容易計算，特別是對高維度的組態空間而言更是如此。其他有些方法是建基於人工位能場，在工作空間 (Workspace) 上定義位能函數，計算機械人的每個組態與目的地組態的相對位能，再根據位能的高低，向位能低的方向進行搜尋，以找出一連串連續的組態通向目的地[1]。此方法雖然會有局部最小值 (Local Minimal) 的問題出現，但已有多個改良方法被提出[2]。

組態空間是用來表示靜態的障礙物。對於環境

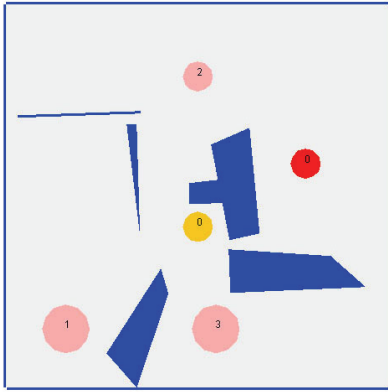


圖 1. 這是一場景，場景中有靜態障礙物(藍色)，不確定性障礙物(粉紅色)和機械人(黃色)，以紅色標示機械人的目的地。

中有動態障礙物的問題，我們常用「組態-時間空間 (Configuration-Time Space, CT-space)」來定義運動計畫的問題。在組態空間加上時間維度，可解決組態空間只能應用在靜態環境的問題[7]。一般而言，CT-space 只能在已知障礙物路徑的情況下使用；對如何於此空間中表示環境中不確定的障礙物，仍是一個研究的課題。另外，有些研究把問題分開成路徑規劃和速度規劃兩個階段進行。首先規劃出可行路徑，然後物體沿著路徑運動時，調整運動速度以避免與不確定性障礙物發生碰撞[10][4][5]。在[13]中，作者在起始位置和目的地之間規劃多個里程碑，再從中內插出可行路徑，把問題分解後再解決之。在[3]中，作者也提出其他類似的方法。另外，有些方法是把路徑規劃和運動控制組合在一個系統裡，首先以路徑規劃器計算出最佳路徑，再用運動控制器嘗試產生跟隨路徑或更新路徑[9]。在[12]中，作者引進機率模型在運動計畫中，預測障礙物和機械人的碰撞時間，從而改變機械人運動。

3. 機率模型

在這一節，我們將對問題描述及前提假設做完整的說明，然後我們將會討論不確定障礙物的未來運動軌跡對我們路徑規劃的風險影響，再提出解決方法的模型。

3.1. 問題定義和符號

我們假設多人虛擬環境可以適當的投影在一個 2D 的工作空間中，而障礙物是由多邊形所描述。為了減少機器人的自由度 (degrees of freedom) 以簡化計算的複雜度，我們假設虛擬演員可以用一個足夠大的圓形來描述。如圖 1 所示，四個圓形代表四個機械人 (編號分別是 0, 1, 2, 3)。虛擬演員的位置可以用 CT-Space 中的 $q = [x, y, t]^T$ 組態來表示， x, y, t 分別表示時間 t 時此虛擬演員座標系

