2-H-9

日本オペレーションズ・リサーチ学会
2004年秋季研究発表会

ソフトウェア危機のMythical Man×Monthを説明する数理モデル

01205220　日本大学生産工学部　篠原　正明
日本大学生産工学部　蜂須　和則
02005660　日本大学生産工学部　*内山　貴夫

1　はじめに

大規模ソフトウェア開発など要員間のコミュニケーションに相当する仕事が必然的に発生するタイプの仕事においては、仕事の完了納期を短縮しようとして、要員を増加しても、効果があがらないどころか逆効果となってしまう現象が報告されている。それの典型例が、ソフトウェア危機であり、Mythical Man×Month、でもある。

本論文では、注目する仕事（ソフトウェア開発、プロジェクト、など）に関して、投入人数に依存した仕事関数（Work Load）を考え、その仕事量関数と本来的に仕事量は Man×Month、あるいは Man×Day、で定義される定義式との非線形連立方程式として、投入人数と所要期間が定まる非線形システムを提案する。これにもとづき所要期間が与えられ、複数の解（動作点）が存在する時は、各動作点の安定性を議論できる。又、最短の所要時間の求め方、ソフトウェア開発プロジェクトの運用上の知見などについて考察する。

2　仕事量関数の導入

ソフトウェア開発など注目する仕事の仕事量（Work Load）y がそれに投入される人数 x の関数 $f(x)$ で表現できると仮定する。

$$y = f(x) = f_0 + f_1 x + f_2 x^2 \qquad (1)$$

ここでは、f(x)を２次の項までとする。

さて、仕事量は man×month あるいは man×day の単位を持ち、従って、完了のための所要時間を t とすれば、次式が定義式として成立する。

$$y = tx \qquad (2)$$

仕事量関数 f(x)の例を以下に説明する。

例１　人数に依存しない場合（タイプⅠ）

広い畑を x 人で耕作し、耕作機械は十分存在すると仮定すれば、その仕事量関数は(3)式のタイプとなる。この場合は、投入人数に反比例して、畑耕作完了の所要時間 t は減少する。

$$f(x) = f_0 \text{(定数項)} \qquad (3)$$

$$t = f_0 / x \qquad (4)$$

例２　人数比例項が含まれる場合（タイプⅡ）

手書き資料をワープロ作業する場合を考えよう。作業者はワープロ技能を身に付けておらず、技能習得のため１人当たり f_1 時間必要とする。

$$f(x) = f_0 + f_1 x \qquad (5)$$

従って、所要時間 t は、定数項 f_1 と x 反比例分の和で与えられる。

$$t = \frac{f_0}{x} + f_1 \qquad (6)$$

例３　担当者間のコミュニケーションが必要となる場合（タイプⅢ）

大規模なソフトウェア開発に大勢のソフトウェア開発技術者が個別に独立して従事する場合を想定しよう。個々の技術者は一対一で個別に開発上の必要な情報を交換するならば x 人の間の一対の数は $x(x-1)/2$ となり、２次項が生じる。f(x)と t

$$f(x) = f_0 + f_1 x + f_2 x^2 \qquad (7)$$

$$t = \frac{f_0}{x} + f_1 + f_2 x \qquad (8)$$

は次式となる。

3　簡単な例題

仕事量関数 $f(x)$ が(9)式で与えられる仕事を想

$$y = f(x) = 3 + x + x^2 \text{(人日)} \qquad (9)$$

定しよう。

この仕事を t=5 日で完了するためには何人投入すればよいか？　この問題を解くためには、(9)式と

$$y = tx = 5x \qquad (10)$$

(10)式を連立すればよい。(図 1)

