スケジュール・ネットワーク全体に分散する余裕の包括的評価に関する研究

藤　竜成 [a)]・高橋 尚己 [b)]・高塚 佳代子 [c)]・山場　久昭 [d)]・油田　健太郎 [e)]・岡崎　直宣 [f)]

Study on Comprehensive Evaluation of Margin to Be Dispersed Through Schedule Network

Ryusei FUJI, Naoki TAKAHASHI, Kayoko TAKATUKA, Hisaaki YAMABA, Kentaro ABURADA, Naonobu OKAZAKI

Abstract

We conducted research on the evaluation of the network schedule of recent projects in which critical paths are changing more frequently due to irregular work delay. We developed a method for expressing the size of the entire float dispersed throughout the schedule network as a region on the vector space (Float solution space). We also proposed a calculation method to measure its size. However, this calculation method has a problem of the calculation amount. Therefore, in this study, we proposed a method to deal with this problem of the computational complexity. Specifically, instead of calculating the absolute amount of the float solution space described above, we proposed an index that holds the magnitude relation of the absolute amount. However, the accuracy of this indicator is still only evaluated easily. In this research, in order to clarify the accuracy of the "single unit volume index" of its most promising candidate, the followings were done. First of all, we implemented a system for index calculation. Next, as an application example, we conducted an accuracy evaluation experiment using a schedule network of 20 patterns from 3 to 15 dimensions. As a result, good results were obtained from the experimental results based on this application example. In other words, if we restrict to the magnitude relation, it was found that there is no problem even if the size of the region on the vector space is evaluated by the single volume index. However, due to the nature of this indicator, there is a possibility that the deviation from the absolute evaluation becomes larger as the dimension becomes higher. Therefore, further examination is necessary in the future.

Keywords: schedule network, critical paths, size of the entire float, float solution space, single unit volume index,

1. はじめに

良いプロジェクト計画立案のためには、ネットワーク・スケジュールの評価と改善ができなくてはならない。従来の管理手法では、評価の尺度として、「全体工期の長さ」、つまり、クリティカルパスの長さという一指標しかなかった。だが、最近のプロジェクトでは、作業遅延が不規則に発生し、クリティカルパスの移り変わりが多いため、従来どおりの「全体工期の長さ」のみを指標とする手法では不十分であり、熟練者の経験や勘に依存した評価・改善に陥りやすいといった問題がある [1)]。このような問題のある近年のスケジュール管理では、従来のようにクリティカルパスの長さのみに着目するのではなく、それ以外のパス (非クリティカルパス) の持つ「フロート」に着目する必要がある。フロートとは、ある Activity("作業項目"と同義) の所要時間をプロジェクトの全体所要時間に影響を及ぼさない範囲で遅らせることができるスケジュールの余裕日数のことを意味する。(正確には"トータル・フロート"と言うが、本稿ではトータル・フロートのことを単に"フロート"と記す。)　しかしフロートの大きさは全体工期に直接影響を及ぼさない。このため、従来の納期管理では直接管理すべき対象ではなかった。しかし、クリティカルパスが移り変わる状況下では、フロートを管理する必要がある。何故なら、たとえある時点においてプロジェクト全体期間に影響しない非クリティカルパス上の Activity の余裕 (つまりフロート) であっても、ひとたび余裕日数を消費してしまうと、それがクリティカルパス上の Activity に変わった途端にプロジェクトの早期完了の可能性を下げてしまうためである。しかし、個々の Activity のフロートが全体工期に及ぼす影響の大きさを測る方法は存在しない。このため、プロジェクト管理にフロートを活用できるかどうかは、現場のプロジェクトマネジャのセンスと経験と勘にかかっているというのが現状である。したがって、プロジェクト管理の現場からは、プロジェクト全体工期の管理にフロートを直接かつ合理的に活用できる方法論の開発が求められている。以上を踏まえ、本研究室では、フロートを合理的に活用できる方法論の開発を目指している [2)3)4)]。この開発のためには先ず、フロートが全体工期に及ぼす影響

