第五章 防災存活路網建構

本研究之防災存活路網之建構，採以可靠、全面、快捷三大目標（圖5.1）。地震災害之不確定性，致使路網遭受破壞時，仍可透過替代通路，確保與災區之連接，此乃路網之可靠；連結所有可能作為避難、收容候選場所之路網，以因應災害來臨時刻之準備，此乃路網之全面；從各類救援單位出發之救援機具，通過最具時間效率之路網，滿足各種緊急活動之需求，此乃路網之快捷。由第一章之防災路網架構，及第二章文獻回顧所獲之心得，並接續第三章之防災路網分析、第四章道路評估；本第五章探討此三大目標，透過採用存活路網、繞路路網、最短路徑樹等路網模型之延伸應用，發展出責任分區路網、系統繞路路網、與互援路網等模型，以作為建構合適之防災路網模型之基礎。最後，並建立此三大目標之相關指標，作為評量的基準；該評估模式可提供評估不同地區存活路網結構之方法。

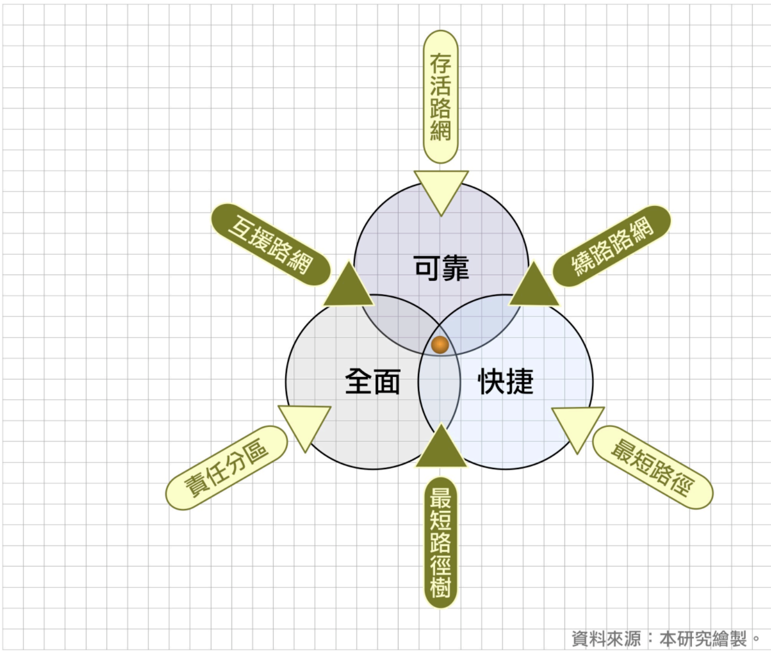


圖5.1 防災路網模型架構

5.1 責任分區路網模型

5.1.1 責任分區之定義

「責任分區」：地震災害期間，時間效率極被關注。由路網所涵蓋的需求點，須有明確指定的救援單位來提供服務；換言之，即便路網錯綜複雜，任一救援單位都非常清楚並了解，它所負責的災點為何、應循走的路線為何。時間將被大量節省，假若每個救援單位在災害發生的時刻，都立即知道該往哪裡去。因此，每一個災點應歸屬於特定一個救援單位；而該救援單位負責的所有災點，便形成其責任分區。分區的目的在提高時間效率；因此歸屬於此救援單位的災點，必低於（或等於）由彼單位來救援將花費的時間成本；此即災點歸屬救援單位之責任分區的基本原則。由時間效率的觀點，透過救援單位區位與最短路徑樹的特性結合，便可獲致救援單位與需求單位之間最有時間效率之路網；首先作出以下相關定義，以進一步闡述。

5.1.2 責任分區之陳述

令 T_{S_i} 為具供給性質之節點 S_i 為根而發展的最短路徑樹，$d_{T_{S_i}}\left(P_{T_{S_i}}\left(D_N\right)\right)$ 表示 T_{S_i} 中從 S_i 至具需求特性之任意末端節點 D_N 最短路徑 $P_{T_{S_i}}\left(D_N\right)$ 的距離成本，$d_{T_{S_i}}\left(D_k, D_N\right)$ 表示 T_{S_i} 中最短路徑途中之任意需求點 D_k 至末端需求節點間 D_N 的距離。

