

自動販売機制御ソフトウェアの開発における OMT 法の有効性の検証

柴山 具久

南山大学 大学院 経営学研究科
shiba@iq.nanzan-u.ac.jp

野呂 昌満

南山大学 情報管理学科
masami@iq.nanzan-u.ac.jp

本研究では、OMT 法と構造化設計法を比較する開発実験を行ない、両者の優劣を定量的に比較して、開発者の立場に立った OMT 法の有効性を検証する。OMT 法は、多重視点からシステムを分析するので、従来のシステム開発法に比べて、生産性や保守性の向上に有効な手段であるといわれている。しかし、この有効性を定量的に検証した報告は少ない。本研究では、系統的に実験計画を立てて、その計画に基づきソフトウェアの開発実験を行ない、開発方法の優劣を比較した。開発に必要とした時間を統計的な方法を使って解析することにより、OMT 法の有効性を定量的に検証した。

Is OMT useful to development of vending machine control software? — From measurement view —

Tomohisa Shibayama

Graduate School of Business
Administration,
Nanzan University

Masami Noro

Department of Information Systems
and Quantitative Sciences,
Nanzan University

This paper concerns an object validation to know usefulness of OMT for vending machine control software over traditional structured methods. The goal of the study is to know, quantitatively, if OMT is better method than the structured one for this type of software development from the developer's perspective. A set of experimental development was taken under the pre-designed experimentation scenario for gathering data to check the validity. The paper describes the design of the experimentation and the way of data interpretation.

1 はじめに

本研究では、OMT(Object Modeling Technique)法 [1]と構造化設計法 [2]を比較する開発実験を行ない、開発者の立場に立ったOMT法の有効性を定量的に検証する。

OMT法は、多重視点からシステムを分析するので、従来のシステム開発法に比べて、生産性や保守性の向上に有効な手段であることといわれている。静的、動的、機能的の3つの視点からシステムを分析することで、複雑なシステムを対象とした場合でも、詳細な部分までの分析を可能にする。また、分析結果として、各視点からの対象システムのモデルを図式化して、豊富なドキュメントを作成する。システムについて多くの種類のドキュメントが用意されていれば、再開発のさい、開発者がシステムについて理解することを助ける。このようなOMT法の特徴があるにもかかわらず、これまでに、それらの特徴が生産性や保守性の向上につながることを定量的に検証した報告は少ない。

これまでは、オブジェクト指向の考えを取り入れれば、システムの品質、または再開発における再利用性が従来のシステムに比べて向上することは報告されている [3]。しかし、それらの報告では、オブジェクト指向システム開発技術だけを対象として開発方法の有効性を評価しており、他の開発方法との優劣は分からない。このように、他の手法と比較せずに出した開発方法についての評価結果は、単一視点からのものであり、主観的なものであるといえる。

本研究では、OMT法の有効性を客観的に評価するために、

- 系統的な実験計画を立てて、
- 統計的な方法を使って実験データを解析した。

系統的に実験を計画するために、田口玄一氏の直交表 [4]を使う。そうして作成した実験計画にもとづき、自動販売機シミュレータを使って、ソフトウェアの再開発実験を行なう。最後に、開発実験によって得られたデータを統計的に解析して、OMT法の有効性を検証する。

2 実験計画

自動販売機シミュレータの再開発実験のために、4つの因子を取り上げて、それらの因子の水準組合せについての実験の計画を立てた。そのさい、系統的に実験を計画するために、田口玄一氏の直交表 [4]を使った。田口の直交表を使えば、因子の数が多い場合でも、系統的に実験回数を減らして、因子の水準組合せ全部について、公平な実験結果を出すことができる。