[a)] 工学専攻機械・情報系コース大学院生
[b)] 情報システム工学科学部生
[c)] 教育研究支援技術センター技術専門職員
[d)] 情報システム工学科助教
[e)] 情報システム工学科准教授
[f)] 情報システム工学科教授

の大きさを測れる必要がある。そこで、本研究室の先行研究では、スケジュール・ネットワーク全体に分散するフロート全体の大きさをベクトル空間上の領域 (“フロートの解空間” 2.1 参照) として表現するための方法が開発されている。また、その大きさを測るための計算方法が提案されている。しかし、この計算方法には計算量の問題がある。そこで、本研究室では、この計算量の問題に対処するための一方法が提案されている。具体的には、上述のフロート解空間の大きさそのもの (絶対量) を算出する代わりに、絶対量の大小関係は保持する指標の提案がなされている。しかしこの指標の精度はまだ簡単にしか評価されていない。そこで本研究では、上記指標の精度を実験的に明らかにすることを目的とし、具体的には以下のことを行った。

1. 指標計算のためのシステムの実装
2. 様々な適用例を用いた精度評価実験

2 は具体的には、適用例に関するフロート解空間の大きさの算出と指標計算の双方を行い、双方の結果を比較し、提案指標がフロート解空間の実体積の大小関係を保持するものになっているかどうかを実験的に確認した。

2. 先行研究

2.1 フロートのベクトル空間表現

プロジェクトがスケジュール・ネットワークで表されるとき、個々の Activity の時間的余裕をパラメータ表示し (“余裕パラメータ”と呼ぶ)、納期制約問題をパラメータベクトルの存在可能領域 (“フロートの解空間”と呼ぶ) を求める問題に帰着させる方法が考案された。フロートの解空間は、グラフ形式のスケジュール・ネットワークをベクトル空間表現に変換することで求められる。具体的には、個々の Activity の所要時間と余裕 (フロート)、Activity 間の先行関係、及び納期制約に基づき、機械的な手順で求められ、余裕パラメータから成る係数 1 の不等式制約の集まりとして得られる。各不等式制約はクリティカルパスに成り得る経路の経路長制約に対応付けられ、各式の余裕パラメータの並び順は各経路の作業順序を表す。

2.2 フロートの解空間の例

図 1-(1) のように、Activity A, B, C があり、期間がそれぞれ 20 日、15 日、10 日とする。A と B、A と C は順序関係があり、B と C は並行関係にある。() 内は Activity の標準的所要時間を表す。x1 と x2 で Activity A の開始時刻と終了時刻、x3 と x5 で B、x4 と x5 で C の開始時刻と終了時刻を各々表し、小文字アルファベットで同大文字の Activity の余裕パラメータを表すと、グラフ形式のスケジュール・ネットワーク (Activity Network と同義) が所定の計算過程を経て (図 1-(2))、ベクトル空間表現 (図 1-(3)) に変換され、フロートの解空間が得られる。

2.3 フロートの解空間の意味と大きさ

フロートの解空間上の各点は、「プロジェクトの全体期間を変えない Activity レベルでの遅延パターン」に相当する。例えば、図 1-(3)-ii) の凸多面体は、同図 i) の図的表現である

