最短径路アルゴリズムに基づく斜面安定解析
―Janbu法へのDijkstra法の応用―

徳島大学工学部　正　山上　拓男
徳島大学工学部　正　植田　康宏
中電技術コンサルタント　正　○小山　正之

1．まえがき： 円弧すべりを対象とした安全率算定式を用いて、与えられた斜面に存在する臨界すべり面を探索することは極めて容易である。すなわち円弧の中心と半径を適当に変化させてやれば、電算機の助けのもとになんなく臨界円を決定できることは周知の通りである。ところが非円形すべり面に基づく安全率算定式を用いて安定解析を行わんとすれば一変して状況は複雑なものとなる。それは仮定すべり面の形状に無限の可能性が存在するからである。そこで筆者らは非円形すべり面理論としてなじみ深い Janbu法を用い、動的計画法(DP)と組み合わせて効率よく臨界すべり面が得られる手法を発表してきた[1]。本報告ではDPに代って、最短径路アルゴリズムの一つであるDijkstra（ダイクストラ）法を導入した安定解析法を提案し、さらに2、3の解析例についてDPを用いた場合と演算効率の面で比較を行う。

2．Dijkstra法に基づく Janbu安定解析アルゴリズム： 本解析に用いた Janbu（簡便）法の安全率算定式を式1)に示す。式1)を式2)のように一般表示しておく。このとき新たな変数Gを持ち込んで式3)を定義すれば、式2)の Fs を最小にすることと式3)のGを最小にすることが全く同等であることが知られている[2]。つまり、すべり面の選び方によって式2)のFsが様々に変化するが、求める臨界すべり面は言うまでもなくFsを最小にするものであり、そのときG自身も最小となっているのである。理論構成上、式2)を直接最小化することはできないので、これに代ってGの最小化を図るのである（式4)）。ただし、式3)中のFsは未知であるから、仮定値から出発して繰り返し計算を行わなければならない。

$$F_s = \frac{1}{\Sigma W_i \cdot \tan\alpha_i} \Sigma \frac{C' \cdot \ell_i \cdot \cos\alpha_i + (W_i - u_i \cdot \ell_i \cdot \cos\alpha_i)\ \tan\Phi'}{\cos^2\alpha_i\ (1 + \tan\alpha_i \cdot \tan\Phi'/F_s)} \quad \cdots 1)$$

$$F_s = \Sigma R_i / \Sigma T_i \quad \cdots 2)$$

ここに

$$R_i = \frac{C' \cdot \ell_i \cdot \cos\alpha_i + (W_i - u_i \cdot \ell_i \cdot \cos\alpha_i)\ \tan\Phi'}{\cos^2\alpha_i\ (1 + \tan\alpha_i \cdot \tan\Phi'/F_s)}$$

$$T_i = \Sigma W_i \cdot \tan\alpha_i$$

$$G = \Sigma (R_i - F_s \cdot T_i) \quad \cdots 3)$$

$$\min G = \min\Sigma (R_i - F_s \cdot T_i) \quad \cdots 4)$$

さて、 Janbu法ではあらかじめ与えられた斜面にスライス分割線となる適当な数の鉛直線を設定する必要があるが、Dijkstra法による臨界すべり面探索には、この鉛直線とその線上に適当に設けた節点を利用する。いま、図1に示すように解析領域を n+1本の鉛直線で分割し順に1､2､…､n+1の番号を付す。またそれらの線上に設けた節点にも各線ごとに 1,2,… の番号を付ければ、任意の引き続く2本の鉛直線 i, i+1 上の任意の節点 j及び kをそれぞれ点(i, j), (i+1, k)と表すことができる。そして、これらの2点を結ぶ軌道$\overline{jk}$に注目し、$\overline{jk}$を可能なすべり面の一部とみなす。このとき四辺形ajkbをスライスと考え、これに対するRi、TiをJanbu法の安全率算定式から求めれば、点(i, j)から点(i+1, k) に至る式3)のGの変化量 DGi(j,k)： DGi(j,k) = Ri+ Fs·Ti　… 5)　が得られる。このDGをDijkstra法の評価関数（これを「重み」という）として解けば、Gが最小となる経路、すなわち臨界すべり面を求めることができる。

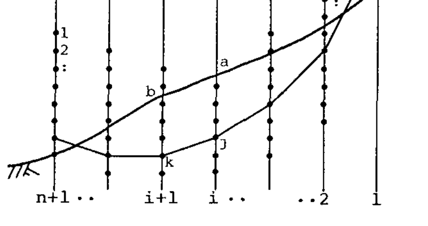

図1　Dijkstra法による臨界すべり面の決定

3．Dijkstra法を適用する際の問題点とその処置： ネットワークの例を図2に示す。数理計画法では、与えられたネットワーク内の任意の2節点間(1→N)で、ある評価関数の値(Σ d ij)を最適