以下では、はじめに、実験で取り上げる因子、およびそれらの因子の水準の選び方について述べる。次に、取り上

げた因子の水準組合せについて述べる。

2.1 因子の決定

本実験では、実験回数を減らすために、ブロック因子二つを含む合計4つの因子を取り上げた。我々は、ソフトウェアの開発実験において実験結果に影響を与える要因は、

- 開発方法
- 要求
- 学習効果の個人差
- 潜在能力の個人差

の四つと考えた。

開発方法 本研究では、OMT法の有効性を他の開発方法と比べることにより検証する。これにより、開発のさいに使う開発方法によって実験結果は変わる。

要求 開発実験では、要求を満たすために、既存のシステムの再開発を行なう。要求には、再開発することが困難なものもあれば容易なものもある。これにより、要求により実験結果は変化し、さらに開発方法の優劣が変わる場合もあると考えることができる。

学習効果の個人差 本研究では、一人の開発者が繰り返し実験をする。一人の開発者が複数回の実験を行なう場合では、開発者の開発について技術力は、開発を繰り返すたびに、向上すると考えることができる。言い替えれば、開発者によってははじめての再開発よりも、それ以降の開発の方がしやすくなる場合がある。このように、一人の開発者が実験を繰り返す場合では、実験に学習効果の個人差が生じて、それにより実験結果は影響を受けるといえる。

潜在能力の個人差 本研究では、4人の開発者により実験を行なう。開発者によっては、ソフトウェアの再開発について、すでにある程度の知識を持っていたり、習熟度が優れている場合がある。このことから、開発者が複数いる場合では、個人の潜在能力が実験結果に影響を与えると考えることができる。

正しい実験の結論を得るためには、これらの要因をすべて実験の因子として取り上げて実験を計画することが理想的である。前述した要因をすべて因子として取り上げて、開発方法と要求を複数の最小である2水準に選び、4人の開発者を用意したことから、二つの個人差それぞれを4水準に選んだ実験計画を考える。この場合、すべての因子の水準組合せ全部についての開発実験をしようとする

と、各因子の水準数からその組合せは、

$$2 \times 2 \times 4 \times 4 = 64$$

(開発方法) (要求) (個人差) (個人差)

通りとなる。この結果、各組合せを1回だけ実験したとしても、合計64回の実験が必要となる。

64回の実験を行なおうとした場合、以下に挙げる問題から、正しい実験結果を出すことができないと考えることができる。

実験期間 本研究では実験期間が一年間である。富士電機のこれまでの経験から、一回の開発実験は最低でも2週間かかることが分かっている。これにより、1年間で64回の実験を終えることは無理といえる。

均一な実験の場 実験回数が多い場合、すべての実験を均等な条件ですることは難しい。

実験回数を減らすために、ブロック因子を取り上げて、実験のやり方を工夫して実験を計画することを考えた。個人の学習効果と個人の潜在能力について、以下のように考えることで、各水準組合せの実験がそれら二つの要因の影響を公平に受けるようにした。

個人の潜在能力 本実験では、4人の開発者により実験を行なう。前述したように、実験において複数の開発者がいる場合、各実験に個人差が生じる。これにより、実験結果は影響を受け、各水準組合せは公平な実験条件での結果といえなくなる。

開発者が複数いる場合でも公平な実験結果を出すために、開発者をブロック因子として取り上げて、水準の組合せすべてが開発者の違いによる影響を公平に受けるようにした。開発者をブロック因子として実験に導入した場合、各水準の組合せは、すべての開発者により実験されることになる。これにより、各水準組合せは、開発者の違いによる影響を公平に受ける。

開発者をブロック因子として取り上げた場合の他の利点として、開発者の違いによる実験結果への影響をデータ解析のさいに別に取り出すことが挙げられる。開発者の効果を、他の効果、または実験誤差に紛れ込ませず、別にとり出せることができれば、実験誤差を小さくすることができる。これにより、因子の水準の比較の精度をあげることができる。

個人の学習効果 本実験では、一人の開発者は、複数回の実験を行なう。この場合、前述したように、実験の繰り返しによる個人の学習効果の違いから、実験結果は、影響を受けて、公平な実験条件での結果といえなくなる。

