

自動清掃機器人控制與清掃效率分析的實境模擬系統

吳豐成、余志成*

國立高雄第一科技大學系統資訊與控制研究所

國立高雄第一科技大學機械與自動化工程系*

摘要

本文探討一個模擬環境的建構，作為清掃機器人控制系統的評估。由清掃路徑與動作的規劃，配合環境偵測感測器的應用，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。典型清掃機器人的基本清掃模態包括隨機、螺旋、循邊與之字型路徑，配合機器人上的感測裝置，本研究歸納機器人運行於一般室內環境所會遇到的障礙物類型以及反應模式，進行清掃軌跡的模擬。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，以作為未來不同清掃策略的效率評估。

關鍵字：清掃機器人、清掃模擬、覆蓋率、清掃效率、ADAMS

1. 前言

隨著資訊科技以及機電整合的快速發展，機器人不再侷限於生產輔具，而逐漸融入人們日常生活中，包括服務及娛樂型機器人，如機器寵物、接待機器人、吸塵打掃機器人與保全機器人等，這類的機器人不但使人們的生活更加便利，同時也提高了生活品質。而其中尤以智慧型自動吸塵器機器人(vacuuming cleaners)，在消費市場中呈現高度的成長，成為最成功的消費型機器人產品。

市面上已經有許多廠商開發出各式各樣的清掃機器人，其功能大多數都相差不大。典型的打掃機器人基本結構包括在基座中設置兩組動力輪，以及一吸入並容置灰塵的集塵裝置，或在基座底端增設滾筒型的捲刷或旋轉型的側刷以提升清潔效果，同時配合控制感測器與路徑設計，使其可自行在室內空間中依行走路徑的設置，隨性自走、螺旋行走、靠牆行走或乙字型的移動，而達成在室內自動位移與清掃，當電力不足時，機器人還會自動搜尋充電站自行充電。

機器人如何在複雜的居家環境中作流暢的清掃動作與閃避各類型的障礙，其感測器的應用成為一個重要的因素，主要可分為接觸式的如碰撞保桿配合感測裝置，或非接觸式的如紅外線[1][2]與超音波感測[3][4]，用來感測障礙物與環境辨識。這些感測方式

各有其優缺點，接觸式的有噪音大與家具碰損的問題，而超音波易有反射干擾與價格高的缺點；而紅外線會有顏色辨識及受環境中燈光的影响問題。但比較起來紅外線在複雜性、角度量測的解析度及價格上，會較適用於居家清掃機器人的清潔路徑使用。

機器人的清掃性能可由兩個指標來評估：覆蓋率(coverage)與清掃率(cleaning rate)，覆蓋率指的是在一段時間內，機器人所清掃過的面積除以所應清掃的面積，而清掃率則是指以清掃面積的增加速率。文獻上有許多關於全域覆蓋路徑演算法的研究[5][6][7]，但多建立在能完整感測環境與掌握行徑路徑的條件下。在產品實現中，一般低階的清掃機器人所使用的標準清掃策略是以隨機搭配循邊的清掃模態，先以直線方式前進，在碰到障礙物時，以隨機的幾個預設角度轉向後再以直線前進。在幾次的隨機碰撞清掃模態之後，轉以循邊式的方式，一方面可清掃牆角較較僻的區域，一方面可使機器人沿牆邊進入其他的空間進行清掃。二者交替直到至預設時間到了或是電源不足時再回到充電站。理論上，除了機器人幾何限制所造成牆角無法清掃的區域之外，若沒有時間限制，這種清掃模態尚可達到接近百分之百的覆蓋率，但其清掃率會隨著時間下降，有些區域可能會重複清掃，但受到電池容量的限制，有些區域可能不會被清掃到。

