

## ソフトウェア品質・原価モデル生成に関する一手法

大和義孝 渡辺順平 黒崎 徹  
(日立 戸塚工場)

### 1. まえがき

近年の大規模ソフトウェア開発においては、開発作業の手戻りをできるだけ少なくするためにソフトウェアの品質および生産性を予測し、それに基づいたリソースを確保し、綿密な生産計画を立案し、着実に実行することが肝要である。このためには精度の高いソフトウェアの品質および生産性に関するモデルを導き出すことが重要である。

ソフトウェアを管理するという観点から、生産性、品質に影響を与えると思はれる要因を選択し、品質、生産性に対するモデル式を導き出した<sup>(1)</sup>。更に大量の生産データを分析し、予測モデル式を容易に導き出すためのプログラムを開発<sup>(2)</sup>、今後のソフトウェアの定量的生産管理を容易にした。

### 2. 品質および生産性の定義

ソフトウェアの品質は、生産されるプログラムに含まれる不良件数の大きさで定まる。これらの不良は、テスト工程および検査工程で検出される。そこで、品質を表すパラメータを、つぎのように定義する。

$$\text{不良検出率} = \frac{\text{テスト・検査工程で検出される不良件数}}{\text{生産プログラム規模}} \quad (1)$$

つぎに、生産性は、単工数で生産できるプログラムの大きさで定まる。そこで、生産性を表すパラメータを、つぎのように定義する。

$$\text{生産効率} = \frac{\text{生産プログラム規模}}{\text{生産工数}} \quad (2)$$

また、生産工程は、設計工程とテスト工程とに大別できるので、生産効率と同様の考え方で、設計効率をつぎのように定義する。

$$\text{設計効率} = \frac{\text{生産プログラム規模}}{\text{設計工数}} \quad (3)$$

ここで、生産プログラム規模は、生産されるプログラムの大きさで、ステップ数で表わし、生産(設計)工数は、生産(設計)に要した工数で、人時で表すことにする。また、分析においては、基準単工数で生産(設計)できるプログラム規模を1として、これに対する比で表わした生産(設計)効率比を用いることにする。

### 3. 品質および生産性に寄与する要因

ソフトウェアの品質および生産性に寄与する要因は表1の如く、大別すると管理的要因と技術的要因に別けられる。今回は、巨視的な品質および生産性を予測する第一ステップとして主に管理的要因を対象に分析した。

品質および生産性の向上を図るためには、

- (1) 既存のプログラムを流用する比率を高める。
- (2) 基本および概要設計に投入する工数の比率を高める。

(3) テスト工程を、オフライン、単体、総合の3工程に大別した場合、前段の工程で可能な限り多くの不良を除去して、総合テスト以降で検出される不良件数を少なくする。

(4) 経験が豊かで技術レベルの高い人員の構成比率を高める。

ことが必要である。

上記の(1)、(2)、(3)、(4)は、それぞれ、プログラムの流用度、設計の工程配分、テストの工程配分、設計熟練度に関するものである。さらに、生産工数は、設計工数とテスト工数とから成る。そこで、これらから、品質および生産性に寄与する要因を表2のように定義する。

情報通信システムについて、設計効率比と不良検出率との関係を表すと、図1になる。ここで、Pは、

$$P = \frac{\text{不良検出件数}}{\text{設計工数}} \quad (4)$$

であって、図1では、基準単位工数の1/10に対する不良件数で示してある。

Pは、表1に示すプログラム固有の技術的要因により変化するものと考えられる。図1の関係におけるバラツキは、このために生じたものと考えられる。そこで、以下では、Pをプログラム固有係数と呼ぶことにし、品質および生産性とそれに寄与する要因との関係を求める場合の補助パラメータとして、用いることにする。これまでに定義した要因(以下設計パラメータ)と、品質および生産性との関係を示すと、図2になり、これらを量的に求めることにする。

