

B6 第5回情報科学「若手の会」夏の会報告

宇都宮 公 訓 (早稲田大学理工学部)

第5回の情報科学「若手の会」が、プログラミング・シンポジウム委員会の援助のもとに、下記のように開催された。

日 時 昭和47年7月12日13時～7月14日13時

場 所 長野県軽井沢町, 早稲田大学追分セミナーハウス

招待者 山内 二郎 先生(青山学院大学)

和田 英一 先生(東京大学)

野崎 昭弘 先生(東京大学)

高木 純一 先生(早稲田大学)

門倉 敏夫 先生(早稲田大学)

広瀬 健 先生(早稲田大学)

足立 暁生 先生(日本アイ・ビー・エム)

参加者 (44名)

山本光璋・海老原義彦・横山茂樹・邱坤郎・和田健一(東北大), 鈴木則久・岸健一・斉藤光男・柿下尚武・疋田輝雄・町田元・府川哲夫(東大), 所真理雄・永田守男・菱沼千明・三橋隆(慶応), 前野年紀(立教大), 富岡恒夫・大場充・塚越(青山学院大), 笠井琢美・中村奉夫・馬場敬信(京大), 島田俊夫・山口喜教(電研), 日比野靖(通研), 水野忠則(三菱電機), 箱崎紘・中島治夫・内田博保・青柳喜久(三井造船), 望月浩一・九尾英雄(キャノン), 中村剛・坂倉正純・岩田茂樹・堀米明・白井紹夫・折戸房雄・面田耕一郎・片山恒次・本間典一・諸橋正幸・宇都宮公訓(早大)

空前の大雨で、山陽本線がズタズタになり、予定されていた九州の方々が参加できなかったのは、とても残念であったが、7人の先生のアドバイスのもと、44名の参加者の中で熱心な討論が交された。討論は、6つのセッションにわけて、参加者があらかじめ用意してこられたテーマを中心に進められた。テーマは、MTCなど情報科学の最近の発展を反映したものから、現実に密着した地味なものまで、この方面における問題点を、ひろく網羅したものであった。

また、高木純一先生には、非常に有意義な特別講演をしていただいた。

ページ数の制限上、参加者の資料をそのままここに掲げることができないため、司会の方々にお願いして、各セッションごとに、討論の様子や話題を伝えていただくことにした。

最後に、御多忙中にもかかわらず、貴重なお時間をさいて、御教示下さった先生方、財政的な援助をしていただいたプログラミング・シンポジウム委員会、日本アイ・ビー・エム(株)、三井造船(株)、沖電気工業(株)、日立製作所(株)、富士通(株)、キャノン(株)、早稲田大学電子計算室に心から感謝の意を表したいと思います。

情報科学「若手の会」

第5回幹事校 早稲田大学

第1 セクション (7月12日)

ハードウェア

司会 坂倉正純

パネラ 中村剛, 白井, 折戸, 面田, 中島, 内田, 九尾, 坂倉, 正田

話題となった資料:

中村剛「非同期回路の合成に設計について」

白井「非同期順序回路の状態割り当て」

折戸「計算機の故障診断について」

面田「論理設計の自動化について」

中島・内田「論理設計の自動化」

九尾「プリント基板の設計システム」

坂倉「NAND論理回路網の自動設計」

討論の内容は、非同期回路と、論理装置の設計自動化に関する分野に分けることができる。

前者に関しては、まず、非同期回路の合成・設計に関して、いくつかの方法が歴史的な流れの形で発表された。flowtable, digital graph, flowchartなどを用いて、非同期回路が設計されてきたが、bounded delay, speed independentの条件をつけたとしても、巨視的に見た場合、synthesisが完了したかについての判定、すなわち、回路のcompletenessが未だ設定されもしていないことが議論された。

状態割り当てに関しては、MOCに対する状態割り当ての方法、さらに、UOCにも適用可能としたTraceyの拡張版が提案された。

非同期理論においては、理論と現実の間に相当のギャップがあり、実際にモノをつくることを意識した解析が必要ではないか、HuffmanとMullerをすこしでも接近させることはできないか、技術者が現場で職人芸的に回路を合成していることをヒントとすべきであるなどが指摘された。

