

## 智慧型清掃機器人清掃效率分析的實境模擬系統

許育塵、余志成\*

國立高雄第一科技大學 機械與自動化工程系

\*通訊: jcyu@nkfust.edu.tw

### 摘要

本文探討一個動態模擬環境的建構，作為智慧型清掃機器人清掃效率的評估。由清掃機器人的控制邏輯規劃，配合環境感測器與障礙感測器的應用，分析機器人的清掃策略對未知環境的覆蓋率與清掃效率。典型清掃機器人的清掃策略是以數個清掃模態循序清掃，其基本清掃模態包括隨機、螺旋、循邊與之字型等路徑，配合機器人上的感測裝置，使機器人可自在室內空間中行走與清掃。本研究利用動態模擬軟體(ADAMS)所提供的模組指令及感測器的設定，建構出虛擬環境與機器人的感測裝置，以實現機器人的運動動作及清掃路徑模擬，並由ADAMS 模擬出來的路徑，透過Visual C#所設計的介面程式，分析清掃機器人在環境中的覆蓋率與清掃效率，以作為清掃策略的效率評估與改進。

**關鍵字：** 清掃機器人、清掃模擬、覆蓋率、清掃效率、ADAMS

### 1. 前言

市售清掃機器人的結構及功能差異不大，其基本結構包括一基座，並於基座中設置一組移動的動力裝置，及可吸塵與集塵的裝置模組，或在基座底端增設滾筒型的拭塵紙及旋轉型的毛刷以提升清潔效果，同時加裝感測器並配合清掃路徑的控制設計，使其可自在室內空間中依行走路徑的規劃，如隨性自走、螺旋行走、靠牆行走或Z字型的移動，而達成在室內的清掃工作。其清掃效率將會受到這些清掃模態的切換時機與時間，以及在感應到各種不同障礙時的反應方式，尤其在清掃某些特定障礙環境時，會發生侷限在局部區域清掃而無法脫困的現象，或是有多個隔間的複雜環境，在指定清掃時間內無法完成全域清掃的問題，因此針對產品機構與障礙感測裝置，進行實機測試就變得非常重要。

機器人的清掃性能可由兩個指標來評估：覆蓋率與清掃效率，覆蓋率是指機器人在可清掃面積內已經清掃過面積之百分比，而清掃效率則是指時間內之覆蓋率。一般清掃效率評估的方法多需以實景時機搭配攝影機進行，利用影像分析評估清掃效率[1][2]，不僅成本昂貴，且不易測試不同環境與變更清掃控制程式，分析環境受限於攝影機可拍攝的範圍，所以一般

都只能從事簡單的單一房間，如4x5(m)的環境，無法進行多房間清掃的分析。

因此建置一個機器人的模擬環境在清掃效率的評估上變得非常重要，現有模擬軟體如Webots[3]常用來作為移動式清掃機器人的模擬[4][5]可建置本文以ADAMS模擬分析軟體進行虛擬環境及機構的建置，藉由感測模擬與清掃控制邏輯的建置，模擬清掃機器人於所建置之虛擬環境下的清掃路徑，並經由後處理的覆蓋率分析，作為清掃策略的評估。

### 2. 清掃機器人的系統架構

清掃機器人主要的組成有運動單元、感測單元、跨障單元與清潔單元如圖1所示。在平面移動住要藉由兩側驅動輪的轉速控制來達到移動與轉向的目的。清掃機器人在動作控制方面，主要區分為以下的幾種清掃模態：螺旋(Spiral)、循邊(Wall-following)、牛耕(Zigzag)、隨機路徑(Random)。(1)螺旋是以逐漸放大的圓弧半徑從內向外圓弧轉，用於局部較骯髒地區的清掃。(2)循邊的方式是以紅外線控制，以固定距離平行牆面移動。(3)牛耕路徑清掃模態，在未觸碰到障礙物時，以直線前進，直到遇到障礙時原地轉90度，直行一段距離後再原地轉90度，而後反覆執行以達到牛耕路徑。(4)隨機路徑模態以直線前進，直到遇到障礙並觸發碰撞感測器，機器人後退一段距離，之後旋轉一角度離開障礙物，而後反覆障礙偵測、轉向、直行的動作。

