

國立臺灣大學公共衛生學院流行病學與預防醫學研究所



博士論文

Institute of Epidemiology and Preventive Medicine

College of Public Health

National Taiwan University

Doctoral Dissertation

在元件網絡統合分析中重建斷裂網絡之條件及證據結構

視覺化之方法

The conditions for reconnecting disconnected networks and
the method for visualizing the evidence structure in
component network meta-analysis

李 驊

Hua Li

指導教授：杜裕康 博士

Advisor: Yu-Kang Tu, DDS, Ph.D.

中華民國 112 年 3 月

March 2023



國立臺灣大學博士學位論文
口試委員會審定書

在元件網絡統合分析中重建斷裂網絡之條件及
證據結構視覺化之方法

The conditions for reconnecting disconnected networks
and the method for visualizing the evidence structure in
component network meta-analysis

本論文係 **李驊** 君(學號 D07849009)在國立臺灣大學流行病學與預防醫學研究所完成之博士學位論文，於民國 112 年 3 月 1 日承下列考試委員審查通過及口試及格，特此證明。

口試委員：

杜祜康

(簽名)

(指導教授)

張淑惠

陳錦華

李文宗

蕭志立



誌謝

首先我想感謝我的家人，特別是我父親，在我在決定攻讀博士班之後一路支持我讀完，家人的支持讓我不用煩惱太多個人因素，能夠專心做研究；再來，謝謝朋友在我心情不好的時候聽我的抱怨、陪我散散心、吃吃下午茶，讓我打起精神來。

接著自然是要感謝指導教授杜裕康老師引導學生找到研究方向，引導學生更為深入的鑽研研究題目。也感謝杜老師研究室畢業的施銘傑博士，在百忙之中抽空與學生討論，並適時給出良好的建議及評價。同時也感謝在中途報告研究成果時，給學生提出建議及問題的諸位師長、同學及學者們。



摘要

網絡統合分析是用於比較多種治療相對效果的工具。最常使用的網絡統合分析模型（Lu & Ades 模型）無法用於分析網絡斷裂的資料。儘管陸續有不同的解決方式被提出，但他們都有各自的缺點。而元件網絡統合分析便是其中一種潛在的分析斷裂網絡的方式。

元件網絡統合分析是基於相加性假設來分析多種組合式治療的元件效果。近年有文獻指出，元件網絡統合分析有可能使斷裂的網絡重新建立起連結，其必要條件是斷裂網絡中的子網絡之間要有共同元件；然而這並非唯一的必要條件，即使滿足前述的條件也不能保證網絡一定能透過元件網絡統合分析重建。因此，本研究的第一個目標是找齊所有的必要條件。我們證明要反應證據結構的 M 矩陣的秩等於元件數才可使所有的元件皆可估計。當斷裂網絡滿足這兩個條件，即可使用元件網絡分析進行重建。

接著，我們藉由發展一種視覺化證據結構的繪圖方式，來研究如何檢查連結不同子網路欠缺何種證據。然而標準的網絡圖採用治療當作結點，無法很好的呈現出元件網絡統合分析的證據結構，尤其是資料一複雜就難以判讀。因此，本研究的第二個目標是使用線性信號流圖以元件為結點呈現它們彼此的關係。新的繪圖方式能清楚地呈現元件本身的證據分布情況。由於是根據標準網絡圖轉換而來，我們也發現標準網絡圖其實可視作我們新圖的特例。而若資料複雜度較高使圖片變得難以判讀，還可以在繪圖前採高斯消去法簡化圖上的訊息，以輔助使用者解讀證據分布之情況。

關鍵字：元件網絡統合分析、線性信號流圖、高斯消去法、視覺化、證據結構



Abstract

Network meta-analysis (NMA) is used to compare the relative effects of multiple treatments. The most popular network meta-analysis model, namely the Lu & Ades model, cannot analyze a disconnected network. Several solutions have been proposed, but they all have their own disadvantages. One potential solution to analyzing a disconnected network is component network meta-analysis (CNMA).

CNMA estimates the effect of each component within multicomponent interventions under the assumption of additive component effects. A recent article pointed out that CNMA could reconnect disconnected networks when distinct subnetworks share common components. However, this condition itself is insufficient for connecting broken networks. Our first objective, therefore, is to find the necessary conditions. We proved that the rank of the design matrix \mathbf{M} needs to be equal to the number of components for identifying the effects of components. A disconnected network will be reconnected as it these two conditions are satisfied.

We, then, investigated how to identify the missing evidence required to connect distinct subnetworks by developing a graphical tool for visualizing the evidence structure. The standard network plot uses treatments as its nodes, which is not suitable for visualizing the evidence structure in CNMA, particularly when the network plot is complicated. Therefore, our second objective is to propose the use of the linear signal-flow graph, which illustrates the relations between components as connections between nodes. Our proposed graph provides clear information about the relations among components within the networks of evidence, and it can be shown that the standard network plot is a special case of our proposed graph. Moreover, we can use gaussian

elimination to extract information within our graph to help the user elaborate the evidence structure.



Keywords: component network meta-analysis, linear signal-flow graph, gaussian elimination, visualization, evidence structure

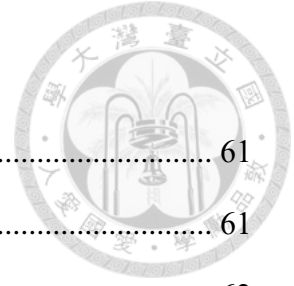


目錄

口試委員審定書	i
誌謝	ii
摘要	iii
Abstract.....	iv
目錄	vi
圖片目錄	ix
第 1 章 研究背景	1
第 2 章 文獻回顧	5
2.1 Lu & Ades 模型.....	5
2.1.1 無法用於斷裂網絡的原因	7
2.2 處理斷裂網絡的方法	7
2.2.1 加入觀察性研究	7
2.2.2 將斷裂的網絡做個別分析	8
2.2.3 其他模型：Baseline 模型	8
2.2.4 其他模型：Arm-based 模型	10
2.2.5 其他模型：人口調整方法	11
2.3 元件網絡統合分析	12
2.3.1 如何連結斷裂網絡	13
2.3.2 共同元件	14
2.3.3 網絡圖之於元件網絡統合分析	14
2.4 結構方程模型的路徑圖	16
2.5 研究目的	19



第 3 章	方法	20
3.1	整理資料結構	20
3.1.1	範例	21
3.2	是否能夠連結	21
3.3	交互作用項	22
3.3.1	範例	23
3.4	視覺化 M 矩陣之訊息	24
3.4.1	線性信號流圖	24
3.4.2	原始 M 矩陣之訊息	26
3.4.3	透過高斯消去法簡化之訊息	30
3.4.4	交互作用項	33
第 4 章	實際範例	36
第 5 章	結果	39
5.1	資料一：無法透過元件網絡統合分析連通之網絡	39
5.2	資料二：可透過元件網絡統合分析重建之網絡	43
第 6 章	討論	48
6.1	利用元件網絡統合分析連接斷裂網絡的條件	48
6.2	信號流圖跟結構方程模式路徑圖之間的差異	49
6.3	標準網絡圖可視為信號流圖的一個特例	50
6.4	信號流圖之延伸應用	51
6.5	未來研究方向	52
參考文獻	53
附錄	61
Part A:	R code for drawing a signal-flow graph	61



1. Packages	61
2. Function: Rearrange a matrix	61
3. Function: Gaussian Elimination	62
4. Function: Draw a signal-flow graph.....	66
5. Load the data of Example 1	73
6. Obtain and visualize the M matrix	74
7. How to add interaction	75
Part B: Dataset for Example 1	79
Part C: Dataset for Example 2	80



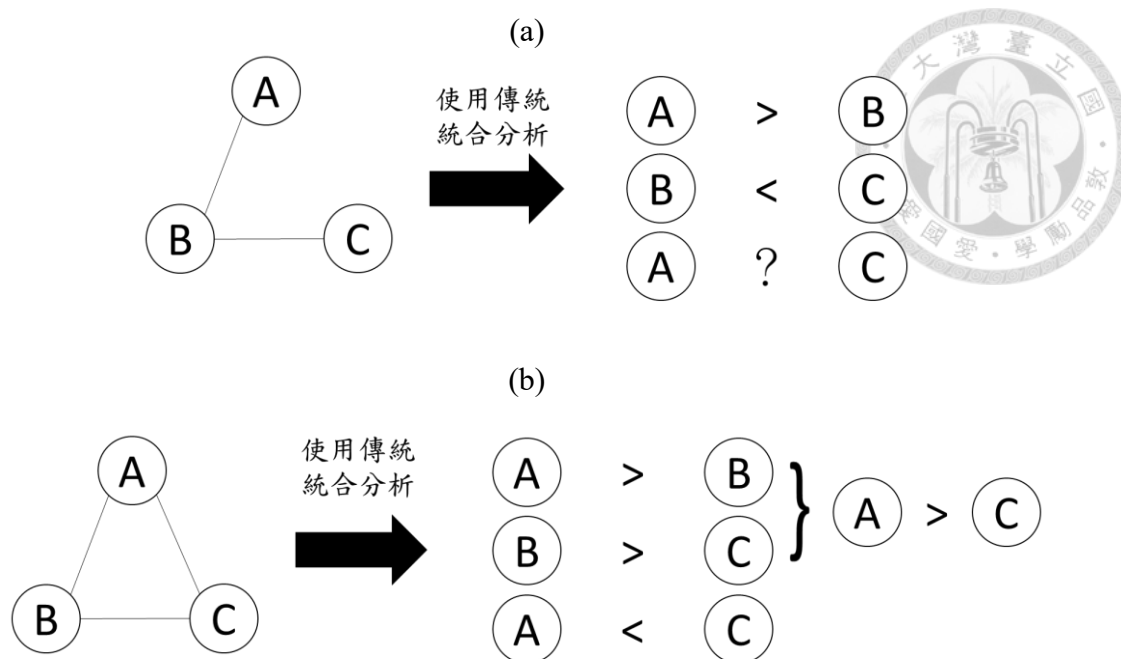
圖片目錄

圖一 使用傳統統合分析的缺點	2
圖二 連通與斷裂之網絡	6
圖三 三種模型潛藏的網絡結構示意圖	10
圖四 以網絡圖呈現元件網絡統合分析之證據分布	13
圖五 元件網絡統合分析的網絡圖不易解讀之處	15
圖六 結構方程模型的路徑圖	17
圖七 線性信號流圖範例	26
圖八 從網絡圖到信號流圖	29
圖九 經簡化之矩陣的線性信號流圖	32
圖十 加入 BC 交互作用項之線性信號流圖	35
圖十一 資料集一之網絡圖	37
圖十二 資料集二之網絡圖	38
圖十三 資料一之結果	41
圖十四 資料一之結果經高斯消去法	42
圖十五 資料二之結果	45
圖十六 資料二之結果經高斯消去法	46
圖十七 資料二重建之網絡	47
圖十八 示意圖	51

第1章 研究背景

實證醫學(Evidence-based Medicine)廣泛應用在臨床照護、醫學教育，以及衛生政策的擬訂等部分。近年資料的產出越發快速，要如以前那般整合手上的資訊變得較為不易；而且，針對同一議題不同文獻或報告會提出不同，甚至完全相反的見解，使臨床決策上遇到困難。因此，現今越來越講究實證醫學。在證據分級系統中，系統性回顧與統合分析具最高等級。透過明確的問題進行全面的文獻搜尋，經過嚴謹的評讀，篩選出值得信賴的研究，再運用統合分析去整合研究結果，幫助研究者決策治療的優劣，使醫療資源能夠有較有效的運用¹。

統合分析(Meta-analysis)是實證醫學中，用以整合不同研究中成對比較治療效果的工具²。傳統統合分析主要針對兩個治療比較兩種治療效果之差異，透過把搜尋篩選而來的每篇研究進行加權平均而整合出結果。然而，隨著越來越多新穎藥物和治療方式的開發，傳統統合分析受限於只能比較兩種治療，無法同步進行超過兩種治療的分析³⁻⁵。例如，若要比較三種治療 A、B、C 的治療效果，需要分別進行三種治療的三組成對比較 (A 比 B、A 比 C、B 比 C) 的統合分析。這其中可能會產生三個問題。首先，並非三種成對比較都有研究進行過直接比較，當缺乏其中一組成對比較時，就沒辦法得到完整的結論 (參見圖一(a))。其次，即使三組成對比較都具有直接證據(Direct evidence)，也可能得到 A 優於 B、B 優於 C、A 劣於 C 的結果，從前面兩組結果來看，A 理應優於 C，但 A 和 C 治療的直接證據卻呈現 A 劣於 C (參見圖一(b))。這個結果很難去下結論，因為證據的可靠度是由研究品質的好壞來決定的，間接證據(Indirect evidence)的可靠程度未必弱於直接證據。第三，若某篇研究同時比較了三種治療，還需要處理三組成對比較彼此之間的相關性問題。因此，為了解決上述問題，延伸傳統統合分析的架構發展出了網絡統合分析(Network meta-analysis)⁶。

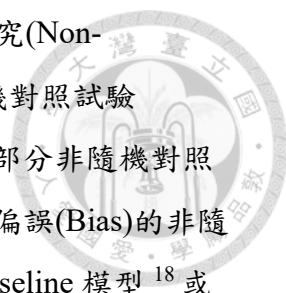


圖一 使用傳統統合分析的缺點

(a) 缺乏 AC 成對比較；(b) 遞疑律不成立。

網絡統合分析克服傳統統合分析限制，能用於比較兩種以上的治療，並同步整合多種治療效果⁷⁻¹¹，因此能涵蓋到更多臨床實務上所面對的問題。同步比較多種治療的想法由 Lumley 於 2002 年提出¹⁰，隨後於 2004 年，Lu 和 Ades 提出目前最廣為使用的、以貝氏統計為架構的模型，並拓展應用到各種類型的資料來進行網絡統合分析¹¹。此模型在傳統統合分析的基礎上，透過一致性 (Consistency) 假設使直接證據與間接證據能同步結合，在處理傳統統合分析問題的同時也能夠加強證據，給予研究者一個綜合性的比較結果，以利研究者評估各個治療療效。

網絡統合分析中牽涉多個治療，因此在呈現資料分布的時候，會以治療為節點繪製網絡圖，兩節點之間的線條代表資料中包含該組成對治療的證據^{12,13}。如圖一所示，(a)和(b)的網絡中皆包含 A、B 和 C 三種治療，而(b)比(a)多了 A 和 C 的成對比較。當網絡中任意兩點以線條直接（如圖一(a)的 A 和 B）或經由其他節點間接（如圖一(a)的 A 和 C）連結時，稱其為連通網絡；反之，則稱其為斷裂網絡。網絡統合分析中，最廣為使用的 Lu 和 Ades 模型使用前必須確保網絡是連通的¹¹。研究者搜集來的資料無法保證網絡必定是連通狀態，當網絡斷裂時，就需要透過其他方式才能使資料得以分析。



要分析斷裂網絡有三種方式。第一種是納入非隨機對照研究(Non-randomized study)。一般分析納入的都會是證據等級較高的隨機對照試驗(Randomized controlled trial)，但當網絡斷裂時，可以藉由納入部分非隨機對照研究來連結斷裂網絡¹⁴。優點是做法簡單；缺點是沒有控制好偏誤(Bias)的非隨機對照研究可能會影響到分析的有效性¹⁵⁻¹⁷。第二種是使用 Baseline 模型¹⁸ 或 Arm-based 模型¹⁹ 來執行網絡統合分析。常用的 Lu 和 Ades 模型是屬於 contrast-based 方法，此方法是基於隨機對照試驗的概念來做模型假設，主要是以相對效果來進行；而 Baseline 模型 (Arm-based 模型) 則會額外針對絕對治療效果做假設，使模型分析時能部分 (完全) 以絕對效果來進行。優點是不需要額外納入容易引入偏誤的非隨機對照研究；缺點是額外的假設會破壞隨機對照試驗的概念，故此方法頗受爭議¹⁸⁻²⁰。第三種是人口調整方法(Population-adjusted methods)^{14,21}。此種方法是借用某一個試驗的個別受試者資料(Individual participant data)來模擬出另一個試驗沒有納入的治療之效果。優點是不會碰到前兩者的問題；缺點在於個別受試者資料不易取得，而且即使有個別受試者資料，模擬出來的效果有無偏誤仍是未知的²²。

除了上述方法外，元件網絡統合分析(Component network meta-analysis)也不失為一種連接斷裂網絡的方法²³。為因應組合式治療的分析，元件網絡統合分析也隨之被提出²⁴。一些臨床治療中，會將個別不同的治療方式組合成新的治療方式。例如愛滋病毒治療中俗稱「雞尾酒療法」的高效能抗愛滋病毒治療，是組合至少三種抗愛滋病毒藥物，以有效控制愛滋病毒的病毒量與傳染風險的治療方式。一般的網絡統合分析中，會將不同組合的治療視為一個獨立治療，也就是說 A、B、A+B 和 B+C 被視為四種不同的治療方式，但這無法回答單獨使用 C，以及其他以 A、B、C 三種治療為元件的組合式治療 (例如 A+C 治療) 的效果是多少。為了回答這個問題，Welton 於 2009 年提出了元件網絡統合分析的模型，透過假設每個元件的治療效果具有相加性(Additivity)，使模型可以直接估計每個元件的治療效果²⁴。例如 A+C 的治療效果為 A 的治療效果加上 C 的治療效果；B+C 的治療效果為 B 的治療效果加上 C 的治療效果。不過最近有研究指出，元件網絡統合分析也是分析斷裂網絡的一種方法^{23,25}。

元件網絡統合分析有分析斷裂網絡的潛力，但並非所有情況都適用。由於元件網絡統合分析一開始是為了分析組合式的治療中每個元件的治療效果²⁴，並非是為了重建斷裂網絡才提出來的，所以早期並未有人將其用於這方面。是後續在一篇 2016 年的文獻中暗示了將其用於斷裂網絡的想法²⁴，留意到這點的 Rucker 以一個簡單的例子，展示了元件網絡統合分析確實可以透過共用的元件重新連結斷裂的網絡（細節於 2.3.1 小節進一步說明）²⁵。然而，這個條件並不適用於所有斷裂網絡，即使有共同的元件也不代表所有的元件都可估計，過去的研究也並未針對其餘的條件做進一步探討。因此，我們要釐清網絡要尚需具備何種條件，才能夠使用元件網絡統合分析應用於斷裂網絡。執行網絡統合分析時，碰到網路斷裂的情況並不罕見。雖然已經有一些克服的方法，但每種方法都有各自的侷限性。若能夠開拓其他的方法，將有助於增進元件網絡統合分析的應用範圍，並且使我們在面對斷裂網絡使有更多的選擇。

此外，為呈現元件網絡統合分析實際的證據分布情況，我們將提出新的繪製網絡圖的方式。元件網絡統合分析是以元件為單位運作，然而現在使用的網絡圖應用在元件網絡統合分析上，仍舊是以原本的組合式治療作為節點來繪圖。該圖無法有效呈現以元件為單位的證據分布情況，對於元件網絡統合分析並無太大的意義。因此，我們希望能提出一個以元件為節點的繪圖方式，使研究者能夠更容易掌握證據的分布情況。

綜上所述，本篇研究要釐清斷裂的網絡要具備何種條件，才能夠使用元件網絡統合分析重新建立連結。重新建立連結也就意味著所有元件皆可估計，因此首先我們會從這個方向著手，再進一步討論要具備何種條件才能使全部元件皆可估計。接著，我們會使用前一階段研究過程中所整理出的訊息，透過線性流圖(Linear signal-flow graphs)的方式繪製出新的網絡圖。該圖用於補助研究者掌握元件網絡統合分析證據分布情況，並在部分元件無法估計的情況下找尋欠缺的證據。

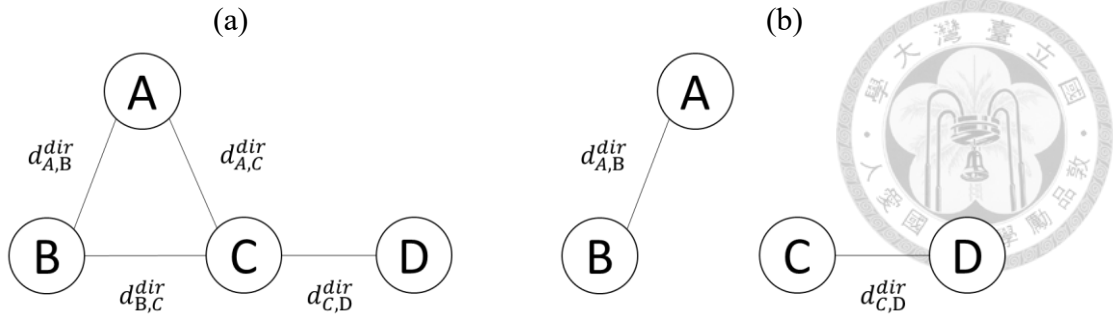
本論文的部分內容已投稿至 Research Synthesis Methods 且被接受。該篇文章作者、題目和 DOI 如下：Hua Li, Ming-Chieh Shih, Yu-Kang Tu. “Graphical evaluation of evidence structure within a component network meta-analysis”.
<https://doi.org/10.1002/jrsm.1623>。

第2章 文獻回顧

本章的 2.1 小節會先介紹最常使用的 Lu & Ades 模型，並說明它無法用於斷裂網絡的原因。2.2 小節說明其他可用於連結斷裂網絡的模型，以及可以連結的原因。2.3 小節介紹元件網絡統合分析，並說明能夠連結網絡的原因。2.4 小節延伸探討重建網絡圖的方式。最後 2.5 小節則重新整理本篇研究之目的。

2.1 Lu & Ades 模型

Lu & Ades 模型是拓展傳統統合分析的假設建立而成¹¹。傳統統合分析模型假設在不同試驗下，成對比較的絕對治療效果會有所差異，但相對治療效果相同，也就是說相對治療效果在不同試驗下具有可交換性(Exchangeability)，其差異皆由隨機抽樣所造成。以比較 A 和 B 治療為例，第 i 個試驗的 A 治療效果($\mu_{i,A}$)和 B 治療效果($\mu_{i,B}$)在不同試驗下會有差異，基於隨機對照試驗的設計概念，可以將同一試驗內的 A 和 B 治療效果視為獨立來計算各個試驗中 A 相對於 B 治療效果($d_{i,A,B} = \mu_{i,B} - \mu_{i,A}$)；而不同試驗的相對治療效果具有可交換性，所以對所有的 i (研究) 來說 $d_{A,B} = d_{i,A,B}$ ，差異由隨機抽樣造成，因此可以將 $d_{i,A,B}$ 加權平均而估計出 $\hat{d}_{A,B}$ 。每個試驗的權重由各自的 A 治療和 B 治療的標準誤(Standard error) $se_{i,A}$ 和 $se_{i,B}$ 決定，為 $\frac{1}{se_{i,A}^2 + se_{i,B}^2}$ 。Lu 和 Ades 延伸此概念加入一致性假設，令 $d_{B,C} = d_{A,C} - d_{A,B}$ 成立，從而使估計成對比較的治療效果時，除了成對比較本身的直接證據($d_{B,C}^{dir}$)以外，也能夠借用其他成對比較形成的間接證據($d_{B,C}^{ind} = d_{A,C}^{dir} - d_{A,B}^{dir}$)，並整合兩者以估計出 $\hat{d}_{B,C}$ (參見圖二 (a))。這避免了傳統統合分析中， $d_{A,B}$ 、 $d_{A,C}$ 和 $d_{B,C}$ 各自估計結果可能不一致(Inconsistent)，或由於缺少直接證據而無法估計的問題¹¹。