另一種清掃策略則是模仿人的清掃方式，以固定的型式循序清掃，例如以牛耕的之字型方式，反覆交錯清掃。之字型清掃則因具規則性，理論上在無障礙空間清掃具有較高的清掃效率，其清掃率隨時間的變化較小，但當遇到障礙時也容易存在死角的問題，而且在實際控制上所面臨的挑戰較高，如何能控制機器人準確的轉向，並能以平行的直線牛耕清掃是一項最大的挑戰，一般需搭配環境辨識的模組。

現有清掃效率的評估多需以實機實地測試的方式，搭配影像分析評估清掃效率[8]，不僅成本昂貴，且不易測試不同環境與變更清掃控制程式。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策略邏輯的建置，自動模擬清掃機器人於所建置虛擬環境下的路徑軌跡，以探討清掃策略規劃對於清掃效率的影響。

2. 清掃機器人的系統架構

本文採用松騰的RV10 趴趴走清掃機器人做為分析載具，並加入平行循邊的控制模組，以改善行進控

*聯絡作者：jcyu@nkfust.edu.tw

制，主要的功能有感測單元、運動單元與清潔單元，機器人系統架構如圖 1 所示。其中機器人現有的感測單元包含了防掉落感測器 3 個、碰撞感測器 2 個，另外在機器人左側額外增加了 2 個循邊感測器；運動單元包含了兩顆直流馬達，控制機器人的運動動作；清潔單元包含兩個毛刷模組及吸塵器模組，硬體架構如圖 1 所示。

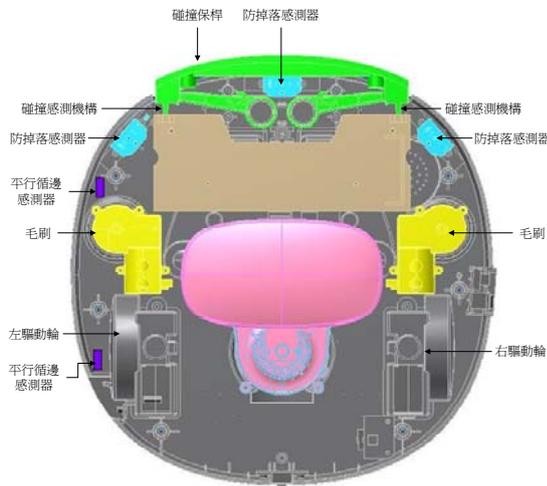


圖 1 清掃機器人構造圖

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：隨機路徑 (Bouncing)、之字形 (Zig-zag)、循邊 (Wall_follow)、螺旋 (Spiral) 等。(1) 機器人隨機路徑模態的進行方式於開始啟動後，直行行走，當碰到任一障礙物並觸發碰撞感測器，若機器人兩驅動輪的中心即為其形體中心，則可原地轉向，如 Roomba。但若轉向中心不在機器人圓形體的中心，為避免轉向時碰撞到其他的障礙，則機器人需先後退一段距離，之後旋轉任一角度如趴趴走機器人，而後反覆障礙偵測、轉向、直行的動作。(2) 之字形的清掃模態則是直行碰觸到障礙後，後退一段距離，再以一側輪為中中心，將機器人轉向 180 度後再直行反覆動作。(3) 循邊的方式又隨著循邊機構設計的不同，有的以鋸齒狀反覆碰撞的方式循邊如松騰的趴趴走，在碰觸到障礙後，後退一小距離，轉向、前進一小距離，再向左側轉後碰撞障礙，反覆沿牆前進。有的以紅外線控制，以固定距離平行牆面移動，如 iRobot 的 Roomba。(4) 螺旋則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。

3. 實境模擬系統的建置

3.1. 動態運動模擬

本研究使用的動態模擬分析軟體 MSC 所發行的 ADAMS 高階的動態分析軟體，在其開放性架構下，建構清掃機器人模型、運動模態及虛擬環境來做模擬與分析，不僅可以檢測機構的正確性，更能確定動作的流程順序 [9]。ADAMS 的模擬執行步驟，首先為建