実験の繰り返しがある場合でも正しい実験結果を出すために、開発順序をブロック因子として取り上げ

て、開発順序の違いによる影響をすべての水準組合せが公平に受けるようにした。開発順序についても、開発者をブロック因子として取り上げることと同様に、その違いによる実験結果への影響をデータ解析のさいに別に取り出すことができる。

以上から、開発者と開発順序をブロック因子として取り上げることにより、個人の学習効果と個人の潜在能力を因子として取り上げる必要がなくなった。なぜなら、データ解析のさいに要因の実験結果への影響を各実験結果から公平に取り出すことができれば、その要因については他の要因のすべての水準との組合せを考える必要はないからである。

以上から、本実験で取り上げる因子をまとめると、表1のようになる。

表 1: 因子とその種類および水準

因子の名前	因子の種類	水準
開発方法	制御因子	OMT法 + C++ SD ¹ + C
要求	表示因子	容易な要求 (2種類) 困難な要求 (2種類)
開発者	ブロック因子	開発者 1 ~ 4
開発順序	ブロック因子	1回目 ~ 4回目

- あとで最適な水準を選ぶために、制御因子として開発方法 (2水準) を取り上げた。
- 再開発の種類を複数にするために、標示因子として要求 (4水準) を取り上げた。
- 実験条件の違いによる影響を別に取り出すために、ブロック因子として開発者 (4水準) と開発順序 (4水準) を取り上げた。

要求の水準については、以下のような問題点に対処するために、4水準に選んだ。本実験では、ブロック因子を二つ取り上げて実験を計画した。一つのブロック因子である開発者は、4人の開発者を用意したことから、4水準となる。これに伴い、実験では二つのブロック因子を取り上げるので、必然的に、一人の開発者は4回の実験を行なうことになる。一人の開発者が4回の再開発を行なう実験において要求を複数の最小である二つだけ用意した場合、一人の開発者は同じ要求のための再開発を繰り返して行なうことになる。一人の開発者が同じ要求のための再開発を行なった場合、以下に示す理由から、正確な実験結果を出すことができないと考えることができる。

- 2回目の再開発では、要求に対する知識が豊富になる。
- C と C++ の記述法は類似している。

¹構造化設計法:Structured Design の略

この問題に対処するために、要求の数を開発者の実験回数と同数にして、一人の開発者が同じ要求のための開発を行わないようにした。前述したように、一人の開発者が4回の再開発を行なう本実験では、要求を四つ用意すれば一人の開発者は毎回違う要求のための再開発を行なうことになる。4回の再開発すべてが違う要求のための再開発であれば、前述したような実験結果への影響は存在しない。以上から、要求の水準を4水準に選んだ。

2.2 実験計画

決定した因子の水準組合せの実験については、田口玄一氏の直交表を使って、系統的に計画した。前述したように、時間および実験の場の条件から、取り上げた因子すべての水準の組合せ全部についての公平な実験をすることは無理である。因子の数が多い場合でも、少ない実験回数で公平な実験結果を得るために、田口玄一氏の直交表を使って実験を計画した。

直交表を使って実験を計画した結果、本研究の実験計画表は、表2のようになった。

表 2: 実験計画表

	1 回目	2 回目	3 回目	4 回目
開発者 1	A1B1	A2B3	A1B4	A2B2
開発者 2	A1B2	A2B4	A1B3	A2B1
開発者 3	A2B4	A1B2	A2B1	A1B3
開発者 4	A2B3	A1B1	A2B2	A1B4

ただし、A1, A2 : 開発方法 (2水準);

B1, B2, B3, B4 : 要求 (4水準)

3 開発実験

開発実験では、実験計画にもとづき、要求を満たすためのソフトウェアの再開発を行なう。再開発するソフトウェアには、商品販売について必要最小限の機能を持つ自動販売機シミュレータを使う。この自動販売機シミュレータに新しい機能を付け加えるための再開発を行なう。