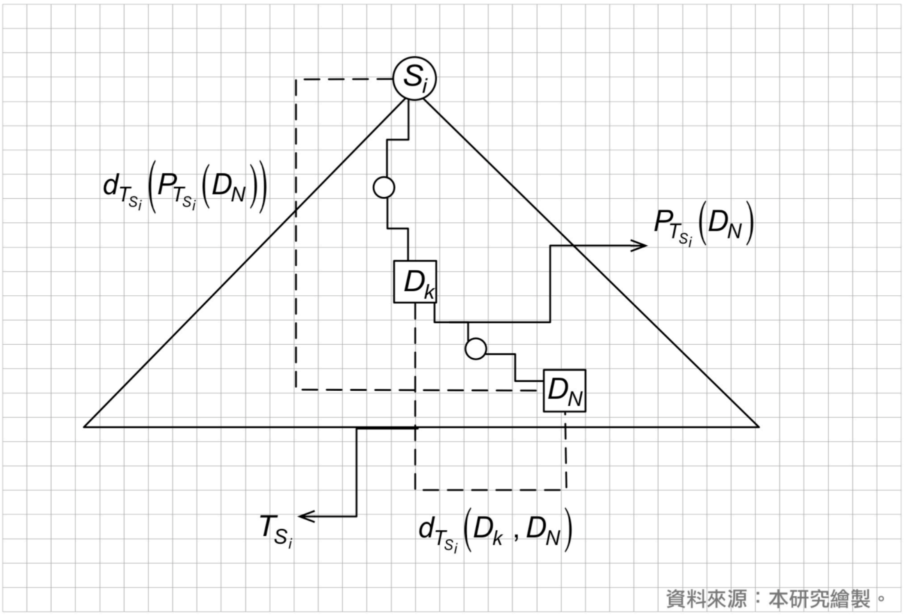


圖5.2 最短路徑樹符號定義

S_i：供給點 i。

T_{S_i}：最短路徑樹，以 S_i 為根。

D_k：最短路徑樹中之中途需求點 k。

D_N：最短路徑樹中之末端需求點 N。

$P_{T_{S_i}}\left(D_N\right)$：最短路徑樹 T_{S_i} 中，由供給點 S_i 到需求點 D_N 之路徑。

$d_{T_{S_i}}\left(P_{T_{S_i}}\left(D_N\right)\right)$：最短路徑樹 T_{S_i} 中，由供給點 S_i 到需求點 D_N 路徑之成本權數。

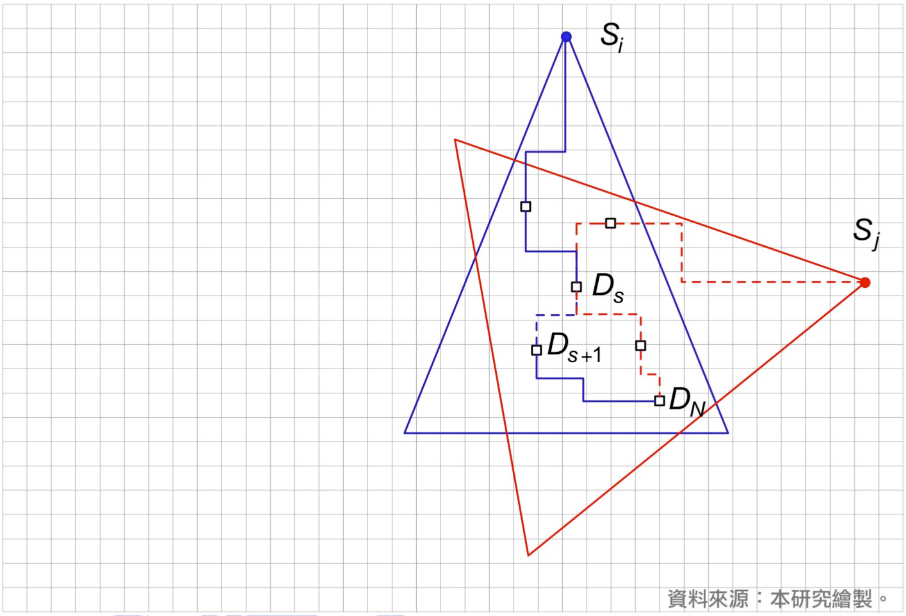
$d_{T_{S_i}}\left(D_k, D_N\right)$：最短路徑樹 T_{S_i} 中，由需求點 D_k 到需求點 D_N 路徑之成本權數。

圖5.2中，三角形範圍內，即為最短路徑樹之範圍；出發中心地 S_i 即為救援單位（供給點），其到達在此範圍內所包含的任何需求點，都具有成本權數最小之特性。路網之中若包含有多個救援單位，則每一救援單位應負責路網之中特定的需求點；此即表示各救援單位所形成的責任分區內，必比其他救援單位到達此區域內的需求點更具效率。透過以下的相關簡理，可以證實這樣的結論：責任分區演算法，可建構出多救援單位最具效率的責任分區路網。

5.1.3 責任分區之原理

1. 簡理一

若某需求點 D_s 分由供給設施 S_i、S_j 分別發展之最短路徑樹涵蓋服務，則 D_s 與任意末端點 D_N 間之任一點 D_{s+1} 必也同時由 S_i、S_j 分別發展之最短路徑樹涵蓋服務。簡言之，若 $D_s \in \left\{P_{T_{S_i}}(D_N), P_{T_{S_j}}(D_N)\right\}$、與 $D_{s+1} \notin P_{T_{S_j}}(D_N), \forall s \in N$，則 $\left|\{D_s\}\right| = 0$。