なお、スイッチング回路のcompletenessが、2値論理の場合、およびそれを多値論理まで発展させた場合のそれぞれについて、正田氏より発表された。

設計の自動化に関しては、故障診断技術の一般論から、IBM System 370のRASまで、ひろく議論された。とくに、availabilityを高める手法に議論が集中した。

マクロな設計言語による記述を変換することによって、論理設計の自動化を行なおうとするアプローチと、実際にモノをつくるという立場にたち、ドキュメンテーションの標準化から入ろうとするアプローチの2つがそれぞれ紹介され、その得失が議論された。IBM社の論理設計システムであるALERTが失敗した理由をよく見きわめ、今後の進むべき方向がいくつか示された。

プリント基板の効率にはある程度目をつぶっても、できるだけはやくプリント基板をつくることを目指した、多種少量生産に向くシステムに持たせるべき特性に関して発表があった。このようなシステムでは、人間と計算機がスムーズにinteractすべきであること、両者が分担すべき作業などが議論された。

NAND論理回路の自動設計では、従来行なわれてきた研究の概略とその分類、今後の方針が述べられた。

(坂倉)

第2セッション(7月12日)

マイクロプログラミング

司会 鈴木則久

パネラ 宇都宮, 馬場, 岸, 島田, 鈴木, 山本

話題となった資料:

宇都宮「新しい世代のコンピュータ」

馬場「High-level Microprogramming Language」

岸「Dynamic Microprocessor」

鈴木「Dynamic Microprogramming の効用」

山本「脳におけるシステムの研究」

5人の発表者による興味深い問題提起が行なわれた。しかし、その他の参加者の反応は低く、また、この分野の研究が広く認識されるまでには到っていない感じがした。

まず最初に、宇都宮氏による、新しい世代のコンピュータの構成方法が1つ提案された。彼は、コンピュータの内部を、いくつかの機能単位に分け、それら全体の制御を司るものとして、また計算機と同じようなものを考え、inner-computer と名付けている。inner computer はwritable control storageを使い、microprocessor のようなものを考えている。これによって、本来のコンピュータの結線はダイナミックに変えていくことができる。彼は、計算機の発展は結局素子の発展であったと追想している。このようなコンピュータは大規模半導体メモリの出現によって、非常に実現の可能性が強くなったと言えよう。

次に、馬場氏による、マイクロプログラム用高級言語の話があった。これは、現在大変労力のかかっているマイクロプログラムのプログラミングを、高級言語を用いてできるようにして、労力を軽減しようというシステムの話である。ユーザは作成しようとする命令をマシーン・インデペンデントな高級言語によって記述する。また、このシステムは汎用なので、コンピュータの構造も記述言語でかく。これによって、デバッグもマイクロ・コードの作成も行なう予定だそうである。しかし、これはまさしく汎用コンパイラ・コンパイラとコンピュータの自動作製で問題となっていたことを実現しなければいけなくなる。すなわち、命令のマシーン・インデペンデントな記述と、コンピュータの構造の記述である。これらがどう解決されるかに興味があるという意見が多かった。また、私の注を申しますと、シミュレータを自動作製して、その上でデバッグしているが、コンピュータ記述言語がうまく動いているなら、命令記述のレベルでデバッグするのと全く同値となる。

3番目には、実際にミニコンにダイナミック・マイクロプロセッサを組み込んで、ダイナミック・マイクロプログラミングを実験しようという話が岸健一氏から行なわれた。このように既製のコンピュータに組み込むのはかなり難しいが、ハード・ワイアードの命令が沢山あるところで、少しダイナミックな命令を付け加えるシステムは、従来のソフトウェアとか、OSなどをこわす恐れが全然ないという点で、パーソナル・コンピュータには有望であろう。また、最近進歩しているレジデント・コンパイラなどを組み込めば、FORTRAN キャリキュレータにもなるだろう。また、実際の設計の細いところは、夏のシンポジウムで発表される。

次は、島田俊夫氏による最近のダイナミック・マイクロプログラム制御のコンピュータの動向が紹介された。これらは主に小型コンピュータでの応用例であった。

最後に、鈴木則久氏によるダイナミック・マイクロプログラミングの、ソフトウェア・システムへの応用の可能性についての話があった。とくに、拡張機能を持った言語とか人工知能のソフトウェアに取り入れられると面白い結果が出