圖二 連通與斷裂之網絡

(a)連通網絡；(b)斷裂網絡。其中 d_{k_1, k_2}^{dir} 為 k_1 和 k_2 治療的治療效果差異。

因為是以相對治療效果進行估計，需要指定一個治療作為參照，因此在納入的治療中，指定 A 治療為整個資料的參考治療(Reference treatment)。在一致性假設下，需要估計的參數只剩下其他治療相對於 A 治療效果的參數($d_{A,k}$)，所以也將他們稱為基礎參數(Basic parameter)。由於並非所有成對比較都包含 A 治療，因此定義 $b(i)$ 治療為第 i 個試驗的基底治療(Baseline treatment)，也就是在缺少 A 治療時，該試驗中被選來當參照的治療。不包含指定的參考治療 (A 治療) 的其他相對治療效果的參數則稱為功能參數(Functional parameter, $d_{b(i),k}$ 且 $b(i)$ 和 k 不等於A)，皆可透過基礎參數線性組合而得。綜上所述，Lu & Ades 模型可以寫成¹¹：

$$y_{i,k} \sim N(\theta_{i,k}, se_{i,k})$$

$$\theta_{i,k} = \beta_{b(i)} + U_{i,k} \delta_{i-b(i),k}$$

$$U_{i,k} = \begin{cases} 1, & k \neq b(i) \\ 0, & k = b(i) \end{cases}$$

$$\delta_{i-b(i),k} \sim N(d_{b(i),k}, \tau^2), d_{b(i),k} = d_{A,k} - d_{A,b(i)}, d_{A,A} = 0$$

$y_{i,k}$ 是第 i 個試驗中 k 治療的治療效果； $\theta_{i,k}$ 是線性轉換過的第 i 個試驗中 k 治療的治療效果； $\beta_{b(i)}$ 為第 i 個試驗中 $b(i)$ 治療療效果； $\delta_{i-b(i),k}$ 是第 i 個試驗中 k 治療相對於 $b(i)$ 治療的治療效果； $d_{b(i),k}$ 為 k 治療相對於 $b(i)$ 治療的平均治療效果，且 τ^2 為其變異數； $\varepsilon_{i,k}$ 為第 i 個試驗中 k 治療的組內變異，其服從平均值為0且變異數為第 i 個試驗中 k 治療的標準誤平方的常態分布。



2.1.1 無法用於斷裂網絡的原因

基於一致性假設所形成的模型，無法應用在斷裂的網絡上²⁶。網絡統合分析中，將每個治療視為不同的節點，納入的試驗會根據其比較的治療將節點彼此相連而形成一個網絡圖，它類似於一般的敘述性統計，能夠呈現資料是由哪些治療的成對比較所組成²⁷。從網絡圖可以觀察納入的資料是否能形成一個連通的網絡，若某些治療節點彼此之間沒有任何直接，或透過其他節點間接形成的連線使他們連通，則可以判斷它是斷裂網絡。從前面敘述的比較 A、B 和 C 三種治療的例子，以及模型本身的結構可以看到，使用 Lu & Ades 模型時，要估計 $d_{B,C}$ 至少要有比較 B 和 C 成對治療的直接證據($d_{B,C}^{dir}$)或其間接證據($d_{B,C}^{ind}$)其中一個。網絡圖中節點對節點的連線為直接證據，透過其他節點間接形成的連線則為間接證據，因此，若網絡斷裂即可判斷 Lu & Ades 模型無法分析該組資料。如圖二(a)的例子中， $d_{B,C}^{dir}$ 和 $d_{B,C}^{ind}$ ($= d_{A,C}^{dir} - d_{A,B}^{dir}$) 都存在；而圖二(b)的例子，兩者都不存在。

2.2 處理斷裂網絡的方法

最常使用的 Lu & Ades 模型按標準的方式做無法處理斷裂網絡，因此應運而生了幾種處理斷裂網絡的方式。

2.2.1 加入觀察性研究

統合分析是基於隨機分派試驗所建構的方法。隨機分派試驗的優點是在研究因果關係的時候，能調整會對結果造成影響的潛在干擾因子，使偏誤盡可能降低。因此，在我們根據特定條件——包含欲研究的族群、治療種類、評估結果的指標——來納入隨機分派試驗時，理論上納進來的試驗之相對效果為不偏估計(Unbiased Estimator)。統合分析由此便假設這些相對治療效果具備可交換性，也就是說，分別來自不同試驗的 A 和 B 的真正的相對效果應該是一樣的，其差異是來自抽樣誤差。

然而當只納入隨機分派試驗會形成一個斷裂網絡時，可以適當放寬納入條件，讓觀察性研究也被納入研究當中。納入更多的研究不保證能使網絡連結起來，但確實是增加連結機會的一種手段。只要網絡連結起來，就能使用 Lu &

Ades 模型進行分析。不過應注意的是，納入的觀察性研究不具備隨機分派試驗的優點，故此種方法可能會引入一些潛在的干擾因子，導致結果存在一些偏差。

2.2.2 將斷裂的網絡做個別分析

在不考慮放寬納入條件且不考慮使用其他模型的情況下，唯一能做的就是將斷裂網絡中的每個子網絡分別以 Lu & Ades 模型做分析，也就是有幾個子網絡就建立幾個 Lu & Ades 模型。此方法的優點是不必額外納入研究；然而缺點就是只能評估位於同一子網絡的治療之間的相對效果，位於不同子網絡的治療之間的相對效果就無從評估了。

上述兩種方法是在不採用其他模型的情況下的處理方式，接下來將介紹改採其他模型的做法，以及每個模型各自的特色及缺點。

2.2.3 其他模型：Baseline 模型

Baseline 模型一開始提出，是由於 Lu & Ades 模型只能提供相對治療效果；然而，某些決策是需要考量到絕對治療效果的。例如說，Lu & Ades 模型可以告訴我們 A 的效果是 B 的兩倍，但是這個兩倍是治療成功率從 1% 變成 2% 還是從 30% 變成 60% 呢？而且實際接受治療時，尚需考量到兩種藥物的成本，不可能所有的病人都用得起最好的藥物。此時就需要衡量使用成本較高之藥物所換取到的進一步的療效，是否值得我們付出的相對應的成本。也就是說，決策者需要知道他所付出的成本換到的是 1% 還是 30%，才能做出較為有利的決策。

上述問題 Lu & Ades 模型沒有辦法回答，於是 Dias 等人根據成本效益分析 (Cost-benefit analysis) 對參考治療做了較強的假設。這個假設讓參考治療也具備可交換性，從而使參考治療的絕對治療效果在模型中服從一個常態分布。至於其餘假設則與 Lu & Ades 模型相同，因此兩者模型結構非常相似¹⁸。

與 Lu & Ades 模型相同，我們同樣指定 A 治療為整個網絡的參考治療。由於假設參考治療可交換，所有試驗的基底治療皆視為 A 治療，所以僅需使用基礎參數，也就是其他治療相對於 A 治療效果的參數($d_{A,k}$)，而不需要功能參數。綜上所述，Baseline 模型可以寫成¹⁸：



$$y_{i \cdot k} \sim N(\theta_{i \cdot k}, se_{i \cdot k})$$

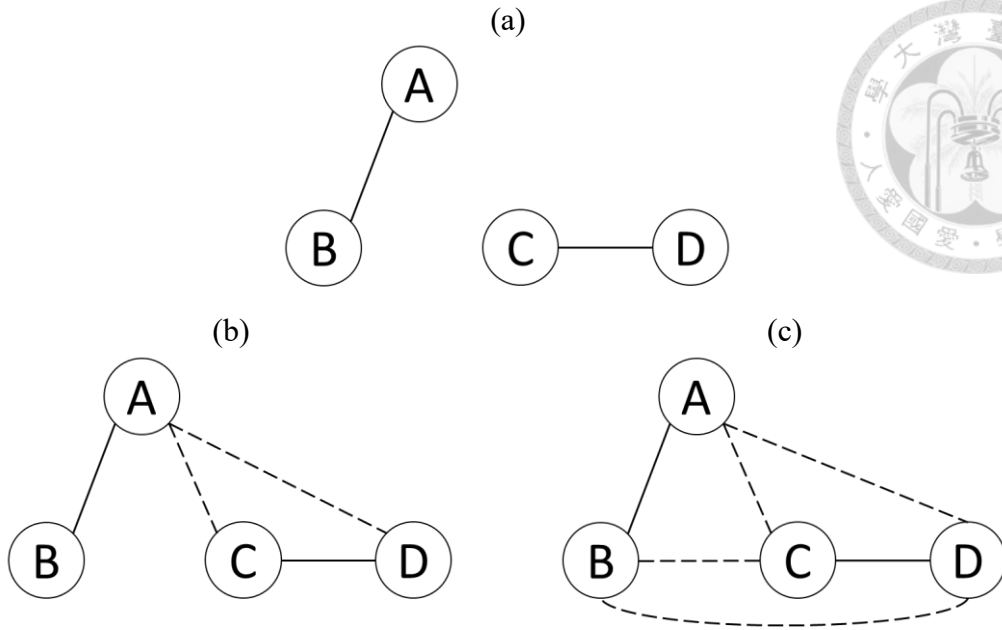
$$\theta_{i \cdot k} = \beta_i + U_{i \cdot k} \delta_{i \cdot A, k}$$

$$\beta_i \sim N(\mu_A, \sigma_A^2), U_{i \cdot k} = \begin{cases} 1, & k \neq A \\ 0, & k = A \end{cases}$$

$$\delta_{i \cdot A, k} \sim N(d_{A, k}, \tau^2), d_{A, A} = 0$$

其中 β_i 為第 i 個試驗中參考治療（A 治療）的治療效果，服從以 μ_A 為 A 治療的平均治療效果與 σ_A^2 為變異數的常態分佈。其餘變數的解釋則與 Lu & Ades 模型相同。

加入參考治療可交換的假設，某種意義上使斷裂的網絡得以重新連結。缺少參考治療的試驗能透過可交換性，借用其他試驗中的參考治療效果的訊息，因此能互相比較。這可以看成所有試驗都具備參考治療，只是部分試驗中的參考治療效果遺失了。這個概念在網絡結構中，等同於將所有治療與參考治療相連結起來。以圖三(a)的例子來說，就是比較 A 和 B 治療的試驗依舊只比較 A 和 B 治療；比較 C 和 D 治療的試驗是由於 A 治療的結果遺失了才只剩下二者，因此將 A 補回圖上便是圖三(b)所呈現之結果。



圖三 三種模型潛藏的網絡結構示意圖

(a) Lu & Ades 模型；(b) Baseline 模型；(c) Arm-based 模型。其中實線為資料具有的成對比較，虛線為透過模型重新連結的成對比較。

2.2.4 其他模型：Arm-based 模型

Arm-based 模型有別於傳統統合分析的概念，主要以絕對治療效果進行分析。它摒棄針對相對治療效果所做的假設，而是針對絕對治療效果，假設所有治療效果都具有可交換性，也就是說，納入的 k 治療的效果在不同試驗中應當相同，所有的誤差由隨機抽樣造成。綜上所述，Arm-based 模型可寫成^{9,19,28}：

$$y_{i \cdot k} \sim N(\theta_{i \cdot k}, se_{i \cdot k})$$

$$\theta_{i \cdot k} = \beta_{k(i)}$$

$$\begin{pmatrix} \beta_{A(i)} \\ \beta_{B(i)} \\ \vdots \\ \beta_{K(i)} \end{pmatrix} \sim MVN \left(\begin{pmatrix} \mu_A \\ \mu_B \\ \vdots \\ \mu_K \end{pmatrix}, \begin{pmatrix} \sigma_A^2 & \rho_{AB}\sigma_A\sigma_B & \dots & \rho_{AK}\sigma_A\sigma_K \\ \rho_{AB}\sigma_A\sigma_B & \sigma_B^2 & \dots & \rho_{BK}\sigma_B\sigma_K \\ \vdots & \vdots & \ddots & \vdots \\ \rho_{AK}\sigma_A\sigma_K & \rho_{BK}\sigma_B\sigma_K & \dots & \sigma_K^2 \end{pmatrix} \right)$$

其中 $\beta_{k(i)}$ 是第 i 個試驗中 k 治療的治療效果； μ_k 是 k 治療的平均治療效果； σ_k^2 是 k 治療的變異程度； $\rho_{k_1 k_2}$ 為描述試驗內部 k_1 和 k_2 治療的相關程度。

由於直接針對絕對治療效果去建構模型，網絡不受原本結構影響。在此假設下，理論上所有試驗都包含所有治療，若某個試驗缺少 k 治療，則可以借用其

他試驗中的 k 治療的訊息^{9,19,28}，這使得所有治療可以直接相互比較。以網絡結構來看，這等同於將所有治療兩兩相連起來。同樣以圖三(a)為例，該網絡納入了 ABCD 四種治療，因此比較 A 和 B 治療的試驗遺失了 C 和 D 治療的結果；比較 C 和 D 治療的試驗遺失了 A 和 B 治療的結果，分別將各自遺失的治療都補回圖上，便是圖三(c)所呈現之結果。總得來說，不論原本資料的網絡結構長得如何，以 Arm-based 模型的角度來看都會是完整的網絡，因此也就不會有網絡斷裂的問題。

上述兩種模型都有針對絕對治療效果做假設，但這一假設會破壞原本基於隨機分派之實驗特性所建構的方法²⁰。隨機分派是在一個試驗中將受試者隨機分派到不同的治療組，此舉能盡可能將不同組別的人口分布具備相同的特徵，藉此來排除其他干擾變項的影響。然而當我們針對絕對治療去做假設時，這等同於假設，使用某種治療的受試者組別是被隨機分派到某個試驗中的。其中 Baseline 模型的假設意味著所有試驗都具備參考治療，沒觀察到是由於某些原因導致遺失了；而 Arm-based 模型的假設則進一步意味著，所有試驗都具備納入的所有治療，沒觀察到也是由於遺失的緣故²⁸。以現實的角度來看，這種假設自然是不合理的。受試者是進入某一試驗之後再被隨機分派到其中一組，而不是先進入某一組別後再被隨機分派到某個試驗中。此外，基於成本考量，也很少有試驗會同時比較被納入的所有治療，大多都是針對研究者關注的幾個治療來進行試驗。

2.2.5 其他模型：人口調整方法

Lu & Ades 模型和上述兩種模型，基本上採用的資料都是聚合資料 (Aggregate Data)，若部分試驗能取得個別受試者資料，亦可採用人口調整方法。該方法本是用於處理不同試驗的族群結構不同的問題上²²。即使設定納入條件，族群結構依舊可能會不同。譬如說，納入的條件是受試者要為成年人，但有的試驗可能族群年紀有的偏年輕，有的偏年老。當年紀會影響治療效果的時候，就有可能使結果產生偏誤。此時若能取得某些試驗個別受試者資料時，

就可以透過該資料去模擬出在另一個試驗的族群結構下，該試驗可能會呈現出的結果。

由於是模擬該試驗在另一個試驗的族群結構下的結果，因此該模擬結果可視作在另一個試驗下所得之結果。以圖三(a)為例，假如取得 AB 試驗的個別受試者資料，即可透過該資料模擬出在 CD 試驗的族群結構下 A 和 B 的治療效果，如此一來四項治療的效果便可直接在 CD 試驗下進行比較了，也就是拿模擬出來的 A 和 B 的結果與實際觀察到的 C 和 D 的結果一起進行比較。此種方法好處是，僅需部分試驗擁有個別受試者資料即可，而擁有個別受試者資料的試驗即使僅包含一個治療也可納入到研究中。缺點則是，由於是透過模擬產生其他試驗中的效果，無法保證其模擬結果的準確性；而且當族群結構差異過大時，是無法進行有效模擬的²²。以一個最極端的例子來說，使用只包含女性受試者的個別受試者資料，沒有辦法模擬出另一個同時包含男女受試者的試驗之結果，尤其性別是干擾因子時，這種模擬更是全無意義。

2.3 元件網絡統合分析

對於某些疾病，臨床試驗會採取組合式的治療方式。他們通常會由幾種基本且單一的治療所構成，而這些基本且單一治療即為元件網絡統合分析中所稱的元件²⁴。以圖四(a)為例，A、B、C 治療為該網絡的元件，其他的治療皆由這三個元件組合而成。不論是單一還是組合式的治療方式，在一般的網絡統合分析中都視為不同且單一的治療方式，模型不會考慮其中是否有哪些治療方式是其他治療的組合。

元件網絡統合分析中，模型基本假設與 Lu & Ades 模型相同，只是多了相加性假設。原本在 Lu & Ades 模型中的功能參數是以基礎參數的線性組合來表示，元件網絡統合分析中，則以代表元件治療效果之參數的線性組合來表示。當納入的治療皆由 t_1, \dots, t_j 元件所組成， k 治療使用的元件集合為 T_k ，則 $T_k \ni t_j$ 表示 k 治療有用到 t_j 元件。根據以上描述，元件網絡統合分析模型可寫成²⁴：

$$y_{i:k} \sim N(\theta_{i:k}, se_{i:k})$$



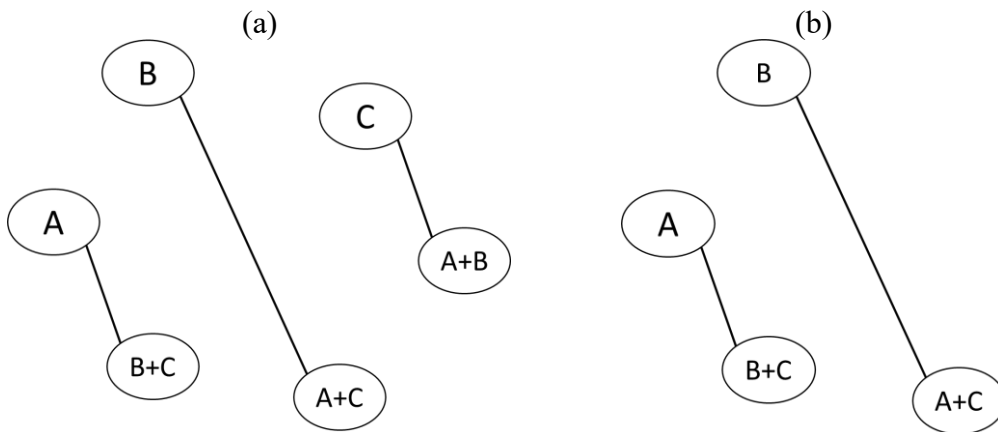
$$\theta_{i \cdot k} = \beta_{b(i)} + U_{i \cdot k} \delta_{i \cdot b(i), k}$$

$$U_{i \cdot k} = \begin{cases} 1, & k \neq b(i) \\ 0, & k = b(i) \end{cases}$$

$$\delta_{i \cdot b(i), k} \sim N(d_{b(i), k}, \tau^2), d_{b(i), k} = D_k - D_{b(i)}$$

$$D_k = \sum_{T_k \ni t_j} D_{t_j}, D_{b(i)} = \sum_{T_{b(i)} \ni t_j} D_{t_j}$$

其中 $\beta_{b(i)}$ 是第 i 個試驗中 $b(i)$ 治療的平均治療效果； D_k 為 k 治療相對於不做治療的效果； $D_{b(i)}$ 為 $b(i)$ 治療相對於不做治療的效果； D_{t_j} 為 t_j 元件相對於不做治療的效果，其餘參數意義則與 Lu & Ades 模型相同。



圖四 以網絡圖呈現元件網絡統合分析之證據分布

(a)可重新連結的斷裂網絡；(b)無法重新連結的斷裂網絡。

2.3.1 如何連結斷裂網絡

在元件網絡統合分析中，所有的成對治療效果都能使用元件效果進行線性組合而得到，因此能透過共同使用的元件建立連結²⁵。以圖四(a) Rucker 舉的亂裂成三個子網絡的例子為例²⁵，每個子網絡中都只有一個成對比較，分別為 A 比上 B+C、B 比上 A+C，以及 C 比上 A+B，全部總共六個治療，由三個元件 A、B 和 C 組成。在相加性的假設下，可以將三組成對比較，用元件治療效果的線性組合表現治療效果的差異：



$$\begin{cases} -D_A + D_B + D_C = d_{A,(B+C)} \\ D_A - D_B + D_C = d_{B,(A+C)} \\ D_A + D_B - D_C = d_{C,(A+B)} \end{cases}$$

其中 D_k 為 k 治療相對於不做治療的效果， d_{k_1,k_2} 為 k_1 和 k_2 的相對治療效果。而在一般網絡統合分析使用基礎參數的線性組合表現治療效果的差異：

$$\begin{cases} d_{A,(B+C)} = d_{A,(B+C)} \\ d_{A,B} + d_{A,(A+C)} = d_{B,(A+C)} \\ d_{A,C} + d_{A,(A+B)} = d_{C,(A+B)} \end{cases}$$

只有 3 組成對比較，卻有 5 個參數 ($d_{A,B}, d_{A,C}, d_{A,(A+B)}, d_{A,(A+C)}, d_{A,(B+C)}$)，這顯然不足以估計出所有的參數。元件網絡統合分析使用 3 個參數 (D_k) 線性組合成 d_{k_1,k_2} ，藉由估計 D_k 可推算出每個元件的治療效果和 d_{k_1,k_2} ，同時也能推算出一般網絡統合分析無法估計之參數，例如 $d_{A,B}$ 。

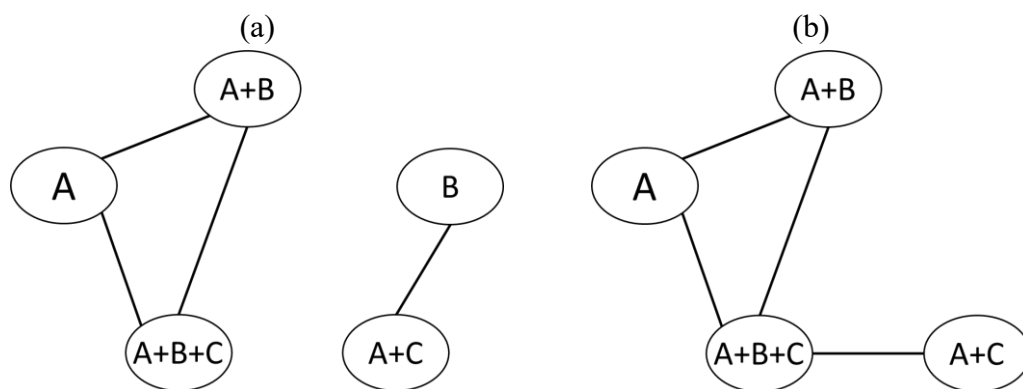
2.3.2 共同元件

在過去的文獻中，僅以圖四(a)的例子解釋元件網絡統合分析可連結斷裂的網絡。例子中的三個子網絡裡，都同時含有 A、B、C 這三種元件。但是這樣是否就表示這三個子網路，一定可以在元件網絡統合分析中相連起來？文獻中沒有說明要怎麼找出能使斷裂網絡重新連結在一起的元件，僅提及共同元件 (Common components) 可使它們連結在一起；且不同子網絡之間可透過不同共同元件連結²⁵。然而，並非所有組合式治療的斷裂網絡，都能透過元件網絡統合分析重新連結起來。以圖四(b)為例，兩個子網絡都有使用到 A、B、C 元件，但無法在元件網絡統合分析中連結兩個子網絡。由此可知資料還需要滿足其他條件才能使其重新連結。

