

不連続なクリティカルパスの原因と その対策に関する一考察

岡本 尽¹・梶谷 浩²

¹正会員 大成建設株式会社 国際支店土木部 (〒163-0606 東京都新宿区西新宿一丁目 25-1)

E-mail: jin@ce.taisei.co.jp

²正会員 金沢大学 理工研究域 環境デザイン学系 構造工学研究室 (〒920-1192 石川県金沢市角間町)

E-mail: masuya@se.kanazawa-u.ac.jp

国際建設プロジェクトにおいて、請負者は契約書に基づき、全体工程をクリティカルパス法で管理することを求められる。クリティカルパスはプロジェクトの最終完成期日を動的に管理する有効な手法だが、様々な原因に起因してクリティカルパスが不連続となることがある。国際建設プロジェクトにおいて工期延長クレームは、請負者の責に起因しない遅延事象が全体工期に与える影響を証明した場合にのみ有効となるため、クリティカルパスが不連続となった工程表で論理的根拠のあるクレーム図書を作成できない。工期延長クレームを効果的に実施するためには、クリティカルパスの不連続についてその原因と対策を熟知する必要があるため、本稿はそれらについてまとめたものである。

Key Words: 国際建設プロジェクト, クリティカルパス, トータルフロート, *Microsoft Project*, リソース制約条件付クリティカルパス法

1. まえがき

国内建設市場での工程管理が完成期日を如何に守るかに着眼した工期管理であるのに対し、国際建設プロジェクトではプロジェクトの遂行過程で契約条件と異なった事象の発生によって影響を受ける契約完成期日を管理するため、定量的かつ論理的な工程解析技術が必要¹⁾²⁾である。一般に国際建設プロジェクトにおいて、プロジェクトの完成予定日を動的に把握・管理したり、工程解析したりする目的のため、建設契約書にはベースラインプログラム（基本工程）はクリティカルパス法（CPM: Critical Path Method）にて作成するよう契約で規定され、契約内容の変更管理をするための基準となる契約図書の一部として利用される³⁾。

CPM はネットワーク工程分析技法であり、プロジェクトの各作業の所用期間並びにそれらの順番を論理的な依存関係で連結することにより、作業毎の工程余裕度を把握したり、最短となるプロジェクト

全体工期を特定することができる⁴⁾。往路・復路分析により得られる最早工程と最遅工程との差分から各作業毎のフロートを計算し、フロートが0か負の値となる作業の一連のつながりをクリティカルパスと呼ぶ。⁴⁾ クリティカルパスはプロジェクトを遂行するための最も長い作業の経路を示すものであり、プロジェクトの最短全体工期を決定する経路となる。

プロジェクトの実績工程を反映しながらクリティカルパスを管理することで、プロジェクトの最終完成期日を動的に管理することができる。各作業における生産性の変更等を含む契約条件の変更や、クレーム・工事の変更に起因する波及効果によって発生する工期延長クレームなどは、各作業が論理的依存関係のネットワークで構築されていることで客観的かつ論理的な分析が可能⁵⁾となる。

クリティカルパスを管理するための様々なプロジェクト管理ソフトウェアがあるが、国際建設プロジェクトでは一般的に契約で *Microsoft Project* あるいは *Primavera* を利用することが規定されること

が多い。これらのソフトウェアの標準的な機能は作業の所要期間と論理的依存関係の設定にとどまらず、作業や資源の作業可能カレンダーの設定や作業の種々の制約条件などといった細かい条件の設定が可能であり、複雑かつより現実に即した工程上の制約条件の設定が可能となっている。

しかし、このような細かい便利な条件設定に起因して、クリティカルパスがプロジェクトの最初から最後まで一貫通貫とならずに、途中で途切れてしまう不連続なクリティカルパスとなることも少なくない。

工期延長のクレームは、契約と異なる条件によって発生した事象によって、クリティカルパス上にある作業が遅延し、全体工程に影響を及ぼすことを証明できた場合にのみ権利を主張できるため、遅延事象がクリティカルパス上にあることを証明することは非常に重要となる。クリティカルパスが不連続となる現象は、このクレーム立証の妨げとなるため、不連続なクリティカルパスの発生について、その原因と発生条件を熟知する必要がある。さもなければ動的に変化するクリティカルパスの分析はおろか、完成想定期日の動的な変化の分析も、工期延長に関わるクレームの論理的根拠の構築も不可能となる。

この論文では、これらクリティカルパスの不連続となる原因とその対策を整理することを目的とする。以下、工程管理ソフトとして Microsoft Project を利用して、考察を進める。

