中華民國第 62 屆中小學科學展覽會
作品說明書

國中組　數學科

030423

Knights Swap—無標號樹之探討

學校名稱：屏東縣立明正國民中學

作者：	指導老師：
國一 林子濠	陳淑慧
國二 林祺祐	陳盈吉
國二 黃楷宸	

關鍵詞：無標號樹圖、$\mathbb{G}(\mathbb{S} \leftrightarrow \mathbb{P})\mathbb{E}$、權重

摘　要

這是延伸自我們在中華民國第 60 屆中小學科學展覽會提交 Crazy Knights 作品，相對於前次已完成的環圖分析，我們認為非環圖尚有擴充發展的可能。

本研究從已知的騎士交換節點非環圖構思，試圖自創工具以產出無標號樹，進一步將無標號樹賦予生成編碼編碼權值後解析規律並驗證之。本研究創發之$\mathbb{G}(\mathbb{S} \leftrightarrow \mathbb{P})\mathbb{E}$生成圖工具能夠擷出無標號樹、二階段編碼系統能解決圖同構問題並進一步得到權重於解析樹形，得到直線、花形、T 形與多分支樹及其通式，依此探討樹圖訊息交換之規律。

壹、　研究動機

本研究係延伸自我們在中華民國第 60 屆中小學科學展覽會提交參展的作品 Crazy Knights。相對於已完成的環圖分析，我們認為非環圖尚有擴充發展的可能。

在圖論研究中，「樹」（Tree）是無環的連通圖，是圖論中是相當重要的一支，因其結構簡單，在有標號的點容易解出，但無標號的樹就不容易有好的解答。根據過往研究，我們發現連方塊雖然能夠以二元編碼方式篩出唯一圖，但缺乏有效的生成工具及檢證異構圖工具，如何解決圖同構問題成為我們持續研究的動力。

貳、　研究目的

一、提出系統化產生樹圖的方法。

二、以樹圖說明交換關係之結構。

三、分析樹圖權重和訊息對交換和交換步數之間的關係。

參、　研究工具

一、數位與實體文書工具。

二、討論紙本記錄與閱讀筆記。

肆、　研究歷程

在圖論研究中，樹(Tree)是無環的連通圖，在圖論中是相當重要的一支。因為樹圖的結構簡單，在有標號的點容易解出，但無標號的樹就不容易有好的解答[1]。樹的解析早於 Arthur Cayley[2][3]標誌樹中心予以概念化，但對於生成唯一圖(distinct tree)我們希望能產生精簡有力的工具，於是檢視前次研究結果[4]之 39 張樹圖(騎士交換節點圖)，尋思擴大範圍產生解析無標號樹(unlabelled trees)的可能，以自創分析方法發現無標號樹產生規律，從其中解析並賦予權值後驗證。以此研究結果再檢視原 39 張樹圖刪除同構之後僅剩下 18 張唯一圖(distinct tree)。在歷屆科展作品中，我們發現尚無任何作品與該子題有關之研究，無標號樹圖從早期以生成函數驗證發展到近五年研究，多借助高階運算工具以擴增該數列為主[5][6]，其方法與結果皆與本研究有所差異。

一、名詞解釋：分為樹圖結構以及分析方法兩部分說明如下。

(一)關於「樹」本身

1. 根(Root)：任一「樹」由左至右起始自 0 開始編號，編號為 0 的點(vertex)是樹的「根」，屬性係唯一點可任意發展為任意非環連通圖。在本研究中，「樹」的任一點可揀選為「根」。
2. 主幹(Log)：是圖最長路徑，選定根發展至終點葉，依序以(−, 0, 1, 2…)呈現。
3. 葉(Leaf)：葉即度序列為 1 的點。以近圖為例 $G=(V,E)$，與葉相連的邊數為 1，標註為$deg_G(v)=1$。
4. 終點葉(End Leaf)：這裡特別指的是「主幹」終點的葉。主幹終點前若有分支則該終點為終點葉。
5. 分支(Branch)：非主幹的點，以b_i表示之。

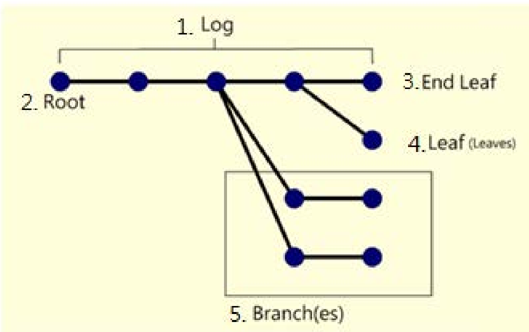


fig. 10-12

(二)分析工具之定義

1. 權重(summation of weights)：分指「點權重」和「圖權重」。前者指每個點的「晚輩」數，後者係將圖中每一點權重予以加總。
2. 對數(pairs)：指騎士對數(pairs)，亦可泛指訊息組數。

3. 交換力：指騎士對交換位置之最少步數，亦泛指訊息組數交換次數。

伍、 研究結果與討論

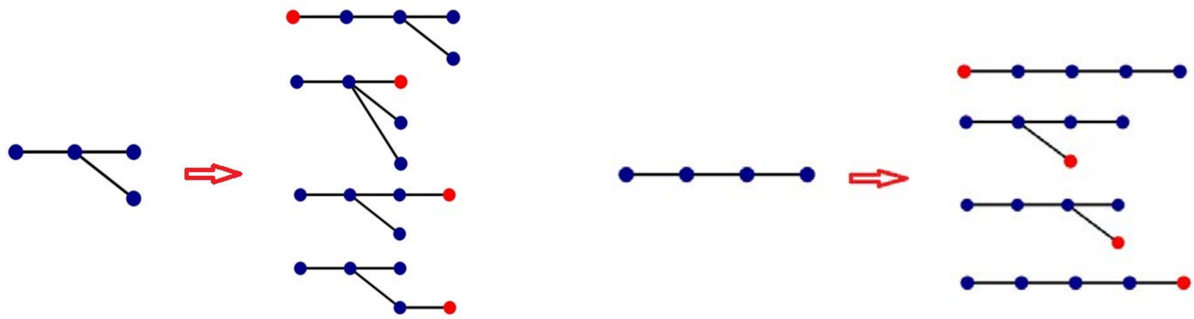
以不同的點當成根會有不同的樹序(tree ordering)，把圖的點排成$v_1, v_2, \dots, v_n$即為樹序，「根」從任一點選擇，其後的排序則由所謂的固定根衍生為主幹(Log)和分支(Branches)即成為「樹」。針對無標號樹我們試著找出篩選機制檢出唯一圖。相對於較常賦予邊權重來構圖，我們以點權重作為計算基礎，比較停駐點數量後生成樹圖並進一步分析群組交換可能。

一、針對無標號樹圖自創生成工具

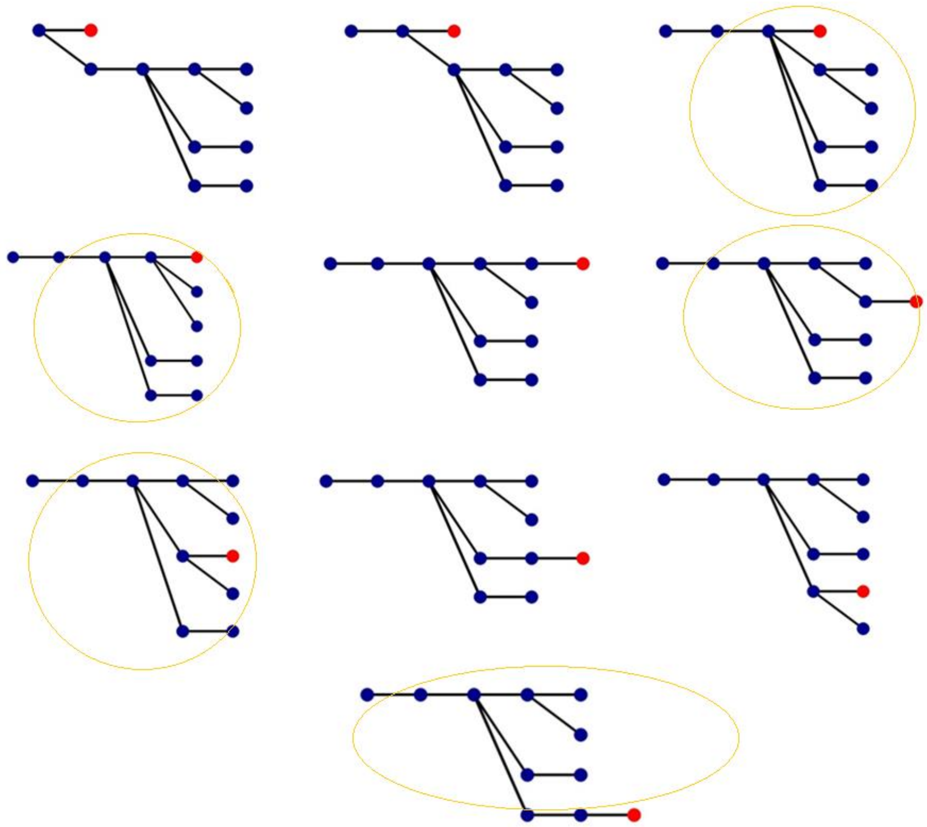
我們設計生成排序挑選編碼系統最終定版$\mathbb{G}(\mathbb{S} \leftrightarrow \mathbb{P})\mathbb{E}$。

在研究前期，所有樹圖都是依據生成(Generate)、排序(Sort)、挑選(Pick)、編碼(Encode)四種子工具產生，簡稱「GPSE」。其中，E(Encode)是研究工具的核心，可將圖形轉為數字代碼重要步驟，有助於將圖系統化、透過編碼分析產生異構圖。

(一) Generate，亦即產生新的樹圖的一個步驟。Generate 是在一個 n 個點的樹圖基礎下，在$V_0 \sim V_{n-1}$的每個點各延伸出一個點，以此產生新圖的步驟，這將會產出 N 張圖。

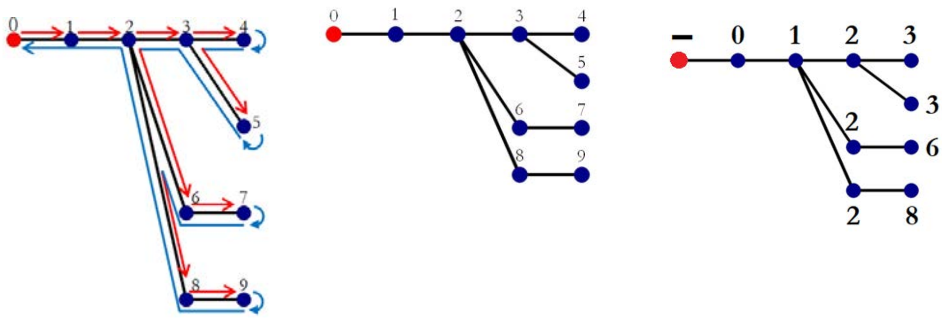


以圖 10-12 為範例，Generate 產出圖形後進一步以紅色標示延伸點(紅點)，原圖會因為紅點經過分類揀選後，得到由原樹圖延伸出的 10 張圖並析出以下 5 種異構圖(圈選)：



(二) Encode，包含標根樹的標號原則及標號樹圖的編碼規則，示例如下：

試將其根(Root)定義為 V_0，並以「摸牆走迷宮」的方式編號，再進行梳理，以 *fig*.10-12 示例編碼過程。首先將根(Root)標為 0，從根開始延伸，沿著邊開始依序標號為 1、2、3、4，在標號至葉(v_4, v_5, v_7, v_9)之後回溯，返回的過程中遇到分岔點v_2和 v_3折返往分岔點的下一個分枝繼續標上新的編號，最終回到根即標號完成。本研究每張圖皆從 0 開始標註，最後一點皆標$v-1$。



① 標號規則　　② 編號結果　　③ 以父節點編碼結果

1. 父節點編碼

標註v_0, v_1, v_2, v_3至v_9之「父點」，若任 1 點前面沒有連接其他點，則標示為「$-$」。將每個點的父點依序串接在一起，形成一個「數對」，稱之為編碼，*fig*.10-12 編碼結果為 (-, 0, 1, 2, 3, 3, 2, 6, 2, 8)且從中得到分枝(branches)。「父節點編碼」方法即以一張樹圖為基底，將樹圖中的每一個點延伸出一個新點(紅點)，形成數張新圖，進一步檢出異構圖。

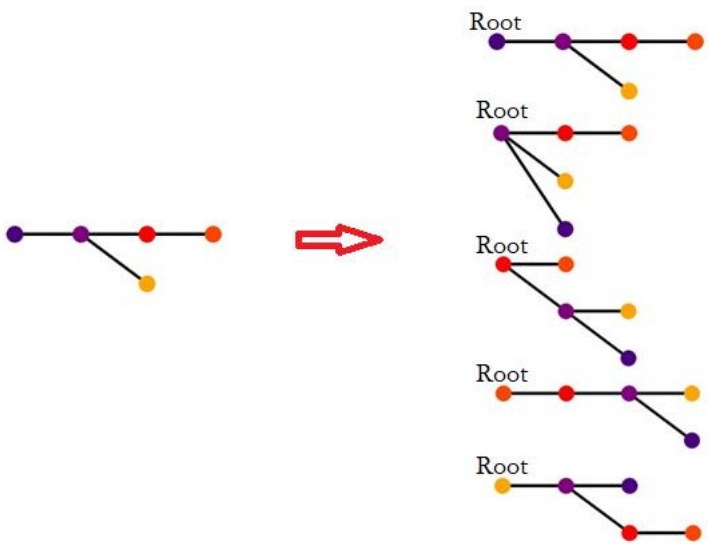
2. 編碼性質的討論

編碼是各點的父點之編號，因此可得到一個n點的圖，編碼會有n項。同時，編碼第$k+1$項(a_k)的點會與編號為k的點相連，即(k, a_k)。

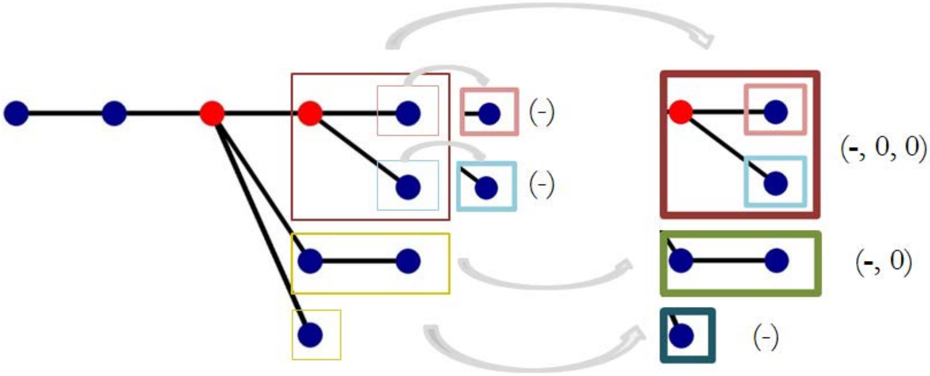
	0	1	2	3	4	5	6	7	8	9
0	0	1	0	0	0	0	0	0	0	0
1	1	0	1	0	0	0	0	0	0	0
2	0	1	0	1	0	0	1	0	1	0
3	0	0	1	0	1	1	0	0	0	0
4	0	0	0	1	0	0	0	0	0	0
5	0	0	0	1	0	0	0	0	0	0
6	0	0	1	0	0	0	0	1	0	0
7	0	0	0	0	0	0	1	0	0	0
8	0	0	1	0	0	0	0	0	0	1
9	0	0	0	0	0	0	0	0	1	0