資料來源：本研究繪製。

圖5.3 簡理一

證明： 令 D_s 為 $P_{T_{S_i}}(D_N)$、$P_{T_{S_j}}(D_N)$ 之共同需求點，則其路徑節點順序可表示為 $P_{T_{S_i}}(D_N) = \langle D_1, D_2, \dots, D_s, \dots, D_k, \dots, D_N \rangle$、與 $P_{T_{S_j}}(D_N) = \langle D_1, D_2, \dots, D_s, \dots, D_k', \dots, D_N \rangle$；由於 $P_{T_{S_i}}(D_N)$、$P_{T_{S_j}}(D_N)$ 是經由最短路徑演算法求得，若 $d_{T_{S_j}}(D_s, D_N) < d_{T_{S_i}}(D_s, D_N)$，則 $P_{T_{S_i}}(D_N)$ 應修正為 $P_{T_{S_i}}(D_N) = \langle D_1, D_2, \dots, D_s, \dots, D_k', \dots, D_N \rangle$ 才合理，反之亦然，因此可得 $D_k = D_k'$；若要能同時滿足簡理一之前提條件（D_s 之定義）：$D_s \in \left\{P_{T_{S_i}}(D_N), P_{T_{S_j}}(D_N)\right\}$、$D_{s+1} \notin P_{T_{S_j}}(D_N), \forall s \in N$，由以上推論結果可證實該 D_s 並不存在，因此 $\left|\{D_s\}\right| = 0$。

2. 簡理二

若某需求點 D_m 分由供給設施 S_i、S_j 分別發展之最短路徑樹涵蓋服務，且 D_m 與供給設施 S_i 間之前一點 D_{m-1} 未由 S_j 發展之最短路徑樹涵蓋服務，則到達某一末端節點 D_N 的路徑上，這樣的 D_m 必定不多於一個。簡言之，若 $D_m \in \left\{P_{T_{S_i}}(D_N), P_{T_{S_j}}(D_N)\right\}$、與 $D_{m-1} \notin P_{T_{S_j}}(D_N), \forall m \in N$，則 $\left|\{D_m\}\right| \le 1$。

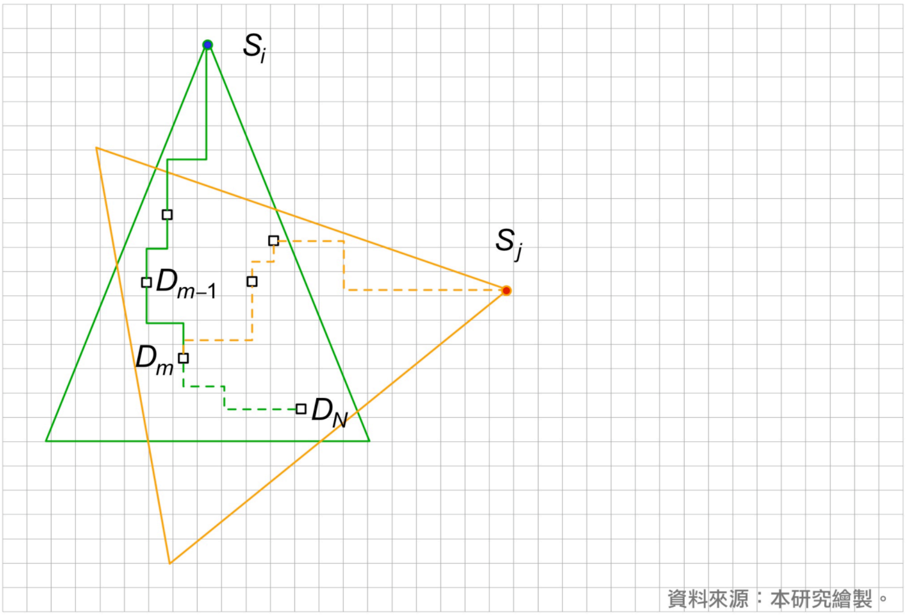


圖5.4 簡理二

證明： 令 D_m 為 $P_{T_{S_i}}(D_N)$、$P_{T_{S_j}}(D_N)$ 之共同需求點，則其路徑節點順序可表示為 $P_{T_{S_i}}(D_N)=\langle D_1, D_2,\dots,D_m,\dots,D_k,\dots,D_N\rangle$、與 $P_{T_{S_j}}(D_N)=\langle D_1', D_2',\dots,D_m,\dots,D_k,\dots,D_N\rangle$；若 $P_{T_{S_i}}(D_N)\subset P_{T_{S_j}}(D_N')$，即 $D_j=D_j', D_{j+1}=D_{j+1}',\dots,D_{m-1}'=D_{m-1}, D_m=D_m,\dots$，則 $|\{D_m\}|=0$；若 $P_{T_{S_i}}(D_N)\cap P_{T_{S_j}}(D_N)=\{D_k\}, k\in K\subset N, |K|>1$，由簡理一可知 $|\{D_s\}|=0$，表示兩個不同的最短路徑樹中之任一最短路徑若合併於 D_m，則從 D_m 以後的路徑就會一直重疊直到 D_N 為止，而不再分離，因此 $|\{D_m\}|=1$；所以 $|\{D_m\}|\le 1$。