2. 制約条件に起因する不連続

Microsoft Project において、制約条件はタスクの開始日や終了日に付与する条件であり、論理的依存関係を維持しながら開始や終了の時間的条件を付与する機能である。制約条件は、“なるべく早く”及び“なるべく遅く”の“弱い制約条件”、“指定日まで開始”、“指定日まで終了”、“指定日以降に開始”、“指定日以降に終了”の“やや弱い制約条件”、“指定日に開始”、“指定日に終了”の“強い制約条件”の3種類のタイプに分別できる。通常、各タスクには“なるべく早く”の弱い制約条件が既定で割り付けられるが、状況に応じてその他の制約条件を割り付けることが可能である。しかし既定以外の制約条件を付与した場合に、その制約条件がクリティカルパスを不連続とする原因となることがあるため、なるべく制約条件を付与することは避けた方がよい。以下にそれぞれの制約条件を利用した場合の不連続の原因と対策を考察する。

(1) 弱い制約条件

弱い制約条件には“なるべく早く”と“なるべく遅く”があり、最早工程の場合の規定値が“なるべく早く”、最遅工程の場合の規定値が“なるべく遅く”となる。最早工程における“なるべく遅く”の制約条件、または最遅工程における“なるべく早く”の制約条件は、ネットワークの自由度を拘束し、複数のクリティカルパスの生成、あるいはフロートを生み、クリティカルパスの不連続の原因となる。図-1は弱い制約条件によるクリティカルパスの不連続の例である。

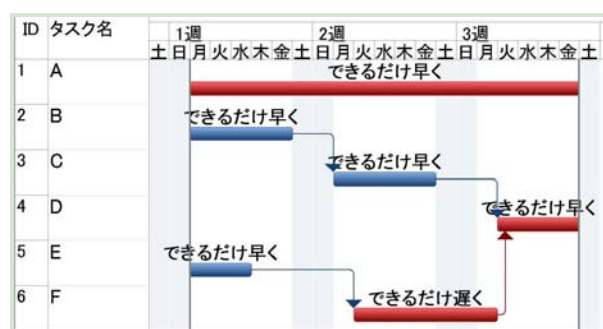


図-1 弱い制約条件に起因するクリティカルパスの不連続

図-1のように全体工期は“タスク A”の20日とする。“タスク F”には“できるだけ遅く”の弱い制約条件が割り付けられている。“後続タスク D”は“できるだけ早く”の制約条件が割り付けられているにもかかわらず、“タスク F”の制約条件に支配されフロート 0 のクリティカルタスクとなる。また、“タスク F”の“先行タスク E”は“できるだけ早く”の制約条件が割り付けられ、その終了日と“タスク F”の開始日との間に3日間のフロートが発生しているため、“タスク E”はクリティカルとはならない。

最早工程において、弱い制約条件“できるだけ遅く”が割り付けられているタスク以降につなげられたタスクはそのタスクとともにクリティカルタスクとなる。これらの一連のクリティカルタスクの先行タスクにはフロートが発生することになり、弱い制約条件が付与されたタスクを境にして、クリティカルパスが不連続となる。

弱い制約条件に起因するクリティカルパスの不連続への対策として、次の3つの方法がある。

- 弱い制約条件を付与したタスクの先行タスクとの依存関係に先行タスクのフロート日数と同日の仮のラグタイムを設定する。この手法は仮の手段であるため、その後の実績工程の記録の度にラグタイム日数を調整しなおす必要があり、推奨される手法ではない。

- 論理的に制約条件を付与しなければならないタスクかどうかを検討し、可能であれば弱い制約条件を“なるべく早く”に戻す。最早工程を求められるケースで“なるべく遅く”の制約条件を付与しなければならないケースは稀有である。どうしても制約条件を付与しなければならない場合は次のc)を適用する。
- 弱い制約条件によってクリティカルとなった一連のタスクをクリティカルパスと見做さないで工程分析を行う。これらのクリティカルは恣意的なものであり、本来の工程ネットワークにおけるクリティカルパスではないので、工期延長の対象となるクリティカルパスとは見做さないで分析するのが適当である。

(2) やや弱い制約条件

やや弱い制約条件には、“指定日以後に開始”、“指定日以降に終了”、“指定日以前に開始”及び“指定日以前に終了”がある。最早工程において、“指定日以後に・・・”の制約条件は、指定日を拘束点として動的なスケジュールの自由度を前方に拘束するため、フロートを生みクリティカルパスを不連続とする原因となる場合がある。

また、最早工程における“指定日までに・・・”の制約条件が付与された場合のフロートの計算は、後続タスクの開始日までの余裕日数期間（フリーフロート）だけではなく指定日の制約条件を守るための余裕日数との少ない方の値で計算されるため、注意が必要である。

a) 制約条件-“指定日以後に・・・”



図-2 “指定日以後に開始”のやや弱い制約条件に起因するクリティカルパスの不連続

図-2において“タスク E”は2週目月曜日を指定日とした“指定日以降に開始”のやや弱い制約条件が割付けられている。“後続タスク C”との間にはフロートがないため、“タスク E”の指定日を起点とし

