

第一章 緒論

第一節 前言

近年來日本相當流行一種“paper-and-pencil puzzles”類型的遊戲，例如：Nurikabe、Nonogram、Sudoku、Light Up 等，由於此類遊戲規則簡單並且具有多樣的盤面變化，遊戲進行過程中透過縝密的思考以完成遊戲即為其樂趣所在，目前在世界各地皆廣受注目。遊戲進行方法為依照特定規則填補部分方格以形成完整的盤面，其共同特徵有：盤面由棋盤格狀之單位組成、遊戲必須遵循特定之規則項目進行、以及初始之盤面具有特定的資訊特徵。

在[4]中，列舉出下列項目為“paper-and-pencil puzzles”類型之遊戲：

- ◆ Nurikabe [3]*
- ◆ Nonogram [8](或 Paint-by-Numbers)*
- ◆ Slither Link [9]*
- ◆ Cross Sum [7](或 Kakkuro)*
- ◆ Number Place [7](或 Sudoku)*
- ◆ Heyawake

以上附註*者為已知被證明是 NP-complete，當中 Nonogram 為在“paper-and-pencil puzzles”中第一項被證明是 NP-complete 的遊戲，其證明者為 Ueda 和 Nagao[8]。對於這些以數學為基礎的理論問題，我們希望探討其是否具有演算法可解。針對這類問題，長久以來有許多數理邏輯學家投入研究並提出了幾種不同的演算法定義，以解決這些理論問題，也因此建立了計算理論(computability theory)。本節將會先對於計算理論的基礎概念作一個介紹。

首先介紹計算複雜性理論(computational complexity theory)，為計算理論的分支，其使用一些數學的方法對計算中所需的各種資源的耗費作定量的分析，並

研究出各類問題在計算複雜程度上的相互關係和基本性質，是演算法分析的理論基礎。當我們計算某一類問題時，往往必須花費時間與儲存空間等資源，而資源耗費即問題大小的函數，其耗費多寡決定於被計算問題的大小，稱為問題對該資源所需求的複雜度。分析複雜度函數之增長、將被計算的問題作分類、研究各種不同資源耗費之間的關係、以及估計資源耗費情況的上界和下界等，皆為計算複雜性理論的主要研究內容。

計算理論中的另一個重要的分支即自動機理論(automata theory)，其為一種數學模型。在自動機理論中，自動機是一個數學概念，作為離散系統的抽象模型，並可作為語言識別器，用來研究演算法、程序和計算複雜性理論。自動機可分成無限自動機(infinite automata theory)與有限自動機(finite automata theory)。無限自動機主要研究課題是研究儲存量無限的離散數字系統之功能和結構及兩者關係的理論；有限自動機論的主要研究課題是研究儲存量有限的離散數字系統的功能和結構以及兩者的關係。任意一種自動機，按其動作的確定程度，大體可分為確定的和非確定的兩類。在每一時刻，根據當時的狀態和輸入，若機器的動作可唯一確定時，則稱該機器為確定性(deterministic)的；若有多個動作可供選擇時，則稱該機器為非確定性(nondeterministic)的。對於同一類型的自動機，確定型可以看作是非確定型的特殊情形。

P 是可在多項式時間內用確定性演算法求出的所有問題之集合；NP 是可在多項式時間內用非確定性演算法求出的所有問題之集合，而目前 P 是否等於 NP 尚無定論。計算理論中另外一項重要概念即 NP-Completeness，它表示出某些問題的固有複雜度。一旦確定一類問題是 NP-Complete 時，就可知道這類問題實際上是具有相當複雜程度的困難問題。目前任意的 NP-complete 問題在 worst case

時，即使使用目前最好的演算法來解，仍然都要花 exponential 的時間才能解決，也就是說 NP-complete 問題在 worst case 下目前都還找不出任何 polynomial time algorithm 來解決。圖 1 為 P、NP、NP-complete 三種問題集合在 $NP \neq P$ 前提下的關係圖。