3. 簡理三

若需求節點 D_k 較靠近供給設施 S_i 則所有在 D_k 與任一末端節點 D_N 之間的需求節點，皆靠近供給設施 S_i。意即，若 $d_{T_{S_i}}\left(P_{T_{S_i}}(D_k)\right)=\min_{\forall S_j\in S}\left\{d_{T_{S_j}}\left(P_{T_{S_j}}(D_k)\right)\right\}$，則 $d_{T_{S_i}}\left(P_{T_{S_i}}(D_{k+1})\right)\le \min_{S_i\ne S_j,\forall S_j\in S} d_{T_{S_j}}\left(P_{T_{S_j}}(D_{k+1})\right)$。

（1） 前導推論：$P_{T_{S_i}}(D_N)=\langle S_i, D_1, D_2,\dots,D_k,\dots,D_N\rangle$ 為最短路徑樹 T_{S_i} 中供給點 S_i 到達需求點 D_N 之路線，若 $d_{T_{S_i}}\left(P_{T_{S_i}}(D_N)\right)<d_{T_{S_j}}\left(P_{T_{S_j}}(D_N)\right)$，則 $d_{T_{S_i}}\left(P_{T_{S_i}}(D_k)\right)<d_{T_{S_j}}\left(P_{T_{S_j}}(D_k)\right), \forall k\in N$。

證明：

A. 情況1

若 $P_{T_{S_i}}(D_N)\subset P_{T_{S_j}}(D_N')$，則情形為 $P_{T_{S_j}}(D_N)=\langle S_j, D_1', D_2',\dots,D_N', S_i, D_1, D_2,\dots,D_k,\dots,D_N\rangle$，則 $d_{T_{S_j}}\left(P_{T_{S_i}}(D_k)\right)=d_{T_{S_j}}\left(P_{T_{S_j}}(S_i)\right)+d_{T_{S_i}}\left(P_{T_{S_i}}(D_k)\right), \forall k\in N$，因距離權數為正，故 $d_{T_{S_j}}\left(P_{T_{S_j}}(S_i)\right)>0$，所以 $d_{T_{S_i}}\left(P_{T_{S_i}}(D_k)\right)<d_{T_{S_j}}\left(P_{T_{S_j}}(D_k)\right), \forall k\in N$。

B. 情況2

若$P_{T_{S_i}}(D_N)\cap P_{T_{S_j}}(D_N)=\{D_N\}$，則$d_{T_{S_i}}(P_{T_{S_i}}(D_k))<d_{T_{S_j}}(P_{T_{S_j}}(D_k))=\infty, \forall k, k\neq N$，且已知$d_{T_{S_i}}(P_{T_{S_i}}(D_N))<d_{T_{S_j}}(P_{T_{S_j}}(D_N))$，所以$d_{T_{S_i}}(P_{T_{S_i}}(D_k))<d_{T_{S_j}}(P_{T_{S_j}}(D_k)), \forall k\in N$。

C. 情況3

若$P_{T_{S_i}}(D_N)\cap P_{T_{S_j}}(D_N)=\{D_k\}, k\in K\subset N, |K|>1$，則由簡理一、簡理二可知$|\{D_m\}|=1$；故可將情況3分成情況1和情況2之組合：令$D_M$即為合併點，$k\leq M$時為情況2，$k>M$時為情況1；因此$d_{T_{S_i}}(P_{T_{S_i}}(D_k))<d_{T_{S_j}}(P_{T_{S_j}}(D_k)), k\leq M$、$d_{T_{S_i}}(P_{T_{S_i}}(D_k))<d_{T_{S_j}}(P_{T_{S_j}}(D_k)), k>M$，所以$d_{T_{S_i}}(P_{T_{S_i}}(D_k))<d_{T_{S_j}}(P_{T_{S_j}}(D_k)), \forall k\in N$。

（2） 簡理三證明

A. 基礎步驟

由簡理一可知上述結論於$|S_i|=2$時成立。

B. 歸納步驟

假設$|S_i|=n$於$n\geq 2$時亦成立，即於n個樹之中存在$P_{T_{S_i}}(D_N)\in T_{S_i}$使得$d_{T_{S_i}}(P_{T_{S_i}}(D_N))\leq \min_{S_i\neq S_j, \forall S_j\in S} d_{T_{S_j}}(P_{T_{S_j}}(D_N))$。當$|S_i|=n+1$，由簡理三之假設已知$d_{T_{S_i}}(P_{T_{S_i}}(D_N))<d_{T_{S_{n+1}}}(P_{T_{S_{n+1}}}(D_N))$；則由前導推論可獲知$d_{T_{S_i}}(P_{T_{S_i}}(D_k))<d_{T_{S_{n+1}}}(P_{T_{S_{n+1}}}(D_k))$。