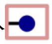
編碼推導圖的相鄰矩陣，將x軸以a_k表示，y軸以k表示。$\left(A_{a_k,k} = A_{k,a_k} = 1\right)$。$A_{x,y}$ 表示相鄰矩陣中的(x, y)。

(三) $\mathbb{S} \leftrightarrow \mathbb{P}$ ，Sort 和 Pick 是兩個整合在一起的步驟，而且在 Generate 之後每一張圖都必須經過 S↔P 步驟，試以 *fig*.4-1 再次延伸：



將每一個點作為根並畫出數張不同根的同構圖，再將這些同構圖每一張進行排序，排序的規則以 *fig*.9-10 說明之，紅色表示分岔點，並框出所有分岔點後方的子樹。



每個分岔點後方的子樹，其上下順序都必須先行排序好，並且離根較遠的分岔點必須優先排序，例如 *fig*.9-10 必須先從點 3 開始排序。標出v_3後的兩子樹各自編碼且先忽略前方所有點，接著比較兩子樹的編碼。本例由於、編碼皆為$(-)$，因此任意子樹排在上下皆可，點 3 即完成排序。

點2的情況較為複雜，同樣在解出各子樹的編碼進行比較，分別為$(-,0,0)$、$(-,0)$、$(-)$，取兩編碼從「$-$」後的第 1 位進行比較。若兩數相等則繼續往下一位比較，重複比較，直到其中有一個數「比較小」或「沒有更多數字」(若沒有更多數字則視為較小)並且不論後面的數字大小，可以依照該規律得到$(-,0,0)>(-,0)>(-)$，最後再將比較大的數字置於上方(編碼相等者上下不拘)。由此得知此圖的排序是正確的，不須調換，倘若順序不正確即需上下搬移。

1. **通例說明**

位數	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
編碼 A	-	0	1	2	3	4	4	4	2	8
關係		=	=	=	=	=	=	=	<	
編碼 B	-	0	1	2	3	4	4	4	3	1

【說明】因編碼 2 的第 9 位>編碼 1 的第 9 位，得到編碼 B>編碼 A。

2. **通例說明**

位數	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
編碼 C	-	0	1	2	3	3	3	2	7	2
關係		=	=	=	=	=				
編碼 D	-	0	1	2	3	3				

【說明】因為編碼 1 的第 7 位為 3，但編碼 2 並沒有第 7 位，得到編碼 C>編碼 D。

3. **反例比較**

位數	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	**9**	**10**
編碼 E	-	0	1	2	3	4	3	6	3	3
關係		=	=	=	=	=	<			
編碼 F	-	0	1	2	3	4	4			

【說明】因編碼 2 的第 9 位>編碼 1 的第 9 位，因此可以得到編碼 F>編碼 E，，即編碼長度不能直接決定編碼大小。

在 P 步驟產生出來的所有同構圖都進行 S 步驟後，將每一張樹圖(整張)進行 E(編碼)，再以上述原則比較該編碼並選出編碼最大的圖。在所有 G 步驟所產生的樹圖進行 P、S、E 步驟過後，最後再把所有重複(意即編碼完全相同)的樹圖刪除掉，即可得到 N+1 點的所有無標號異構樹圖。若欲求出 N+2 點的所有無標號異構樹圖，則將 N+1 點的所有無標號異構樹圖進行上述 G、S、P、E 四個步驟即可。

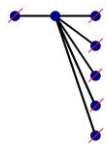
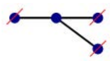
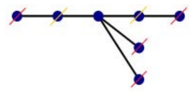
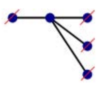
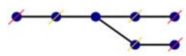
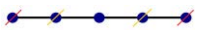
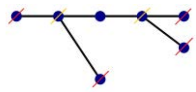
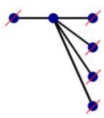
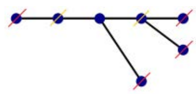
Pick 在初始的構想中亦採賦點權重 Sw 作為選擇代表圖的依據—選擇 SW 最大者SW_{max}，但 SW 的大小與排序無關，只與哪一個點作為根有關聯，所以在第一階段的初始構想是先進行 P(Pick)才進行 S(Sort)；進入研究第二階段，我們發現 Pick 在不同排序下會有不同的篩選結果，必須先進行 Sort 才能加以 Pick，所以最後統合交錯進行，成為$\mathbb{G}(\mathbb{S} \leftrightarrow \mathbb{P})\mathbb{E}$。

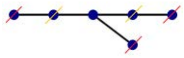
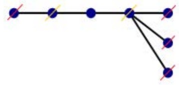
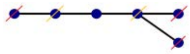
二、以「中心樹」概念分析生成樹圖結果

Cayley 於 1857 提出「樹」是一種可分析型態，他將樹分為單中心(Central)與雙中心(bicentral)，我們特別注意到圖中心長度分析(the centre or bicentre “of length”)可以協助我們區辨析出異構圖。因此我們進一步將$\mathbb{G}(\mathbb{S} \leftrightarrow \mathbb{P})\mathbb{E}$ 產生的結果以中心樹概念分類比對。

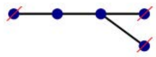
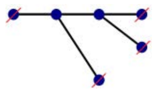
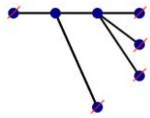
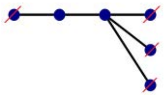
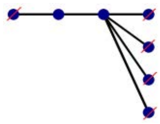
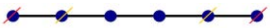
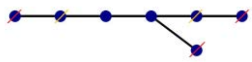
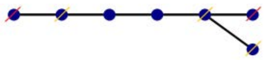
(一) $3 \leq \nu \leq 7$

1. 單中心樹(central trees)

Tree no.	Graphs & codes	Tree no.	Graphs & codes
3-1	 -, 0, 1	7-1	 -, 0, 1, 1, 1, 1, 1
4-1	 -, 0, 1, 1	7-4	 -, 0, 1, 2, 3, 2, 2
5-1	 -, 0, 1, 1, 1	7-5	 -, 0, 1, 2, 3, 2, 5
5-3	 -, 0, 1, 2, 3	7-6	 -, 0, 1, 2, 3, 3, 1
6-1	 -, 0, 1, 1, 1, 1	7-7	 -, 0, 1, 2, 3, 3, 2

6-4	 -, 0, 1, 2, 3, 2	7-8	 -, 0, 1, 2, 3, 3, 3
6-5	 -, 0, 1, 2, 3, 3	7-11	 -, 0, 1, 2, 3, 4, 5

2. 雙中心樹(bicentral trees)

Tree no.	Bicentre	Tree no.	Bicentre
4-2	 -, 0, 1, 2	**5-2**	 -, 0, 1, 2, 2
6-2	 -, 0, 1, 2, 2, 1	**7-2**	 -, 0, 1, 2, 2, 2, 1
6-3	 -, 0, 1, 2, 2, 2	**7-3**	 -, 0, 1, 2, 2, 2, 2
6-6	 -, 0, 1, 2, 3, 4	**7-9**	 -, 0, 1, 2, 3, 4, 3
7-10	 -, 0, 1, 2, 3, 4, 4		

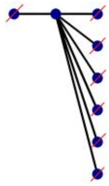
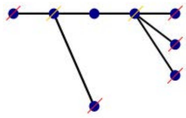
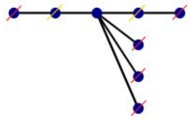
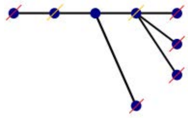
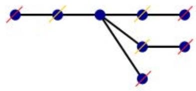
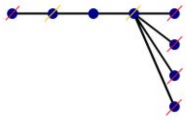
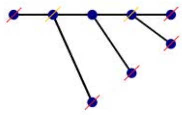
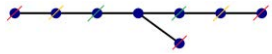
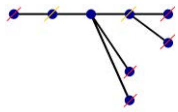
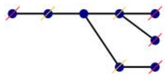
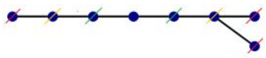
【結果】

樹節點數	**3**	**4**	**5**	**6**	**7**
單中心樹	1	1	2	3	7
雙中心樹	0	1	1	3	4
生成樹量	1	2	3	6	11

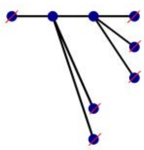
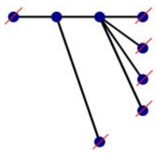
【結論】$\mathbb{G}(\mathbb{S} \leftrightarrow \mathbb{P})\mathbb{E}$產生 3 點樹至 7 點樹異構圖數量與 Cayley 中心樹結果一致。

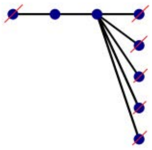
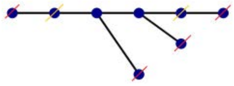
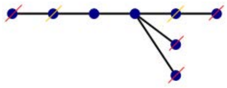
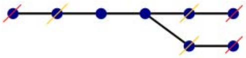
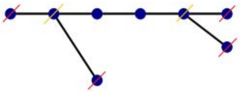
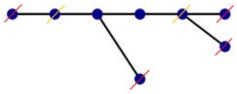
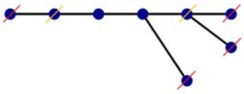
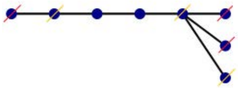
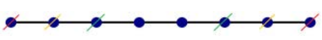
(二) ν = 8

1. 單中心樹(central trees)

Tree no.	Graphs & codes	Tree no.	Graphs & codes
8-1	 -, 0, 1, 1, 1, 1, 1, 1	8-10	 -, 0, 1, 2, 3, 3, 3, 1
8-5	 -, 0, 1, 2, 3, 2, 2, 2	8-11	 -, 0, 1, 2, 3, 3, 3, 2
8-6	 -, 0, 1, 2, 3, 2, 5, 2	8-12	 -, 0, 1, 2, 3, 3, 3, 3
8-7	 -, 0, 1, 2, 3, 3, 2, 1	8-20	 -, 0, 1, 2, 3, 4, 5, 3
8-8	 -, 0, 1, 2, 3, 3, 2, 2	8-21	 -, 0, 1, 2, 3, 4, 5, 4
8-9	 -, 0, 1, 2, 3, 3, 2, 6	8-22	 -, 0, 1, 2, 3, 4, 5, 5

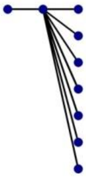
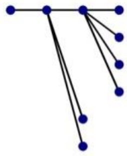
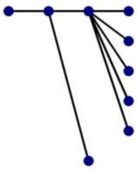
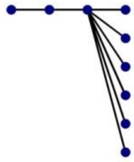
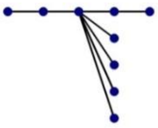
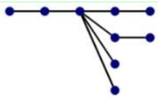
2. 雙中心樹(bicentral trees)

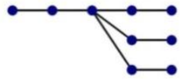
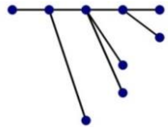
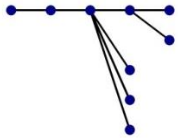
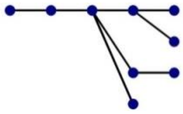
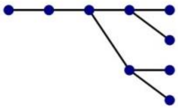
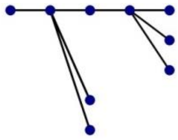
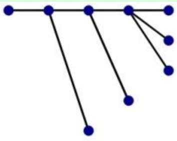
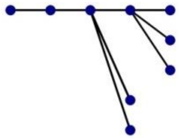
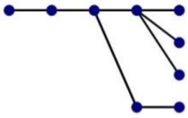
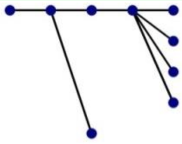
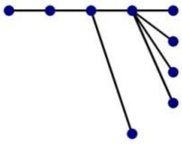
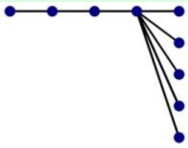
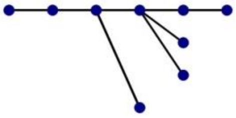
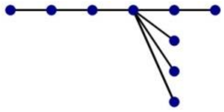
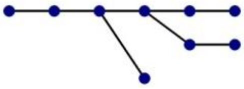
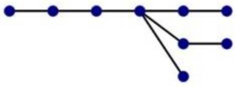
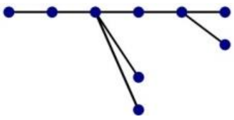
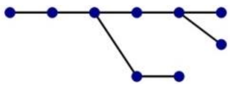
Tree no.	Bicentre	Tree no.	Bicentre
8-2	 -, 0, 1, 2, 2, 2, 1, 1	8-3	 -, 0, 1, 2, 2, 2, 2, 1

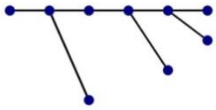
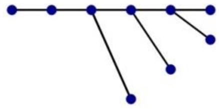
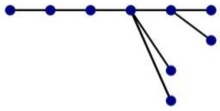
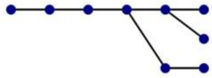
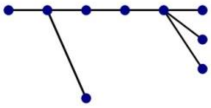
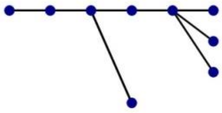
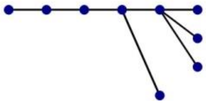
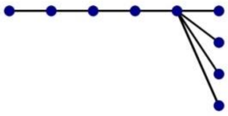
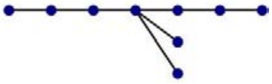
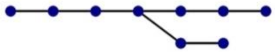
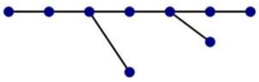
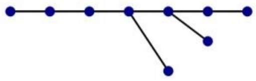
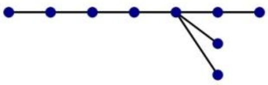
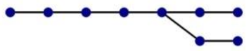
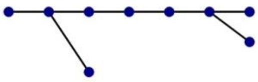
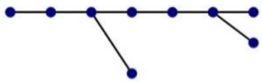
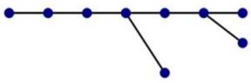
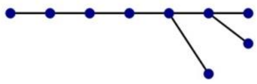
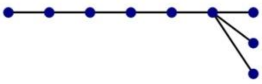
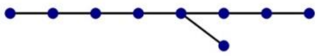
Tree no.	Bicentre	Tree no.	Bicentre
8-4	 -, 0, 1, 2, 2, 2, 2, 2	8-13	 -, 0, 1, 2, 3, 4, 3, 2
8-14	 -, 0, 1, 2, 3, 4, 3, 3	8-15	 -, 0, 1, 2, 3, 4, 3, 6
8-16	 -, 0, 1, 2, 3, 4, 4, 1	8-17	 -, 0, 1, 2, 3, 4, 4, 2
8-18	 -, 0, 1, 2, 3, 4, 4, 3	8-19	 -, 0, 1, 2, 3, 4, 4, 4
8-23		-, 0, 1, 2, 3, 4, 5, 6	

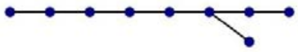
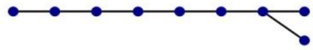
【結果】 8 點圖可析出 12 個單中心樹和 11 個雙中心樹，合計 23 中心樹，
亦與 Cayley 中心樹概念一致，結果相符。

(三) ν = 9 (雙中心、單中心分別以 Bi-, Uni-標註於編號下方)