表2 採用した要因と定義

プログラム構成係数	= 1 - (流用比率)
設計工程係数	= 1 - (設計工程構成比率)
テスト工程係数	
	= $\frac{\text{総合テスト・検査の不良検出件数}}{\text{テスト・検査の検出不良件数}}$
生産工程係数	= 1/2 [(設計工程係数) + (テスト工程係数)]
設計熟練度係数	= 要員構成比率
但し	
流用比率	= $\frac{\text{流用プログラム規模}}{\text{生産プログラム規模}}$
設計工程構成比率	
	= $\frac{\text{基本および概要設計工数}}{\text{基本、概要および詳細設計工数}}$
要員構成比率	
	= $\frac{\text{方式・機能・詳細設計可能な要員工数}}{\text{全員工数}}$

表1 品質および生産性に寄与する要因

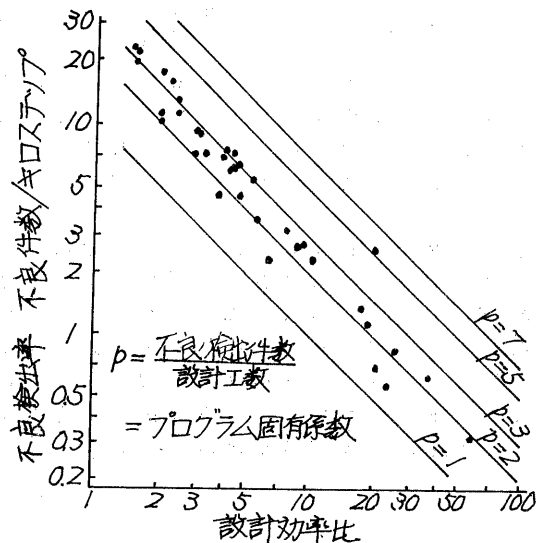
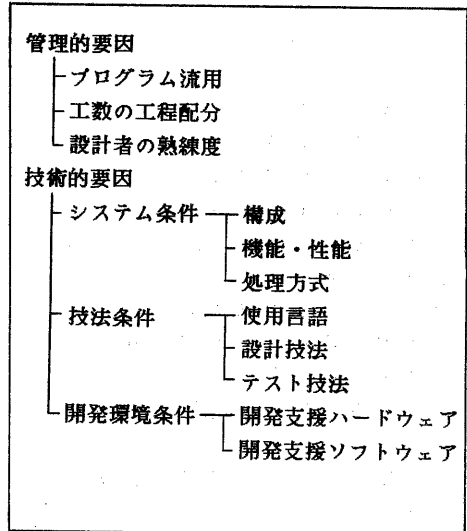