そうである。

以上が発表の内容であるが、感想としては、マイクロプログラミングの実際の技術経験がほとんどなく、システム的なことはなかった。これは、大学の計算機が、このような目的のために使えるのは大変少いためであろう。さらに、この分野の発展的研究が望まれる。

また、山本光璋氏から、脳におけるシステムの研究に関する最近の成果が発表された。脳における単一ニューロンの自発2射系列にマルコフ性を有するものがみつかったことは、それが単なる信号のキャリアということだけではなく、生体システムを維持するために積極的に働いているという可能性を示すということであった。したがって、このマルコフ性のより詳細な内容を明らかにすることにより、神経回路網のソフト的構造に関する知見が得られるだろうとしている。

(鈴木)

第3セッション(7月13日)

OS, コンパイラ, データ・ベース

司会 前野年紀

バネラ, 海老原, 横山, 中村奉夫, 諸橋, 岩田, 本間, 片山

話題となった資料:

海老原「多重CP系の1モニタコントロール方式」

横山「計算機システムのパフォーマンスの評価」

中村奉夫「Program Behavior について」

諸橋「プログラム解析の手法」

岩田「改良型Syntax Oriented Translator について」

本間「ソフトウェアの自動生成」

片山「General Simulator Generator」

海老原氏から、東北大学の新しいTSSであるAOBAシステムが紹介され、この2CPUの対称形マルチプロセッサを1モニタ・コントロール方式にして、資源効率向上などを図ろうとしていることが述べられた。

横山氏は、計算機システムの解析・評価を、CPUの利用率のみを問題にするのではなく、メモリ・アロケーションの方法を含めて行なう必要があるという提案をされた。

Storage hierarchy systemの効率評価のための手法が中村奉夫から紹介された。このためにプログラムが必要とするメモリ・アドレスに関する情報を集める方法としてトレースなどがある。また、形式的なプログラムの定義(例えば、nonstationary Markov process)を採用して数学的に解析することも試みられている。Denningの“Principle of locality”を使ったworking set modelもその一例である。

諸橋氏は、プログラムのダイナミック・トレースの結果をFORTRANレベルにまで持ち上げ、フローチャートまでかせるシステムを作成した。IBM System/360を対象としているので、ゼネラル・レジスタの解析に特別な考慮が払われている。

岩田氏からは、ミニ・フォートランのコンパイラを、syntax-oriented translatorとして作報したとの報告があった。parsing processに関して、宣言文の取扱い、meta-variableを使用する方法など、和田、野崎先生からいくつかのアドバイスがあった。

ソフトウェアの自動生成の一方法が、本間氏から述べられた。このシステムは、PROGENS (PROgram GE

neration System)と言われ、machine independent languageによるprocedure & data と、description language によるmachine structure & behaviorが入力されると、記述されたmachine上で働くプログラムを出力する。中間段階に3アドレス方式のGeneralized Computerを想定し、そのためのGeneral Assemblerを用いるようになっていく。procedure & data の記述と、それを動かすmachineを切離して考えることを提案している。

最後に、片山氏から、computer C_1 上で任意のcomputer C_3 のためのcomputer C_2 シミュレータを発生させるプログラム、G. S. G. を試作しているとの報告があった。(前野)

第4セッション(7月13日)

応用システム

司会 所真理雄

パネラ 邱, 菱沼, 日々野, 堀米, 富岡, 塚越, 三橋, 望月, 水野, 柿下

話題となった資料:

邱「Computer Network」

菱沼「ミニコン・ネットワークについて」

日比野「Data StructureのFormalizationについて」

堀米「グラフ理論を用いた記憶域有効利用のための1つの試み」

富岡「DYNAMO処理プログラムに対するネットワーク技法の応用」

塚越「Loopの検査」

三橋「喉頭疾患と声帯音源のスペクトルおよびピッチ周期のゆらぎ」

望月「CRTを応用したシミュレーション・システム」

水野「産業計算制御とプログラミング言語の標準化」

柿下「“もの”のつくりについて」

本セッションにおける議論は、“Man-Machine Systemとしての観点から、現在のcomputer systemのあり方が妥当なものであるか”という点を根拠として各方面から検討を加えるというものであった。