Tree no.	Graphs & codes	Tree no.	Graphs & codes
9-1 **Uni-**	 -, 0, 1, 1, 1, 1, 1, 1, 1	9-2 Bi-	 -, 0, 1, 2, 2, 2, 2, 1, 1
9-3 Bi-	 -, 0, 1, 2, 2, 2, 2, 2, 1	9-4 Bi-	 -, 0, 1, 2, 2, 2, 2, 2, 2
9-5 Uni-	 -, 0, 1, 2, 3, 2, 2, 2, 2	9-6 Uni-	 -, 0, 1, 2, 3, 2, 5, 2, 2

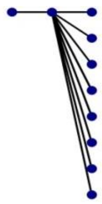
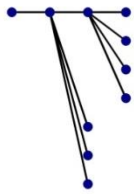
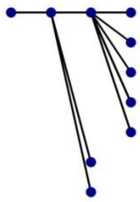
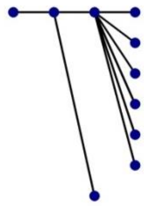
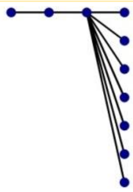
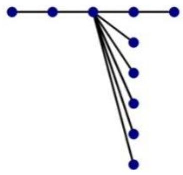
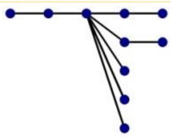
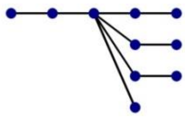
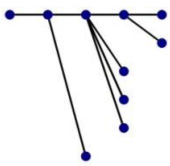
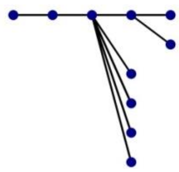
Tree no.	Graphs & codes	Tree no.	Graphs & codes
9-7 Uni-	 -, 0, 1, 2, 3, 2, 5, 2, 7	9-8 Uni-	 -, 0, 1, 2, 3, 3, 2, 2, 1
9-9 Uni-	 -, 0, 1, 2, 3, 3, 2, 2, 2	9-10 Uni-	 -, 0, 1, 2, 3, 3, 2, 6, 2
9-11 Uni-	 -, 0, 1, 2, 3, 3, 2, 6, 6	9-12 Uni-	 -, 0, 1, 2, 3, 3, 3, 1, 1
9-13 Uni-	 -, 0, 1, 2, 3, 3, 3, 2, 1	9-14 Uni-	 -, 0, 1, 2, 3, 3, 3, 2, 2
9-15 Uni-	 -, 0, 1, 2, 3, 3, 3, 2 ,7	9-16 Uni-	 -, 0, 1, 2, 3, 3, 3, 3, 1
9-17 Uni-	 -, 0, 1, 2, 3, 3, 3, 3, 2	9-18 Uni-	 -, 0, 1, 2, 3, 3, 3, 3, 3
9-19 Bi-	 -, 0, 1, 2, 3, 4, 3, 3, 2	9-20 Bi-	 -, 0, 1, 2, 3, 4, 3, 3, 3
9-21 Bi-	 -, 0, 1, 2, 3, 4, 3, 6, 2	9-22 Bi-	 -, 0, 1, 2, 3, 4, 3, 6, 3
9-23 Bi-	 -, 0, 1, 2, 3, 4, 4, 2, 2	9-24 Bi-	 -, 0, 1, 2, 3, 4, 4, 2, 7

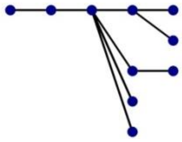
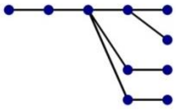
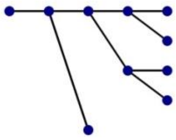
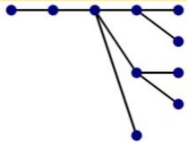
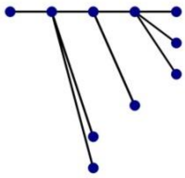
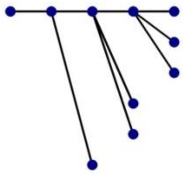
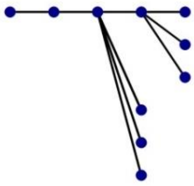
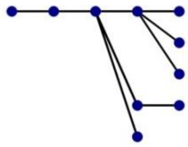
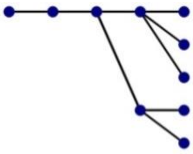
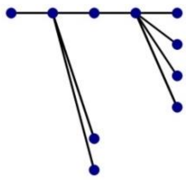
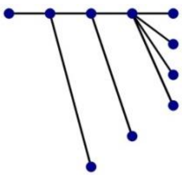
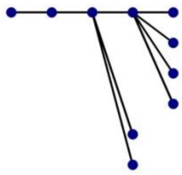
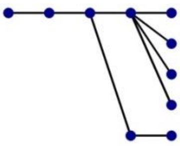
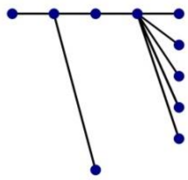
Tree no.	Graphs & codes	Tree no.	Graphs & codes
9-25 Bi-	 -, 0, 1, 2, 3, 4, 4, 3, 1	9-26 Bi-	 -, 0, 1, 2, 3, 4, 4, 3, 2
9-27 Bi-	 -, 0, 1, 2, 3, 4, 4, 3, 3	9-28 Bi-	 -, 0, 1, 2, 3, 4, 4, 3, 7
9-29 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 1	9-30 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 2
9-31 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 3	9-32 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 4
9-33 Uni-	 -, 0, 1, 2, 3, 4, 5, 3, 3	9-34 Uni-	 -, 0, 1, 2, 3, 4, 5, 3, 7
9-35 Uni-	 -, 0, 1, 2, 3, 4, 5, 4, 2	9-36 Uni-	 -, 0, 1, 2, 3, 4, 5, 4, 3
9-37 Uni-	 -, 0, 1, 2, 3, 4, 5, 4, 4	9-38 Uni-	 -, 0, 1, 2, 3, 4, 5, 4, 7
9-39 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 1	9-40 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 2
9-41 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 3	9-42 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 4
9-43 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 5	9-44 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 4

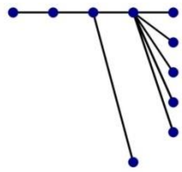
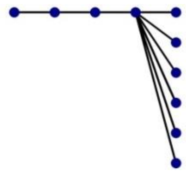
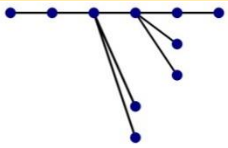
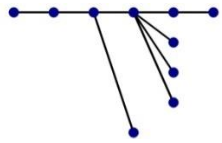
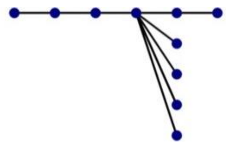
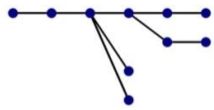
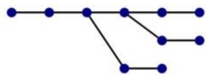
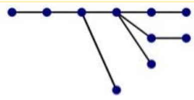
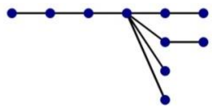
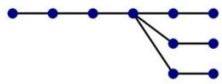
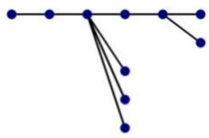
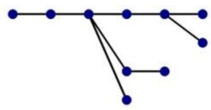
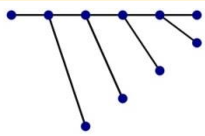
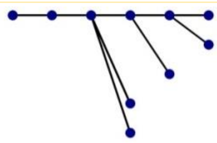
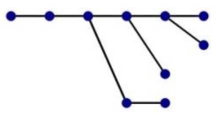
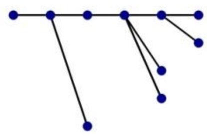
Tree no.	Graphs & codes	Tree no.	Graphs & codes
9-45 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 5	9-46 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 6
9-47 Uni-			-, 0, 1, 2, 3, 4, 5, 6, 7

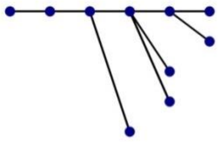
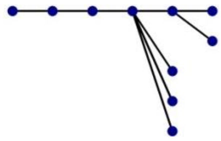
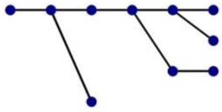
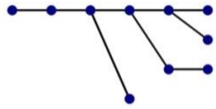
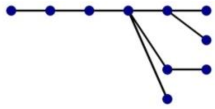
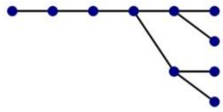
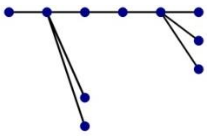
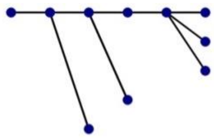
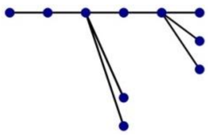
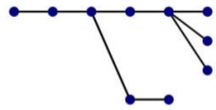
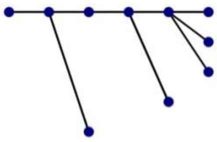
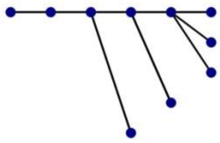
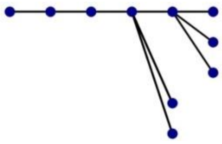
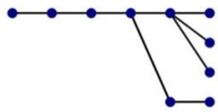
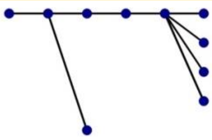
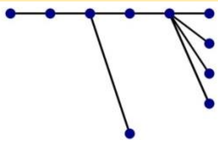
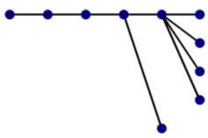
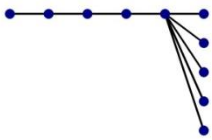
【結果】 9 點樹圖可析出 27 個單中心樹和 20 個雙中心樹，合計 47 中心樹，結果與 Cayley 中心樹概念相符。

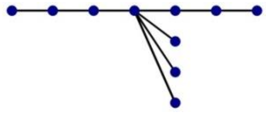
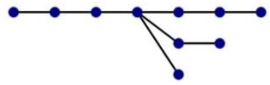
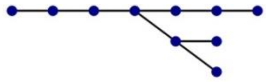
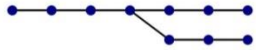
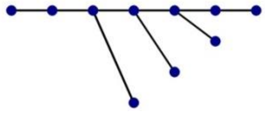
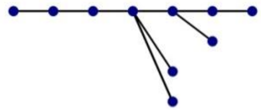
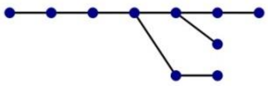
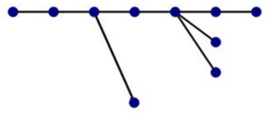
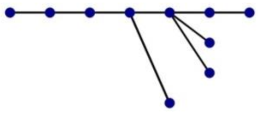
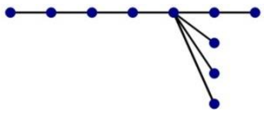
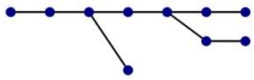
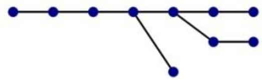
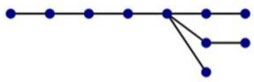
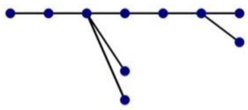
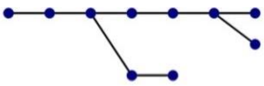
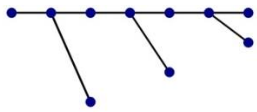
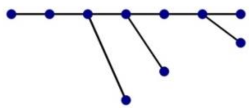
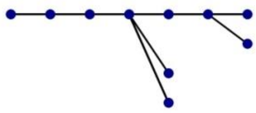
(四)ν = 10　　(雙中心、單中心分別以 Bi-, Uni-標註於編號下方)

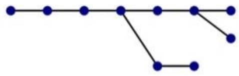
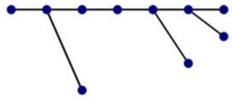
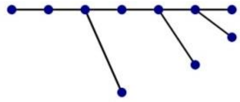
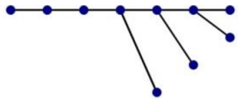
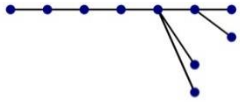
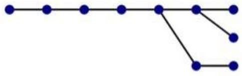
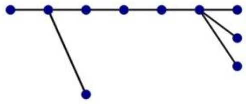
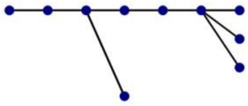
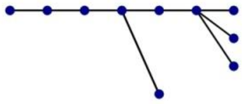
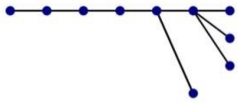
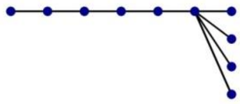
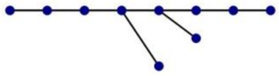
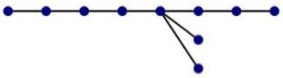
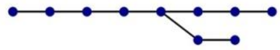
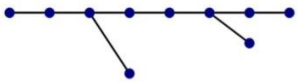
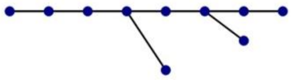
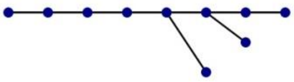
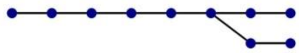
No.	Trees	No.	Trees
10-1 Uni-	 -, 0, 1, 1, 1, 1, 1, 1, 1, 1	10-2 Bi-	 -, 0, 1, 2, 2, 2, 2, 1, 1, 1
10-3 Bi-	 -, 0, 1, 2, 2, 2, 2, 2, 1, 1	10-4 Bi-	 -, 0, 1, 2, 2, 2, 2, 2, 2, 1
10-5 Bi-	 -, 0, 1, 2, 2, 2, 2, 2, 2, 2	10-6 Uni-	 -, 0, 1, 2, 3, 2, 2, 2, 2, 2
10-7 Uni-	 -, 0, 1, 2, 3, 2, 5, 2, 2, 2	10-8 Uni-	 -, 0, 1, 2, 3, 2, 5, 2, 7, 2
10-9 Uni-	 -, 0, 1, 2, 3, 3, 2, 2, 2, 1	10-10 Uni-	 -, 0, 1, 2, 3, 3, 2, 2, 2, 2

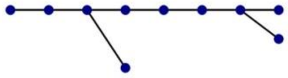
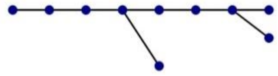
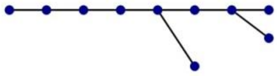
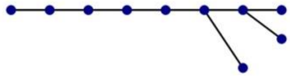
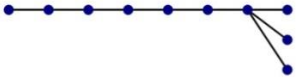
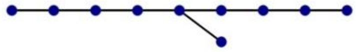
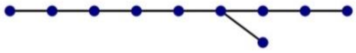
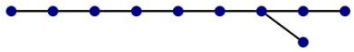
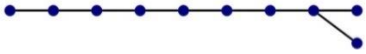
No.	**Trees**	No.	**Trees**
10-11 Uni-	 -, 0, 1, 2, 3, 3, 2, 6, 2, 2	10-12 Uni-	 -, 0, 1, 2, 3, 3, 2, 6, 2, 8
10-13 Uni-	 -, 0, 1, 2, 3, 3, 2, 6, 6, 1	10-14 Uni-	 -, 0, 1, 2, 3, 3, 2, 6, 6, 2
10-15 Uni-	 -, 0, 1, 2, 3, 3, 3, 2, 1, 1	10-16 Uni-	 -, 0, 1, 2, 3, 3, 3, 2, 2, 1
10-17 Uni-	 -, 0, 1, 2, 3, 3, 3, 2, 2, 2	10-18 Uni-	 -, 0, 1, 2, 3, 3, 3, 2, 7, 2
10-19 Uni-	 -, 0, 1, 2, 3, 3, 3, 2, 7, 7	10-20 Uni-	 -, 0, 1, 2, 3, 3, 3, 3, 1, 1
10-21 Uni-	 -, 0, 1, 2, 3, 3, 3, 3, 2, 1	10-22 Uni-	 -, 0, 1, 2, 3, 3, 3, 3, 2, 2
10-23 Uni-	 -, 0, 1, 2, 3, 3, 3, 3, 2, 8	10-24 Uni-	 -, 0, 1, 2, 3, 3, 3, 3, 3, 1

