

虛擬蜘蛛建網及步行的互動式模擬系統

Interactive Simulation of Web Construction and Locomotion for Virtual Spider

林翰農
國立政治大學資訊科學系
台北市指南路二段 64 號
s8908@cs.nccu.edu.tw

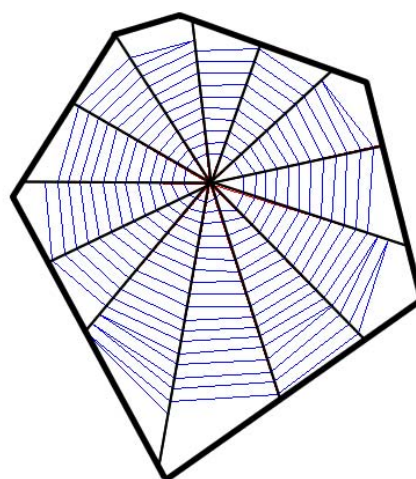
李蔡彥
國立政治大學資訊科學系
台北市指南路二段 64 號
li@nccu.edu.tw

摘要

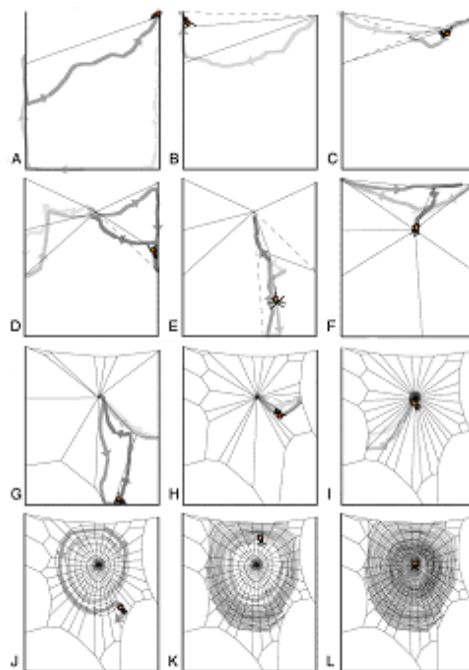
虛擬生物的模擬技術，有助於降低電腦動畫應用的製作成本，並可創造出教育價值高的娛樂應用。文獻中有關虛擬生物的模擬，對蜘蛛之類的節肢動物的運動規劃，卻並不多見。本論文嘗試以程序性動畫設計法則，研究模擬蜘蛛網的建構及虛擬蜘蛛在蜘蛛網上自動規劃行走運動。根據生物學文獻及實際觀察，我們將蜘蛛網的結網規則建立模擬程序，並透過幾何及偏好參數的設定建構蜘蛛網。而蜘蛛在網上的步行模擬，則是根據蜘蛛網的結構及移動的目標，透過路徑計畫演算法搜尋合法的路徑。本系統建造蜘蛛網的部分已經可以成功的模擬出蜘蛛網的型態，也能夠自動規劃蜘蛛在蜘蛛網上的運動路徑。透過互動式的人機界面，使用者可以動態的破壞蜘蛛網結構，並設定獵物位置，然後交由系統產生可行的蜘蛛運動模擬，並以 3D 即時動畫的方式呈現。

1. 簡介

製作電腦動畫的方式主要有動作擷取(Motion Capture)和取動作關鍵格(key-framing)以及程式自動產生動畫這三種方法。一般製作動畫主要是利用動作擷取和取動作關鍵格這兩個方式產生動畫。動作擷取的特點是利用真人演員的表演動作，讓電腦讀取後產生虛擬演員的動作。其優點是可以得到看起來很自然的動作模組，但是要把動作套到虛擬人物的時候，仍然需要做關節間距的調整，而且得到的動作不能適應其他環境或是地型的變化。取動作關鍵格的方法常見於坊間的 3D 動畫製作軟體，使用者可以將做好的模型依照時間先後，每間隔一段時間調整模型在該時間點要擺放的位置，此動作就是在設定模型的動作關鍵格。之後，讓軟體利用內插法(interpolation)產生關鍵格與關鍵格之間連接的動作。而程式自動產生的動畫可以做到動作擷取美中不足的地方。此方式又稱「程序性動畫」，係根據運動原理及規則，以程序性演算法模擬動作，優點是重複性及適用性高。此方法的缺點是不同動作所需的演算程序多不相同，因此要設計通用的程序，或要收集到豐富的模擬程序並不容易。



圖一：最粗的線條是外框，輻射狀的普通粗線是輻射絲，螺旋狀的細線是螺旋絲。擁有黏性的絲線只有螺旋絲。



圖二：蜘蛛網的建造順序是外框-> 輻射絲 -> 螺旋絲。圖片來源出自[5]。

我們的目標在於以程序式動畫的方式，自動模擬蜘蛛網與蜘蛛在蜘蛛網上的行走模式。而此研究的應用目標包含兩點，第一是在教育與娛樂的應用。藉由程式模擬蜘蛛的行為，讓一般人可以藉由虛擬的方式，觀察蜘蛛的行為與蜘蛛網的構成，不需到野外捕抓蜘蛛也不會破壞生態。能藉由模擬系統來讓一般人了解蜘蛛結網的行為以及結網的規則和特性；另外使用者可以和虛擬蜘蛛互動，決定蜘蛛在蜘蛛網上行走的終點，藉以觀察蜘蛛在蜘蛛網上的步行達到育教娛樂的效果。第二點是提供電腦動畫或電影特效的製作工具。本研究所設計出的系統提供出一個基本的架構給想要做出蜘蛛網或是蜘蛛爬行運動的動畫應用。設計者利用此程式即可產生出動畫雛型，加以修飾後即可呈現出客製化的蜘蛛網效果及蜘蛛爬行動畫。