まず第1の論議は、ミニコンピュータ・コンプレックスについて為された。要約すると以下の通りである。

ミニコンピュータ・コンプレックスが何を目的としているか、どこにメリットがあるのか完全に見きわめることがまずなされなければならない。そして、目的によって、同種のミニコンピュータによるコンプレックスか、異種のミニコンピュータによるコンプレックスか、それとも、中型、大型機を含むコンプレックスとするかを選択せねばならない。一方、今後の計算機のコストを考えると、大型、中型が安くなることはあまり考えられないが、ミニコンや入出力装置はどんどん安くなるであろうという点も忘れてはならない。総合的なcost/performanceでシステム設計しなくてはならない。

第2に、設計や製図の自動化についての論議が行なわれた。そもそも設計とは何か、製図とは何かという話が歴史的でないしは概念的に述べられ、CADやDAにおける基本的な問題点が挙げられた。しかしながら、実際にものをつくる場合の問題点は、理論的、技術的な部分にあるのではなく、多くはCADやDAを利用する人間の集団の側にあり、人間の教育がCADやDAの第1歩であるかも知れないという結論が出された。また、情報処理におけるデータ構造をformalizeした興味ある報告や、生産工程における制御用言語についての議論があり、定式化や標準化の重要性が強

調された。

その他に、喉頭ガンの計算機による発見のためのシステムや、グラフ理論を用いた記憶域の有効利用のための方法についての提案もなされた。

(所)

第5セクション(7月13日)

オートマトン, 人工知能

司会 足田輝雄

パネラ 笠井, 町田, 和田健一, 斉藤, 島田, 永田, 所

話題となった資料:

笠井「言語とMTC」

町田「Krohn-Rhodesの理論」

和田健一「Syntax-Directed Translationについて」

斉藤「学習と言語との関係について」

島田「Problem-solving用語PLANNERについて」

永田「Artificial Intelligence雑感」

所「設計言語ADLとその論理回路設計への応用」

笠井氏からは、言語とそのderivation treeの集合との関係についての氏自身の結果が話され、ついで、treeというdata structureに働くプログラムの正当性についての研究の抱負が述べられた。

町田氏から、finite state automatonの分解に関するKrohn-Rhodesの定理が紹介された。

コンピレーションにおいて、セマンティカルな部分をformalizeしてシンタクティカルに行なう方法と、それに関係する話が、和田氏から行なわれた。

斉藤氏は、人間における学習と言語との密接な関係を通して、コンピュータの学習を考えてみてはという提案をされた。

島田氏から、最近、注目を浴びているPLANNERについて、例を用いて平易な紹介が行なわれ、さらに、ロボットや自然言語処理への応用が期待されることが述べられた。また、PLANNERをimplementした際のアメリカ人のバイタリティにあらためて学ぶものを感じた。

永田氏からは、汎用数式処理システムについて、それが壁にぶつかっていることが述べられ、次に、theorem provingの分野においての方法があるresolution principleやPLANNERについて言及され、さいごに、AI全体に対して、その意義や方法について述べられた。

所氏の汎用設計システム記述言語ADLが、とくにこのセクションで紹介された。ADLは、設計プロセスにおけるiterationに着目し、簡単なプログラムによって、大きな省力化を目指す、DAに対する非常に現実的なアプローチであると思われる。

(正田)

第6セクション(7月14日)

その他

司会 柿下尙武

パネラ 青柳, 箱崎, 前野, 大場

話題となった資料：

青柳「Graphic Display Software」

箱崎「Real Time Physical Distribution Simulation System」

前野「会話型デバギング」

大場「Man-Machine Communicationによる設計作業」

本セクションでは、種々の事情から第1セクションから第5セクションまでに討論できなかった資料を取上げることにした。

まず、青柳氏から、ミニコンのためのソフトウェアを開発する上で問題となる点、graphic displayの応用面、使用上の問題点およびミニコンと大型機との関係などを議論するよう提案があり、参加者の考え、経験談などが交わされた。

