

ノイマン型コンピュータ教材を用いた計算機アーキテクチャ教育支援環境

今井慈郎*, 金子敬一**, 中川正樹**

* 香川大学総合情報基盤センター, ** 東京農工大学工学部情報コミュニケーション工学科

* 〒761-0396 香川県高松市林町 2217-20 E-mail: imai@eng.kagawa-u.ac.jp

** 〒184-8588 東京都小金井市中町 2-24-16 E-mail: {k1kaneko, nakagawa}@cc.tuat.ac.jp

あらまし

ノイマン型コンピュータ教材 VisuSim を用いた計算機アーキテクチャ教育支援環境について述べる。教材 VisuSim は Java で記述された計算機シミュレータであり、内部構造・動作の可視化やアセンブリ言語プログラミング支援などの特徴を有する。学部講義「計算機アーキテクチャ」での VisuSim の活用事例を紹介し、活用のための教育支援環境について説明し、学習者への個別化や教材の適応性などの試みについても現状を報告する。
キーワード: アーキテクチャ教育支援, 計算機内部の可視化, 計算機ビジュアルシミュレータ, 教材の使用法

An e-Learning Design for von Neumann Computer Architecture and its Application to Real Education

Yoshiro Imai*, Keiichi Kaneko** and Masaki Nakagawa**

* Kagawa University ** Tokyo University of Agriculture and Technology

* 2217-20 Hayashi-cho, Takamatsu, Kagawa, 761-0396 Japan E-mail: imai@eng.kagawa-u.ac.jp

** 2-24-16 Nakacho, Koganei, Tokyo, 184-8588 Japan E-mail: {k1kaneko, nakagawa}@cc.tuat.ac.jp

Abstract

Von Neumann Computer is one of the basic structures for Computer Architecture Education. In order to help students understand basic architecture, a graphical computer simulator has been developed to demonstrate how a computer works graphically. It is written in Java programming language and executable in the stand-alone application mode as well as in Java applet one on the browser. It is used not only as an educational system for classroom lecture of Computer Architecture but also as Web-based educational environment for assembly programming.

We will introduce how to apply our computer simulator, called "VisuSim", into real Computer Architecture lecture and also explain a mechanism of support environment to utilize the simulator in the Web-based education. Finally, we will report our challenge of personalization and adaptation for each learner.

keywords: e-Learning Design, Education for Neumann architecture, Computer visual simulator, Adaptation

1. はじめに

IT(ICT)教育が注目されて数年が経過したが、教育効果を上げ、教育効率を高めるためには、教育技術の向上と教育手法の改善が必要となる。汎用的な教育の議論を行うことも重要であろうが、ここでは、筆者らが直面する計算機教育における、教育技術や教育手法の中からテーマを絞り込んで、扱うべき教育対象も限定して議論を進めたい。

幸い、情報処理学会「コンピュータと教育」研究会での発表という機会を得た。計算機アーキテクチャ教育にポイントを絞り、計算機教育における教育技術の向上と教育効果の改善を目指した筆者らの試みについて報告し、有意義な議論やコメントをいた

できれば、まさに「願ったり、叶ったり」の状況である。発表内容は、以下の通り。

まず、2節では教育技術について、「可視化」という観点から議論する。次に、教育手法について、「個別化」と「適応性」について、どのような機能を如何に実現できるかについて述べる。このような議論を受けて、3節では計算機アーキテクチャ教育に利用する試作ツール作成の経験に基づいて事例紹介し、「可視化」や「個別化」「適応性」についてその問題点と筆者らが考える対応について詳述する。4節では、試作ツールの適応例を示し、「可視化」「個別化・適応性」を如何に実現するかについて、議論を進め、まとめとして、今後の課題なども明らかにしたい。

2. 教育技術と教育手法に関する考察

本節では、以降の議論の基本を整理し、教育技術の「可視化」および教育手法の「個別化」「適応性」について具体的に述べ、問題を提起する。

2.1 教育技術としての「可視化」

物理(自然科学)の現象をシミュレーションなどで映像化し、可視化することで、より理解を深める効果は今更、言及するまでもない。本稿では、可視化の具体例として、これまでブラックボックスとして扱われることもあったコンピュータの内部構造や動作について可視化する効果について述べる。