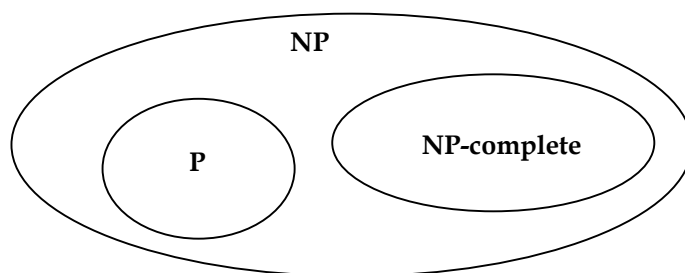


圖 1. 假設 $NP \neq P$ ，P、NP、NP-complete 三種問題集合的關係圖

NP-complete 問題中的其中一項理論：若某個 NP-complete 的問題可在 polynomial time 內解出，則所有 NP 問題皆可在 polynomial time 內解決，也就是說 $NP = P$ 。但是這個假設「所有 NP-complete 問題皆可在 polynomial time 內解決」目前幾乎不太可能，因此所有 NP 問題皆可在 polynomial time 內解決的可能性也很低。

目前為止，針對 NP-complete 問題，一般會採取幾種方式解決：找尋近似解、針對其困難尋找 Heuristic 的演算法、或者是將問題範圍縮小來解決。

第二節 問題定義

在“paper-and-pencil puzzles”類型的遊戲裡，其中一項目前十分受到歡迎的遊戲為“Light Up”，是世界知名遊戲 Sudoku 的發明者—日本 Nikoli 公司（全名為 Puzzle Communication Nikoli）所推出的一項益智遊戲。這個遊戲是一種

binary-determination logic puzzle，其規則十分簡單，但所構成的盤面與遊戲過程變化則相當多樣並有趣；2006 年，Nikoli 公司曾推出兩本書，內容為有關於 Light Up 這個遊戲的完全介紹。

Light Up 這個遊戲的主要概念為存在一個有黑色方格與白色方格的棋盤狀盤面，該如何放置燈泡(light bulbs)使得所有白色方格皆為照明狀態為該遊戲之目的，圖 2 為其初始與終止盤面：

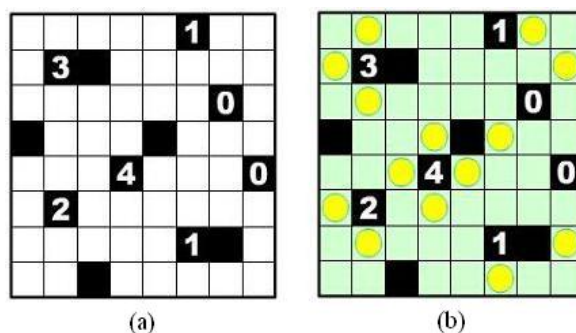


圖 2. (a)初始盤面 (b)終止盤面

其燈泡放置方式必須符合下列規則：

1. 燈泡只能放置於白色方格，盤面中的數字表示水平方向與垂直方向的鄰近位置之燈泡應放置個數。
2. 被放置燈泡的位置其同一列(row)與同一行(column)向外的所有連續白色方格將呈現一個十字型的照明狀態直到遇到黑色方格或者邊界方可停止，相當於照射的動作。
3. 放置燈泡時，不可如圖 3 出現燈泡彼此照射之情形。
4. 放置燈泡直到所有白色方格皆呈現照明狀態為止。

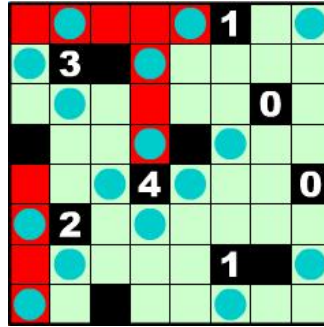
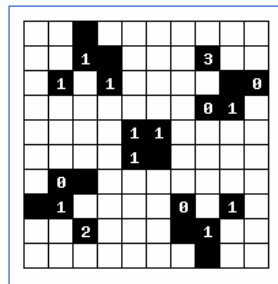