図1 設計効率比と不良検出率

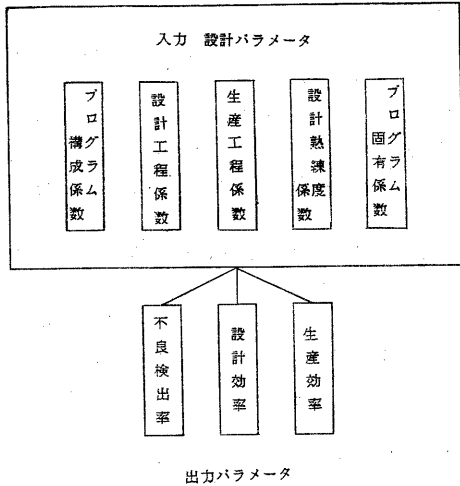


図2 品質および生産性解析系統図

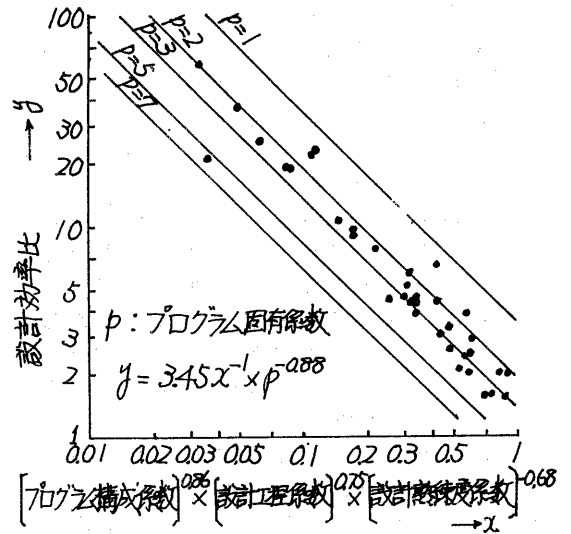


図3 設計パラメータと設計効率比

4. 分析結果

情報通信システムのソフトウェアに関するデータを集積し、これに基づいて、図2に示した関係について、重回帰分析<sup>(3)</sup>を行なった。この結果つぎの関係、すなわち、

- (1) 設計パラメータと設計効率比 図3
  - (2) 設計パラメータと生産効率比 図4
  - (3) 設計パラメータと不良検出率 図5
- が量的に得られた。これらの図は、いずれも

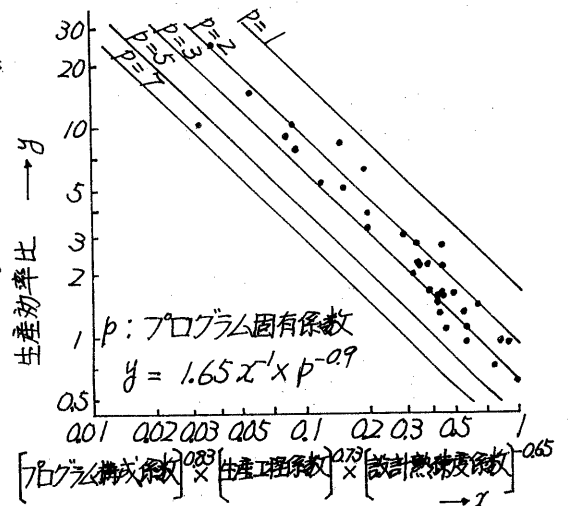


図4 設計パラメータと生産効率比

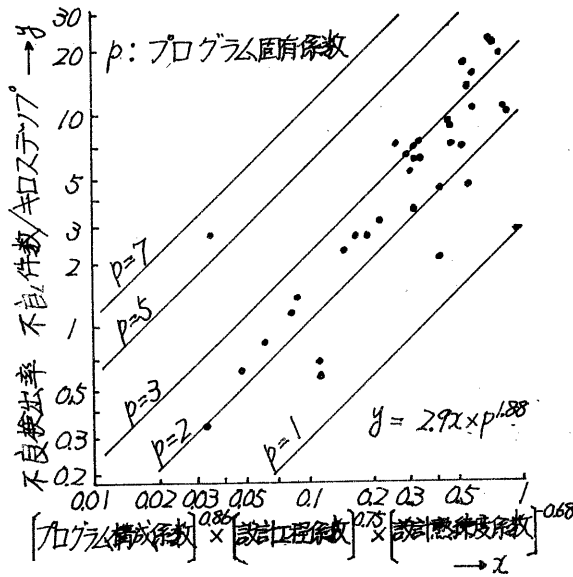


図5 設計パラメータと不良検出率

プログラム固有係数をパラメータとして、表したものである。

さらに、図3～図5の各図に示した式を、それぞれの関係に回帰式としてあてはまることはF-検定の結果、危険率0.5%以下で有意であった。

### 5. 品質および生産性の予測方法

4項の分析結果を用いて、入力である設計パラメータ（プログラム構成係数、設計および生産工程係数、設計熟練度係数、プログラム固有係数）から、出力である品質および生産性（不良検出率および生産効率、設計効率）を予測する方法を、系統的に示したものが、図6である。図6における予測手順を述べると、つぎのようになる。

(1) 設計開始時点で、プログラム構成係数、設計および生産工程係数、設計熟練度係数およびプログラム固有係数を設定する。

表1の定義からプログラム構成係数、設計工程係数、設計熟練度係数は、それぞれ流用比率、設計工程構成比率、要員構成比率により定まる。同じく、生産工程係数は、設計工程係数とテスト工程係数によって定まり、テスト工程係数は、総合テスト以降の工程で検出された不良件数の全検出不良件数に対する比率である。