たクリティカルパスとなる。“先行タスク D”には2日間のフロートがあるため、クリティカルとはならない。

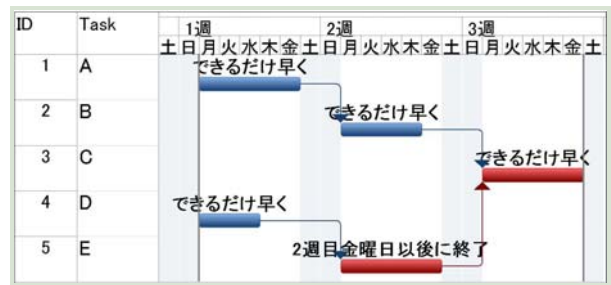


図-3 “指定日以後に終了”のやや弱い制約条件に起因するクリティカルパスの不連続

図-3において、“タスク E”は2週目の金曜日を指定日とする“指定日以降に終了”の制限条件が割り付けられており、指定日を基点として“タスク E”の所要期間から逆算して開始日が決定されている。指定日を拘束点として前方固定されてしまうため、先行タスクにフロートが発生している。“タスク A”や“タスク B”は、“後続タスク C”の開始日に対して1日のフロートがあるため、クリティカルパスとはならない。

図-2、図-3ともに“先行タスク D”がどんなに早く終了していても、“指定日以降に・・・”の制約条件が割り付けられている“タスク E”は指定日を拘束点として前方固定されてしまい、“先行タスク D”にフロートが発生するため、クリティカルパスは制約条件の指定日を起点とした制約条件を付与されたタスク以降に移行し、クリティカルパスは不連続となる。

このケースでクリティカルパスの不連続が発生した場合の対策として、次の4つの方法がある。

- 制約条件を付与したタスクの先行タスクとの依存関係に先行タスクのフロート日数と同日の仮ラグタイムを設定する。この手法は仮の手段であるため、その後の実績工程の記録の度にラグタイム日数を調整しなおす必要があり、あまり推奨される手法ではない。
- 契約上許されるのであれば、他のタスクの所要期間を見直し、やや弱い制約条件によってクリティカルとなった一連のタスクがクリティカルパスとならないよう工程を見直す。一般的に契約でベースライン工程に修正を加えることは許されないが、可能であればこの方法が一番合理的な方法である。
- 論理的に制約条件を付与しなければならないタスクかどうかを検討し、やや弱い制約条件

を“なるべく早く”に戻す。よく見られる“指定日以降に・・・”の制約条件を付与しているケースは資機材の調達の場合などだが、調達期間となる先行タスクを設定することで、制約条件を“なるべく早く”に戻すことができる。

- 上記の方法が適用できない場合には、プロジェクトの開始から始まるクリティカルパスとすることはできない。やや弱い制約条件が付与されたタスクの指定日を起因とするクリティカルパスで工程を分析することになる。

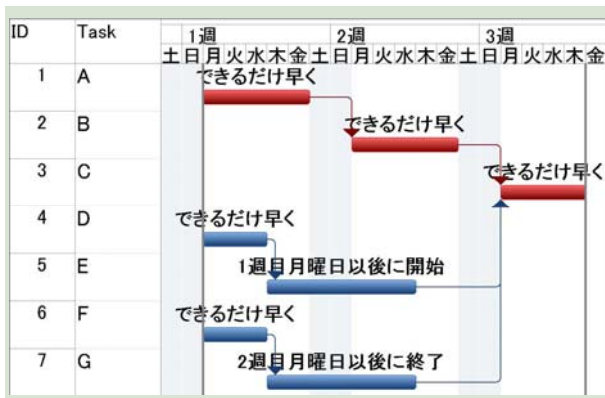


図-4 指定日条件を満たした“指定日以降に・・・”のやや弱い制約条件

図-4 では、“指定日以後に開始・終了”の制約条件を満たしているケースである。制約条件を満たしている場合には、フロートの計算は後続タスクとの時間的な余裕時間で計算され、クリティカルパスの不連続や複数のクリティカルパスの発生の原因とはならない。しかし、プロジェクトの進捗中に実績記録を動的に行った場合に“タスク A”、“タスク B”、“タスク D”、“タスク F”の開始日・終了日が早まった場合には、図-2 や図-3 のケースと同様に指定日を拘束点とした前方固定となり、クリティカルパスの不連続の問題が発生する可能性があることに留意しておかなければならない。

b) 制約条件- “指定日までに・・・”

図-5、図-6 ともに“指定日までに・・・”の弱い制約条件が割り付けられている例であり、指定日の条件を満たしているケースである。ともに“タスク E”、“タスク F”とその後続タスク“タスク C”との間に3日の余裕期間があるが、トータルフロートの値は1日と3日と異なった値を示している。