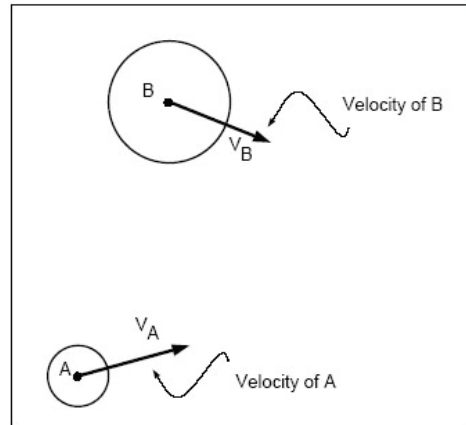


圖 2. 機械人 A 和不確定性障礙物 B

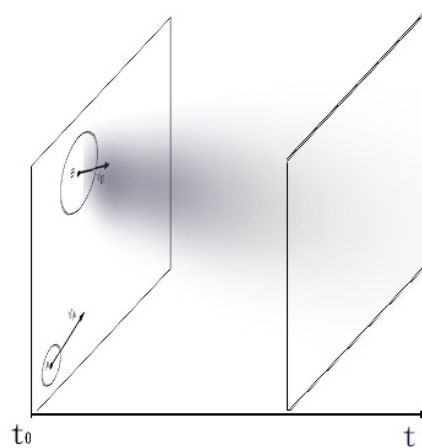


圖 3. 對應圖 2 場景的機率-時間圖。橫軸代表時間，顏色代表機率，顏色較深機率較大。

統原點在世界座標系統的位置 x, y 。我們假設運動規劃器能取得不確定性障礙物的軌跡 $P = [q_{t-n}, \dots, q_{t-1}, q_t]$, t 為目前的時間點, n 為紀錄的軌跡長度。對於未來的軌跡，我們則是無法明確知道。但是，根據障礙物的性質，我們可以期望不同的障礙物可能有不一樣的活動力；例如，代表機車和汽車的障礙物，因為機車比較靈活，所以代表機車的障礙物的角度變化也可能比汽車大。

如圖 2 所示，這裡的兩個圓形 (A 和 B) 分別代表虛擬演員和不確定性障礙物。在時間 t_0 時, A, B 的所在位置和速度向量分別記為 v_A, v_B 。假設 B 沿 v_B 方向移動的可能性很高，則 A, B 繼續沿目前的方向移動將很有可能發生碰撞。為使 A 的運動計畫產生的路徑合乎真人運動，我們希望建立機率模型來預測未來各個組態上 B 存在的機率。如圖 4 所示，我們將 B 未來可能位置的機率，以灰階顯示。而計畫器的目的在為 A 規劃出一條路徑，避開障礙物存在機率較高的部份。

3.2. 機率分配模型

為了預測障礙物的未來的位置，我們根據歷史

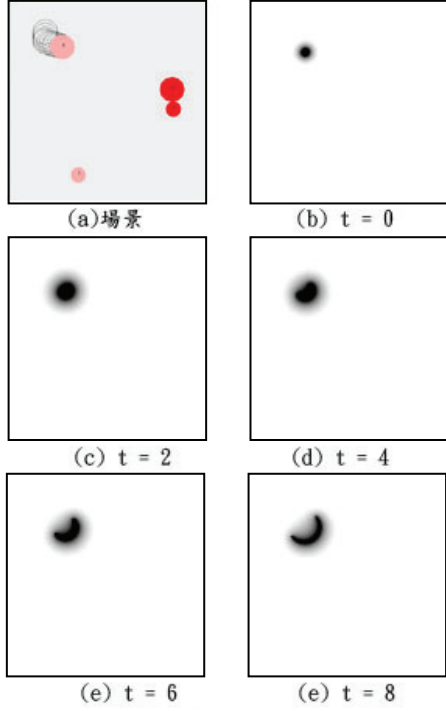


圖 4. 預測障礙物位置的機率圖，顏色越深表示機率越大。

軌跡推算出障礙物的預期速度 v_p 。

$$v_p = v + \sum_{i=0}^{n-1} \frac{\Delta v_i}{2^i} \quad (1)$$

在式(1)中， Δv_i 表示軌跡 P 中 $[q_{t-i}]$ 與 $[q_{t-1}]$ 之差。 v_p 是表示瞬時速度 v 與歷史軌跡加權後的變化量之和。在變化量的計算中，越近的變化越重要。在理想的情況下，我們預測障礙物會沿 v_p 的方向移動，從目前的位置 q_c ，到達預測的位置 q_p 。

$$q_p = q_c + v_p * \Delta t_i \quad (2)$$

Δt 越大，我們預測障礙物的位置越遠，其不確定性也就越大。我們假設不確定障礙物可能的位置分佈，是以推估的位置為中心，呈高斯常態分佈 [9]。換言之，我們是以此預測的位置，做為測量障礙物存在機率的隨機變數 x 。我們假設此高斯分佈的標準差與時間的大小成正比，則經過 i 步後的標準差 σ_i 如公式(3)所示：

$$\sigma_i = \sigma_0 * \alpha(i) \quad (3)$$

其中， $\alpha(i)$ 是一個標準差放大的函式。針對一個特定時間點，我們可以藉由下列公式，計算不確定障礙物可能位置的機率圖。

$$p(x, i) = \frac{1}{\sigma_i \sqrt{2\pi}} \exp\left(-\frac{\|x - q_p\|^2}{2\sigma_i^2}\right) \quad (4)$$