また、プログラム固有係数は、同種の装置に関するプログラムの過去の実績値に基づいて定める。

(2) 上記を入力として、図3、図4により、設計効率比、生産効率比の予測値を求め、これを効率に換算することにより、設計効率および生産効率の予測値が得られる。これらの予測値の逆数に生産プログラム規模を乗ずることにより、推定設計工数および推定生産工数を求める。

(3) 推定生産工数と推定設計工数との差を求めることにより、推定テスト工数を得る。

(4) (1)の設定値を入力として、図5により、不良検出率の予測値を求め、これに生産プログラム規模を乗ずれば、テスト工程および検査工程で検出されるべき不良件数の和の推定値を得る。

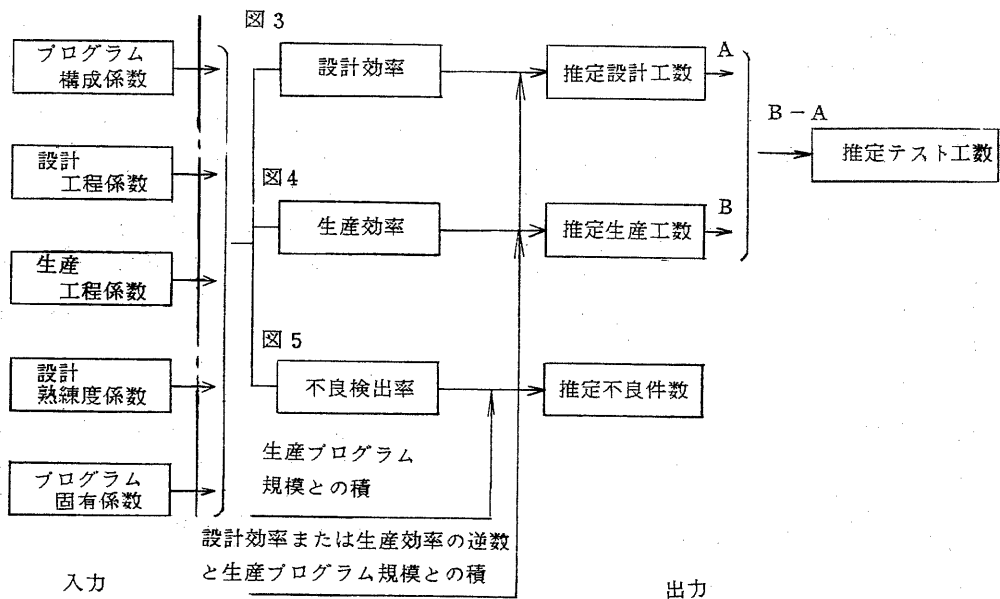


図6 品質および生産性予測に関する系統図

## 6. データ分析とモデル式の位置付

ソフトウェア生産管理体系におけるデータ分析とモデル式の位置付けは図7.1の通りである。

すなわちプログラム開発に当っては、設計の開始以前に、開発しなければならないプログラム規模、開発期間、保有する人員等により開発計画を立てる。その計画に基づいて品質、工数等の予測値を算出し、品質、生産性の目標値と日程計画を立てる。それに従ってプログラム開発を行う。プログラムの開発の過程における各工程毎の工数、品質等の各種データを収集し、集計する。

この集計した実績データと目標値および日程計画等の予定と比較し、差があれば実績値に基づいて再予測を行い、目標値、日程計画等の修正を行う。この再予測は図7.2に示すように、概要設計および詳細設計の終了時点で行う。

一方各プロジェクトの開発過程で収集したデータを集計し、定期的に分析を行い新たなモデル式を算出し、モデル式の更新を行う。このモデル式により、その後の予測値算出を行う。この様にデータ分析とモデル式はソフトウェアの生産管理体系の中に組込んで、定量的な生産管理を可能にした。