自動販売機シミュレータの再開発のために、容易な要求を二つと困難な要求を二つの合計四つの要求を用意した。開発方法には、複雑なシステムを対象とした開発に適しているものもあれば、その逆に、単純なシステムを対象とした開発に適しているものもある。このことを考慮して、4つの要求を無作為に選ばず、難易度の違う要求を均等に選んだ。難易度の違う要求を均等に選んで再開発実験をすることにより、要求の難易度について偏りのない、公平な実験結果を出すことができる。

用意した四つの要求を以下に示す。

- 容易な要求として、

- 自動返金タイマー制御
- 販売時間帯機能
- 困難な要求として、
- 割引機能
- 交互販売機能

我々は、富士電機の開発者のこれまでの経験から見積もった難易度をもとに、これらの要求を選んだ。

OMT法と構造化設計法を比較するために、同一の要求仕様書からそれぞれの開発方法にもとづき作成した二つの自動販売機シミュレータを用意した。

表 3: ドキュメント

OMT 法	構造化設計法
要求仕様書	
制約の指針 制約条件 要求仕様書	
分析・設計	機能別ソフト設計仕様書
データ辞書	疑似タスク関連図
オブジェクトモデル	機能別関数仕様書
動的モデル	共通フラグ一覧
機能モデル	共通アーター一覧
クラスの仕様	
メソッドのアルゴリズム	
以上 OMT 法	以上 富士電機
シミュレータのコード	
C++	C
その他	
OMT 法	C の教科書
C++ の教科書	昨年度の卒論 (一部)
昨年度の卒論 (全部)	
・要求仕様 (個人) 実験計画にもとづき、担当する要求の要求仕様。 ・自動販売機についての資料 (全員共通) 用語辞書 サービスマニュアル	

OMT法にもとづき分析および設計をして、C++ を使って実現したシミュレータ このシミュレータは、94年度の卒業研究 [5] において製作したものであり、オブジェクト指向の考えを取り入れたソフトウェアである。このシミュレータに関するドキュメントは、OMT法が示すドキュメント一式が用意されている (表3参照)。

構造化設計法にもとづき設計して、C を使って実現したシミュレータ 従来までの開発方法で開発されたソフトウェアとして、富士電機の開発方法と同じ方法で製作し

たシミュレータを用意した。このシミュレータは、富士電機が従来から使っている開発方法と同様の方法で製作されており、構造化設計法にもとづき設計され、Cを使って実現されている。このシミュレータに関するドキュメントは、富士電機がソフトウェアを開発するさいに作成するドキュメントのなかから、必要最小限と思われるものを抽出して用意した(表3参照)。

4 実験データ

開発実験のさいには、各工程において、開発に使った時間を、[方法理解],[要求理解],[自動販売機理解],[開発],[その他]の項目に分類して記録する。開発に使った時間を分類して記録することにより、開発方法についての時間を得ることができる。また、項目別に開発使用時間を解析することで、開発方法を多数の視点から比較することができる。

4.1 データの収集

本実験では、開発使用時間記録表(表4参照)を使って、開発に使った時間を項目別に分類して記録する。開発に使った時間には、開発方法を理解するために使った時間、要求を理解するために使った時間など、多くの種類の時間が含まれていると考えることができる。開発に使った時間の中から開発方法についての時間を得るためには、それらを分けて記録する必要がある。本実験では、表4のように項目を設定して、開発に使った時間を項目別に分類して記録することにより、開発方法についての時間を得るようにした。

表 4: 開発使用時間記録表

開発工程 作業内容	分析	設計	実現	その他
方法(開発法)理解				
自動販売機理解				
要求理解				
開発				
その他				

方法理解 開発方法(開発法)について理解するために使った時間を記録する。方法について理解する時間は、開発経験の少ない開発者には必要となるが、経験の豊富な開発者ならば、不必要と考えることができる。これにより、一般的²な開発における開発方法の優劣を比較しようとする本研究では、方法理解の時間を本

