

1. 前言

參數化設計並非是個全新的概念，1960 年代電腦輔助設計的先驅 Ivan Sutherland 在他的博士論文中，就是以參數化的概念做為他 Sketchpad 系統的核心，透過所設計的介面，修改參數值以獲得不同的設計元件，這些觀念在早期電腦輔助機械製造中開始應用。今日電腦軟體技術的進步，除了維持著當年 Sutherland 的 Sketchpad 之精神，運算速度更快與演算法更複雜，導致近年來參數化系統在電腦輔助建築設計中受到廣泛的重視。在許多電腦輔助建築設計套裝軟體中，內建的參數化圖塊與圖庫，例如：門、窗、樓梯、...等成為不可或缺的輔助設計工具，而更友善的介面方便設計參數之客製化，讓不諳電腦程式的設計者也可以設計個人的客製化參數元件。參數化設計工具使得建築設計產生了許多嶄新的概念以及革命性的影響，因此參數化的電腦輔助設計方式在介入設計過程中開始扮演有別於以往單純工具性的角色，在建築設計的思維上產生了觀念上的重大變革。

目前許多電腦輔助設計軟體中，都具有可以調整參數的樓梯或浴廁設計系統元件，這是參數化設計最直接的使用。早期的參數化設計大都仰賴套裝軟體中既有的參數化元件，近年來因為軟體的友善介面與強調「客製化設計」，所以「腳本」(Script，也就是電腦程式)開始被注意與使用，例如 Rhinoceros (犀牛) 因為使用簡單的 BASIC 電腦程式語言，操作入門門檻不高，很容易學習。Rhinoceros 的免費插件 (Plug-in) Grasshopper (蚱蜢) 以「關聯式模型」(Associative Modeling)，連結 (Link) 系統中所設計好的「元件」(Component)，間接達到「腳本」的效果，讓不懂撰寫電腦程式的設計者也可以嘗試參數化設計；Grasshopper 因為是免費插件軟體，目前在許多建築學院是練習參數化設計的入門工具。

2008 年威尼斯建築雙年展時，倫敦 Zaha Hadid 建築師事務所的合夥人 Patrik Schumacher 發表了《Parametricism as Style—Parametricist Manifesto》(參數化主義的建築風格—參數化主義者宣言)，參數化設計似乎成為都市和建築發展中新的課題；一種以關聯式的幾何建構方法，透過數位工具和腳本，產生一連串連續的但卻又是差異性的設計 (Continuous Differentiation)，邏輯性的參數操作，產生參數化的建築設計。有別於傳統設計透過特定條件與經驗決定設計形式，參數化設計能動態地調整參數關聯因子，即時產生結果，調整、分析、評估設計方案的幾何形體。參數化設計的優勢是將與設計有關的參數邏輯透過電腦運算解決，即時產生設計結果，使得建築設計流程變成一種回饋性的機制。

2. 研究目的

新的設計思考方式與思維：許多當代建築應用數位技術，例如：Frank O. Gehry 所設計的畢爾包古根漢美術館應用 CATIA 系統進行參數化設計，並結合數位製造將建築物完成。澳洲 RMIT 的 Mark Burry 教授所率領的團隊，根據 Antonio Gaudi 於巴塞隆納所遺留下來聖家堂的設計圖稿與元件並加以參數化，加速了未完成的聖家堂的建造過程，並同時以數位製造的技術，讓 Gaudi 的遺作得以在更短的時間內完成。

由大量製造到大量客製化：今日的設計需求已經由現代主義所強調的「大量製造」(Mass Production) 轉變成為「大量客製化」(Mass Customization)，除了消費激增、市場競爭、與個

性化等諸多因素外，「量身訂做」似乎是更合情合理的設計態度。這背後當然也受到數位設計與數位製造的推波助燃，「參數化」是「大量客製化」這設計典範轉移的重要技術核心。

更具經濟效益的初步設計方案研究：以 Gensler Architects 於中國上海所設計的 Shanghai Tower 超高層建築為例，過程中 Gensler 將三角形的平面逐層旋轉，以參數化設計測試由地平面至塔頂平面分別旋轉 60、90、120、150、180、與 210 度測試側向風力，最後決定以旋轉 120 度的流線形的造型；這造形卸除了 24% 的側向風力，在這案例中每減少 5% 的風力，約可以節省 1200 萬美金的結構費用（超高層建築的樑與柱的斷面面積可以因此而縮小，因為所受側向風力減少的緣故），成功的以合理的造型詮釋結構的問題。

高階的設計符號與系統思維：最近所發展的參數化設計擺脫了傳統電腦輔助繪圖中以鍵盤與滑鼠將一條條的線段輸入電腦螢幕上，以高階的幾何「元件」（Component）定義所設計物件之間的關係，「元件」至少包括輸入、輸出、物件屬性與屬值、幾何條件與限制、資料結構等。大部份的情況下都不需要點選繪圖軟體選單（Menu）上的繪圖符號，而是藉由定義「元件」之間關係的定義以達到描述設計的作用，也就是說設計者面對新的參數化工具，將以相較於以往更「高階」的語言模式進行設計推理與思考。

新工具與新形式：因為在參數化設計中，所有的設計線段不再是一筆筆鍵入，所以只要調整設計的參數就非常容易修改，可以大量減少以往設計修改時得重新建模的麻煩。另外，面對日益新穎的建築造型，許多造型已經無法以序列的線段描繪出來。其實早期這些造型都得仰賴逐行的參數化「腳本」的撰寫，目前也有些撰寫參數化「腳本」的高手將已經寫好的「腳本」編撰為免費插件軟體供人下載，重點是這些幾何形體與造型都是得經過運算得出來的，結合數學、邏輯與幾何，成為詮釋新一代的數位建築形式的方法。

因應建築設計中複雜的運算的趨勢，我們也可以在一些國際大型的事務所中開始發現到，例如：Norman Forster 建築師事務所的「特別模型專家小組」（Specialist Modeling Group），與 Zaha Hadid 建築師事務所的「設計運算研究小組」（Computational Design Research Group），根據不同設計方案的需求，派遣出小組中的成員支援，這些成員都具有參數化設計能力與電腦程式撰寫背景的專家。