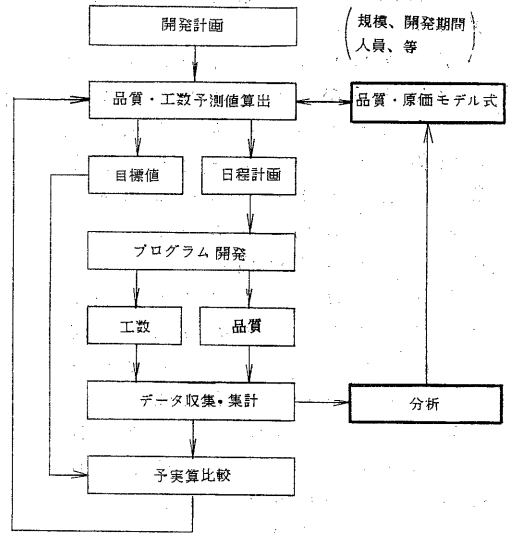


図7.1 ソフトウェア生産管理体系

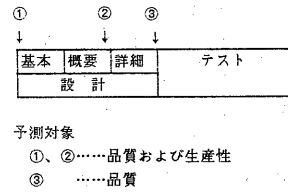


図7.2 品質および生産性の予測時期

## 7. 品質および原価モデルの自動生成

上述した如く、過去の実績データを収集し、分析することにより、巨視的な品質および生産性を予測するモデル式を導き出す手法を見出した。このモデル式は、ソフトウェアの開発環境の変化などにより値が変わるので、定期的に更新する必要がある。ところが最近ではデータ量が多くなり、かつ分析の迅速化が必要なので、データ収集と分析をコンピュータの会話形機能を活用し自動化した。

図8にこの自動化プログラムの機能階層を示す。ソフトウェア品質・原価モデル生成システムは、データ収集機能、データ演算機能、データ分析機能の三つのサブシステムで構成している。

### (1) データ収集機能

品質および工数に関する粗データは個別のデータベースに蓄積してある。即ち図9において、品質データベースは不良検出の度に発行する変更連絡票を随時登録し、工数

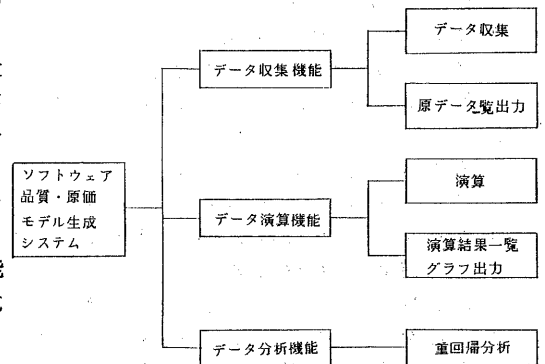


図8 機能階層図

データベースは作業時間を設計者毎に記録した工数票を定期的に登録している。

データ分析を行うためにデータを収集する場合は、まず、プロジェクト名をTSS端末から入力し、収集すべきデータを指定し、品質・工数の各データベースの検索機能により該当のプロジェクトに関するデータを抽出する。

各データはデータの取り扱いを容易にするため、後の分析に必要な単位に集約し、ファイルに出力する。すなわち、品質に関してはプロジェクト工程別に、工数に関してはプロジェクト、工程および要員熟練度別に集計する。

プログラム規模はライブラリ管理システム<sup>(4)(5)</sup>からデータを抽出しプロジェクト別にプログラム規模ファイルに出力する。

次に演算処理を容易にするためこれらのデータをプロジェクト毎に編集し、プロジェクト別原データファイルに出力する。それと同時にプロジェクト別のデータ一覧も出力する。

出力するデータ項目はプロジェクト名、プログラム規模、設計工数、基本から詳細設計までの工程別工数、要員クラス別工数、工程別不良件数、等である。

## (2) データ演算機能

次に収集したデータの統計処理を行うため異常値データの排除を行う。

図10に示す如く、プロジェクト別原データファイルからデータを取り出し、不良検出率、設計効率率、生産効率率の演算処理をプロジェクト別に行い、プロジェクト別演算結果ファイルに出力する。