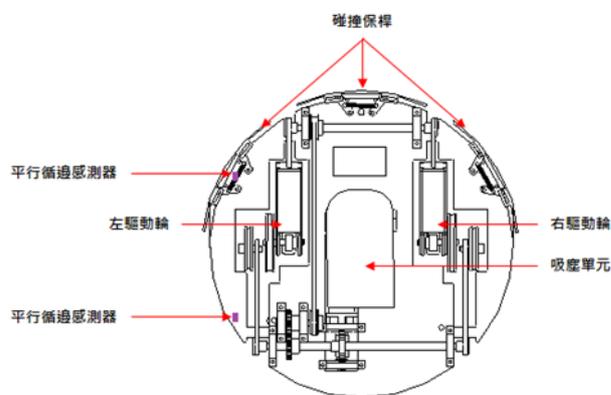


圖1. 清掃機器人構造圖

### 3. 實境模擬系統的建置

本文利用動態模擬軟體 ADAMS (Automatic Dynamic Analysis of Mechanical Systems)所提供的模組建構虛擬環境與機構，並進行模擬與分析。

#### 3.1 運動機構與環境建置

首先匯入清掃機器人3D模型，並進行接頭(Joint)設置，限定其物件的運動方式及自由度，建構出清掃機器人的運動機構，以Tire模組建構清掃機器人輪胎與路面，減少模擬所需運算時間，並進行驅動器(Motion)設置，作為清掃機器人傳動系統。機器人是藉由左、右輪兩顆馬達來控制進行動作如前進、後退及轉彎等動作；圖2為Motion的配置圖，MOTION/1與MOTION/2 分別表示機器人的左驅動輪與右驅動輪，可由Motion 視窗定義相關參數，語法格式為每秒的旋轉角度，以右手定則定義方向。

根據環境的需求，建構出環境地圖，如單一房間、兩房兩廳之環境，使清掃機器人在設計的環境中做模擬，並利用碰撞(Contact)模組的功能來設定機器人與障礙物之間的碰撞關係。

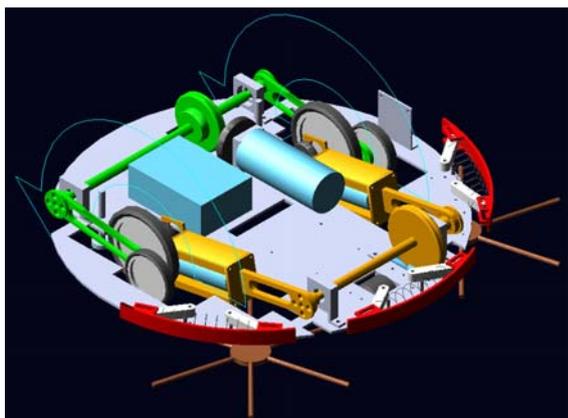


圖2. 機器人 Motion 配置圖

#### 3.2 感測器設定

清掃機器人在任一環境行進時，常需依賴感測器做環境障礙的辨識，因此經由ADAMS 中感測器(Sensor)的功能，建立系統模擬行為的控制機制，使ADAMS在觸發預先設定之感測器條件，停止原先動作並執行後續特定動作。

##### 3.3.1 碰撞感測設定

如圖3所示，擺臂與碰撞保桿利用插銷與滑槽的關係固定在一起，同時擺臂受到彈簧的拉力使擺臂抵在固定座，而擺臂上有設定旋轉接頭，因此當碰撞保桿受到障礙物的擠壓時，擺臂會做圓周運動，並帶動碰撞保桿觸發感測器。左、右感測器的機制為極限開關，所以將其概念運用在ADAMS 設定中。作法是利用兩個Marker 的距離作為定義感測器觸發的條件，當量測距離小於門檻值時，即可定義此感測器為觸