2.3.3 網絡圖之於元件網絡統合分析

前面我們以網絡圖來呈現證據的結構，然而該圖無法很好的呈現元件網絡統合分析的證據結構。網絡圖是從常見的網絡統合分析的角度出發，以治療做為節點來呈現證據如何構成，因此無法有效呈現出以元件為單位的元件網絡統合分析的證據構成。甚至可能會造成誤判，讓人以為某些不可估計的元件是可

估的。這點即使是在網絡並未斷裂的圖上也一樣。以圖五的兩幅網絡圖來看，(a)圖的網絡斷裂，所以無法透過一般的網絡統合分析評估全部的治療效果，但可以藉由元件網絡統合分析評估每個元件的效果；而(b)圖的網絡並無斷裂，所以能夠直接使用一般的網絡統合分析得到節點代表的四個治療的效果，但它無法藉由元件網絡統合分析推估每個元件的效果。對元件網絡統合分析的模型不夠熟悉的話，乍一看(b)圖會感覺所有元件都是可估計的，然而仔細觀察所有的成對比較，可以發現他們都無法提供 A 元件的證據，因為在成對治療效果相消的過程中，A 元件的效果都會被消除。



圖五 元件網絡統合分析的網絡圖不易解讀之處

(a) 全部元件皆可估計之斷裂網絡；(b) A 元件不可估計之未斷裂網絡。

斷裂網路是否能夠重建，判斷依據除了須具備共用的元件以外，最重要的便是所有的元件效果是否都是可估計的²⁵。儘管常用的網絡圖花時間仔細推敲依舊能夠辨識是否全部的元件都能估計，但卻十分沒有效率。而且隨著治療或元件數量的增加，圖片的複雜度上升也會使判讀所需的時間越來越長。研究者除了需要一一看圖上的各組成對比較提供了那些元件的證據，整理完了還要再進一步解析才能得知。因此，我們嘗試以元件為節點繪製新的網絡圖，使圖上能清晰地呈現出元件的證據構成。以元件為節點就可改採成對比較所提供的元件訊息來呈現證據的組成及結構，藉此省去研究者一一檢視每組成對比較的過

程。同時，也能讓人判斷需要再額外納入哪種研究才能讓證據足夠估計全部的元件效果。

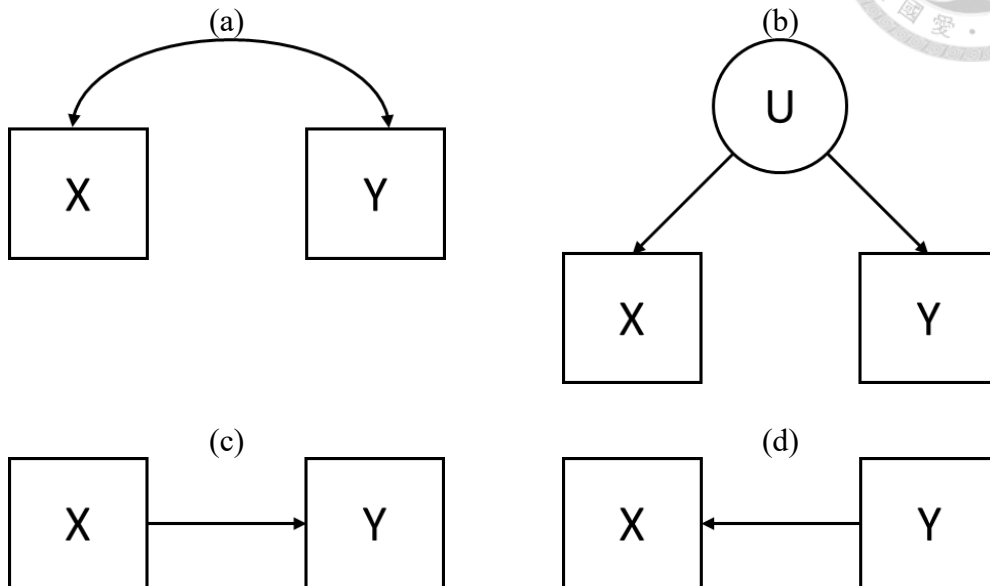
圖型改以元件為節點和以治療作為節點最大的不同點在於，每組成對比較提供的證據所涉及的節點數，會落入從 1 至元件總數的範圍內，而不僅僅是兩個。例如圖五(a)的網絡圖中，A+C 比上 B 治療的成對比較涉及兩個節點；然而若以元件為節點，該成對比較將涉及三個節點 A、B 和 C 元件。此外，伴隨著這個不同點所產生的另一個問題在於，不具方向性的線條已然不足以呈現元件之間的關係構成。原本網絡圖能用兩節點之間的線條，表示出資料中具有兩節點所代表之治療的直接證據，相當於指出該證據為哪兩個治療的相對治療效果。但是當涉及的節點數超過兩個時候，就會有三者之間直接證據是如何組合的問題。再次借用上述的例子來舉例，A+C 比上 B 治療的成對比較以元件為單位來繪圖的話，圖型上應清楚表達出該成對治療所貢獻的是 A 加上 C 減去 B 元件的效果。然而僅以線條來構成彼此的關係時，圖型描述的關係網定然會缺少方向性。因此，應當在線條中引入箭頭，使圖型能呈現關係裡的方向性。

2.4 結構方程模型的路徑圖

而說到由節點與箭頭所構成的圖，就不得不提結構方程模型裡的路徑圖^{29,30}。該圖源自有向無環圖(DAG, Directed Acyclic Graph)，其使用概念相當早便開始，凡舉祖譜、工作並行與否之排程圖等，具有不斷往前、不會後退，時而分岔、時而匯聚之特性的皆可視為 DAG。

結構方程模型裡的路徑圖構成包含節點與箭頭。用來代表變數的節點分為兩種類型，方形代表觀察變數，圓形則代表未觀察到變數（或稱潛在變數）。用來代表變數關係的箭頭也分為兩類型，雙箭頭表示無方向性的線性相關，雙箭頭則表示具有方向性的因果關係。假設 X 和 Y 為兩觀察變數，兩者之間的關係分為以下三類：一，當兩者具備線性關係且無方向性時，可用雙箭頭將代表兩者的方形節點連接（參見圖六(a)）。二，若此相關性是由於某個未觀察到變數 U 同時影響 X 和 Y 所造成的話，則會從代表 U 的圓形節點各別以單箭頭指向代

表 X 和 Y 的方形節點（參見圖六(b)）。三，若 X 和 Y 之間具有因果關係，則會由作為因的變數以單箭頭指向作為果的變數（參見圖六(c)和(d)）²⁹。



圖六 結構方程模型的路徑圖

(a) X 與 Y 具有線性相關；(b) X 和 Y 受到未觀察到變數 U 的影響；(c) X 會影響 Y；(d) Y 會影響 X。

路徑圖中不同的箭頭指向代表不同的涵義，即使模型在圖六(a)至(d)都可以估計出 X 和 Y 的關係，然而在沒有做實驗的情況下，僅憑統計結果無從得知何種關係才正確²⁹。若要逐一排查每種關係的話，首先能從 X 和 Y 彼此之間的因果關係檢查。當改變 X 而 Y 卻未受影響，表示 X 不是 Y 的因，可以將圖六(c)的情況排除；當改變 Y 而 X 卻未受影響，表示 Y 不是 X 的因，可以將圖六(d)的情況排除。若能觀察到變數 U 的話，就能夠檢查 U 是否能影響 X 和 Y 了。

儘管引入箭頭會讓人率先聯想到結構方程模型的路徑圖，但其箭頭的含意與我們設想中的新網絡圖的箭頭含意仍有差別，這使兩者產生了決定性的不同。路徑圖箭頭包含因果關係，不可隨意逆轉方向^{29,30}；然而在以元件為節點的網絡圖中，箭頭僅需反映元件證據構成的關係即可，這使得箭頭的方向在某

種程度上是可逆轉的。再次借用上述 A+C 比上 B 治療的成對比較來說明。在原本的網絡圖中之所以沒有箭頭，是因為它只需要透過線條表達出 A+C 治療和 B 治療之間有直接證據就好，至於是 A+C 治療減去 B 治療的相對效果，抑或是 B 治療減去 A+C 治療的相對效果並不是很重要。以元件為節點也是同樣的概念，儘管要引入箭頭，但圖上反映的是 A 加上 C 減去 B，抑或是 B 減去 A 再減去 C 都可以。也就是說，以元件為節點的網絡圖，僅需借用方向性的概念反映證據構成的關係即可。這使得一組成對比較將可能呈現出不同的圖型。

箭頭含意的差別更會使得兩邊最終產出的圖片有很大的差異。路徑圖上的兩個節點之間只會有唯一的一個連線（根據關係可能為單箭頭或雙箭頭）；與此相對，以元件為節點的網絡圖則可能不只一個。以元件為節點的網絡圖中，兩節點之間的一個箭頭最多就反映出一種成對比較的證據關係。當有多種成對比較所提供之證據皆涉及這兩個節點時，兩節點之間便會出現複數個箭頭，其中甚至可能會出現兩節點相互指向的情況。以圖五(a)的網絡圖來說明，A+C 和 B 的成對比較涉及 A、B 和 C 元件，而 A 和 A+B+C 的成對比較則涉及 B 和 C 元件。這兩者有可能同時在 B 和 C 節點之間分別產生出一個箭頭，使兩節點間具有兩個箭頭。總得來說，新的網絡圖只是要引入方向性的概念，雖然看似會成為有向圖，實則不然。

再來，二者在繪製箭頭和係數取得的方式上也有決定性的差異。路徑圖是根據因果關係先繪製出圖上的箭頭，再按照繪製出來的圖建立模型（方程式）估算出係數；信號流圖則是先有方程式（附帶著係數），再去琢磨箭頭該如何呈現。也就是說，前者是在建立好關係結構的情況下求關係的強度；後者則是在既定的關係強度下將其呈現出來。就繪圖目的而言，前者是要盡可能呈現「真實的關係」，對圖上的任何更動皆需理論支持；後者則單純著重於「呈現」本身，轉化成圖的過程中即使做了精簡的動作也無妨，畢竟方程式本身就是可以簡化的。

2.5 研究目的

綜上所述，本篇研究主要是探討如何利用元件網絡統合分析重建斷裂之網絡，而研究目的可分為三部分：

首先，探討欲滿足何種條件，元件網絡統合分析才可重建斷裂之網絡。由於主流的模型無法處理斷裂之網絡，且其餘處理斷裂網絡之方法各有其優缺點；而採用元件網絡統合分析，雖然缺點是僅適用於組合式治療的資料，但優點是能在不破壞核心假設（相對治療效果具備可交換性）的情況下重建斷裂網絡。然而並非所有情況下都能透過元件網絡統合分析重建網絡，除了子網絡間須具有共同元件以外，尚需滿足其他條件。因此本篇研究的第一個目的便是探討使其重建網絡之條件。

其次，為元件網絡統合分析提出新的繪圖方式。傳統網絡圖是將納入的每種治療單獨視作一個節點來呈現資料之結構，但套用到元件網絡統合分析就無法很明確地呈現元件分布之結構。因此，本研究將提出新的方法以元件本身為節點來呈現資料之結構。

最後，簡化圖形訊息以利判讀。我們的圖在資料複雜的情況下會有些瑣碎，箭頭也會較難閱讀，因此我們將借用高斯消去法簡化圖片上的訊息，以利研究者能夠以較為簡約的圖片得到與原本的圖同等的結果。



第3章 方法

模型本身還是如同 Lu & Ades 模型，建構在相對治療效果的架構下²⁴，要判別元件是否能重新連結兩個子網絡，取決於能否估出所有元件的治療效果。

3.1 整理資料結構

假定網絡中總共有 P 種成對比較、共 K 種不同的治療，所有的治療由 t_1, \dots, t_j 共 J 個元件組成。

首先，先建立一個矩陣以呈現每種成對比較是比較哪兩種治療。若第 p 個成對比較的兩個治療為 $k_{p,1}$ 和 $k_{p,2}$ ($k_{p,1}$ 不會等於 $k_{p,2}$)。以列和行分別代表成對比較及治療，則能作出 $P \times K$ 維度的 B 矩陣：

$$B = [b_{p,k}], b_{p,k} = \begin{cases} -1, & \text{if } k = k_{p,1} \\ 1, & \text{if } k = k_{p,2}, \\ 0, & \text{otherwise} \end{cases}$$

其中第 p 列代表第 p 個成對比較，第 k 行代表第 k 種治療。矩陣的每列只會有兩個非零的元素，1 和 -1 各一個。其含義代表該列對應的成對比較 p 提供的證據為 $k_{p,2}$ 治療減去 $k_{p,1}$ 治療之平均治療效果。

接著，建立一個矩陣以呈現每種治療是由何種元件所構成。令組成 k 治療之元件的集合為 T_k ，所以 $T_k \ni t_j$ 表示 k 治療中有用到 t_j 元件。以列和行分別代表治療及元件，則能作出 $K \times J$ 維度的 C 矩陣：

$$C = [c_{k,j}], c_{k,j} = \begin{cases} 1, & \text{if } T_k \ni t_j \\ 0, & \text{otherwise} \end{cases}$$

其中第 k 列代表 k 治療，第 j 行代表 t_j 元件。

再來，將 B 矩陣乘 C 矩陣以取得 M 矩陣。該矩陣表示了每個成對比較提供了何種元件的線性組合：

$$M = [m_{p,j}], m_{p,j} = \sum_k b_{p,k} c_{k,j}$$

其中第 p 列代表第 p 個成對比較，第 j 行代表 t_j 元件。

原則上要跳過建立**B**和**C**矩陣的步驟，直接建立出**M**矩陣也可以。先建立二者的好處是，**B**矩陣本身呈現的就是一般網絡統合分析的資料結構，能看出是否有治療的效果會在相消的過程中被抵銷；而**C**矩陣則是在加入交互作用項時，稍作調整就能得到新的**M**矩陣（方式請參考 3.3 小節），不論是手動調整還是寫成指令，都絕對比直接使用原本的**M**矩陣去調整還要來得容易。所以若不需參考、對比原本的資料結構，且已有現成指令能取得**M**矩陣或是加入交互作用項的**M**矩陣，略過二者的建立步驟也無所謂。

3.1.1 範例

以圖四(a)為例，資料有 $P = 3$ 種成對比較，比較的治療為 A 、 B 、 C 、 $A + B$ 、 $A + C$ 、 $B + C$ 共六種，皆由 A 、 B 、 C 三種元件組成。有三個子網絡，第一個子網絡比較 A 和 $B + C$ ；第二個子網絡比較 B 和 $A + C$ ；第三個子網絡比較 C 和 $A + B$ 。**B**、**C** 和 **M** 矩陣分別為：

$$B = \begin{matrix} & A & B & C & A+B & A+C & B+C \\ \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 \end{bmatrix} \end{matrix},$$

$$C = \begin{matrix} & A & B & C \\ \begin{matrix} A \\ B \\ C \\ A+B \\ A+C \\ B+C \end{matrix} & \begin{bmatrix} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 0 & 1 & 1 \end{bmatrix} \end{matrix}$$

$$M = \begin{matrix} & A & B & C \\ \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} -1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{bmatrix} \end{matrix}$$

3.2 是否能夠連結

要能夠使網絡連結，先決條件是能夠將 $D = (D_{t_1}, \dots, D_{t_j})'$ 估計出來。能夠估計出來代表有唯一解，估計不出來代表沒有唯一解，所以我們只需要在有解的前提下，檢查是否是唯一解。首先，計算 $M_{P \times J}$ 的秩(rank)：

$$\text{rank}(M) = r_M$$



根據線性代數的定理，若 $r_M = J$ ，則對任意 $\mathbf{r} \in \mathbb{R}^J$ ，方程組 $\mathbf{M} \cdot \mathbf{d} = \mathbf{r}$ 的 $\mathbf{D} = (D_{t_1}, \dots, D_{t_j})'$ 有解則唯一。 r_M 代表該方程組可估計的參數數量上限，也就是說，若要使斷裂網絡能連結，參數數量上限要等於元件數（滿足 $r_M = J$ ）才可使全部參數可估計。

3.3 交互作用項

有些藥物彼此之間可能會有拮抗或加成作用，此時需要將交互作用項納入考量²⁴。若要考慮某些元件的交互作用項，例如 t_{j_1} 和 t_{j_2} 的交互作用（令其為 $t_{j_1*j_2}$ ），相當於把 $t_{j_1*j_2}$ 也當作一個元件加入模型^{24,25}。為了加入 $t_{j_1*j_2}$ ，在定義 T_k 時要將交互作用也當作元素包含進去，所以若 $T_k \ni t_{j_1}$ 且 $T_k \ni t_{j_2}$ ，則 $T_k \ni t_{j_1*j_2}$ 。

\mathbf{B} 矩陣不需修改，只要在 \mathbf{C} 矩陣新增代表交互作用項的行即可。如果只考量 $t_{j_1*j_2}$ （為符號呈現方便，將其視作為 t_{j+1} 元件）一個交互作用項，只需作如下的修改：

$$\mathbf{C}_{K \times (J+1)} = [c_{k,j}], c_{k,j} = \begin{cases} 1, & \text{if } T_k \ni t_j \\ 0, & \text{otherwise} \end{cases}$$

其餘步驟皆相同。

交互作用項可依照研究者的假說自由加入，也可放入超過兩種治療的交互作用項²⁴。研究中全部可能的交互作用項可於一開始就當作元素定義進 T_k 中，後續分析可根據欲探討之交互作用項調整 \mathbf{C} 矩陣之行數。然而要注意加入 n 個交互作用項等於多 n 個參數要估計，必須重新用 3.2 小節的方法判斷是否足以重建網絡。同時，網絡重建的連線可能會與未放入交互作用項的有差別，只要參數不同都需要重新以 3.3 小節的規則繪製重建的網絡圖。

雖說交互作用項可根據假說加入，但對於資料而言要滿足三個條件才有意義。其一是納入的治療中，至少要有一個治療包含該交互作用項涉及的所有元件。以 t_{j_1} 和 t_{j_2} 的交互作用（ $t_{j_1*j_2}$ ）來舉例，若納入的研究中，沒有任何一個治療有合併使用 t_{j_1} 和 t_{j_2} 元件，會使得所有的 T_k 都不包含 $t_{j_1*j_2}$ 元素。即使在 \mathbf{C} 矩陣新增代表 $t_{j_1*j_2}$ 的一行，該行的元素也會全為 0。 \mathbf{B} 、 \mathbf{C} 矩陣相乘後，代表 $t_{j_1*j_2}$ 的

那行元素也會全為 0，也就是說，沒有任何成對比較能提供用以估計 $t_{j_1*j_2}$ 效果的證據。其二是與包含該交互作用項之治療成對的治療，至少要有一個不包含該交互作用項。也就是說合併使用 t_{j_1} 和 t_{j_2} 元件的治療不能全都成對出現，這會使得訊息成對相消之後，反而無法提供任何 $t_{j_1*j_2}$ 的訊息。其三是交互作用項中涉及的元件本身要可估計，才能將元件的主效果和交互作用項分開。也就是說要估計 $t_{j_1*j_2}$ 的話， t_{j_1} 和 t_{j_2} 元件也一定要可估計才行。因此在加入一個交互作用項之前，除了確認涉及該交互作用項的元件都可估計以外，也要確定納入的治療中，至少有一個治療有合併使用涉及該交互作用項的元件，且其訊息不會在成對相消的過程中被消去。

3.3.1 範例

延續 3.1.1 以圖四(a)為例。該例中包含 $A + B$ 治療，若想於研究中探討 A 元件和 B 元件的交互作用， B 矩陣和前面一樣為：

$$B = \begin{matrix} & A & B & C & A+B & A+C & B+C \\ \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} -1 & 0 & 0 & 0 & 0 & 1 \\ 0 & -1 & 0 & 0 & 1 & 0 \\ 0 & 0 & -1 & 1 & 0 & 0 \end{bmatrix} \end{matrix}$$

C 矩陣則新增代表 A 元件和 B 元件交互作用 ($A * B$) 的一行元素。治療中合併使用 A 元件和 B 元件的治療為 1；若沒有兩者皆無使用或僅用到其中一個元件則為 0：

$$C = \begin{matrix} & A & B & C & A * B \\ \begin{matrix} A \\ B \\ C \\ A+B \\ A+C \\ B+C \end{matrix} & \begin{bmatrix} 1 & 0 & 0 & 0 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 \\ 1 & 1 & 0 & 1 \\ 1 & 0 & 1 & 0 \\ 0 & 1 & 1 & 0 \end{bmatrix} \end{matrix}$$

最後將兩者相乘可以得到新的 M 矩陣，並計算出它的秩：

$$M = BC = \begin{matrix} & A & B & C & A * B \\ \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} -1 & 1 & 1 & 0 \\ 1 & -1 & 1 & 0 \\ 1 & 1 & -1 & 1 \end{bmatrix} \end{matrix}$$

$$\text{rank}(M) = 3$$

有四個參數 (D_A, D_B, D_C, D_{A+B}) 要估計，但秩卻只有 3，故可判定加入交互作用項後，證據不足以估計所有的參數。在本例中，秩最多就是 3，因此無論加入何種交互作用都會導致證據不足以估算所有參數。



3.4 視覺化 M 矩陣之訊息

即使能透過 M 矩陣的秩可知曉網絡是否能重建，可當結果表示無法重建時，就需進一步探討是哪些元件不可估。 M 矩陣是判斷網絡是否能夠透過元件網絡統合分析重建的依據，其中更是蘊含著以元件治療的角度來看證據組成的訊息，因此我們嘗試將其以類似一般網絡圖的方式視覺化。