²本稿で使う「一般的」とは、経験の豊富な開発者が開発を行なっているようなことをいう。

来の再開発にはない学習時間として別に考える必要がある。

要求理解 要求理解には、要求について理解した時間を記録する。要求を理解するさいには、要求仕様書だけが必要であり、開発方法は関係ない。これにより、開発方法を比較する場合には、要求理解の項目の時間を比較する時間から除外する必要がある。しかし、各開発方法を使った開発を比較をする場合には、要求を理解することも開発工程の一つであるので、比較する時間に含める必要がある。

自動販売機理解 自動販売機理解には、それぞれの開発方法で用意されるドキュメントを読むために使った時間を記録する。言い替えれば、この時間は、既存システムの構成要素(データ、関数など)、またその関係について理解する時間である。これにより、自動販売機理解に使った時間は、一般的な開発すべてに必要な時間といえる。

開発 開発には、要求を満たした自動販売機を分析、設計、実現するために使った時間をそれぞれ記録する。開発に使った時間も、自動販売機理解に使った時間と同様に、一般的な開発すべてに必要な時間といえる。

4.2 比較方法

本研究では、開発使用時間記録表の項目別に開発に使った時間を解析することにより、多重視点から OMT 法と構造化設計法の優劣を比較する。以下では、解析項目と項目別に記録した開発使用時間の関係について説明する。

総合

開発のさいの状況を一切考えず、単純に、開発使用時間記録表に記録された時間の総和を使って二つの開発方法を比較する。本研究では、

- 田口の直交表を使って、系統的に実験計画を立てた、
- 要求の難易度の違いを係数を使って標準化する(詳細後述)、

の二つの点から、各水準組合せは、均等な条件のもとで実験されたと考えることができる。これにより、我々は、開発に必要なすべての時間として、開発に使った時間の総和を公平に比較できると考えた。

各項目

開発時間記録表の項目(行)それぞれについて、二つの開発方法を比較する。我々は、前述したように、開発に使う時間から開発方法に関係する時間だけを取り出すために、項目を設定した。開発者は、その項目にもとづき開発に使った時間を記録する。このことを利用して、開発使用時間を項目別に解析することにより、以下に示す視点か

ら二つの開発方法を比較できると考えた。

- 方法理解 … 開発方法は理解しやすいか。
- 自動販売機理解 … ドキュメントは理解しやすいか。
- 開発 … 開発方法は使いやすいか。

これらの項目と視点の関係は、項目に記録する時間に対応する開発の内容から簡単に考えることができる。

自動販売機理解と要求理解と開発

開発についての学習時間を除いて、一般的な開発に必要なとする時間を使って、二つの開発方法を比較する。開発に使った時間は、大きく、学習時間と実質開発時間に分けることができる。実質開発時間とは、一般の開発で必要とする時間のことで、既存のドキュメントからシステムについて理解して、要求を満たしたシステムを製作するために使った時間を表す。一般の開発においては、開発者はすでに開発方法を知っており、学習時間は必要ない。一般的な開発者の視点から二つの開発方法を比較するために、実質開発時間だけを使って、二つの開発方法を比較する。

実質開発時間によって二つの開発方法を比較するために、自動販売機理解、要求理解、開発の三つの項目を以下のように使う。

- 開発方法について比較 … 自動販売機理解 + 開発
- 開発方法を使った開発について比較 … 自動販売機理解 + 要求理解 + 開発

我々は、前述したように、要求を理解する時間は、開発方法には関係のない時間ではあるが、ソフトウェアを開発するさいの一工程の時間ではあると考えた。この考えから、以上のような三つの項目の使い方を考えた。