3. 文獻探討

大部份三維繪圖軟體都提供腳本（Script）編撰的功能，例如：AutoCAD 的 AutoLISP、SketchUp 的 Ruby、MAYA 的 MEL（結合 UNIX 與 Java 的語法）、3DMax 的 MaxScript，與 Rhinoceros 的 VB Script、C#、Python，或是為不諳程式設計者所發展的 Grasshopper（VB.net、C#、Python）等；ActionScript（Java）、Processing（Java）亦可以編撰程式與產生圖形化輸出。除此之外，參數化設計的工具最顯著的例子就是 Rhinoceros 的許多插件程式，以資料庫的結構而言，ParaCloud 是以 Excel 試算表的「關聯式」資料庫（Relational Database）的方式呈現；Grasshopper 則是主要以「階層式」資料庫（Hierarchical Database）帶點「網絡式」資料庫（Network Database）的方式呈現。

Colin Rowe 於 1947 年所著的《*the Mathematics of Ideal Villa*》（理想別墅的數學）一書中分析 Andrea Palladio 所設計的 12 棟別墅的幾何圖解如出一轍，他將 Palladio 別墅設計研

究的結果與 Le Corbusier 的加爾歌別墅做一比較，發現類似的幾何圖解。Christopher Alexander 在《*Note on the Synthesis of the Form*》(形式綜合之綱要)一書亦以階層式系統(Hierarchical System)來描述形式綜合的一種方式。雖然 Christopher Alexander 後來轉往「模式語言」《*a Pattern Language*》上發展，然而他在初期確有以電腦程式模擬設計的想法，並嘗試撰寫 FORTRAN 程式以驗證他的想法。在其之後的著作《*The Timeless Way of Building*》中提到參數化牛棚設計的可能，這是一種非常接近「參數化」的設計描述方式：

牛棚從其模式中得到其結構：牛棚有一定的形體，大致上是一個長方形，有一個儲存乾草的中心部份，旁邊的側廊是牛隻站立的地方；中心和側廊之間有一排柱子，沿著這排柱子是供牛吃草的飼料槽，一端有大門或雙門，也許通道的另一端有一個較小的門，以便牲畜進出。以下是美國加利福尼亞傳統牛棚的模式：一個長方形牛棚，30~50英尺寬，40~250英尺長，(長度3X英尺，X是單向牛棚容納牛隻的數量)。把牛棚內部份成三個平行的楹：牛隻在外面兩楹，中楹儲草，中楹16~38英尺寬。有時，一個側楹可以比中間稍短，因而長方形平面有個凹口。中楹邊緣和側楹之間放兩排柱子，等距，選擇柱間距7~17英尺。牛棚屋頂做成對稱坡頂，並使側楹屋面的坡度比中楹屋面的坡度較緩或相等，這樣就在主要柱子上沿著屋頂桁架形成折線，兩種坡度水平角都在20~40度之間。邊柱高約7~10英尺，屋脊高約15~25英尺。

(*The Timeless Way of Building*, by Christopher Alexander, 1979.)

1980年George Stiny所發表的《*Introduction to Shape and Shape Grammars*》則是以造型文法的角度探討圖形構成的法則，William Mitchell《*the Logic of Architecture*》則是以1980年代盛行的人工智慧語言 PROLOG (Programming in Logic) 的語法來說明建築元件與整體之間的關係說明。近年以特定軟體說明參數化設計原理的書籍有 Kostas Terzidis 發表的《*Algorithmic Architecture*》(2003)、《*Expressive Forms*》(2006)，Robert Woodbury 的《*Elements of Parametric Design*》(2011) 與 Arturo Tedeschi 的《*Parametric Architecture with Grasshopper*》分別是以 MAYA、Geometric Component、與 Grasshopper 軟體的使用邏輯說明參數化設計的要素。Roger H. Clark 與 Michael Pause 所著的《*Precedents in Architecture: Analytic Diagrams, Formative Ideas and Partis*》則是將建築作品依照各種不同的幾何類形對於平、立、剖面與量體加以分類整理，以類似 Durand 的類型學方式整理現代建築。

相關的期刊文獻可參考倫敦 Wiley Academic 出版的 Architectural Design (AD) 雙月刊，AD 近幾年來每隔兩期就會討論有關數位建築設計的議題，其中以2002年的159期《*Versioning: Evolutionary Techniques in Architecture*》，2005年的176期《*Design through Making*》，2006年的180期《*Techniques and Technologies in Morphogenetic Design*》、182期《*Programming Cultures*》、184期《*Architextiles*》，2007年的185期《*Elegance*》、2008年的192期《*Versatility and Vicissitude*》、194期《*Protoarchitecture*》、196期《*Neoplastic Design*》、2009年的198期《*Closing the Gap: Information Models in Contemporary Design Practice*》、200期《*Digital Cities*》、202期《*Patterns of Architecture*》，2010年的206期《*New Structuralism*》，2011年的210期《*Protocell Architecture*》，2012年的212期《*Mathematics of Space*》，與2013年的222期《*Computation Works*》與本研究相關；其中各期的主題亦與本研究的子題有著重要的關聯。AD 於2011年所出版的專書《*Computational Design Thinking*》與《*Scripting Culture*》其中數篇文章有著這方面重要的討論。英國倫敦建築聯盟近年的出版品與學生作品集亦可以做為建築教育與未來實驗性方案的參考。

4. 研究方法

本研究的資料蒐集透過網際網路、設計作品集、與相關書籍回顧，整理出系列的「參數化設計」的相關資料，並分析其所仰賴的理論，以及各種不同「參數化設計」的操作技術。並將經過整理後不同的「參數化設計」方式分門別類，整理出使用參數化設計工具各種可能與條件限制。並且對於參數化設計工具所產生的數位形式與傳統「建築形式」之差異，以及「參數化設計」所衍生的問題加以討論。本研究討論正在發展中「參數化設計」的方向與思維，進而對於參數化設計的工具、應用、意義、與精神提出看法。