一般網絡圖的視覺化是透過 B 矩陣來完成，該矩陣的特點是每列只會有兩個非零的元素。其中一個必為+1，另一個則必為-1，代表兩個治療效果相減。在視覺化時，只需要以每行所代表的治療當作圖形的節點，並將每列非零元素所對應的兩個節點以線相連即可。然而 M 矩陣一列的非零元素少則一個，多則全部皆非零，所以無法簡單以成對的方式連接兩個元件治療。即使一列剛好只有兩個非零元素，也有可能同為正號或負號，就算將兩者對應的元件治療相連，其含意也跟一般網絡圖藉由異號的+1和-1所連出來的不同，不能視作是差異。於是我們打算在圖中引入方向性的概念以克服這點。儘管提到具有方向性的圖，首先會聯想到結構方程模式的路徑圖，然而路徑圖中的方向的含意，與我們設想於網絡圖中呈現的相去甚遠。所以我們將 M 矩陣視為一組線性方程式，並引入了能與線性方程組互相轉換的線性信號流圖(Linear signal-flow graph)的概念。

接下來 3.4.1 會先介紹線性信號流圖的繪製方式，3.4.2 則介紹如何在線性信號流圖的基礎上將 M 矩陣視覺化，接著 3.4.3 會闡述簡化圖型線條的方法。

3.4.1 線性信號流圖

倘若有 L 條方程式包含 N 個未知變數，其中 y_j 為已知變數， x_j 為未知變數。每條線性方程式可表示成：



$$\sum_{k=1}^N c_{jk} x_k = y_j \quad \text{其中 } 1 \leq j \leq L$$

首先，對線性方程組進行如下處理：

$$0 = \sum_{k=1}^N c_{jk} x_k - y_j$$

$$x_j = \sum_{k=1}^N c_{jk} x_k - y_j + x_j$$

$$x_j = \sum_{k=1}^N (c_{jk} + \delta_{jk}) x_k - y_j, \quad \text{其中 } \delta_{jk} = \begin{cases} 0, & \text{if } j \neq k \\ 1, & \text{if } j = k \end{cases}$$

以處理完的方程組來作圖。線性信號流圖能夠與線性方程組互相轉換，其圖形由節點及箭頭組成，箭頭上會有加權的係數。節點代表線性方程組的變數 (x_j)，而加權的係數則是方程組中的係數 ($c_{jk} + \delta_{jk}$)，其中 δ_{jk} 是由於等號左右同加了一個 x_j ，因此係數會比原本加 1。信號只會依節點的方向，由一個節點 (x_k) 流到另一個節點 (單獨在左側的 x_j)。線性信號流圖中只能表達信號和係數相乘，以及數個信號的相加，不過已然足以表示線性方程組。當一信號延著箭頭從一個節點指向另一個節點時，此信號會乘以箭頭上的係數，若幾個箭頭指向同一個節點時，這幾個信號會相加；若要表示相減，則將對應係數調整為負數即可³¹。

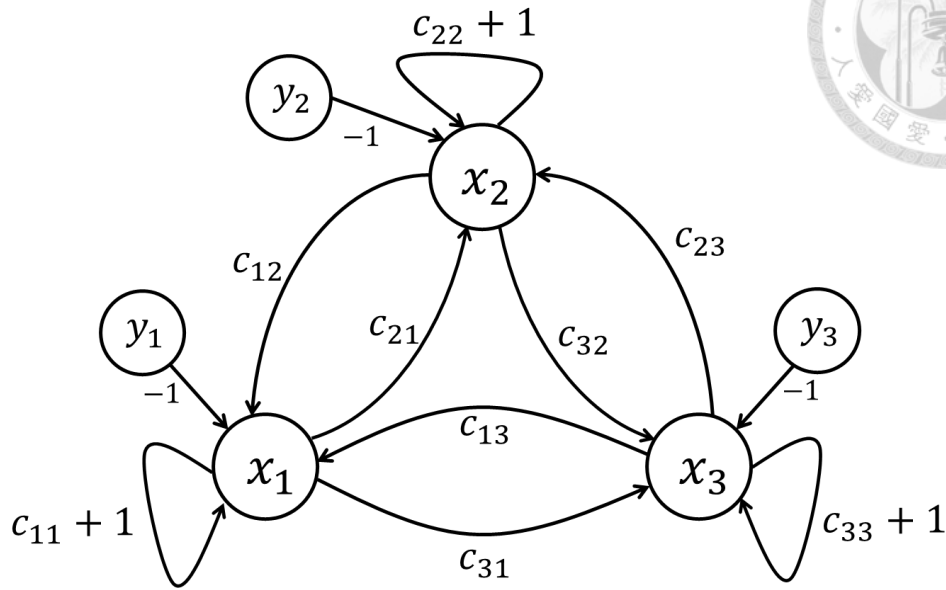
以一個簡單的例子來說明。如果現在有三條方程式包含三個未知變數，則方程組可表示為：

$$\begin{cases} c_{11}x_1 + c_{12}x_2 + c_{13}x_3 = y_1 \\ c_{21}x_1 + c_{22}x_2 + c_{23}x_3 = y_2 \\ c_{31}x_1 + c_{32}x_2 + c_{33}x_3 = y_3 \end{cases}$$

將全部變項移至左側，再交換等號左右兩側的內容，並於等式左右同時加上某一個 x_j 後，方程組表示為：

$$\begin{cases} x_1 = (c_{11} + 1)x_1 + c_{12}x_2 + c_{13}x_3 - y_1 \\ x_2 = c_{21}x_1 + (c_{22} + 1)x_2 + c_{23}x_3 - y_2 \\ x_3 = c_{31}x_1 + c_{32}x_2 + (c_{33} + 1)x_3 - y_3 \end{cases}$$

最後將以上方程組單獨在左側的變數作為信號流入的節點繪製出圖七。



圖七 線性信號流圖範例

節點代表方程式中的參數；箭頭代表信號流向，其係數為信號之強度。

需留意的是方程組和線性信號流圖的對應並非唯一³²，就例如上述方程組也可以處理成這樣：

$$\begin{cases} x_2 = c_{11}x_1 + (c_{12} + 1)x_2 + c_{13}x_3 - y_1 \\ x_3 = c_{21}x_1 + c_{22}x_2 + (c_{23} + 1)x_3 - y_2 \\ x_1 = (c_{31} + 1)x_1 + c_{32}x_2 + c_{33}x_3 - y_3 \end{cases}$$

以這個結果再重新繪圖會畫出有別於圖七的圖，不過兩者要表達的訊息是一致的。

3.4.2 原始M矩陣之訊息

M矩陣可視作由方程組的係數所構成。以圖四(a)的例子在 3.1.1 取得的M矩陣來看：

$$\mathbf{M} = \begin{matrix} & \begin{matrix} A & B & C \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} & \begin{bmatrix} -1 & 1 & 1 \\ 1 & -1 & 1 \\ 1 & 1 & -1 \end{bmatrix} \end{matrix} = \begin{bmatrix} c_{11} & c_{12} & c_{13} \\ c_{21} & c_{22} & c_{23} \\ c_{31} & c_{32} & c_{33} \end{bmatrix},$$

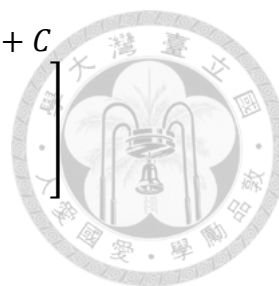
而第p組成對比較（第p條方程式）所對應的 y_p 便是該成對比較所提供的相對治療效果。

線性信號流圖一般是每條方程式（一列）對應不同節點，由係數構成的矩陣為正方形；而 M 矩陣的列數對應納入的試驗數量，行數則對應元件治療的數量，兩邊數量通常不相等。不過限定正方形矩陣是在解一般方程組的唯一解時才需要。 M 矩陣不必顧慮已知變數的數值（也就是方程組的 y_p ），是否有解跟它是否為正方形無必然關係，因此矩陣即便是長方形也無妨；基於同樣的理由，線性信號流圖上的 y_p （參見圖七）對 M 矩陣而言也能省略。也就是說在圖上我們只考慮訊息是如何流動的，而不去考慮流入了什麼訊息。所以我們仍舊以一系列對應一個節點的方式去繪圖，只不過在對應的方式做調整，改成從一系列元素非零的位置所對應到的元件治療中擇一作為該列的節點，如此一來就不會限定每列一定要對應到不同的節點。

根據選擇節點的方式不同，會繪製出不同的圖，不過表達的訊息是一致的³²，因此我們傾向繪製較為精簡的圖。為了使圖能夠盡可能的精簡，每列在選擇節點的時候遵從以下規則。其一，當非零的元素僅有一個，以該位置對應到的元件治療為節點。其二，當非零的元素超過一個，且+1的數量不多於-1時，將從左至右第一個出現的+1所對應的元件治療作為節點；而+1的數量多於-1時，則將從左至右第一個出現的-1所對應的元件治療作為節點。

在此規則下，同樣的資料以不同的順序輸入（即做列交換或行交換）仍舊可能繪製出不同的圖。雖然可能呈現出不同的圖，但本質上都是源於同一組方程式。在圖上的差異基本上只會體現在某幾列訊息流動的方向是都流入 x_j ，還是都流入另一個結點 x_k 。由於 x_k 和 x_j 之間必定會有訊息流動，同一列式子呈現出的訊息流動可以藉此兩者互通，因此完全不會影響判讀。這點於後面的範例採取指定的選法後，再採取另一選取方式所繪製的圖加以說明。

用 M 矩陣去繪圖能得到最原本的元件治療證據分布狀況。以圖五(a)為例，原始的網絡圖如圖八(a)所示，由四種成對比較組成。該網絡圖得到的 M 矩陣如下：



$$B = \begin{matrix} & A & B & A+B & A+C & A+B+C \\ \text{contrast 1} & \left[\begin{array}{ccccc} 1 & 0 & 0 & 0 & -1 \\ 1 & 0 & -1 & 0 & 0 \\ 0 & 0 & -1 & 0 & 1 \\ 0 & -1 & 0 & 1 & 0 \end{array} \right] \end{matrix}$$

$$C = \begin{matrix} & A & B & C \\ A & \left[\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 1 & 1 & 0 \\ 1 & 0 & 1 \\ 1 & 1 & 1 \end{array} \right] \\ B & \\ A+B & \\ A+C & \\ A+B+C & \end{matrix}$$

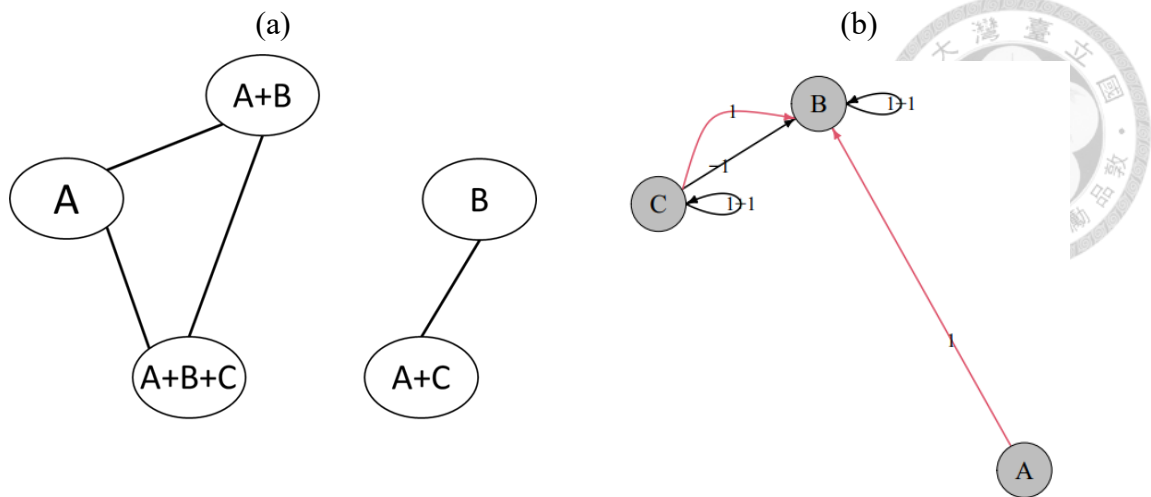
$$M = BC = \begin{matrix} & A & B & C \\ \text{contrast 1} & \left[\begin{array}{ccc} 0 & -1 & -1 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \\ 1 & -1 & 1 \end{array} \right] \\ \text{contrast 2} & \\ \text{contrast 3} & \\ \text{contrast 4} & \end{matrix}$$

第一列代表 contrast 1 提供的為 A 治療減去 A+B+C 治療的效果，其餘以此類推。

我們在前面所述的規則下，以方程組來展現如何將其處理成圖片的過程，其中 y_p 在圖上可忽略，要直接假設為 0 也可以，但未免方程式移項過程中造成誤導，我們在式子上依舊保留它們：

$$\begin{matrix} \text{contrast 1} \\ \text{contrast 2} \\ \text{contrast 3} \\ \text{contrast 4} \end{matrix} \begin{matrix} A & B & C \\ \left[\begin{array}{ccc} 0 & -1 & -1 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \\ 1 & -1 & 1 \end{array} \right] \end{matrix} \rightarrow \begin{cases} -x_B - x_C = y_1 \\ -x_B = y_2 \\ x_C = y_3 \\ x_A - x_B + x_C = y_4 \end{cases} \rightarrow \begin{cases} x_B = -x_C - y_1 \\ x_B = (1+1)x_C + y_2 \\ x_C = (1+1)x_C + y_3 \\ x_B = x_A + x_C - y_4 \end{cases}$$

將最後的方程組以線性信號流圖的方式視覺化可得圖八(b)的圖。由於一方程式可能會產出超過一個以上的箭頭，因此圖中僅產出一個箭頭的方程式以黑色代表；而其餘會產出兩個箭頭以上之方程式將有各自的顏色。圖八(b)中的 A 和 C 同時以粉色（非黑色）箭頭指向 B 且係數皆為+1，代表 **M** 矩陣含有 A 和 C 相加減去 B 元件效果的訊息；C 以黑色箭頭指向 B 且係數為-1，代表 **M** 矩陣含有 B 加上 C 元件效果的訊息；某節點如 B 和 C 一樣有自己指向自己的箭頭時，代表該元件必定有解，原因是該結果對應回原本的方程式，表示有某個成對比較單獨提供了該元件之證據。從圖八(b)可判斷 B 和 C 元件必定有解，且資料具備 A、B 和 C 元件作線性組合的效果之訊息，固可推斷 A 也必定有解。



圖八 從網絡圖到信號流圖

(a) 網絡圖；(b) 以 M 矩陣繪製出的線性信號流圖。圖上呈現了四個方程式：三個黑色箭頭各代表一個方程式，兩個粉色箭頭則代表一個方程式。

就算制定繪圖規則，根據資料的排列順序依舊有可能導致同一組資料繪製出不同的圖。不過這並不是個問題，即使圖不盡相同，仍能從中解讀出同樣的結果，因為這不會改變它們代表的是同一組方程式的事實³²。我們延續上述例子，嘗試採取另一種選取方式。將 C 矩陣的第二、三行做行交換，使 M 矩陣也跟著交換：

$$M = BC = \begin{matrix} \text{contrast 1} \\ \text{contrast 2} \\ \text{contrast 3} \\ \text{contrast 4} \end{matrix} \begin{bmatrix} A & C & B \\ 0 & -1 & -1 \\ 0 & 0 & -1 \\ 0 & 1 & 0 \\ 1 & 1 & -1 \end{bmatrix}$$

接著依舊照原本的規則選取，選取出來的結果會略有不同。第一列由左至右第一個係數為-1所對應的變成 C 元件，其餘選定的流入節點按照規則不會改變。這個流程可視作採取不同的選取方式。以新的矩陣作圖，會使得圖八(b)上由 C 節點指向 B 節點且係數為-1的線條，變成由 B 節點指向 C 節點且係數為-1的線條。該線條的方向改變絲毫沒有影響判讀，依舊能根據和前面一樣的邏輯得出相同的結論，也就是 A 、 B 、 C 三者皆有解。由此可知，即使不按照規則來選定節點也無妨，加上規則是為了讓最後呈現的圖片能盡可能使用最少的線條。

從線性信號流圖可以推斷哪些元件是能解的出來的，且該圖能夠確實呈現以元件為節點的訊息結構。然而當訊息複雜度較高時，圖上可能會出現過多箭頭，再加上係數可能有正有負，判讀的難度亦會隨之上升。因此，除了保留原始訊息的圖以外，還需要能輔助我們判讀結果的輔助圖。輔助圖將由 M 矩陣經過高斯消去法簡化的矩陣轉換成線性信號流圖而得。

3.4.3 透過高斯消去法簡化之訊息

我們將於此小節分兩個階段採用 Gauss-Jordan 方法³³，首先做正向消去 (Forward elimination)，使矩陣呈現上三角矩陣；再做反向消去使最終矩陣呈現簡化列階梯形矩陣 (Reduced row echelon form)³⁴。不同階段簡化訊息的程度不同，三個階段（原始的、只做正向消去法的，以及簡化成簡化列階梯形矩陣的）繪製出來的圖與原始圖做比較也各有其優缺點。

在做消去法之前，我們先將矩陣以列交換整理過一遍，而且在執行正向消去法的過程中，每處理好一行都要重新做列交換整理一遍矩陣。整理的方式是從第一行開始，根據該行數值的絕對值由大往小把每列重新排過；若第一行數值的絕對值相同，則以第二行數值的絕對值大小來排列；若第二行的也相同，則再繼續往下一行來比較，一直比到最後一行為止。下面我們用一個簡單的例子來看：

$$\begin{array}{cccc}
 & c1 & c2 & c3 \\
 r1 & \begin{bmatrix} -4 & 1 & -1 \\ 1 & 1 & -3 \\ 2 & -4 & 1 \\ -4 & -4 & -3 \\ 2 & 3 & 2 \\ 2 & 2 & -3 \\ 1 & 3 & 2 \\ -4 & -4 & -3 \end{bmatrix} & \xrightarrow{\text{rearr()}} & \begin{array}{ccc}
 c1 & c2 & c3 \\
 r8 & \begin{bmatrix} -4 & -4 & -3 \\ -4 & -4 & -3 \\ -4 & 1 & 1 \\ 2 & -4 & 1 \\ 2 & -3 & 2 \\ 2 & 2 & -3 \\ 1 & -3 & 2 \\ 1 & 1 & -3 \end{bmatrix} & ,
 \end{array}
 \end{array}$$

由於是根據絕對值來排序，因此第一行為 -4 的列會排在比較上面；而第一行絕對值相同的則比較到了第二行的絕對值大小，其餘的以此類推。

此舉是為了使節點與節點的關係（也就是矩陣中非零的部分）盡可能地保留於右上角，以免在使用 Gauss-Jordan 方法做正向消去時，把本應有關係的部

分消去而無法呈現於圖上。我們希望在正向消去過後，減少原始圖上結構重複的箭頭，只保留最基礎的箭頭以反應節點間本就存在的關係。不進行整理的話，某些在原始圖上本應存在以示兩節點之關係的箭頭，在正向消去的過程中會被完全消去，從而使兩者看起來像是獨立一樣。再次以圖八(a)產出的 M 矩陣當作範例，並將 contrast 簡寫成 c 。當我們的資料只有 $c1, c2, c3$ 時，在矩陣以 $c2, c3, c1$ 的順序排列的狀態下執行正向消去：

$$\begin{array}{l} c2 \\ c3 \\ c1 \end{array} \begin{array}{cc} B & C \\ \begin{bmatrix} -1 & 0 \\ 0 & 1 \\ -1 & -1 \end{bmatrix} \end{array} \rightarrow \begin{array}{l} r1 = -c2 \\ r2 = c3 \\ r3 = c1 - c2 + c3 \end{array} \begin{array}{cc} B & C \\ \begin{bmatrix} 1 & 0 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \end{array}$$

以此結果去做圖，B 和 C 節點之間會看起來沒有關係。若將 0 透過行列交換使其盡可能集中於左下角，也就是採用原本 $c1, c2, c3$ 的順序執行正向消去：

$$\begin{array}{l} c1 \\ c2 \\ c3 \end{array} \begin{array}{cc} B & C \\ \begin{bmatrix} -1 & -1 \\ -1 & 0 \\ 0 & 1 \end{bmatrix} \end{array} \rightarrow \begin{array}{l} r1 = -c1 \\ r2 = c3 \\ r3 = c2 - c1 - c3 \end{array} \begin{array}{cc} B & C \\ \begin{bmatrix} 1 & 1 \\ 0 & 1 \\ 0 & 0 \end{bmatrix} \end{array}$$

圖上依舊能透過 $r1$ 保留 B 和 C 之間的聯繫。

將圖八(a)產出的 M 矩陣做行列交換後，再做正向消去：

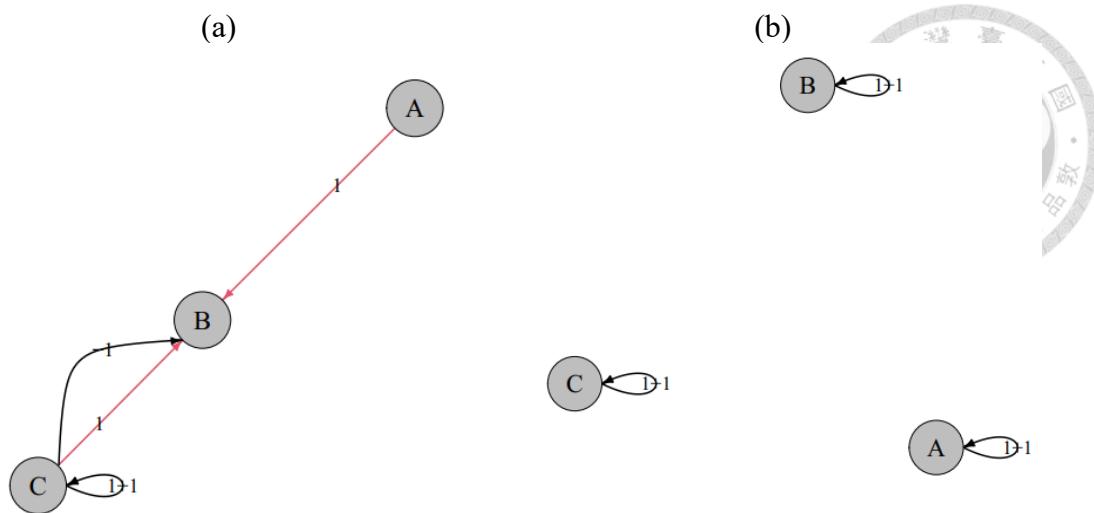
$$\begin{array}{l} M = \\ \\ \\ \end{array} \begin{array}{l} c1 \\ c2 \\ c3 \\ c4 \end{array} \begin{array}{ccc} A & B & C \\ \begin{bmatrix} 0 & -1 & -1 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \\ 1 & -1 & 1 \end{bmatrix} \end{array} \rightarrow \begin{array}{l} c4 \\ c1 \\ c2 \\ c3 \end{array} \begin{array}{ccc} A & B & C \\ \begin{bmatrix} 1 & -1 & 1 \\ 0 & -1 & -1 \\ 0 & -1 & 0 \\ 0 & 0 & 1 \end{bmatrix} \end{array}$$

$$\begin{array}{l} M_F = \\ \\ \\ \end{array} \begin{array}{l} r1 = c4 \\ r2 = -c1 \\ r3 = c2 - c1 \\ r4 = c3 - c2 + c1 \end{array} \begin{array}{ccc} A & B & C \\ \begin{bmatrix} 1 & -1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \end{array}$$