図-5 “指定日までに開始”のやや弱い制約条件

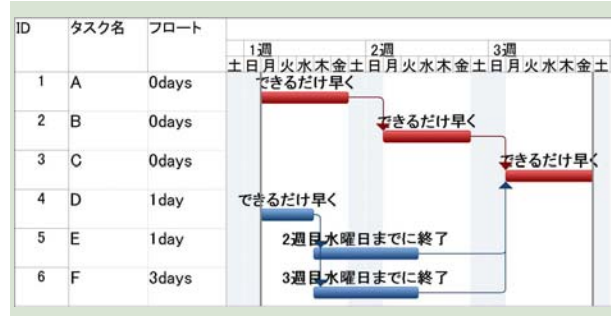


図-6 “指定日までに終了”のやや弱い制約条件

図-5 において、“タスク E”は1週目金曜日を指定日、“タスク F”は2週目金曜日を指定日とした“指定日までに開始”の制約条件が割り付けられている。“後続タスク C”の開始日までにともに3日間の期間があるが、それぞれのトータルフロートの値は異なっている。

図-6 において、“タスク E”は2週目水曜日を指定日、“タスク F”は3週目水曜日を指定日とした“指定日までに終了”の制約条件が割り付けられている。“後続タスク C”の開始日までにともに3日間の期間があるが、それぞれのトータルフロートの値は異なっている。

図-5、図-6 ともに“タスク E”を考えると、“後続タスク C”を遅延させないためには3日の期間延長が可能であるが、指定日の制約条件を守るためには1日の遅延しか許されない。従って、3日>1日で“タスク E”のフロートは1日となる。

“タスク F”も同様に、後続タスクを遅延させないためには3日の期間延長が可能であり、指定日の制約条件を守るためには6日の遅延まで許される。従って、3日<6日で“タスク F”のフロートは3日となる。

これらのタスクの“先行タスク D”はフロートの少ない“タスク E”のフロート1日をそのまま継承することになる。プロジェクト全体工期の遅延に対するフロート計算ではなく、指定日という制約条件に起因するフロートとなっていることに注意しておく必要がある。制約条件を満たしている弱い制約条件によって、クリティカルパスが不連続となることはない。

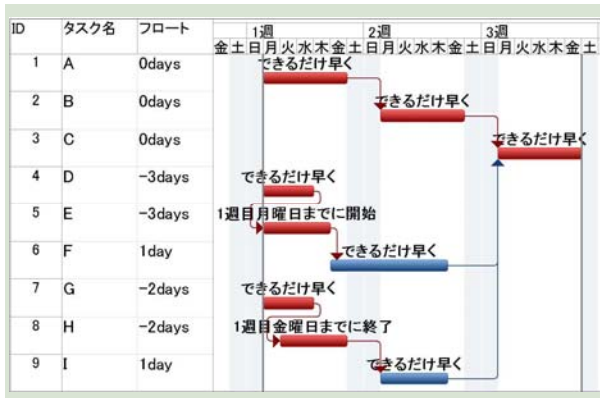


図-7 条件を満たさない“指定日までに・・・”のやや弱い制約条件によるクリティカルの発現

図-7は指定日の制約条件を満たしていない場合の“指定日までに・・・”の制約条件が付与されているケースである。指定日を満たすために必要な回復日数がネガティブフロートとして計算され、“指定日までに・・・”の弱い制約条件が割り付けられているタスクはクリティカルとなる。“先行タスク D, G”も同様に指定日の制約条件を満たすために必要な回復日数がネガティブフロートとして継承されているのに対し、“後続タスク F, I”はクリティカルにはならず、クリティカルパスが不連続となっている。ただし、このクリティカルパスはプロジェクト全体のクリティカルパスではなく、制約条件を満たすためのフロート計算に起因するクリティカルパスである。

プロジェクトの計画時点で指定日条件を満たしていても進捗に応じて実績値を記録することによって、この制約条件が割り付けられているタスクが遅延して制約条件を満たさなくなった場合には、全体工程に対するフロート計算でなく、指定日条件に起因するフロート計算によって、新たな経路のクリティカルパスが発生する原因となることに留意しなければならない。



図-8 期限日を設定した例

実際の工程管理の実務でやや弱い制約条件を利用するケースで一番多く見られるのが“指定日までに終了”であり、プロジェクトの中間マイルストーン

が契約で規定されている場合である。やや弱い制約条件を付与すると、図-7に示したとおり制約条件を満たさなくなった場合に論理的依存関係よりも制約条件を優先してスケジュールされてしまうが、制約条件の代わりに期限日を設定すると、図-8に示すとおり、論理的依存関係を維持したスケジューリングをしてくれる。フロートの計算は期限日を起源とした計算となり、ネガティブフロートは発生する。よって、動的なスケジューリングを確保するために、やや弱い制約条件“指定日までに・・・”はなるべく使用せず、開始日あるいは終了日にマイルストーンを設定し、期限日を付与して管理するほうが良い。