従って、$5x = 3 + x + x^2$ の２次方程式を解いて、答えは x =1 と x =3 である。すなわち、この例では、

「1人」と「3人」という2つの答えが存在することになる。

4 安定性の議論

(9)式と(10)式の方程式において以下の方向性を仮定する。すなわち、(9)式においては、投入人数xが与えられると仕事量が評価でき、t 一定の(10)式において、仕事量yが与えられると投入人数が評価できる。

$$y \leftarrow 3 + x + x^2 \qquad (11)$$
$$x \leftarrow y / 5 \qquad (12)$$

(11)式右辺におけるxの微小変動が一巡して(12)式で左辺xに及ぼす効果を巡回係数利得 P で評価する。

$$\mathrm{P} = f'(x)/t = (1+2x)/5 \qquad (13)$$

(i) $x=1$ は、$\mathrm{P}=0.6<1$ となり、安定な動作点である。
(ii) $x=3$ は、$\mathrm{P}=1.4>1$ となり、不安定な動作点である。

以上の考察により、$x=1$ が現実的な解と思われ、$x=3$は一時的には実現したとしても非常に不安定、すなわち、$x=3$人でこの仕事を実施すると、より多くの人数が必要となる状況へ発散する危険を内包していると言える。これこそ、まさに、ソフトウェア危機である。

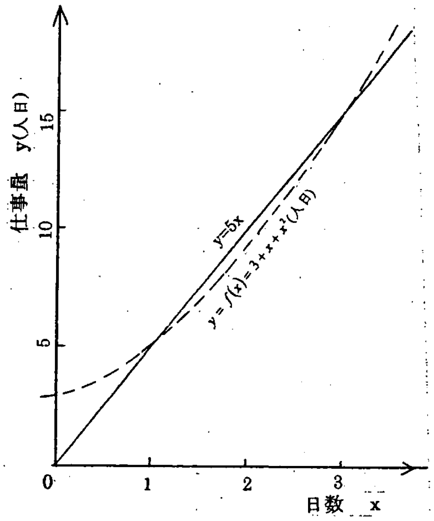

図1 Mythical Man Month の非線形方程式系

5 考察

(1) タイプⅠ($f_0>0$)とタイプⅡ($f_0>0$ $t>f_1$)の仕事は常に1つの安定動作点をもつ。

(2) タイプⅢの仕事では、もし、解が存在するならば1つの安定動作点と不安定動作点の組を持つ。すなわち、不安定動作点が存在しうる状況は、タイプⅢに限定され、タイプⅠとタイプⅡの仕事では、ソフトウェア危機あるいは Mythical Man×Month ごとき現象は発生しえない・・・と言える。

(3) タイプⅢの仕事では、非線形システムを連立非線形方程式として表現した時に、所要時間 t を小さく設定すると交点が存在しない場合が生じる(解なし)。すなわち、タイプⅢの仕事では、その仕事を完了可能な最短時間 T が存在する。T は次式(14), (15)を連立して求まる。

$$\mathrm{T} = f'(x) \qquad (14)$$
$$\mathrm{T}x = f(x) \qquad (15)$$

$$\text{すなわち、} \quad x = \sqrt{\frac{f_0}{f_2}} = \sqrt{3}(\text{人}) \qquad (16)$$

$$\mathrm{T} = f_1 + 2\sqrt{f_0 f_2} = 1 + 2\sqrt{3}(\text{日}) \qquad (17)$$

すなわち、この仕事は$\sqrt{3}=1.732$(人)投入すれば、$\mathrm{T}=1+2\sqrt{3}=4.464$(日)で完了可能であり、これ以上の短縮は無理である。なお、この種の動作点における巡回係数利得 P は常に1である。この点を考慮すれば、この数値は限界値と考えられ、T=4.464よりも多少大き目が現実解であろう。

6 おわりに

ソフトウェア開発に伴う Mythical Man×Month を説明する非線形方程式系の複雑系数理モデルを提案した。動作点の安定性を議論することにより、現実には不安定で存在しないだろう動作点を特徴づけることが出来た。

ソフトウェア工学の立場から言えば、所与のソフトウェア開発プロジェクトの仕事量をいかにして、定数項f_0と1次項f_1x が支配的になるように、再構成することが肝要である。又、そのために、ソフトウェア工学の諸技法が開発されて来たと言っても過言ではない。