得到上三角矩陣 M_F 後，調整成繪圖用的方程式：

$$\begin{cases} x_A - x_B + x_C = y'_1 \\ x_B + x_C = y'_2 \\ x_C = y'_3 \\ 0 = y'_4 \end{cases} \rightarrow \begin{cases} x_B = x_A + x_C - y'_1 \\ x_B = -x_C + y'_2 \\ x_C = (1+1)x_C + y'_3 \\ 0 = y'_4 \end{cases}$$

其中 $y'_1 = y_4$ 、 $y'_2 = -y_1$ 、 $y'_3 = y_2 - y_1$ ，而 $y'_4 = y_3 - y_2 + y_1$ 。忽略所有的 y'_r ，最後轉換成圖九(a)的線性信號流圖。



圖九 經簡化之矩陣的線性信號流圖

(a) M 矩陣經正向消去所產生之上三角矩陣 M_F 的線性信號流圖。圖上呈現了三個方程式：兩個黑色箭頭各代表一個方程式，兩個粉色箭頭則代表一個方程式。(b) M_F 矩陣經反向消去所產生之簡化列階梯形矩陣 M_{FB} 的線性信號流圖。圖上呈現了三個方程式：三個黑色箭頭各代表一個方程式。

經列交換並以消去法精減訊息，能使 M 矩陣中重複而龐雜的訊息被大幅減去。圖九(a)呈現的資訊較為集中，箭頭數量會比圖八(b)還少。雖然簡化了原本元件訊息的分布，但是圖九(a)依然能呈現訊息主要的脈絡。從圖上一樣能看到 B 自己指向自己必然有解；而圖上 B、C 元件間黑色箭頭顯示兩者間具有線性關係，因此可推得 B 也有解；最後，兩個粉色箭頭指出 A 和 B、C 元件之間具有線性組合，因此也可推得 A 有解。

執行到正向消去使下三角矩陣被消為零就己能大幅增進判讀速度，不過要使訊息再更精煉的話，還需要做反向消去將上三角矩陣能消去的非零元素進一步消除。將 M_F 矩陣做反向消去得到 M_{FB} 矩陣：

$$M_F = \begin{matrix} r1 = c4 & \begin{bmatrix} A & B & C \\ 1 & -1 & 1 \\ 0 & 1 & 1 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{bmatrix} \\ r2 = -c1 \\ r3 = c2 - c1 \\ r4 = c3 - c2 + c1 \end{matrix}$$

$$\rightarrow \mathbf{M}_{FB} = \begin{matrix} r'1 = r1 + r2 - 2 * r3 \\ r'2 = r2 - r1 \\ r'3 = r3 \\ r'4 = r4 \end{matrix} \begin{matrix} A & B & C \\ \left[\begin{array}{ccc} 1 & 0 & 0 \\ 0 & 1 & 0 \\ 0 & 0 & 1 \\ 0 & 0 & 0 \end{array} \right] \end{matrix}$$



將 \mathbf{M}_{FB} 矩陣調整成繪圖用的方程式：

$$\begin{cases} x_A = y_1'' \\ x_B = y_2'' \\ x_C = y_3'' \\ 0 = y_4'' \end{cases} \rightarrow \begin{cases} x_A = (1 + 1)x_A + y_1'' \\ x_B = (1 + 1)x_B + y_2'' \\ x_C = (1 + 1)x_C + y_3'' \\ 0 = 0 \end{cases}$$

其中 $y_1'' = y_1' + y_2' - 2y_3'$ 、 $y_2'' = y_2' - y_1'$ 、 $y_3'' = y_3'$ ，而 $y_4'' = y_4'$ 。忽略所有的 y_i'' ，最後繪製成圖九(b)的線性信號流圖。此時若所有元件皆可估計，那所有的節點都會自己指向自己且兩兩之間不會有箭頭，如圖九(b)；若有元件不可估計，則會有某些節點兩兩相連，且整條關係鏈中完全無自己指向自己的節點。

總得來說， \mathbf{M} 的線性信號流圖反應元件原本的證據分布情況，不過當研究數上升或元件數上升使得圖的結果不好判讀時，可以透過高斯消去法簡化矩陣的訊息以繪製出較為精煉的圖。經簡過後 \mathbf{M}_F 的線性信號流圖在降低判讀難度的同時，依舊保留了元件間最基礎的關係。而進一步精簡的 \mathbf{M}_{FB} 的線性信號流圖能直接看出哪些元件不可估，並找出使其可估的研究最少需包含哪些訊息，所以在都可估的情況下較沒什麼意義。基本上 \mathbf{M}_F 和 \mathbf{M}_{FB} 的線性信號流圖屬於輔助解讀的工具，前者適合針對關係的評估，後者則適合用在不可估時判斷欠缺的證據種類。

3.4.4 交互作用項

在 3.3 小節中提過，若想估計元件的交互作用項，可以將交互作用項當作一個新的元件放入模型。這會使得 \mathbf{M} 矩陣多出代表交互作用項的行，所以視覺化時，圖上也會跟著多出代表交互作用項的節點。

延續圖八使用之例子，在 \mathbf{M} 矩陣中加入代表 B 和 C 之交互作用項的行：



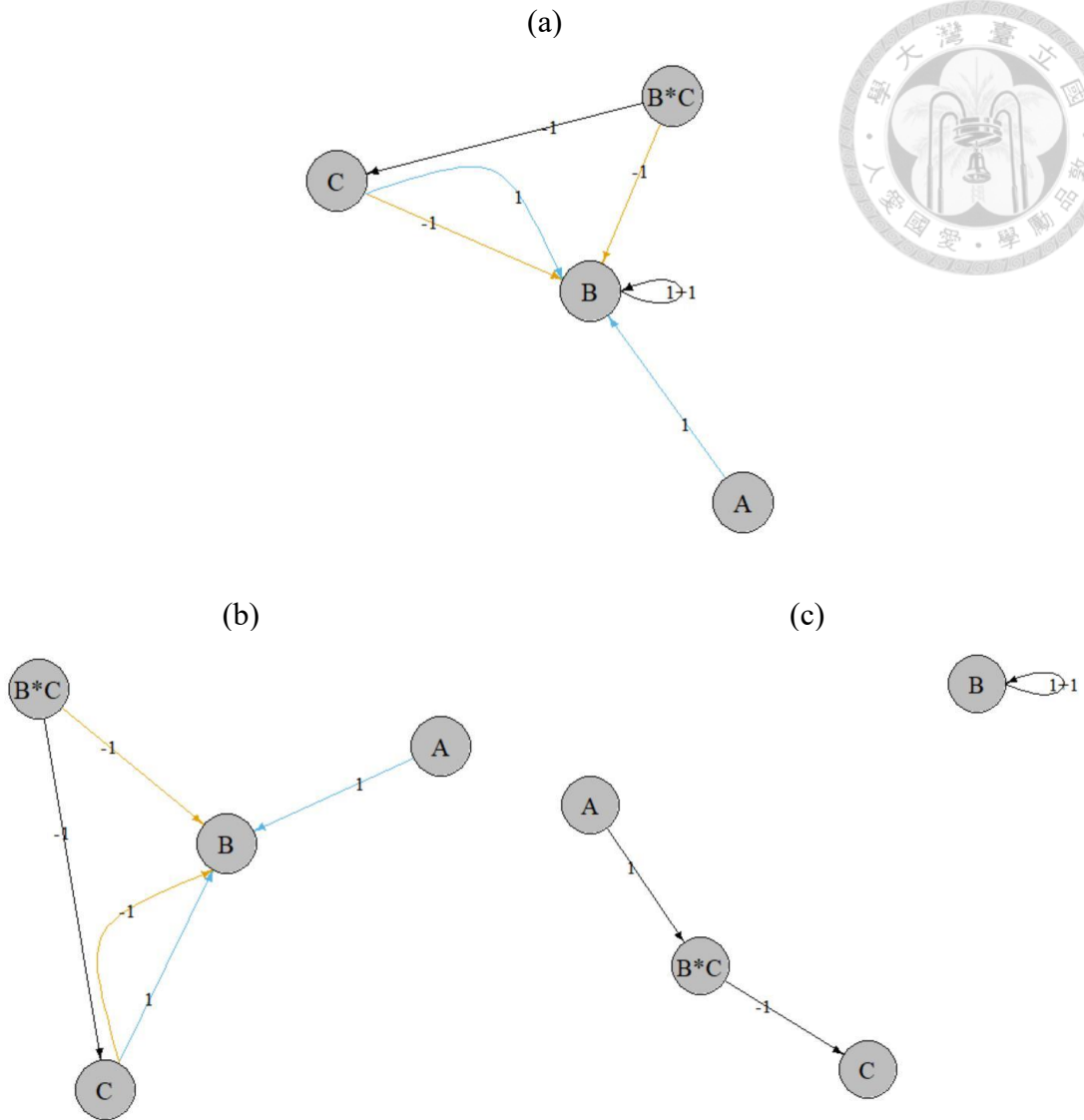
$$M^* = \begin{matrix} & A & B & C & B * C \\ c1 & 0 & -1 & -1 & -1 \\ c2 & 0 & -1 & 0 & 0 \\ c3 & 0 & 0 & 1 & 1 \\ c4 & 1 & -1 & 1 & 0 \end{matrix}$$

以其為基礎繪製結果如圖十(a)所示，多了 B*C 的節點。圖片的判讀方式也與前述相同，從圖十(a)可以看到 B 元件可估計。由 B*C 和 C 指向 B 的兩個係數為-1 之黃色箭頭，涵蓋了由 B*C 指向 C 的係數為-1 之黑色箭頭，以及 B 自己指向自己的黑色箭頭兩者的訊息，因此這部分依舊只能得出 B 可估、B*C 和 C 不可估的結論。剩下的藍色箭頭涉及 A、B、C 三個元件，但其中僅有 B 可從其他箭頭得知可估，故也不足以推得剩下的兩個元件 A 和 C 可估。綜上所述，加入 BC 交互作用項時，僅元件 B 可估。

將矩陣做行列交換並透過高斯消去法簡化後可得：

$$M_F^* = \begin{matrix} & A & B & C & B * C \\ 1 & -1 & 1 & 0 \\ 0 & 1 & 1 & 1 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{matrix} \quad \text{和} \quad M_{FB}^* = \begin{matrix} & A & B & C & B * C \\ 1 & 0 & 0 & -1 \\ 0 & 1 & 0 & 0 \\ 0 & 0 & 1 & 1 \\ 0 & 0 & 0 & 0 \end{matrix}$$

在圖十(b)可以看到與圖十(a)唯一的差異就是，B 自己指向自己的線條消失了，在判讀上反倒會造成一些劣勢，無法得知 B 單獨有解。這跟我們做了列交換有關，若要再優化這部分，得考慮連行交換也一起做，或是在做列交換的時候加上一些限制讓只有一個元素是非零的列不要移動。從圖十(c)精簡到最減的結果來看，若要使其可估計的話，值得考慮的便是能夠單獨提供 A 或 C 元件之證據的成對比較，如 A 與 A+C 治療的成對比較；或是能提供 A 和 C 元件效果差異的證據，如 A 與 C 治療的成對比較。



圖十 加入 BC 交互作用項之線性信號流圖

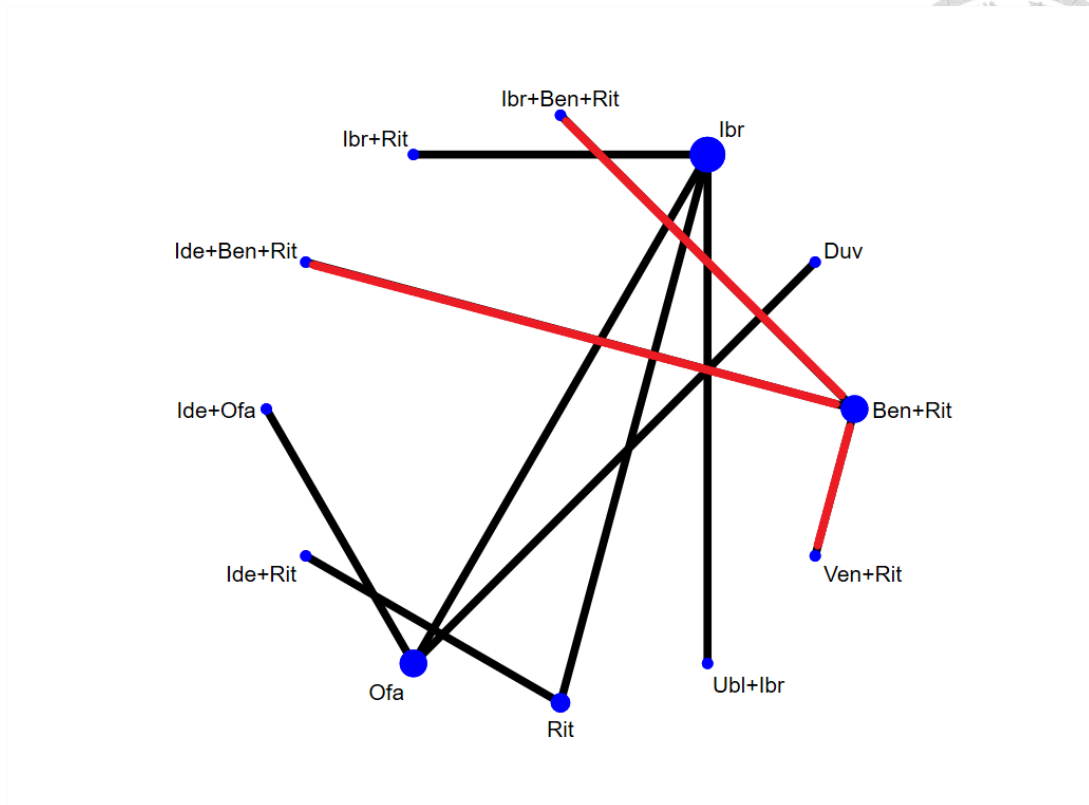
(a) M^* 矩陣之線性信號流圖。圖上呈現了四個方程式：兩個黑色箭頭各代表一個方程式，同為黃色和淺藍色的箭頭則各代表一個方程式。(b) M_F^* 的線性信號流圖。圖上呈現了三個方程式：一個黑色箭頭代表一個方程式，同為黃色和淺藍色的箭頭則各代表一個方程式。(c) M_{FB}^* 的線性信號流圖。圖上呈現了三個方程式：三個黑色箭頭各代表一個方程式。

綜上所述，在 3.1 小節我們採用兩個矩陣整理資料結構，使交互作用項能以較為靈活的方式添加；加入交互作用項後，是否依舊能夠重建連結的判斷方式也跟 3.2 小節的判斷方式一致。而套用到 3.4 小節視覺化成圖形時，雖然每次添加/刪減交互作用項都需要重新繪製新的圖，但由於調整方式很靈活，繪圖本身也不費時，而且加入交互作用項本來就是在建構另一個模型，所以倒也算不上是缺點。至此方法已介紹完畢，接下來將用兩個實際範例來展現結果。

第4章 實際範例

為呈現不可以及可以透過元件網絡統合分析重建網絡之差異，本研究將使用兩個資料集分別呈現兩者的結果。

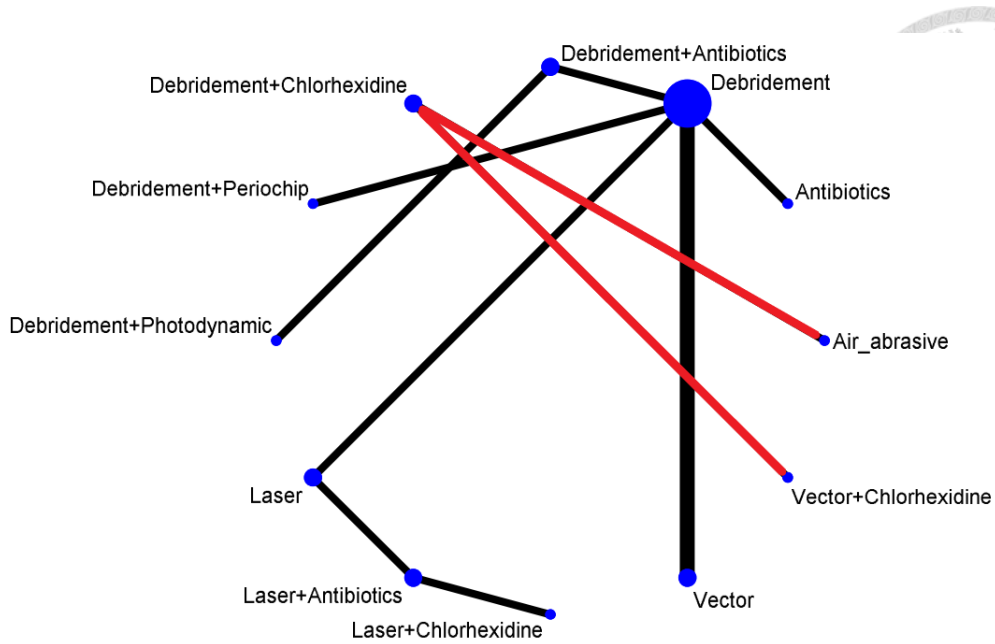
資料集一來自 Chen 等人於 2019 年發表的網絡統合分析研究，該研究探討了用於慢性淋巴細胞性白血病(Chronic lymphocytic leukemia)之新型標靶治療的療效³⁵ (參見附錄 Part B)。不過由於兩項因素使得我們實際使用的資料與該研究有所出入，其一，該研究發表之後作者群有更新過資料，本研究所使用之資料是透過其中一位作者取得，因此會比已發表的內容多三篇隨機對照試驗³⁶⁻³⁸；其二，該研究為了使斷裂的網絡連結起來而納入了兩篇非隨機對照研究，然而考量到本研究欲探討元件網絡統合分析重建斷裂網絡之性質，故於本研究中將二者拿掉，使網絡在僅納入隨機對照試驗時為斷裂狀態。因此，此資料集最終納入 10 篇隨機對照試驗，以包含基於 8 種元件治療所組合而成的 12 種治療為節點，形成了一個斷裂成兩個子網絡的網絡結構 (參見圖十一)。



圖十一 資料集一之網絡圖

圖中含有 10 篇隨機對照試驗，網絡圖斷裂成兩個子網絡，黑線與紅線分別代表其中一個子網絡。12 種治療由以下 8 個元件組成：Bendamustine (Ben) 、 Duvelisib (Duv) 、 Ibrutinib (Ibr) 、 Idelalisib (Ide) 、 Ofatumumab (Ofa) 、 Rituximab (Rit) 、 Ublituximab (Ubl)和 Venetoclax (Ven)。

資料集二則以實際資料的節點來生成模擬資料。由於公開發表的文獻中，幾乎不會有網路斷裂的研究，因此我們先以治療為組合式這點著手，挑選出了一篇探討植體周圍炎非手術治療之療效的研究³⁹（參見附錄 Part C）。接著以該篇的治療名稱為基礎，模擬出了一個斷裂成兩個子網絡的網絡結構。本研究並未採用較為實際的方式生成資料，一方面是由於本研究著重於斷裂網路之重建，透過元件網絡統合分析估計出的結果並不是很重要；另一方面則是用作參考的文獻中實際上是以兩個指標來評估療效，僅看單一指標的話，並非每個治療都有該指標的資料，因此也無實際數據可供參考（參見圖十二）。



圖十二 資料集二之網絡圖

圖中含有 11 組成對比較，網絡圖斷裂成兩個子網絡，黑線與紅線分別代表其中一個子網絡。12 種治療由以下 8 個元件組成：Air_abrasive(Air)、Antibiotics(Ant)、Chlorhexidine(Chl)、Debridement(Deb)、Laser(Las)、Periochip(Per)、Photodynamic(Pho)、Vector(Vet)。

第5章 結果

本章利用前一章所介紹的資料來進行方法的展示。5.1 節先從網絡無法透過元件網絡統合分析連通的資料一的結果開始報告，接著 5.2 節則是透過元件網絡統合分析連通的資料二。

5.1 資料一：無法透過元件網絡統合分析連通之網絡

透過圖十一的網絡圖可以得到圖十三(a)的 M 矩陣。計算它的秩會得到 7，也就是說最多只能估出 7 個元件的效果，比實際元件數 8 還要小，故儘管子網絡間有 Ide 和 Rit 兩個共同元件，該網絡依然無法透過元件網絡統合分析重新連接。接下來為判斷哪些元件無法被估計，將 M 矩陣轉化為線性信號流圖，結果如圖十三(b)所示。圖片可以分為四塊，第一、二塊顯示資料中存在能直接估計 Ubl 和 Ide 效果的成對比較，且兩者的效果皆不受其他元件影響。第三塊顯示資料中存在能直接估計 Ibr 和 Rit 效果的成對比較，不過兩者之間連通的箭頭表示兩者效果有關聯。根據係數可以判斷成正向關係，Ibr 的效果越好，Rit 的效果也會越好。至於 Ofa，雖然無法直接估出它的效果，但它與 Ibr 之間也有箭頭連通，考慮到 Ibr 效果本身就估得出來，我們可以間接估出 Ofa 的效果。根據類似的道理，Duv 也能間接估計而得。最後一塊顯示 Ben 和 Ven 之間有正向關係，但僅能知道兩者之間有關聯，無法估計出唯一解。

圖十三(b)的第三塊有較多箭頭連通節點，再加上有一些元件沒辦法直接估計出來，所以判讀起來較為困難。此時可以使用高斯消去法簡化 M 矩陣（結果參見圖十四(a)），轉化為線性信號流圖，結果如圖十四(b)所示。圖上可以看到連通的箭頭被大幅簡化，直接呈現出了哪些元件可估計、哪些元件不可估計。結果與圖十三(b)一樣，Ben 和 Ven 兩者無法被估計。根據圖上的訊息，若要再納入研究使所有元件皆可估計，一是能提供 Ben 和 Ven 呈反向關係之訊息的成對比較，也就是研究納入合併使用 Ben 和 Ven 之治療，且其中納入之治療至少有一個完全未使用 Ben 和 Ven，例如比較 Ben+Ven 和 Rit 的研究，或是比較

Ben+Ven+Rit 和 Ibr 的研究；二是能單獨提供 Ben 或 Ven 的訊息的成對比較，例如比較 Ben 和 Rit，或是比較 Ven+Ben 和 Ven。