因此，本論文所描述之系統包含兩個目標：第一點是藉由使用者指定蜘蛛網的外框讓程式自動產生虛擬蜘蛛網的架構。圖一即是擷取自我們的程式自動產生的結果，其中最粗的線段代表的就是外框，輻射絲和螺旋絲的部分就由程式依據給定的規則自動產生。第二點是模擬蜘蛛在蜘蛛網上的行走運動規劃，其行走的環境也就是之前第一點產生的虛擬蜘蛛網。

在第二單元會簡單介紹過去相關研究，其中包括多腳機器人和模擬蜘蛛網的部分；第三單元會提到關於蜘蛛網的建構規則以及虛擬蜘蛛機器人的身體構造定義；第四、五單元會探討蜘蛛網建構以及蜘蛛步行的演算法，第六單元討論實做方法及實驗結果，第七單元是未來發展與結論。

2. 相關研究

關於蜘蛛網的模擬已經有人[1]作出蜘蛛結網歸納出一般性的規則，但是它所用的方法是先給予一些建造網子的規則以及一些變數，再利用基因演算法(Genetic Algorithm)和評估蜘蛛網的真似程度的評估程式來調整衍生出的蜘蛛網。而我們的目標也以他們歸納出來的蜘蛛網架構為主要參考原則來建立蜘蛛網。但是我們不同於[1]的地方，在於我們的蜘蛛網產生出來之後必須成為蜘蛛機器人行走的環境。他們的結果是最終產生的蜘蛛網要用來和真實世界的蜘蛛網做比較，其目標在於蜘蛛網的擬真程度。至於蜘蛛或多腳機器人步行的模擬目前已經有相當多的研究成果[2,3,4,8]，但絕大多數的研究目標都是把焦點放在行走於平面或崎嶇的環境上，目前尚未有人研究蜘蛛在蜘蛛網上行走的模擬；並且，不論在蜘蛛腳關節的自由度上或是腳的數目上以及行走的步伐模式，都不是在模擬真正蜘蛛的八隻腳構造，最常見到的是四隻腳以及六隻腳的蜘蛛機器人(spider robot)。

在互動的層面上，除了虛擬蜘蛛網可以由使用者給定外框後自動產生蜘蛛網，也可以藉由參數控制板上的滑桿調整蜘蛛網的屬性。在與蜘蛛機器人的互動上是在於使用者可以利用放置獵物促使蜘蛛機器人從現在位置走向獵物所在地。這一點與[10]提過的三層自主式機器人溝通的中階溝通也就是指定終點令蜘蛛自動感測週遭環境再自動找出路徑與步行的規劃，都是通知機器人要移動到哪個地方的工作目標導向的命令。

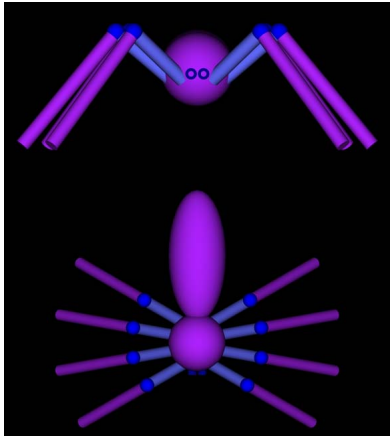
在虛擬生物的方面來說，分成模擬現實世界存在的生物與完全虛構的生物這兩種。在模擬現實世界的方面來說有[11]模擬真實魚的運動。其他常見的模擬生物也有像是[10]模擬狗與人類的互動。而我們論文所提出的模擬蜘蛛也是屬於模擬真實生物的範疇之中。在虛構生物的方面，有[12]所提出的利用基因演算法(genetic algorithm)從簡單的肢節構造繁衍出複雜型體的生物與該生物的運動行為，像是跑、跳和游泳。以及[13]給定虛擬生物一些常識讓牠們在環境中做出合於常識判斷的行為。但過去這些生物模擬系統都未曾以蜘蛛做為模擬的主題。

3. 問題定義

3.1. 環境定義

蜘蛛網分成立體網和平面網，立體網像是比較著名的黑寡婦，就是利用類似垂簾的網子捕捉獵物，甚至也有地洞型的陷阱網。這些構造複雜的網可以相當有效率的捕捉到獵物，不過構造比較簡單的平面網除了有花費的絲線量比較少的優點之外他的捕捉獵物的功能也不遜於立體網的效果。平面網又有分成圓形網及不規則型的網子，不規則的網子結網規則不容易歸納出來，不同的品種也有其不同的規則，就算是同種的蜘蛛也可能會因為壓力、天氣、溼度等變因影響網子的結構。我們著重的目標是針對平面圓網型的蜘蛛網進行模擬。