立模擬環境的參數設定，接著為模擬物件 3D 實體的建立，結構簡單的模擬物件可由 ADAMS 內建的圖形工具建置；構件複雜的物體，則可由 3D 繪圖軟體建構出外型後，再轉入 ADAMS；ADAMS 為模型建立流程的第一階段為模型 3D 實體 (parts) 的建立，可由 ADAMS 內建或由繪圖軟體建好後載入，並設定零件材質與密度。第二階段設定模型的機構限制接點 (joints)，限定其物件的運動方式及自由度，還有建立物件與物件之間的相互運動關係，第三階段定義感測器觸發的狀態條件，第四階段給予模型運動的動力 (motion)，使模型可依照實體的動力來進行運動，隨後即可進行動態模擬。

3.2. 環境建構

以現有產品 RV-10 作為模擬的載具，配合其感測裝置及碰撞機構做 ADAMS 的模擬環境設定，針對機器人在行進過程中若遇到障礙物時，利用感測器判斷做適當的閃避動作，因此必須先設定感測器參數與建構虛擬的環境，以進行後續的模擬工作。本文模擬的清掃機器人，係為輪型走行的機器人，而非固定機構的運動模擬，因此需根據與環境的互動結果即時反應，因而模擬行走時會使用到 ADAMS 的「輪胎」與「路面」模組與碰撞感測功能，藉以建構出所需要的虛擬環境。

障礙物的建置由 ADAMS 內建的模組做設定，如牆壁、障礙物等，將建構的障礙物利用 Fixed_Joint 固定在路面上，就完成如圖 2 的障礙物設定；為了使機器人能夠達到避障的功能，將設置碰撞感測器，使機器人在行走時若遇到障礙物，可以反應以繞行或避障障礙物，可運用 ADAMS 提供了碰撞模組的功能來設定機器人與障礙物之間的碰撞關係。

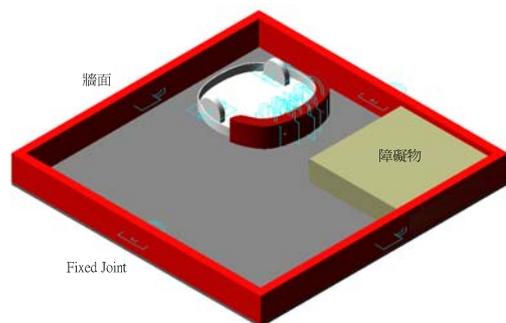


圖 2 障礙物設定

3.3. 行進方向設定

機器人的運動方式是在 JOINT 設定 MOTION 來驅動，MOTION 可設定驅動速度、方向、與開關的時間，須配合機器人的行進動作做規劃，機器人是藉由左、右輪兩顆馬達來控制行進動作如前進、後退及轉彎等動作；圖 3 為 MOTION 的配置圖，MOTION/1 與 MOTION/2 分別表示機器人的右驅動輪與左驅動輪，可由 Joint Motion 視窗定義 MOTION 的相關參

數，語法格式為每秒的旋轉角度，以右手定則定義方向。以圖 3 的設定為例，MOTION/1 與 MOTION/2 皆為正 Z 軸的方向，代表驅動輪馬達皆正轉，因此機器人將朝向負 X 軸方向前進。

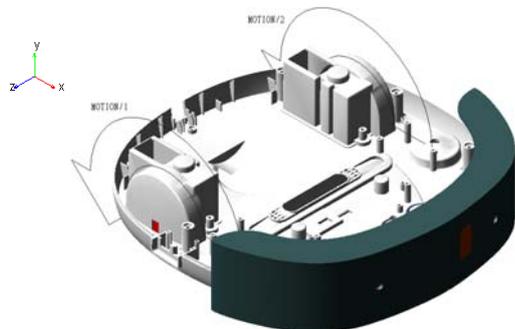


圖3 機器人 MOTION 設定位置示意圖

3.4. 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 中感測器的功能，可建立系統模擬行為的控制機制，使 ADAMS 在預先設定運動控制邏輯條件，自動依據遇到的障礙物做特定的運動。

