

# ソフトウェア開発計画自動立案システム ～ファースト・トラッキングを行う工程の選択方法～

塩田 智子<sup>†</sup> 高須賀 公紀<sup>†</sup> 林雄一郎<sup>‡</sup> 木下大輔<sup>‡</sup> 八重樫理人<sup>‡</sup> 橋浦弘明<sup>†</sup> 古宮誠一<sup>†</sup>  
芝浦工業大学 工学部 情報工学科<sup>†</sup> 芝浦工業大学 大学院 工学研究科<sup>‡</sup>

## 1. はじめに

我々は制約に基づくソフトウェア開発計画の自動立案システムについての研究を行っている[1][2][3]。

工程遅延発生時の対策案の立案方法として、これまではクラッシングという手法が使われてきた[2][4]。クラッシングとは余剰要員をプロジェクトのクリティカルパス上の工程に追加投入する事でプロジェクト全体の所要期間の短縮を図る方法である。しかしながら、さらに短期間で開発するためには、Agile と呼ばれる超短納期ソフトウェア開発を採用しなくてはならない。そこで、ファースト・トラッキング[4]による工程遅延の回復案の立案方法を提案する。ファースト・トラッキングとは先行作業が終了する前に後続作業を開始することである。これまでは、ファースト・トラッキングを行うことはソフトウェア開発ではリスク回避の視点から、タブーとされてきた。しかしながら、超短納期でのソフトウェア開発を可能とするために、このタブーを冒してでも工程遅延の回復を図る必要がでてきた。そこで、できるだけリスクの少ないクリティカルパス上の工程に対してファースト・トラッキングを行い工程遅延に対する対策案を立案する。

本稿では、工程遅延に対するファースト・トラッキングによる対策案の立案のためのシステムの変更点、各工程の持つ技術的リスク、ネットワークの位置によるリスク(以下ネットワークリスクと呼ぶ)、全体的リスクの計算方法とファースト・トラッキングを行う工程の決定方法について述べる。

## 2. ソフトウェア開発計画立案問題における制約

古宮らは、ソフトウェア開発計画が満たさなくてはならない条件を制約とし、ソフトウェア開発計画立案問題を制約に基づく組み合わせ最適化問題を解くことだと捉えた[1]。

その制約には、(1)作業順序に関する制約、(2)リソースの割り当て条件に関する制約、(3)リソース割り当て可能期間に関する制約、(4)リ

ソースの能力的限界に関する制約の4種類がある。しかしながら、ファースト・トラッキングを行うためには(1)の作業順序に関する制約の解釈を改める必要が出てきた。

## 3. ファースト・トラッキング

### 3.1 従来の作業順序に関する制約の解釈と制約条件の解釈の変更

古宮らの作業順序に関する制約の解釈を以下に記す。

「作業順序に関する制約」

ソフトウェア開発の各工程は、中間成果物を媒介して、それらの実施順序が決定する。例えば、工程 b を実施するには、工程 a による中間成果物  $\alpha$  が工程 b に着手する前に生成されていなければならないとする。これを工程 b の事前条件と呼ぶ。また、工程 b の中間成果物  $\beta$  が工程 c に着手する前に生成されていなければならないとする。これを、工程 b の事後条件と呼ぶ。このため、中間成果物  $\alpha$ 、 $\beta$  によって、工程 a、b、c の実施順序が決まる。このような制約を作業順序に関する制約と呼ぶ。

ファースト・トラッキングを実施するためには、これまでの作業順序に関する制約の解釈を緩める必要がある。そこでファースト・トラッキングを実施できるようにする為に、ファースト・トラッキングを適用する工程だけは先行作業の完了を待たずに後続作業が開始できるようにした(図1参照)。

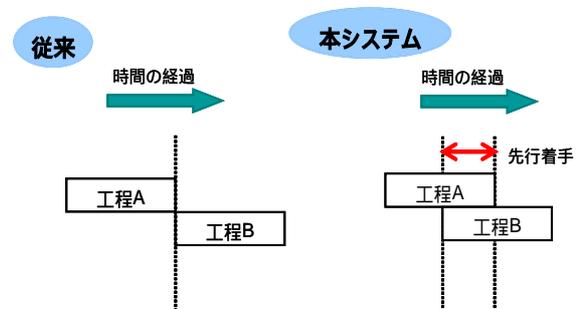


図1：作業順序に関する制約の解釈の変更

T.Shioda<sup>†</sup>,K.Takasuka<sup>†</sup>,Y.Hayashi<sup>‡</sup>,D.Kinoshita<sup>‡</sup>,R.Yaegashi<sup>‡</sup>,H.Hashiura<sup>†</sup> and S.Komiya<sup>†</sup>  
<sup>†</sup>Shibaura Institute of Technology  
<sup>‡</sup>Graduate School of Shibaura Institute of Technology

### 3.2 ファースト・トラッキングを行う工程の選択方法

ファースト・トラッキングを行った場合は、先行作業が完全に終了した後に後続作業を行う場合に比べて、先行作業によって、予め定められた仕様通りに中間成果物が得られる保証がないので、リスク(失敗可能性の予測値)が当然高くなる。それでもファースト・トラッキングを導入せざるえない場合には、技術的リスクとネットワークリスクから全体的リスクを求め、全体的リスクが最小となるクリティカルパス上の工程を選択し、選択された工程に対してファースト・トラッキングを行うことにした。

