

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

富田 旋^{a)}・高塚 佳代子^{b)}・山場 久昭^{c)}・油田 健太郎^{d)}・岡崎 直宣^{e)}

Consideration of Application of Unified Management System of Time and Cost to Large Scale Project

Meguru TOMITA, Kayoko TAKATUKA, Hisaaki YAMABA, Kentaro ABURADA, Naonobu OKAZAKI

Abstract

In project management of large-scale, complex and diversified recent large-scale systems, the problem of the time change due to various factors is remarkable, and the aspect of the target project has been transformed into a direction with large uncertainty. Meanwhile, the current project progress management method is based on the idea at the beginning of development which required less consideration of uncertainty, that is, the idea that guides the longest work path (critical path) of the project. Thereafter, studies on functional expansion such as critical chain and project buffer are being conducted, but when applying it in practice, it is currently left to the experience rule of object dependence and experience and intuition of expert. Therefore, this laboratory has been conducting research aiming at newly developing a rational decision support method on project management which does not require experienced persons' intuition and intuition. Specifically, first, we have newly constructed a model and method for process control of projects using vector space representation with higher analytical power and computational power as compared with the conventional project management method PERT. Furthermore, we devised a method to visualize the interrelationship between projects' tasks which is important in executing rescheduling of projects but only obtained by experienced knowledge of experts and the dynamics of their relationships, developed as process control system. It is under implementation.

Keywords: project management, critical path, PERT, float, DRAG

1. はじめに

大規模・複雑・多様化した近年の大規模システムのプロジェクト管理においては、様々な要因による所要時間変動の問題が顕著であり、対象プロジェクトの様態は大きな不確定変動を持つ方向に変貌している。一方、現状のプロジェクト進捗管理手法は、不確定性を考慮する必要性の低かった開発当初の考え方、即ちプロジェクトの最長作業経路(クリティカルパス)を指針とする考え方が基本となっている¹⁾。その後、クリティカルチェーンやプロジェクトバッファといった機能拡張に関する研究が行われているが、実際に応用するときには、対象依存の経験則や熟練者の経験や勘にゆだねられているのが現状である²⁾。したがって、近年の現実の問題に対応するためには、既存の枠組みに囚われない自由な発想でプロジェクト管理問題を再構築する必要がある。

そこで、本研究室では、熟練者の経験や勘を必要としないプロジェクト管理上の合理的な意思決定支援法を新たに開発することを目的とし研究を行ってきた。具体的には、まず、従

来のプロジェクト管理手法 PERT と比べ、より高い分析力と計算力を持つベクトル空間表現を用いたプロジェクトの工程管理のためのモデル及び手法を新たに構築した³⁾(2.2節)。更に、プロジェクトの再スケジューリングを実行する上で重要だが熟練者の経験知によってしか得られなかったプロジェクトの作業間の相互関係やその関係性のダイナミクスを可視化する方法を考案し、工程管理システムとして開発・実装中である(2.3節)。

ところで、大規模かつ所要時間変動の問題が顕著な近年のプロジェクトの再スケジューリングの実施に大きく貢献する可能性のある新たな管理指標“DRA”がプロジェクト管理の実務家の間で注目されはじめている⁴⁾⁵⁾。3.1節で後述する DRAG の概念や定義は必ずしも分かり易いものではない。しかし、DRAG で出来る事やその使い方は非常にシンプルで分かり易い。具体的には、例えば、DRAG を使うと、プロジェクトの時間短縮をしたい時、短縮可能な作業と可能な短縮幅をピンポイントに知ることができる。更に上手く使うと、コスト効率の良い短縮方法を特定できる可能性がある。従来手法に、各作業の着手上の余裕時間を表す管理指標 Total Float(以降“Float”と記す)があり、この Float を技巧的に繰り返し用いることで DRAG の機能を実現できないわけではない。しかし計算量の問題が著しく現実的ではない。DRAG は上述のような管理上知るべき情報を最低限の計算量で知ることができ、近年大規模化するプロジェクトの管理には欠かせない強力な指

