

ソフトウェアアクリエーション：交換接続制御プログラムの自動設計用の専用エキスパートシステム

化 吉龍, 河野 善彌, 陳 慧, ヘルーズ. H. ファー

埼玉大学 工学部 情報システム工学科

〒338 浦和市下大久保255

電話：048-858-3487, FAX:048-858-3716, koono@cit.ics.saitama-u.ac.jp

あらまし この論文は状態構造を持つソフトウェアの1例について、人の設計の知識を取り出して再利用する自動設計システムの報告である。原理研究と実際的なシステムの試作の両面から報告している。これは人の設計知識をある設計工程の入出力変換と捕えて、その設計知識をエキスパートシステム化して自動設計を行なわせる方式をとる。これはハードウェアでのDesign Automationと同じ技術であり、ソフトウェアでも視点を変える事により可能になる事を指摘し、この新しい考えで各種の自動設計システムが可能になる事を論じている。

Software Creation: An Automatic Software Design System By A Dedicated Expert System For A Switching Control Program

Zenya KOONO, Hui CHEN and Behrouz H. FAR

Department of Information and Computer Sciences,

Faculty of Engineering, Saitama University

255 Shimo-okubo, Urawa, 338, Japan

Phone: +81-48-858-3487, FAX: +81-48-858-3716, koono@cit.ics.saitama-u.ac.jp

Abstract This paper report on an automatic design system for a software, which reuses human design knowledge. It reports on both basic research and prototype model for actual system. It acquires design knowledge as an input and output transformation rule of a design process as the design knowledge, and an expert system designed by the knowledge designs software automatically. It also discusses that the essential technology is the same as that of Design Automation Systems in hardware, and points out that various automatic design may be realized and used from this new view point.

1. はじめに

この論文は、状態構造を持つソフトウェアの専用的な自動設計システムの研究報告である。

ソフトウェアの自動設計はソフトウェア関係者の夢で、これまでに多くの研究がある。人の設計があまり明解でなく「自動設計とは如何にすれば良いのか」の共通思想が無い事が問題である。本報告は人に倣う自動設計[河野93b]に拠っている。

人は多くの知識に支えられている。人の知識をそのまま自動設計等の知識システムに移行させようとすると、最後には莫大な量の常識問題に遭遇してしまう。この方式は必要な知識に限定して実用化するもので、知識量を意図的に限定する考えに立つ。

2. 交換システムの自動設計

2. 1 交換システムの概要

図1は研究対象とした原理的な電話交換システムの構成と状態の遷移を示す。左端の電話器から発信すると局側で検知され、ダイヤル信号受信器が接続されダイヤル音が出される。電話番号を送出すると、相手の空きを確認し呼出信号(音)を着信側(発信側)の電話機に送出する。着信側の応答により両者間が接続され、通話が行われる。通話を終えると元の空き状態に戻る。

この交換の機能は強いシーケンス性を持つから有限状態機械(Finite State Machine, FSM)でモデル化でき、図1上の接続状態図の遷移で振舞を表わせる。この状態遷移のルートがプログラムに対応する。接続制御プログラムはスイッチ網を制御し、運用プログラムは使用者のサービスクラスや電話晩号等をデータベースで制御する。

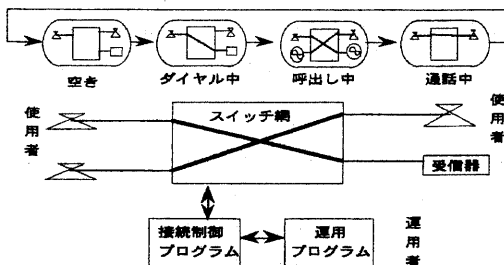


図1 原理的な電話交換システムと状態遷移

交換システムの仕様表現の国際規格SDL (Specification and Description Language)は早くから制定された。初期は状態遷移図のみ、現在ではテキスト形式(SDL-PR)と図式表示(SDL-GR)の両表現形式を持つ。SDLでのシステムは、幾つかの拡張有限状態機械(Extended Finite State Machine, EFSM)が相互にメッセージ通信する事により実現される。システム、ブロックの階層構成があり各EFSMは状態遷移を持つ。状態遷移図はプロセス図と呼ばれ、状態図とフローチャートを組合わせた形である。本研究はこの図を作業用に使用する。SDL以後には、その前位のMessage Sequence Chart (MSC)が規格化された。