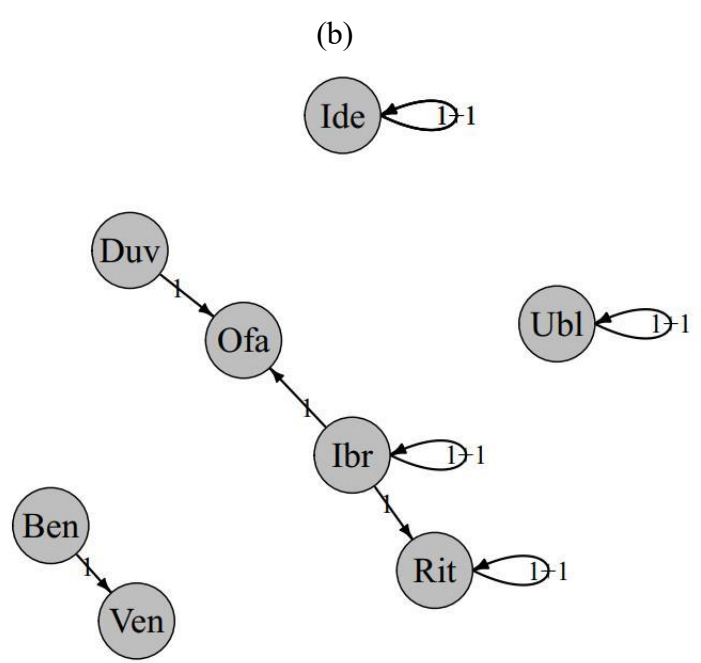
綜上所述，資料一無法透過元件網絡統合分析重建網絡；它無法估計出 Ben 和 Ven 的效果，根據圖所整理的訊息，需要再補充能提供足以使 Ben 和 Ven 估計出來的研究。





(a)

	Ben	Duv	Ibr	Ide	Ofa	Rit	Ubl	Ven
c1	0	0	0	1	0	0	0	0
c2	0	1	0	0	-1	0	0	0
c3	0	0	1	0	-1	0	0	0
c4	0	0	1	0	0	-1	0	0
c5	0	0	-1	0	0	0	0	0
c6	1	0	0	0	0	0	0	-1
c7	0	0	0	-1	0	0	0	0
c8	0	0	0	1	0	0	0	0
c9	0	0	0	0	0	-1	0	0
c10	0	0	0	0	0	0	-1	0

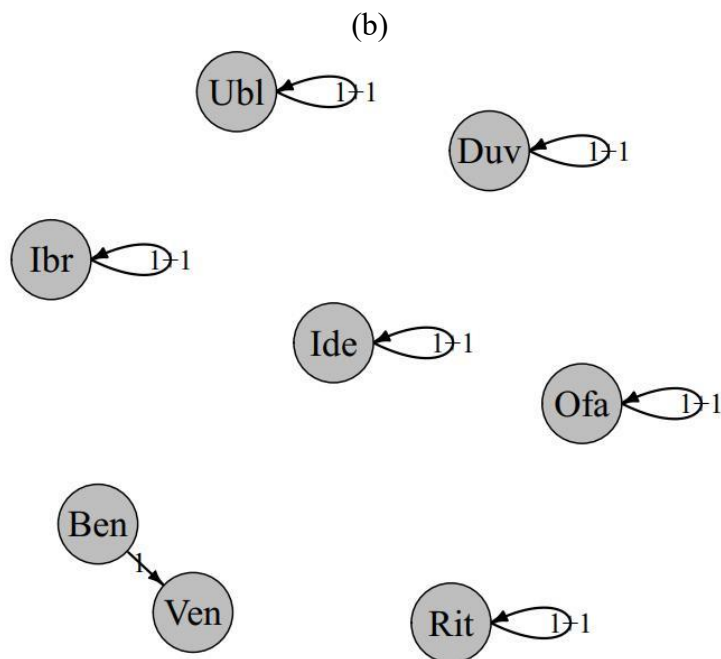


圖十三 資料一之結果

(a) M 矩陣；(b) M 矩陣的線性信號流圖。圖上包含了十個方程式：十個黑色箭頭各代表一個方程式，其中 Ide 自己指向自己的地方有三個箭頭重疊在一起而看上去僅有八個箭頭，不過不影響結構判讀。

(a)

	Ben	Duv	Ibr	Ide	Ofa	Rit	Ubl	Ven	Combination
row 1	1	0	0	0	0	0	0	-1	$1*c6$
row 2	0	1	0	0	0	0	0	0	$1*c2-1*c3+1*c5$
row 3	0	0	1	0	0	0	0	0	$-1*c5$
row 4	0	0	0	1	0	0	0	0	$1*c1$
row 5	0	0	0	0	1	0	0	0	$-1*c3-1*c5$
row 6	0	0	0	0	0	1	0	0	$-1*c4-1*c5$
row 7	0	0	0	0	0	0	1	0	$-1*c10$
row 8	0	0	0	0	0	0	0	0	$-1*c1+1*c8$
row 9	0	0	0	0	0	0	0	0	$-1*c4-1*c5+1*c9$
row 10	0	0	0	0	0	0	0	0	$1*c1+1*c7$



圖十四 資料一之結果經高斯消去法

(a) M_{FB} 矩陣，其中 Combination 描述各列是由來自圖十三(a)的 $c1 \sim c10$ 如何線性組合而成；(b) M_{FB} 矩陣的線性信號流圖。圖上包含了七個方程式：七個黑色箭頭各代表一個方程式。

5.2 資料二：可透過元件網絡統合分析重建之網絡

透過圖十二可以得到圖十五(a)的 M 矩陣。計算它的秩會得到 8，也就是說最多只能估出 8 個元件的效果，與元件數 8 相同，且子網絡間有 Deb、Chl 和 Vec 三個共同元件，故可判斷該網絡可透過元件網絡統合分析重新連接。以圖形判斷哪些元件無法被估計時，將 M 矩陣轉化為線性信號流圖，結果如圖十五(b)所示。圖片可以分為兩塊，第一塊顯示資料中存在能直接估計 Per 效果的成對比較，且其效果皆不受其他元件影響。第二塊則顯示資料中存在能直接估計 Ant 效果的成對比較。雖然無法直接估出 Deb、Chl 和 Pho 的效果，但它們與 Ant 之間有箭頭連通，考慮到 Ant 效果本身就估得出來，我們可以間接估出三者的效果。根據類似的道理，Vec、Las 和 Air 也能間接估計而得。綜上所述，全部元件效果皆可估計，與透過 M 矩陣得到的結果一致。

圖十五(b)的第二塊也是有較多箭頭連通節點，再加上不是所有元件都能直接估計出來，所以判讀起來較為困難。此時可以使用高斯消去法簡化 M 矩陣（參見圖十六(a)），轉化為線性信號流圖，結果如圖十六(b)所示。圖上可以看到連通的箭頭被大幅簡化，直接呈現出了哪些元件可估計、哪些元件不可估計。結果與圖十五(b)一樣，全部元件效果皆可估計。

綜上所述，資料二可以透過元件網絡統合分析重建網絡。而根據圖十二及 M 矩陣的訊息，重建的情況如圖十七所示。Air_abrasive 僅在較小的子網絡中存在，因此不會透過它重建網絡。Debridement 在較大的子網絡中，僅 Debridement 分別跟 Antibiotics、Laser、Vector 的成對比較會提供出 Debridement 的訊息，其餘包含 Debridement 的成對比較都會將 Debridement 的效果抵銷，因此，較小子網絡的 Debridement+Chlorhexidine 僅跟較大子網絡中的 Debridement 建立連結（綠色虛線）。Chlorhexidine 在 Debridement+Chlorhexidine 和 Vector+Chlorhexidine 的成對比較中會被抵銷，因此，它只會在 Debridement+Chlorhexidine 和 Air_abrasive 的成對比較中與另一個網絡中的 Chlorhexidine 建立連結（黃色虛線）。而包含 Vector 的治療在兩個

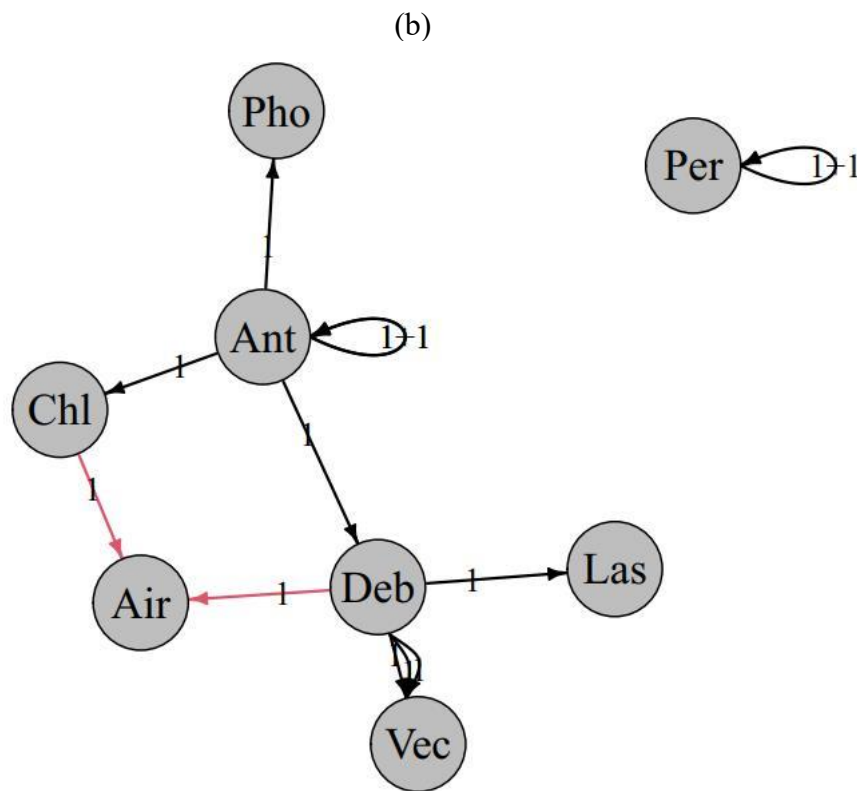
子網絡中分別只有一個，因此就是那兩個治療 Vector 和 Vector+Chlorhexidine 建立連結（藍色虛線）。

綜上所述，資料二可以透過元件網絡統合分析重建網絡；而網絡的重建會透過兩個子網絡間共有的 Debridement、Chlorhexidine 和 Vector。



(a)

	Air	Ant	Chl	Deb	Las	Per	Pho	Vec
c1	0	0	0	1	-1	0	0	0
c2	0	0	0	0	0	-1	0	0
c3	1	0	-1	-1	0	0	0	0
c4	0	-1	0	0	0	0	0	0
c5	0	1	0	0	0	0	-1	0
c6	0	0	0	1	0	0	0	-1
c7	0	0	0	1	0	0	0	-1
c8	0	0	0	1	0	0	0	-1
c9	0	1	0	-1	0	0	0	0
c10	0	-1	0	0	0	0	0	0
c11	0	1	-1	0	0	0	0	0

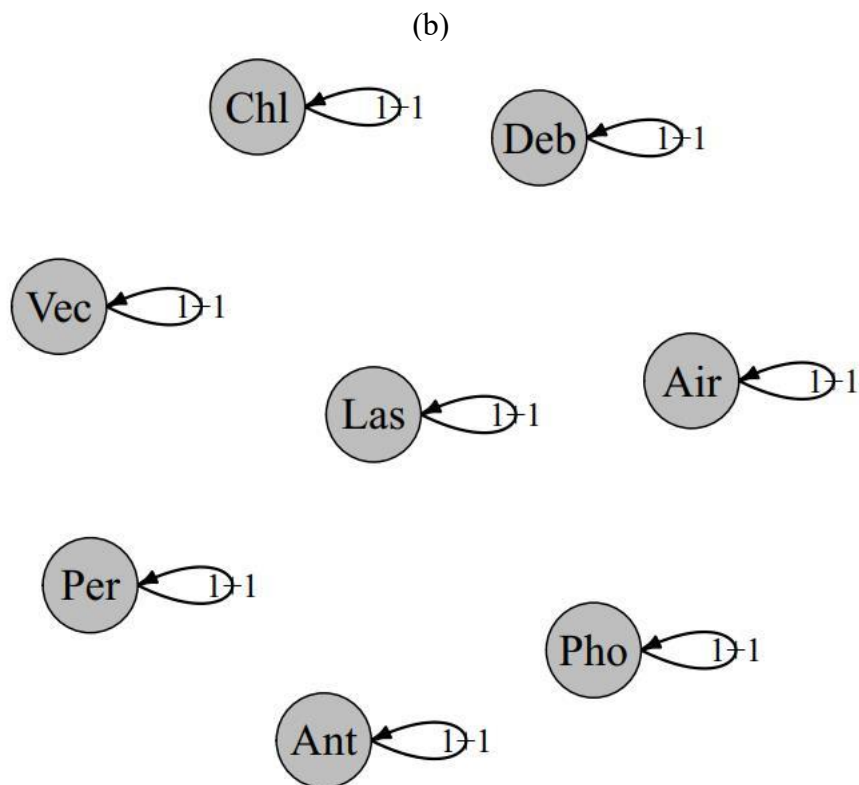


圖十五 資料二之結果

(a) M 矩陣；(b) 線性信號流圖。8 個元件分別為：Air_abrasive (Air)、Antibiotics (Ant)、Chlorhexidine (Chl)、Debridement (Deb)、Laser (Las)、Periochip (Per)、Photodynamic (Pho)、Vector (Vec)。圖上包含了十一個方程式：十個黑色箭頭各代表一個方程式，同為粉色的箭頭則代表一個方程式。其中 Ant 自己指向自己的地方有三個箭頭重疊在一起而看上去僅有十個方程式，不過不影響結構判讀。

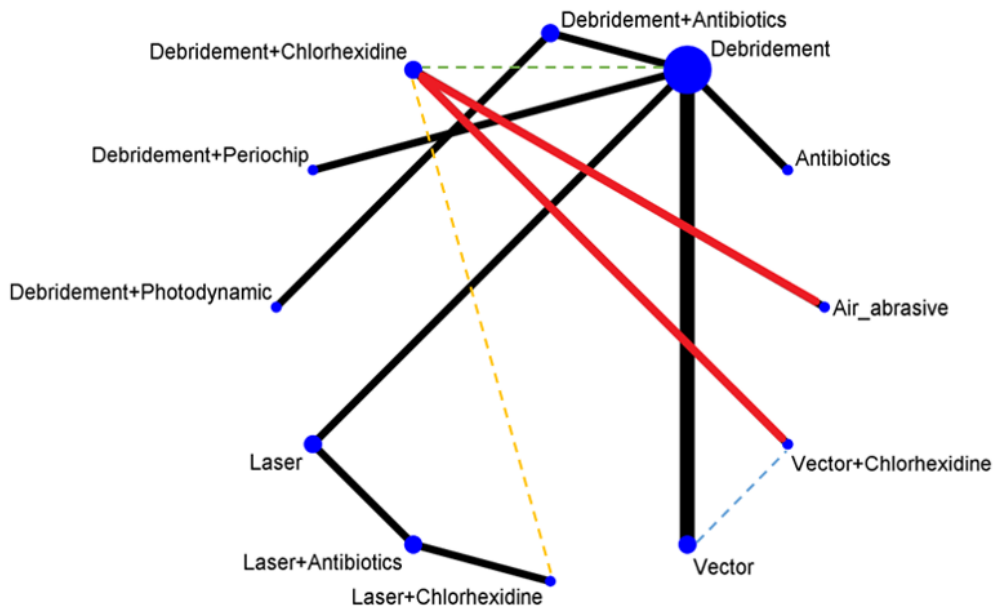
(a)

	Air	Ant	Chl	Deb	Las	Per	Pho	Vec	Combination
row 1	1	0	0	0	0	0	0	0	$1*c3-2*c4-1*c9-1*c11$
row 2	0	1	0	0	0	0	0	0	$-1*c4$
row 3	0	0	1	0	0	0	0	0	$-1*c4-1*c11$
row 4	0	0	0	1	0	0	0	0	$-1*c4-1*c9$
row 5	0	0	0	0	1	0	0	0	$-1*c1-1*c4-1*c9$
row 6	0	0	0	0	0	1	0	0	$-1*c2$
row 7	0	0	0	0	0	0	1	0	$-1*c4-1*c5$
row 8	0	0	0	0	0	0	0	1	$-1*c4-1*c6-1*c9$
row 9	0	0	0	0	0	0	0	0	$-1*c6+1*c8$
row 10	0	0	0	0	0	0	0	0	$-1*c4+1*c10$
row 11	0	0	0	0	0	0	0	0	$-1*c6+1*c7$



圖十六 資料二之結果經高斯消去法

(a) M_{FB} 矩陣，其中 Combination 描述各列是由來自圖十五(a)的 $c1 \sim c10$ 如何線性組合而成；(b) M_{FB} 矩陣的線性信號流圖。圖上包含了八個方程式：八個黑色箭頭各代表一個方程式。



圖十七 資料二重建之網絡

根據資料二網絡結構及 M 矩陣所提供之訊息，資料二的網絡重建情況會是如此。其中綠、黃、藍色虛線分別為透過 Debridement、Chlorhexidine、Vector 建立起來的聯繫。

第6章 討論

本研究首先釐清了欲使斷裂網絡可透過元件網絡統合分析，除了子網絡間需有共同元件以外，其 M 矩陣的秩也要等於納入的元件數。接著，現行標準的網絡圖無法有效的反應出元件網絡統合分析的資料結構，因此我們採用線性信號流圖的方式，以元件為節點來呈現元件網絡統合分析的資料分布情況。最後，為避免資料複雜度較高使圖片過於瑣碎而難以判讀，我們採用了高斯消去法分階段簡化訊息。簡化程度較少的圖能保留元件間連結，簡化程度較高的能直接判讀出哪些元件可估，或不可估的元件欠缺何種類型的研究。

6.1 利用元件網絡統合分析連接斷裂網絡的條件

某些特殊的斷裂網絡即使適用元件網絡統合分析，也不能說它透過其重建網絡，因此這裡要先釐清一下重建網絡的意思。前面提過斷裂的網絡還需要加上子網絡具有共同元件這一條件才能重建²⁵，因此斷裂網絡光是在元件網絡統合分析中可估，不代表它就能重建網絡。當各自獨立使用元件網絡統合分析，證據也足夠估計出各自的元件效果時，斷裂網絡間即使沒有共同元件也能夠同時在一個元件網絡統合分析的模型下進行估計。例如說，第一個子網絡的治療由 A、B 和 C 三種元件構成；第二個子網絡則由 D、E 和 F 三種元件構成。當兩個子網絡各自使用元件網絡統合分析，全部的元件皆可得出解時，同時將它們放在同一個元件網絡統合分析的模型亦可得出解。然而實際上 A 至 C 元件以及 D 至 F 元件兩邊之間的效果不會相互影響。放在同一個模型與各自分析的差異在於，異質性的估計應該會與各自估計會有些不同，從而使最後估計出來的效果略有不同。因此，在欠缺共同元件的情況下，即使元件網絡統合分析能運作，也不能說它是在重建網絡。要說其可重建，一定要如資料二的圖十七一樣，子網絡間具有共同元件。

總得來說，單獨看 M 矩陣的秩需等於納入的元件數這一條件，僅用於判斷元件網絡統合分析可否運作。無論是斷裂抑或是連通的網絡，首先要得到將成

對比較所蘊含之訊息映射到每個元件的 M 矩陣，再計算它的秩以進行判讀。當秩等於元件數時，所有的元件皆可估計。斷裂網絡的元件皆可估計，再加上子網絡間具有共同元件便意味著網絡能被重建²⁵。而當秩數小於元件數時，代表有元件不可估計；二者數字的差距則代表至少有幾個元件是不可估的。

當有部分元件無法估計時，其餘元件估計出來的結果依然是可以參考的。不過若是要讓全部元件皆可估的話，一種做法是採用 Moore-Penrose generalized inverse 強行估計出來²⁵，然而被強行估計出來的結果並沒有任何意義。因此，需要進一步釐清是哪些元件無法估計出來，它們目前尚欠缺何種證據，才能得知需額外納入何種證據才可使全部元件皆可估計。所以，我們需要一個能實際呈現元件證據分布的工具。元件網絡統合分析的證據分布無法簡單地從原本用於網絡統合分析的網絡圖上看出來。以往只要網絡連通，常用的網絡統合分析的模型就能執行的觀念⁴⁰，可能還會讓人誤以為只要網絡未斷裂，即代表所有的元件都可估計，然而實際上卻並非如此（如圖五(b)）。因此，本研究提出了新的繪圖方式以利研究者評估元件證據的分布情況。

6.2 信號流圖跟結構方程模式路徑圖之間的差異

元件網絡統合分析是在估計元件的效果，因此我們圖的節點也應用元件取代組合式治療。每組成對比較就是兩個治療的差異，因此在標準的網絡圖中，只需考慮是哪二者的差異即可，用一條線即可呈現出來。但當我們將資料結構轉換成以元件為基礎呈現時，每組成對比較涉及的元件數可能會大於兩個，在相減抵銷後，甚至可能提供的訊息只涉及一個元件。因此，在其涉及之節點數或大或小的情況下，不可能只用線條就反映在圖上，勢必要在圖上引入箭頭。而我們參考的便是線性信號流圖的方式，藉由引入了方向性的概念，使圖形能夠呈現涉及節點數量較為多樣的情況。

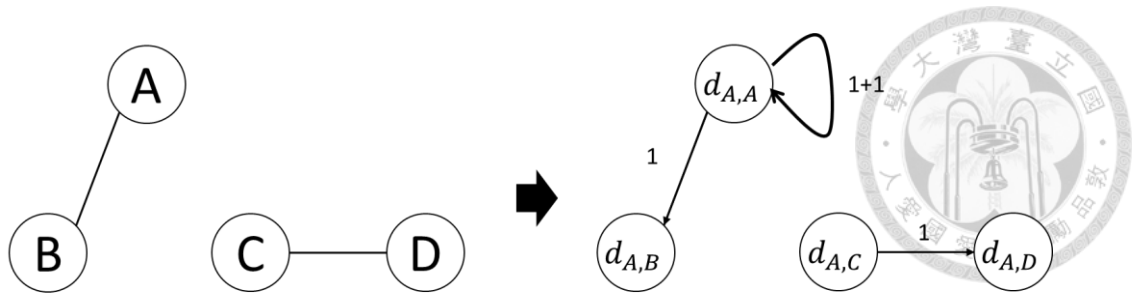
雖說在圖型中引入方向性，使線條附上箭頭，會不禁讓人聯想到結構方程模型的路徑圖²⁹，但由於兩圖上箭頭的意義截然不同，使圖型產生兩個決定性的差異。其一，本研究提出的繪圖方式並不唯一，同一證據結構可以有不同的

圖型表現方式³²。儘管如此，它們也都能判讀出相同的結果。其二，結構方程模型的路徑圖中，兩節點之間最多只會有一個單箭頭或雙箭頭的連結³⁰，而本研究提出的圖有可能超過一個單箭頭。透過新的圖可以直接解讀出是哪些元件不可估計。

而二者在繪製箭頭和係數取得的方式上也有決定性的差異。路徑圖是先根據因果關係建立箭頭再求係數；信號流圖則是先有係數再求箭頭。也就是說，二者的繪圖目的存在差異，信號流圖更注重呈現，呈現的過程中即使做出訊息簡化的動作也無妨，而這對於需要透過理論來建構箭頭的路徑圖來說是不行的。