^{a)}工学専攻機械・情報系コース大学院生

^{b)}教育研究支援技術センター技術専門職員

^{c)}情報システム工学科助教

^{d)}情報システム工学科准教授

^{e)}情報システム工学科教授

標と言える。

しかし、DRAGの算出ルールを完全に一般化し自動化することは従来のPERT法の枠組みでは出来なかった。このため、一部の実務家が経験と直感でDRAGの値を見積もって試しに使ってみるに留まっていた。これに対し、本手法を用いると、このDRAG算出の手順化が容易に行えるため(3.2節)、これまで出来なかったDRAGの実践的活用に向けた取り組みが可能と考えられる。

そこで、本研究室では現在、提案手法に基づき開発中の工程管理システムの大規模プロジェクトへの対応力の強化、及び機能拡充を目的とし、本システムにDRAG算出機能を追加実装し、DRAGの実践的活用方法について検討を進めている。この研究の一環として、本論文では以下のことを行った。

- (1) 工程管理モデル生成プロセスの自動化
- (2) 本研究室で開発されたDRAG算出手順の妥当性検証
- (3) コスト効率を考慮したプロジェクト時間短縮方法の特定にDRAG使用が有用であることを実証する実験
- (4) 拡張・拡充された工程管理システムの大規模プロジェクト管理への適応可能性検証と問題点の抽出

2. 工程管理システムの開発

2.1 諸概念

本論文では、納期とは、プロジェクト開始から完遂日までの期間とし、契約された納期(契約納期)と区別する。クリティカルパスとはプロジェクトの納期に直接影響を及ぼす最長作業経路、安全余裕とは納期契約違反を防ぐ為にプロジェクトに予め与えられる時間的余裕を表す。

2.2 工程管理モデル

本研究室の先の研究³⁾で、プロジェクトの作業全体の遅延をパラメータベクトルで表し、納期制約問題をベクトルの存在可能領域(“遅延許容領域”と呼ぶ)を求める問題に帰着させる方法が考案され、この遅延許容領域が工程管理上有用なモデルになり得ると分った。これを工程管理モデルと呼ぶ。

遅延許容領域は、各作業の所要時間と遅延を考慮しながら作業間の先行関係を不等式制約で表し、その制約全体を納期制約と併せて簡略化して得られる不等式制約の集まりである。また、遅延許容領域を成す制約式各々は、プロジェクトの最長作業経路となり得る経路長各々に対する制約に相当し、タスク分析能力としては従来手法のPERTと同等の能力を持つ。更に、実数ベクトルで処理できる利点として高い分析力と計算力を特長として持つ。この特長により、グラフ理論ベースのPERTより深い分析ができる可能性を持つ。例えば、冒頭で述べたDRAGの計算可能性はその一例である。また、各作業の遅延そのものに着目したモデルであるため、余裕時間に関する分析は従来法よりも直接的かつ簡潔に行える。具体的には、各作業の着手上の余裕時間を表す指標Floatは、遅延許容領域を制約条件とし、着目作業のFloatを目的関数とし、これを最大化する制約充足問題として扱える。

2.3 工程管理システム

本研究室で開発中の工程管理システムの特徴は、従来は重点管理項目(クリティカルパスの所在や安全余裕残量)を見出す計算過程で使用していた指標Floatを、これら重点管理項目の可視化に直接利用するという点である。

2.3.1 システム準備段階

工程管理モデル、即ち遅延許容領域を生成する。具体的には、作業の前後関係と各作業の所要時間及び作業遅延を表す遅延パラメータに基づき、最長と成り得る作業経路の全ての経路長を算出するが、これらは遅延パラメータから成る線形多項式となる。これらに納期制約を与えて得られる線形不等式の集まりが遅延許容領域となる。

2.3.2 システム運用段階

目標スケジュールが示す作業間の順序関係に従ってプロジェクトを実行し、実行中変化していくFloatとDRAGを遅延許容領域に基づき次々と算出し、実行中の重点管理項目の変化していく様子を、指標Floatの並びから直感的に認識できるように可視化する(可視化のイメージは4.3節の図4参照のこと)。なお、具体的な可視化の方法は以下の通りである。