首先我們先介紹蜘蛛網的特性。就外觀而言，它主要分成三個部分，外框、輻射絲、螺旋絲。外框幾乎都是由凸多邊形組成的，內部則有輻射絲與螺旋絲如圖一。在模擬蜘蛛網之前，我們先說明真正的蜘蛛是如何建造網子。不論在國外或國內學界都有一種通用的程序來描述一隻織平面圓網的蜘蛛，如圖二。建造網子的順序是先建構出外框(圖二步驟 A-C)再以此為支架架設輻射線(radii line) (圖二步驟 D-I)，這時就可以開始製造輻射線了。值得觀察的一點是，有些品種的蜘蛛在建完輻射線之後會先架起鷹架再建螺旋線。所謂鷹架指的是蜘蛛會由內向外大略織出間距相當寬且不愈黏性的輻射絲，建置完成之後再由外而內織間距較小且具黏性的輻射絲。依我們的推測有可能是因為蜘蛛有偏好走線與線之間的



圖三：蜘蛛的正面圖與俯視圖。

交叉點的結果，因為這樣的走法比較穩固，舖出來的絲線也可以比較精準整齊。有關蜘蛛的行走偏好，在下一個小節會有更詳盡的解釋。在我們這個研究中會跳過這個搭鷹架的步驟，直接搭建輻射絲，因為這個行為只是蜘蛛自己為了建出整齊的絲線而做的行為，也有相當多的蜘蛛也跳過了這個步驟。附帶一提，在 Samuel Zschokke 的網頁中[5] 有錄下花園十字蛛(*Araneus diadematus*)整個結網過程的影片，我們也有參考這個影片所呈現的現象，並且藉由這個影片做為定義蜘蛛網規則的參考根據。

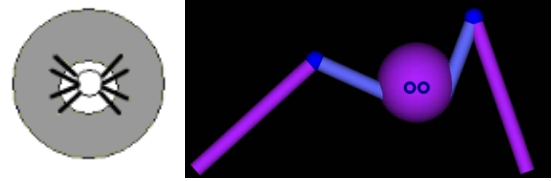
因此我們所模擬的蜘蛛網除了根據[1]的原則之外，也有參考生物學上的定義而組成現在的虛擬蜘蛛網。在稍後演算法的部分會詳盡的說明如何產生出蜘蛛網的過程。

3.2. 蜘蛛構造定義

現在我們知道了有螺旋蜘蛛絲是有黏性的，我們就可以大致決定蜘蛛可行走的區域就是輻射絲。不過值得注意的一點，真實世界的蜘蛛的腳是不會被自己的絲黏到的，因為它們的腳底有特殊的油脂並且接觸面積也小。不過他們還是比較不偏好走螺旋線，因為會減低黏絲的黏度，也會減少能困住獵物的時間。並且在觀察實際蜘蛛爬行時我們發現到蜘蛛在已搭建完成的網子上，對於螺旋線和輻射線上的交點也有者相當高的偏好。

這種現象的原因有可能是源自於結網的蜘蛛因為在網上移動最主要的感官是觸覺以及本體感覺來預測腳的落點，由於結網蜘蛛的視覺相當弱只能感測光線而非像人類一樣可以利用視覺的判斷，所以大致上牠們會有腳向前伸再往後勾線類似摸索的一個動作，這樣的結果也就會很自然的讓腳站在交點上。所以我們在虛擬蜘蛛如何選擇落腳點的偏好也有提高這一方面的喜好程度。

真實的蜘蛛有八隻腳，每隻腳各有七個關節。這在模擬上有相當大的困難由於每個關節都



圖四：蜘蛛可以踩的範圍。

要有個別的自由度和關節彎曲度的限制，單單一隻腳的計算複雜度及各個關節點的位置判斷就非常複雜。既要簡化腳的關節但又要兼顧真實性，我們的做法是將彎曲度較不顯著的關節和其他關節簡化，並使得簡化過的那一段腳有較大的自由度以模擬原本的腳可以踩到的範圍。所以目前在這裡定義出的虛擬蜘蛛的身體結構有八隻腳，每隻腳有各分有上下兩段，也就是兩個自由度兩個關節。事實上，我們可以用肉眼可以看出蜘蛛的腳分為三段，但是我們可以用兩段來模擬三節的腳，也可以達到相同的效果。每隻腳我們設定相同固定的長度，並且每隻腳都是以身體中心為原點。

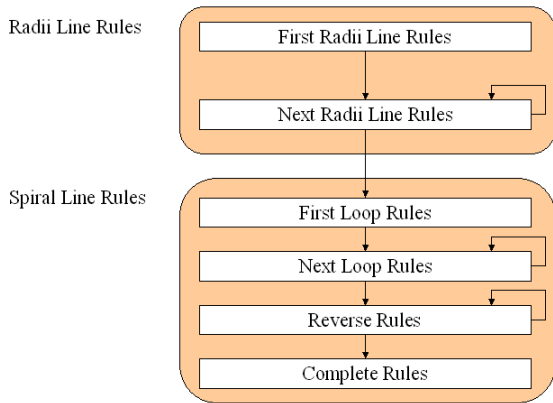
關於身體的部分，真實的蜘蛛身體分為兩段頭胸部和腹部，腳都分布在頭胸部的部分。在這邊的定義則是簡化為一個球體並有自己的座標以及旋轉的角度，這三個元素加起來就是身體的組態(configuration)，如圖三所示。身體後方的橢圓只是為了外觀上的完整而加上的，不具特別的意義。我們假設蜘蛛的腳有內外關節的限制；所以向內向外可以踏的範圍有限制，所以我們把這個範圍畫成一個甜甜圈型如圖四，以身體中心為圓心的灰色甜甜圈就是蜘蛛可以踩的範圍。