交換システムは1965年以来実用化された高信頼度、高速応答特性の実時間システムである。現在ではMline規模に達し、人の設計では競争環境に耐えられないのでサービスソフトウェア自動生成が実用に入っている。これはMSC,SDLと略人の設計工程に倣い、サービス提供者の描いた仕様を段階的に人やエキスパートシステムが詳細化する方式である。

2. 2 原理研究の概要[Koono92, 93a, 河野93b]

筆者等は、状態構造のある場合の代表例として接続制御系を、また無い場合の代表例として運用系を取上げて、ボトムアップに原理研究を行ってきた。後者は汎用的な知的CASEツール[陳9797a, 97b]に進化し、前者は専用的なシステムに向かった。ここでは後者の原理研究の概要を記す。

図2に接続制御系の設計[河野96]の流れを示す。発信着信の両者を含む「呼」の状態遷移図が仕様であり、後続する「プロセス分割」[大森93, 95]はこれを発信と着信側のEFSMの状態遷移図に変換する。これは上位の概念レベルでは1→2プロセスの階層展開であるが、純自然言語概念以外の変換則なので階層的展開の連鎖を利用できないから、この段階全体の入力から出力への変換則を用いて自動設計させる。タスク挿入は資源を状態遷移させる機能(SDLではタスク)を状態差から求めて状態遷移ルートに挿入する。以後は各機能を階層的に展開し詳細化する。

正常なサービス手順以外(例 途中放棄)の遷移原因に対応する、または資源を割当られない等による各種の準正常機能[大森95]の設計がこの他にあるが省略した。

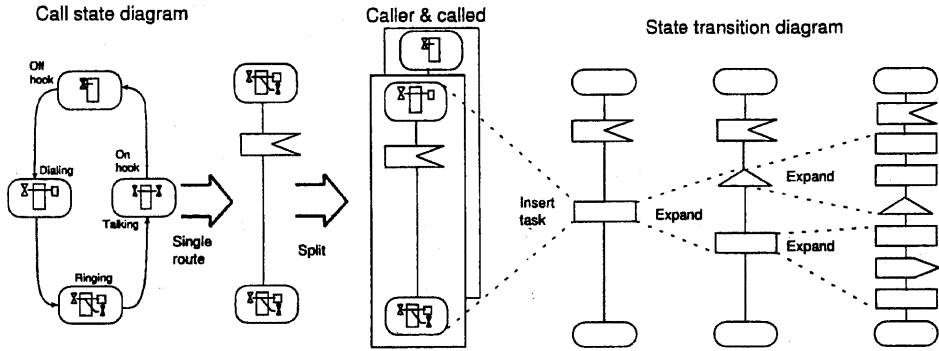


図2 設計の流れ

図3 [Koono93a]は、段階毎に詳細化が進む様子を
示し、横軸に設計階梯をとり縦軸は図面シン
ボル数を単位として詳細度を表した。図のように指
数的に増大して約15倍に増え、SDLからC
コードに変換すると更に約10倍に増え、全体で
約150倍程度、最終的にサービスあたり約1K
行程度のプログラムになる。図4は設計知識量を
設計ルールと呼ぶ小さな変換ルールを単位として
計数した時、知識量についての「習熟特性」を示
す。サービス数を増すにつれ設計ルール総計数が
初めに急激に立上がり、勾配は次第に低下するが
何時までも緩やかに増え、人に似た特性を示す。

3. 自動設計用エキスパートシステム

3.1 システム構成

これまでの原理研究に基づき、本格的なシステ
ム化の為にシステム構成を定め、接続制御系の中
心であるプロセス分割とタスク挿入を試作した。
図5 [Hua96]はその構成で、大別してプロセス分

割、タスク挿入、および階層展開による詳細化
[Chen96, 陳97c]から成る。ここではプロセス分割
とタスク挿入を報告する。この系ではSDL-G
R系の設計情報と(仕様を示す)接続状態図の図
形情報とが主流設計情報になる。

(1) プロセス分割[大森93, 95, Hua96]