- クリティカルパスは、Floatの最小値(或いは0でないDRAG)を持つ作業列をハイライト表示させることで所在を可視化できる。
- 安全余裕残量は、各時点でのFloatの最小値をハイライト表示させることでその変遷を可視化できる。なお、Floatの最小値が負となるタイミングが契約納期逸脱のタイミングとなる
- 納期短縮可能な作業項目は0でないDRAGを持つ作業項目、可能な短縮幅はDRAGの大きさとして検出できる。

2.4 工程管理モデル(遅延許容領域)生成手順の自動化

実装したシステムでは、作業間の前後関係と各作業の所要時間及び作業遅延を表す遅延パラメータを与えると、以下の式で表される各作業の開始時刻と終了時刻を順に求める計算を作業間の前後関係に従って順に繰り返し行わせる。

作業pの開始時刻 $\geq \text{Max}[p_j \text{の終了時刻} \mid p_j \text{は} p \text{の先行作業全て}]$

作業pの終了時刻 = pの開始時刻+所要時間+遅延パラメータ
最終作業の終了時刻(複数通り)が得られると、即ちそれらが最長と成り得る作業経路の経路長に各々相当する式となっている。

3. DRAG算出手順について

3.1 DRAGの概念定義

DRAGとは、プロジェクトを構成する作業項目ごとに測ることができ、その作業項目がどれだけ納期に影響を及ぼしているか(納期を押ししているか)を直接表す指標である。最もオーソドックスな使い方は、納期を出来る限り早めたい場合はDRAGが最も大きな作業に着目するという点である。ま

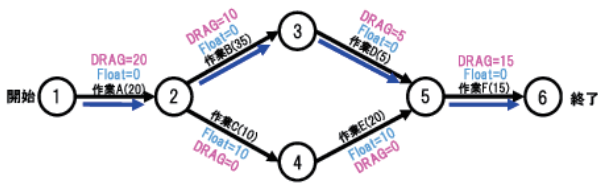


図 1: ネットワーク・ダイアグラムの例

た、DRAG の値に納期遅れに対して課せられる違約金の額を掛けたものを DRAG Cost と呼び、これを使うと作業時間短縮に掛かるコストと納期遅れによる損失のトレードオフの問題を簡潔に扱える。

3.2 遅延許容領域に基づく DRAG の算出手順

図 1 は、6 つの作業を持つ簡単なプロジェクトの Activity on Arc 型のネットワーク・ダイアグラムの例である。() 内は各作業の標準的所要時間である。同じアルファベットの大字と小文字は、ある作業の名称と、同作業の遅延パラメータを表す。本プロジェクトの納期を 50 とすると、本例の遅延許容領域は以下の不等式制約の集まりとして表せる。各作業の Float と DRAG は図 1 の通りである。着手上の余裕時間 (Float) が 0 ではない作業 (C,E)、つまり非クリティカルパス上の作業の DRAG は 0 になる。一方、Float を持たない作業 (A,B,D,F)、つまりクリティカルパス上の作業は、いずれも 1 以上の DRAG を持つ。ここで、もし DRAG=20 の作業 A をこれより小さい 18 だけ短縮すると、プロジェクトの工期もそのまま 18 短縮される。一方、DRAG = 10 の作業 B をこの DRAG を越す大きさの 18 短縮しても、プロジェクトの工期は 10 までしか短縮されず、残り 8 の短縮分は短縮コストの無駄となる。なお、遅延許容領域を用いると、DRAG は以下のルールで求められる。

- (R1) クリティカルパス上にない作業要素の DRAG は全て 0
- (R2) クリティカルパス上にあり、かつ他に平行作業がない作業は、それ自体の所要期間が DRAG となる
- (R3) クリティカルパス上にあり、平行作業のある作業は、クリティカルパスの経路長制約の定数項 (X) と、クリティカルパスの次に制約の厳しいパスの経路長制約の定数項 (Y) との差 $Y - X$ が DRAG 値となる。
- (R4) ただし、その作業の所要時間が $Y - X$ より小さい時は、それ自体の所要期間が DRAG となる。