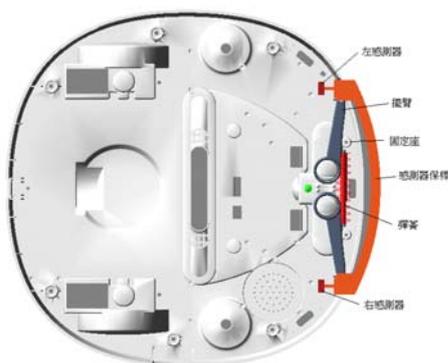


圖4 碰撞感測器機構

3.4.1. 碰撞感測設定

RV-10 的感測機構如圖 4 所示，在初始位置時擺臂與感測器保桿固定在一起，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，因此當感測器保桿受到障礙物的擠壓時，擺臂會做圓週運動，並帶動感測器保桿觸發感測器。左、右感測器做動的機制如同光遮斷器一樣，所以將其概念運用在 ADAMS 設定中。作法是利用圖 5 的兩個 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，即可地定義此距離感測器為觸發。

3.4.2. 防掉落感測設定

機器人在行走時可能會遇到高低落差的地面如樓梯，為防止機器人掉落而摔壞，RV10 在基座的左前方、正前方、右前方安裝了三個紅外線作為防掉落感測器。本研究以 ADAMS 提供的 MARKER 作為感

測器，如圖 6 所示。以 MARKER 與斷差邊界的距離定義感測器觸發的條件，在 ADAMS/View 工具列上的 Simulate-Sensor-New 開啟設定感測器的對話框設定參數；觸發的條件是由 MARKER_A 的位置座標與地面高低落差邊緣上的 MARKER_B 與 MARKER_C 所構成的直線方程式作為感測器觸發的依據，如圖 7，當 MARKER_A 與 MARKER_B 與 MARKER_C 所構成的邊界方程式距離小於一門檻值時即觸發。

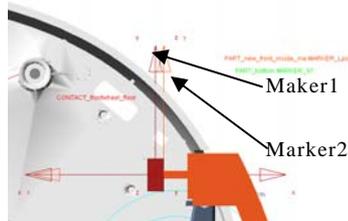


圖5 以 Marker 設定模擬紅外線遮斷器

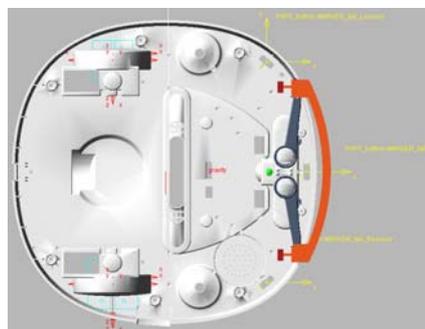


圖6 防掉落感測器示意圖

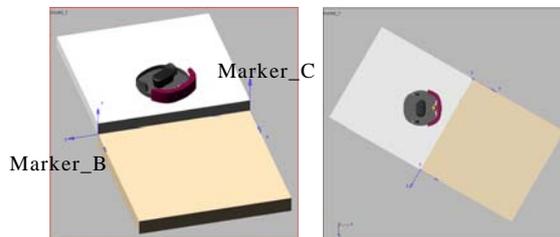


圖7 防掉落感測器

3.4.3. 平行牆面感測設定

機器人在運動時亦會沿著牆壁行走，可用於尋找充電站、清掃牆角的灰塵與尋找房間入口等，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 8）；利用 IRW0 與 IRW1 到牆壁的距離差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，如圖 8 的 MARKER_Reference1 與 MARKER_Reference2，並由這兩個 Marker 算出牆面的邊界方程式，分別將 IRW0 與 IRW1 的位置資訊代入，藉由 Function Builder 中的建立距離量測指令： $(.model_1.FUNCTION_MEA_IRW0)$ 與 $(.model_1.FUNCTION_MEA_IRW1)$ ；算出 IRW0 與 IRW1 至邊界方程式的距離，當 IRW0 與 IRW1 到邊