發。

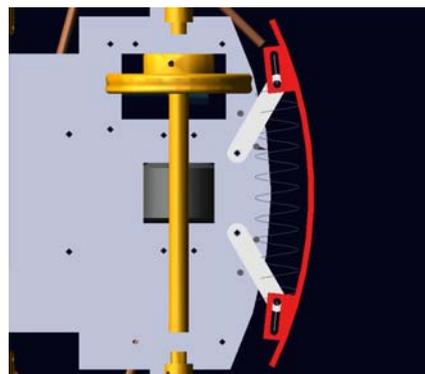


圖3. 碰撞感測器

##### 3.3.2 平行牆面感測設定

機器人在運動時會沿著牆壁行走，可用於尋找清掃牆角的灰塵與移動到下個區域等，本研究在機器人左側設定了兩個Marker (IRW0 與IRW1)，代表兩個平行循邊的紅外線距離感測器(圖4)，利用IRW0 與IRW1到牆面的距離值做為機器人平行牆面的依據，所以在每個牆面設定兩個Marker，就可利用方程式求解感測器到牆面的距離。

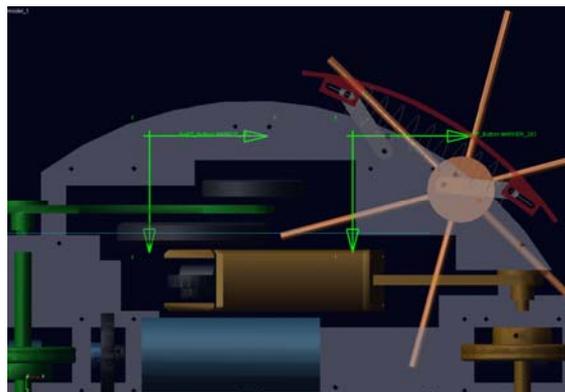


圖4. 紅外線感測器

#### 3.4 感測器設定

為便於之後的清掃模態控制，我們將運動模態撰寫成模組化的指令，利用程序腳本(Simulation Script)的功能來進行，Script設定可在模擬過程中改時間間距、求解方式或模型的架構等，使模擬工作能有更多的彈性與變化。

Script可由Adams/solver裡的command寫成ACF檔，進行模擬設定或改變模型的狀態如Motion 的參數值，並配合Fortran所編譯之使用者定義子程序(consub) 作為機器人控制邏輯，如圖5其語法為control/fun=user(r1,r2,r3,r4,r5...); r1是用來指定運動的方式，若r1=1~4為機器人的基本運動，如前進、後退、左轉及右轉；r1=21~24為進行模態清掃，如螺旋、循邊、牛耕與隨機路徑；r1=31為進行策略清掃。r2~r4為MOTION/1~3的旋轉接頭，其用來得知改變驅

動輪所需之轉動量。r5~r10為碰撞感測器之Marker，用來判斷與決定機器人旋轉方向。r11~r14為螺旋、循邊、牛耕與隨機路徑所要進行之時間或次數。r15為設定清掃時間，即到達設定時間機器人將停止清掃。

對於兩個驅動輪的速度變更乃利用一Fortran程式段，呼叫函式MODIFY 來執行ADAMS 模型中驅動輪參數的改變，分別設定 MOTION/1 與 MOTION/2，來達到後退、左轉、右轉等動作。

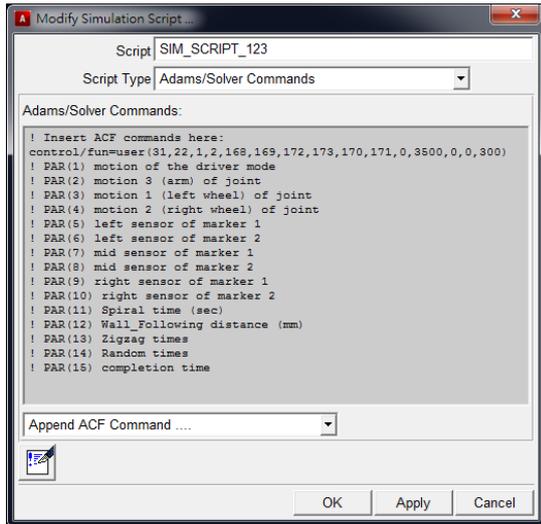


圖5. Script 設計畫面

#### 4. 清掃路徑模擬與覆蓋率分析

##### 4.1 清掃模擬

利用ADAMS使清掃機器人在設計的環境中做模擬，利用撰寫程序腳本(Script)並配合Fortran所編譯之使用者定義子程序(consub)作為機器人控制邏輯，以實現機器人的運動動作及清掃路徑模擬，因此就可以藉由模擬的方式，預先瞭解機器人在真實環境中清掃的狀況，並將不同的清掃策略在不同的環境做模擬。圖6為機器人在單一房間之清掃情形與清掃路徑，可看出此路徑是由螺旋、循邊、牛耕加上隨機路徑而成。