これはサービスの仕様を表す単一-EFSMから2
ESMへの階層的展開で、人の設計に倣うアルゴ
リズム(図5の3段階の図式な変換)を採用した。

1. 呼びの接続状態図の接続経路を発信側と着
信側に分割する。
2. 仕様中のイベントの行先に応じた側の状態
遷移ルートに転記する。
3. 上の2でイベントが挿入されない側に
同期用のイベントを追加する。

(2) タスク挿入[大森93, 95, Hua96]

これは、旧状態から新状態へ各資源の状態を選
移させる機能(タスク)を作り、状態遷移ルート
に挿入する。概念レベルでは1:1対応で図のよ
うに資源毎に遷移前と後状態とで2次元の表より

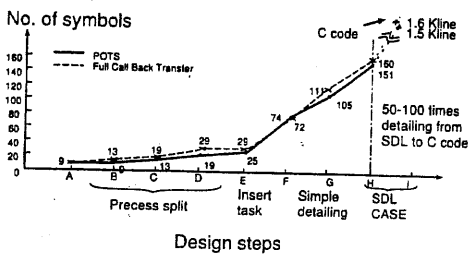


図3 詳細度の増大

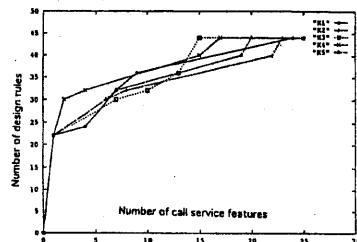


図4 設計ルールの習熟特性

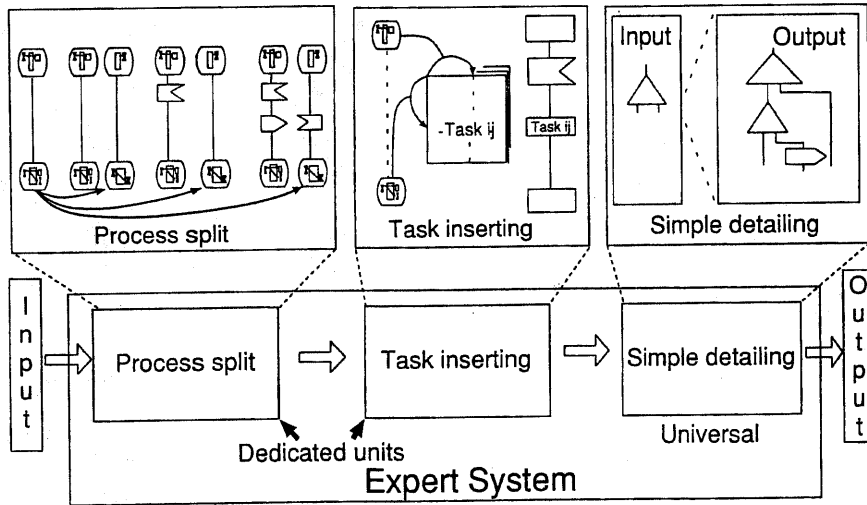


図5 システム構成

タスクを求めた後に、下記を行う。

- ・状態遷移ルートは分岐して複数の終着状態を持つ事がある。これを考慮し夫々のタスクを状態遷移ルート中のリンクに、重複と無駄を避けて挿入する事。

3. 2 システムのデータ構造[Hua96]

このシステムは以下の条件を必要とする。

- ・人とエキスパートシステムが設計する、
- ・SDL表記のプログラム情報と純図形情報である接続状態図情報の両情報を扱う。

前者は各種の自動設計と親和性を要し、後者は今後生じる各種の仕様表記や各種のハードウェアを利用する拡張等に耐える事が必要で、次の方針を定めた。

- ・プログラムレベルの情報は各種のプログラム表記情報と合わせて標準化する[Chen96].
- ・接続状態図情報はフレーム形式の知識表記構成とし、如何なる仕様表記やハードウェアの図式表記にも対処可能にする。

3. 2. 1 SDL 設計情報の表記

設計中のプログラムレベルの情報は図形表記SDL-GR表記に統一し、人の作業が終了した後にGR⇔PR変換をしてSDL-PRを用いる。(研究初期の方針から変え、設計者は適用分野技術の専門

技術者とシプログラム言語レベルは隠蔽する。)