4回の開発実験を組分けして

4回の実験を組分けして、はじめてソフトウェアを開発する場合と一般的なソフトウェア開発の場合の関係から、二つの開発方法を比較する。開発方法についての学習、および開発に対する「慣れ」などを考えた場合、はじめてソフトウェアを開発するさいには多くの時間を費やすと考えることができる。しかし、その後の開発では、開発方法によっては、はじめの開発の経験が活かされて、開発方法を理解する時間を必要とせずにソフトウェアを開発できる場合がある。我々は、この点について開発方法を比較するために、四回の実験を以下のように組分けして、それらの組の関係から二つの開発方法を比較することを考えた。

- はじめの二回と後の二回 4回の実験をはじめての二回と後の二回の二つの組に分け、それらの組の間の開発使用時間の大小関係から、二つの開発方法を比較する。はじめの組の開発の方が後の開発より多く

時間を使った場合、その開発方法では、はじめての開発では手間が掛かるが、そののち、開発を繰り返すたびに、必要とする時間が減ると考えることができる。一方、二つの組の間に差がなく、使用した時間が減っていない場合、その開発方法を使った開発では、常に一定の手間が掛かるといえる。

- はじめ(後)の二回の開発 はじめ(後)の二回の開発だけで、二つの開発方法を比較する。はじめの二回の開発の比較では、はじめてソフトウェアを開発するさいに各開発方法を使った場合の優劣を比較できる。一方、後二回の開発の比較では、一般的なソフトウェア開発における各開発方法の優劣を比較できる。
- 一回目と残り三回 開発に対する「慣れ」を除外するために、二回目以降の開発で使った時間を使って、二つの開発方法を比較する。本実験で用意した開発者は、全員、これまでにソフトウェアの再開発を経験したことがない者である。これにより、第一回目の開発実験の結果は、ソフトウェアの開発に対する「慣れ」の影響を大きく受けていると考えることができる。このソフトウェア開発に対する「慣れ」の影響を除外するために、第一回目の開発を除いて、二回目から四回目の開発使用時間を使って開発方法を比較する。

5 データ解析

データの解析は、開発方法についての真値を得るために、実験データを補正して行なう。はじめに、実験のやり方などをもとにして、実験データの構造模型を設定した。そののち、構造模型をもとに、実験結果に大きな影響を与えている要求の難易度について実験データを補正した。実験データの補正後の開発方法の比較について、本稿では、分散分析について説明する。

5.1 データの構造模型

本実験では、開発方法、要求、開発者、開発順番の四つを因子として取り上げて実験を行なったことから、実験データの構造模型を、

$$x_{ijkl} = \mu + \alpha_i + \beta_j + (\alpha\beta)_{ij} + \gamma_k + \varepsilon_l + e_{ijkl} \quad (1)$$

ただし、 μ : 一般平均;

α_i : 各開発方法(A_i 水準)の効果;

β_j : 要求の難易度(B_j 水準)の効果;

$(\alpha\beta)_{ij}$: A と B の交互作用;

γ_k : k 行の効果; ε : l 列の効果;

e_{ijkl} : データ x_{ijkl} に入っている実験誤差

のように設定した。

我々は、この実験データの構造模型をもとに、データの補正を要求について行ない、分散分析を行なった。

5.2 要求の難易度の標準化

要求の難易度に対して各実験結果を公平なものにするために、要求の難易度についての標準化を行ない、実験データを補正する。開発実験では、4種類の要求のための再開発を行なった。2.1節で述べたように、この4種類の要求は、困難と容易の2種類に分けることができる。この難易度の差をなくして、すべての実験結果(開発使用時間)が同じ難易度の要求のための再開発実験の結果となるように、要求の難易度についての標準化を行なう。

要求の難易度についての標準化を行なうために、各要求の難易度に対して係数を算出して、難易度を数値化することを考えた。困難と容易の二つの難易度の基準は、富士電機の開発者が、これまでの経験から、簡単に見積もったものである。標準化をするさいには、それらの難易度を数値で表す必要がある。本研究では、富士電機の開発者に、各要求について、必要開発工数、変更するプログラム行数などのアンケートに答えてもらい、その結果から係数を算出した。