IT(ICT)教育において、コンピュータを活用する教育の重要性は他言するまでもない。問題は、どれほど効果的に、コンピュータを理解させ、その応用技術などを修得させることができるかである。コンピュータを単なるブラックボックスと見なしては、問題点を解決できない。もちろん、階層的アーキテクチャという観点からすれば、高級言語レベルも、機械語レベルも、あるいは電子回路レベルもあり多種多様だが、以降では、コンピュータの構造や動作をより具体的に理解できるレジスタトランスフェラブルでのコンピュータアーキテクチャについて、可視化という観点から議論したい。

例えば、アドレス修飾法などを議論する上で、対比して関連する方式を説明することは効果的であり、それを可視化することで、実効アドレスなどの概念を教示することも容易である。可視化技術を教育に応用する事例はコンピュータ教育だけでも先行研究も多く、有意義な教育効果の報告事例も多い。[2][3]

2.2 教育手法としての「個別化」「適応性」

次に、「個別化」および「適応性」について議論したい。教育手法として、1対1を基本とする対面教育や小集団教育が望ましいのは明らかだ。効率を度外視すれば、徒弟制度のような個人レベルの教育が、かつて、技術教育の根幹を為した時期が長かった。しかし、大学の大量化が叫ばれて久しい今日、一度に数十人から数百人を相手にする1対多型マスプロ教育を避けて通れない状況になった。もちろん、工学部などでは比較的条件が良いと思われるが、効率重視の方向は今後も継続と思われる。

1対多型教育が本当に効率的かと言えば、筆者らの経験でも、答えは否定的となる。では何故、小集団教育や対面教育が現実的ではないかと言えば、コストの議論を避けることはできないだろう。

初期コストは幾分？高くなるが、巧く仕組みを作れば、教育システムや教材をユーザである学習者の

個性や理解度に沿うような仕組み作りも可能ではないか。ここでは、学習者の個性を識別する手法を「個別化」と表現する。また、学習者の個性を識別した結果、その理解度や問題解決能力に即した教育を行うことで、個人レベルの教育効率を向上させる手法を「適応性」を有すると表現する。

以降では「個別化」と「適応性」とは1対多型マスプロ教育を、言わば仮想的に1対1教育(あるいは小集団教育)へと質的に転換する手法であると捉えて、参考文献や経験に基づき具体的に議論する。

(a) 学習者の理解過程の可視化

図1を参照して欲しい。学習者は時間経過と共に、理解度を増加させ、個人差はあるものの、最終的には定常状態、即ち、飽和状態となり、結果として、学習意欲も低下するという研究報告(文献[1]に紹介記事)がある。

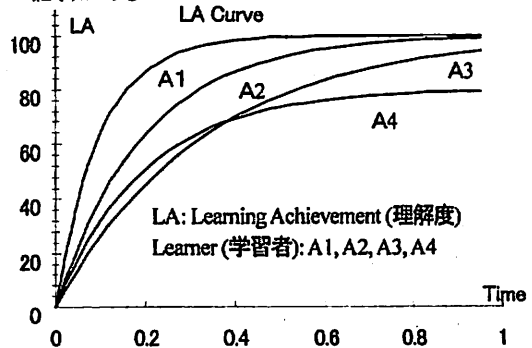


図1 学習者の理解度増加曲線(文献[1]より)

筆者らの個人的な経験では、図1のような凡その振る舞いは確かに感じられるが、特に新しい課題や領域、あるいはもっと卓近に項目に出会う時、立ち上がりにも個人差が目立ち、むしろ図2のような振る舞いを目にすることが多い。