公式(4)顯示隨著時間的增加，預測位置的標準

差會越來越大，也就是物體的預測會越來越不準確，而組態空間上的機率分配也將會越趨平均。

圖 4 為我們根據此機率模型對場景中的不確定障礙物所計算出的若干張機率圖。圖 4(a) 為我們虛擬世界的場景，共有兩個虛擬替身機械人。位於左下方的機械人為我們所控制的虛擬演員，而位於左上方的機械人為不確定性障礙物。在圖 4(a) 的場景中，我們也顯示用以預測不確定障礙物未來位置的軌跡。圖 4(b)~(f) 顯示不同時間 t 時，不確定障礙物在場景上的機率圖；圖中顏色越深，表示機率越大。

3.3. 風險評估

對於不確定物體，我們用以下兩項性質區別其風險的高低：(一) 若距離越接近則風險越大；(二) 若不確定物體和機械人的相對移動使距離接近則風險越大。所以我們根據這些特性定義以下風險函數：

$$Risk(q) = w_1 d + w_2 (v_r - v_o) \cdot \left(\frac{q_r - q_o}{|q_r - q_o|} \right) \quad (5)$$

d 為機械人與障礙物之間的距離， $v_r - v_o$ 為相

對速度， $\left(\frac{q_r - q_o}{|q_r - q_o|} \right)$ 為相對位置的單位向量，

$(v_r - v_o) \cdot \left(\frac{q_r - q_o}{|q_r - q_o|} \right)$ 的意義是單位時間內相對距離的變化量，而 w_1 與 w_2 為以上兩項的權重。

我們評估一個障礙物在 i 步以後對虛擬演員的影響，是在障礙物可能出現的範圍內平均取樣若干個組態，使用公式(2)和公式(5)計算各組態的機率和風險值後將兩者相乘，代表障礙物在 i 步後出現在這個組態的影響值。最後，所有取樣組態的影響值的平均，就是障礙物在 i 步後對虛擬演員的風險影響值。

4. 產生路徑

4.1. 路徑規劃

為了同時考量靜態障礙物和動態障礙物，我們將此路徑計畫的問題定義在虛擬演員的 CT-space 中。我們先使用 NF1 演算法 [11] 為靜態障礙物計算出一個靜態的虛擬位能場，以做為指導搜尋的經驗法則 (Heuristic)。對於動態確定障礙物，我們把障礙物的路徑記錄在 CT-space 中，以做為碰撞偵測的依據。對於不確定障礙物，我們使用上一節中介紹的機率分配模型來評估虛擬演員在各組態的風險值，進而改變該組態的位能。我們是以最佳優先 (Best-First Search) 演算法 (如圖 5 所示) 的方式在 CT-space 中搜尋路徑。

Algorithm: Uncertain_BFPInput: q_{init}, q_{goal} .Output: path P

1. Initialize a priorityQueue Q
2. Insert c_{init} into Q
3. **while** Q is not empty
4. **begin**
5. $q = \text{Dequeue}(Q)$
6. **while** every neighbor q' of q that is unvisited
7. **begin**
8. $U(q') = \text{potential}(q') + \text{Risk}(q')$
9. Insert q' into Q
10. **If** q' is q_{goal} **then**
11. **return** the path by tracing from q' back to q_{init}
12. **end**
13. **end**
14. **return** failure;

圖 5. Uncertain_BFP 路徑搜尋演算法

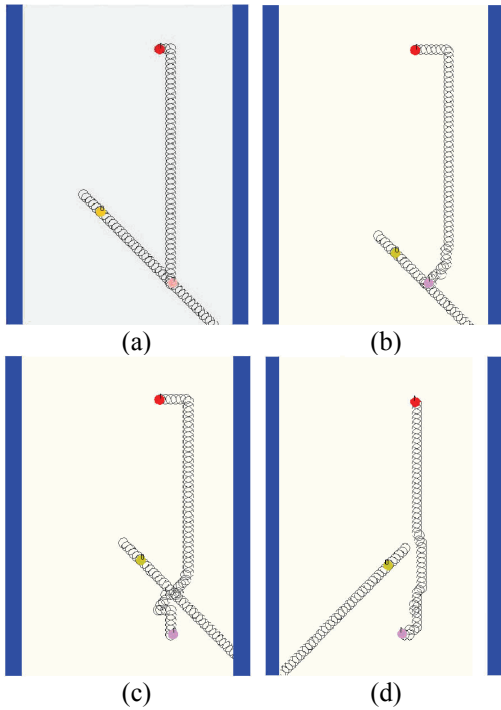


圖 6. 障礙物在不同位置不同移動方向時對路徑規劃的影響

搜尋是從起始組態 q_{init} 開始，先將 q_{init} 放進佇列 Q 中並記錄為已拜訪過。然後，在搜尋的每一個迴圈中，我們從 Q 中取出加權位能值最低的組態 q ，再拜訪其相鄰且沒有被拜訪過的組態 q' 。若 q' 不為目的地，則以加權位能值為鍵插入 Q 中；若 q' 為目的地 q_{goal} ，則回傳從 q_{init} 到 q' 的路徑 P 。如果在搜尋過程中若 Q 耗盡則結束搜尋，表示沒有路徑能從起始組態到達目的組態。