エキスパートシステム内ではプログラ的と純図形的と兩種の設計情報を統一的にフレーム形式で扱う。まず、状態遷移図を含むフローチャート類、構造化チャート、データフロー図を含め統一的に処理するべく標準化[Chen96]を行ない、統一的なシンボルにした。図6のように、共通特性は図形の種別とシンボル間の接続情報、各シンボル内に表記する情報、および展開の親子関係となる。この標準図形をフレーム形式に対応させた。

3. 2. 2 接続状態図の表記

フレーム形式の知識を用い、図面を階層的に捕えて表記する。図7は、状態遷移図の表記の構成

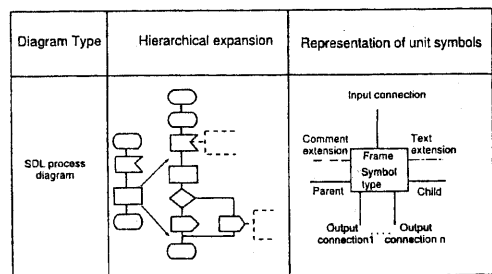


図6 シンボルの統一的な表記

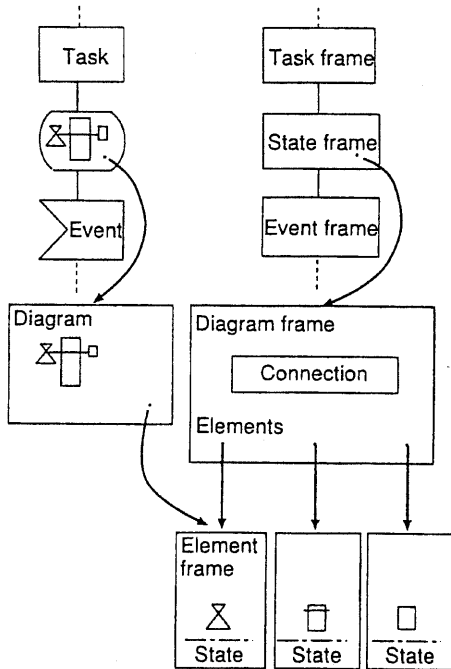


図7 図面の知識の表記

を示す。

- ・対人インタフェイスでは、SDL-GR図面と接続状態図を夫々のウィンドウに表示する。
- ・SDL-GR表記の状態シンボルのテキスト延長記号により接続状態図へリンクさせる。
- ・接続状態図全体をフレーム形式で表記する為に、図面を全て階層構成で表す。
- ・図面を構成する要素間の接続状態は、その機能を果たすスイッチ網に持たせる。
- ・資源毎に状態や種別等の属性情報を持つ。
- ・図面と各資源を階層的に接続する。
- ・階層性を考慮してフレーム形式に置換える。

以上によりエキスパートシステムは、プログラマ的と純図形的な両情報を自由に読み書きできる。

3. 3プロセス分割[Hua96]

この特徴的な機能は、入力接続状態図を読んで出力接続状態図を作りまたSDL-GR状態遷移図を作る。接続状態図の図式的な分割を説明する。

呼びのSDL-GR状態遷移図中に状態シンボルがあると、テキスト延長を経て接続状態図に至り、必要な情報にアクセスする。発信と着信側のSDL-GR状態遷移図に状態シンボルを作り、テキスト延

長で接続した新しい接続状態図の作成を始める。

- ・呼びの接続状態図の接続経路を発信と着信に分割する。
- ・呼びの接続状態図の全端末の状態を読み、発信端末を知り、発信側の処理を行ない、これ以外の端末は着信側の処理を行う。
- ・(発信や着信側の) 端末にスイッチを経由してつながる資源は、スイッチ網に対応した接続情報から読出せる。これら各資源の状態は資源毎情報から読出せる。これらを組合せ、発/着信毎の新接続状態図の情報を作る。
- ・発/着信側の接続状態図に記入。

3. 4タスク挿入[Hua96]

終着接続状態への移行に必要な資源毎の機能(タスク)を作る。図8は実現アルゴリズムを示し、左より処理前、タスク挿入の準備済み、挿入後の各状態遷移図を示す。

1. タスク挿入の準備

左図の旧状態から始めて関係する新状態を洗い出し、新状態番号をリンクに記入する。

- ・旧状態(の着目遷移原因入力)からツリーウォークを開始する。(ツリーウォークは旧から全ての新状態を含むツリーにつき行う。)
- ・新状態に至るとそれを記録して一旦停止し、新状態スタンプを動作させる。
- ・新状態スタンプは新状態から遡り、出発した旧状態に至る。この時、状態、分岐、状態遷移入力等で区切られるリンクを単位として、各リンクにタスク挿入場所を準備してこの新状態番号をスタンプする。旧状態に戻ると停止し、ツリーウォークを再開。
- ・ツリーウォークは新状態から折返して出発し、分岐があれば折り返して新状態に向け下る。新状態に至ると、前記と同じ処置をし、出発点に戻れば打切る。