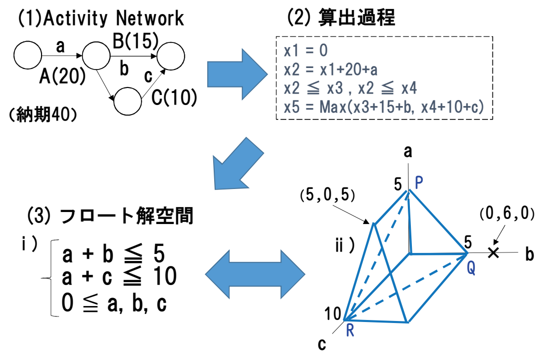

図 1. フロートの解空間生成の流れ

が、その境界及び内部の点は全て「プロジェクトの全体期間を変えない Activity レベルでの遅延パターン」に対応付けられる。例えば、境界上の点 (5,0,5) は「プロジェクトの全体期間を変えない Activity レベルでの遅延パターン」に対応付けられる。実際、Activity A と C が共に 5 日遅延してもプロジェクト全体期間は納期 40 日を逸脱しない。一方、本解空間からはみ出ている点 (0,6,0) は「プロジェクトの全体期間を変えない Activity レベルでの遅延パターン」に対応付けられない。実際、Activity B が 6 日遅延すると、たとえ A,C の遅延が 0 であったとしても、全体期間は納期 40 日を超過する。また、フロートの解空間の大きさは「全体期間を変えない遅延パターンの数」で概ね評価できる。このとき、この遅延パターンの数が、スケジュール・ネットワーク全体に分散する余裕 (フロート) の包括的な評価値となる。つまりフロート解空間の大きさが “プロジェクト全体余裕の大きさ”ということである。実際、この値が大きいスケジュール・ネットワークを持つプロジェクトほど、全体工期を遅延させるリスク (遅延リスク) は抑えられ、全体工期の遅延を回避させるためのスケジュール修正はしやすくなる。

2.4 フロートの解空間の大きさの算出

フロートの解空間は、係数が＋ 1 の連立一次不等式で表されることから第 1 象限上の原点周りの N 次元凸多面体 (“Nd-CP” と記す) を成す。従って、スケジュール・ネットワークの全体余裕の大きさ評価のためには Nd-CP (1≤ N) の体積を求めればよい。しかし、N 次元凸多面体の大きさの体積計算は、高次元では一般化できない。そこで、体積計算のかわりに、N-次元空間内の格子を凸多面体に被せ (図 2-(1))、格子点の数で評価する方法を導入した。しかし、これには計算量の問題があり、格子点数え上げに掛かる計算量は次元の指数オーダとなる。実際、15 次元のフロート解空間 (図 6 参照) の分析において、15d-CP の格子点を格子刻み幅 1 日で数え上げた場合、最大で 16 時間程度かかった。現在、Activity 数が 1000 程度の実務レベルのプロジェクトへの適用を目指しているため、この方法は現実的ではない。そこで、高次元でも計算量が少なくて済む「指標」を導入した。この指標としての要件は、大小関係の評価に限定すれば、ある程度の精度が見込めるものであることとし、その指標候補として Lp ノルムや原点-重心間の距離など試したが、そのなかで、群を抜いてよい結果を示したものが単体の超体積であった。具体的には、フロート解空間に内接する最大の単体である。なお、n 次元単体

(n-Simplex) とは、(n+1) 個のアフィン独立な点によって作られる図形のことで、0 次元単体は一点、1 次元単体は線分、2 次元単体は三角形、3 次元単体は四面体、n 次元単体は n+1 胞体・・のようになる (図 2-(2))[5)]。例えば図 1 のフロート解空間 (図 2-(1)) の単体は (図 2-(3)) となる。このような単体は、高次元でも体積計算の一般化が可能で、計算量の問題もない。しかし、その精度については確かめる必要がある。

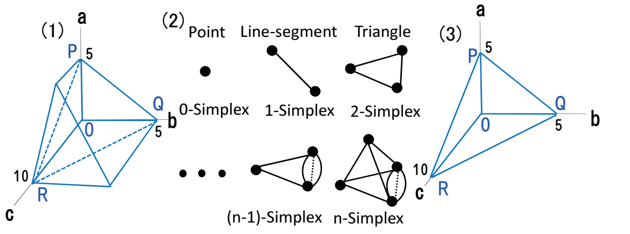

図 2. フロート解空間の大きさ評価

3.　指標計算のためのシステム

N 次元単体の超体積 NdSV は次の式で求められる。

$$NdSV = \frac{\sqrt{det[L^T L]}}{N!}$$

(L：N 次元単体のある頂点から出る互いに線形独立な n 本の有向辺の方向ベクトル $(l1, \cdots, ln)$ を各列成分に持つ m × n 行列)