5.1.4 責任分區之演算法

簡理三係由簡理一、簡理二推論而來，而且為責任分區之重要基礎。簡理三說明了，若想同時由多個供給節點自行發展最短路徑樹，則可透過快速將子樹歸給某一責任分區之方法，而無須重複節點距離的確認，此法將大幅提升演算效率，並快速區分出最佳的責任分區。而在各責任分區之結果中，達到每個需求節點所受到供給節點提供之服務、其權數成本必定小於（或等於）由其他供給節點所提供服務之目標。

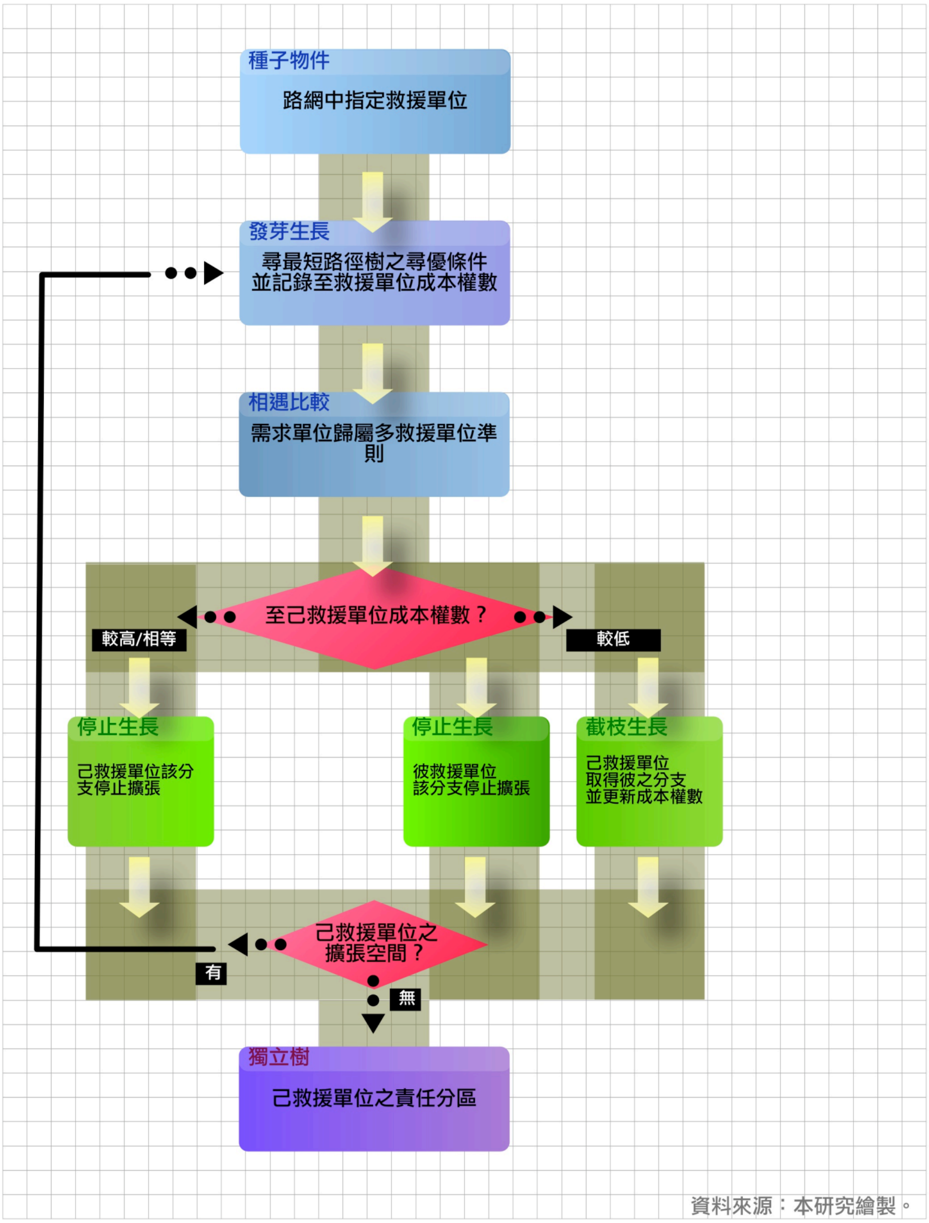


圖5.5 責任分區演算法概念

責任分區演算法概念如圖5.5。首先，將所有救援單位（供給點）當作多個種子物件，投入由第三章所產出的基礎路網之中；其次，採以最短路徑之尋優法使每一個種子發芽擴張，以觸及需求點，並記錄該點與種子之間的成本權數。當不同種子發芽產生之樹，於某一需求點相遇時，則進行優劣比較。基於簡理三，若相遇之需求點離某種子成本權數較高（或相同）時，則樹就不再繼續往此處生長；反之，若成本權數較低時，則除了將該需求點歸於己，原先該需求點生長過的子樹，也都截枝於新的樹，對新的種子更新成本權數後繼續生長。被截枝端則停止生長。此程序不斷進行，直到基礎路網結構已沒有種子發芽生長的空間，責任分區就確立完成。