圖 6 表示機械人在路徑規劃時，不確定障礙物在不同位置、不同移動方向對我們路徑規劃的影

Algorithm: Update_PathInput: P, l, r

1. $S = \text{Perturb_Path}(P, l, r)$
2. **if** S is not **nil** **then**
3. replace path[$l \sim r$] with S
4. **else**
5. $m = (l + r) / 2$
6. Update_Path(P, l, m)
7. Update_Path(P, m, r)

圖 7. 更新路徑演算法

Algorithm: Perturb_PathInput: P, l, r .Output: path segment S .

1. $r_1 = \text{averageRisk}(P)$
2. **while** getNewMidPoint(m') is not **nil**
3. **begin**
4. pre = interpolation($P[l], m'$)
5. post = interpolation($m', P[r]$)
6. $S = \text{merge}(\text{pre}, \text{post})$
7. $r_2 = \text{averageRisk}(\text{segment})$
8. **if** $r_2 \leq r_1$ **then**
9. **return** S ;
10. **end**
11. **return** **nil**;

圖 8. 修改路徑演算法

響。在圖 6(a)~(d) 中，我們的虛擬演員以相同的起始及目的位置作路徑規劃。在圖 6(a) 和圖 6(b) 中的例子，在顯示障礙物在不同位置但相同方向時對規劃出的路徑的影響。圖 6(a) 中障礙物距離較遠，所以沒有明顯的影響；相對而言，圖 6(b) 中障礙物距離較近，所以規劃出的路徑先往右上方移動，再移向目的地。圖 6(c) 和圖 6(d) 中，障礙物在相同位置但以不同方向前進。圖 6(c) 的障礙物移動方向往虛擬演員方向移動，所以規劃的路徑影響較大。圖 6(d) 中障礙物以遠離機械人的方向移動，所以影響較少。

4.2. 路徑修改

在規劃好路徑後，電腦控制的虛擬演員會根據此一路徑移動。但是，我們無法保證此路徑在動態障礙物的移動下，將來完全不會發生碰撞。為此，在虛擬演員移動的過程中，我們不斷更新環境資訊（障礙物的位置和移動方向），局部修改路徑，增加路徑執行的安全性。

我們更新路徑的演算法如圖 7 所示。此演算法進行遞迴呼叫把路徑分解再修改。輸入為 P 的一段子路徑。 l 為欲修改子路徑的起點， r 為終點。首先我們修改 l 到 r ，若能找出一條較安全的子路徑，則代換原來的子路徑，否則分解成兩小段再修改。

找出較安全的子路徑的演算法如圖 8 所示。此

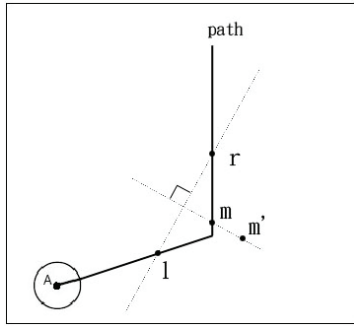


圖 9. 從組態 l 及 r 之間找出 m'