ただし、ここでの指標計算の対象となる N 次元単体は原点を含む第 1 象限上のものに限られる。そのため、

$$NdSV = \prod_{\lambda \in \Lambda} s_\lambda$$

$(s_\lambda : s_\lambda$座標軸の切片、$\lambda = 1, \cdots, N)$

のように、N 個の切片の積で求められる。本研究ではこの数式に基づき指標計算システムを実装した。

4.　精度評価実験

実験方法は次のとおりである。まず格子幅 1 の格子点数で各作業項目の 1 日遅延当たりの「スケジュールの全体余裕」(つまり「フロートの解空間」) の減少率を求め、減少率の降順に作業項目を並べる。同様に、単体体積指標で評価した結果も同様に並べる。そして、これらの結果の順序関係を比較し、順序関係の一致率が高いなら、単体体積評価は概ね信頼できると判断する。以下、5 個の適用例に関する実験結果を示す。

4.1　ケース 1

図 3 はネットワーク図は Activity 数 10、次元数 5 のスケジュール・ネットワークであり、表記方法は基本的に図 1 と同じである。赤いパスは初期段階での最長作業経路 (クリティカルパス) を表す。「パラメータ統合」では、フロートが一定となる余裕パラメータ同士を統合して 1 パラメータに置き換えた結果を示している。分岐・合流点間の一経路上の Activity はフロートが一定となる性質があるためである。実験結果で、パラメータにアンダーバーがついた変数の意味は次のとおりである。例えば d1 は、d1(a1,a5,a10) より、Activity A1,A5,A10 の 1 日遅延当たりのフロート解空間縮小率を表している。また、d1 が赤字になっているのは、Activity A1,A5,A10 がクリティカルパス上の Activity であることを表している。本ケースの実験結果は、格子点数と単体指標による評価は一致率 100% となり、「完全に一致」となった。

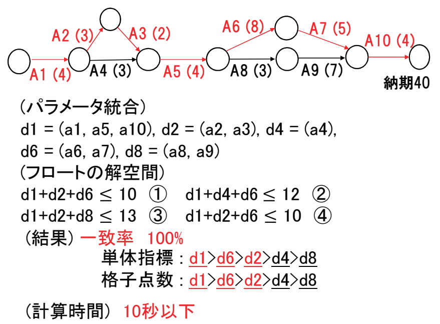

図 3. Activity 数 10、次元数 5 のケース

4.2　ケース 2

図 4 のネットワーク図は Activity 数 13、次元数 8 のスケジュール・ネットワークであり、他の表記方法や記号の意味は 4.1 と同じである。本ケースの実験結果は、格子点数と単体指標による評価は一致率 100%となり、「完全に一致」となった。

4.3　ケース 3

図 5 のネットワーク図は Activity 数 15、次元数 10 のスケジュール・ネットワークであり、他の表記方法や記号の意味は 4.1 と同じである。本ケースの実験結果は、格子点数と単体指標による評価は一致率 100%となり、「完全に一致」となった。

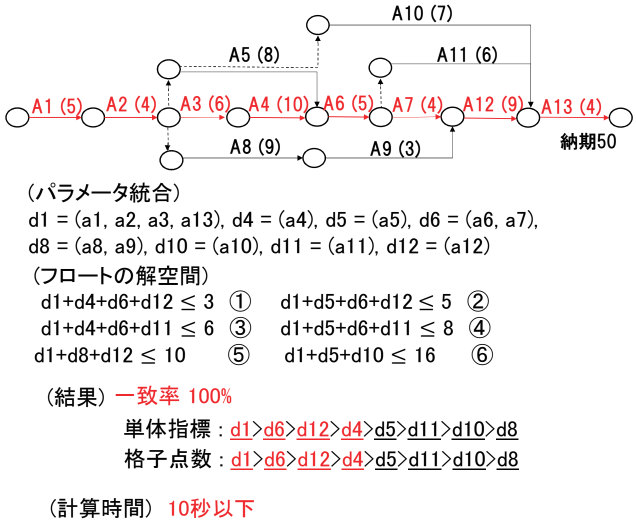

図 4. Activity 数 13、次元数 8 のケース

4.4　ケース 4

図 6 のネットワーク図は Activity 数 15、次元数 15 のスケジュール・ネットワークであり、他の表記方法や記号の意味は 4.1 と同じである。本ケースの実験結果は、格子点数と単体指