參數化設計 (Parametric Design)：大部份的電腦輔助設計、製造、與工程 (CAD、CAM、CAE) 軟體都不可避免的運用「參數化」 (Parametricization) 的特性，參數化用以聯接尺寸 (Dimension)、變數 (Variable) 與幾何 (Geometry)，因此當變數的數值改變，所設計的幾何形體也會跟著改變。參數 (Parameter) 是一個與其他變數相關的變數，這些變數的未知值可以透過事先定義好變數之間的關係式來求得。透過參數化的方法，設計的修改與產生許多可能的設計解答可以在極短的時間內完成，相較於傳統電腦輔助繪圖每次修改設計就得重新繪製圖說的複雜步驟，以下幾點是關於參數化設計應用於設計過程中的特性：

- A. **釐清設計原件中各個子物件之間的關係與關聯** (關於建築教育)：參與階段性設計的參數，是由設計條件、限制、材料、經費、效益、... 等因素所決定，參數化設計首先得確定那些是參與設計的關鍵因素，這些因素與幾何形體的關係與數位模型的生產，是參數化設計的特色。
- B. **根據既有案例加以參數化產生類似的設計案例** (關於案例研究)：設計者可以將建築史上的典例將其參數化，而產生類似的建築物，這方面涵蓋部份「造型文法」 (Shape Grammars) 的研究。曾經被探討過的著名建築師包括：文藝復興時期的 Andrea Palladio 在他的別墅系列設計中使用了由古典建築重新發現的永恆與普遍原則，以及與現代主義同期的 Frank Lloyd Wright，與後現代主義時期代表地域主義的 Mario Botta 所設計的建築物。以類似的想法，設計者可以針對階段性的設計解答加以參數化，以推敲出更多設計的可能性。
- C. **根據不同的設計需求發展出相關的參數化模型** (關於系統思考)：在設計進行的階段當中，傳統的電腦輔助繪圖 (Computer-Aided Drafting) 雖然有其便利性，但是當設計的需求部份更改時，無法只修改部份電腦模型，往往得將電腦模型重新製作，非常耗費時間。參數化模型容許設計者以較為寬鬆的設計條件，但是更嚴謹的思考來描述設計，並方便之後進行修改。
- D. **提供不同設計條件與限制下的設計解答與選擇** (關於類型思考)：當設計條件與限制的考慮不同時，有許多可能的可行性方案 (Feasible Solutions)，透過參數的調整 (此時參數對應設計條件與限制)，可以產生許多不同的設計可能性，評估比較相同參數化模型所產生的不同子設計，提供設計者做下一步的選擇，或是最後設計方案的決定。
- E. **以時間為基礎的動態序列形式變化與演化過程** (關於數位形式)：參數化模型中加入了 Timer 的使用，可以觀察既定形式演化的過程，以及整個演化過程的疊加結果。這是傳統的繪圖方式不容易產生，Greg Lynn 曾在其著作《Animate Form》對於這樣的「動態形式」的原理、原因、方法、與應用加以論述。

F. 模擬設計完成後製造端之參數化數位製造實踐（關於數位製造）：今日的參數化設計幾乎可以繪製出所想像得到的形態，因此參數化設計可以產生更複雜的結構、形體、與細部等，設計完成後，如何以系統化的將製造元件參數化，運用數位製造完成是建構上的另一課題。目前數位化的設計案都是在電腦螢幕虛擬無重力的狀況被模擬出來，面對無法以傳統營建方法製造出來的複雜空間與造型，將所設計的三維形體以數位製造技術營建出來將是數位建築設計脫離電腦螢幕之後的另一大挑戰。「數位製造」(Digital Fabrication) 結合新的材料與工法，也將慢慢的成為數位化之後的重要課題，這牽涉到建築設計完成後怎麼生產模具與怎麼蓋起來的問題，例如：瑞士聯邦理工學 (ETH-Z) 院建築研究所著手研發以自動化機械手臂堆砌營建磚牆。此外參數化設計所承諾的如之前所提及，不只是為了大量製造而是大量客製化，為了客製化勢必得產生不同的模具以供製造，但是模具的開發昂貴且耗時，因此參數化模具提供另一個思考的向度，以生產經濟且多樣化的產品。

5. 研究結果

關於研究結果的呈現，擬就上述研究方法中所提到參數化設計的六個重點與意義，分別由以下六個實際操作的案例中描述參數化設計於建築設計思考與教學上的可能及潛力。

A. 此參數化案例操作是假設台北地區一棟具有中庭之展覽館，衡量其周邊的日照與風向，對於其開窗做適當之處理。在 Grasshopper 中建立建築物之參數化模型，並以 Grasshopper 的插件程式 GECO 為介面，將建築物模型經過 GECO 送至 EcoTect 軟體，評估其日照與通風效益，經反覆調整參數後得出較佳之建築物與開口部之造型 (圖 1)。