界方程式的距離小於一門檻值時，即代表機器人已經與牆壁平行。

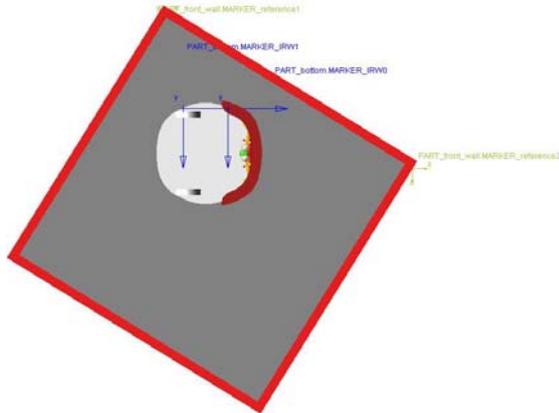


圖8 平行牆面感測器設定

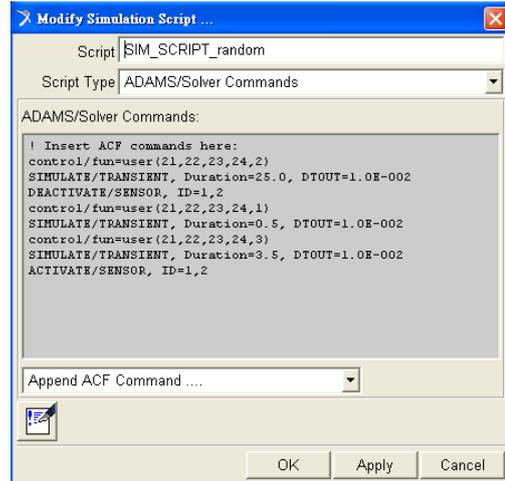


圖9 隨機路徑

3.5. 模組化運動模擬指令

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，模組化是由 ADAMS 裡 Simulate-simulation script 的功能來進行，Script 設定可在模擬過程中改變求解方式、時間間距或模型架構等，同時配合 MACRO 的功能來做出參數化，讓使用者可自行定義 MOTION 速度與方向。

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 的參數值與初始值，再由使用者先規劃好所要做的 ADAMS 模擬去執行。如圖 9 所示為隨機路徑的 Script；在程式編輯區第一行的語法 control/fun=user(r1,r2,r3,r4,r5) 是透過 FORTRAN 的語法紀錄及改變模型的相關參數；r1 與 r2 表示左驅動輪上旋轉接頭的 Marker，其中 r1 代表定義在驅動輪上的 Marker，r2 代表訂在機身上的 Marker，因此可藉由兩個 MARKER 的相對轉動量得到驅動輪的轉動量。同理，r3 與 r4 表示右驅動輪上旋轉接頭的 Marker。而 r5 是用來指定基本運動方式，若 r5=1 表示機器人後退；r5=2 表示機器人前進；r5=3 表示機器人原地左轉；r5=4 表示機器人原地右轉。