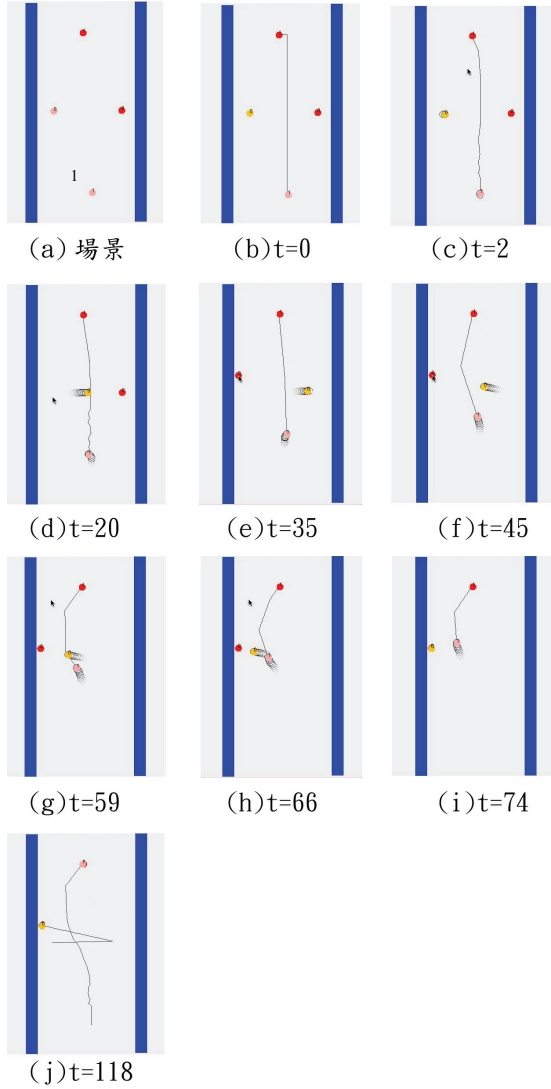


圖 10. 路徑規劃實驗的範例

演算法是先計算 P 的平均風險值，然後根據 l, r 的中點 m 向內或向外延伸得到另一點 m' (如圖 9 所示)，再使用內插法產生一段從 l 到 m' 再到 r 的路段 S 。如果這個路段的平均風險值比原來的低，表示成功，否則失敗。當 P 完全更新以後，我們會比較平均風險值和路徑長度與原平均風險值和原路徑長度，以決定修改是否合理。

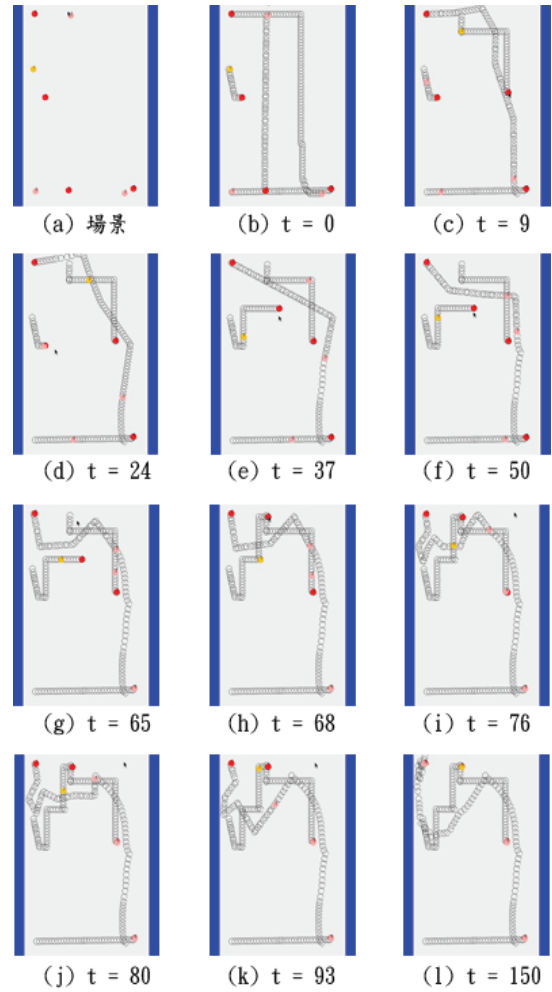


圖 11. 另一個路徑規劃實驗的範例

以上述演算法更新的路徑，會有不圓滑或很多轉折的現象，但是我們在下節的實驗結果中顯示，此路徑在執行過程中，會一直被更新而減少轉折的現象。

5. 實驗結果

圖 10 是我們使用上述方法所產生的路徑規劃的例子。圖 10(a)為機械人所處的場景，藍色標示為靜態障礙物，紅色標示為機械人的標的地。圖中位於左方的是不確定障礙物 (編號 0)，位於下方的是我們控制的虛擬人物 (編號 1)。各圖中均顯示出它們過去的路徑和目的地以方便觀察整個過程，其中 t 代表經歷的時間。圖 10(b)是我們一開始所規劃出的路徑，由於障礙物很遠，影響不大，所以路徑直線進行。圖 10(c)和圖 10(d)顯示出障礙物越來越接近時所作出的路徑修改。圖 10(e)為當 $t=29$ 時，不確定障礙物的影響變低，路徑修正趨於直線且不確定障礙物改變其目的地。圖 10(f)和圖 10(g)顯示機械人的原路徑因為障礙物的移動方向改變，使得平均風險值變大，所以對路徑作出修