なお、ある作業に平行作業があるか否かは、各制約条件式のアルファベットの並びの比較で機械的に判断できる。

4. 検証実験

近年実際に実施され成功裏に終了したプロジェクトの工程表 (ガントチャート) の実データ一式を使用し、以下の 3 種類 (4.1~4.3) の検証実験を行った。

4.1 DRAG 算出手順の妥当性検証

比較的小規模の住宅建築プロジェクト (作業項目数 30 程度) を適用例とし、開発された DRAG 算出手順の妥当性を確認した。

表 1 DRAG を持つ作業の短縮が及ぼす影響の解析

作業		P6	P7	P25	P40	P29	P13	P23	P33	P16	...
遅延パラメータ		d6	d37	d25	d40	d29	d13	d23	d33	d16	...
① Float		-1	-1	-1	-1	9	-1	-1	44	0	
DRAG		3	5	10	10	0	8	3	0	0	
② Float	P40を2短縮	1	1	1	1	9	1	1	44	8	
③ Float	# 5短縮	4	4	4	4	9	4	4	44	11	
④ Float	# 8短縮	7	7	7	7	9	7	7	44	14	
⑤ Float	# 10短縮	9	9	9	9	9	9	9	44	16	
⑥ Float	# 11短縮	9	9	9	10	9	9	9	44	17	
⑦ Float	# 12短縮	9	9	9	11	9	9	9	44	17	

実験の流れと結果

目標スケジュールを表すガントチャートから抽出した順序制約に納期制約を加えて簡約化し、都合 12 個の制約条件式から成る遅延許容領域を得た。そして、工程表に従ってシミュレーション実行し、実行中変化していく Float を算出し、Float の最小値 (図中青字) を観測し、これが負となる納期制約逸脱のタイミングを検出し、その時の DRAG を算出した。表 1-①はその時の Float と DRAG である。更に、この納期制約逸脱を回避するための作業時間短縮箇所を作業 P40 (DRAG = 10) と仮定し、P40 短縮幅と余裕残量との関係を調べた (表 1-②~⑦)。その結果、P40 の持つ DRAG を越さない短縮幅 (②~⑤) では短縮幅分の最小 Float の増加 (つまり納期短縮) が見られ、DRAG を越す短縮幅 (⑥、⑦) ではそれ以上の短縮が見られない。このことから、本実験で求められた DRAG は正しいことが確認された。以上のような検証実験を十分な回数行った結果、考案した算出手順は妥当と判断した。

4.2 コスト効率を考慮し時間短縮方法を特定する実験

同じく住宅建築プロジェクトを適用例とし、コスト効率を考慮したプロジェクト時間短縮方法の特定に DRAG が有用であることを実験的に確認した。

4.2.1 実験の流れと結果

プロジェクトのシミュレーション実行中 (図 2 のネットワーク図参照)、契約納期を 10 日超過することが確定し、この時点で計画見直しを行うこととし、納期短縮のための作業短縮コストと契約納期違約金の和で決まる損失コストが最小となる短縮方法を見出すこととした。

表 2 は、着目時点でクリティカルパス上に在る作業各々に関する情報である。各作業に掛かる標準的なコスト及び短縮コストは、この種のプロジェクトに関する一般の見解に基づき設定した。即ち、標準的なコストに関しては総工費に対する標準的なパーセンテージ等で算出した。また、時間短縮コストに関しては、この種の作業の場合一般に、標準的所要時間に対する短縮幅の占める割合が小さいほど人員等の追加に