その演算を一覧表およびグラフに出力し、このグラフをみてデータの取捨選択を行い、再度演算処

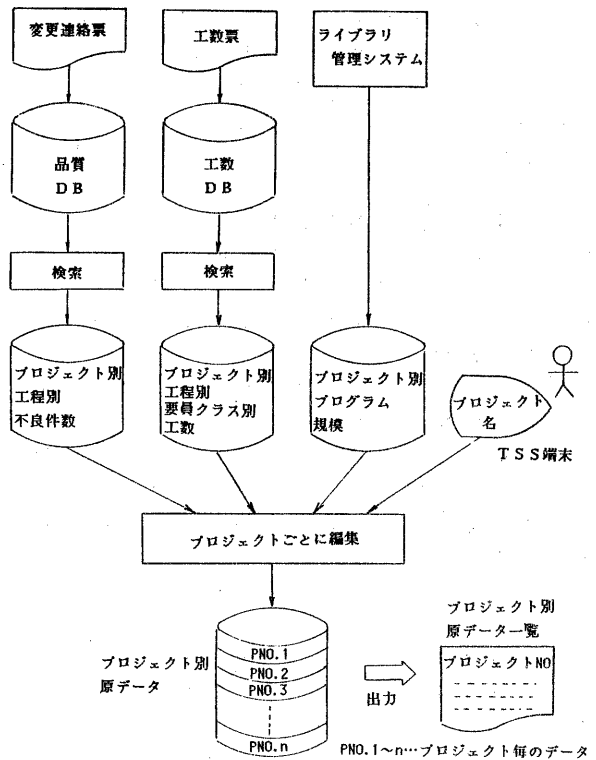


図9 データ収集機能

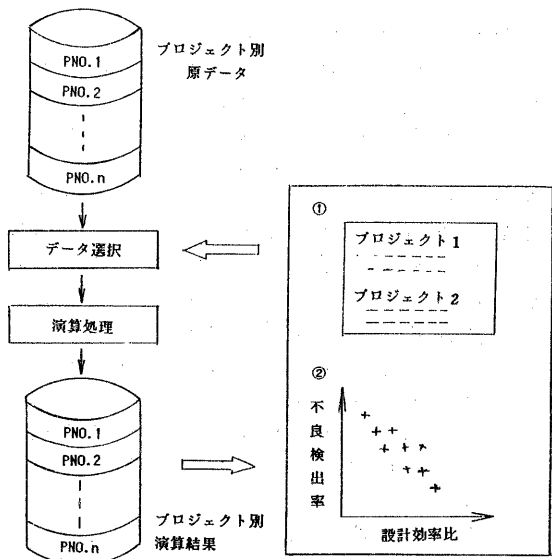


図10 データ演算機能

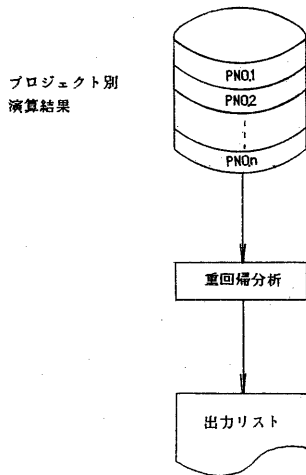


図11 データ分析機能

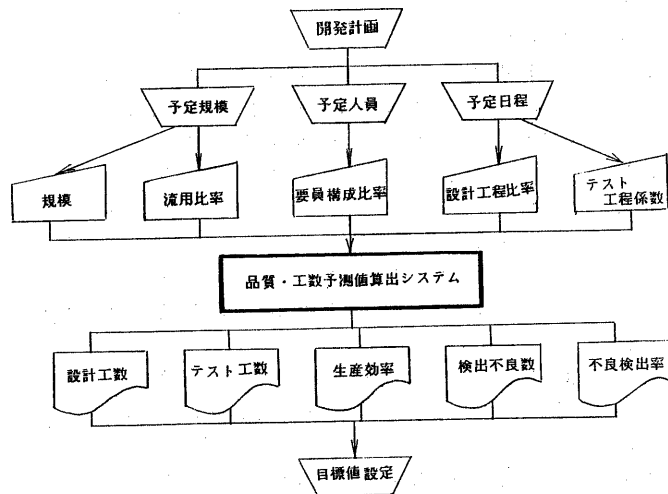


図12 予測値算出システム

理をする。このループを何度も繰り返し必要なデータのみをプロジェクト別演算結果ファイルに蓄積する。この操作はTSS端末を使用し会話形で行う。

(3) データ分析機能

プロジェクト別演算結果ファイルからデータを取り出し、重回帰分析プログラムにより分析を行い表3に示すモデル式の係数および指数 $a1 \sim a3 \sim e1 \sim e3$ を計算する。この結果をLPに出力する。以上によりモデル式が求まる。

表3 品質および原価モデル

$\begin{aligned} \text{設計効率比} &= a1 (\text{プログラム構成係数})^{b1} \cdot (\text{設計工程係数})^{c1} \cdot (\text{設計熟練係数})^{d1} \cdot P^{e1} \\ \text{生産効率比} &= a2 (\text{プログラム構成係数})^{b2} \cdot (\text{設計工程係数})^{c2} \cdot (\text{設計熟練係数})^{d2} \cdot P^{e2} \\ \text{不良検出率} &= a3 (\text{プログラム構成係数})^{b3} \cdot (\text{設計工程係数})^{c3} \cdot (\text{設計熟練係数})^{d3} \cdot P^{e3} \end{aligned}$ <p>但し <math>a1 \sim a3, b1 \sim b3, c1 \sim c3, d1 \sim d3, e1 \sim e3</math>: 実数  <math>P</math>: プログラム固有係数</p>
--