改。圖 10(h)為當 $t = 66$ 時，不確定障礙物在機械人面前經過且開始遠離。圖 10(i)顯示虛擬演員對路徑作出修正。圖 10(j)顯示虛擬人物從起點走到目的地的完整路徑。

圖 11 是我們另一個路徑規劃的例子。圖中編號 0, 1, 2 的機器人是不可確定障礙物，位於右下方的編號 3 的機械人是我們所控制的虛擬演員，如圖 11(a)所示。圖 11(b)是我們一開始各機械人規劃的路徑。圖 11(c)顯示機械人 2 改變它的目的地。圖 11(e)顯示機械人 1 選擇一個新的目的地繼續移動。圖 11(h)顯示機械人 1 改變目的地。經過一系統的路徑修改後，虛擬演員在 $t = 150$ 時到達目的地。

6. 結論

在本論文中，我們提出一個新的虛擬演員運動計劃與更新的方法。我們預測可移動障礙物將來的位置和速率，評估演員未來在各位置上的風險值，進而找出一條風險較低且能到達目的地的路徑。隨著時間及環境的改變，我們每走一步後便檢討一次路徑的風險值，在局部的範圍裡，調整路徑以達到局部的最安全路徑。未來我們希望此研究成果能與 3D 虛擬環境系統相結合，以進行虛擬演員模擬的實驗。

7. 致謝

感謝 IM-lab 全體成員的建議。另外，此研究在國科會計畫（編號[NSC94-2815-C-004-008-E] 及 [NSC94-2213-E-004-006]）的支助下完成，特此致謝。

參考文獻

- [1] J. Barraquand, B. Langlois and J.-C. Latombe, “Numerical Potential Field Techniques for Robot Path Planning”, in *IEEE Transactions on Systems, Man and Cybernetics*, pages 224-241, 1992.
- [2] V. Boor, M. Overmars and A.F. van der Stappen, “The Gaussian Sampling Strategy for Probabilistic Roadmap Planners”, in *Proc. of the 1999 Intl. Conf. on Robotics & Automation*, 1999.
- [3] O. Brock and O. Khatib, “High-speed Navigation Using the Global Dynamic Window Approach”, in *Proc. of the 1999 Intl. Conf. on Robotics and Automation*, 1999.
- [4] P. Fiorini and Z. Shiller, “Robot Motion Planning in Dynamic Environments”, in *Proc. of the 1995 Intl. Symp. of Robotic Research*, 1995.
- [5] P. Fiorini and Z. Shiller, “Motion Planning in Dynamic Environments using Velocity Obstacles”, in *Intl. J. of Robotic Research*, 1998.
- [6] M. Fisz, *Probability Theory and Mathematical Statistics*, Wiley, 1963.
- [7] Th. Fraichard, “Trajectory Planning in a Dynamic Workspace: a ‘State-Time Space’ Approach”, in *Intl. J. of Advanced Robotics*, pages 75-94, 1999.
- [8] Y. K. Hwang and N. Ahuja, “Gross Motion Planning”, in *ACM Computing Surveys*, pages 219-291, 1992.
- [9] L. Ikemoto, O. Arıkan and D. Forsyth, “Learning to Move Autonomously in a Hostile World”, in *Proc. of the 2005 Intl. Conf. on Electrical Engineering and Computer Sciences*, 2005.
- [10] K. Kant and S. W. Zucker, “Towards Efficient Trajectory Planning: the Path-Velocity Decomposition”, in *Intl. J. of Robotic Research*, pages 72-89, 1986.
- [11] J. Latombe, *Robot Motion Planning*, Klumer, Boston, MA, 1991.
- [12] J. Miura and Y. Shirai, “Probabilistic Uncertainty Modeling of Obstacle Motion for Robot Motion Planning”, in *Robotics and Mechatronics*, pages 534-541, 2002.
- [13] J. Zhang and A. Knoll, “An Enhanced Optimization Approach for Generating Smooth Robot Trajectories in the Presence of Obstacles”, in *Proc. of the 1995 European-Chinese Automation Conf.*, England, 1995.