これにより中央図のように各リンクに関係する新状態名が記録される。

2. 各リンクの共通な終着状態を考慮して資源毎のタスクを記入する。

- ・新状態毎のタスク表を作る。資源毎に前状態と新状態から2次元の表を引き、タスクを求め、タスク表に記入する。
- ・前状態からツリーウォークを開始する。リンク毎にスタンプがあればタスク記入を行い、

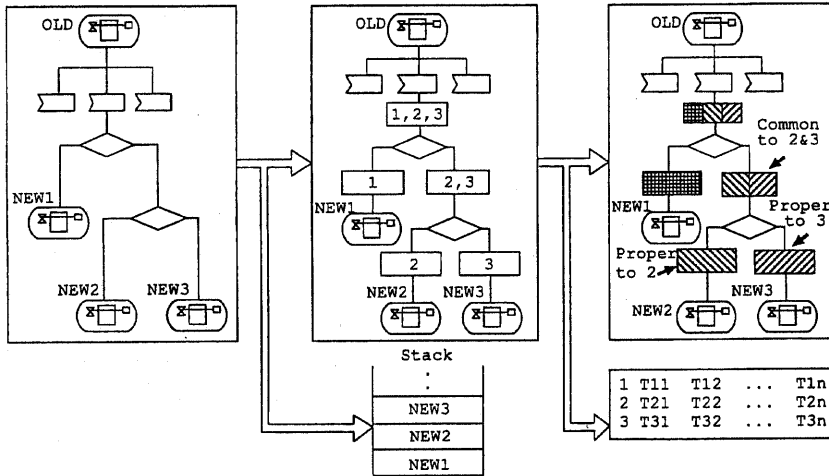


図8 タスク挿入の実現アルゴリズム

新状態に至ると折返す。

- ・スタンプがあると、対応する新状態のタスク表を取出す。新状態が2以上なら共通なタスクを求める。資源毎に共通なタスクを求め、あればスタンプに代えて記入し、タスク表からは消去する。共通な新状態が無ければそのタスク表のタスクの全てををスタンプ箇所に転記。
- ・出発点に至るまで上記を繰り返す。

これにより右の図のように、関係するタスクが重複なく記入される。

4. 検討

4. 1 自動設計の性能と効用

エキスパートシステム構築言語ES/KERNELを用いてこのシステムを構築した。処理の複雑なタスク挿入でも30秒程度で実行でき、実用に供しうる処理速度である。構築言語は知的処理向けのマクロ言語であり原理的に処理速度が遅くなりがちであるが、このシステムでは専用化によりPerformanceが向上させたと理解できる。

このような自動設計システムをS D LのCASEツールに組み込み、更に(接続状態図の為の)絵書きツールと連動できれば、

- ・開発期間短縮
 - ・同コスト減少
 - ・品質向上
- 等の効果を挙げる事が容易になる。

図9は、自動設計と入手設計のコスト比較を示

す。このシステムの開発Costは省力設備の初期投資に当たり、これが大きいと自動設計が有利となる分岐点が右にずれる。支援系の開発担当にはこの開発コストの低下が、また使用者には標準化等により使用回数を増す事が、夫夫求められる。

図10はエキスパートシステム構築用マクロ言語の種類毎の使用頻度であり、システムを作る中心技法により変わるが、少数の種類が大部分を占めている。これはシステム開発工数低下の手がかりになりそうだ、と考えている。このようなマクロ言語を使うと純プログラムレベルの工数は小さく、適用分野の専門知識を習得する期間とシステムの論理をマクロレベルに展開する工数が大き