制約条件を満たしていないやや弱い制約条件“指定日までに・・・”や期限日に起因するクリティカルパスの不連続への対策は、単純に制約条件を満たすように施工するか、制約条件の指定日や期限日をずらすことでクリティカルを発生させないことである。計画段階で制約条件を満たしていないことは通常では考えられないので、計画と異なった事象の発生によって遅延が発生して中間マイルストーンの契約条件を満たさなくなったと考えられる。遅延の理由が正当に工期延長の権利のないものであれば、自己責任で工程を回復しなければならない。遅延の理由が正当に工期延長の権利のあるものであれば、工期延長してもらって制約条件の指定日や期限日をずらすことになる。

(3) 強い制約条件

強い制約条件には、“指定日に開始”や“指定日に終了”がある。これらの制約条件は条件が固定的であるため、先行タスクや後続タスクの余裕度（フリーフロート）の有無に関わらず、常にクリティカルタスクとなる。このタスクの前後のタスクのトータルフロートがなければクリティカルはそのタスクにも継承される。トータルフロートが正の値であれば、クリティカルはそこで途切れ、クリティカルパスの不連続の原因となる。



図-9 “指定日に・・・”の強い制約条件

図-9において，“タスク C”は1週目金曜日を指定日とする“指定日に開始”，“タスク F”は2週目水曜日を指定日とする“指定日に終了”の制約条件が割り付けられている。ともに，“先行タスク B, E”には1日の余裕期間（トータルフロート）があり，“後続タスク D, G”にも3日の余裕期間（トータルフロート）がある。しかし，“指定日に・・・”の制約条件が割り付けられているため，“タスク C, F”は前方にも後方にも動かすことができないため、クリティカルタスクとなる。

実務上、強い制約条件を付与しなければならないケースはほとんどない。また、この制約条件に起因するクリティカルパスの不連続性に関して、クリティカルとなる範囲が極めて限定的なため、問題を特定しやすい。このクリティカルは明らかに指定日に起因するものであり、前後のタスクによって動的に動かないことを考慮に入れば、このクリティカルは工程遅延分析の対象と看做さないのが適当である。

以上記述してきたように、制約条件は工程上の開始日・終了日の細かい条件設定が可能だが、指定日を基準としてフロートが計算され、クリティカルパスを不連続とする原因の一つとなるため、極力使用しないようにするのが望ましい。どうしても制約条件を付与せざるを得ない場合には、制約条件を設定したタスクの一覧リストを作成しておき、プロジェクトの遂行中いつでもこれらのタスクをトレースできるようにして、工期延長の工程分析の対象となるクリティカルパスのみを特定できるようにすることが重要である。

3. 継続日設定に起因する不連続

作業期間に継続日を設定したタスクは、クリティカルパスの不連続の原因の一つとなることがある。継続日は作業時間に関連した期間ではなく、経過時間に関する期間を表現する機能である。通常一日当たりの稼働時間は8時間で計算されるのに対し、継続日は非稼働時間を含む一日当たり24時間の経過時間で計算される。通常カレンダーの設定に従って土日祭日などの非稼働日を除いて計算されるのに対し、継続日は土日祭日などを含んで計算される。例えば、コンクリートの養生期間は土日祭日などの非稼働日に関係なく時間で規定されるため、継続日を利用してコンクリートの養生期間を表現する場合がある。



図-10 継続日に起因するクリティカルパスの不連続

図-10のように、コンクリート打設が月曜日で、コンクリート養生期間が4継続日の場合を考える。コンクリート打設は通常の稼働時間に合わせて、月曜日の朝8時から夕方17時までとなり、養生は月曜日の夕方17時から金曜日の夕方17時となる。養生に後続する脱枠作業は翌土曜日の朝8時から開始となるため、金曜日の夕方17時から土曜日の朝8時までの時間に、後続タスクの非稼働時間帯のフロートが生じる。つまり、養生作業が翌土曜日の朝8時まで15時間遅延したとしても、後続タスクの型枠脱枠作業の開始時間に影響を及ぼさないのである。

継続日を作業期間に適用した場合に発生するクリティカルパスの不連続に対しては、非稼働時間帯のフロートを加味した期間を継続時間として設定することで解決することができる。すなわち今回のケースでは

$4 \text{ 日} \times 24 \text{ 時間/日} + 15 \text{ 時間} = 111 \text{ 継続時間}$
で、111継続時間を入力すればよいことになる。