4. 虛擬蜘蛛網的建造規則

我們以 Rule-base 的方法，依據自然界的原則[5] (如圖五)來建造蜘蛛網。圖五中的步驟是由上而下的順序先後建出蜘蛛網；規則可以分成兩類，第一類是建造輻射絲的原則(Radii Line Rules)，這一類的規則在使用者定義完外框之後就可以開始實做建造輻射絲。第二類螺旋絲建造原則(Spiral Line Rules)是做完第一類的規則之後才能實做螺旋絲。完成以上兩類規則就可以製出蜘蛛網。只要使用者輸入的外框是凸多邊型就是合法的，我們就可以使用以上兩類規則產生虛擬蜘蛛網。除了給定外框外，我們還需要指定四個參數來產生蜘蛛網，關於這四個參數請詳見下節。接下來針對這兩類規則中的細部規則做更詳細的敘述。

4.1. 輻射絲規則(Radii Line Rules)

使用者傳入的參數中與輻射絲的建構有關的兩個。一個是調整輻射線的數目，輻射線的架設就是取決於蜘蛛的測量與蜘蛛的腳長有關，因為蜘蛛會用自己的腳像是尺量看看兩條輻射線的距

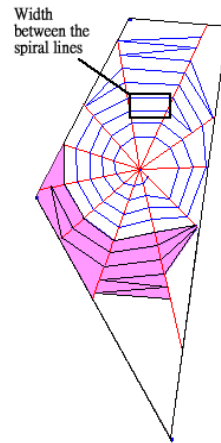


圖五：蜘蛛網的建構原則

離是否太寬。如果太寬的話再多加一條輻射絲，所以簡單的說腳越短的蜘蛛在相同面積的網子上做出的輻射線會越多。另一個是螺旋線之間容許的角度誤差值；蜘蛛不會做出非常精確平均分配輻射線之間夾角的蜘蛛網，所以需要誤差值來讓網子看起來比較自然。這問題其實並沒有確切的數據可以告訴我們，就算是同品種的蜘蛛也會因為不同的個體依照不同的習慣去做出不同的網子，所以這方面算是蜘蛛本身的偏好。關於演算法部分，首先在得到外框和參數之後，依照第一類規則(Radii Line Rule)的第一項朝特定方向建造的第一條輻射絲(First Radii Line Rule)。再依據第二項規則(Next Radii Line Rule)及使用者給定的兩個參數，得到下條輻射絲所在的地方。重複第二項規則直到沒有空間放輻射絲或是輻射絲的數目到達使用指定的數目。

4.2. 螺旋絲規則(Spiral Line Rules)

使用者指定的另外兩個參數與螺旋絲的規則有關。一個是指定螺旋絲之間的寬度，就是圖五中的粗邊矩形中兩條細線的間距。螺旋線的寬度也是取決於蜘蛛的腳長，螺旋線之間的寬度絕對不會大於蜘蛛橫向的身長。這是因為在建造螺旋線時蜘蛛是一邊沿著已經建好的螺旋線一邊量距離一邊搭線，並由實際觀察的結果也能證明此原理。另一個參數是離中心點多遠才開始放置螺旋線的距離，這裡我們將中心點到最內圈螺旋線之間的部分叫做中心網(hub)，這點也是因為蜘蛛的偏好而決定的。接下來就是螺旋線的演算法部分。一般來說，蜘蛛會離中心軸一段距離才開始放有黏性的螺旋絲，我們可以從剛剛使用者指定的參數得到數值。依照實際觀察，通常蜘蛛網的輻射絲中心會在比較偏高，原因是因為蜘蛛通常在中心網(hub)的地方等待獵物，並且他往下衝的速度比往上爬的速度快了許多。此現象除了是因為地心引力的關係外，也是基於蜘蛛行走的特性。蜘蛛一般由上向下快速移動時會將自己的絲線一端黏在現在的位置上，再往下一跳助跑再爬，所以蜘蛛在蜘蛛網上或其他地形向下跑會比



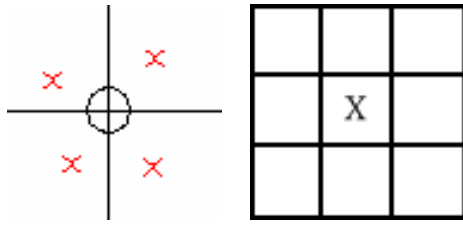
圖六：當外框是狹長型的情況，蜘蛛會想辦法填滿網子，利用每一個角落。陰影的部分就是補線