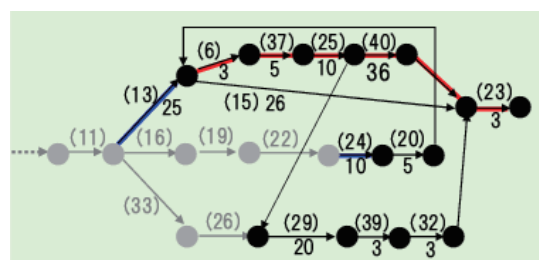


図 2: 建築プロジェクトのネットワーク図

表2 計画見直し時にクリティカルパス上に在る作業の情報

	所要時間(日)	遅延コスト(万円)	DRAG(日)	短縮にかかるコスト(万円)												
				1日	2日	3日	4日	5日	6日	7日	8日	9日	10日			
仮設工事(d6)	3	30	3	15												
給排水設備工事(d37)	5	20	5	4	9	20										
左官・タイル工事(d25)	10	100	10	22	25	30	38	53	80							
外構工事(d40)	36	200	10	20	21	23	26	30	35	42	50	60	72			
コーキング工事(d23)	3	18	3	10												

表3 短縮日数ごとのコスト最小の短縮方法と損失

(1) 一作業のみでの短縮に限定した場合

短縮日数 (計画経過日数)	0日 (10日)	1日 (9日)	2日 (8日)	3日 (7日)	4日 (6日)	5日 (5日)	6日 (4日)	7日 (3日)	8日 (2日)	9日 (1日)	10日 (0日)
コスト最小の短縮方法		d37	d37	d37	d40	d40	d40	d40	d40	d40	d40
短縮にかかるコスト... A		4	9	20	26	30	35	42	50	60	72
計画経過済資金(万円/日) ... B	60	54	48	42	36	30	24	18	12	6	0
損失... A+B	60	58	57	62	62	60	59	60	62	66	72

(2) 複数作業に跨る短縮も可とした場合

短縮日数 (計画経過日数)	0日 (10日)	1日 (9日)	2日 (8日)	3日 (7日)	4日 (6日)	5日 (5日)	6日 (4日)	7日 (3日)	8日 (2日)	9日 (1日)	10日 (0日)
コスト最小の短縮方法		d37(1)	d37(2)	d37(2)	d40(4)	d37(1)	d37(1)	d37(2)	d37(3)	d37(2)	d37(2)
短縮にかかるコスト... A		4	9	19	26	30	34	39	44	51	59
計画経過済資金(万円/日) ... B	60	54	48	42	36	30	24	18	12	6	0
損失... A+B	60	58	57	61	62	60	68	67	66	67	69

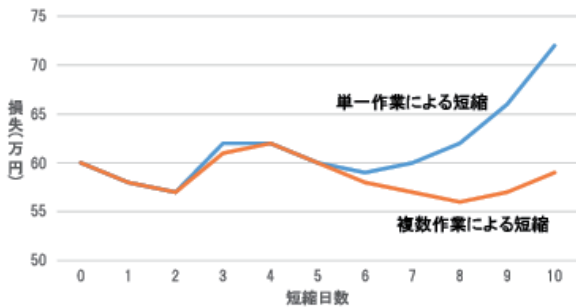


図3: 短縮日数と損失の関係

よる時間短縮の余地が大きく(短縮し易く)、逆にその割合が大きいほど短縮は困難となり、ある大きさ以上になると物理的に短縮不可能となる。表2の斜線は「短縮不可能」を意味する。

表3には、表2に基づき短縮日数ごとのコスト最小の短縮方法を求めた結果を示している。ただし、表3-(1)は一作業のみでの短縮と限定した場合の結果であり、表3-(2)は複数作業に跨っての短縮も可とした場合の結果を示している。一作業限定での短縮の場合、可能な短縮の上限は各作業のDRAGの値がそのまま上限となる。このため、コスト最小の短縮方法は、各作業のDRAGと短縮コストが分れば簡単に求められる。一方、複数作業に跨った短縮も可とする場合、短縮幅の上限は単純に当該作業のDRAGの値とはならない。並行作業系列を持つ作業の場合、その作業のDRAGのみならず並行作業列のFloatの大きさにも依存して上限が決まる。今回の実験では、この上限は論理的に考えながら求めた。