シミュレーション経過を目で見ること、試行錯誤が可能なインクラーティブなシミュレーションであること、シミュレーション・データが任意に得られることを目的として開発した、リアル・タイム・シミュレーション・システムが箱崎氏から報告された。しかも、経済性を重視して、独自のグラフィック・ディスプレイを開発し、これをミニコンで制御する方法を採用している。このシステムの大きな特色は、ミニコンの計算スピードの不足をカバーするため、ディスプレイ部、統計、シミュレーション・データ処理部にファームウェアがとり入れられていることと、シミュレーションの時間軸を実時間も含め自由に換えられ、シミュット状況を刻々と観察できることである。

をある。

次に、前野氏から、タイプライタを使つての会話型のデバギングについて、貴重な報告があった。会話型の虫取りでは、プログラムの追跡指定がフレキシブルである、タイプライタから割込んだり、ブレイク・ポイントをあらかじめ仕掛けておく実行中断ができる、実行の再開指定が容易であり、レジスタ、メモリなどの内容がシンボリックに吟味できる、プログラムをソースレベルで簡単に修正できる、出来上ったプログラムの登録や基本動作をまとめて新しい命令として登録することが容易であるなど、多くの利点がある。

最後に、大場氏から、simulationを用いたinductive approachによる設計作業に関する意見が述べられた。そして、よりよい設計にするためには人間とコンピュータがinteractionが重要であり、on-line simulationが望まれる。しかも、人間の試行錯誤を容易に、かつ能率的に行なわせるために、parameter化のサブルーチンを付加するようにした方がよいことが指摘された。
(宇都宮)

特別講演(7月14日)

高木 純一 先生

政治、経済から美術、文学にいたるまで、その歴史を大切にしているのに、理工では、なぜ歴史を取扱わないのかという疑問を抱きまして、学部を卒業してまもなく、わたしは、われわれの方面における歴史をコツコツ勉強するようになりました。歴史をやれば、必然的に他の分野との関り合いが生じ、自然科学が経済や政治と無関係でないことがよく分ります。わたしは、限定された領域で、限定されたテーマを厳密に議論することを止めて、自由に物事を考えることを始めました。

コンピュータの持つ1つの特徴は、公平にものを考えてくれることでありましようが、人間は本質的に主観的であり、主観の強い人文科学、とくに、政治学をどのようにシステムティックにとらえて行くかを考えております。そのためには、政治学や政治にたずさわる人達と共通の言葉を見つけ出すことから始めなければなりません。あるとき、政治家に

コンピュータの話をしていて、“システム”という言葉が非常に広く使われていることに気がつきました。“システム”は、いまや流行語なのですね。この言葉の定義は難しく、各人都合のよいように考えているのですが、天文家、宗教家をふくめて、その最大公約数をとってみますと、それは“秩序”ということなのですね。そういう、非常に広いシステムを、限定しないで、社会、人文、自然科学、工学にまたがらせようということです。産業革命以来、分業によって発達してきた諸科学は、今日分化による悲劇を見せ始めています。これらを、システムという面から再総合し直そう。つまり、何をやっても、他のことを少しは行えて行こうということなんです。今日では、“専門違い”だからといって、知らないことを誇りにしてはいけません。みなさんは、“コンピューター筋”ではありませんか。芭蕉は決して誇りに思っ“この道一筋”と感慨にふけたのではないのですよ。幸い、“システム”という共通言語があるのですから、これを駆使して、他の分野に迫って行って下さい。

わたしは、システム論的発想を諸科学に試みることを始めたわけです。“システムとは何か”を語る場合には、ある程度の定量性を考えなければなりません。システムの中を流れる情報や物流の量を考えます。“エントロピー”という言葉は熱力学から出たものなのですが、これはぜひ社会科学にも持込まなければなりません。エントロピーが増大するのは気が楽なのですが、社会科学では、エントロピーが減少することが多いですね。聖書にはとても素晴らしいことが書いてありますが、そのうち、信仰を篤くすることを説いた一節に、経済的な比喩を用いて“持てる者はなお与えられて余りあり。持たぬ者は持てるものをも取らるるなり”というのがあります。これは自由経済の原理みたいなものなのですが、社会科学には、こういうふうに入エントロピーが減少する傾向をもつシステムがよくあります。社会全体の価値感として、それは困るというのであれば、積極的に是正して行かなければなりません。