較快。因此我們會把蜘蛛網的中心向上偏移的現象也加入這個模擬程式，不過跳躍的行為並不列入這次的模擬。

我們取得 hub 的參數之後就可以放置螺旋線。我們放置蜘蛛絲的做法是先決定好在那一條輻射絲開始放螺旋絲，開始用第二類第一項規則(First Loop Rule)。在螺旋線繞完一圈後回到開始蜘蛛絲放置的螺旋線上的，其距離中心點是使用者設定的參數「輻射線之間的寬度」加上 hub 的距離。接這就進行第二項規則(Next Loop Rule)，依照使用者輸入的寬度和上一圈的螺旋絲保持距離產生絲線。當蜘蛛重複做螺旋絲的動作做到無法輻射線與外框之間沒有足夠的空間可以放螺旋線的時候，就開始遵照第三項規則(Reverse Rule)反向再做第二項規則(Next Loop Rule)的動作。最後再檢查是否有把整個蜘蛛網填滿。所謂「填滿」的意義是，真實的蜘蛛不會讓最外圈的輻射絲到外框之間有太大的空隙，因為這樣浪費了蜘蛛網的空間，讓能獵補的空間少了許多，所以他們會盡量把螺旋線填滿任何一個可用的角落，如圖五所示。這就是最後一項規則(Complete Rule)。

4.3. 特殊狀況及實做考量

關於設定外框，本系統不同於過去其他研究的做法在於我們允許使用者自由設定蜘蛛網的外框，而外框的形狀一定要是凸多邊形。需要此強制要求的原因是蜘蛛一般不會把網子的形狀編成凹多邊形。因為一來這樣的網子不如凸多邊形可以有效的利用每一個角落而會有浪費空間的現象。二來凹多邊形的形成有可能是因為外圍支架的張力不均而導致網子比較容易纏在一起。所以我們假設外框皆為凸多邊形。由於周遭地型的關係，蜘蛛所搭建的外框不一定會是完整的圓型，而是以長型居多。長型的外框會使得上下兩側最接近外框的螺旋絲的距離變遠。因此都需進行螺旋絲線的折返修正，以填補空隙，讓做出的網子與真實蜘蛛網更為接近。



圖七：(左圖)蜘蛛需要踩在每個象限的其中一個立足點這樣才能算是平衡。(右圖)X 代表蜘蛛周圍八個空位代表蜘蛛下一步可能選擇的方向。

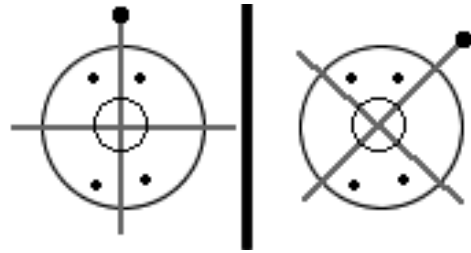
雖然透過誤差值參數的設定，每一次的編織出來的蜘蛛網都不盡相同，但要考慮所有的變因，以模擬完整的蜘蛛網建構過程，仍是一件困難的工作。例如，部分蜘蛛網會有殘缺的情形發生。對這些不易掌握的變因，我們以提供人機界面的方式，讓使用者能以做部分的修改。例如，我們在程式中增加了”剪線”的功能，也就是把網子弄得殘缺再讓蜘蛛行走，做法就是藉由點選螺旋絲的線段移除該線段。一般蜘蛛在獵捕蟲子時會趁獵物在蜘蛛網上掙扎的時候衝到他面前再用自己吐出的絲和部分的網子將獵物打包好，所以會有出現網子殘缺的現象，在實際上有些蜘蛛會忍受殘破的網繼續待下去，也有些蜘蛛會補網。補好的網子會和原先的網子又些不同，在螺旋線的部分就會在破掉的區域來回折返由內側補到外圍。

整體而言，虛擬蜘蛛網和真實的蜘蛛網不像的評估工作，由於缺乏生物學上的統計資料，我們只能交給使用者來做主觀的評估，而非像[1]利用和真實蜘蛛網的上下左右的絲線總長度來制訂基因演算法中的適應函數，以比較評估來決定是否產生出理想的蜘蛛網。儘管如此，我們覺得目前透過系統控制版的參數調整，我們可以找出一個滿意的答案，並且也可以製造出一個適合蜘蛛機器人行走的環境。

5. 計畫及模擬蜘蛛的運動

在[2]中將蜘蛛機器人（在機器人學中將規劃的本體統稱為機器人）的運動規劃分為全域(Global)和區域(Local)兩種規劃方式(planning)。全域規劃(Global planning)不考慮機器人是如何移動腳步，而只考慮如何將機器人身體從起點移到終點，且不會碰到障礙物的規劃。區域規劃(Local planning)則是考慮腳步的移動，以從目前的站立走到下一個站立的位置，達成全域運動的目標。

假設我們目前考慮的是平面的蜘蛛網。因此全域規劃所使用的組態空間(Configuration Space)是一個包含蜘蛛 X 和 Y 軸座標及旋轉角度 θ 的三維空間。因此，我們將機器人的全域組態表示成 $q = (X, Y, \theta)$ 。全域路徑規劃的目的在將蜘蛛從目前的位置，找到一條合法的路徑，讓蜘蛛可以到達所欲