4.2.2 結果の考察

表3、図3より、短縮幅が長い時ほど、複数作業に跨った短縮まで考えた方が、損失の小さい短縮方法が得られる可能

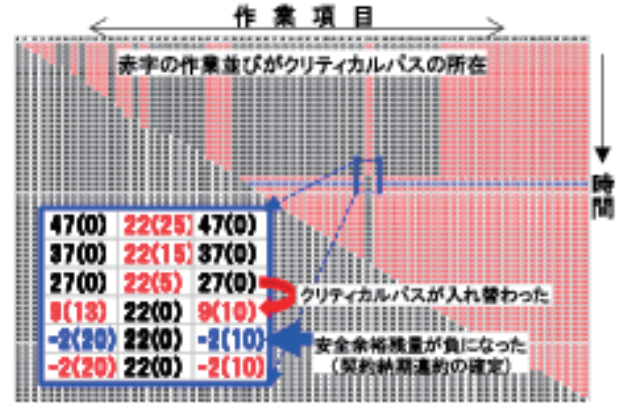


図4: 比較的大規模なプロジェクトへの本手法の適用結果

性が大きいことが分かった。従って、今回の実験では論理的思考に従って求めたが、「複数作業を組み合わせた短縮での各作業の短縮幅の上限」を見出す方法の手順化には意義があると考えられる。

4.3 大規模プロジェクトへの適応可能性検証と問題抽出

比較的大きな規模のプロジェクトとして、水力発電設備の定期修理管理(オーバーホール)の作業項目数130程度のプロジェクトを適用例とし、提案する工程管理手法が、近年の大規模プロジェクト管理に資する技術足り得るかを検証する為の実験を行った。また、実データを使用する場合、実験データとしての不完全さをどのように補うかの問題があるが、この種の問題点の抽出を行った。

実験の流れと抽出された問題点及び結果

目標スケジュールを表すガントチャートから抽出した順序制約に納期制約を加えて簡約化し、項数30~69の都合30個の制約条件式から成る遅延許容領域を得た。

ここまでの問題点として、プロジェクトの初期作業ではないが、先行作業を明確に持たない、つまり開始タイミングが明確でない作業が本例では6個程確認された。そこで、その様な作業の先行作業は全て初期作業とした。同様に、後続状態を明確に持たない作業が本例では19個確認されたが、その全ての作業の後続作業はプロジェクトの最終作業とすることとした。この方法は、本手法を適用する上で便宜上与える順序関係の数を最小限に抑え、引いては生成される遅延許容領域の制約条件式の数(つまり最長作業経路の候補数)を最小限に抑えられる方法である。

次に、工程表に従ってシミュレーション実行し、実行中変化していくFloatとDRAGを算出し、Floatに基づくクリティカルパスと安全余裕残量の変化の可視化の可能性を確認した(図4)。以上のことから、作業数150程度のプロジェクトへの適用は可能と判断した。

5. まとめ

本研究室で開発中の工程管理システムの大規模プロジェクトへの適用に向けた整備拡充を行い、作業数150程度のプロジェクトへの適用可能性を確認した。また、本システムに追加実装されたDRAG算出機能の妥当性確認した。更に、コス

ト効率を考慮したプロジェクト時間短縮方法の特定に DRAG が有用であることを実証すると共に、DRAG と Float の併用による効果を示唆した。

参考文献

- 1) クリティカルチェーン・プロジェクトマネジメント - Wikipedia, (2017/02/01 アクセス).
- 2) 藤原他:資源制約付きプロジェクト・スケジューリング問題に関する基礎的研究, 数理解析研究所講究録, pp.125-130 (2009).
- 3) 韓鳳天: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, 宮大工学修士学位論文 (2016).
- 4) Stephen A.Devaux : The Drag Efficient, Defense AT&L, January-February (2012).
- 5) 佐藤知一:スケジュールの DRAG とはどんな尺度か、タイム・コンサルタントの日誌から available from(<http://brevis.exblog.jp/20000432/>), (2017/02/01 アクセス).