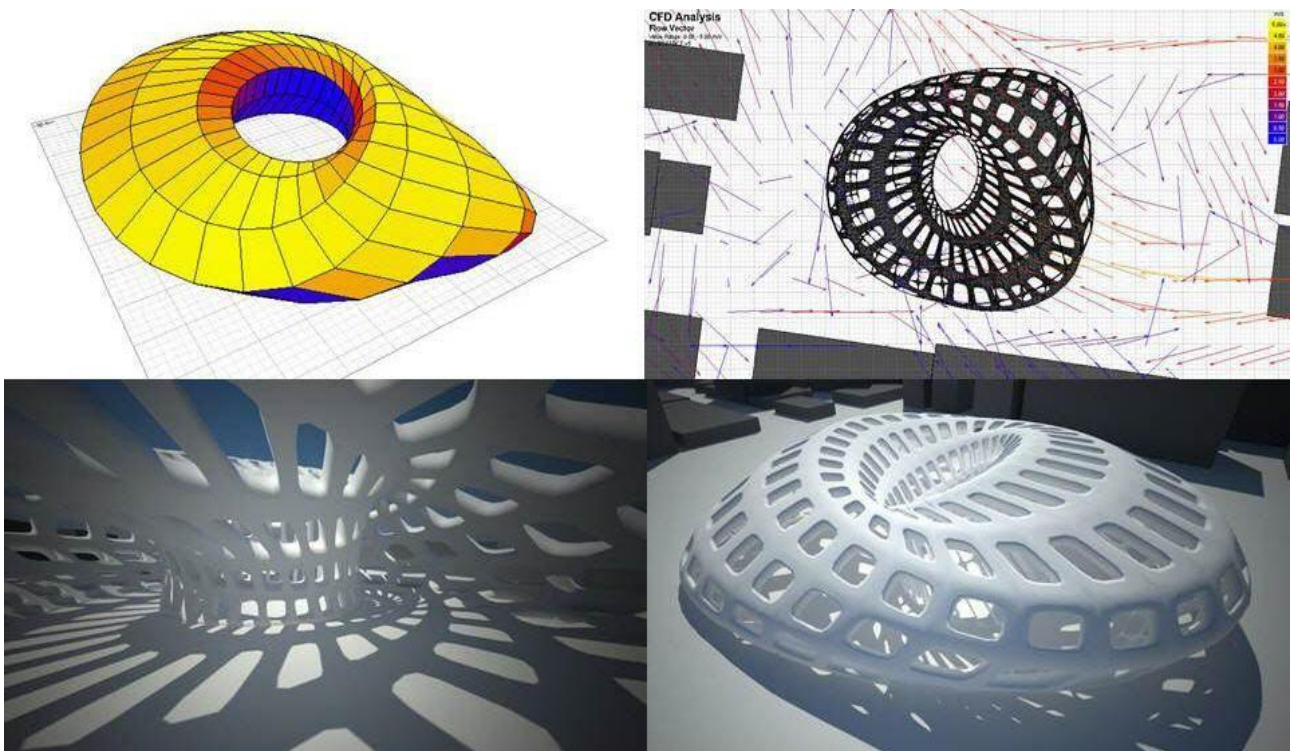


圖 1：以 Grasshopper 插件程式 GECO 對於建築物的日照與周圍風向之模擬，黃雅如製作。

B. Pier Luigi Nervi 於 1957 年所設計的 Palazzetto Dello Sport 與 1968 年所設計的 Norfolk Scope Arena，因為同為 Nervi 先生所設計，因此可以發現這兩座體育館有其非常相似之處。運用同一參數化模型的處理可以將這兩座體育館以不同的參數值繪製出（圖 2A）。之後只要改變參數，就可以改變體育館的幾何形狀、柱子數量、結構、排列關係等。並可以產生 Nervi 未曾設計過的體育館平面、透視、結構提供下一步的設計操作分析（圖 2B）。