圖 10 中為 FORTRAN 程式段中的參數 PAR(1)~PAR(5) 分別對應到 Script 的輸入參數 r1~r5，其中函式 SYSFNC 表示用來記錄 r1 相對於 r2 的轉動量，這是因為驅動輪在轉動時隨著時間的變化 r1 會相對於 r2 做旋轉，藉由 SYSFNC 可傳回相對位置記錄於第 4 個引數（如 DIS1），作為下一個模擬時 MOTION 初始位置的設定。函式 WRITE(COMMND, 100) 即輸出改變 ADAMS 的 Motion 的程式設定。此程式例 r5=1 表示機器人後退的語法，圖 10 中可看出 Motion/1 與 Motion/2 的設定以 FUN=+90D 指定，即每秒 90 度的轉動速度，根據圖 3 的座標方向設定機器人將後退，同理可改變個別方向達到前進、左轉、右轉的動作。而後呼叫函式 MODIFY 用來執行 ADAMS 模型中驅動輪上的改變。

```
C
C==Excutable code=====
C == Forward motion =====
Motor=PAR(5)
IF (Motor .EQ. 1) THEN
C --- Initial static analysis of motion 1-----
IM=PAR(1) r1
JM=PAR(2) r2
IPAR(1)=IM
IPAR(2)=JM
CALL SYSFNC ('A2', IPAR, 2, DIS1, ERRFLG)
WEI1=DIS1
C --- Change the IC DISPLACEMENT OF MOTION/1 -----
WRITE (COMMND, 100) 'MOTION/1,VELOCITY,ICDISP=',WEI1,
& ', FUN =+90D'
CALL MODIFY (COMMND, STATUS)
C --- Initial static analysis of motion 2-----
KM=PAR(3) r3
LM=PAR(4) r4
JPAR(1)=KM
JPAR(2)=LM
CALL SYSFNC ('A2', JPAR, 2, DIS2, ERRFLG)
WEI2=DIS2
C --- Change the IC DISPLACEMENT OF MOTION/2 -----
WRITE (COMMND, 100) 'MOTION/2,VELOCITY,ICDISP=',WEI2,
& ', FUN =+90D'
CALL MODIFY (COMMND, STATUS)
```

圖 10 對應 MOTION 變更 Script 的 FORTRAN 程式

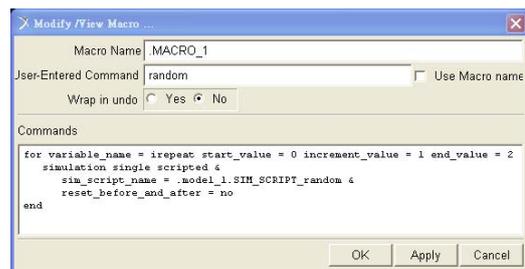


圖 11 Random 清掃模態的巨集指令

而機器人行進策略的程式則是以巨集(MACRO)來撰寫，MACRO 是一組命令集，在 ADAMS 裡可以自行設計命令，或是嵌入 ADAMS 內建的巨集功能，即可以完成 ADAMS/View 一系列的命令。設計者可在 ADAMS command language 介面裡寫出自己想要執行的項目，而在 MACRO 中可使用參數，參數化清掃模態便是利用它的語法將變數寫入指令內。所以在進行指令時，便會要求使用者輸入所需之參數，並依照輸入參數進行其模擬。參數化的動作指令也有助於之後所做模擬。所以使用巨集可完成以下的功能：(1)自動完成重複性的操作；(2)延伸 ADAMS 的

基本功能；(3)自動產生模型；(4)建立模型的多個變量。圖 11 是 Random 清掃模態對應的巨集範例，在 Command Window 裡可導入先前撰寫的 Script，進行迴圈與終止條件控制。

4. 清掃路徑模擬與覆蓋率分析

4.1. 運動動作模擬

4.1.1. 凹角牆面循邊模擬

圖 12 說明平行循邊路徑的模擬，當機器人由隨機路徑模態轉換成平行循邊時，機器人在環境中任一位置，當行走時碰到任何一面牆壁，會先觸發碰撞感測器，此時機器人一律向右旋轉，這是因為平行循邊感測器安置在機器人的左側邊，在旋轉同時，感測器 (IRW0、IRW1) 會執行上述小節設定的感測器，當旋轉到滿足感測器的設定條件時，就會觸發平行感測器，代表機器人已經平行牆壁，開始直走，直到碰到下一障礙，再重複上述的動作，就可完成凹角牆面的循邊的動作模擬。