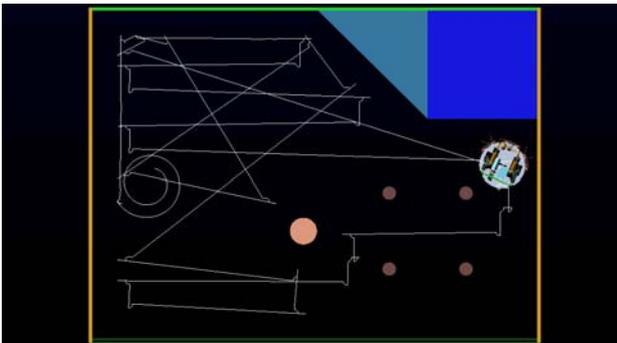


圖6. ADAMS 清掃路徑

##### 4.2 清掃效率分析

由ADAMS模擬出來的路徑，透過Visual C#所設計的程式介面，分析清掃機器人在環境中的覆蓋率與清掃效率，並把重複清掃次數以5種不同灰階顯示，觀察清掃機器人覆蓋分佈情形，以作為未來不同清掃策略的效率評估，再藉由ADAMS與Fortran匯出之資料，觀察清掃機器人在該環境之平均碰撞距離與最大碰撞距離，作為該環境大小的評估與機器人控制參數，進而增加清掃效率。

圖7為圖6之路徑分析，除了可看出覆蓋分布，更可以看出重複覆蓋次數，並利用線圖表示其覆蓋率與時間的關係(圖8)。同時記錄機器人的平均碰撞距離與最大碰撞距離，用來未來評估該環境之大小與複雜度，可作為清掃邏輯自動判斷的參考依據。圖9與圖10兩房一廳之環境分析，可看出現有清掃邏輯的缺點，可能有些區域機器人不會清掃到，造成雖清掃90分鐘，其覆蓋率不足80%。