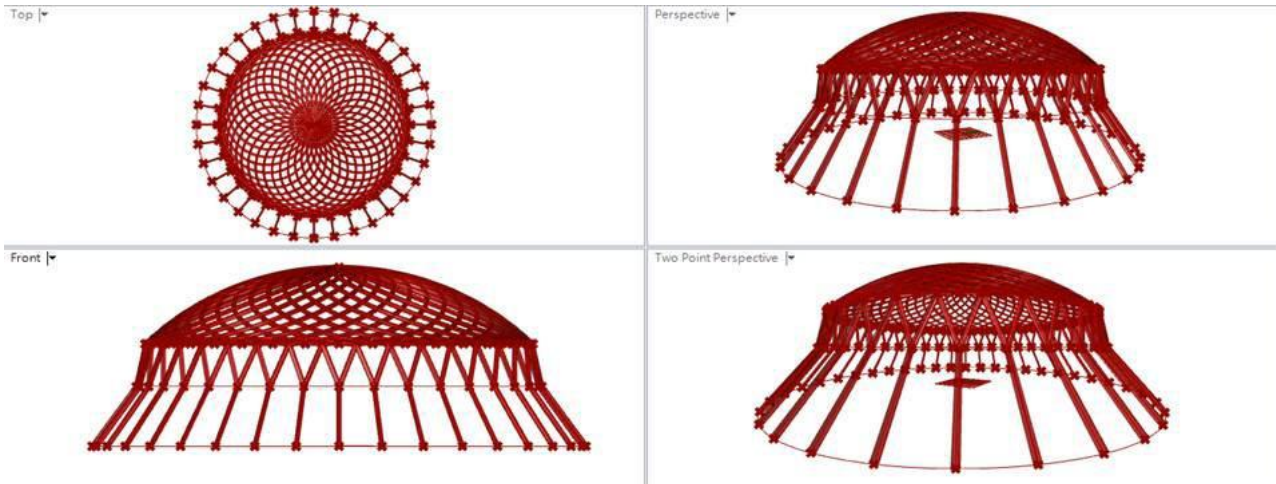


圖 2A：Palazzetto dello Sport 之參數化模型，陳珍誠製作。

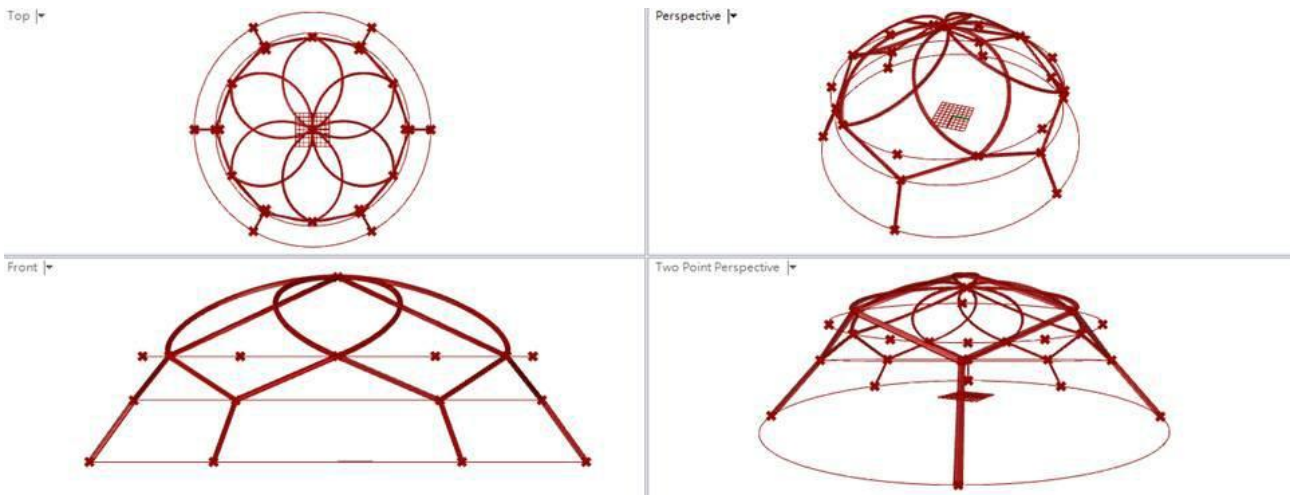


圖 2B：調整 Palazzetto dello Sport 之參數化模型，以產生另一棟可能的體育館設計，陳珍誠製作。

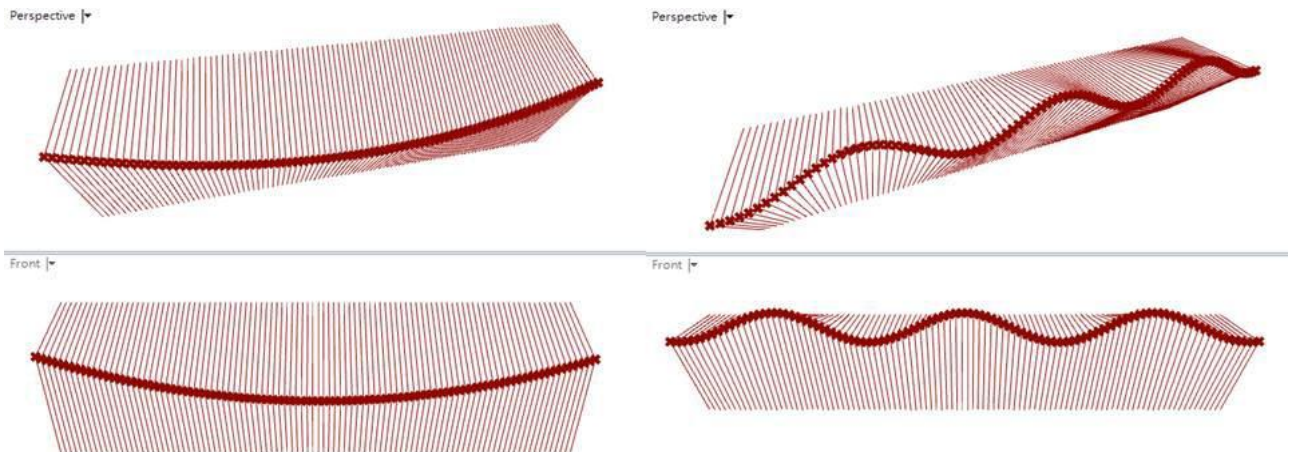


圖 3：Factory Building for Ernsting Company 開口部以不同曲線開啟方式之模擬，陳珍誠製作。

- C. 有別於台灣之鐵捲門是上下平行的水平紋理，Sandiago Calatrava 於 1986 所設計 Factory Building for *Ernsting Company* 之鐵捲門是以由左到右垂直方向為紋理。而鐵捲門中間開啟的折線為拋物線的一部分。經過參數化模型的處理後，可以將拋物線置換為各種不同的曲率或曲線，產生不同的合理造型與鐵捲門開啟方式，在此案例中可以發現到設計細部幾何研究與參數化模型之關聯性（圖 3）。
- D. 假設一個九宮格的平面配置，在其中可以放入 0 至 9 個單元，以參數化模型可以衍生出下列所有可能的設計排列組合（圖 4）。之後可以繼續運用此一模型考慮篩選條件，例如：置入單元的數量、座向、關係、... 等，並且可以更進一步討論置入單元的機能，與不同機能單元之間彼此的相鄰關係，以產生合理的平面。