4. カレンダーに起因する不連続

カレンダーはプロジェクト全体やタスク、リソースなどのオブジェクトに対して、稼働日（稼働時間）を設定する機能である。土日祝祭日の設定や特定の曜日の施工制約条件のあるリソースなどの稼働日設定に有効な機能である。例えば日曜祭日にはコンクリート供給ができない場合にカレンダーの設定をすることで動的に日曜祭日にコンクリート打設が計画されないように管理することができる。カレンダーの設定は実際のプロジェクトにおいて現実的な稼働日設定ができるため、非常に有用な機能である。

しかし、このカレンダーの機能によって、フロートが発生してクリティカルパスが不連続となる場合がある。



図-11 複数のカレンダーによるクリティカルパスの不連続

図-11において、先行タスクには週7日稼働用カレンダー、後続タスクには週6日稼働用カレンダー（日曜日休み）が割り付けられている。先行タスクは土曜日で終了しているが、後続タスクは日曜日が非稼働日のため最早稼働日の月曜日が開始日となっている。先行タスクは1日の作業延伸・遅延が発生しても後続タスクの開始日に影響を及ぼさないため、1日のフロートが発生しており、クリティカルパスは不連続となる。

カレンダーの稼働日設定に起因するクリティカルパスの不連続への対策としては、次の3つの方法がある。

- クリティカルの判定基準をフロート日数0日から1日（連休がある場合にはその連休の日数）へと変更すると、クリティカルパスが連続となる。ただし、カレンダーの稼働日設定に起因しない本来クリティカルパスとされないタスクまでクリティカルパスとして表示されることになるため、この方法は適切な手法ではない。
- 対象となる非稼働日に仮のダミータスクを挿入してクリティカルパスを連続化させる。静的に工程分析を行う場合には、その場しのぎではあるがこの手法が最適である。ただし、その後の実績記録によって工程が動的に変動した場合には、適宜このダミータスクを削除する必要がある。
- 対象となる非稼働日の日数と同期間のラグタイムを設定する。この手法も前項b)と同様にその場しのぎではあるが、クリティカルパスを連続化することができる。この場合にもその後の実績記録によって工程が動的に変動した場合には、適宜このラグタイムを削除する必要がある。

カレンダーの非稼働日設定に起因するクリティカルパスの不連続は、実際にフロートが発生しているため、厳密な意味でクリティカルパスが不連続となるのは正しい。工程分析の際には上記のどの手法を取って分析するのかを発注者側に事前に理解してもらって承諾を得ておく必要がある。

5. 実績値の記録に起因する不連続

プロジェクトの工程管理は、契約上の工程管理の基準となる基本工程（ベースライン工程）に示した論理的依存関係に従って、実績値を記録して全体工程を更新しながら行うことになる。実績記録を行っ

ていくことで、各作業の論理的依存関係が維持されているため、プロジェクトの全体工程は動的に管理されてゆく。しかし、工程表が動的に管理されるためには、この論理的依存関係に変更がないことが前提であり、実際の施工手順が論理的依存関係と異なる場合には、実績記録を行うことによってクリティカルパスが不連続となることがある。

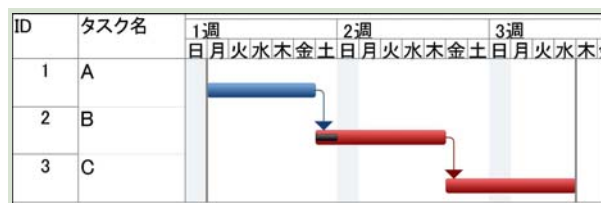


図-12 実績値入力によるクリティカルパスの不連続

例えば、図-12のように“タスク A, B, C”が直列に連結されて全てのタスクがクリティカルパスで計画されていた工程に、“タスク A”の実績値を記録せずに“タスク B”の実績値を記録した場合、“タスク A”はクリティカルタスクでなくなり、クリティカルパスが不連続となる。

クリティカルタスクであった“タスク A”がクリティカルでなくなった理由は、クリティカルタスクである“タスク B”の開始条件が“タスク A”の終了だったが、“タスク B”の開始実績が記録されてしまったために“タスク A”が1日遅延してもその後の“タスク B”や“タスク C”に影響を及ぼさなくなったためである。より詳細に考察すると、実績記録のなされた“タスク B”の残存部分の開始日、すなわち2週目の日曜日が現状での“タスク B”の開始日と認識されることになり、先行タスクである”タスク A”の終了予定日である1週目の金曜日との差で、”先行タスク A”のフロートは1日と計算されることになる。

クリティカルパスが不連続となったのは、当初の論理的依存関係と異なった実績値を入力したからであり、工程分析を行う場合には、当初の論理的依存関係を修正するかどうか、修正するのであればどのように修正するのかを発注者と協議して事前に承諾を得る必要がある。具体的な修正方法は論理的依存関係を実績に合致するように組み替える方法と、ラグタイムで修正する方法等が考えられる。