### 3.3 技術的リスクの見積り方法

技術的リスクとは、各工程が持っている技術的なリスクのことである。技術的リスクは事前にプロジェクト管理者によって見積られ、システムに与えられている。

### 3.4 ネットワークリスクの計算方法

ネットワークリスクの計算方法は、直列構成でも並列構成でも同じである[3]。計算方法は以下になる。

$$R_N = 1 - \prod A_j$$

$$= 1 - \prod (1 - r_j)$$

$r_j$ :スケジュール表(ネットワーク)上の各工程のリスク  
 $A_j$ :スケジュール表(ネットワーク)上の各工程稼働率  
 $R_N$ :スケジュール表(ネットワーク)上の位置による各工程リスク

### 3.5 全体的リスクの計算方法

工程  $i$  の全体的リスクは以下のように定義した[3]。

$$R_i = 1 - (1 - R_{iT})(1 - R_{iN})$$

$R_i$ : 全体的リスク  $R_{iT}$ : 技術的リスク  
 $R_{iN}$ : ネットワークリスク

この式の妥当性は次のように考えれば理解できる。(  $1 - R_{iT}$  ) は技術的リスク  $R_{iT}$  のみを考慮したときの工程  $i$  の稼働率に相当し、(  $1 - R_{iN}$  ) はネットワークリスク  $R_{iN}$  のみを考慮したときの工程  $i$  の稼働率に相当する。一方、(  $1 - R_{iT}$  ) (  $1 - R_{iN}$  ) は技術的リスク  $R_{iT}$  とネットワークリスク  $R_{iN}$  を同時に考慮したときの工程  $i$  の稼働率に相当する。このとき、 $1 - R_i = (1 - R_{iT})(1 - R_{iN}) = 1 - R_{iT} - (1 - R_{iT})R_{iN}$   $1 - R_{iT}$  が常に成り立つので、 $R_i$   $R_{iT}$  が常に成り立つ。同様に  $R_i$   $R_{iN}$  も常に成り立つ。即ち、技術的リスク  $R_{iT}$  とネットワークリスク  $R_{iN}$  を同時に考慮したときの(工程  $i$  の)リスクの値は、 $R_{iT}$  または  $R_{iN}$  のみを考慮したときの(工程  $i$  の)リスクの値よりも、常に小さくないことが示される。このことは、技術的リスクとネットワークリスクを同時に考慮す

ることが無意味でないことを示している。求められた全体的リスクが最小の工程に対してファースト・トラッキングを行う。

工程で工程遅延が発生した場合を例に挙げて、ファースト・トラッキングを行う工程の選択方法を説明する(図2参照)。

図2では工程 c、工程 d がクリティカルパス上の作業であるとする。従って、工程 c、工程 d のいずれかの工程においてファースト・トラッキングを行う。この時、工程 c、工程 d の全体的リスクはそれぞれ、0.28、0.352 となる。全体的リスクが最小の工程に対してファースト・トラッキングを行うので、工程 c が 0.28 と最小の値になる。よって、工程 c に対してファースト・トラッキングを行う。このようにして、ファースト・トラッキングを行う工程を選択する。

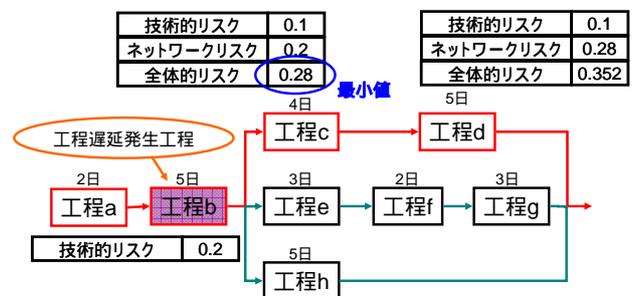


図2：ファースト・トラッキングを行う工程の選択例

## 4. 終わりに

本稿では、工程遅延に対するファースト・トラッキングによる対策案の立案のために、制約条件の変更、ファースト・トラッキングを行う工程の選択方法、各工程の持つ技術的リスク、ネットワークリスク及び全体的リスクの計算方法とファースト・トラッキングを行う工程の決定方法を例題を用いて述べた。

[参考文献]

- [1] 古宮誠一、澤部直太、樫山淳雄“制約に基づくソフトウェア開発計画の立案,” 電子情報通信学会論文誌 D-I Vol.J79-D-I No.9, pp.544-557, 1996.
- [2] R.Yaegashi, D.Kinoshita, H.Hashiura, K.Uenosono, S.Komiya, "Automatically Creating a Schedule Plan as Countermeasures by Means of "Crashing" against Process Delay," Joint Conference on Knowledge-Base Software Engineering 2004 (JCKBSE2004), No4, pp.24-36, 2004.
- [3] 八重樫理人、木下大輔、橋浦弘明、上之園和宏、古宮誠一, "工程遅延発生時におけるファースト・トラッキングによる対策案の自動立案," 電子情報通信学会「システム開発論文」論文誌特集号, 平成 17 年 2 月掲載予定.
- [4] Project Management Institute (著) A Guide to the Project Management Body of Knowledge (PMBOK Guide) 2000 Edition, 2000.