6.3 標準網絡圖可視為信號流圖的一個特例

判讀元件是否可估，也可用於解釋 Lu & Ades 模型為何要求網絡圖一定要相連。由於我們的圖本身是以 Lu & Ades 模型為標準繪製的網絡圖的矩陣結構為基礎，再轉換成以元件為節點的圖，因此二者具備一定關聯性。標準的網絡圖可以說是我們的圖的一種特例。以標準網絡圖來看，每個治療都是相互獨立的，這等同於我們所定義的元件，也就是說每個治療可視為一個元件。將每個治療視作單一元件，以我們的方式繪圖的話，此時每個節點代表的治療 k ，可視為 2.1 小節的 $d_{A,k}$ 。在原本標準網絡圖的結構上，每條線再加上箭頭。根據我們之前的討論，這樣的一個信號流圖，是沒有辦法求得各個治療的絕對效果，這也是為什麼 Lu & Ades 模型網絡統合分析只會給出相對效果。這時將 $d_{A,A}$ 定義為 0，也就是把 A 治療的效果定為零，所以圖上原本代表參考治療的 $d_{A,A}$ 節點會多出自己指向自己的箭頭，這時候其他的治療也就跟著可以求解。所以斷開且未與參考治療相連的子網絡，節點彼此之間只會有箭頭相連，而沒有自己指向自己的箭頭，因此，這些節點所代表的治療就無法求解。以圖三(a)的例子來看，會變成圖十八的樣子。按照我們的圖的解釋來說， $d_{A,A}$ 可估計； $d_{A,B}$ 與 $d_{A,A}$ 相連，因此也可估計；而 $d_{A,C}$ 和 $d_{A,D}$ 則沒有自己指向自己的節點，因此二者不可估計。



圖十八 示意圖

以圖三(a)的例子示範為何標準網絡圖會我們的圖的特例

6.4 信號流圖之延伸應用

儘管我們的圖可能會隨著資料複雜度上升而變得難以解讀，但也能透過高斯消去法萃取出最精華的訊息再進行繪圖，從而大幅簡化圖片使研究者能快速進行判讀。釐清是哪些元件無法估計後，我們也能根據圖片填補資料缺乏的訊息，使元件網絡統合分析能夠運作。除了鑑別需填補的資料外，我們的圖的一大優勢是能判斷哪些元件會受偏誤影響。以圖十五為例，Per 與其他元件并未相連，因此，偏誤若出現在 Per 上並不會影響到其他元件的估計；反之，若出現在其他任一元件上，則除了 Per 以外的元件都將受影響。這是標準的網絡圖無法判斷的，看圖十二可能會誤以為 Periochip(Per)的偏誤是在較大的子網絡中傳遞。

我們的方法也適用加入交互作用項的情況，這類的情況標準的網絡圖又更難看出。只需要將交互作用項當作新的元件加入分析中，本研究的所有方法皆可直接套用。不過須留意的是，雖說是當作新的元件加入，但本質上還是交互作用項，它勢必會與交互作用項所涉及之元件有一定程度的牽扯⁴¹。資料要滿足三個條件，加交互作用項才有意義，否則加了也一定沒辦法估計。因此在加入之前，最好先檢視這一步驟對資料是否有意義。

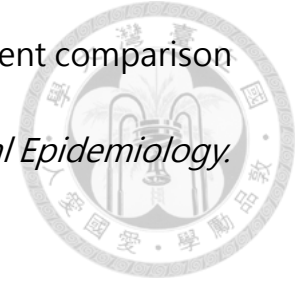
6.5 未來研究方向

目前我們的圖僅能呈現資料結構，未來可以嘗試加入更多訊息以做證據強度或偏誤傳遞的評估。一般的網絡統合分析圖中，也有些研究者透過考慮每個試驗的權重以評估網絡間訊息傳遞的比例^{42,43}，再進一步探討偏誤傳遞的可能情況⁴⁴。在 3.4.1 小節中，線性信號流圖 y_i 的部分（也就是每個成對比較的效果差異）不會影響到資料結構的呈現，因此我們先將其去除了。這使得我們僅能看出當某個元件有偏誤時，哪些與之相聯的元件也會受其影響，但卻無法評估出影響的強度。此外，由於每個研究有其各自的權重，要評估證據強度也要將其考慮進去。未來可以將每個成對比較的效果差異及其權重合併考慮，再呈現於圖上，以利研究者評估元件之間的證據強度或偏誤影響的情況。

參考文獻



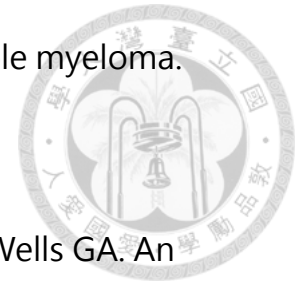
1. Burns PB, Rohrich RJ, Chung KC. The levels of evidence and their role in evidence-based medicine. *Plastic and Reconstructive Surgery*. 2011;128(1):305-310.
2. Uman LS. Systematic reviews and meta-analyses. *Journal of the Canadian Academy of Child and Adolescent Psychiatry*. 2011;20(1):57-59.
3. DerSimonian R, Laird N. Meta-analysis in clinical trials. *Controlled Clinical Trials*. 1986;7(3):177-188.
4. Efthimiou O, Debray TP, van Valkenhoef G, et al. GetReal in network meta-analysis: a review of the methodology. *Research Synthesis Methods*. 2016;7(3):236-263.
5. Biondi-Zoccai G. *Umbrella reviews: evidence synthesis with overviews of reviews and meta-epidemiologic studies*. Springer International Publishing; 2016.
6. Higgins JPT, Whitehead A. Borrowing strength from external trials in a meta-analysis. *Statistics in Medicine*. 1996;15(24):2733-2749.
7. Zhang J, Fu H, Carlin BP. Detecting outlying trials in network meta-analysis. *Statistics in Medicine*. 2015;34(19):2695-2707.



8. Mills EJ, Bansback N, Ghement I, et al. Multiple treatment comparison meta-analyses: a step forward into complexity. *Clinical Epidemiology*. 2011;3:193-202.
9. Zhang J, Carlin BP, Neaton JD, et al. Network meta-analysis of randomized clinical trials: reporting the proper summaries. *Clinical Trials*. 2014;11(2):246-262.
10. Lumley T. Network meta-analysis for indirect treatment comparisons. *Statistics in Medicine*. 2002;21(16):2313-2324.
11. Lu G, Ades AE. Combination of direct and indirect evidence in mixed treatment comparisons. *Statistics in Medicine*. 2004;23(20):3105-3124.
12. Salanti G, Higgins JPT, Ades AE, Ioannidis JPA. Evaluation of networks of randomized trials. *Statistical Methods in Medical Research*. 2007;17(3):279-301.
13. Shim S, Yoon B-H, Shin I-S, Bae J-M. Network meta-analysis: application and practice using Stata. *Epidemiology and Health*. 2017;39:e2017047-e2017047.
14. Schmitz S, Maguire Á, Morris J, et al. The use of single armed observational data to closing the gap in otherwise disconnected

evidence networks: a network meta-analysis in multiple myeloma.

BMC Medical Research Methodology. 2018;18(1):66.



15. Reeves BC, Higgins JPT, Ramsay C, Shea B, Tugwell P, Wells GA. An introduction to methodological issues when including non-randomised studies in systematic reviews on the effects of interventions. *Research Synthesis Methods*. 2013;4(1):1-11.
16. Cameron C, Fireman B, Hutton B, et al. Network meta-analysis incorporating randomized controlled trials and non-randomized comparative cohort studies for assessing the safety and effectiveness of medical treatments: challenges and opportunities. *Systematic Reviews*. 2015;4:147.
17. Grimes DA, Schulz KF. Bias and causal associations in observational research. *The Lancet*. 2002;359(9302):248-252.
18. Dias S, Welton NJ, Sutton AJ, Ades AE. Evidence synthesis for decision making 5: the baseline natural history model. *Medical Decision Making*. 2013;33(5):657-670.
19. Hong H, Chu H, Zhang J, Carlin BP. A Bayesian missing data framework for generalized multiple outcome mixed treatment comparisons. *Research Synthesis Methods*. 2016;7(1):6-22.

- 
20. Dias S, Ades AE. Absolute or relative effects? Arm-based synthesis of trial data. *Research Synthesis Methods*. 2016;7(1):23-28.
21. Rücker G, Schmitz S, Schwarzer G. Component network meta-analysis compared to a matching method in a disconnected network: A case study. *Biometrical Journal*. 2021;63(2):447-461.
22. Phillippo DM, Ades AE, Dias S, Palmer S, Abrams KR, Welton NJ. Methods for population-adjusted indirect comparisons in health technology appraisal. *Medical Decision Making*. 2018;38(2):200-211.
23. Mawdsley D, Bennetts M, Dias S, Boucher M, Welton N. Model-based network meta-analysis: a framework for evidence synthesis of clinical trial data. *CPT: Pharmacometrics & Systems Pharmacology*. 2016;5(8):393-401.
24. Welton NJ, Caldwell DM, Adamopoulos E, Vedhara K. Mixed treatment comparison meta-analysis of complex interventions: psychological interventions in coronary heart disease. *American Journal of Epidemiology*. 2009;169(9):1158-1165.
25. Rücker G, Petropoulou M, Schwarzer G. Network meta-analysis of multicomponent interventions. *Biometrical Journal*. 2020;62(3):808-821.

- 
26. Goring SM, Gustafson P, Liu Y, Saab S, Cline SK, Platt RW. Disconnected by design: analytic approach in treatment networks having no common comparator. *Research Synthesis Methods*. 2016;7(4):420-432.
27. Chaimani A, Higgins JP, Mavridis D, Spyridonos P, Salanti G. Graphical tools for network meta-analysis in STATA. *PloS One*. 2013;8(10):e76654.
28. Zhang J, Chu H, Hong H, Virnig BA, Carlin BP. Bayesian hierarchical models for network meta-analysis incorporating nonignorable missingness. *Statistical Methods in Medical Research*. 2017;26(5):2227-2243.
29. Tu Y-K. Directed acyclic graphs and structural equation modelling. In: Tu Y-K, Greenwood DC, eds. *Modern Methods for Epidemiology*. Dordrecht: Springer Netherlands; 2012:191-203.
30. Mulaik SA. *Linear causal modeling with structural equations*. 1st ed. New York: Chapman and Hall/CRC; 2009.
31. Coates C. Flow-graph solutions of linear algebraic equations. *IRE Transactions on Circuit Theory*. 1959;6(2):170-187.

- 
32. Deo N. *Graph theory with applications to engineering and computer science*. Mineola, New York: Dover Publications; 2016.
33. Fraleigh JB, Bearegard RA. *Linear algebra*. 3rd ed. Taipei: Pearson Education Taiwan Ltd.; 2012.
34. Leon SJ, Pillis LGd. *Linear algebra with applications*. 10th ed: Pearson; 2021.
35. Chen PH, Ho CL, Lin C, et al. Treatment outcomes of novel targeted agents in relapse/refractory chronic lymphocytic leukemia: a systematic review and network meta-analysis. *Journal of Clinical Medicine*. 2019;8(5).
36. Furman RR, Cheng S, Lu P, et al. Ibrutinib resistance in chronic lymphocytic leukemia. *New England Journal of Medicine*. 2014;370(24):2352-2354.
37. Burger JA, Sivina M, Jain N, et al. Randomized trial of ibrutinib vs ibrutinib plus rituximab in patients with chronic lymphocytic leukemia. *Blood*. 2019;133(10):1011-1019.
38. Sharman JP, Coutre SE, Furman RR, et al. Final results of a randomized, phase III study of rituximab with or without idelalisib followed by

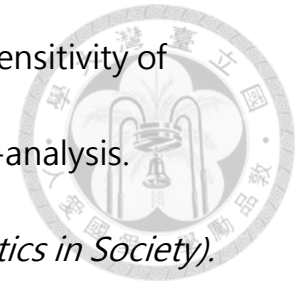


- open-label idelalisib in patients with relapsed chronic lymphocytic leukemia. *Journal of Clinical Oncology*. 2019;37(16):1391-1402.
39. Faggion CM, Jr., Listl S, Frühauf N, Chang HJ, Tu YK. A systematic review and Bayesian network meta-analysis of randomized clinical trials on non-surgical treatments for peri-implantitis. *Journal of Clinical Periodontology*. 2014;41(10):1015-1025.
40. Chaimani A CD, Li T, Higgins JPT, Salanti G. Chapter 11: undertaking network meta-analyses. In: Higgins JPT TJ, Chandler J, Cumpston M, Li T, Page MJ, Welch VA, ed. *Cochrane Handbook for Systematic Reviews of Interventions version 6.3 (updated February 2022)*. Cochrane; 2022.
41. Cox DR. Interaction. *International Statistical Review / Revue Internationale de Statistique*. 1984;52(1):1-24.
42. König J, Krahn U, Binder H. Visualizing the flow of evidence in network meta-analysis and characterizing mixed treatment comparisons. *Statistics in Medicine*. 2013;32(30):5414-5429.
43. Papakonstantinou T, Nikolakopoulou A, Rücker G, et al. Estimating the contribution of studies in network meta-analysis: paths, flows and streams [version 3; peer review: 2 approved, 1 approved with reservations]. *F1000Research*. 2018;7(610).

44. Phillippo DM, Dias S, Ades AE, Didelez V, Welton NJ. Sensitivity of treatment recommendations to bias in network meta-analysis.

Journal of the Royal Statistical Society, Series A (Statistics in Society).

2018;181(3):843-867.



附錄



Part A: R code for drawing a signal-flow graph

This section introduces the main code in the R file. All of the R code and all datasets mentioned in our article are in the associated R file.

1. Packages

Package “netmeta” is used to obtain the M matrix and to draw a standard network plot using an existing function; “igraph” is used to generate a signal-flow graph.

```
## Loading packages

if(!require(netmeta)) {install.packages("netmeta"); library(netmeta)}

if(!require(igraph)) {install.packages("igraph"); library(igraph)}
```

2. Function: Rearrange a matrix

This function is used to rearrange a matrix by sorting the matrix along its 1st column, followed by sorting its 2nd and 3rd columns, and so on. Notice that we use the absolute value to order the matrix. Take an example:

$$\begin{array}{l} r1 \\ r2 \\ r3 \\ r4 \\ r5 \\ r6 \\ r7 \\ r8 \end{array} \begin{array}{l} c1 \\ c2 \\ c3 \end{array} \begin{bmatrix} -4 & 1 & -1 \\ 1 & 1 & -3 \\ 2 & -4 & 1 \\ -4 & -4 & -3 \\ 2 & 3 & 2 \\ 2 & 2 & -3 \\ 1 & 3 & 2 \\ -4 & -4 & -3 \end{bmatrix} \xrightarrow{\text{rearr()}} \begin{array}{l} r8 \\ r4 \\ r1 \\ r3 \\ r5 \\ r6 \\ r7 \\ r2 \end{array} \begin{array}{l} c1 \\ c2 \\ c3 \end{array} \begin{bmatrix} -4 & -4 & -3 \\ -4 & -4 & -3 \\ -4 & 1 & 1 \\ 2 & -4 & 1 \\ 2 & -3 & 2 \\ 2 & 2 & -3 \\ 1 & -3 & 2 \\ 1 & 1 & -3 \end{bmatrix}$$

```
## Rearrange a matrix

rearr <- function(a){

  # Check the object type

  if(!is.matrix(a)){
```



```
stop("The argument 'a' should be a matrix.")
}

# Create a diagonal matrix to record the transformation matrix
tr = diag(1, nrow=nrow(a)); colnames(tr) = rownames(tr) = rownames(a)

# Sort the matrix along its 1st column, followed by sorting its 2nd and 3rd
columns, and so on.

tr <- tr[rev(do.call(order, data.frame(abs(a))))],]

a.new <- tr %*% a

# The argument 'a' is an input, the M matrix.
# M.sort = tr %*% a

return(list(M.sort = a.new, trans.row = tr))
}
```

3. Function: Gaussian Elimination

This function conducts forward Gaussian elimination (Twoside = F) or two-sided Gaussian elimination (Twoside = T). The argument 'a' is an input, the M matrix. The output 'inverse' records the linear combination of each row.

```
## Gaussian Elimination

GaussElimination <- function(a, Twoside = F) {

  # Check the object type

  if(!is.matrix(a)){

    stop("The argument 'a' should be a matrix.")

  }

}
```


nr <- nrow(a) # The number of rows

nc <- ncol(a) # The number of columns

Create a matrix to record the linear combinations

inv.a <- diag(nr)

Create a matrix to record the eliminated matrix

a.eli <- a



Step 1: Forward

Take Figure1b for example:

$$\mathcal{M} = \begin{matrix} & & a_1 & a_2 & a_3 \\ \text{trial 1} & \left[\begin{array}{ccc} 0 & 1 & 0 \end{array} \right] \\ \text{trial 2} & \left[\begin{array}{ccc} 0 & 1 & 1 \end{array} \right] \\ \text{trial 3} & \left[\begin{array}{ccc} 0 & 0 & 1 \end{array} \right] \\ \text{trial 4} & \left[\begin{array}{ccc} -1 & 1 & -1 \end{array} \right] \end{matrix}$$

$$\xrightarrow{\text{rearr}()} \begin{matrix} & & a_1 & a_2 & a_3 \\ t4 & \left[\begin{array}{ccc} -1 & 1 & -1 \end{array} \right] \\ t2 & \left[\begin{array}{ccc} 0 & 1 & 1 \end{array} \right] \\ t1 & \left[\begin{array}{ccc} 0 & 1 & 0 \end{array} \right] \\ t3 & \left[\begin{array}{ccc} 0 & 0 & 1 \end{array} \right] \end{matrix}, \text{trial is abbreviated as } t$$

$$\xrightarrow{a_1} \begin{matrix} & & a_1 & a_2 & a_3 \\ -t4 & \left[\begin{array}{ccc} 1 & -1 & 1 \end{array} \right] \\ t2 & \left[\begin{array}{ccc} 0 & 1 & 1 \end{array} \right] \\ t3 & \left[\begin{array}{ccc} 0 & 1 & 0 \end{array} \right] \\ t1 & \left[\begin{array}{ccc} 0 & 0 & 1 \end{array} \right] \end{matrix} \xrightarrow{\text{rearr}()} \begin{matrix} & & a_1 & a_2 & a_3 \\ -t4 & \left[\begin{array}{ccc} 1 & -1 & 1 \end{array} \right] \\ t2 & \left[\begin{array}{ccc} 0 & 1 & 1 \end{array} \right] \\ t3 & \left[\begin{array}{ccc} 0 & 1 & 0 \end{array} \right] \\ t1 & \left[\begin{array}{ccc} 0 & 0 & 1 \end{array} \right] \end{matrix}$$

$$\xrightarrow{a_2} \begin{matrix} & & a_1 & a_2 & a_3 \\ -t4 & \left[\begin{array}{ccc} 1 & -1 & 1 \end{array} \right] \\ t2 & \left[\begin{array}{ccc} 0 & 1 & 1 \end{array} \right] \\ -t2 + t3 & \left[\begin{array}{ccc} 0 & 0 & -1 \end{array} \right] \\ t1 & \left[\begin{array}{ccc} 0 & 0 & 1 \end{array} \right] \end{matrix} \xrightarrow{\text{rearr}()} \begin{matrix} & & a_1 & a_2 & a_3 \\ -t4 & \left[\begin{array}{ccc} 1 & -1 & 1 \end{array} \right] \\ t2 & \left[\begin{array}{ccc} 0 & 1 & 1 \end{array} \right] \\ -t2 + t3 & \left[\begin{array}{ccc} 0 & 0 & -1 \end{array} \right] \\ t1 & \left[\begin{array}{ccc} 0 & 0 & 1 \end{array} \right] \end{matrix}$$

$$\xrightarrow{a_3} \begin{matrix} & & a_1 & a_2 & a_3 \\ -t4 & \left[\begin{array}{ccc} 1 & -1 & 1 \end{array} \right] \\ t2 & \left[\begin{array}{ccc} 0 & 1 & 1 \end{array} \right] \\ t2 - t3 & \left[\begin{array}{ccc} 0 & 0 & 1 \end{array} \right] \\ t1 - t2 + t3 & \left[\begin{array}{ccc} 0 & 0 & 0 \end{array} \right] \end{matrix} \xrightarrow{\text{rename rows}} \begin{matrix} & & a_1 & a_2 & a_3 \\ r1 & \left[\begin{array}{ccc} 1 & -1 & 1 \end{array} \right] \\ r2 & \left[\begin{array}{ccc} 0 & 1 & 1 \end{array} \right] \\ r3 & \left[\begin{array}{ccc} 0 & 0 & 1 \end{array} \right] \\ r4 & \left[\begin{array}{ccc} 0 & 0 & 0 \end{array} \right] \end{matrix} = \mathcal{M}_F$$

Here we omitted y_i . If we did not omit y_i , it would appear that this system of linear equation does not have a solution:

$$\begin{cases} -y_4 = a_1 - a_2 + a_3 \\ y_2 = a_2 + a_3 \\ y_2 - y_3 = a_3 \\ y_1 - y_2 + y_3 = 0 \end{cases}$$

The reason is that we have not considered weights and residuals when re-writing the matrix as a system of equations. Actually, this is a regression.

```
n0 <- 0 # The number of columns that do not have a leading coefficient
```

```
for (i in 1:nc){
```

```
  # Rearrange the matrix
```

```
  inv.a <- rearr(a.eli)$trans.row %*% inv.a
```

```
  a.eli <- rearr(a.eli)$trans.row %*% a.eli
```

```
  # Find the index of the leading coefficient for the i-th component
```

```
  i.r <- which(a.eli[,i]!=0)
```

```
  # Check whether or not all elements in i-th column are zero
```

```
  i.r <- ifelse(length(i.r)>0,
```

```
    i.r[i.r>(i-n0-1)][1], # Obtain the index
```

```
    NA) # All elements in i-th column are zero
```

```
  if(!is.na(i.r)){
```

```
    if(i.r < nr){ # The 'i.r' row is the last row when 'i.r == nr'
```

```
      # Forward Elimination
```

```
      for (k in seq(i.r+1,nr,1)){
```

```
        lead <- a.eli[k,i]/a.eli[i.r,i]
```

```
        inv.a[k,] <- inv.a[k,] - lead*inv.a[i.r,]
```

```

a.eli[k,] <- a.eli[k,] - lead*a.eli[i,r,]
}
}
# Let the lead coefficient be 1
inv.a[i,r,] <- inv.a[i,r,]/a.eli[i,r,i]
a.eli[i,r,] <- a.eli[i,r,]/a.eli[i,r,i]
}else n0 <- n0 + 1
}