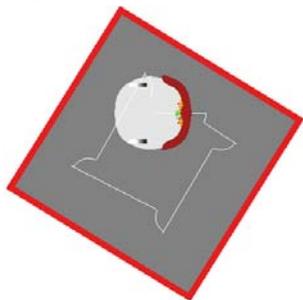


圖 12 平行循邊路徑

4.1.2. 凸角牆面循邊模擬

機器人在遇到凸角狀態時，則必須另外設計感測狀態，以讓機器人可判斷出遇到的障礙物為凸角，本研究作法在凸角的頂點設計了一個 MARKER_convex，利用機器人側面平行循邊所用的 MARKER_IRW0 與 MARKER_convex 的距離做為觸發感測器的條件，在平行循邊時，若兩個 MARKER 的距離小於一個門檻值時即觸發感測器，因此就可判斷出凸角，再藉由編碼器的計算使機器人直走到機器人的機身脫離障礙物，接著在左轉 90 度繼續循邊的清掃，模擬過程如圖 13 所示。

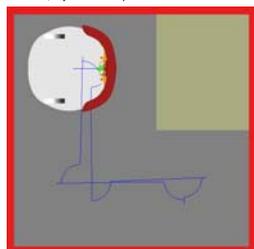


圖 13 凸角循邊模擬路徑

4.1.3. 之字型行進模擬

在之字型的模擬，機器人直走碰到障礙物時，碰撞感測器觸發，此時機器人後退一小段距離，接著原地右轉 90 度，在開始直走一小段距離，此距離依照吸塵器的吸口大小決定清掃路徑的間距，再右轉 90 度，接著直走直到碰到下一個障礙物，觸發碰撞感測器後返。同理，後退一小段距離，接著開始左轉 90 度，直走一段距離，再左轉 90 度，即完成一次之字型的模擬。

4.2. 清掃效率分析

本系統可運用於清潔機器人的覆蓋率，本研究透過 ADAMS 模擬後將資料輸出做處理，並藉由 Visual C# 的程式設計，做覆蓋率與清掃效率的計算。本節將以隨機、循邊與組合模式分析這幾種路徑的覆蓋率。

圖 14 為隨機路徑的模擬，模擬時間為 117 秒，共執行了四次 Random 的動作，其覆蓋率為 12.8 %。圖 15 為平行循邊的清掃模擬，在模擬時間為 205 秒內的模擬結果，其覆蓋率為 22.0 %。圖 16 為隨機與平行循邊路徑的模擬結果，模擬時間為 150 秒，其覆蓋率為 15.4 %。圖 17 為隨機與趴趴走機器人循邊路徑的模擬結果，因其採用反覆碰撞、後退、轉向、行進、轉向的方式進行循邊，其行進路徑呈現鋸齒狀，模擬時間為 226 秒，其覆蓋率為 22.7 %。