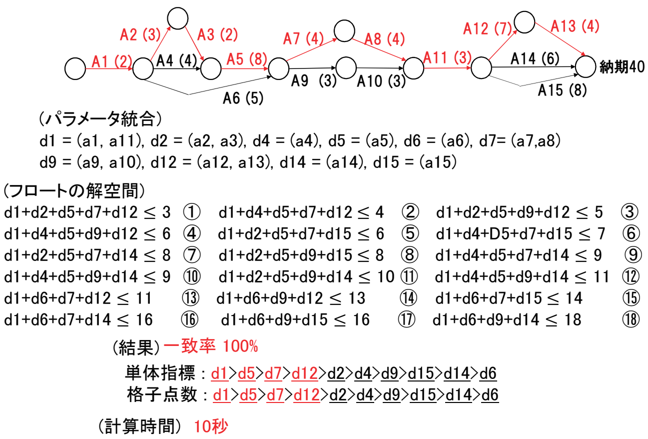

図 5. Activity 数 15、次元数 10 のケース

標による評価は一致率約 96.2%となり、「概ね一致」となった。

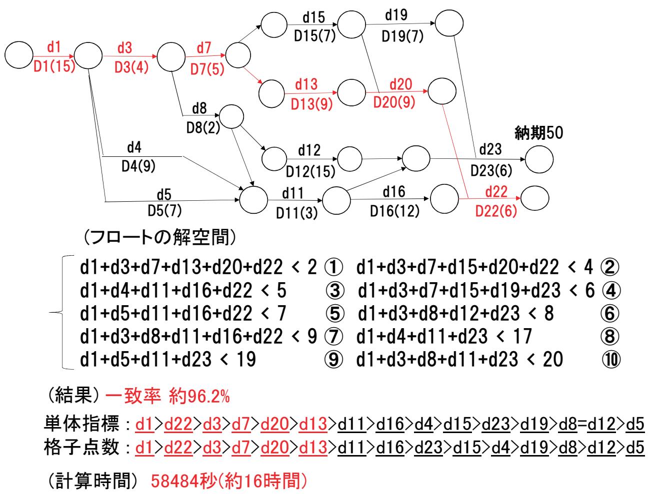

図 6. Activity 数 15、次元数 15 のケース

4.5　ケース 5

図 7 のネットワーク図は Activity 数 15、次元数 15 のスケジュール・ネットワークであり、他の表記方法や記号の意味は 4.1 と同じである。本ケースの実験結果は、格子点数と単体指標による評価は一致率約 95.6%となった。具体的には、順序関係が一致しなかった組は (d11=d16,d20),(d23,d13), (d23,d4),(d15,d4) の 4 組だった。そこで次に、格子幅を 1 から 0.5 へ細かくし、再度評価したところ、不一致の順序関係は (d11=d16,d20) の一組のみとなり、一致率は向上し約 98.9%となった。この結果から言えることは、本ケースの場合、1 日刻みの格子点評価より単体指標による評価の方がより正しかった可能性である。冒頭で、格子点評価は「概ね正しいことが保証されている」と書いたが、尖った領域の評価は荒くなる傾向にある。つまり形状依存で評価がばらつくということである。なお、格子点数に基づく体積評価が正しく行われるためには格子幅は次元 N に対し少なくとも $\frac{1}{N-1}$ の細かさが必要と予想されている [6]。しかし、格子幅を細かくする程計算時間の問題は大きくなる。

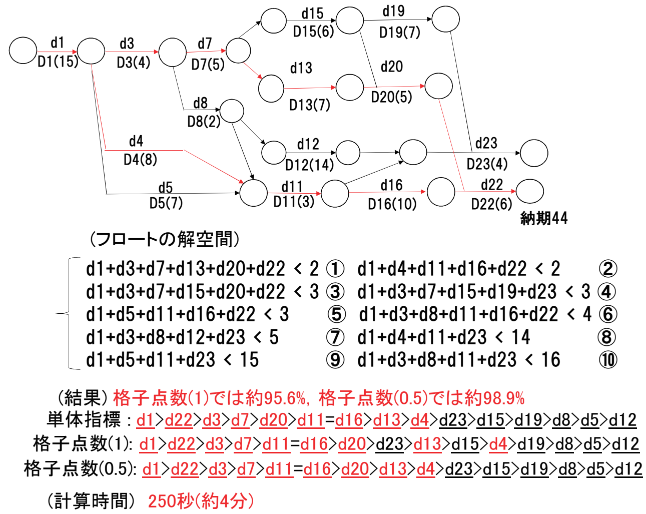

図 7. Activity 数 15、次元数 15 のケース (2)