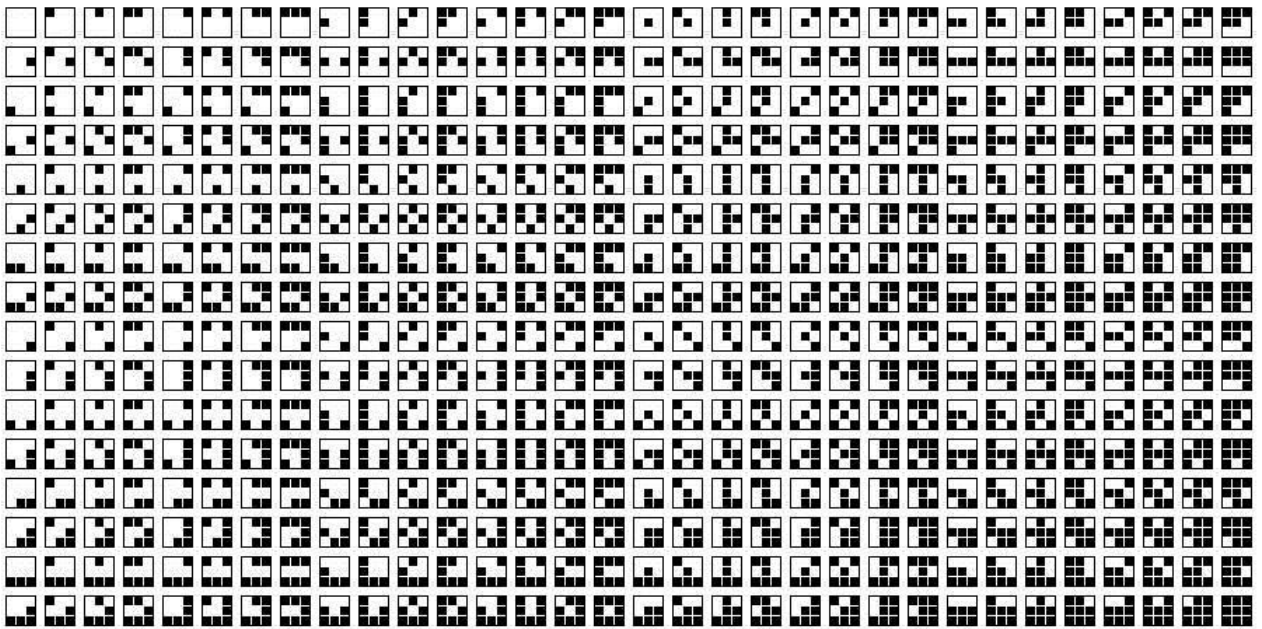


圖 4：九宮格中填入 0 至 9 個單元的所有可能方案，陳珍誠製作。

- E. 以下案例是以電腦模型中的「後設球」（Meta-Ball）為出發點，在第一象限上任意佈點，之後分別對於 X 軸與 Y 軸做鏡射處理，並開始在這些點上佈上「後設球」，並賦予權重 W （圖 5）。當後設球的權重改變時，各點的引力範圍隨之改變，可以產生不同的引力場線圖。隨著時間軸（Z 軸）方向的改變，可以想像成斷面高度上的變化，因此產生類似穹頂式的空間結構，並且可以因為佈點的改變，產生不同的穹頂結構。

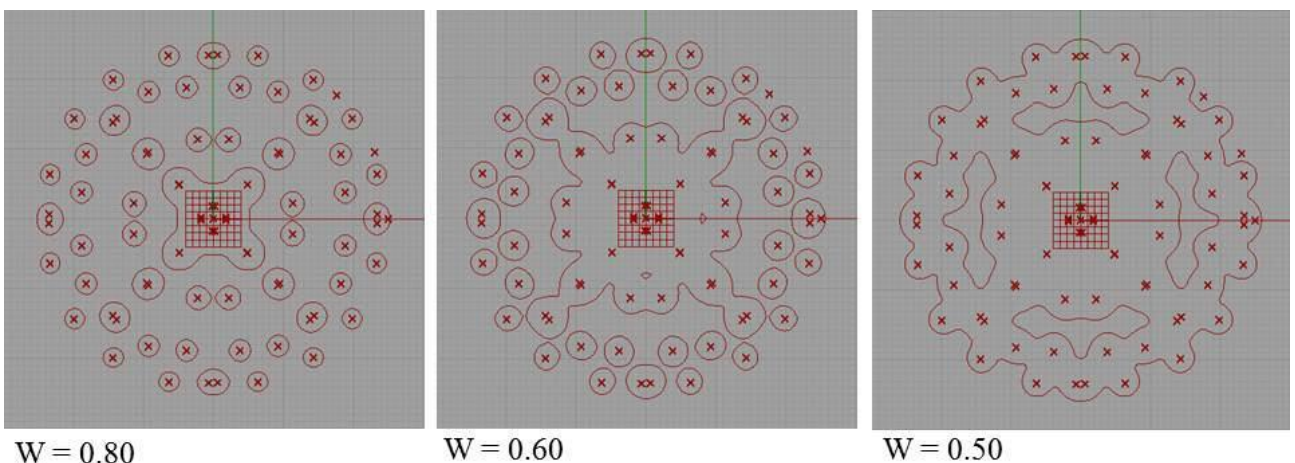


圖 5：以後設球經歷時間序列設定不同球體比重之模擬圖，陳珍誠製作。

F. 以下案例主要應用於數位製造中之模具製作，當模具的三維量體在繪圖軟體中完成後，為了將模具量體以二維切割機完成，在模具往上方（Z 軸）向度上根據板材之厚度，做等分之切割，然後將各個剖切的平面展開，最後將剖切面的周邊線條萃取出，所存圖檔就可以直接提供二維切割機切割。完成各部分剖面元件切割之後，加以組裝即成為模具。以上是數位製造中「加法製造」(Additive Manufacturing)的方法，亦稱為「剖切法」(Sectioning)。當然以這種方法製作出來的模具有縫隙，但是對於精度要求不高的模具開發，是非常的簡便與經濟。