5.2 系統繞路路網模型

本研究採用存活路網模型，做為防災路網規劃的主要方法之一；此乃由於地震造成路網破壞之不確定性。存活路網係在探討路網連結度之問題，本節即以道路之節線觀點，透過該模型之延伸應用來尋找替代路徑的方法，因此構建之路網，為因應道路可能被破壞的情況。

存活路網模型之防災路網，以口語表達之意，就是保留了路網具有替代道路的特性；換而言之，面對預期性的路網破壞，已有使用替代道路做繞路的準備；因此，防災路網規劃，如能將繞路成本的特性納入考量，並深化於存活路網模型之中，將使得路網更具效率。繞路是基於原先路線無法使用，而採行不同替代路徑的觀念。所謂替代路徑，係指整條原路線的過程中，任一途中地點，發生轉換至其它路徑的情形。假若，無論災害發生地點在原路線上的任一位置，皆可透過特定替代路徑重新連接兩點，此即前述之存活路網模型；然而，繞路除了是有替代路徑之功能外，仍須要考量繞路的成本。倘若繞路之替代路徑為成本極高之繞路路徑，即使仍可以連結供需節點，但就救災的時效性而言，也失去了繞路的意義。因此，繞路還須含有最短路徑的概念。而繞路模型提供了方法，以指出路網中某一路段之毀損，將造成最可觀的繞路成本；同理，該模型也可尋找出繞路成本最小之替代路徑，以便提高路網的效率，此即為繞路模型對防災路網貢獻之處。

本研究所採用之繞路模型，係由Nardelli等學者所提出。該研究探討最短路徑樹中任意兩點間之路線的繞路問題。若該兩點間之最短路徑中任一路段無法通行時，則必須從原路網中，找尋另一路徑來連通此兩點，此即為繞路；而某一路段無法通行，並造成原來兩點間之繞路最遠，此即為繞路之繞路敏感路段。本研究則利用此觀念，來構建系統繞路模型。

5.2.1 系統繞路之定義

「系統繞路」：任一路段被破壞的情形下，單一救援單位透過某一橋段，以繞達全部需求場所之路徑。由於地震災害破壞之不確定性高，若任一路段遭受地震破壞後，能立即明瞭取代之橋段之所在，則防災路網將依然明確，並能將責任分區內之救援單位，有系統地與全部的需求場所連接起來。

5.2.2 系統繞路之陳述

存活路網為探討路網失靈之議題，成本最小的存活路網結構就是二邊連通圖，即移除任一路段，仍可透過替代路徑使原來任意兩個節點相互連通。然而，此時連通的路徑，並不隱含繞路成本趨小化的概念；因此，此存活路網結構下的繞路成本，有可能與先前未繞路之路徑成本高出許多。系統繞路模型秉持兩個觀點：1.預期道路可能被災害破壞、然未設定何路段將失靈的情形下，考量繞路成本趨小化；2.未預先設定何需求點於災時會有真實需求的立場下，考量供給點與整體需求點之間的總繞路成本趨小化。

前述之責任分區路網模型，具有未需繞路時，最具時間效率之結構。其最短路徑樹是規劃單一供給點至其範圍可及所有需求點最有效率的路網規劃方法。然而，樹結構有「樹的任一節線都是割線（cut edge）」的特性，因此沒有路段的容錯（fault tolerance）能力。所以，任一路段被破壞，都會導致樹中一個節點以上必須繞路的情形發生；而這些需要透過繞路才能與供給點連通之節

點，其個別繞路成本的總合，本研究定義為「系統繞路成本」。系統繞路模型，即為在最短路徑樹結構下，以系統繞路成本最小化之尋優方式，應用繞路模型所構建之存活路網。

5.2.3 系統繞路之原理

系統繞路模型如圖5.6，以及下述符號與定義說明。

假若令T為圖形G的最短路徑樹，其根（root）為r；$d(s,r)$表示G中任一節點s至根r最短路徑$P_G(s,r)$的距離。假若除去某一邊e，令$U_{T_r,e}$為擁有根的子樹（subtree），而$L_{T_r,e}=T_r-U_{T_r,e}-e$表示為剩下的樹。根據Nardelli等人的研究，在e移除後，$U_{T_r,e}$中所有的點與r之間的距離都不會改變，然而$L_{T_r,e}$中的點與r之間的距離卻有可能增加。令$B_{T_r,e}(m,a)$為$L_{T_r,e}$與$U_{T_r,e}$之間的橋段（m在$L_{T_r,e}$而a在$U_{T_r,e}$），$d\left(B_{T_r,e}(m,a)\right)$表示橋段的成本；則此時$r$即可透過含有$B_{T_r,e}(m,a)$的路徑到達$L_{T_r,e}$中的任何一點，而且不再需要經過$e$。值得注意的是，此時$L_{T_r,e}$中所有的點，都一定要有屬於$T_r$以外的邊，才有能力與$r$連結。這裡我們定義了一個指標，來衡量$L_{T_r,e}$中所有的點繞路到達$r$的總成本，也就是系統繞路成本：

$$SDC_{T_r-e}=\sum_{s,m\in L_{T_r,e},\,a,r\in U_{T_r,e}}\left\{d(s,m)+d\left(B_{T_r,e}(m,a)\right)+d(a,r)\right\} \qquad (5\text{-}1)$$