7. リソース平準化に起因する不連続

工程管理ソフトウェアには、リソースを平準化する機能があり、このリソースの平準化⑥の機能がクリティカルパスの不連続の原因となる場合がある。

リソースの平準化とは、作業の開始を遅らせたり分割したりすることによって、リソースの重複割当や割当超過を解決する方法である。リソースの最大使用量制限等の制約条件を満たすように、作業の開始日・終了日を調整してくれる。

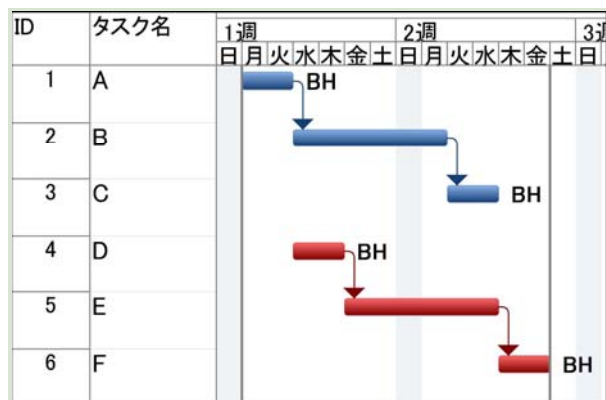


図-13 リソースの平準化に起因するクリティカルパスの不連続

リソースの平準化の機能について、例えば図-13を参照されたい。“タスク A, B, C”及び“タスク D, E, F”が直列に連結されている。“タスク A, C, D, F”には“BH”というリソースが割り付けられており、このリソースの最大使用量制限として1台が設定されている。つまり同時期に“BH”リソースは1台しか利用できない。例えば、“タスク A”と“タスク D”は“BH”リソースが割り付けられているため、同時期には施工できず、“タスク D”は“タスク A”が終了してから開始するように“タスク D”の開始日が調整されている。

この工程表でのクリティカルパスは“タスク D, E, F”と、プロジェクトの途中から始まっており、クリティカルパスは不連続となっている。“タスク A, B, C”はクリティカルパスと表示されていない。

“タスク A, B, C”がクリティカルではないと表示される理由について考察する。“タスク A, B, C”のフロートを調べると2日のフロートをもっている。例えば“タスク D”が仮に1週目月曜日から開始した場合を想定すると、“タスク A”は1週目の水曜日までに開始すれば全体工程に影響を及ぼさない。同じリソースを共有する“タスク A”と“タスク D”との間に明示的な論理的依存関係が存在しないため、どちらを先に施工してもよいためにフロートが発生し、非クリティカルとなる。

しかし、“タスク A”の作業期間を1日延長して3日としてみると、更新された工程は図-14 のようになる。

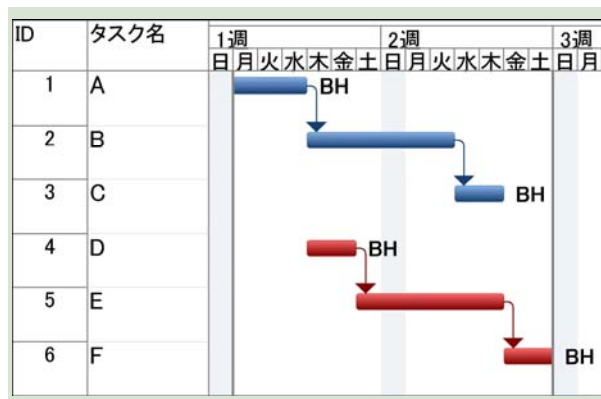


図-14 非クリティカルタスクの延長

非クリティカルタスクである“タスク A”の期間を1日延長すると、プロジェクトの全体工期は2週目の金曜日から土曜日へと1日延伸されることになる。本来作業期間の遅延や延長が全体工期に影響を及ぼすようなタスクはクリティカルタスクとなるべきであり、“タスク A”はクリティカルタスクだと考えるのが妥当である。

クリティカルパスが不連続となる理由は、リソースの割り付けられたタスク間に明示的な依存関係がないためであり、クリティカルパスの不連続への対策はこの依存関係を設定することである。

図-15は図-13の工程表における“BH”リソースの施工順番を論理的依存関係で連結したものである。リソースの施工順番を明示的に設定することで、図-13の全てのタスクがクリティカルタスクであることがわかる。リソースの施工順番は通常プロジェクトの進捗に合わせて変化するため、リソースの施工順番を明示的に設定する作業は工程分析作業でのみ静的に行うのが良いと考える。