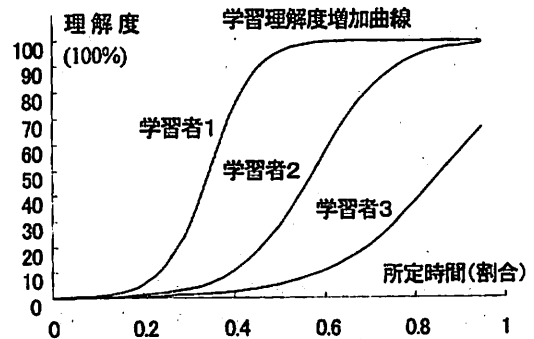


図2 経験に基づく学習者の理解度増加曲線

これは以下のような結論として要約される。

- 学習者は、与えられた理解対象について、初め戸惑いや不慣れなどの理由もあって、理解度の上昇率は鈍い
- 時間経過と共に、学習者の理解度は徐々に向上し、その傾きは急峻になる
- 最終的には学習者の理解度は飽和し、定常状態に至る

この議論で、重要なことは、学習者が個別にしかもテーマごとに理解度曲線を持っていること、教育効率を上げようとするれば、その傾向を識別する手法が必要であること、などである。これまでの議論で言えば、「個別化」をどのように実現するかという点が大きな課題になる。

一般に、理解度曲線を決定論的に議論することは、難しい場合も多い。確率論的に扱う方がより有用であろう。では、どのようなモデルが可能であろうか。以下では、「個別化」を如何に実現するかという観点から議論する。筆者らは、学習者に具体的な問題や課題を回答してもらうことで、そのスコアや所要時間から理解度を推定し、「個別化」情報を図る。スコアは、1問が何点という尺度でも良いが、例えば、10問中の回答数や正解数という形でも良い。

一方、所要時間の測定は、可能であれば、タイムトライアルとし測定できる。これが難しい局面では、一定時間での回答数や正解数という形式で、間接的に測定可能である。学習者に自己申告させる方法も検討すべきかもしれない。

(b) 適応性を実現するためのモデル化

次に問題となるのは、「個別化」をどのように活かして効率の良い教育を実現するかである。その際、問題となるものの1つは、教材や教育システムが個々の学習者に対して有する「適応性」の実現である。いくら学習者の理解度を識別する「個別化」を行っても、それに応じた「適応性」をシステムが有していないと、その後の発展が閉ざされる。

例えば、理解度が急上昇していると判断すれば、飽和する前より難易度の高い、次なる学習対象に移行することで、個々の学習者のモチベーションを保ちつつ、教育の効率を実現できる。また、理解度が低迷状況に留まっていると判断すれば、同種の問題を繰り返し提示することで慣れさせ、自信を持たせながら、少しずつ理解度向上局面への誘導を図ることも可能となる。このような手法を導入することで、学習者に応じた問題提示ができることで教育の効率化が実現できていると言える。

では、どのようなモデルを立てるか。筆者らのアイデアでは、学習者の理解度に対し、図3のような状態遷移モデルを対応させ、それを図4のような推移行列で表現することで、理解度やその状況に応じた判断・対応などの処理や計算を客観的かつ定量的に扱うことができる。

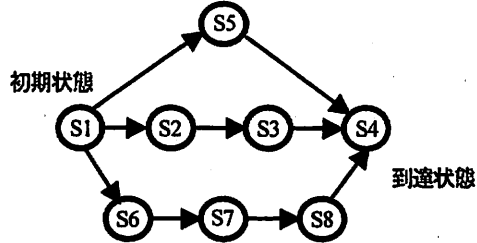


図3 理解度の増加を表す状態遷移モデル

$$\begin{pmatrix} 0 & 1 & 0 & 0 & 1 & 1 & 0 & 0 \\ 0 & 0 & 1 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 1 & 0 \\ 0 & 0 & 0 & 0 & 0 & 0 & 0 & 1 \\ 0 & 0 & 0 & 1 & 0 & 0 & 0 & 0 \end{pmatrix}$$

図4 理解度状態遷移モデルに対応する推移行列

対象となる課題・項目に、状態遷移モデルを対応させる。初期状態から出発する各遷移系列は理解度曲線を、各状態(ノード)は目標達成までの理解状況を示し、各状態(ノード)に、その状況に応じた助言や参考データを提示するための基準、チェックのための課題レベルなどが対応する。複数の経路は同一目標に向かうことになり、学習者の理解度が高ければ、より少ないノード数の遷移系列に対応させて、教育効率を改善させることが可能となる。