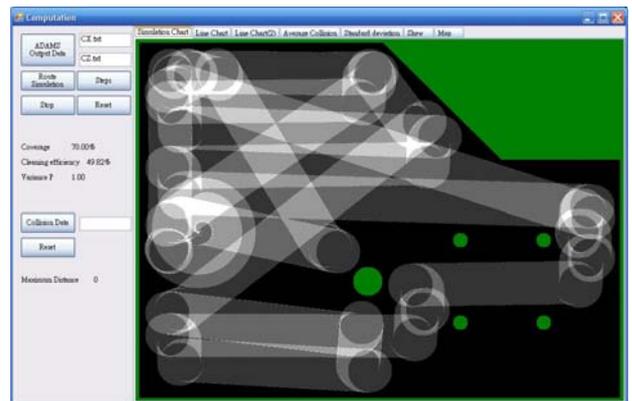


圖7. 覆蓋路徑分布

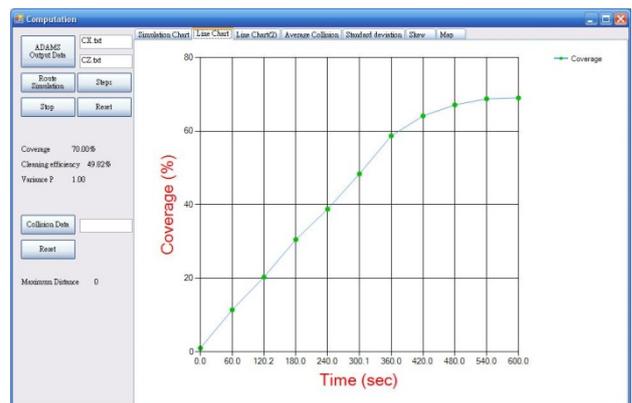


圖8.單一房間覆蓋率曲線圖

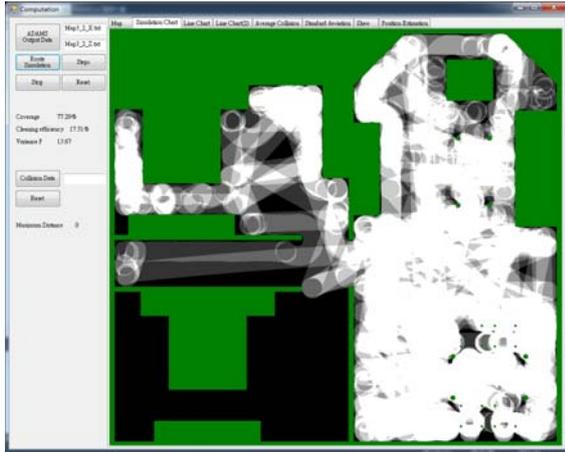


圖9.兩房一廳覆蓋路徑分析

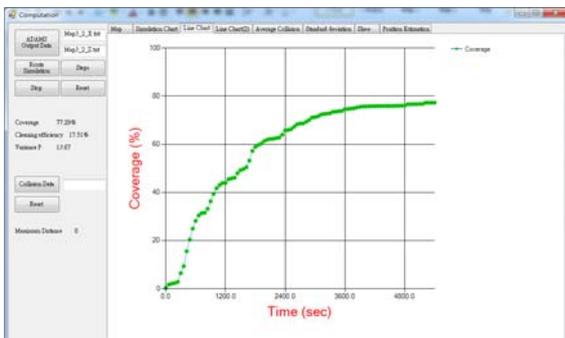


圖10.兩房一廳覆蓋率曲線圖

## 5. 系統應用範例

### 5.1 清掃效率評估

在清掃效率的評估，以兩個時間點進行觀察：(1) 清掃前期：觀察機器人在短時間內之清掃覆蓋率，覆蓋率愈高，代表機器人可以在愈短的時間內達到清掃的效果。定量評估：覆蓋率(%)。(2) 清掃後期：覆蓋率基本上以達到可接受的程度(90%以上)，之後的清掃在覆蓋率的提升有限，但觀察機器人之覆蓋率分布，希望在可清掃範圍內，其清掃次數應是越均勻越好，而非過度集中在某些區域。定量評估：覆蓋變異率(coverage variation coefficient, CVC)。

$$CVC = \frac{\text{清掃格點的清掃次數標準差}}{\text{清掃格點的平均清掃次數}} \quad (1)$$

CVC 越低代表整體清掃效率越好，因為隨著平均清掃時間的提升，區域平均清掃次數自然提升，但清掃的標準差越小，代表即清掃越均勻。

### 5.1 清掃控制邏輯改善

圖11與圖12為某一清掃機器人在單一空房間(4x5m之空間)以隨機 $t_1$ 分鐘、循邊模態 $t_2$ 分鐘、螺旋模態交替，以所建立的模擬系統在清掃15分鐘與30分鐘之覆蓋分布情形，由圖中可見其覆蓋分布大約都集中牆邊，而中間區域清掃不足，這是因為其控制邏輯之

Wall-Following執行時間過長與Random來回清掃次數過少所造成。

因此在此將其控制邏輯進行下列之修改：(1)將Random模態控制條件由時間控制改為碰撞次數控制。以碰撞次數作為模態控制條件，較不容易受環境大小影響，在狹小的空間會因次數的控制，在短時間就能切離此模態，避免重複清掃，而在空曠空間以次數控制就能增加中間區域的清掃。(2)將Wall-Following執行時間變為一個隨清掃時間改變的參數，介於短循邊( $t_s$ )與長循邊( $t_l$ )之間變動。短循邊用於清掃初期作為局部清掃的移動，長循邊用於清掃後期，將機器人帶到另一個空間進行清掃。(3)因Spiral容易與之前Random路徑重複覆蓋，但Spiral對於大空間則具有完整覆蓋的優點，因此初期則以隨機與循邊為主要清掃模態，把Spiral順序移到清掃時間的後期再加入，藉以提升整體平均覆蓋次數。