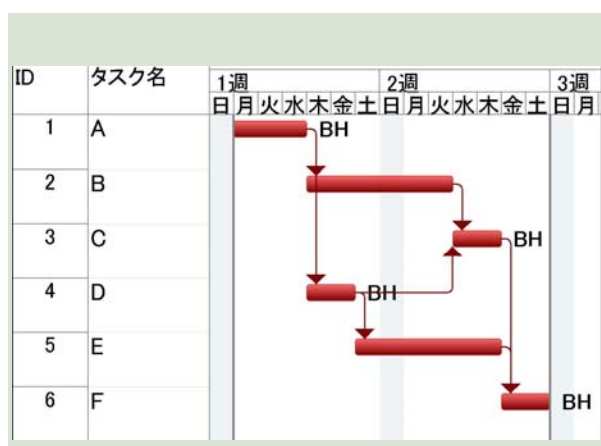


図-15 リソースの順番を論理的依存関係で連結した工程例

作業の論理的依存関係には大きく2種類のタイプがあり、物理的あるいは契約的にその順番でなければ施工できない強制依存関係⁷⁾と、施工者の都合で設定する任意依存関係⁸⁾がある。例えば前者は橋梁

工事における下部工→上部工の施工順番であり、後者は基礎杭打設順番の基礎P1→基礎P2などが上げられる。

強制依存関係はプロジェクトの進捗中に変更されることはほとんどない依存関係であるが、任意依存関係は例えば型枠の転用計画や班の施工順序計画に従って随時見直され、変更される可能性のある依存関係である。

契約上求められる基本工程（ベースライン工程）が契約図書となることを考慮し、容易に変更・修正できるものではないため、提出する基本工程（ベースライン工程）には強制依存関係のみを設定すべきである。ただし、施工者の施工前提条件であるリソースの転用計画や施工順番はプロジェクトを進める上での重要な施工戦略であるため、補足文書として任意依存関係も設定した工程表も提出しておく必要があると考える。任意依存関係を設定していない工程表はリソースの平準化によるクリティカルパスの不連続の原因となるため、強制依存関係のみの工程では工程分析が不十分なものになる可能性が高く、プロジェクトの進捗に合わせて変化する任意依存関係の補足文書を添付し、この補足文書を更新することで、基本工程（ベースライン工程）を変更せずに、動的な基本工程（ベースライン工程）の更新ができるからである。

8. 結論

以上、クリティカルパスが不連続となる原因と対策について考察し、まとめた。国際建設プロジェクトにおいて、契約と異なる事象によって発生する全体工期の遅延は、その遅延原因が請負者の責任でな

く、かつその事象によって全体工程に影響を及ぼすことを証明できた場合に工期延長クレームすることができる。全体工程に影響を及ぼすことを証明するためには、その事象がクリティカルパス上にあることを証明する必要があるが、上記にまとめたとおりに様々な理由でクリティカルパスが不連続となる場合があり、工期延長クレームを適切に作成することを妨げるケースが多い。

本論文が国際建設プロジェクトにおける工期延長クレームのための、クリティカルパスの不連続問題の解決の助けとなれば幸いである。

参考文献

- 1) 草柳俊二：定量的分析を基盤とした国際建設プロジェクトの契約管理，土木学会論文集No.609/Vi-41, 87-09, 1998.12
- 2) 草柳俊二：国際建設プロジェクトの実務から見た建設市場開放に関する課題と対策，土木学会論文集No.510/Vi-26,165-174,1995.8
- 3) 国際建設プロジェクトの契約管理◇基礎知識と実務◇，社団法人海外建設協会,2009.1
- 4) A Guide to the Project Management Body of Knowledge Fifth Edition：Project Management Institute Inc.,6. Project Time Management, pp176, 2013
- 5) 武井輝国：国際契約工事管理 そのクレーム管理技術,2007
- 6) A Guide to the Project Management Body of Knowledge Fifth Edition：Project Management Institute Inc.,6. Project Time Management, pp179, 2013
- 7) A Guide to the Project Management Body of Knowledge Fifth Edition：Project Management Institute Inc.,6. Project Time Management, pp157, 2013
- 8) A Guide to the Project Management Body of Knowledge Fifth Edition：Project Management Institute Inc.,6. Project Time Management, pp158, 2013

(2014. 10. 21 受付)

Discussion on Reasons and Countermeasures for a Fragmented Critical Path

Jin OKAMOTO and Hiroshi MASUYA

In international construction projects, contractors are required to dynamically manage the project schedule with critical path method in order to meet the completion date agreed in the contract. The critical path method is an effective way to dynamically manage the completion date. However, the critical path could be fragmented due to various reasons. The claim for “Extension of Time” is only valid when the delay is caused outside of scope and proves to affect the project completion date. Therefore, it is impossible to develop a logical claim document with a fragmented critical path programme. This paper discusses and suggests various reasons and countermeasures for this issue in order to develop effective EOT claim documents.