学習者の理解度を明確にするため、自己申告や小テストなどを行い、得られた資料それを推移確率行列で表現する。学習者の理解度の初期値は明確ではないので、まず規定値を前提に、自己申告や小テストで遷移する状態がどれか確率モデルである以上、個々の学習者は状態遷移図上の、どのノードの位置に自身が現時点で立っているかで、ヘルプメッセージや参照情報が異なる。今後どのような課題を与えるかなどもノードの位置を基準とする。これにより教育システムや教材に適応性を組み込むことが可能となり、柔軟かつ効率的な教育の実現が支援できる。

図3の例であれば、理解度の低い学習者が、4つの状態を遷移するという経路に対して、より理解度が高いと判断されるスコアなどを挙げた学習者は、通過する状態数も少ない経路に対応し、飽和状態に陥ることがなく、次の学習目標へと効率よく移行することが可能となる。

3. 計算機の内部構造・動作を可視化するための教材 VisuSim

教材に可視化機能を持たせることで、教育効果を上げようという試みはこれまで多くあった。計算機教育の分野に限っても複数の研究がなされている。[2][3][4] 本節では、筆者らが提供する教材：計算機の内部構造や動作を視覚化する目的のシミュレータ VisuSim について紹介する。まず、基本となる GUI や操作性、シミュレーション機能を示し、教材として必要なるオンラインヘルプ機能や参考となるサンプルファイルの参照機能について述べる。また、教師・学習者間での意思疎通を支援するメール送受信機能を説明する。

3.1 VisuSim の可視化機能

VisuSim は計算機アーキテクチャ教育を支援するため開発された計算機シミュレータ教材であり、GUI は同時に、計算機の内部構造やその動作を可視化する構成となっている(図5を参照)。

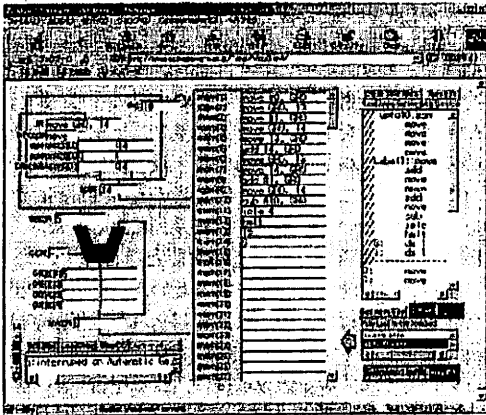


図5 計算機シミュレータ教材 VisuSim の GUI

Java で実装されているため、スタンドアロンアプリとしても、ブラウザ上のアプレットとしても動作し、ほとんど全てユーザ環境で稼働できる。単純な操作性を持ち、初期化、プログラムのロード、ステ

ップ実行、連続実行などの操作ボタンをマウスでクリックすることで、レジスタトランスファレベルの内部動作を表示する。

表1に VisuSim 内蔵シミュレータで実行可能な命令セットを示す。簡単なロードストア命令の組合せからインデックスレジスタ利用の多重ループ処理、あるいはスタック操作を伴う再帰的処理などの機械命令(アセンブリ言語プログラム)を処理できる。

表1 VisuSim で実行可能な命令セット

	Opcode	1st Operand	2nd Operand
Control Instruction	halt	None	None
	noop	None	None
Jump Instruction	jump	Direct Addr RegId IndirAddr (Reg)	None
	jpgt		
	jpge		
	jpit		
	jples		
	jpgeq		
	jpgre		
	call		
	ret	None	
Unary Operation Instruction	neg	Direct Addr	None
	push	RegId	
	pop	IndirAddr (Reg)	
Binary Operation Instruction	move	Direct Addr IndirAddr (Reg) Immed Val	RegId
		RegId	Direct Addr RegId IndirAddr (Reg)
	add	Direct Addr	RegId
		RegId IndirAddr (Reg) Immed Val	
	sub	Direct Addr	RegId
		RegId IndirAddr (Reg) Immed Val	
	mul	RegId	RegId
		Immed Val	
	div	Direct Addr	RegId
		RegId IndirAddr (Reg) Immed Val	
and	Direct Addr	RegId	
	RegId		
	IndirAddr (Reg) Immed Val		
or	Direct Addr	RegId	
	RegId		
xor	Direct Addr RegId IndirAddr (Reg) Immed Val	RegId	