No.	Trees	No.	Trees
10-25 Uni-	 -, 0, 1, 2, 3, 3, 3, 3, 3, 2	10-26 Uni-	 -, 0, 1, 2, 3, 3, 3, 3, 3, 3
10-27 Bi-	 -, 0, 1, 2, 3, 4, 3, 3, 2, 2	10-28 Bi-	 -, 0, 1, 2, 3, 4, 3, 3, 3, 2
10-29 Bi-	 -, 0, 1, 2, 3, 4, 3, 3, 3, 3	10-30 Bi-	 -, 0, 1, 2, 3, 4, 3, 6, 2, 2
10-31 Bi-	 -, 0, 1, 2, 3, 4, 3, 6, 2, 8	10-32 Bi-	 -, 0, 1, 2, 3, 4, 3, 6, 3, 2
10-33 Bi-	 -, 0, 1, 2, 3, 4, 3, 6, 3, 3	10-34 Bi-	 -, 0, 1, 2, 3, 4, 3, 6, 3, 8
10-35 Bi-	 -, 0, 1, 2, 3, 4, 4, 2, 2, 2	10-36 Bi-	 -, 0, 1, 2, 3, 4, 4, 2, 7, 2
10-37 Bi-	 -, 0, 1, 2, 3, 4, 4, 3, 2, 1	10-38 Bi-	 -, 0, 1, 2, 3, 4, 4, 3, 2, 2
10-39 Bi-	 -, 0, 1, 2, 3, 4, 4, 3, 2, 8	10-40 Bi-	 -, 0, 1, 2, 3, 4, 4, 3, 3, 1

No.	Trees	No.	Trees
10-41 Bi-	 -, 0, 1, 2, 3, 4, 4, 3, 3, 2	10-42 Bi-	 -, 0, 1, 2, 3, 4, 4, 3, 3, 3
10-43 Bi-	 -, 0, 1, 2, 3, 4, 4, 3, 7, 1	10-44 Bi-	 -, 0, 1, 2, 3, 4, 4, 3, 7, 2
10-45 Bi-	 -, 0, 1, 2, 3, 4, 4, 3, 7, 3	10-46 Bi-	 -, 0, 1, 2, 3, 4, 4, 3, 7, 7
10-47 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 1, 1	10-48 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 2, 1
10-49 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 2, 2	10-50 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 2, 8
10-51 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 3, 1	10-52 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 3, 2
10-53 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 3, 3	10-54 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 3, 8
10-55 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 4, 1	10-56 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 4, 2
10-57 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 4, 3	10-58 Bi-	 -, 0, 1, 2, 3, 4, 4, 4, 4, 4

No.	**Trees**	No.	**Trees**
10-59 Uni-	 -, 0, 1, 2, 3, 4, 5, 3, 3, 3	10-60 Uni-	 -, 0, 1, 2, 3, 4, 5, 3, 7, 3
10-61 Uni-	 -, 0, 1, 2, 3, 4, 5, 3, 7, 7	10-62 Uni-	 -, 0, 1, 2, 3, 4, 5, 3, 7, 8
10-63 Uni-	 -, 0, 1, 2, 3, 4, 5, 4, 3, 2	10-64 Uni-	 -, 0, 1, 2, 3, 4, 5, 4, 3, 3
10-65 Uni-	 -, 0, 1, 2, 3, 4, 5, 4, 3, 8	10-66 Uni-	 -, 0, 1, 2, 3, 4, 5, 4, 4, 2
10-67 Uni-	 -, 0, 1, 2, 3, 4, 5, 4, 4, 3	10-68 Uni-	 -, 0, 1, 2, 3, 4, 5, 4, 4, 4
10-69 Uni-	 -, 0, 1, 2, 3, 4, 5, 4, 7, 2	10-70 Uni-	 -, 0, 1, 2, 3, 4, 5, 4, 7, 3
10-71 Uni-	 -, 0, 1, 2, 3, 4, 5, 4, 7, 4	10-72 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 2, 2
10-73 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 2, 8	10-74 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 3, 1
10-75 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 3, 2	10-76 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 3, 3

No.	**Trees**	No.	**Trees**
10-77 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 3, 8	10-78 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 4, 1
10-79 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 4, 2	10-80 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 4, 3
10-81 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 4, 4	10-82 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 4, 8
10-83 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 5, 1	10-84 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 5, 2
10-85 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 5, 3	10-86 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 5, 4
10-87 Uni-	 -, 0, 1, 2, 3, 4, 5, 5, 5, 5	10-88 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 4, 3
10-89 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 4, 4	10-90 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 4, 8
10-91 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 5, 2	10-92 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 5, 3
10-93 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 5, 4	10-94 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 5, 5
10-95 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 5, 8	10-96 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 6, 1

No.	Trees	No.	Trees
10-97 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 6, 2	10-98 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 6, 3
10-99 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 6, 4	10-100 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 6, 5
10-101 Bi-	 -, 0, 1, 2, 3, 4, 5, 6, 6, 6	10-102 Uni-	 -, 0, 1, 2, 3, 4, 5, 6, 7, 4
10-103 Uni-	 -, 0, 1, 2, 3, 4, 5, 6, 7, 5	10-104 Uni-	 -, 0, 1, 2, 3, 4, 5, 6, 7, 6
10-105 Uni-	 -, 0, 1, 2, 3, 4, 5, 6, 7, 7	10-106 Bi-	-, 0, 1, 2, 3, 4, 5, 6, 7, 8

【結果】10 點樹圖可析出 55 個單中心樹和 51 個雙中心樹，合計 106 中心樹，生成結果與 Cayley 中心樹概念析出結果一致。

【結論】本研究以$\mathbb{G}(\mathbb{S} \leftrightarrow \mathbb{P})\mathbb{E}$ 產生的樹圖以中心樹概念驗證比對結果如下表

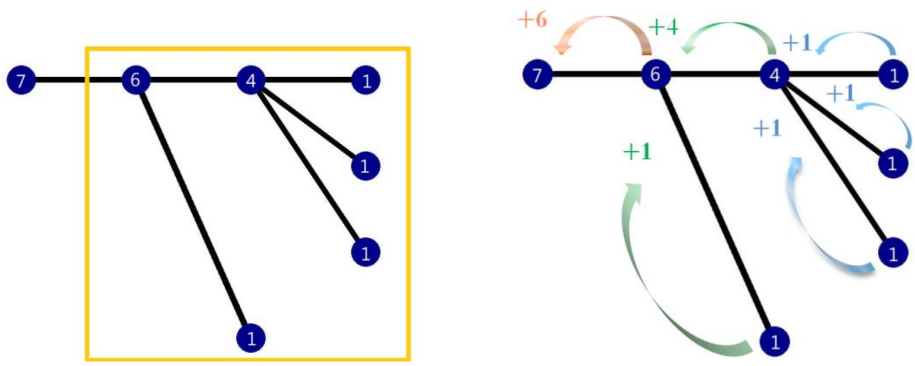
節點數	3	4	5	6	7	8	9	10
單中心	1	1	2	3	7	12	27	55
雙中心	0	1	1	3	4	11	20	51
合計	1	2	3	6	11	23	47	106

我們試以 Cayley 中心樹方法分析本研究主幹特性，發現奇偶性可以得到，刪除葉的次數(以$\mathcal{k}$表示)，得到主幹長度決定中心數型態。

主幹	單中心	雙中心
點數 vertices	$2\mathcal{k}+1$ 奇數	$2\mathcal{k}+2$ 偶數
邊數 edges	$2\mathcal{k}+2$ 偶數	$2\mathcal{k}+1$ 奇數

三、權重用於最大交換對數和最小交換步數之關聯

根據$\mathbb{G}(\mathbb{S} \leftrightarrow \mathbb{P})\mathbb{E}$生成樹圖結果，我們除了以中心樹方法檢核，再進一步將圖中各點賦予權重(weighted graph)，觀察比較各圖在交換效益上的規律。以 *fig.*7-2 為例。



$$\mathrm{SW}=\sum_{n=1}^{V} d(v_n)\text{，}d(v)\text{為點}v\text{的晚輩數}$$

*fig.*7-2 編碼為 -, 0, 1, 2, 2, 2, 1，設葉的權重為 1，由葉的權重依序回推至根，得到各點的權重$d(v_0), d(v_1), d(v_2), d(v_3), d(v_4), d(v_5), d(v_6)$，推導點權重值[7, 6, 4, 1, 1, 1, 1]，權重總和 21。

（一）生成樹圖 $3 \le v \le 5$

圖編號 Tree No.	父節點編碼 Code	賦點權重 Weights of vertices	權重總和 Summation	對數 pairs	交換能力 Moves
3-1	-, 0, 1	[3, 2, 1]	6	0	N/A
4-1	-, 0, 1, 1	[4, 3, 1, 1]	9	1	6
4-2	-, 0, 1, 2	[4, 3, 2, 1]	10	0	N/A
5-1	-, 0, 1, 1, 1	[5, 4, 1, 1, 1]	12	1	6
5-2	-, 0, 1, 2, 2	[5, 4, 3, 1, 1]	14	1	6
5-3	-, 0, 1, 2, 3	[5, 4, 3, 2, 1]	15	0	N/A

【結果】*fig.* 3-1, 4-2, 5-3 皆為一直線。*fig.* 5-1 僅一個分叉點，故呈現花狀。
fig. 5-2 呈現 T 型。

（二）生成樹圖組 $6 \le v \le 8$

圖編號 Tree no.	父節點編碼 Code	賦點權重 Weights of vertices	權重總和 Summation	對數 pairs	交換能力 Moves
6-1	-, 0, 1, 1, 1, 1	[6, 5, 1, 1, 1, 1]	15	2	10
6-2	-, 0, 1, 2, 2 ,1	[6, 5, 3, 1, 1, 1]	17	2	12
6-3	-, 0, 1, 2, 2, 2	[6, 5, 4, 1, 1, 1]	18	1	6
6-4	-, 0, 1, 2, 3, 2	[6, 5, 4, 2, 1, 1]	19	1	6
6-5	-, 0, 1, 2, 3, 3	[6, 5, 4, 3, 1, 1]	20	1	6
6-6	-, 0, 1, 2, 3, 4	[6, 5, 4, 3, 2, 1]	21	0	N/A
7-1	-, 0, 1, 1, 1, 1, 1	[7, 6, 1, 1, 1, 1, 1]	18	2	10
7-2	-, 0, 1, 2, 2, 2, 1	[7, 6, 4, 1, 1, 1, 1]	21	2	12
7-3	-, 0, 1, 2, 2, 2, 2	[7, 6, 5, 1, 1, 1, 1]	22	2	10
7-4	-, 0, 1, 2, 3, 2, 2	[7, 6, 5, 2, 1, 1, 1]	23	2	16
7-5	-, 0, 1, 2, 3, 2, 5	[7, 6, 5, 2, 1, 2, 1]	24	2	18
7-6	-, 0, 1, 2, 3, 3, 1	[7, 6, 4, 3, 1, 1, 1]	23	2	12

圖編號 Tree no.	父節點編碼 Code	賦點權重 Weights of vertices	權重總和 Summation	對數 pairs	交換能力 Moves
7-7	-, 0, 1, 2, 3, 3, 2	[7, 6, 5, 3, 1, 1, 1]	24	2	12
7-8	-, 0, 1, 2, 3, 3, 3	[7, 6, 5, 4, 1, 1, 1]	25	1	6
7-9	-, 0, 1, 2, 3, 4, 3	[7, 6, 5, 4, 2, 1, 1]	26	1	6
7-10	-, 0, 1, 2, 3, 4, 4	[7, 6, 5, 4, 3, 1, 1]	27	1	6
7-11	-, 0, 1, 2, 3, 4, 5	[7, 6, 5, 4, 3, 2, 1]	28	0	N/A

【結果】*fig.* 6-6, 7-11 皆為一直線；*fig.* 6-1, 7-1 僅一個分叉點，故呈現花狀；*fig.* 6-4, 6-5, 7-9, 7-10 呈現 T 型；其餘皆為多分枝圖。

(三)生成樹圖 $\nu = 8$

圖編號 Tree no.	父節點編碼 Code	賦點權重 Weights of vertices	權重 Summation	對數 pairs	交換能力 Moves
8-1	-, 0, 1, 1, 1, 1, 1, 1	[8, 7, 1, 1, 1, 1, 1, 1]	21	3	14
8-2	-, 0, 1, 1, 2, 2, 2, 1	[8, 7, 4, 1, 1, 1, 1, 1]	24	2	12
8-3	-, 0, 1, 2, 2, 2 ,2, 1	[8, 7, 5, 1, 1, 1, 1, 1]	25	2	10
8-4	-, 0, 1, 2, 2, 2, 2 ,2	[8, 7, 6, 1, 1, 1, 1, 1]	26	2	10
8-5	-, 0, 1, 2, 3, 2, 2, 2	[8, 7, 6, 2, 1, 1, 1, 1]	27	2	10
8-6	-, 0, 1, 2, 3, 2, 5, 2	[8, 7, 6, 2, 1, 2, 1, 1]	28	2	16
8-7	-, 0, 1, 2, 3, 3, 2, 1	[8, 7, 5, 3, 1, 1, 1, 1]	27	3	18
8-8	-, 0, 1, 2, 3, 3, 2, 2	[8, 7, 6, 3, 1, 1, 1, 1]	28	2	12
8-9	-, 0, 1, 2, 3, 3, 2, 6	[8, 7, 6, 3, 1, 1, 2, 1]	29	2	12
8-10	-, 0, 1, 2, 3, 3, 3, 1	[8, 7, 5, 4, 1, 1, 1, 1]	28	2	12
8-11	-, 0, 1, 2, 3, 3, 3, 2	[8, 7, 6, 4, 1, 1, 1, 1]	29	2	12
8-12	-, 0, 1, 2, 3, 3, 3, 3	[8, 7, 6, 5, 1, 1, 1, 1]	30	2	16
8-13	-, 0, 1, 2, 3, 4, 3, 2	[8, 7, 6, 4, 2, 1, 1, 1]	30	2	12
8-14	-, 0, 1, 2, 3, 4, 3, 3	[8, 7, 6, 5, 2, 1, 1, 1]	31	2	16
8-15	-, 0, 1, 2, 3, 4, 3, 6	[8, 7, 6, 5, 2, 1, 2, 1]	32	2	18
8-16	-, 0, 1, 2, 3, 4, 4, 1	[8, 7, 5, 4, 3, 1, 1, 1]	30	2	12
8-17	-, 0, 1, 2, 3, 4, 4, 2	[8, 7, 6, 4, 3, 1, 1, 1]	31	2	12
8-18	-, 0, 1, 2, 3, 4, 4, 3	[8, 7, 6, 5, 3, 1, 1, 1]	32	2	12
8-19	-, 0, 1, 2, 3, 4, 4, 4	[8, 7, 6, 5, 4, 1, 1, 1]	33	1	6
8-20	-, 0, 1, 2, 3, 4, 5, 3	[8, 7, 6, 5, 3, 2, 1, 1]	33	2	26
8-21	-, 0, 1, 2, 3, 4, 5, 4	[8, 7, 6, 5, 4, 2, 1, 1]	34	1	6
8-22	-, 0, 1, 2, 3, 4, 5, 5	[8, 7, 6, 5, 4, 3, 1, 1]	35	1	6
8-23	-, 0, 1, 2, 3, 4, 5, 6	[8, 7, 6, 5, 4, 3, 2, 1]	36	0	N/A