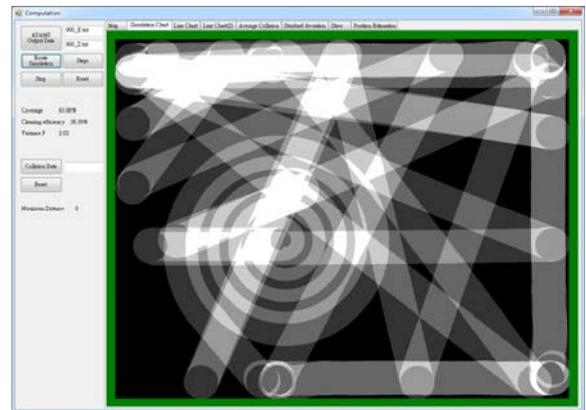


圖11.單一房間清掃 15 分鐘覆蓋分布

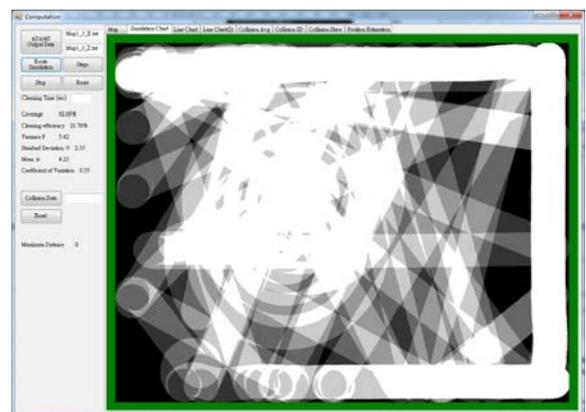


圖12.單一房間清掃 30 分鐘覆蓋分布

### 5.2 清掃改善結果

圖13與圖14為修改控制邏輯後在單一空房間清掃15分鐘與30分鐘之覆蓋分布情形，其減少了在牆邊的重複清掃並增加了中間區域清掃，分別再三不同出發點進行模擬，其結果如表1與圖15，在清掃初期15分鐘的平均覆蓋率提升了9%，而在覆蓋變異係數(CVC)即清掃的均勻性而言，則改善了18%，比較