この部分では、可視化機能に限定した話題について述べた。VisuSim で採用しているアーキテクチャはノイマン型計算機としては最小セットであろう。しかし、これまでの使用経験や他研究者との議論の結

果、導入教育には適性レベルの内容であろうと考えている。

3.2 オンラインヘルプ機能と参考ファイル参照機能

教材としての VisuSim に必要とされる機能はその操作性と共に、ヘルプ機能や参考プログラムの参照機能である。これまでの VisuSim 使用経験からヘルプ機能や参照機能を効果的に利用することで、ユーザに有益な情報を通知できることも明確になった。

まず、オンラインヘルプ機能について述べる。ユーザにとって、VisuSim の命令セットなどを確認し、代表的な命令の利用法などを随時確認できることは必須事項である。また、操作に関する情報を得ることも重要である。このような情報を VisuSim 使用中に入手できるサービスを提供するのが、オンラインヘルプ機能である。GUI 上のヘルプボタンをクリックすることで、サーバに照会要求を送信し、情報を取得し、オンラインヘルプメッセージウィンドウ上に表示する(図6を参照)。

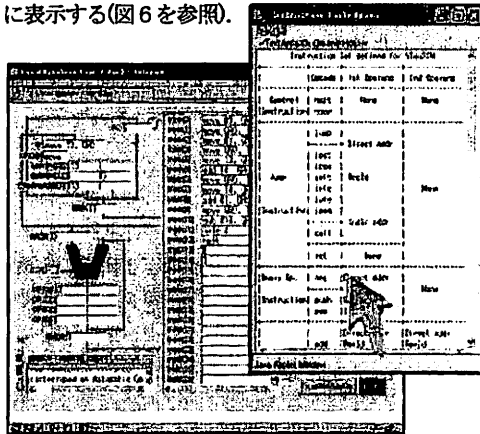


図6 オンラインヘルプウィンドウの表示

次に、参考となるファイルを参照する機能について述べる。VisuSim は内蔵するシミュレータで計算機の動作を可視化する。そこで、ループ処理やスタック処理を例題として、計算機の内部動作を示したい場合、参考となるプログラムを与えて、シミュレータで実行させながら処理を視覚化することになる。参照機能では、予め教師によって用意された複数のサンプルプログラムを選択し、サーバ上からダウンロードしてファイル取得することできる。これは、教材提示という意味でも、初心者がゼロからプログラムを作成しなくてもよいという意味でも有効な機能である。手順は以下の通りである。

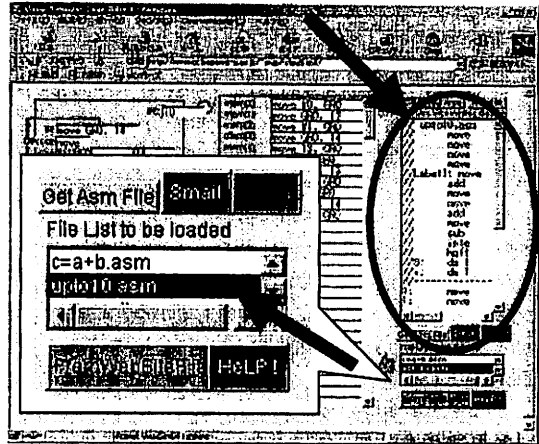


図7 参照ファイル名の表示とファイル読み込み表示

- (1) GetAsmFile ボタンをクリックして、選択可能なサンプルファイル群のリストを確認
- (2) 表示されたリストから参照したいファイル(サンプルプログラム)名をクリックして指定(図7の左矢印を参照)
- (3) スクラッチパッド領域に読み出して、ファイルの内容を確認(図7の右矢印を参照)
- (4) シミュレーションを実行したい場合、スクラッチパッド領域内のプログラムを、ProgLoad ボタンを押してメモリ上に転送