【結果】*fig.* 8-23 皆為一直線；*fig.* 8-1 僅一個分叉點，故呈現花狀；*fig.* 8-20, 8-21, 8-22 呈現 T 型；其餘皆為多分枝圖。

(四)生成樹圖 $\nu = 9$

圖編號 Tree no.	父節點編碼 Code	賦點權重 Weights of vertices	權重總和 Summation	對數 pairs	交換能力 Moves
9-1	-, 0, 1, 1, 1, 1, 1, 1, 1	[9, 8, 1, 1, 1, 1, 1, 1, 1]	24	3	14
9-2	-, 0, 1, 2, 2, 2, 2, 1, 1	[9, 8, 5, 1, 1, 1, 1, 1, 1]	28	3	16
9-3	-, 0, 1, 2, 2, 2, 2, 2, 1	[9, 8, 6, 1, 1, 1, 1, 1, 1]	29	3	16
9-4	-, 0, 1, 2, 2, 2 ,2, 2, 2	[9, 8, 7, 1, 1, 1, 1, 1, 1]	30	3	14
9-5	-, 0, 1, 2, 3, 2, 2, 2, 2	[9, 8, 7, 2, 1, 1, 1, 1, 1]	31	3	22

圖編號 Tree no.	父節點編碼 Code	賦點權重 Weights of vertices	權重總和 Summation	對數 pairs	交換能力 Moves
9-6	-, 0, 1, 2, 3, 2, 5, 2, 2	[9, 8, 7, 2, 1, 2, 1, 1, 1]	32	3	22
9-7	-, 0, 1, 2, 3, 2, 5, 2, 7	[9, 8, 7, 2, 1, 2, 1, 2, 1]	33	3	28
9-8	-, 0, 1, 2, 3, 3, 2, 2, 1	[9, 8, 6, 3, 1, 1, 1, 1, 1]	31	3	20
9-9	-, 0, 1, 2, 3, 3, 2, 2, 2	[9, 8, 7, 3, 1, 1, 1, 1, 1]	32	3	20
9-10	-, 0, 1, 2, 3, 3, 2, 6, 2	[9, 8, 7, 3, 1, 1, 2, 1, 1]	33	3	22
9-11	-, 0, 1, 2, 3, 3, 2, 6, 6	[9, 8, 7, 3, 1, 1, 3, 1, 1]	34	3	22
9-12	-, 0, 1, 2, 3, 3, 3, 1, 1	[9, 8, 5, 4, 1, 1, 1, 1, 1]	31	2	12
9-13	-, 0, 1, 2, 3, 3, 3, 2, 1	[9, 8, 6, 4, 1, 1, 1, 1, 1]	32	3	20
9-14	-, 0, 1, 2, 3, 3, 3, 2, 2	[9, 8, 7, 4, 1, 1, 1, 1, 1]	33	3	22
9-15	-, 0, 1, 2, 3, 3, 3, 2 ,7	[9, 8, 7, 4, 1, 1, 1, 2, 1]	34	3	24
9-16	-, 0, 1, 2, 3, 3, 3, 3, 1	[9, 8, 6, 5, 1, 1, 1, 1, 1]	33	3	16
9-17	-, 0, 1, 2, 3, 3, 3, 3, 2	[9, 8, 7, 5, 1, 1, 1, 1, 1]	34	3	16
9-18	-, 0, 1, 2, 3, 3, 3, 3, 3	[9, 8, 7, 6, 1, 1, 1, 1, 1]	35	2	10
9-19	-, 0, 1, 2, 3, 4, 3, 3, 2	[9, 8, 7, 5, 2, 1, 1, 1, 1]	35	3	24
9-20	-, 0, 1, 2, 3, 4, 3, 3, 3	[9, 8, 7, 6, 2, 1, 1, 1, 1]	36	2	16
9-21	-, 0, 1, 2, 3, 4, 3, 6, 2	[9, 8, 7, 5, 2, 1, 2, 1, 1]	36	2	12
9-22	-, 0, 1, 2, 3, 4, 3, 6, 3	[9, 8, 7, 6, 2, 1, 2, 1, 1]	37	2	16
9-23	-, 0, 1, 2, 3, 4, 4, 2, 2	[9, 8, 7, 4, 3, 1, 1, 1, 1]	35	3	24
9-24	-, 0, 1, 2, 3, 4, 4, 2, 7	[9, 8, 7, 4, 3, 1, 1, 2, 1]	36	2	12
9-25	-, 0, 1, 2, 3, 4, 4, 3, 1	[9, 8, 6, 5, 3, 1, 1, 1, 1]	35	3	18
9-26	-, 0, 1, 2, 3, 4, 4, 3, 2	[9, 8, 7, 5, 3, 1, 1, 1, 1]	36	3	18
9-27	-, 0, 1, 2, 3, 4, 4, 3, 3	[9, 8, 7, 6, 3, 1, 1, 1, 1]	37	2	12
9-28	-, 0, 1, 2, 3, 4, 4, 3, 7	[9, 8, 7, 6, 3, 1, 1, 2, 1]	38	2	12
9-29	-, 0, 1, 2, 3, 4, 1, 1, 1	[9, 8, 6, 5, 4, 1, 1, 1, 1]	36	2	12
9-30	-, 0, 1, 2, 3, 4, 4, 4, 2	[9, 8, 7, 5, 4, 1, 1, 1, 1]	37	2	12
9-31	-, 0, 1, 2, 3, 4, 4, 4, 3	[9, 8, 7, 6, 4, 1, 1, 1, 1]	38	2	12
9-32	-, 0, 1, 2, 3, 4, 4, 4, 4	[9, 8, 7, 6, 5, 1, 1, 1, 1]	39	2	10
9-33	-, 0, 1, 2, 3, 4, 5, 3, 3	[9, 8, 7, 6, 3, 2, 1, 1, 1]	38	2	16
9-34	-, 0, 1, 2, 3, 4, 5, 3, 7	[9, 8, 7, 6, 3, 2, 1, 2, 1]	39	2	18
9-35	-, 0, 1, 2, 3, 4, 5, 4, 5	[9, 8, 7, 5, 4, 2, 1, 1, 1]	38	2	12
9-36	-, 0, 1, 2, 3, 4, 5, 4, 3	[9, 8, 7, 6, 4, 2, 1, 1, 1]	39	2	12
9-37	-, 0, 1, 2, 3, 4, 5, 4, 4	[9, 8, 7, 6, 5, 2, 1, 1, 1]	40	2	16
9-38	-, 0, 1, 2, 3, 4, 5, 4, 7	[9, 8, 7, 6, 5, 2, 1, 2, 1]	41	2	18
9-39	-, 0, 1, 2, 3, 4, 5, 5, 1	[9, 8, 6, 5, 4, 3, 1, 1, 1]	38	2	12
9-40	-, 0, 1, 2, 3, 4, 5, 5, 2	[9, 8, 7, 5, 4, 3, 1, 1, 1]	39	2	12
9-41	-, 0, 1, 2, 3, 4, 5, 5, 3	[9, 8, 7, 6, 4, 3, 1, 1, 1]	40	2	12
9-42	-, 0, 1, 2, 3, 4, 5, 5, 4	[9, 8, 7, 6, 5, 3, 1, 1, 1]	41	2	12
9-43	-, 0, 1, 2, 3, 4, 5, 5, 5	[9, 8, 7, 6, 5, 4, 1, 1, 1]	42	1	6
9-44	-, 0, 1, 2, 3, 4, 5, 6, 4	[9, 8, 7, 6, 5, 3, 2, 1, 1]	42	2	26
9-45	-, 0, 1, 2, 3, 4, 5, 6, 5	[9, 8, 7, 6, 5, 4, 2, 1, 1]	43	1	6
9-46	-, 0, 1, 2, 3, 4, 5, 6, 6	[9, 8, 7, 6, 5, 4, 3, 1, 1]	44	1	6
9-47	-, 0, 1, 2, 3, 4, 5, 6, 7	[9, 8, 7, 6, 5, 4, 3, 2, 1]	45	0	N/A

【結果】*fig.* 9-47 皆為一直線；*fig.* 9-1 僅一個分叉點，故呈現花狀；*fig.* 9-44, 9-45, 9-46 呈現 T 型；其餘皆為多分枝圖。

(五)生成樹圖組　ν＝10

圖編號 Tree no.	父節點編碼 Code	賦點權重 Weights of vertices	權重總和 Summation	對數 pairs	交換能力 Moves
10-1	-, 0, 1, 1, 1, 1, 1, 1, 1, 1	[10, 9, 1, 1, 1, 1, 1, 1, 1, 1]	27	4	18
10-2	-, 0, 1, 2, 2, 2, 2, 1, 1, 1	[10, 9, 5, 1, 1, 1, 1, 1, 1, 1]	31	4	20
10-3	-, 0, 1, 2, 2, 2, 2, 2, 1, 1	[10, 9, 6, 1, 1, 1, 1, 1, 1, 1]	32	3	16
10-4	-, 0, 1, 2, 2, 2, 2, 2, 2, 1	[10, 9, 7, 1, 1, 1, 1, 1, 1, 1]	33	4	20
10-5	-, 0, 1, 2, 2, 2, 2, 2, 2, 2	[10, 9, 8, 1, 1, 1, 1, 1, 1, 1]	34	3	18
10-6	-, 0, 1, 2, 3, 2, 2, 2, 2, 2	[10, 9, 8, 2, 1, 1, 1, 1, 1, 1]	35	3	18
10-7	-, 0, 1, 2, 3, 2, 5, 2, 2, 2	[10, 9, 8, 2, 1, 2, 1, 1, 1, 1]	36	3	20
10-8	-, 0, 1, 2, 3, 2, 5, 2, 7, 2	[10, 9, 8, 2, 1, 2, 1, 2, 1, 1]	37	3	24
10-9	-, 0, 1, 2, 3, 3, 2, 2, 2, 1	[10, 9, 7, 3, 1, 1, 1, 1, 1, 1]	35	4	22
10-10	-, 0, 1, 2, 3, 3, 2, 2, 2, 2	[10, 9, 8, 3, 1, 1, 1, 1, 1, 1]	36	3	16
10-11	-, 0, 1, 2, 3, 3, 2, 6, 2, 2	[10, 9, 8, 3, 1, 1, 2, 1, 1, 1]	37	3	16
10-12	-, 0, 1, 2, 3, 3, 2, 6, 2, 8	[10, 9, 8, 3, 1, 1, 2, 1, 2, 1]	38	3	24
10-13	-, 0, 1, 2, 3, 3, 2, 6, 6, 1	[10, 9, 7, 3, 1, 1, 3, 1, 1, 1]	37	4	24
10-14	-, 0, 1, 2, 3, 3, 2, 6, 6, 2	[10, 9, 8, 3, 1, 1, 3, 1, 1, 1]	38	3	20
10-15	-, 0, 1, 2, 3, 3, 3, 2, 1, 1	[10, 9, 6, 4, 1, 1, 1, 1, 1, 1]	35	3	16
10-16	-, 0, 1, 2, 3, 3, 3, 2, 2, 1	[10, 9, 7, 4, 1, 1, 1, 1, 1, 1]	36	3	16
10-17	-, 0, 1, 2, 3, 3, 3, 2, 2, 2	[10, 9, 8, 4, 1, 1, 1, 1, 1, 1]	37	3	16
10-18	-, 0, 1, 2, 3, 3, 3, 2, 7, 2	[10, 9, 8, 4, 1, 1, 1, 2, 1, 1]	38	3	16
10-19	-, 0, 1, 2, 3, 3, 3, 2, 7, 7	[10, 9, 8, 4, 1, 1, 1, 3, 1, 1]	39	3	18
10-20	-, 0, 1, 2, 3, 3, 3, 3, 1, 1	[10, 9, 6, 5, 1, 1, 1, 1, 1, 1]	36	3	16
10-21	-, 0, 1, 2, 3, 3, 3, 3, 2, 1	[10, 9, 7, 5, 1, 1, 1, 1, 1, 1]	37	4	22
10-22	-, 0, 1, 2, 3, 3, 3, 3, 2, 2	[10, 9, 8, 5, 1, 1, 1, 1, 1, 1]	38	3	16
10-23	-, 0, 1, 2, 3, 3, 3, 3, 2, 8	[10, 9, 8, 5, 1, 1, 1, 1, 2, 1]	39	3	16
10-24	-, 0, 1, 2, 3, 3, 3, 3, 3, 1	[10, 9, 7, 6, 1, 1, 1, 1, 1, 1]	38	3	16
10-25	-, 0, 1, 2, 3, 3, 3, 3, 3, 2	[10, 9, 8, 6, 1, 1, 1, 1, 1, 1]	39	3	16
10-26	-, 0, 1, 2, 3, 3, 3, 3, 3, 3	[10, 9, 8, 7, 1, 1, 1, 1, 1, 1]	40	3	16
10-27	-, 0, 1, 2, 3, 4, 3, 3, 2, 2	[10, 9, 8, 5, 2, 1, 1, 1, 1, 1]	39	3	24
10-28	-, 0, 1, 2, 3, 4, 3, 3, 3, 2	[10, 9, 8, 6, 2, 1, 1, 1, 1, 1]	40	3	16
10-29	-, 0, 1, 2, 3, 4, 3, 3, 3, 3	[10, 9, 8, 7, 2, 1, 1, 1, 1, 1]	41	3	20
10-30	-, 0, 1, 2, 3, 4, 3, 6, 2, 2	[10, 9, 8, 5, 2, 1, 2, 1, 1, 1]	40	3	24
10-31	-, 0, 1, 2, 3, 4, 3, 6, 2, 8	[10, 9, 8, 5, 2, 1, 2, 1, 2, 1]	41	4	36
10-32	-, 0, 1, 2, 3, 4, 3, 6, 3, 2	[10, 9, 8, 6, 2, 1, 2, 1, 1, 1]	41	3	24
10-33	-, 0, 1, 2, 3, 4, 3, 6, 3, 3	[10, 9, 8, 7, 2, 1, 2, 1, 1, 1]	42	3	24
10-34	-, 0, 1, 2, 3, 4, 3, 6, 3, 8	[10, 9, 8, 7, 2, 1, 2, 1, 2, 1]	43	3	24
10-35	-, 0, 1, 2, 3, 4, 4, 2, 2, 2	[10, 9, 8, 4, 3, 1, 1, 1, 1, 1]	39	3	16