(4) 予測値の算出

上述で求めた品質および生産性のモデル式を使用し、会話形で予測値を求める。図12に示す如く、プログラム規模、要員構成比率、設計工程比率、テスト工程係数等を経験および開発計画に基づいて投入すると、設計工数、テスト工数、生産効率、検出不良数、不良検出率がただちに求まる。

なお、プログラム固有係数は実行上一定として扱っている。

7. むすび

情報通信システムのソフトウェアについて、その品質および生産性とそれに寄与する要因との関係を量的に明らかにし、これに基づいた品質および生産性のモデル式を導くとともに、このモデル式を容易に導き出すためのプログラムを開発した。

この結果、

(1) モデル式により、

(a) 設計開始時点で、これらの要因を設定して、品質および生産性の予測を行いこれに基づいた生産計画の立案が可能となった。

(b) 中間工程においても同様の予測を行い、生産計画の修正が可能となった。

(2) モデル式を導き出すためのプログラム開発によって、

(a) 従来人手によったためモデルの決定に多大の時間を要していたが短時間で決定できる様になった。

(b) より多くの変数を用いた重回帰分析が可能となった。

以上により、ソフトウェアの定量的管理に大きく寄与し現在これを実用に供している。

今後の課題としては、品質と生産性に寄与する既述プログラム固有係数、即ち技術的要因の明確化によるモデル式の汎用化があり、さらに、検討を行なっていく予定である。

終わりに、御指導御協力を頂いた各位に厚く御礼申し上げます。

#### 参考文献

- 1) 渡辺、他：“ソフトウェアの品質および生産性予測法の一事例について”  
日科技連 第2回ソフトウェアにおける品質管理シンポジウム(1982-09)
- 2) 丸、他：“会話型によるソフトウェア原価、品質モデルの生成手法”  
情処学会、第28回(S59前期)全国大会、3J-3、
- 3) 小林：“相関・回帰分析法入門”日科技連出版社(1982)
- 4) 峯尾、他：“交換ソフトウェアにおける維持管理システム”  
信学会交換研究会 SE81-110
- 5) 大和、他：“通信システム用ライブラリ管理の一手法”  
日科技連 第3回ソフトウェアにおける品質管理シンポジウム(1983-09)