圖 3. 燈泡互相照射則為不允許

目前許多網站皆有此 Light Up 遊戲之應用程式可實際操作，以圖 4 為例 [11]，該網站包含非常多種 Light Up 盤面，可以選擇 Light Up 盤面之行列數大小以及難易程度。圖 4(a)、(b)、(c)、(d)為四個 10×10 的 Light Up 盤面，其難易程度偏簡易；以及圖 4(e)、(f)為兩個 10×10 的 Light Up 盤面，其難易程度偏困難。

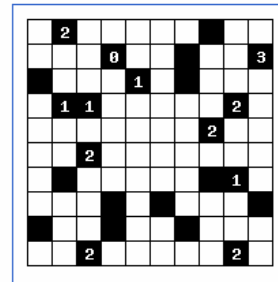
若以圖 4(f)之一個 10×10 的 Light Up 困難盤面為例，依照 Light Up 規則進行放置燈泡之動作，其操作技巧有以下幾個步驟（如圖 5）：(1)首先尋求盤面中是否存在有一數字方格所示數字與其自身在水平方向與垂直方向相鄰可放置燈泡的白色方格數目相等，若存在，則可先行在該白色方格位置放置燈泡（如圖 5(a)）；(2)在放置第一個燈泡後，燈泡即會將其相鄰之同列與同行所有連續白色方格照明，此時部份數字方格之相鄰白色方格數目已減少，因此再次檢查盤面中是否存在有一數字方格所示之數字與其自身在水平方向與垂直方向之相鄰白色方格數目相等，將符合數目並可放置燈泡之白色方格同樣放置燈泡，放置後燈泡自行執行照明之動作，並重覆此步驟直到無法再找到數字與相鄰可放置燈泡之白色方格數目相等的數字方格（如圖 5(b)）；(3)以數目字由大至小之順序開始試著放置符合該數字數目之燈泡於其相鄰可放置燈泡之白色方格中的任意位置（如圖 5(c)、(d)、(e)），此時剩餘之白色方格若可找到填入燈泡之位置使所有白色方格

被照明，並且不違反規則，則得到 Light Up 盘面之解（如圖 5(f)）。若剩餘之白色方格無法全部被照明，則將方才試放之燈泡取下，並重覆步驟(3)之動作。



10x10 Light Up Easy Puzzle ID: 939,218

Ready Start Over Print New Puzzle

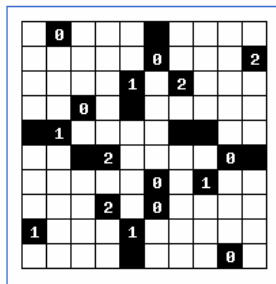


10x10 Light Up Easy Puzzle ID: 8,644,677

Ready Start Over Print New Puzzle

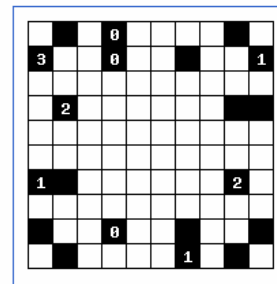
(a)

(b)



10x10 Light Up Easy Puzzle ID: 3,152,618

Ready Start Over Print New Puzzle

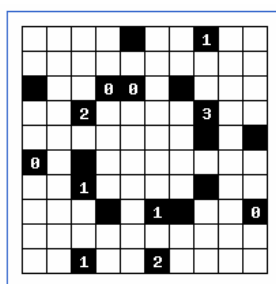


10x10 Light Up Easy Puzzle ID: 2,866,100

Ready Start Over Print New Puzzle

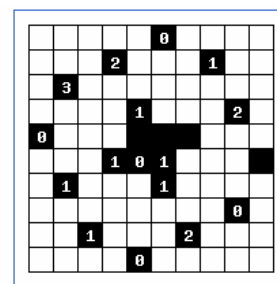
(c)

(d)



10x10 Light Up Hard Puzzle ID: 1,358,158

Ready Start Over Print New Puzzle



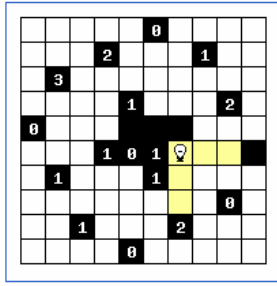
10x10 Light Up Hard Puzzle ID: 6,687,118