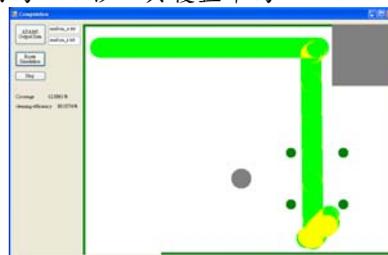


圖 14 隨機清掃路徑

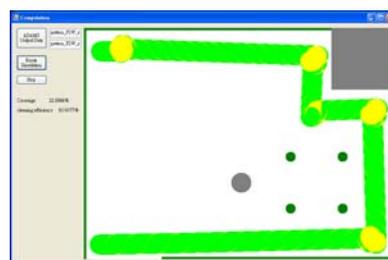


圖 15 循邊的清掃路徑

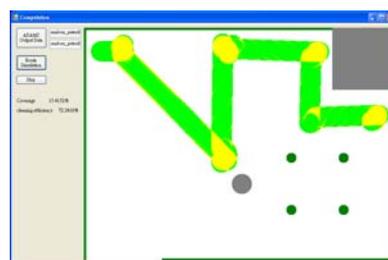


圖 16 隨機與平行循邊組合的清掃路徑

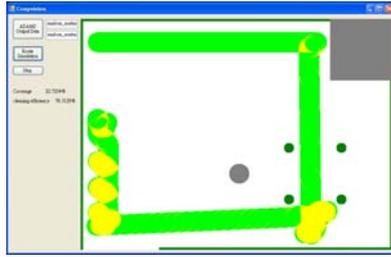


圖17隨機與趴趴走機器人循邊路徑模擬圖

5. 結論

本論文的研究目的在於設計一個實境模擬系統，並可以根據環境的需求，建構出環境地圖，使清掃機器人在設計的環境中做模擬，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做模擬，比較其好壞，因此利用此方式可分析出清掃效率，而目前一般的作法只能透過建構實體，經由實際測試得到覆蓋率，如此就會造成成本的提高，但若透過模擬的方式，就可減少成本的費用，且在未來的設計應用上就更具彈性。本文以 RV_10 趴趴走機器人的例子做比較，原有的設計在清掃過程中，其循邊的方法為利用碰撞的方式，沿著牆壁行走，此方法的缺點為花費時間較多，且在清掃時，是採用碰撞的方式沿著牆面行走，所以在牆面附近的區域，會造成特定形狀面積的區域清掃不到，所以在研究過程中將機器人機身上加了平行循邊的紅外線感測裝置與自製的編碼裝置，並透過提出的平行循邊方法，改善原有設計的清掃問題，並由提出的模擬方式驗證此方法的可行性。

6. 參考文獻

- [1] Chen, C.H. and Song, K.T., 2005, "Complete Coverage Motion Control of a Cleaning Robot Using Infrared Sensors," Proceedings of the IEEE International Conference on Mechatronics, Taipei, Taiwan, pp. 543-548.
- [2] Choi, J.S. and Park, S.K., 2002, "Sensor-Based Motion Planning for Mobile Robots," FIRA Robot World Congress.
- [3] Kim, S. and Kim, Y., 2003, "A mobile robot localization using ultrasonic sensors in indoor environment," Robot and Human Communication, Proceedings. IEEE/RSJ international Conference on intelligent Robots and Systems, pp. 52-57.
- [4] Kim, S., 2004, "Autonomous Cleaning Robot: Roboking System Integration and Overview," Proceedings of the 2004 IEEE International Conference on Robotics & Automation New Orleans, pp. 4437-4441.
- [5] Oh, J. S., Choi, Y. H., Park, J. B., and Zheng, Y. F., 2004, "Complete coverage navigation of cleaning robots using triangular-cell-based map," IEEE Transactions on Industrial Electronics, Vol.51, No.3.
- [6] Garciar, E. and Gonzalez de Santos, P., 2004, "Mobile-robot navigation with complete coverage of

unstructured environments," CSIC Robotics and Autonomous Systems, pp.195-204.

- [7] 林宗德，2005，居家清潔機器人之全域覆蓋路徑規劃與實現，國立成功大學，碩士論文。
- [8] 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, June 20-23, 2007, pp.167-172.
- [9] 李敏暘，2003，適形越障探測載具之全域行動分析與模擬，國立台灣科技大學，碩士論文。

Analysis of the Control and Cleaning Efficiency for Autonomous Cleaning Robots Using ADAMS

Feng-Cheng Wu¹ and Jyh-Cheng Yu²

¹Institute of System Information and Control

²Department of Mechanical and Automation Engineering

National Kaohsiung First University of Science and Technology

Abstract

This paper discusses the cleaning simulation of a domestic cleaning robot in a virtual environment. The cleaning path is automatically generated from the cleaning strategy that controls the interacting response of the robot with a simulated domestic environment. Collision, drop and wall-following sensors are established in the robot to detect obstacles. The coverage and cleaning efficiency for the cleaning strategy can be analyzed in the simulation system. 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 cleaning path for a specified cleaning strategy can be derived and transformed to the coverage rate and the cleaning efficiency through an 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.

Keyword: Cleaning Robot, Cleaning Simulation, Coverage, Cleaning Efficiency, ADAMS