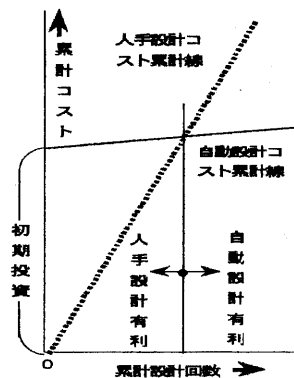


図9 自動設計のコスト評価

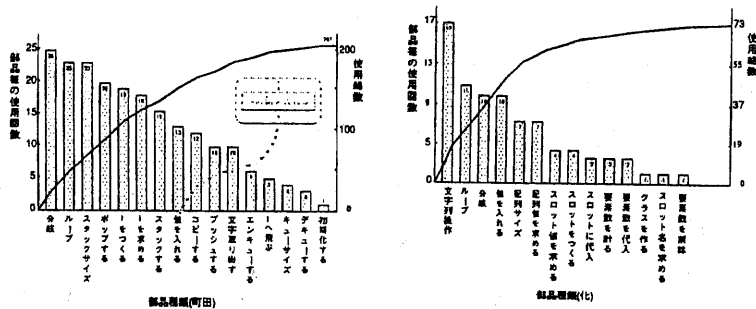


図10 各種の部品使用状況

い。これら部品を標準化して使い易さの改善はかり、システムの設計者が自動設計システムを簡単に開発できる方法を研究する計画である。

一方、使用者側には価値感を転換する事が求められる。設計方法(例えばプロセス分割)から接続状態図等の表記法を標準化して同じ自動化設備を使用可能にする事、その開発の方法、更にその基礎や部品の共通化等を標準化して同じ機能を各所で使えるようにする事の両面がある。これらの効果は目に見え難いけれども間接あるいは巨視的に大きなコスト低減をもたらす事に留意して、これら自動化システムを積極的に利用して効果を挙げる事が望まれる。

4. 2 設計の自動化とその推進

人の設計に倣う方針から、これまで本システムをエキスパートシステムとして説明した。しかし、人の設計アルゴリズムを自動設計システムに移し替えた自動化システムをDesign Automation (DA) Systemと呼ぶ。本システムはソフトウェア用のDA Systemでもある。

これまで人工知能では、知識を系統的に獲得する事は出来ない、またエキスパートシステムを系統的に作る事は困難である、と考えられてきた。しかし、現在ではモデル化(設計では設計工程をモデル化)する事により系統的に知識が獲得[Ford93, Koono94, 陳97c]でき、体系的な知識をエキスパートシステムに系統的に再構成[Chen96, 陳97c]する事ができる。このような視点から見るとDesign Automationは現在盛んに用いられるエキスパートシステムの1応用技術と言えよう。

本報告のように、人の設計アルゴリズム(これはルールベースの知識)を利用する専用の自動設計システムは100%の自動化を達成し易く、

・期間短縮 ・工数節減 ・品質向上
等の大きな効果/効用を挙げられる。この種の技

術が広く使われる事が望まれる。

従来この種の部分的な自動化システムはあまり利用されていない。それは使用者の心理として「何かもっと完全な自動化」を求める気持ちがある事も一因である。しかし、完全な自動化は必ずしも効果的でない。完全な自動設計システムは多量の知識を必要とする。(それは永年の教育を受け更に適用分野の技術を習得しないと1人前の設計は出来ないのと同様)特に汎用性や広範な変更/拡張に耐える事を狙うと、基礎レベルからのより多量の正しい知識を要する。これまでのシステム開発の「デバッグ」に見られるように、人や組織は自己業務につき必ずしも正しい知識の体系を保有する訳ではない。多量の正しい知識の獲得には、従来と同様に多大な「デバッグ」工数を要する。故に夢のような自動設計は当面は望み薄で、

・投資が小さくて効果が大きい作業
を中心に自動化する事にならざるを得ない。人の作業体制と共存する為に、

・人の作業工程に合わせ段階的に、
投資効果を挙げる為に、

・作業量の多くて簡単な設計工程の下流や
階層的な工程の下層の作業から、

自動化する事になろう。自動設計システムが投資を考慮しても成立ち、同時に使用者もその効果を享受するには、

・設計の各種の標準化を行い、
・局所最適でなく広い観点で評価する
事が求められよう。

これまでソフトウェアの設計は、まか不可思議なものと考えてきたのではなからうか?別論文[河野97]は自動設計のみでなくソフトウェア工学一般の基礎を示している。設計をプログラムの視点でなく、人の概念展開の観点で見ええすればよい、自動設計の為に「高成熟度組織の設計知識を系統