Ready Start Over Print New Puzzle

(e)

(f)

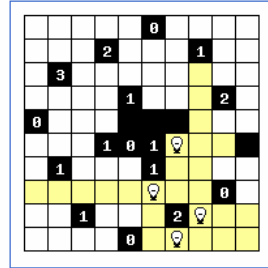
圖 4. 六個 10x10 Light Up 初始盘面[11]



10x10 Light Up Hard Puzzle ID: 6,687,118

Ready Start Over Print New Puzzle

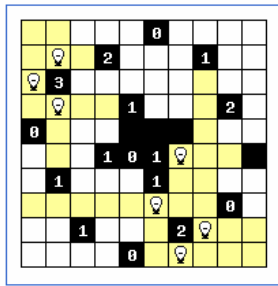
(a)



10x10 Light Up Hard Puzzle ID: 6,687,118

Ready Start Over Print New Puzzle

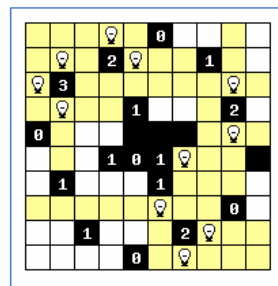
(b)



10x10 Light Up Hard Puzzle ID: 6,687,118

Ready Start Over Print New Puzzle

(c)



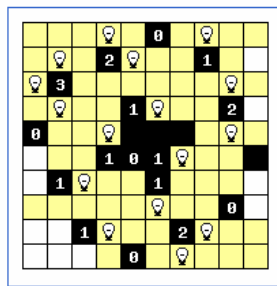
10x10 Light Up Hard Puzzle ID: 6,687,118

Ready Start Over Print New Puzzle

(d)

Congratulations! You have solved the puzzle in 00:53.71

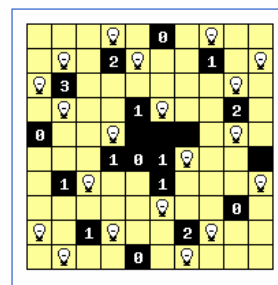
Submit your score to the Hall of Fame



10x10 Light Up Hard Puzzle ID: 6,687,118

Ready Start Over Print New Puzzle

(e)



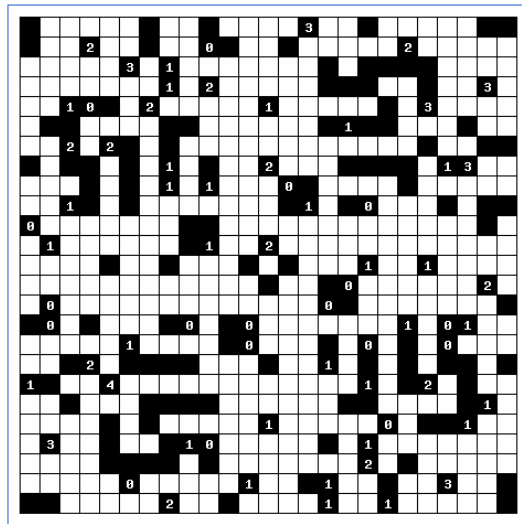
10x10 Light Up Hard Puzzle ID: 6,687,118

Start Over Print New Puzzle

(f)

圖 5. 10x10 Light Up 盤面之求解步驟[11]