（最大または最小）にするような径路を求める問題を最短径路問題と呼んでいる。Dijkstra法はグラフ理論の中の最短径路問題を扱う一手法である。Dijkstra法の他にも様々な最短径路アルゴリズムが提案されているが、この方法が最も効率がよい[3]ということで本解析に採用した。ただし、Dijkstra法には、「ネットワーク内のすべての枝の重み（2節点間の評価関数ｄij）が非負でなければならない。」という制約条件がある。本解析においては、「枝の重み」に相当するものは、言うまでもなく前節で述べた節点間のＧの変化量DG（式5)）である。ところが DG は軌道の選び方に応じて変化する値であり負の値をとることもありうる。したがってDijkstra法をそのまま本解析に持ち込むことはできない。そこで筆者らは、本解析で扱うような特別なネットワークについてのみ適用可能なある操作を行い、すべてのDGの値をいったん非負の状態に変換したのちDijkstra法を適用するという工夫を試みた。すなわち、図3に示すようないくつかの規則正しく連なるcolumn（本研究では、隣り合う2本のスライス分割線間をcolumnと呼んでいる）によって分けられたネットワークにおいて、まず各columnごとにDGの最小値を求め、もしそれが負となるcolumnが存在すれば、そのcolumn内のすべての枝にそのcolumnのDGの最小値の絶対値を加えるという操作である。紙数の制約上この方法が許される根拠については割愛する。また、これまで触れなかったが、図1あるいは図3に示したようにすべり面の両端が未知な場合でもなんら問題なく解が得られるということを付け加えておく。

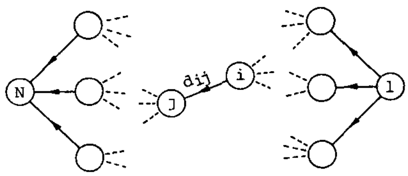

図2　ネットワークの例（1）

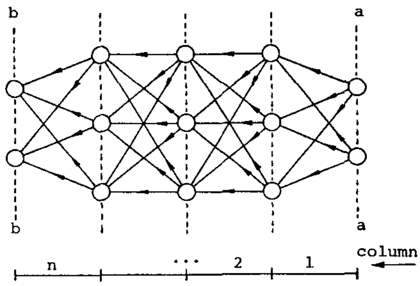

図3　ネットワークの例（2）

4．適用例および結言： 本解析法の適用例として、図4、5に示す斜面を採用した（物性値は表1参照）。図中に、臨界すべり面と各反復段階のすべり面を示す。解析結果はDPによるそれと一致した。また、表2に示すようにDijkstra法のCPU time（徳島大学情報処理センターFACOM M-360）はDPに比べわずかに多いが、両者ともほぼ同程度の演算時間であった。

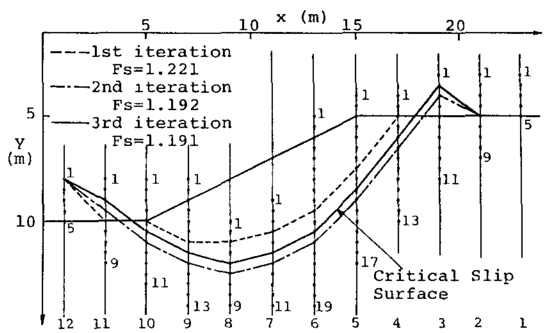

図4　解析例1（Dijkstra法を用いた場合）

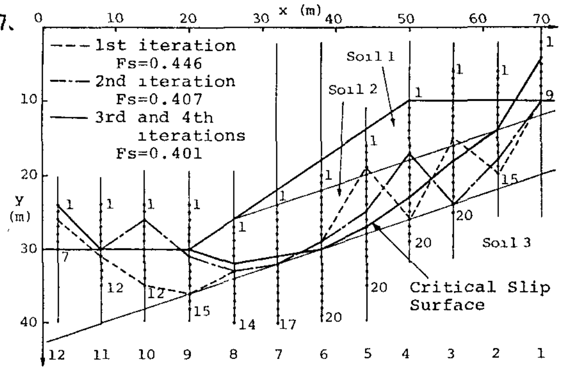

図5　解析例2（Dijkstra法を用いた場合）

表1　材料定数

材料定数		C (tf/m²)	φ (degree)	γ (tf/m³)
解析例1		1.0	10.0	1.80
解析例2	Soil 1	3.0	12.0	1.92
	Soil 2	1.0	5.0	1.92
	Soil 3	30.0	40.0	1.92

表2　CPU time

CPU time	Dijkstra法	D P
解析例1	11.59 sec	9.49 sec
解析例2	15.39 sec	12.35 sec

［参考文献］1)山上、植田：Janbu法に基づく臨界すべり面の決定について、第40回土木学会、pp.135-136、昭和60年. 2)Iu.P.Petrov：Variational Methods in Optimum Control Theory、Academic Press、pp.144-147、1968. 3) Yen,J.Y.: An Algorithm for Finding Shortest Routes from All Source Nodes to a Given Destination in General Networks、Quarterly Applied Mathematics, Vol.27、No.4、1970.