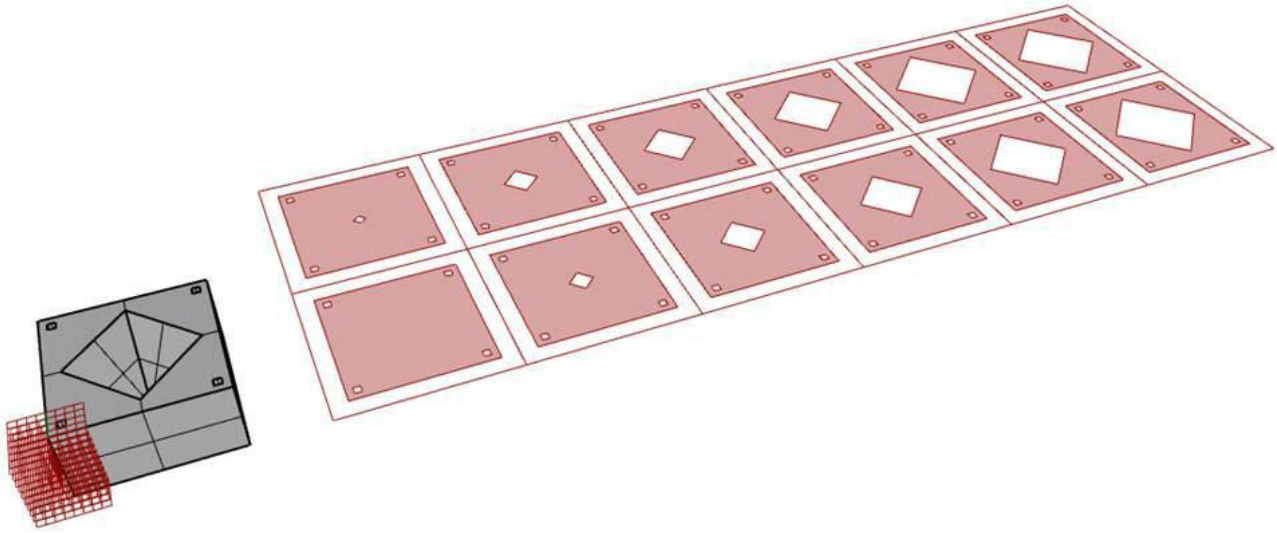


圖 6：參數化模具拆解圖，陳珍誠製作。

6. 結論與建議

參數化對於建築設計過程中的意義：資訊技術的基礎

階層式的系統化設計思維：Herbert A. Simon 在他所著的《the Science of Artificial》（人工科學）一書當中提及，設計系統可以視為一「幾乎可分解的階層式系統」（Near Decomposable Hierarchical System），在這範疇中設計可被視為「定義清楚的問題」（Well-Defined Problem）。William Mitchell 在《Computer-Aided Architectural Design》（電腦輔助建築設計）一書中則更進一步地將「階層式系統」區分為「離散式」（Discrete）與「交錯式」（Overlapping），他強調一般常見的設計問題較不容易以簡單的「離散式階層系統」描述，大部份是「交錯式階層系統」。在此，參數化設計的思考被可以想像成將設計問題切割成不同層級的子問題，子問題又可以被分解為更小的子問題，因此解決設計問題的程序，可以視為階層式系統的解析。

參數化設計的知識再現：以人工智慧「知識再現」（Knowledge Representation）的角度而言，我們可以 Marvin Minsky 所發展的「框架式結構」（Frame Representation）與 Roger Schank 所發展的「腳本」（Script，這裡所謂的 Script 與目前電腦程式設計中所慣稱的 Script 有些許的差異）說明「再現」參數化設計的基本資料結構。當然參數化可以不只包括單一的「框架式結構」，可以是數個或多個「框架式結構」，在高階可以「階層式系統」或語意網路（Semantic Networks）聯接。

參數化設計包含程序式知識與宣告式知識：人工智慧中簡單的將知識分為四類：宣告式知識

(Declarative Knowledge)：分為物體的性質(概念)與物件之間的關係(層級)，程序式知識 (Procedural Knowledge)：執行事情的先後次序，濃縮式知識 (Compiled Knowledge)：可立即使用，有時不太知道原因(如背誦式知識)，與後設式知識 (Meta-Knowledge)：關於知識的知識；這樣的分類法雖然過於簡單與絕對，但是對於知識不同使用的範圍有了大致上的界定，方便我們思考上的辯證。以人工智慧中知識表達模型而言，大抵脫離不了以 Marvin Minsky 所發展的 Frame (框架式結構) 傳達之前所提到的「宣告式知識」，主要處理「物件的知識」與「物件之間的關係」；或是以 Roger Shank 所發展的 Script (腳本) 來傳達「程序式知識」。

由上而下的精煉與由下而上的組合：Clive L. Dym 與 Raymond E. Levitt 所著的《*Knowledge-Based Systems in Engineering*》中提到設計問題的解決模型 (Solution Methods for Design Problems)。他們提出知識型的工程設計系統主要是以下列五種方式處理工程設計的問題：1. 單純的選擇 (Pure Selection)，2. 原型的選擇與精煉 (Prototype Selection and Refinement)，3. 由基本的設計元件組合出單一的設計解答 (Assembling Unique Solutions from Elementary Components)，4. 階層式衍生、測試、消去、與評估以獲得解答 (Hierarchical Generation, Testing, Elimination, and Evaluation of Solutions)，與 5. 類比與變形 (Analogy and Mutation)。簡單的來說，這五種方式中的 1 與 2 是以「由上而下的精煉」(Top-Down Refinement) 來處理「宣告式知識」，3 與 4 則是以「由下而上的組合」(Bottom-Up Composition) 來處理「程序式知識」，這四種方式 John Gero 概括的以「規範式設計」(Routine Design) 統稱之；而第 5 種方式則是就設計的修改，John Gero 則是稱之為「改良式設計」(Innovative Design)，「改良式設計」包括「原型修改」(Prototype Modification) 與「原型組合」(Prototype Combination)；以上的討論可以提供本研究參數化設計「演算」部份的依據。

不同的參數化設計元件之間可以資料結構的連結來表達其關聯性，這些關聯性是由數據流 (Data Flow) 所定義，以「關聯式資料庫」(Relational Database) 與「串列」(List)，或「階層式資料庫」(Hierarchical Database) 與「樹狀結構」(Tree) 等數位模型所組構而成。隨著運算流程不斷地更改資料結構中的數據，直到演算完成後，於資料結構中注入合理的設計數據為止，因此參數化設計亦可視為資料結構與設計程序的關聯網路。

參數化設計的傳統與基礎：建築類型學的演進

建築類型學成熟於歐洲啟蒙時期的法國，起源於人們對於世界的經驗，以及從這種經驗中所得出的某種理性概念。類型學的發展與巴黎美院的學院派建築師有著淵源，Francois Blondel 是第一位於美院講授建築的教授，他本著理性主義尋求普遍的建築原則。他與他的學生如：Eugene Emmanuel Viollet-le-Duc (勒杜) 與 Etienne-Louis Boullée (布雷) 等，建立法國古典建築的傳統。而 Boullée 的學生 Jean-Nicolas-Louis Durand 所著的《古代與現代諸相類似建築物的類型學手冊》則是最早的建築類型學書籍，他在書中一共總結了 72 種建築的幾何組合基本型，這種類型學與圖像學的綜合將建築納入嚴謹的標準化和類型學的系列關係中進行考察。19 世紀 Quatremère de Quincy 在其著名的《建築百科辭典》提及：「類型並不意味事物形象的抄襲與模仿，而是一種觀念，這種觀念即是形成模型的法則，類型則是人們根據思考模型所產生絕不相似作品的概念。雖然類型學的基礎起源於古典建築時期，然而現代主義時期 Le Corbusier 的「多米諾住宅」(Maison Dom-ino)、「建築五要素」與 Frank Lloyd Wright 的大草原住宅風格仍然延續著類型學的部分思考。