```



Step 2: Backward

Following the above example:

$$\mathcal{M}_F = \begin{matrix} & a_1 & a_2 & a_3 \\ r1 & \begin{bmatrix} 1 & -1 & 1 \end{bmatrix} \\ r2 & \begin{bmatrix} 0 & 1 & 1 \end{bmatrix} \\ r3 & \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \\ r4 & \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \end{matrix}$$

$$\begin{matrix} & a_1 & a_2 & a_3 \\ r1 - r3 & \begin{bmatrix} 1 & -1 & 0 \end{bmatrix} \\ \xrightarrow{a_3} r2 - r3 & \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \\ r3 & \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \\ r4 & \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \end{matrix}$$


$$\begin{matrix} & a_1 & a_2 & a_3 \\ (r1 - r3) + (r2 - r3) & \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \\ \xrightarrow{a_2} r2 - r3 & \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \\ r3 & \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \\ r4 & \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \end{matrix}$$

$$\begin{matrix} & a_1 & a_2 & a_3 \\ r'1 & \begin{bmatrix} 1 & 0 & 0 \end{bmatrix} \\ \xrightarrow{\text{rename rows}} r'2 & \begin{bmatrix} 0 & 1 & 0 \end{bmatrix} \\ r'3 & \begin{bmatrix} 0 & 0 & 1 \end{bmatrix} \\ r'4 & \begin{bmatrix} 0 & 0 & 0 \end{bmatrix} \end{matrix} = \mathcal{M}_F$$

```

if(Twoside){
  for (i in nc:2){

```



```

# Find the index of the leading coefficient for the i-th component
i.r <- which(a.eli[,i]!=0)

# Check whether or not all elements in i-th column are zero
i.r <- ifelse(length(i.r)>0,
              i.r[length(i.r)],
              1) # All elements in i-th column are zero
if(a.eli[i.r,i]!=0 & i.r>1 & sum(abs(a.eli[i.r,1:(i-1)]))==0){
  # Backward Elimination
  for (k in rev(seq(1,i.r-1,1))) {
    lead <- a.eli[k,i]/a.eli[i.r,i]
    inv.a[k,] <- inv.a[k,] - lead*inv.a[i.r,]
    a.eli[k,] <- a.eli[k,] - lead*a.eli[i.r,]
  }
}
}
}

# a.eli = inv.a %*% a

return (list(M.new = a.eli, inverse = inv.a))
}

```

4. Function: Draw a signal-flow graph

This function is used to visualize the M matrix by drawing a signal-flow graph. The argument 'M' is an input, the M matrix.

```

## Draw a graph

DrawGraph <- function(M, RowNames=T, node.size = 30, node.label.cex =
1.4, arrow.label.cex = 1.1){

```

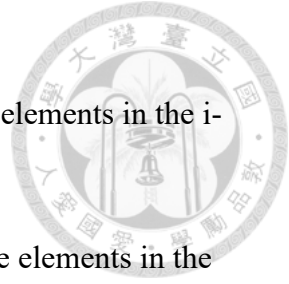


```
# The row names represent components
# The column names represent the study names (or study id), when
'RowNames=T'
if(is.null(rownames(M)) | is.null(colnames(M))) {
  stop("The row or column names is NULL. Please use rownames() or
colnames() to set the names.")
}

## Step 1: Remove the rows that all elements are zero
M <- M[rowSums(abs(M))!=0,]
if(!is.matrix(M)) { # Avoid the type of M becoming vector
  M <- t(as.matrix(M))
  rownames(M) = "1"
}

## Step 2: The algorithm of transformation
# Create a matrix to record the right-hand-side equations
# All yi are omitted
M.rhs <- M
# Create a vector to record the position of the selected node for each row (each
equation)
# It also represents the left-hand-side equations
node.in <- rep(NA, nrow(M))

# The transformation of each row (each equation)
for (i in 1:nrow(M)) {
```



Step 2.1: Calculate the number

```
count.p <- sum(M[i, ] == 1) # The number of the positive elements in the i-  
th row  
count.n <- sum(M[i, ] == -1) # The number of the negative elements in the  
i-th row
```

Step 2.2: Change the sign if needed and select a node put on the left-hand side of the equation

```
if(count.p*count.n == 0){ # Check that all elements have the same sign  
  # If only one element is non-zero, then let that element become positive.  
  # Otherwise, let all elements become negative.  
  if(count.p + count.n == 1) M.rhs[i, ] <- abs(M.rhs[i, ]) else M.rhs[i, ] <- -  
abs(M.rhs[i, ])  
  # Select the left-most non-zero element as the node on the left-hand side.  
  node.in[i] <- which(M.rhs[i, ] != 0)[1]  
}else{ # When the elements have differing signs  
  # If the number of the positive elements is less than  
  # the number of the negative elements, then we change  
  # the signs of all elements (namely, the two sides of  
  # the equation are multiplied by -1 simultaneously;  
  # the left-hand side of the equation is zero at this moment).  
  if(count.p < count.n) M.rhs[i, ] <- -M.rhs[i, ]  
  # After changing the sign,  
  # select the left-most negative element as the node on the left-hand side.  
  node.in[i] <- which(M.rhs[i, ] < 0)[1]  
}
```



Step 2.3: Adjust the coefficients of the equation to -1 (or 1)

```
# After Gaussian eliminating,  
# the element of the selected node is not always -1 (or 1)  
# as the row has more than two non-zero elements;  
# the equation, therefore, is divided by the absolute  
# value of the selected element.
```

```
M.rhs[i,] <- M.rhs[i,]/abs(M.rhs[i,node.in[i]])
```

Step 2.4: Add the variable representing selected node one the two-side

```
# To the two sides of the equation, add the variable representing the selected  
node.
```

```
M.rhs[i,node.in[i]] <- M.rhs[i,node.in[i]] + 1
```

```
}
```

Step 3: Create a table for the "graph_from_data_frame" function

```
# Create an empty table
```

```
name.c <- c("from", "to", "label.coef", "label.study") # The column name for  
the table
```

```
M2graph <- matrix(rep(NA,sum(M.rhs != 0)*length(name.c)), ncol =  
length(name.c))
```

```
M2graph <- as.data.frame(M2graph); colnames(M2graph) <- name.c
```

```
# Fill the table
```

```
index <- 1;for (i in 1:nrow(M.rhs)) { # For the i-th row (equation)
```

```
  # Which nodes will point to the selected node
```



```
from.i <- which(M.rhs[i, ]!=0)
M2graph[index:(length(from.i)-1+index),"from"] <-
colnames(M.rhs)[from.i]
# Which node will be selected
M2graph[index:(length(from.i)-1+index),"to"] <-
rep(colnames(M.rhs)[node.in[i]],length(from.i))
# The coefficient of each arrow in the i-th row
M2graph[index:(length(from.i)-1+index),"label.coef"] <-
as.vector(t(M.rhs[i, from.i]))
# The i-th row sourcing from which study (or trial)
if(RowNames){
  M2graph[index:(length(from.i)-1+index),"label.study"] <-
rep(rownames(M.rhs)[i],length(from.i))
} else { # If the matrix that the user input does not have study names (row
names).
  M2graph[index:(length(from.i)-1+index),"label.study"] <-
rep(i,length(from.i))
}

# The index for the table
index <- index + length(from.i)
}

## Step 4: Create a plot form by "graph_from_data_frame" function
pg <- graph_from_data_frame(d = M2graph,directed = T)
```




Step 5: Adjust the arguments in the "pg" object

Step 5.1: Adjust the arrow color

Set color for the equations generating one arrow

```
E(pg)$color = "black"
```

Create a color bar for the equations generating more than two arrows.

If the number of the equations generating more than two arrows

is more than 7, add more colors.

```
colorbar = palette() # The default color for "col="
```

```
colorbar = colorbar[!palette() %in% "black"] # Remove "black"
```

Find the equations generating more than two arrows

```
count.arrows <- table(M2graph$label.study) # Calculate the number of the  
arrows
```

```
arrows2 <- names(count.arrows[count.arrows>1]) # Whose arrows' numbers  
are more than 2
```

Set the arrow color for the equations generating more than two arrows

```
index.arrows2 <- M2graph$label.study %in% arrows2
```

```
E(pg)$color[index.arrows2] =
```

```
colorbar[as.numeric(as.factor(M2graph$label.study[index.arrows2]))]
```

Step 5.2: Adjust the arrow width

```
E(pg)$width = 1.5
```

Step 5.3: Adjust the arrow curvature

```
arrow.curvature = rep(0, nrow(M2graph)) # Range 0-1
```



```
# Find the potentially overlapping arrows
oa <- as.data.frame(table(M2graph[,c("from","to")]))
oa <- oa[oa$Freq>1,]
# Adjust the arrow curvature
if(nrow(oa)>0){
  for (i in 1:nrow(oa)) {
    index <- M2graph$from == oa$from[i] & M2graph$to == oa$to[i]
    arrow.curvature[index]<-seq(0,1,length.out=oa$Freq[i])
  }
}
# Set the arrow curvature
E(pg)$curved = arrow.curvature

# Step 5.4: Label the arrows
E(pg)$label = as.character(M2graph$label.coef) # Change data type (from
number to character)
E(pg)$label[apply(M2graph, 1, function(x){return(x[1]==x[2])})] = "1+1" #
Point to itself

## Step 6: Draw plot
plot(pg, edge.arrow.size=.5, edge.label.cex = arrow.label.cex ,
      edge.label.color = "black", vertex.shape = "circle",
      vertex.size = node.size, vertex.label.cex = node.label.cex,
      vertex.label.color = "black", vertex.color = 'gray')
}
```

5. Load the data of Example 1

This example dataset can be found in Part B of this supplement. The function “netgraph” could not draw a network plot for a disconnected network, so we wrote a function to draw it (but the size of the nodes and width of the edges are fixed).

```
## Example 1: Chronic Lymphocytic Leukemia

Ex1 <- data.frame(id = c(2, 3, 1, 7, 4, 5, 6, 8, 9, 10),
                  n = c("261 (174/87)", "319 (160/159)", "389 (195/196)",
                        "160(106/54)", "578 (289/289)", "389 (194/195)", "416 (207/209)", "220 (110/110)",
                        "208 (104/104)", "117 (59/58)"),
                  t1 = c("Idelalisib+Ofatumumab", "Duvelisib", "Ibrutinib",
                        "Ibrutinib", "Ibrutinib+Bendamustine+Rituximab", "Venetoclax+Rituximab",
                        "Idelalisib+Bendamustine+Rituximab", "Idelalisib+Rituximab", "Ibrutinib",
                        "Ublituximab+Ibrutinib"),
                  t2 = c("Ofatumumab", "Ofatumumab", "Ofatumumab",
                        "Rituximab", "Bendamustine+Rituximab", "Bendamustine+Rituximab",
                        "Bendamustine+Rituximab", "Rituximab", "Ibrutinib+Rituximab", "Ibrutinib"),
                  lnpsfshr = c(-1.34707364796661, -0.653926467406664, -
                              2.26336437984076, -1.71479842809193, -1.59454929994035, -1.77195684193188, -
                              1.10866262452161, -1.89711998488588, 0.150142658429719, -
                              0.581605805827038),
                  selnpsfshr = c(0.183812794578587, 0.149217754061151,
                                0.180294089952991, 0.274525365514299, 0.155552441740024,
                                0.197242318426909, 0.144212706390322, 0.319582389922288,
                                0.378985732075359, 0.484490089563701))
```

6. Obtain and visualize the M matrix

“discomb” is a ready-made function in package “netmeta”. You also can create the B and C matrices to obtain the M matrix as in our article, if “discomb” cannot work for your data.

```
## Create the M matrix

disEx1 = discomb(lnpfshr, selnpfshr, t1, t2, id, reference.group = "Ofatumumab",
                 sm="md", data = Ex1)

M1 <- disEx1$X.matrix

# Abbreviate the names of components

colnames(M1) <- substr(colnames(M1), start = 1, stop = 3)

# Visualize the M matrix

DrawGraph(M1)

title("Visualization of the M matrix\n(Example 1)")

# Obtain the M.F and M.FB matrices

M.F <- GaussElimination(M1, Twoside = F)$M.new

M.FB <- GaussElimination(M1, Twoside = T)$M.new

# Visualize the M.F and M.FB matrices

DrawGraph(M.F)

title("Visualization of the M.F matrix\n(Example 1)")

DrawGraph(M.FB)

title("Visualization of the M.FB matrix\n(Example 1)")
```

7. How to add interaction

We will use Example 2 to show how to add an interaction to our graph.

```
## Example 2: Hypothetical data

Ex2 <- data.frame(id = c(1, 2, 5, 7, 12, 3, 4, 6, 8, 9, 10),
                  trt1 = c("Laser", "Debridement+Periochip",
                          "Air_abrasive", "Debridement", "Debridement+Antibiotics", "Debridement",
                          "Vector+Chlorhexidine", "Vector", "Debridement", "Laser",
                          "Laser+Chlorhexidine"),
                  trt2 = c("Debridement", "Debridement",
                          "Debridement+Chlorhexidine", "Debridement+Antibiotics",
                          "Debridement+Photodynamic", "Vector", "Debridement+Chlorhexidine",
                          "Debridement", "Antibiotics", "Laser+Antibiotics", "Laser+Antibiotics"),
                  mean1 = c(0.7, 2.18, 0.4, 0.56, 0.19, 0, 0.8, 0, 0.9, 0.19, 0),
                  sd1 = c(1.56, 1.2, 0.7, 0.3, 1.19, 1.39, 0.5, 1.99, 1.99, 1.89, 0.61),
                  n1 = c(16, 30, 15, 14, 20, 17, 21, 11, 13, 37, 14),
                  mean2 = c(0.6, 1.69, 0.5, 1.17, 0.16, 0.4, 0.9, -0.1, 0.7, 0.34, 0.3),
                  sd2 = c(2.51, 1.07, 0.8, 0.27, 1.3, 1.22, 0.8, 3.3, 1.15, 1.75, 1.13),
                  n2 = c(14, 26, 15, 14, 20, 14, 21, 11, 14, 58, 16),
                  mean = c(0.1, 0.49, -0.1, -0.61, 0.03, -0.4, -0.1, 0.1, 0.2, -0.15, -
0.3),
                  se = c(0.775955632015866, 0.303372074167375,
0.274469184669852, 0.107868968131313, 0.394087553723789,
0.469006638429304, 0.205866345916355, 1.16189891596003, 0.631733616833363,
0.386451765918043, 0.326166861328019))

## Obtain the M matrix
```

```

disEx2 = discomb(mean, se, trt1, trt2, id, reference.group = "Antibiotics",
                 sm="md", data = Ex2)

## Add the interaction "Debridement*Chlorhexidine" #####

B2 <- disEx2$B.matrix

C2 <- disEx2$C.matrix

C2.int <- cbind(C2,int = C2[,"Debridement"]==1 & C2[,"Chlorhexidine"]==1)

M2.int <- B2 %*% C2.int

# Abbreviate the names of components

colnames(M2.int) <- substr(colnames(M2.int), start = 1, stop = 3)

```



When we add the interaction to the model, the M matrix is:

	<i>Air</i>	<i>Ant</i>	<i>Chl</i>	<i>Deb</i>	<i>Las</i>	<i>Per</i>	<i>Pho</i>	<i>Vec</i>	<i>int_{Deb*Chl}</i>
<i>id 1</i>	0	0	0	1	-1	0	0	0	0
<i>id 2</i>	0	0	0	0	0	-1	0	0	0
<i>id 5</i>	1	0	-1	-1	0	0	0	0	-1
<i>id 7</i>	0	-1	0	0	0	0	0	0	0
$\mathcal{M}_{int} =$ <i>id 12</i>	0	1	0	0	0	0	-1	0	0
<i>id 3</i>	0	0	0	1	0	0	0	-1	0
<i>id 4</i>	0	0	0	1	0	0	0	-1	1
<i>id 6</i>	0	0	0	1	0	0	0	-1	0
<i>id 8</i>	0	1	0	-1	0	0	0	0	0
<i>id 9</i>	0	-1	0	0	0	0	0	0	0
<i>id 10</i>	0	1	-1	0	0	0	0	0	0

, and the rank of \mathcal{M}_{int} is 9.

```

# Visualize the M matrix

DrawGraph(M2.int)

title("Visualization of the M matrix\n(Example 2 with interaction)")

# Obtain the M.F and M.FB matrices

```

```
M.F <- GaussElimination(M2.int, Twoside = F)$M.new  
M.FB <- GaussElimination(M2.int, Twoside = T)$M.new
```



```
# Visualize the M.F and M.FB matrices
```

```
DrawGraph(M.F)
```

```
title("Visualization of the M.F matrix\n(Example 2 with interaction)")
```

```
DrawGraph(M.FB)
```

```
title("Visualization of the M.FB matrix\n(Example 2 with interaction)")
```



Visualization of the M matrix (Example 2 with interaction)

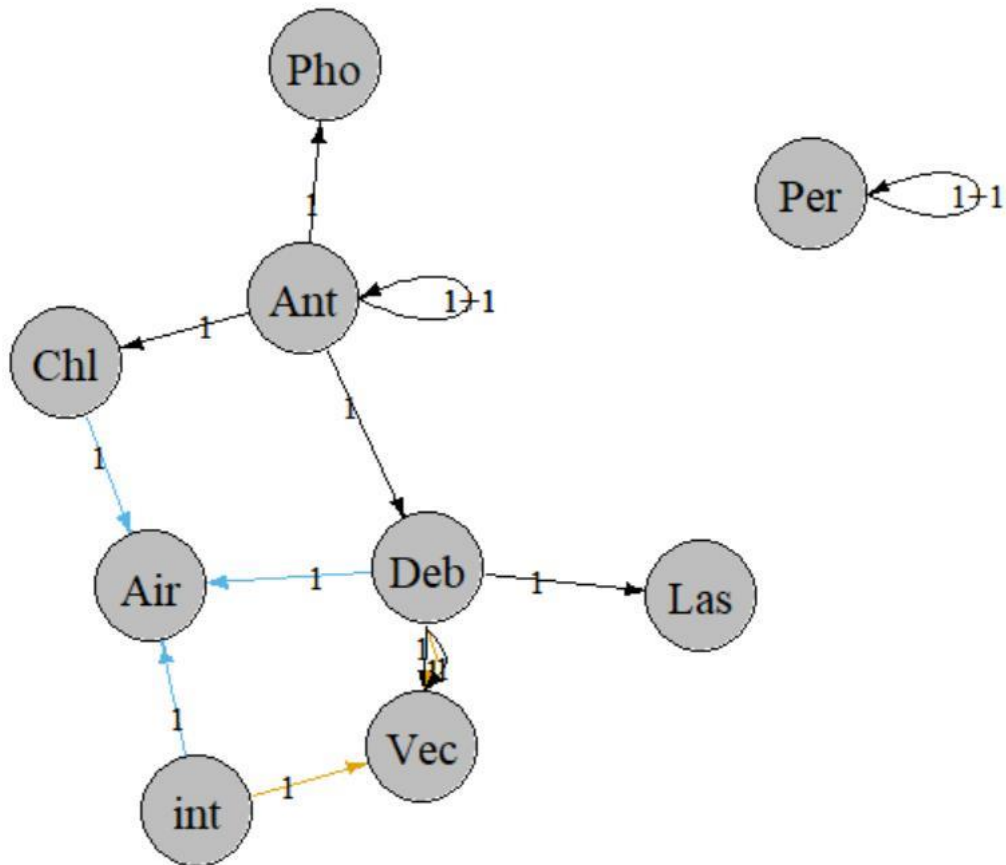


Figure S1. The visualization of \mathcal{M}_{int} in Example 2.

The \mathcal{M}_{int} matrix is the \mathcal{M} matrix with the addition of an interaction between Debridement(Deb) and Chlorhexidine(Chl).

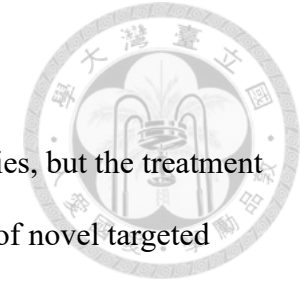
The difference between Figure 8b and Figure S1 is the node ‘int’ and the arrow corresponding to the interaction. The rank of the \mathcal{M}_{int} matrix is equal to the number of components (8+1, an interaction could be seen as a new component). Thus, the graph shows that all components including the interaction are identifiable.



Part B: Dataset for Example 1

The data was extracted from a published network meta-analysis which assessed treatment effects of novel targeted agents for treatment of chronic lymphocytic leukemia.

id	n	t1	t2	lnpfsr	selnpfsr
2	261 (174/87)	Idelalisib+Ofatumumab	Ofatumumab	-1.34707	0.183813
3	319 (160/159)	Duvelisib	Ofatumumab	-0.65393	0.149218
1	389 (195/196)	Ibrutinib	Ofatumumab	-2.26336	0.180294
7	160(106/54)	Ibrutinib	Rituximab	-1.7148	0.274525
4	578 (289/289)	Ibrutinib+Bendamustine+Rituximab	Bendamustine+Rituximab	-1.59455	0.155552
5	389 (194/195)	Venetoclax+Rituximab	Bendamustine+Rituximab	-1.77196	0.197242
6	416 (207/209)	Idelalisib+Bendamustine+Rituximab	Bendamustine+Rituximab	-1.10866	0.144213
8	220 (110/110)	Idelalisib+Rituximab	Rituximab	-1.89712	0.319582
9	208 (104/104)	Ibrutinib	Ibrutinib+Rituximab	0.150143	0.378986
10	117 (59/58)	Ublituximab+Ibrutinib	Ibrutinib	-0.58161	0.48449



Part C: Dataset for Example 2

This example's treatments and components were taken from a published study of non-surgical peri-implantitis therapies, but the treatment effects are hypothetical. The data was extracted from a published network meta-analysis which assesses treatment effects of novel targeted agents for the treatment of chronic lymphocytic leukemia.

id	trt1	trt2	mean1	sd1	n1	mean2	sd2	n2	mean	se
1	Laser	Debridement	0.7	1.56	16	0.6	2.51	14	0.1	0.775956
2	Debridement+Periochip	Debridement	2.18	1.2	30	1.69	1.07	26	0.49	0.303372
5	Air_abrasive	Debridement+Chlorhexidine	0.4	0.7	15	0.5	0.8	15	-0.1	0.274469
7	Debridement	Debridement+Antibiotics	0.56	0.3	14	1.17	0.27	14	-0.61	0.107869
12	Debridement+Antibiotics	Debridement+Photodynamic	0.19	1.19	20	0.16	1.3	20	0.03	0.394088
3	Debridement	Vector	0	1.39	17	0.4	1.22	14	-0.4	0.469007
4	Vector+Chlorhexidine	Debridement+Chlorhexidine	0.8	0.5	21	0.9	0.8	21	-0.1	0.205866
6	Vector	Debridement	0	1.99	11	-0.1	3.3	11	0.1	1.161899
8	Debridement	Antibiotics	0.9	1.99	13	0.7	1.15	14	0.2	0.631734
9	Laser	Laser+Antibiotics	0.19	1.89	37	0.34	1.75	58	-0.15	0.386452
10	Laser+Chlorhexidine	Laser+Antibiotics	0	0.61	14	0.3	1.13	16	-0.3	0.326167