圖編號 Tree no.	父節點編碼 Code	賦點權重 Weights of vertices	權重總和 Summation	對數 pairs	交換能力 Moves
10-36	-, 0, 1, 2, 3, 4, 4, 2, 7, 2	[10, 9, 8, 4, 3, 1, 1, 2, 1, 1]	40	3	26
10-37	-, 0, 1, 2, 3, 4, 4, 3, 2, 1	[10, 9, 7, 5, 3, 1, 1, 1, 1, 1]	39	4	24
10-38	-, 0, 1, 2, 3, 4, 4, 3, 2, 2	[10, 9, 8, 5, 3, 1, 1, 1, 1, 1]	40	3	18
10-39	-, 0, 1, 2, 3, 4, 4, 3, 2, 8	[10, 9, 8, 5, 3, 1, 1, 1, 2, 1]	41	3	18
10-40	-, 0, 1, 2, 3, 4, 4, 3, 3, 1	[10, 9, 7, 6, 3, 1, 1, 1, 1, 1]	40	3	18
10-41	-, 0, 1, 2, 3, 4, 4, 3, 3, 2	[10, 9, 8, 6, 3, 1, 1, 1, 1, 1]	41	3	18
10-42	-, 0, 1, 2, 3, 4, 4, 3, 3, 3	[10, 9, 8, 7, 3, 1, 1, 1, 1, 1]	42	3	16
10-43	-, 0, 1, 2, 3, 4, 4, 3, 7, 1	[10, 9, 7, 6, 3, 1, 1, 2, 1, 1]	41	3	18
10-44	-, 0, 1, 2, 3, 4, 4, 3, 7, 2	[10, 9, 8, 6, 3, 1, 1, 2, 1, 1]	42	3	18
10-45	-, 0, 1, 2, 3, 4, 4, 3, 7, 3	[10, 9, 8, 7, 3, 1, 1, 2, 1, 1]	43	2	12
10-46	-, 0, 1, 2, 3, 4, 4, 3, 7, 7	[10, 9, 8, 7, 3, 1, 1, 3, 1, 1]	44	2	12
10-47	-, 0, 1, 2, 3, 4, 4, 4, 1, 1	[10, 9, 6, 5, 4, 1, 1, 1, 1, 1]	39	2	12
10-48	-, 0, 1, 2, 3, 4, 4, 4, 2, 1	[10, 9, 7, 5, 4, 1, 1, 1, 1, 1]	40	3	18
10-49	-, 0, 1, 2, 3, 4, 4, 4, 2, 2	[10, 9, 8, 5, 4, 1, 1, 1, 1, 1]	41	3	22
10-50	-, 0, 1, 2, 3, 4, 4, 4, 2, 8	[10, 9, 8, 6, 3, 1, 1, 1, 1, 1]	42	3	24
10-51	-, 0, 1, 2, 3, 4, 4, 4, 3, 1	[10, 9, 7, 6, 4, 1, 1, 1, 1, 1]	41	3	18
10-52	-, 0, 1, 2, 3, 4, 4, 4, 3, 2	[10, 9, 8, 6, 4, 1, 1, 1, 1, 1]	42	3	18
10-53	-, 0, 1, 2, 3, 4, 4, 4, 3, 3	[10, 9, 8, 7, 4, 1, 1, 1, 1, 1]	43	2	12
10-54	-, 0, 1, 2, 3, 4, 4, 4, 3, 8	[10, 9, 8, 7, 4, 1, 1, 1, 2, 1]	44	2	12
10-55	-, 0, 1, 2, 3, 4, 4, 4, 4, 1	[10, 9, 7, 6, 5, 1, 1, 1, 1, 1]	42	3	16
10-56	-, 0, 1, 2, 3, 4, 4, 4, 4, 2	[10, 9, 8, 6, 5, 1, 1, 1, 1, 1]	43	3	16
10-57	-, 0, 1, 2, 3, 4, 4, 4, 4, 3	[10, 9, 8, 7, 5, 1, 1, 1, 1, 1]	44	3	16
10-58	-, 0, 1, 2, 3, 4, 4, 4, 4, 4	[10, 9, 8, 7, 6, 1, 1, 1, 1, 1]	45	3	16
10-59	-, 0, 1, 2, 3, 4, 5, 3, 3, 3	[10, 9, 8, 7, 3, 2, 1, 1, 1, 1]	43	3	30
10-60	-, 0, 1, 2, 3, 4, 5, 3, 7, 3	[10, 9, 8, 7, 3, 2, 1, 2, 1, 1]	44	2	16
10-61	-, 0, 1, 2, 3, 4, 5, 3, 7, 7	[10, 9, 8, 7, 3, 2, 1, 3, 1, 1]	45	3	30
10-62	-, 0, 1, 2, 3, 4, 5, 3, 7, 8	[10, 9, 8, 7, 3, 2, 1, 3, 2, 1]	46	3	32
10-63	-, 0, 1, 2, 3, 4, 5, 4, 3, 2	[10, 9, 8, 6, 4, 2, 1, 1, 1, 1]	43	3	34
10-64	-, 0, 1, 2, 3, 4, 5, 4, 3, 3	[10, 9, 8, 7, 4, 2, 1, 1, 1, 1]	44	3	36
10-65	-, 0, 1, 2, 3, 4, 5, 4, 3, 8	[10, 9, 8, 7, 4, 2, 1, 1, 2, 1]	45	3	18
10-66	-, 0, 1, 2, 3, 4, 5, 4, 4, 2	[10, 9, 8, 6, 5, 2, 1, 1, 1, 1]	44	3	30
10-67	-, 0, 1, 2, 3, 4, 5, 4, 4, 3	[10, 9, 8, 7, 5, 2, 1, 1, 1, 1]	45	3	30
10-68	-, 0, 1, 2, 3, 4, 5, 4, 4, 4	[10, 9, 8, 7, 6, 2, 1, 1, 1, 1]	46	2	10
10-69	-, 0, 1, 2, 3, 4, 5, 4, 7, 2	[10, 9, 8, 6, 5, 2, 1, 2, 1, 1]	45	3	24
10-70	-, 0, 1, 2, 3, 4, 5, 4, 7, 3	[10, 9, 8, 7, 5, 2, 1, 2, 1, 1]	46	3	34
10-71	-, 0, 1, 2, 3, 4, 5, 4, 7, 4	[10, 9, 8, 7, 6, 2, 1, 2, 1, 1]	47	2	16

圖編號 Tree no.	父節點編碼 Code	賦點權重 Weights of vertices	權重總和 Summation	對數 pairs	交換能力 Moves
10-72	-, 0, 1, 2, 3, 4, 5, 5, 2, 2	[10, 9, 8, 5, 4, 3, 1, 1, 1, 1]	43	3	22
10-73	-, 0, 1, 2, 3, 4, 5, 5, 2, 8	[10, 9, 8, 5, 4, 3, 1, 1, 2, 1]	44	3	24
10-74	-, 0, 1, 2, 3, 4, 5, 5, 3, 1	[10, 9, 7, 6, 4, 3, 1, 1, 1, 1]	43	3	18
10-75	-, 0, 1, 2, 3, 4, 5, 5, 3, 2	[10, 9, 8, 6, 4, 3, 1, 1, 1, 1]	44	3	18
10-76	-, 0, 1, 2, 3, 4, 5, 5, 3, 3	[10, 9, 8, 7, 4, 3, 1, 1, 1, 1]	45	3	22
10-77	-, 0, 1, 2, 3, 4, 5, 5, 3, 8	[10, 9, 8, 7, 4, 3, 1, 1, 2, 1]	46	3	24
10-78	-, 0, 1, 2, 3, 4, 5, 5, 4, 1	[10, 9, 7, 6, 5, 3, 1, 1, 1, 1]	44	3	18
10-79	-, 0, 1, 2, 3, 4, 5, 5, 4, 2	[10, 9, 8, 6, 5, 3, 1, 1, 1, 1]	45	3	18
10-80	-, 0, 1, 2, 3, 4, 5, 5, 4, 3	[10, 9, 8, 7, 5, 3, 1, 1, 1, 1]	46	3	18
10-81	-, 0, 1, 2, 3, 4, 5, 5, 4, 4	[10, 9, 8, 7, 6, 3, 1, 1, 1, 1]	47	2	12
10-82	-, 0, 1, 2, 3, 4, 5, 5, 4, 8	[10, 9, 8, 7, 6, 3, 1, 1, 2, 1]	48	2	12
10-83	-, 0, 1, 2, 3, 4, 5, 5, 5, 1	[10, 9, 7, 6, 5, 4, 1, 1, 1, 1]	45	2	12
10-84	-, 0, 1, 2, 3, 4, 5, 5, 5, 2	[10, 9, 8, 6, 5, 4, 1, 1, 1, 1]	46	2	12
10-85	-, 0, 1, 2, 3, 4, 5, 5, 5, 3	[10, 9, 8, 7, 5, 4, 1, 1, 1, 1]	47	3	34
10-86	-, 0, 1, 2, 3, 4, 5, 5, 5, 4	[10, 9, 8, 7, 6, 4, 1, 1, 1, 1]	48	2	12
10-87	-, 0, 1, 2, 3, 4, 5, 5, 5, 5	[10, 9, 8, 7, 6, 5, 1, 1, 1, 1]	49	2	10
10-88	-, 0, 1, 2, 3, 4, 5, 6, 4, 3	[10, 9, 8, 7, 5, 3, 2, 1, 1, 1]	47	3	36
10-89	-, 0, 1, 2, 3, 4, 5, 6, 4, 4	[10, 9, 8, 7, 6, 3, 2, 1, 1, 1]	48	2	16
10-90	-, 0, 1, 2, 3, 4, 5, 6, 4, 8	[10, 9, 8, 7, 6, 3, 2, 1, 2, 1]	49	2	18
10-91	-, 0, 1, 2, 3, 4, 5, 6, 5, 2	[10, 9, 8, 6, 5, 4, 2, 1, 1, 1]	47	2	12
10-92	-, 0, 1, 2, 3, 4, 5, 6, 5, 3	[10, 9, 8, 7, 5, 4, 2, 1, 1, 1]	48	3	36
10-93	-, 0, 1, 2, 3, 4, 5, 6, 5, 4	[10, 9, 8, 7, 6, 4, 2, 1, 1, 1]	49	2	12
10-94	-, 0, 1, 2, 3, 4, 5, 6, 5, 5	[10, 9, 8, 7, 6, 5, 2, 1, 1, 1]	50	2	16
10-95	-, 0, 1, 2, 3, 4, 5, 6, 5, 8	[10, 9, 8, 7, 6, 5, 2, 1, 2, 1]	51	2	18
10-96	-, 0, 1, 2, 3, 4, 5, 6, 6, 1	[10, 9, 7, 6, 5, 4, 3, 1, 1, 1]	47	2	12
10-97	-, 0, 1, 2, 3, 4, 5, 6, 6, 2	[10, 9, 8, 6, 5, 4, 3, 1, 1, 1]	48	2	12
10-98	-, 0, 1, 2, 3, 4, 5, 6, 6, 3	[10, 9, 8, 7, 5, 4, 3, 1, 1, 1]	49	3	36
10-99	-, 0, 1, 2, 3, 4, 5, 6, 6, 4	[10, 9, 8, 7, 6, 4, 3, 1, 1, 1]	50	2	12
10-100	-, 0, 1, 2, 3, 4, 5, 6, 6, 5	[10, 9, 8, 7, 6, 5, 3, 1, 1, 1]	51	2	12
10-101	-, 0, 1, 2, 3, 4, 5, 6, 6, 6	[10, 9, 8, 7, 6, 5, 4, 1, 1, 1]	52	2	26
10-102	-, 0, 1, 2, 3, 4, 5, 6, 7, 4	[10, 9, 8, 7, 6, 4, 3, 2, 1, 1]	51	2	26
10-103	-, 0, 1, 2, 3, 4, 5, 6, 7, 5	[10, 9, 8, 7, 6, 5, 3, 2, 1, 1]	52	2	26
10-104	-, 0, 1, 2, 3, 4, 5, 6, 7, 6	[10, 9, 8, 7, 6, 5, 4, 2, 1, 1]	53	1	6
10-105	-, 0, 1, 2, 3, 4, 5, 6, 7, 7	[10, 9, 8, 7, 6, 5, 4, 3, 1, 1]	54	1	6
10-106	-, 0, 1, 2, 3, 4, 5, 6, 7, 8	[10, 9, 8, 7, 6, 5, 4, 3, 2, 1]	55	0	0

【結果】*fig.*10-106 皆為一直線；圖 10-1 僅一個分叉點，故呈現花狀；*fig.*10-102, 10-103, 10-104, 10-105 呈現 T 型；其餘皆為多分枝圖。

四、可放置最多對數並完成位置交換的可能性

Tree	對數、交換步數和權重的關係				
v=3	SW=6		v=6	SW=15	
	pairs	steps		pairs	steps
	0	N/A		2	10
v=4	SW=9			SW=17	
	pairs	steps		pairs	steps
	1	6		2	12
	SW=10			SW=18	
	pairs	steps		pairs	steps
	0	N/A		1	6
v=5	SW=12			SW=19	
	pairs	steps		pairs	steps
	1	6		1	6
	SW=14			SW=20	
	pairs	steps		pairs	steps
	1	6		1	6
	SW=15			SW=21	
	pairs	steps		pairs	steps
	0	N/A		0	N/A

Tree	對數、交換步數和權重的關係			
v=7	SW=18		SW=25	
	pairs	steps	pairs	steps
	2	10	1	6
	SW=21		SW=26	
	pairs	steps	pairs	steps
	2	12	1	6
	SW=22		SW=27	
	pairs	steps	pairs	steps
	2	10	1	6
	SW=23		SW=28	
	pairs	steps	pairs	steps

2	12	0	0
SW=24			
pairs	steps		
2	12		

Tree	對數、交換步數和權重的關係			
v=8	SW=21			
	pairs	steps	SW=30	
	3	14	pairs	steps
	SW=24		2	12
	pairs	steps	SW=31	
	2	12	pairs	steps
	SW=25		2	12
	pairs	steps	SW=32	
	2	10	pairs	steps
	SW=26		2	12
	pairs	steps	SW=33	
	2	10	pairs	steps
	SW=27		1	6
	pairs	steps	2	26
	2	10	pairs	steps
	3	18	1	6
	SW=28		SW=35	
	pairs	steps	pairs	steps
	2	12	1	6
	SW=29		SW=36	
	pairs	steps	pairs	steps
	2	12	0	0

五、權重用於最大交換對數和最小交換步數之關聯

權重在 n 點時有最大值SW_{max}和最小值SW_{min}。在形態上有直線樹、花狀樹、T 形樹和多分支樹。

（一） 直線樹

以每個點所含的權重加總計算。每個點所含的權重即為晚輩的數量(包含自己)，根據權重和圖的關係，得到「點的數量」、 「圖權重」以及「交換最大對數」和「最小交換步數」，四者關聯，其最大值可用一般式表示：

$$SW_{max} = n + (n-1) + \cdots + 3 + 2 + 1(\textit{原始點權重和})$$
$$= \frac{n(1+n)}{2}$$
$$= \frac{n^2+n}{2}$$

7-11 8-23 

9- 47 10-106

經由權重計算得到 10 點圖中最大的值，比較各圖後我們發現若有分岔點會減少權重。直線圖若每次都增加主幹的尾端，在沒有分叉的情形，每增加一個點，權重依序加 1，所以會產生最大值。

(二)花狀樹

圖形呈現「花狀」時 SW 最小(分岔點只有一個，並且除了根和分岔點以外的點都是葉)，即$1 \times (n-2) + (n-1) + n$，也等於$3n-3$。

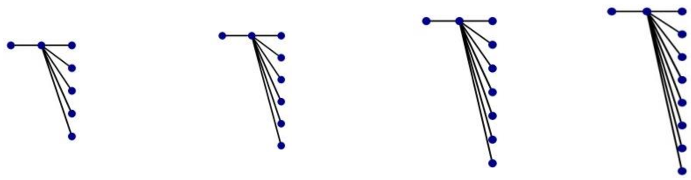
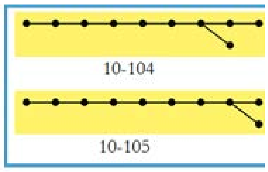


fig. 7-1　　*fig. 8-1*　　*fig. 9-1*　　*fig. 10-1*

當樹圖呈現花狀時，則最大交換對數為$\left\lfloor \frac{n}{2} - 1 \right\rfloor$，步數則是$4\left\lfloor \frac{n}{2} - 1 \right\rfloor + 2$，並且必為該點數之所有數圖中最有效率的解法。