建築類型學之特點有三：1.分類是有層次的，分類之下還可以有子分類；2.分類可以根據不同的標準與方法；與 3.不同的分類中仍有共同的成分。雖然類型學算是建築學中傳統思維的一部份，但是上述的類型學特點與現代參數化設計的觀是念息息相關的。類型學的概念中有兩種作用：1.為實踐服務的理論建構—是對於建築分類的大量知識進行處理與組織的綜合方法，包括：理論、規範和歷史；與 2.做為認知的知識結構—為了有效解決設計過程中的問題，將宣告式的知識與程序式的知識聯繫起來。

數位圖解對於建築設計過程中的意義：普通圖解與特殊圖解

帕特里克·舒馬赫 (Patrik Schumacher) 2011 年出版了兩冊的《建築學的自組織系統—建築的新典範》(《The Autopoiesis of Architecture – a New Framework of Architecture》)，舒馬赫在書中說明了「圖解」(Diagram) 與「繪圖」(Drawing) 是建築「交流」(Communication) 的重要的「再現」(Representation) 方式，某種程度我們可以將建築設計過程視為產生「圖解」與「繪圖」的過程，他將圖解區分為：

1. 「普通(陣列)圖解」(Metric-Ordinary Diagram)：Jean-Nicolas-Louis Durand 的類型學圖解與現代主義的正交圖解，方便討論設計基本元素如何組合 (Composition)。
2. 「特殊圖解」(Metric-Extraordinary Diagram)：1980 年代中期至 1990 年代末期的解構建築圖解，是類似哲學家吉爾·德勒茲(Gilles Deleuze)所謂的開放式圖解，運用這些圖解創造特殊的空間形態。
3. 「特殊參數化圖解」(Parametric- Extraordinary Diagram)：1990 年代中期，應用電腦動畫軟體產生開放式圖解，以拓樸學 (Topology) 的變形與動力學處理建築表面以達到「發現形態」(Form-Finding) 的目的。
4. 「普通參數化圖解」(Parametric-Ordinary Diagram)：2000 年以後，參數化圖解逐漸地由特殊圖解轉變為普通圖解，以程式腳本或圖像化網絡語言 (這裡指的是 Grasshopper)，討論設計參數之間的關聯網路 (Associative Networks)。

Schumacher 對於「普通(陣列)圖解」與「特殊圖解」或許可以化解關於建築「機能」(Function) 與「形式」(Form) 爭議，「普通(陣列)圖解」是比較可以使用在「問題解決」(Problem-Solving) 上，而「特殊圖解」比較可以使用在「形式發現」(Form-Finding) 上，當然這樣的分別也不是必然的(在這裡這樣說明是避免分類法上的盲點)。

最後，我們以今天的視覺化腳本(Visual Programming，類似電腦程式中的「流程圖」(Flow Chart)，或是泛指 Grasshopper 的編程介面)中的「連結」與「元件」，來說明 Schumacher 的「參數化圖解」(Parametric Diagram) 中設計的「抽象化概念」(Abstraction) 之不同可能性。參數化圖解可以被視為：1.繪圖(設計)流程圖(將建構設計物件的次序與限制依序排列)、2.設計元件之關係圖(Associative Network)、3.產生設計原型(Design Prototyping)的工具、4.衍生設計(Generative Design)的可能、5.動態形式(Animate Form)演化的圖解過程、6.以時間為基礎的序列過程(例如：Timer 元件的使用)、與 7.路徑圖(機器手臂的行進路線)等。回歸到本研究的出發點，「參數化設計」透過類似電腦程式或流程圖的「邏輯」思維，成為可以被合理化且接受的設計思維，關於「參數化設計」發展的討論可以開啟對於數位建築與建築設計更多面向的了解。

7. 參考文獻

建築理相關文獻：

- 《*Animate Form*》, Greg Lynn, Princeton Architectural Press, New York, NY, 1999。
- 《*Atlas of Novel Tectonics*》, Reiser+Umemoto, Princeton Architecture Press, New York, NY, 2006。
- 《*Note on the Synthesis of Form*》, by Christopher Alexander, Harvard Press, Massachusetts, MA, 1964。
- 《*Structure in Nature is a Strategy for Design*》, Peter Pearce, The MIT Press, Cambridge, MA, 1990。
- 《*The Function of Ornament*》, Edited by Farshid Moussave & Michael Kubo, Barcelona, Spain, 2006。
- 《*The Function of Form*》, by Farshid Moussave, Actar, New York, NY, 2006。
- 《*The Mathematics of Ideal Villa and other Essays*》, by Colin Rowe, MIT Press, Massachusetts, MA, 1976。
- 《*The Science of Artificial Intelligence*》, by Herbert A. Simon, MIT Press, Massachusetts, MA, 1970。
- 《*The Timeless Way of Building*》, by Christopher Alexander, Oxford University Press, New York, NY, 1979。

數位建築相關文獻：

- 《*Algorithmic Architecture*》, Kostas Terzidis, Architectural Press, Burlington, MA, 2006。
- 《*Computational Design Thinking*》, Edited by Archim Menges & Sean Ahlquist, John Wiley & Sons, West Sussex, UK, 2011。
- 《*Cecil Balmond*》, A+U=Architecture & Urbanism, Tokyo, Japan, November, 2006。
- 《*Digital Design and Manufacturing*》, Edited by Daniel Schdek, Martin Bechthold & etc., John Wiley & Sons, Hoboken, NJ, 2005。
- 《*Element of Parametric Design*》, Robert Woodbury, Routledge, Oxon, OX, 2010。
- 《*Expressive Form : A Conceptual Approach to Computational Design*》, Kostas Terzidis, Spon Press, New York, NY, 2003。
- 《*The Architecture of Variation*》, Lars Spuybroek, Thames & Hudson, London, UK, 2009。
- 《*Parametric Architecture with Grasshopper*》, Arturo Tedeschi, Le Penseur, Brienza, Italy, 2011。
- 《*Tooling*》, Benjamin Aranda & Sanford Kwinter, Pamphlet Architecture 27, Princeton Architecture Press, New York, NY, 2006。
- 《*Typological Formations :Renewable Building Types and the City*》, Edited by Christopher C. M. Lee, AA Publications, London, UK, 2007。

其他領域：

- 《*On Growth and Form*》, D'Arcy Thompson, Cambridge University Press, Cambridge, UK, 1961。