系統繞路成本為用來衡量由於e所引起所有節點的繞路成本總合。然而實際上所謂「所有節點」僅指$L_{T_r,e}$中的需求節點，因為$U_{T_r,e}$中的需求節點無需繞路即可到達。然而，很明顯地，如果每一個$L_{T_r,e}$中的節點為達最短繞路的目的，都想擁有自己的橋段$B_{T_r,e}(m,a)$來到達r，則這將是可觀成本的路網結構，在此本研沿用Tarjan橋段的概念，究提出「共同橋段」的做法以簡化路網結構，當e被移除，利用$L_{T_r,e}$中原來的路網結構，讓所有其中的節點依循原路網結構，到達其中特定的一個集合點m，並透過相同的橋段$B_{T_r,e}(m,a)$來到達r，則成本將可大幅降低。而此時的繞路成本問題便簡化為共同橋段的選擇問題。本研究中，將這樣的問題稱作為系統繞路問題（SD Problem）。系統繞路問題具有「共同集合點、共同橋段」之特性，因而在此訂了另一個參數來衡量共同集合的成本，也就是聚集成本（Merging Cost）：

$$MC_G(m)=\sum_{s\in G}d(s,m) \qquad (5\text{-}2)$$

聚集成本為$L_{T_r,e}$中所有節點到達其中特定聚集點m的總距離；因此，若令$V_{L_{T_r,e}}$表示為$L_{T_r,e}$的節點集合，則前述之系統繞路成本SDC_{T_r-e}就可以轉換成：

$$SDC_{T_r-e}=MC_{L_{T_r,e}}(m)+\left|V_{L_{T_r,e}}\right|\times\left[\,d\left(B_{T_r,e}(m,a)\right)+d(a,r)\,\right] \qquad (5\text{-}3)$$

其中第一項表示為$L_{T_r,e}$節點在聚集點m的聚集成本，第二項則表示由聚集點m到根r的進入成本，其中包含了橋段的權數與進入點a至r的距離加總，再乘上$L_{T_r,e}$中的節點數目。

系統繞路的成本為子樹$L_{T_r,e}$中的每一個節點，繞路至根r的距離總合。上述方式可將系統繞路成本分為聚集成本、進入成本兩項；從其中可以發現，聚集成本若要降低，則聚集點傾向靠近子樹的中心位置；然而，$L_{T_r,e}$中的節點數量亦左右著進入成本，其可視為單一節點進入成本的權數（此因每一個節點的進入成本皆相同之故），因此子樹的需求節點數量越多，進入成本就越顯得重要，

因而決定了進入點的位置；而由於聚集點、進入點彼此同屬路網中，共同橋段的兩個鄰點；因此可預期到，聚集點與進入點必為緊密相互影響方能確立的情形。

聚集成本並非在構建樹的時候就已獲得，直覺的計算方式，則是在獲知破壞路段位置後，由其子樹每一個節點為出發點，進行深度優先搜尋、或再一次透過最短路徑演算來獲得；但事實上並不需如此大費周章。既然初始已獲得樹的結構，即表示已熟知樹結構中任何一點到達根的路徑；透過這個特性，並利用交集、聯集的演算，就可以快速獲得聚集成本，如式（5-4）。透過上述方式以及圖5.6之說明，則可容易而快速的獲得聚集成本。

$$MC_{L_{T_r,e}}(m)=\sum_{s\in L_{T_r,e}}\left\{\sum_{e_i\in P_G(m,r)\cup P_G(s,r)} w(e_i)-\sum_{e_j\in P_G(m,r)\cap P_G(s,r)} w(e_j)\right\} \tag{5-4}$$