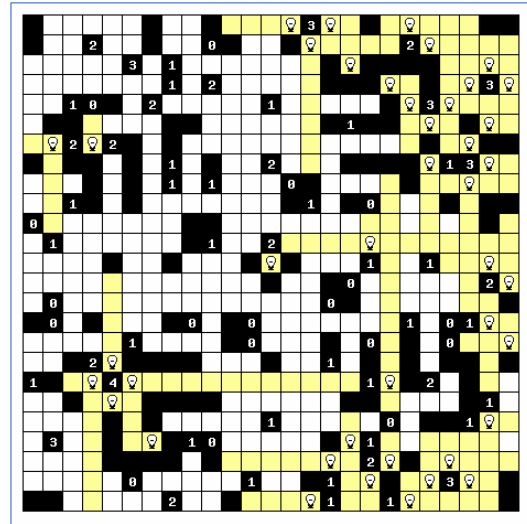
若面對較大的盤面，我們一樣可以依照上述步驟操作，如圖 6(a)為一個 25x25 的 Light Up 盤面，圖(b)至圖(d)為求得其解之過程。盤面較大，則愈難找到確定可放置燈泡的位置，因此要經過不斷的試驗，找出適當的位置放置燈泡，此過程亦為本遊戲的趣味所在。



25x25 Light Up Easy Puzzle ID: 9,993,036

Ready Start Over Print New Puzzle

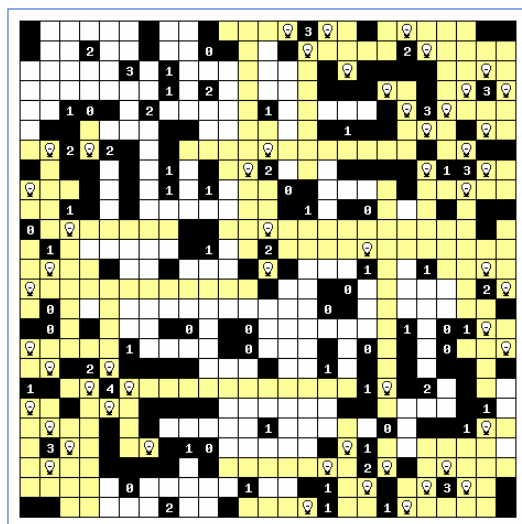
(a)



25x25 Light Up Easy Puzzle ID: 9,993,036

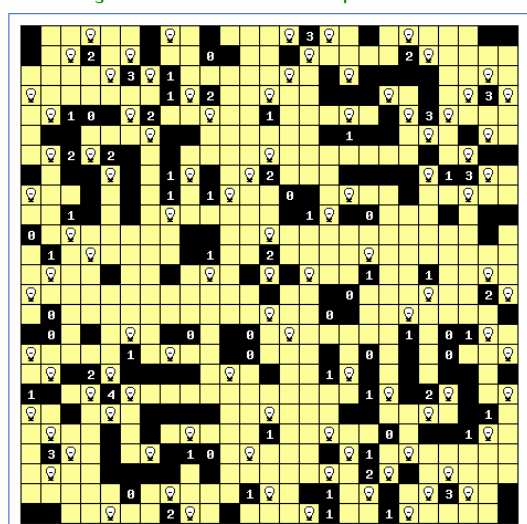
Ready Start Over Print New Puzzle

(b)



25x25 Light Up Easy Puzzle ID: 9,993,036

Ready Start Over Print New Puzzle



Congratulations! You have solved the puzzle in 33:23.23

25x25 Light Up Easy Puzzle ID: 9,993,036

Ready Start Over Print New Puzzle

圖 6. 25x25 Light Up 盤面之求解步驟[11]

第三節 文獻探討

目前關於這個遊戲的研究論文與相關文獻並不多，就我們所知，僅有一位學者 Brandon McPhail 在 2005 年時證明了「給定一個 Light Up 的盤面判定此盤面是否存在合理的解？」的難度是 NP-complete，其使用方法為將電路滿足問題 (Circuit-SAT) 在 polynomial time 內轉換成 Light Up 問題，成功地證明出 Light Up 是 NP-complete 問題[4]。

第四節 論文組織

本論文首先在第一章中介紹關於計算理論的基礎，以及論文中需要應用之基礎觀念；第二章中為關於本論文中研究主題之背景介紹、遊戲規則介紹、以及相關文獻參考；第三章是利用一個 deterministic finite automaton(DFA)解決 $1 \times n$ LCP；第四章是利用一個 nondeterministic finite automaton(NFA)解決 $2 \times n$ LCP 及討論其一致性問題；第五章為結論以及未來的研究方向。