(三) T 形樹：數量不多，卻必然存在，有相當代表性。



$$\frac{(f_0+1)[n+(n-f_0)]}{2} + \frac{(n-f_0-2)(n-f_0-1)}{2} + 1$$
$$= \frac{(f_0+1)(2n-f_0)}{2} + \frac{(n-f_0-2)(n-f_0-1)}{2} + 1$$
$$= \frac{2n \cdot f_0 - {f_0}^2 + 2n - f_0}{2} + \frac{n^2 - 2n \cdot f_0 - n + {f_0}^2 + f_0 - 2n + 2f_0 + 2}{2} + 1$$
$$= \frac{n^2 - n + 2f_0 + 4}{2}$$
$$= \frac{n(n-1)}{2} + f_0 + 2$$

(四)多分支樹(multi-branches)

$$SW(G) = \frac{V(V+1) - S_1 + 2S_2}{2}$$

$$S_1 = \sum_{i=0}^{f-1} V(b_i) \cdot \left[2V - 2f_i - \left(\sum_{j=0}^{i} V(b_j) \right) - 2i - 1 \right]$$

$$S_2 = \sum_{k=0}^{f-1} SW(b_k)$$

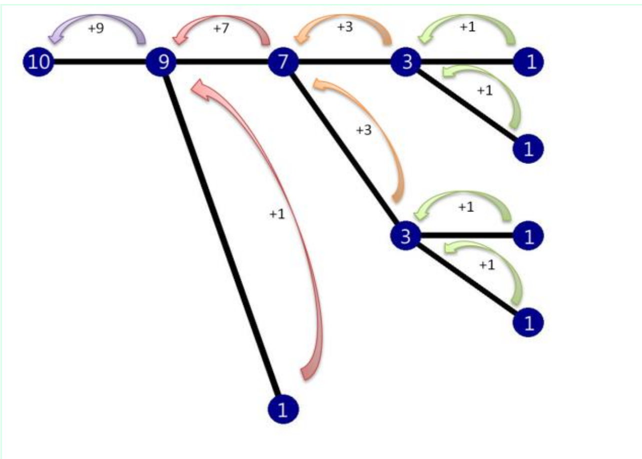
【說明】$V(G)$為圖G的點數，b_i為第i個分支，f_i為第i個分岔點，f為分岔點的數量。
由此得到多分支樹可計算性。

（五）表格判別法

我們使用各點編碼搭配編號，將其轉為二維表格，適用於各種圖形。

fig. 10-13	**-**	**0**	**1**	**2**	**3**	**3**	**2**	**6**	**6**	**1**
編號	0	1	2	3	4	5	6	7	8	9
各點權重	1	1	1	1	1	1	1	1	1	1
	9	1	3	1			1			
		7	3	1			1			
總和	10	9	7	3	1	1	3	1	1	1

【步驟】1. 將圖編碼、編號依序列出，並在下方留下權重欄位。
2. 將各點權重加 1。
3. 依序從編號最大的點推至編號最小的點（**左←右**）。
4. 將該點目前權重加至與其父節點（編碼）同編號點。
5. 重複 STEP4，直到每一點操作過 1 次。
6. 將各點權重加總，並填入下方總和行。
7. 將各點的總和加總即為該圖的 SW。



依序得到 *fig.*10-13 權重　10+9+7+3+1+1+3+1+1+1 = 37

陸、　研究結論

一、本研究創發之$\mathbb{G}(\mathbb{S} \leftrightarrow \mathbb{P})\mathbb{E}$是用以產生確認唯一樹工具，其流程能將所有同構圖以同型態出現，再予以刪除即可得到異構圖。

二、交換最大對數以及最少步數在部分類似圖形上呈現規律。若為直線形則無法交換；若由單點延伸出所有點其對數為$\left\lfloor\frac{n}{2}-1\right\rfloor, n \geq 4$，交換步數為$\left\lfloor\frac{n}{2}-1\right\rfloor \times 4+2, n \geq 4$，此類樹圖交換之步數必為相同點數之樹圖中最有效率方法。

三、單中心與雙中心用以區辨樹的性質，本研究據以判斷樹圖之異構圖數量。

四、樹的權值越大則交換效率越低。

直線樹圖權值$SW_{max}=n+(n-1)+\cdots+3+2+1$(*原始點權重和*)

$$=\frac{n(1+n)}{2}$$

$$=\frac{n^2+n}{2}$$

五、圖形呈現「花狀」時 SW 最小，通式如右：$3n-3$。

六、T 形樹得通式如右： $\frac{n(n-1)}{2}+f_0+2$

七、多分支樹(multi-branches)權重通式如下：

$$SW(G)=\frac{V(V+1)-S_1+2S_2}{2}$$

$$S_1=\sum_{i=0}^{f-1} V(b_i)\cdot\left[2V-2f_i-\left(\sum_{j=0}^{i} V(b_j)\right)-2i-1\right]$$

$$S_2=\sum_{k=0}^{f-1} SW(b_k)$$

八、將各點當前權重加至與其父節點(編碼)相同編號的點是表格判別法，利用各點編碼搭配編號，將其轉為二維表格，適用於各種圖形。

柒、 未來研究方向

本次研究我們針對樹圖發展出工具，但在交換步數及對數尚無法突破，留待未來探究。

捌、 參考文獻

〔1〕張鎮華(2020)。演算法觀點的圖論。臺北市：台大出版中心出版。

〔2〕A. Caley. On the Analytical Forms Called Trees, with application to the theory of chemical combinations. The British Association for the Advancement of Science. Pp. 257-305.

〔3〕A. Caley. On the Analytical Forms Called Trees.American Jounal of Mathematics, 1881, Vol. 4, No. 1 (1881) pp. 266-268. The Johns Hopkins University Press.

〔4〕黃楷宸、林子濠等(2020)。Crazy Knights. 中華民國第 60 屆全國科學展覽會數學科作品說明書。

〔5〕N. J. A. Sloane and Simon Plouffe, The Encyclopedia of Integer Sequences, Academic Press, 1995 (includes this sequence).

〔6〕P. Leroux et B. Miloudi, (1992). Généralisations de la formule d'Otter

【評語】030423

本作品延伸第 60 屆科展作品 Crazy Knights。相對於已完成的環圖分析，作者們探討非環圖的情形，從先前的騎士交換節點非環圖構思，嘗試自創工具以產出無標號樹，並將無標號樹賦予生成編碼權值，解析其規律。頂點數為 n 的不同構的無標號樹有多少個？這是一個有趣而且有難度的問題。本作品的作者們依據中心點的個數來分類，運用特殊的標號方式，針對頂點數不超過 10 的無標號樹有多少種給出了答案。能夠把所有不超過 10 個點的不同構的無標號樹都具體的構造出來是一個不容易的工作。光從這件事就可以看得出來作者們確實是花費了許多心思來完成這項研究工作，這點很值得肯定。比較美中不足的是，在整個論述的過程中完全沒有提到構造一個長度為 n 的序列的原因，感覺上只是將所有的無標號樹與對應的序列表列出來而已。對於給定的兩個無標號樹，如何藉助它們所對應的序列來判斷它們是否同構？這一點，似乎也沒有交代。作者們或許可以考慮用比較小的 n 值來舉例說明建構的原則和判別是否同構的方法。如果可以說明使用新的標號方式會比原有的標號方式要來的好，作品會更有價值。

作品簡報

030423

Knights Swap

無標號樹之探討

研究動機

1. 試探研究可行性[1]。
2. 基於補充研究缺口[4] 。

Crazy Knights (2020)

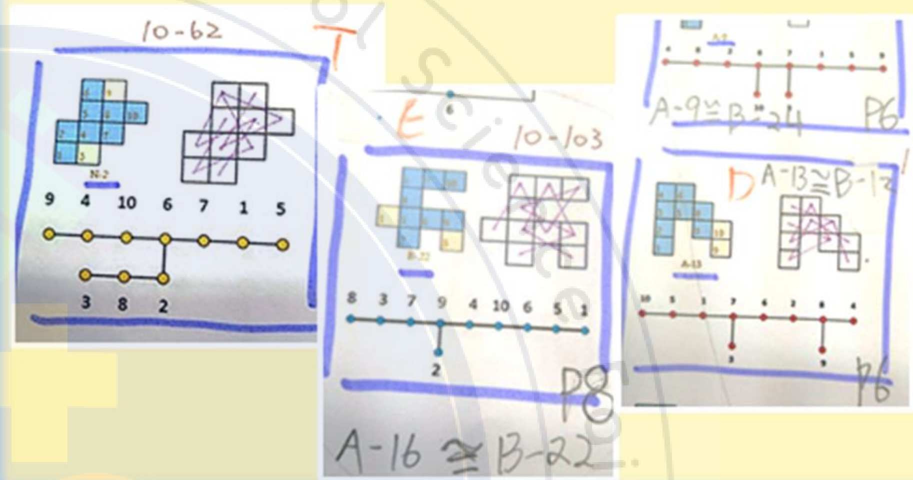


圖1 既有研究之同構圖檢視

研究目的

1. 提出系統化產生樹圖的方法。
2. 以樹圖說明交換關係之結構。
3. 分析樹圖權重和訊息對交換和交換步數之關係。

Encode

1. 令根(Root)為V_0並依序編號。
2. 自根開始延伸，沿邊依序標號。
3. 標號至葉(v_4, v_5, v_7, v_9)之後回溯。
4. 分岔點處理v_2和 v_3折返分岔點續標至根。
5. 最後一點皆標$v-1$。

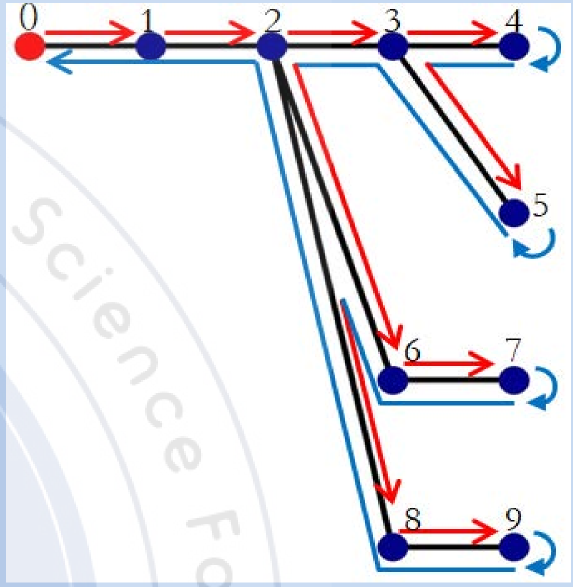


圖2 摸牆走迷宮

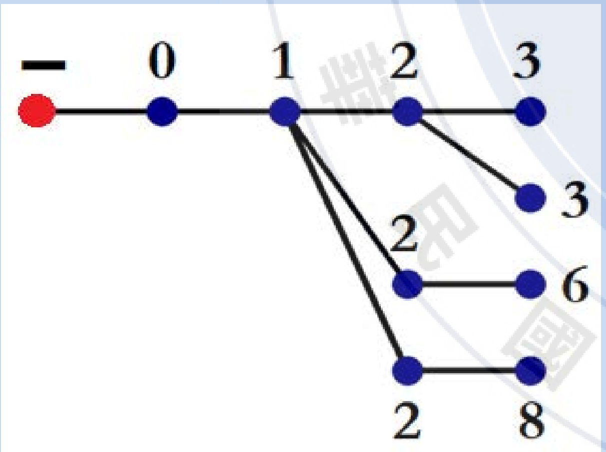


圖3 從父節點編碼形成數對

1. 父節點編碼：標註所有點之「父點」，任1點前面沒有連接其他點標示為「–」。
2. 編碼：每一點之父點依序串接形成數對。

Example

fig.10-12 encoded as

(–, 0, 1, 2, 3, 3, 2, 6, 2, 8)

Encode 性質討論

1. 編碼是各點父點之編號，一個n點的圖編碼有n項。
2. 編碼第$k+1$項(a_k)點與編號k點相連，即(k, a_k)。
3. 相鄰矩陣，將x軸以a_k表示，y軸以k表示。

   $\left(A_{a_k,k} = A_{k,a_k} = 1\right)$。$A_{x,y}$ 表示相鄰矩陣的(x, y)。

	0	1	2	3	4	5	6	7	8	9
0	0	1	0	0	0	0	0	0	0	0
1	1	0	1	0	0	0	0	0	0	0
2	0	1	0	1	0	0	1	0	1	0
3	0	0	1	0	1	1	0	0	0	0
4	0	0	0	1	0	0	0	0	0	0
5	0	0	0	1	0	0	0	0	0	0
6	0	0	1	0	0	0	0	1	0	0
7	0	0	0	0	0	0	1	0	0	0
8	0	0	1	0	0	0	0	0	0	1
9	0	0	0	0	0	0	0	0	1	0