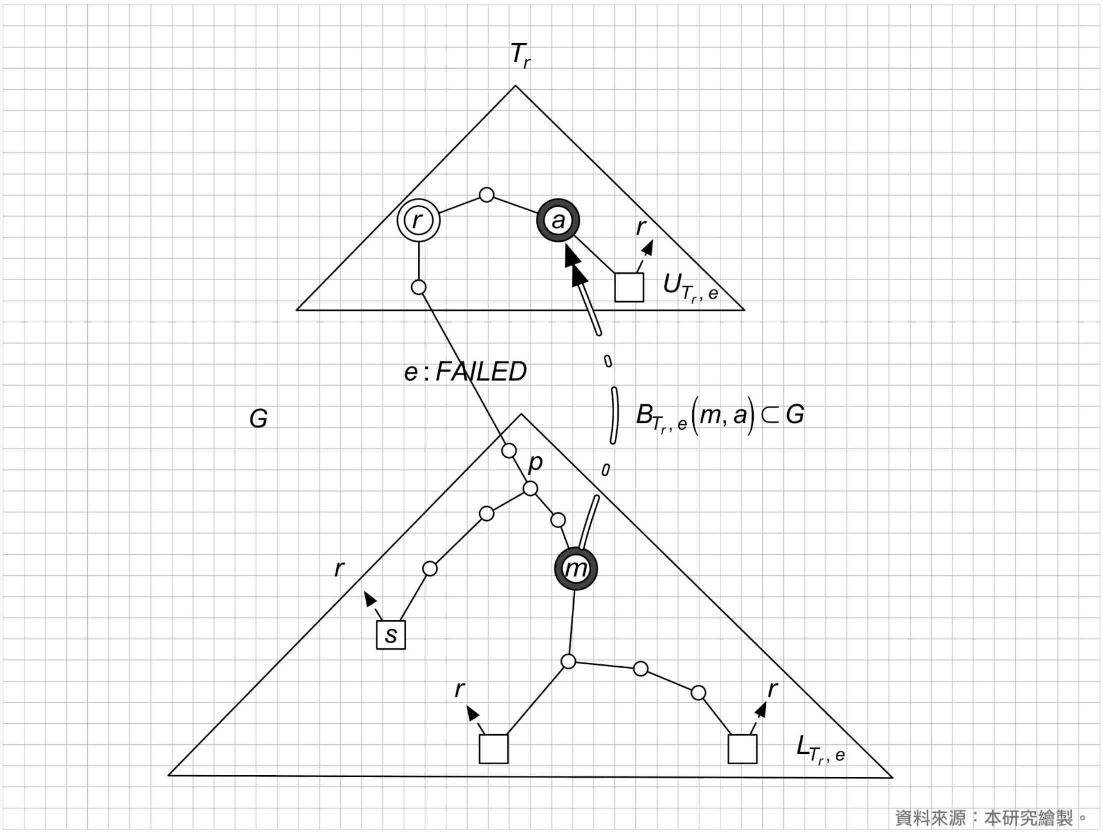


圖5.6 系統繞路模型

因此，成本最小化之系統繞路問題即成為下式：

$$MSDC_{T_r-e}=\min_{m\in L_{T_r,e},\, a\in U_{T_r,e}} SDC_{T_r-e} \tag{5-5}$$

而滿足成本最小化之系統繞路問題之繞路路段即為：

$$MSDE_{T_r-e}=\left\{ B_{T_r,e}(m,a) \,\middle|\, m\in L_{T_r,e}, a\in U_{T_r,e}, SDC_{T_r-e}=MSDC_{T_r-e}\right\} \tag{5-6}$$

因此，探討最短路徑樹中每一路段預期破壞的情形，則所有的繞路路段即為：

$$DE_{T_r} = \bigcup_{e \in T_r} MSDE_{T_r - e} \qquad (5\text{-}7)$$

最後，平衡路網結構成本、以及繞路成本之路網，亦即滿足系統繞路成本最小的二連通圖，即為：

$$2ECON_{T_r} = T_r \cup DE_{T_r} \qquad (5\text{-}8)$$

5.2.4 系統繞路之演算法

系統繞路演算法包含下列步驟：

1. 步驟一：確認基礎路網

由確認之供給點為根，發展最短路徑樹 T_r，並記錄供給點至任一需求點之間的最短路徑。

2. 步驟二：基礎路網路段毀損模擬

掃描基礎路網之最短路徑樹的所有邊，依次視為毀損路段，將原基礎路網分為包含供給單位之上半樹 $U_{T_r, e}$ 、以及包含需求單位之下半樹 $L_{T_r, e}$ 。

3. 步驟三：尋找系統繞路成本最小之連接橋段

透過式（5-3）來尋找系統繞路成本，並求取擁有最小系統繞路成本之橋段 $B_{T_r, e}$ 。

4. 步驟四：系統繞路路網之構成

掃描基礎路網所有邊後，所有繞路橋段之集合，與基礎路網之集合，即為最小系統繞路路網 $T_r \cup \left(\bigcup_{e \in T_r} B_{T_r, e} \right)$ 。

5.3 互援路網模型

責任分區路網模型，提供所有需求點與救援單位之間最快捷的路徑；系統繞路模型，確保任一路段毀損之時，責任分區內之路網具有最佳橋段使總體繞路成本最小。而互援路網模型，以最具效率的方式，將自己責任分區內之需求點，橋接至不同責任分區的救援單位。

5.3.1 互援路網之定義

「互援」：地震災害的特性為不確定性，責任分區中的救援單位，因救援量能不足、或自身受到地震損害，致無法擔起自負責任的情況時，此時須由其分區外的其他救援單位互相支援，以替代自身的救援責任。救援能力將被確保，假若每個救援單位在災害發生的時刻，都立即知道該循什麼路線、去支援哪些區域。