圖八：甜甜圈中的四個小黑點是蜘蛛的立足點，灰色的軸上的黑圓是蜘蛛頭朝向的方向，左邊是合法的站位，右邊是不合法的，因為在四個象限的站立點有兩個象限是空的。在相同的(X,Y)座標位置會得到相同的立足點，但是會因為蜘蛛的朝向角度的不同，可能會產生合法或不合法的組態。

前往的目標地點。本論文的重點將放在全域路徑的規劃上，至於區域運動的規劃，由於環境中暫時無突出的障礙物，因此在全域路徑找到以後，蜘蛛的腳如何移動將是較簡單的關鍵格內差問題而已。因此，以下將以介紹全域路徑的規劃方法為主。

5.1. 蜘蛛在蜘蛛網上的合法落腳點

在開始規劃蜘蛛路徑之前，我們必須對蜘蛛的行動能力與限制，進行瞭解及問題的定義。根據第 3.2 節的定義，我們假設蜘蛛的腳所能採到的範圍，是一個以蜘蛛中心為圓心的甜甜圈區域。而落在此區域的蜘蛛網節點，便是蜘蛛能落腳的落腳點。我們將此類合法落腳點的集合記為 $H(q)$ 。

一個合法的蜘蛛組態 q_{valid} ，除了必須不能與環境中的障礙物碰撞外，還必須能穩定而平衡的站立。根據三腳架的原理[4]，以身體中心為原點可畫出四個象限，只要讓蜘蛛的其中四隻腳在個別不同的四個象限（如圖七所示），我們就可以確定蜘蛛是在平衡的位置。雖然這只是個充分而非必要的條件，但與一般蜘蛛所使用的原則十分相近，因此我們以此簡化的原則定義蜘蛛站立的合法性。根據這個原則，一個合法的蜘蛛組態，必須讓蜘蛛能在這四個象限中找到至少一個點落腳。值得注意的是，對相同的甜甜圈區域，由於蜘蛛面向角的不同，其合法性有可能產生不同的狀態。例如，蜘蛛在旋轉的時候是以身體中心為圓心做旋轉，但在旋轉過程中同一組落腳點可能由合法的站位變成不合法的。圖八中落腳點便是一個的例子。

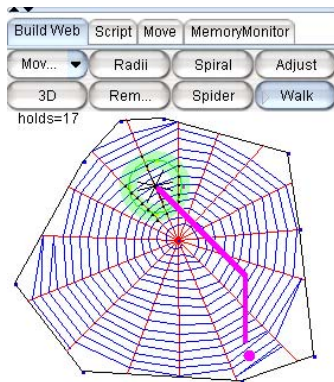
5.2. 蜘蛛移動的合法性

接下來要解決的問題是蜘蛛如何在網上移動。要解決這個問題，我們必須知道蜘蛛腳與落腳點的配對以及蜘蛛如何前進與換腳。

我們依照實際觀察，得知蜘蛛的行走模式是用左邊第一、三隻腳和右邊第二、四隻腳一起挪



圖九：蜘蛛如何利用甜甜圈的可採範圍取得下一步的資訊。

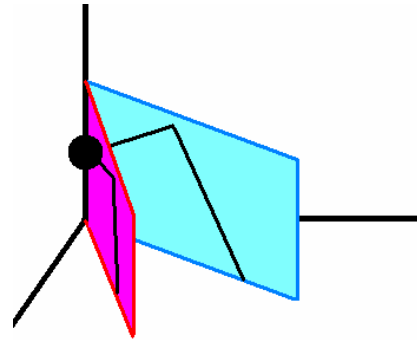


圖十：顯示蜘蛛經過程式計算後產生到終點的路徑。粗線代表的是蜘蛛從起點到終點的路徑，圓點是終點。動之後再讓左邊第二、四隻腳和右邊第一、三隻腳移動。蜘蛛便是重複著這兩組的模式進行身體的移動。因此我們採用的模擬方式即是以四腳為一組的方式交互運動。由於我們設定蜘蛛的每隻腳都有其關節旋轉程度的限制，我們可以計算出蜘蛛會選擇用哪一隻最適合的腳站在上面。

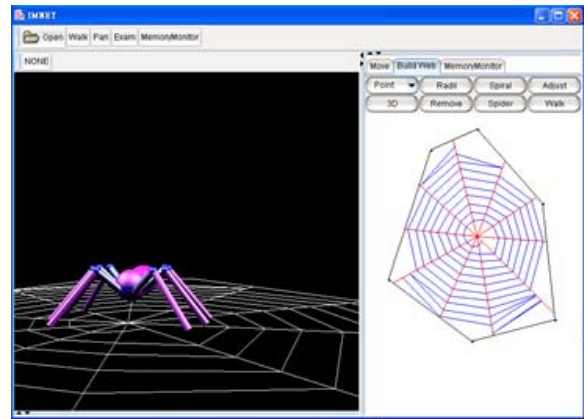
接下來要決定的是蜘蛛如何合法行走前進。換言之，我們要決定蜘蛛如何移動身體重心至新的組態，並進行換腳。一個合法的下一步，必須是現在的落腳點能延伸到的蜘蛛組態。而此新的組態必須能提供另一組合法的落腳點，供另外四隻腳進行區域運動的替換。如圖九所示，黑點是目前站立的點，淺色甜甜圈是它目前身體位置的腳可到達範圍，深色甜甜圈是身體位置向前移動但不動腳可到達的範圍。所以黑色的點一定落在兩個甜甜圈交疊的範圍。再來就可以決定新的腳的落點了也就是白色的點。