いま、“価値”という言葉が出てまいりましたが、価値ということを考えなければ、学問の総合化はできないでしょう。自然科学や工学では、価値といっても比較的狭く、学問では一般に、客観が尊ばれ、主観が排斥されてきました。しかし、主体性をもつことは主観を前面に押し出すことですので、社会学、経済学、政治学では主観を取り入れなくてはなりません。各人まちまちの価値観がゴチャゴチャに入り乱れていても、全体としてうまく行く社会を考えなければなりません。社会における政治形態を論じると、“多数の価値観を内蔵している autonomous system ”ということができます。しかも、文明もそうなのですが、政治にも目標がありません。したがって、常に試行錯誤なんです。この“試行錯誤”という手段は、コンピュータの中にぜひ取入れて欲しいですね。

個人を1つのシステムと考えても、矛盾した価値体系を内蔵しています。自然言語もそういう中から生まれましたから、反対語というのが沢山あり、人間はAを考えるとき、同時に non Aを考える習性をもっています。試行はトリーになります。Aか non Aかを決定して先へ進み、さらにデジションを下して先へ進みます。しかし、大切なことは、人間は、デジションを下して一方を選択しても、他方を決して捨ててしまったわけではないですね。Aを選択して進んで行って、行きづなれば、また non Aまでもどってやり直してみます。そうしながら、常に中庸が尊ばれています。

コンピュータと社会を近づけるには、研究課題として、もうひとつ“成長”を取り入れなければなりません。人間社会といえども歴史により成長がありますが、とくにこれが積極的にやられているのは生物です。人間社会では“数学が進んでも政治は貧困である”というように、各部で反応速度がとても違っているのですが、生物では、とても調和がとれた成長が行なわれています。生物の生まれてから死ぬまでの成長のタイプには、“建て増し型”と“脱皮型”の2つがあり、とても教示的です。生物におけるシステムは成長しています。成長しながら死んでいます。寿命を考えなければなりません。そういうことをしながら、人間を研究して行くことが大切です。解剖だけが道ではありません。登る道はいろいろ用意されているのです。

(2時間弱の講演から 宇都宮)

本 PDF ファイルは 1965 年発行の「第 6 回プログラミング—シンポジウム報告集」をスキャンし、項目ごとに整理して、情報処理学会電子図書館「情報学広場」に掲載するものです。

この出版物は情報処理学会への著作権譲渡がなされていませんが、情報処理学会公式 Web サイトの https://www.ipsj.or.jp/topics/Past_reports.html に下記「過去のプログラミング・シンポジウム報告集の利用許諾について」を掲載して、権利者の検索をおこないました。そのうえで同意をいただいたもの、お申し出のなかったものを掲載しています。

過去のプログラミング・シンポジウム報告集の利用許諾について

情報処理学会発行の出版物著作権は平成 12 年から情報処理学会著作権規程に従い、学会に帰属することになっています。

プログラミング・シンポジウムの報告集は、情報処理学会と設立の事情が異なるため、この改訂がシンポジウム内部で徹底しておらず、情報処理学会の他の出版物が情報学広場(=情報処理学会電子図書館)で公開されているにも拘らず、古い報告集には公開されていないものが少からずありました。

プログラミング・シンポジウムは昭和 59 年に情報処理学会の一部門になりましたが、それ以前の報告集も含め、この度学会の他の出版物と同様の扱いにしたいと考えます。過去のすべての報告集の論文について、著作権者(論文を執筆された故人の相続人)を探し出して利用許諾に関する同意を頂くことは困難ですので、一定期間の権利者搜索の努力をしたうえで、著作権者が見つからない場合も論文を情報学広場に掲載させていただきたいと思えます。その後、著作権者が発見され、情報学広場への掲載の継続に同意が得られなかった場合には、当該論文については、掲載を停止致します。

この措置にご意見のある方は、プログラミング・シンポジウムの辻尚史運営委員長(tsuji@math.s.chiba-u.ac.jp)までお申し出ください。

加えて、著作権者について情報をお持ちの方は事務局まで情報をお寄せくださいますようお願い申し上げます。

期間：2020 年 12 月 18 日～2021 年 3 月 19 日

掲載日：2020 年 12 月 18 日

プログラミング・シンポジウム委員会

情報処理学会著作権規程

<https://www.ipsj.or.jp/copyright/ronbun/copyright.html>