的に獲得する」とは、最も合理的かつ理想的な設計手順を取り出す事に他ならず、これは多くの人に出来る事と考える。従来と大きく異なる事は、

- ・基礎的ルールのみが知識でなく技能的知識、
ルールの知識など各種の知識を使う、
別論文[陳97b]は技能的な知識を用い設計知識を自動獲得し、これを再利用し自動設計を行う汎用的なシステムである。
- ・完全な自動化でなくても投資効果が大きく役に立つ事が重要、

等の考え方と取組み方の転換と考える。

5. むすび

本システムは、

- ・設計知識を入力変換ルールとして取出し、
- ・これを再利用する自動設計システム

である。試作した構成方式と特性について報告した。本例は、1種のエキスパートシステムであるが、同時に従来のDesign Automationとも考えられる。今後はこれまでと異なる各種の視点や取組での自動化が積極的に使われる事が望まれる。

謝辞

この研究は多くの学生各位の努力の成果である。特に馬場、藪内、大森、滝沢氏等の貢献に深く感謝します。また、日立製作所にはエキスパートシステムについて、また電気通信普及財団およびNTTには交換システムの設計知識の研究へのご援助について、深く感謝します。

文献

- [Chen96] Chen, H., Machida, K., Far, B.H. and Koono, Z.: Software Creation: A systematic construction method of expert systems used for design, The third World Congress on Expert Systems, pp. 577 - 584, 1996.
- [陳97a] 陳, ファー, 堤, 河野: ソフトウエアクリエーション: Intelligent CASEツールの方式について, 信学技報KBSE96-26-32, 1997.
- [陳97b] 陳, 勇, 堤, 河野: 設計知識の自動獲得と自動設計を行う知的CASEツール, 情処学会ソフトウエア工学研究会資料, 1997年5月, 1997.
- [陳97c] 陳, ファー, 河野: ソフトウエア自動設計における系統的なエキスパートシステムの構築—設計工程からの設計知識の獲得と再現, 人工知能学会誌, 1997. 7 (予定)
- [Ford93] Ford, K. M. and Bradshaw, J. M. eds.: Knowledge Acquisition as Modeling, Intl. Jour. of Intelligent Systems, vol. 8, no. 1, pp.1 - 7, Jan. 1993.
- [Hua96] Hua, Ohmori, Chen, Far and Koono.: Software Creation: An Intelligent CASE Tool Experiment for Switching Software, International Conference on Communication '96, 1996.
- [Koono92] Z. Koono, T. Baba and T. Yabuuchi. Software Creation: A Trial for Switching software. Proc. 5th JCCNSS '92, pp. 261-265, 1992.
- [Koono93a] Koono, Z., Far, B. H., Baba, T., Yamasaki, Y., Ohmori, M. and Hatae, K.: Software creation: Towards automatic software design by simulating human designers, The Fifth International Conference on Software Engineering and Knowledge Engineering 1993.
- [河野93b] 河野, ファー, 馬場, 山崎, 大森.: ソフトウエアクリエーション: 人間の特性に立脚した自動設計方式, 情報処理学会ソフトウエア工学研究会資料93-3, 1993.
- [Koono94] Koono, Z. Far, B. H., Sugimoto, T., Ohmori, M. and Chen, H.: A systematic approach for design knowledge acquisition from documents, The Third Japanese Knowledge Acquisition for Knowledge-Based Systems Workshop, JKAW '94, 1994.
- [河野96] 河野, ファー: 拡張有限状態機械をモデルとするシステムの設計方法, 情報処理学会ソフトウエア工学研究会資料, 1996. 1.
- [河野97] 河野, 陳, ファー: 人の設計知識の構造とソフトウエア工学, 情報処理学会ソフトウエア工学研究会資料, 1997. 5.
- [大森93] 大森麻理, Far, B.H., 河野善弥: ソフトウエアクリエーション: 交換接続プログラムの設計ルール, 1993年春季大会, B-525電子情報通信学会, 1993.
- [大森95] 大森麻理, 馬場健, B. H. ファー, 河野善弥, ソフトウエアクリエーション: 交換接続プログラムの設計知識, 電子情報通信学会ソフトウエア工学研究会資料, KBSE95-20, pp. 37-44, 1995.