要求の難易度について係数を設定したのち、開発に使ったそれぞれの時間を係数で割ることにより、要求の難易度についての標準化を行った。

5.3 分散分析

比較項目を比較する方法の一つとして、本稿では、本研究の中で比較方法がもっとも複雑な分散分析について説明する。本実験では、実験データのばらつきを与える原因として、データの構造模型から、

- 開発方法として A_1A_2 の2個の水準で実験がなされている、
- 開発者が複数いる、
- 実験の順序が違っている、
- 実験誤差がある、

の4つを考慮することができる。要求の種類については、その難易度についての標準化をしている(5.2節参照)ので、実験データのばらつきの原因にならないと考えた。

4つの原因別に全体のばらつきを分解すると、式2を得る。

$$S_T = S_R + S_C + S_A + S_e \quad (2)$$

ただし、 S_T : 総平方和; S_R : 行間平方和;
 S_C : 列間平方和; S_A : A 間平方和;
 S_e : 誤差平方和

この分散分析の式を使って、仮説の検定を行ない、仮説が棄却されたならば、各水準の効果を推定して、最適水準を決定する。

比較項目について二つの開発方法を比較したのち、その結果をもとに、OMT法が従来の開発技術と比べて生産性・保守性の向上に有効かどうかを考察する。

6 おわりに

本稿では、系統的に立てた実験計画にもとづいた自動販売機シミュレータの再開発実験、およびその実験結果を使ったOMT法と構造化設計法の優劣の比較について述べた。正しい実験結果を出すためには、取り上げる因子の数が多く場合でも、すべての因子の水準組合せ全部が公平に実験されるように実験を計画する必要がある。本稿では、すべての水準組合せを公平に実験して正しい実験結果を出すために、ブロック因子を導入し、田口の直交表を使って実験を計画したことについて述べた。また、二つの開発方法の比較においては、項目別に記録した開発使用時間から得られる結果を設定して、その結果の算出方法について述べた。

我々は、本研究が、企業が開発技術を従来の開発技術(構造化開発法)から、新しく、オブジェクト指向システム開発技術に移行するための判断基準になると考える。本研究では、OMT法と構造化設計法の優劣を定量的に比較した。比較では、企業におけるソフトウェアの再開発を仮定してシミュレータを使った再開発実験を行ない、その実験から得たデータを解析した。このことから、本研究の解析結果は、実際の開発で実験した場合の結果と同等といえ、企業が開発方法を移行するさいの判断基準になると考える。

現在、開発実験が終了しており、今後は、我々が設定した比較項目について、実験結果の開発使用時間を解析する必要がある。解析ののち、その解析結果をもとに、OMT法と構造化設計法のそれぞれについて、利点、欠点を考察して、二つの開発方法の優劣を比較する。

謝辞

本研究を進めるにあたり、様々な形で御協力いただきました富士電機(株)電子制御部の山本斎さん、横田幸雄さん、繁田雅信さん、杉野一彦さんをはじめとする皆様に心から感謝いたします。

参考文献

- [1] J.Rumbough, M.Blaha, W.Premarlani, F.Eddy, W.Lorenson : *Object Oriented Modeling and De-*

sign , Prentice Hall International,1991.

- [2] E. Yourdon,L.L.Constantine,,:*Structured Design*, Englewood Cliffs,Yourdon Press,1979.
- [3] W. L. Melo, L. Briand, V. Basili, : “Measuring the Impact of Reuse on Quality and Productivity in Object-Oriented Systems”,Technical Report University of Maryland,1995.
- [4] 田口 玄一, 横山 巽子「実験計画法」,(財)日本規格協会, 1979
- [5] 後藤 崇夫, 前田 剛, “オブジェクト指向による自動販売機制御ソフトウェアの設計と試作”, 南山大学 経営学部 情報管理学科 卒業論文, 1995