4.6　実験の考察

今回のケース 1〜4 の実験結果からは、15 次元程度までのスケジュール・ネットワークであれば、格子点数の代わりに単体体積を指標として利用できる可能性は十分あるという結果が得られた。以下、他のケースと違って、格子点数と単体体積による評価との間に若干の食い違いがあったケース 5 について考察する。ケース 5 が他のケースと異なる点は、パス間の長さの差が極めて小さく、複数パスが同時にクリティカルパスになることすらあるという点である。このため、検出すべき差異、すなわち全体余裕 (フロート解空間) に与える影響大きさの差異 (領域体積の差異) も必然的に小さくなってしまう。このような領域体積のごく小さな差異を検出するには、実際には格子幅をもっと十分に狭くする必要がある。厳密には次元 N に対し少なくとも $\frac{1}{N-1}$ の細かさが必要と予想されている [6]。しかし、格子幅を細かくするほど計算時間の問題は大きくなる。以上のことから、大規模プロジェクトの管理のみならず、ケース 5 のように規模はそれほどでなくともクリティカルパスが頻繁に入れ替わるスケジュール・ネットワークを持つプロジェクト管理のためにも、計算量の問題への対策は極めて重要ということがわかった。

5.　まとめ

作業遅延が不規則に発生し、クリティカルパスの移り変わりが多い最近のプロジェクトのネットワーク・スケジュールの評価に関わる研究を行った。具体的には、スケジュール・ネットワーク全体に分散するフロート全体の大きさとして表される「スケジュールの全体余裕」を、ベクトル空間上の領域「フロートの解空間」の大きさとして、格子点数を数え上げるやり方で算出する際の計算量の問題を解決する課題に取り組んだ。本問題の解決策として挙がっていたのは、フロート解空間の大きさを大小関係に限定すると評価できるような指標に指標化することだったが、本研究ではその最有力候補の「単体体積指標」の精度を明らかにするため、次のことを行った。まず、指標計算のためのシステムの実装を行った。次に、適用例

として、今回は、3〜15 次元の都合 20 パターンのスケジュール・ネットワークを用いた精度評価実験を行った。その結果、今回の適用例に基づく実験結果からは、よい結果が得られた。即ち、大小関係に限定すれば、フロート解空間の大きさを、単体体積指標で評価しても概ね問題ないという結果が得られた。しかし、今回は高々 15 次元のスケジュール・ネットワークに対してしか精度評価していない。比較対象の格子点数による評価がそれ以上の次元では困難というのが理由である。本指標の性質上、高次元ほど絶対評価とのずれが大きくなる可能性はある。そのため今後更なる検証は必要である。

参考文献

1) 佐藤 知一：プロジェクトにおけるスケジュールと費用のトレードオフを考える, スケジューリング学会シンポジウム講演会論文集, pp.135-137, 2015.

2) 韓 鳳天：水力発電設備の定期修理管理プロジェクトへの適用による工程管理システムの評価, 宮崎大学修士学位論文, 2016.

3) 富田 旋：時間とコストの一元管理のためのプロジェクト管理システムの大規模プロジェクトへの適用に関する検討, 宮崎大学学士学位論文, 2017.

4) 門川 真樹：プロジェクトのスケジュール・ネットワークの遅延リスク耐性の評価について, 宮崎大学学士学位論文, 2018.

5) http://www7.atwiki.jp/neetubot/pub/neetbot-1.0.pdf, (2018/01 閲覧).

6) J. E. Reeve : On the volume of lattice polyhedra, Proceedings of the London Mathematical Society(3d series), vol.7, pp.378-395, 1957.