圖12與圖14也可看出修改方式可提升清掃的均勻性，有此可看出本系統在未來分析清掃邏輯的效益上，將可提供一個客觀且簡便的評估系統，作為設計清掃邏輯的輔助用。

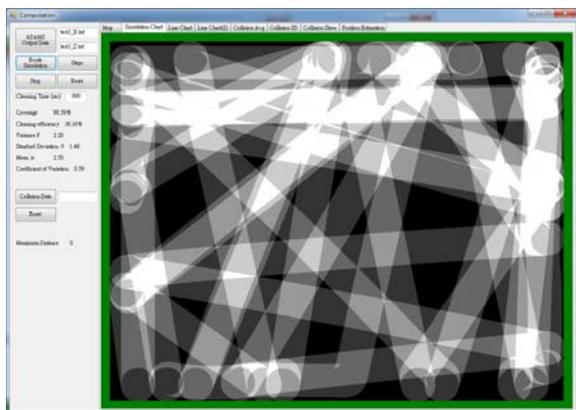


圖13.修改邏輯之清掃 15分鐘覆蓋分布

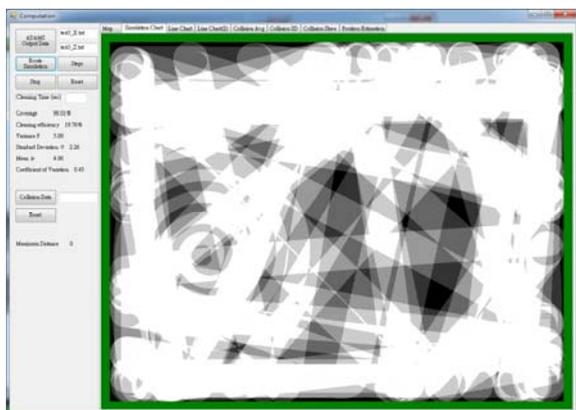


圖14.已修改邏輯之清掃 30分鐘覆蓋分布

表 1. 控制邏輯修改前後清掃效率比較

	15分鐘覆蓋率(%)	30分鐘CVC
修改前	83.08	0.55
修改後	90.59	0.45

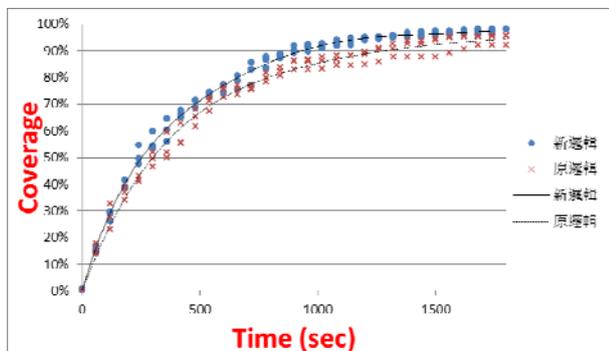


圖15.修改清掃邏輯與原有邏輯的覆蓋曲線比較

## 6. 結論

本文探討一個模擬環境的建構，作為智慧型清掃機器人控制系統的評估。利用動態模擬軟體ADAMS所提供的模組建構虛擬環境與機構，由清掃機器人的控制邏輯規劃，配合環境感測器與障礙感測器的應用，分析機器人的清掃策略對未知環境的覆蓋率與清掃效率的影響。

藉由模擬的方式，預先瞭解機器人在真實環境中清掃的狀況，並將不同的清掃策略在不同的環境做模擬。並由ADAMS 模擬出來的路徑，透過Visual C#所設計的程式介面，分析清掃機器人在環境中的覆蓋率與清掃效率，觀察清掃機器人覆蓋分佈情形，以作為未來不同清掃策略的效率評估，再藉由ADAMS與Fortran匯出之資料，觀察清掃機器人在該環境之平均碰撞距離與最大碰撞距離，作為該環境大小的評估與機器人控制參數，進而增加清掃效率。

透過模擬的方式，可運用在各種環境，且不受環境大小影響，在控制方面可快速修改控制邏輯與模擬，可大幅減少分析所需時間與成本的費用，在未來清掃機器人的控制設計評估將會有非常大的幫助。

## 7. 參考文獻

- [1] Rhim, S., Ryu, J.-C., Park, K.-H., Lee, S.-G., 2007, "Performance Evaluation Criteria for Autonomous Cleaning Robots," *IEEE International Symposium on Computational Intelligence in Robotics and Automation*, Jacksonville, FL, USA, pp.167-172, June 20-23, 2007.
- [2] Palacin, J., Palleja, T., Valganon, I., Pernia, R., Roca, J. "Measuring Coverage Performances of a Floor Cleaning Mobile Robot Using a Vision System," *IEEE International Conference on Robotics and Automation*, 2005
- [3] <http://www.cyberbotics.com/>
- [4] Srivastava, S.; Nandi, G. C., "Localization of mobile robots in a network using mobile agents," *Computer and Communication Technology (ICCCT), 2010 International Conference on*, vol., no., pp.415,420, 17-19 Sept. 2010
- [5] Ke Cheng; Yi Wang; Dasgupta, P., "Distributed area coverage using robot flocks," *Nature & Biologically Inspired Computing, 2009. NaBIC 2009. World Congress on*, pp.678,683, 9-11 Dec. 2009.

## **Reality Simulation System for the Analysis of Cleaning Efficiency of Intelligent Cleaning Robots**

Yu-Chen Hsu and Jyh-Cheng Yu\*

Department of Mechanical and Automation Engineering,  
National Kaohsiung First University of Science and  
Technology

\*Corresponding: jcyu@nkfust.edu.tw

### **Abstract**

This paper discusses the construction of a reality simulation environment to evaluate the cleaning efficiency of an intelligent cleaning robot. The cleaning path is automatically generated from the cleaning strategy that controls the interacting response of the robot with a simulated domestic environment. Collision and wall-following sensors are established in the robot to detect obstacles. Typical cleaning modes include random bouncing, spiral, wall following, and zigzag movements. This research applies Automatic Dynamic Analysis of Mechanical System, ADAMS, to construct a virtual prototyping environment that combines sensors, simulation scripts and macros to determine the response for different obstacles. The coverage and cleaning efficiency for the cleaning strategy can be analyzed in the simulation system. The cleaning path for a specified cleaning strategy can be derived and transformed to the coverage rate and the coverage uniformity through a analysis program written in Visual C#. The proposed methodology adopts a soft prototyping to evaluate the cleaning strategies of a cleaning robot that can greatly reduce experimental costs and increase development efficiency. An application example is presented to illustrate the effectiveness of the proposed system.

*Keywords:* Cleaning Robot, Cleaning Simulation,  
ADAMS, Coverage, Cleaning Efficiency