圖4 編碼後相鄰矩陣

Generate 性質討論

於 n 點樹基礎下，V_0~V_{n-1}每一點各延伸1次，以此產生新圖，因此將會產出 n 張圖。

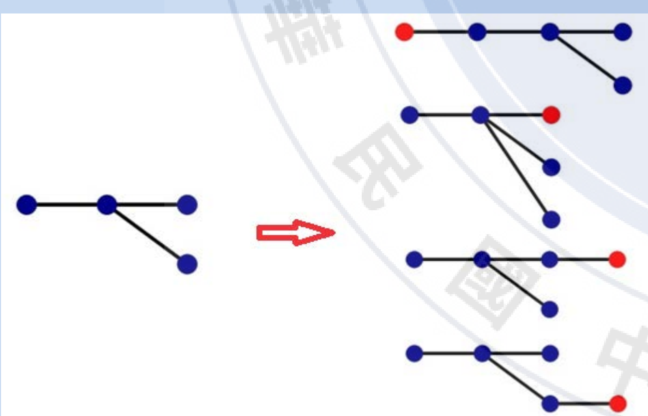


圖5 「3點樹」延伸

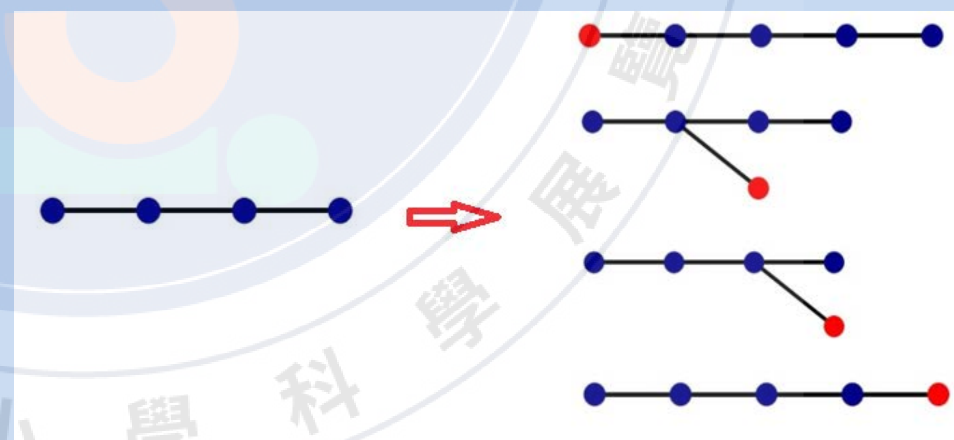


圖6 「4點樹」延伸

$\mathbb{S} \leftrightarrow \mathbb{P}$性質討論

Generate每一張圖

都必須經過S↔P步驟，

將每一點視為「根」，

並繪出數張同構圖。

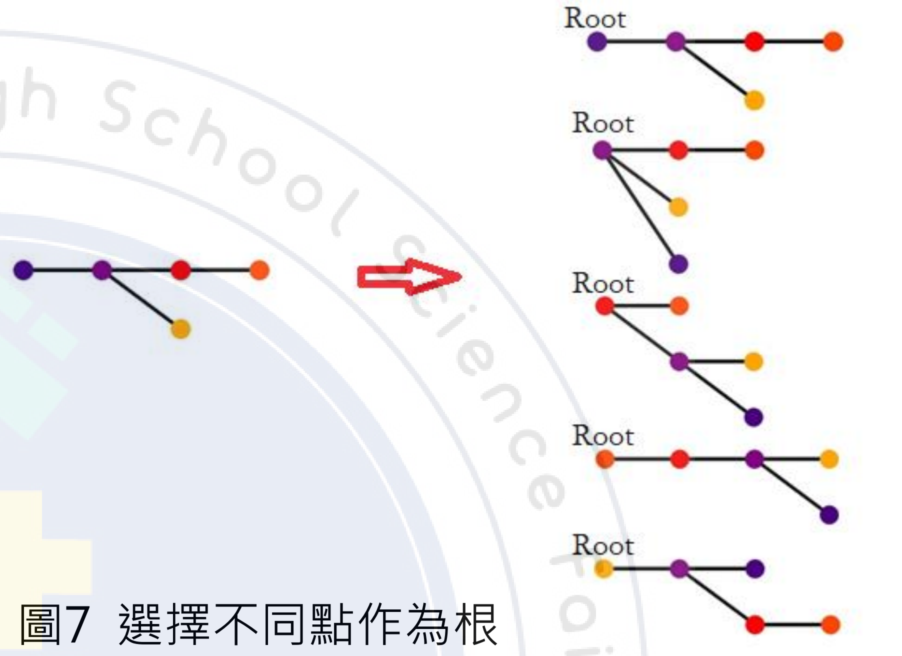


圖7 選擇不同點作為根

1. 將每一張同構圖進行Sort。
2. 紅色標示分岔點。
3. 框出分岔點後方子樹。
4. 分離→觀察→依上下排序。
5. 離根較遠的分岔點優先排序。

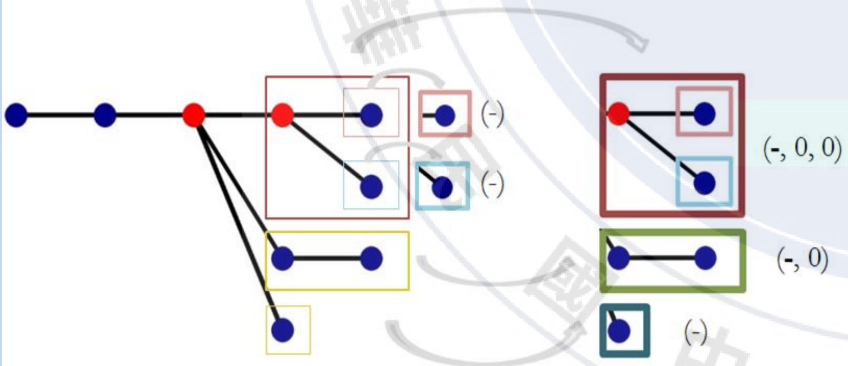


圖8 將分支進行排續

編碼比較原則

通例說明

位數	1	2	3	4	5	6	7	8	9	10
編碼 A	-	0	1	2	3	4	4	4	2	8
關係		=	=	=	=	=	=	=	<	
編碼 B	-	0	1	2	3	4	4	4	3	1

【說明1】因編碼2的第9位>編碼1的第9位，得到編碼B>編碼A。

位數	1	2	3	4	5	6	7	8	9	10
編碼 C	-	0	1	2	3	3	3	2	7	2
關係		=	=	=	=	=				
編碼 D	-	0	1	2	3	3				

【說明2】因為編碼1的第7位為3，但編碼2並沒有第7位，得到編碼C>編碼D。

反例比較

位數	1	2	3	4	5	6	7	8	9	10
編碼 E	-	0	1	2	3	4	3	6	3	3
關係		=	=	=	=	=	<			
編碼 F	-	0	1	2	3	4	4			

【說明3】因編碼2的第9位>編碼1的第9位，因此可以得到編碼F>編碼E，即編碼長度不能直接決定編碼大小。

研究結果

1. 在Pick步驟產生出來的所有同構圖都進行Sort步驟後，將每一張樹圖(整張)進行Encode(編碼)，再以前述原則比較該編碼並選出編碼最大的圖。
2. 在所有Generate步驟所產生的樹圖進行P、S、E步驟過後，最後再把所有重複(編碼完全相同)的樹圖刪除掉，可得到$n+1$點的所有無標號樹圖，產出結果皆為異構圖。
3. 若欲求出$n+2$點的所有無標號異構樹圖，則將$n+1$點的所有無標號異構樹圖進行上述G、S、P、E四個步驟即可。

n點樹圖	3	4	5	6	7	8	9	10	11
樹圖數量	1	2	3	6	11	23	47	106	235

證明

G可以推導所有n點樹

定義n點樹係$T_n, n > 1, n \in \mathbb{N}$

任意樹刪除葉l必產生$n-1$點T_{n-1_1}

l之相鄰點為m且$m \in V(T_{n-1_1})$

若$T_{n-1_2} \cong T_{n-1_1}$且有一點$m' = m,\ m' \in V(T_{n-1_2})$

則自m'延伸之T_{n_2}與自m延伸自之T_{n_1}同構

$$T_{n_2} \cong T_{n_1} \cong T_n$$

$\therefore T_n$必定能在$n-1$點T_{n-1}生成。

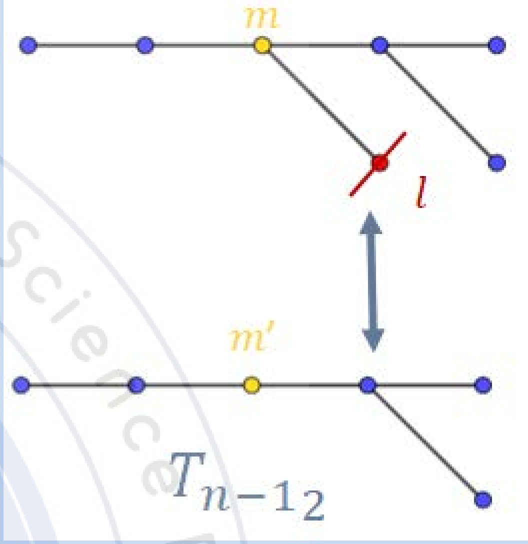


圖9 刪除任意葉

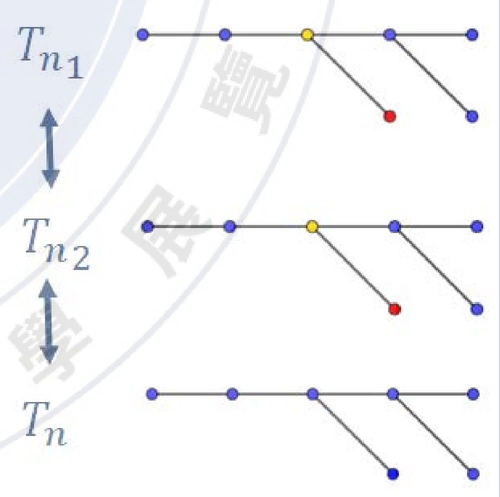


圖10 從葉的相鄰點延伸

A. Cayley [2] [3]

"the total number of distinct trees with N knots"

No. of Knots	3	4	5	6	7	8	9	10	11	12
No. of Central Trees	1	1	2	3	7	12	27	55	127	284
No. of Bicentral Trees	0	1	1	3	4	11	20	51	108	267
Total	1	2	3	6	11	23	47	106	235	551

研究發現

Cayley中心樹方法檢證$\mathbb{G}(\mathbb{S} \leftrightarrow \mathbb{P})\mathbb{E}$

No. of Knots	3	4	5	6	7	8	9	10	11
Total	1	2	3	6	11	23	47	106	235

研究發現　主幹Log特性

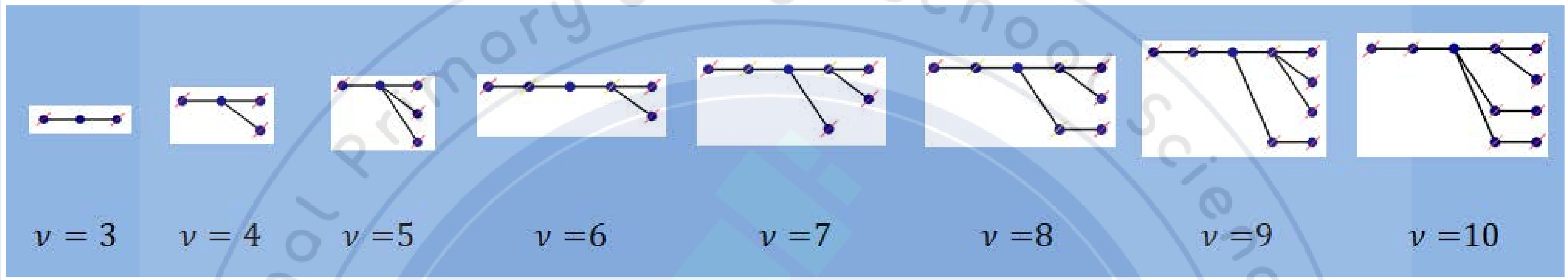


圖11　奇數點主幹中心樹

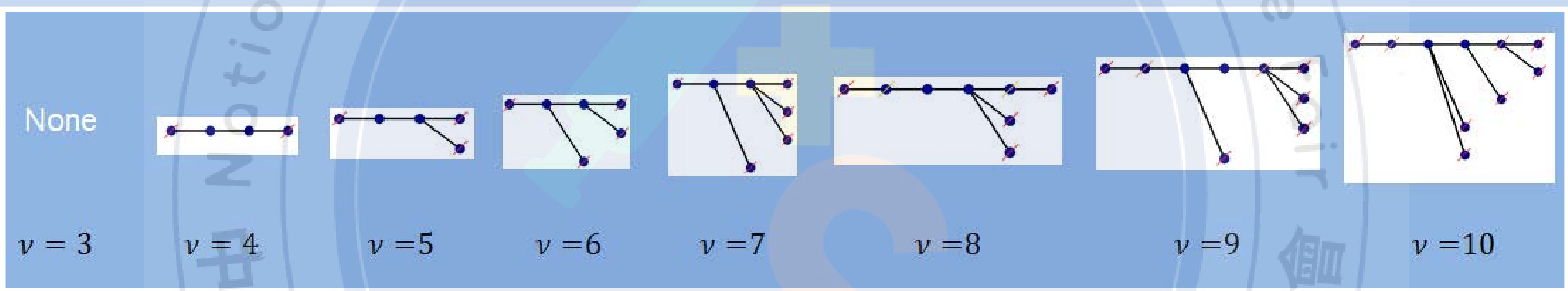


圖12　偶數點主幹中心樹

研究發現　主幹長度與刪除葉次數k的關係

Log	Central Trees	Bicentral Trees
vertices	odd	even
	2k+1	2k+2
edges	2k	2k+1

證明　所有樹圖皆為雙/單中心樹

若$deg(T_n) = \overbrace{\{0,0,\ldots,0\}}^{V(T_n)}, V(T_n) > 1$則任意2點$v, u$必不相連。

$\because\ deg(T_n) = \overbrace{\{0,0,\ldots,0\}}^{V(T_n)}$若且唯若$V(T_n) = 1$

$\therefore$若以C_n表n中心圖之中心$, n > 1$則$deg(C_n) \neq \{0\}\left(\overbrace{\{0,0,\ldots,0\}}^{V(C_n)}\right)$

若有$C_n, n > 1$則$deg(C_n) = \overbrace{\{1,1,\ldots,1\}}^{V(C_n)}$。

$$\because\ E(C_n) = \frac{\sum_{i=1}^{V(C_n)} \deg(v_i)}{2}$$

且$E(C_n) = V(C_n) - 1$，且$\sum_{i=1}^{V(C_n)} \deg(v_i) = V(C_n)$

$\therefore$簡化上式並解得: $V(C_n) - 1 = \frac{V(C_n)}{2}, V(C_n) = 2$

任意符合中心條件的樹C_n滿足$V(C_n) = 1$或2，得證。

圖G, $deg(G) = \overbrace{\{0,0,\ldots,0\}}^{V(T_n)}$
任意2點v, u必不相連，G不是樹。

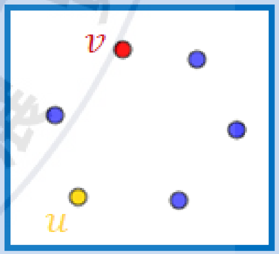


圖13 每一點度序列皆為0之非樹圖

研究結論

賦點權重SW分類樹型態

直線 *Path*

圖14 6點直線圖

$$\frac{V^2+V}{2}$$

花狀 *Flourish*

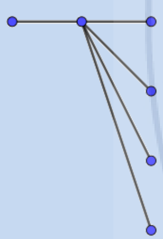


圖15 6點花狀圖

$$3V-3$$

T型 *T-type*

圖16 7點T型圖

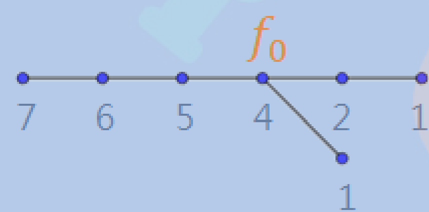
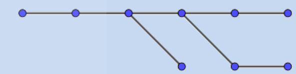


圖17 分岔點註記

$$\frac{V^2-V}{2}+f_0+2$$

(3, 4)

圖18 延伸過程①



(6), (3, 4)

圖19 延伸過程②

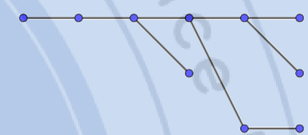


圖20 延伸過程③

多分支 *Multi-branches*

$$SW(G)=\frac{V(V+1)-S_1+2S_2}{2}$$

$$S_1=\sum_{i=0}^{f-1} V(b_i)\cdot\left[2V-2f_i-\left(\sum_{j=0}^{i} V(b_j)\right)-2i-1\right]$$

$$S_2=\sum_{k=0}^{f-1} SW(b_k)$$

參考文獻

〔1〕張鎮華(2020)。演算法觀點的圖論。臺北市：台大出版中心出版。

〔2〕A. Caley. On the Analytical Forms Called Trees, with application to the theory of chemical combinations. The British Association for the Advancement of Science. Pp. 257-305.

〔3〕A. Caley. On the Analytical Forms Called Trees. American Journal of Mathematics, 1881, Vol. 4, No. 1 (1881) pp. 266-268. The Johns Hopkins University Press. Stable URL: http://www.jstor.org/stable/2369158

〔4〕黃楷宸、林子濠等(2020)。Crazy Knights. 中華民國第60屆全國科學展覽會數學科作品說明書。

〔5〕N. J. A. Sloane and Simon Plouffe, The Encyclopedia of Integer Sequences, Academic Press, 1995 (includes this sequence).

〔6〕P. Leroux et B. Miloudi, (1992). Généralisations de la formule d'Otter https://oeis.org/A000081 / a0000812.pdf