5.3. 蜘蛛路徑的規劃

根據上述兩小節有關組態合法性及移動合法性的定義，我們可以開始進行蜘蛛身體移動的規劃。我們所使用的演算法是低維度運動規劃問題常用的最佳優先搜尋演算法(Best-First Search)。在此搜尋法則下，蜘蛛可以朝八個方向移動(如圖七右圖)，每一格代表的一個單位座標長度。從八個候選位置中篩去不合法的組態，再從剩下的找出距離終點最近的點移動。只要違反上兩小節所述有關組態或移動的合法性，便是一個不合法的鄰



圖十一：在 3D 環境中的蜘蛛



圖十二：立體蜘蛛機器人與蜘蛛網

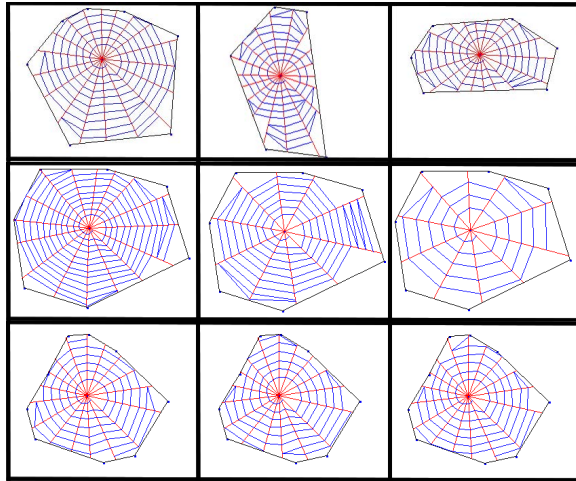
居組態。每一個組態都會記錄蜘蛛上一個位置是從哪裡移到目前這個組態，如果走到無法再移動的時候，就追溯回去以前的組態，對於還沒走過的組態再進行搜尋。圖十中的範例，便是以此演算法所規劃出的路徑結果。

6. 系統實做與實驗結果

6.1. 系統實做

本系統立體化的實作與介面是使用 IMLAB (Intelligent Media Lab)所開發的環境 IMBrowser 來進行 3D 化的呈現，以及動畫的撥放。人機介面包含 2D 與 3D 的呈現部分，使用者在 2D 的環境控制參數產生蜘蛛網與蜘蛛機器人互動，然後透過 3D 的場景呈現成果。要將 2D 資料還原成 3D 動畫，我們需要對蜘蛛的運動，做進一步的計算。例如，我們把蜘蛛步行時一隻腳的挪動看成是一個面的挪動。如圖十一所示，淺色的腳是之前的位置，深色是現在腳的位置。在一隻腳的平面上，我們可以用反向動力學(Inverse kinematics)的公式及腳的落點，反推出所有關節點的位置，而蜘蛛的 3D 資料也就能進一步還原。

在蜘蛛網的實作上，我們在所定義的 2D 蜘蛛網座標上，多加一個高度的參數值，以算成 3D 座標。實際程式的結果在圖十二，右邊的控制面



圖十三：程式產生的蜘蛛網範例。最上排的表示系統可以適應不同的外框，產生出結果。中間一排是相同外框加上不同參數產生的結果。最下方的是相同外框與相同參數可以產生不同的結果。

板是之前的產生蜘蛛網與蜘蛛步伐的程式與 2D 部分的呈現，左邊面板是呈現蜘蛛與蜘蛛網的立體化結果。

利用上一段提到的反向動力學決定腳擺放的姿勢與關節角度的計算後，我們透過 IMBrowser 所提供之動畫劇本輸入格式(XAML)，將計畫出的蜘蛛步伐以關鍵格(key frame)的方式，指定蜘蛛的組態，再利用 IMBrowser 的內插功能將 3D 動畫播放出來。

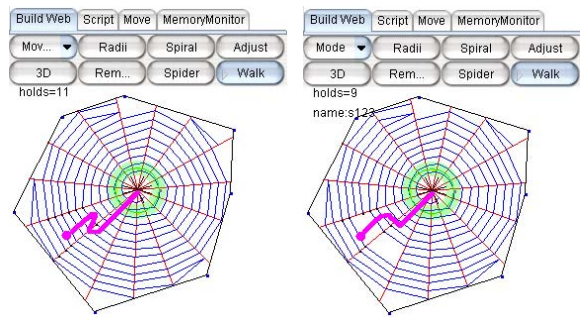
6.2. 實驗結果

6.2.1. 蜘蛛網

蜘蛛網的組成是由外框、輻射線、螺旋線所組成的；在使用者定義好外框之後，再透過系統所提供的參數調整，系統便可依據前節定義的規則，計算出蜘蛛網其餘的部分，得到各式各樣的蜘蛛網。由於誤差值的設定，包含亂數的計算，因此就算是相同的參數設定，也會產生不同的蜘蛛網結果。

在這個單元中，我們會針對系統適應外框的多變性、相同外框面對不同的參數值和相同外框面對相同的參數值這三種情況進行實驗和評估。而圖十三便是將這三種狀況呈現出來；最上一排的三張圖分別用不同的外框配合相同的參數產生出的結果。本系統只要是凸多邊形的外框就可以視為合法的外框，不但具有適應性也有依照結網規則做出符合定義的蜘蛛網。中間一排三張圖代表用相同外框不同參數產生的網子，在螺旋絲之間的寬度和輻射線的數量上都有明顯不同的設定。

6.2.2. 蜘蛛步行



圖十四：粗線為蜘蛛規劃出來的路徑。我們把右邊的蜘蛛網部分的螺旋絲刪除，改變蜘蛛行走的環境，蜘蛛規劃的路徑也會因此改變。

蜘蛛規劃出的路徑會因為環境的改變而有不同的結果。如圖十四，左圖是完整的蜘蛛網，由於可以踩的踏腳點有很多且分布平均，所以蜘蛛前進的方向幾乎是以直線到達獵物所在地。右圖的蜘蛛網是經過使用者剪裁過的，可以讓蜘蛛踩的踏腳點也會因此減少，導致蜘蛛必須搜尋其他要繞過這個障礙的路徑到達終點。從這個例子可以了解系統可以適應地形的改變，產生到終點的路徑。

7. 結論與未來發展

這篇研究的獨特性在於模擬蜘蛛在蜘蛛網上的行走規劃及讓使用者能以互動的方式影響蜘蛛和蜘蛛網的模擬結果。蜘蛛網的構造及設定上都是經由我們從自然界的觀察所歸納出來的結果進行模擬。

之後我們希望利用目前做的結果作為基礎，發展更進一步的模擬。例如在蜘蛛網的方面，增加修補網子的功能，改變蜘蛛在蜘蛛網上的立足點，使搜尋路徑的變化程度增高。在互動性的部分，我們將來預計讓使用者可以動態的改變蜘蛛要到達的目的地，也就是可以隨時在蜘蛛正在朝目標移動的時候即時改變終點的位置。另外，我們希望未來能更簡化參數值的設定，讓使用者直接設定蜘蛛的大小和腿長這兩個參數，就可以得到相對的蜘蛛網；也就是利用更高階的參數設定取代目前直接影響網子的低層參數，達到更直覺方便使用者的參數設定。

8. 參考文獻

- [1] T. Krink, and F. Vollrath, "Analysing spider web-building behaviour with rule-based simulations and genetic algorithms," in *Journal of Theoretical biology*, No.185, P.321-331, 1995.
- [2] C. Eldershaw and M. Yim, "Motion planning of legged vehicles in an unstructured environment," in *Proc. of 2001 IEEE International Conference on Robotics and Automation*, 2001.
- [3] T. Bretl and S. Rock and J.-C. Latombe, "Motion Planning for a Three-Limbed Climbing Robot in

- Vertical Natural Terrain,” in *Proc. of 2001 IEEE International Conference on Robotics and Automation*, 2001.
- [4] J.-D. Boissonnat, O. Devillers, and S Lazard, ”Motion Planning of Legged Robots,” in *Proc. of Workshop on the Algorithmic Foundations of Robotics(WAFR’94)*, 1997.
- [5] S Zschokke, ”Web construction”
<http://www.unibas.ch/dib/nlu/staff/sz/webconstruct.html>
- [6] S Zschokke, ”The influence of the auxiliary spiral on the capture spiral in *Araneus diadematus*,” *Bulletin of the British Arachnological Society* 9: 169-173 ,1993.
- [7] S Zschokke, ”Form and function of the orb-web,” in *European Arachnology 2000 (Toft S & Scharff N, eds)*, pp. 99-106, Aarhus University Press, Aarhus, 2002.
- [8] T. Krink and F. Vollrath, ”Emergent properties in the behaviour of a virtual spider robot,” in *Proc. of the Royal Society*, pp. 2051-2055, 1998.
- [9] T. Bretl, T. Miller, S.M. Rock, and J.C. Latombe, ”Climbing Robot in Nature Terrain,” in *Proc. of 7th Int. Symp. on Artificial Intelligence, Robotics and Automation in Space*, Nara, Japan, May 2003.
- [10] B. M.Blumberg and T. A. Galyean, ”Multi-Level Direction of Autonomous Creatures for Real-Time Virtual Environments,” in *Proc. of ACM SIGGRAPH*, 1995.
- [11] X. Tu, and D. Terzopoulos, ”Artificial Fishes: Physics, Locomotion, Perception, Behavior,” in *Proc. of SIGGRAPH*, 1994.
- [12] K. Sims, ”Evolving Virtual Creatures,” in *Proc. of SIGGRAPH*, 1994.
- [13] J. Funge, X. Tu, and D. Terzopoulos, ”Cognitive Modeling: Knowledge, Reasoning, and Planning for Intelligent Characters,” *Proc. of SIGGRAPH*, 1999.