以上の手順でサンプルプログラムをメモリ上にロードすれば、VisuSim の内蔵シミュレータにより、メモリ上のプログラムは、ステップ実行あるいは連続実行可能である。また選択したファイル内容を直接メモリに読み上げるより簡便な方法も可能である。

オンラインヘルプ機能や参考ファイル参照機能は2節で述べた適応性の1つとして位置付けることが可能である。具体的な説明や応用は次節で紹介するが、ユーザである学習者の理解度を予め把握できればより適切なヘルプメッセージや参照ファイルを提供することが可能となる。

3.3 メール送受信機能

VisuSim は計算機の内部構造や動作の可視化機能を持ち、GUIにより簡便な操作性を提供できる。オンラインヘルプ機能やサンプルプログラム参照機能を活用することで、自学自習が容易な計算機アーキテクチャ教材として利用できるはずだった。しかし、実際に演習などに利用すると問題点も明らかに

なった。次のような場合、ユーザである学習者は機能不足に悩まされるという不満を持った。

サンプルプログラムを取得し、必要に応じてヘルプ機能を活用して、プログラムを作成したとする。どうも意図した動作をしない場合、学習者は迷路に入ってしまう。近くに友人や教師が居れば、その場で確認することもできるが、宿題などの場合、一人で問題解決に挑戦することになる。迷路から脱出するための手段が提供されていないという不満を感じることになり、教材としての評価も下がる。教師が近くに居ても、集団での演習利用の場合には同様の問題が生じる。

ある学生が陥った問題の内容を教師に伝えたい場合、その状況を正確に伝達することが必要となるが、一般には容易ではない状況も多い。対面であれば可能な場合もあるが、教師が一人で数十名の学習者を指導している場合、各学習者の直面する状況を正確に把握することは、問題解決を期待される教師にとって必要であるのは明らかだが、難しい状況も少なくない。では教育システムとしてはどのような支援が可能となるか。VisuSim を使用する教育現場でも同様の問題点が明らかになった。

このような使用経験に基づき、数年前、VisuSim を改善する際に、学習者と教師間で情報交換を容易に実現できるよう、メールハンドラを Java で記述し、メール送受信機能を組み込んだ。メール送受信機能の概要は以下の通り(詳細は文献[5]を参照)。

- 送信は内蔵する SMTP クライアント(Java で記述)が担当。Java アプレットの性質上、Web アプリとして起動された場合、ダウンロードマシンをサーバとしてメールを送信

- 受信は同じく内蔵する POP3 クライアントが担当。これも Java アプリとしての制約を受け、ユーザ認証後、サーバからメール内容を取得

加えて、専用のメール送受信機能を内蔵として組み込んでいるメリットを活かし、VisuSim の内部状態など含むシミュレーション環境を送信メールの内容に保存し、送信することができる。一方、受信したメール内容から保存された他者のシミュレーション環境を、受け側の VisuSim の内部状態としてリストアし再現できる機能を実装した。

このようなメール送受信で受け渡される内部状態としては、VisuSim の各種レジスタやメモリ内容が含まれ、これに加えて、スクラッチパッドの内容までをシミュレーション環境として含む。そのため、演習などで作成中のプログラムに付随する情報を総て遣り取りすることが可能となった。その結果、図 8 に示すように、左の学習者側は自身の VisuSim の状況を内蔵のメール送信用 SMTP クライアントを利用して教師などに送ることができる。右の教師側では受信したメールから学習者の VisuSim の内部情報などをリストアし、同様に教師自身の VisuSim 上で再現でき、問題箇所を共有・確認できる。

非同期的に教師は学習者の体験したシミュレーション状況を共有することが可能となる。この一連のメール送受信機能およびシミュレーション環境の保存・リストア機能を活用することで、VisuSim のユーザ間では、それぞれの内部状態などを相互に交換し合うことで、ユーザ間あるいは学習者・教師間でのコラボレーション支援を実装できることになる。これまで述べてきた機能を 2 節で議論した観点から見直してみたい。

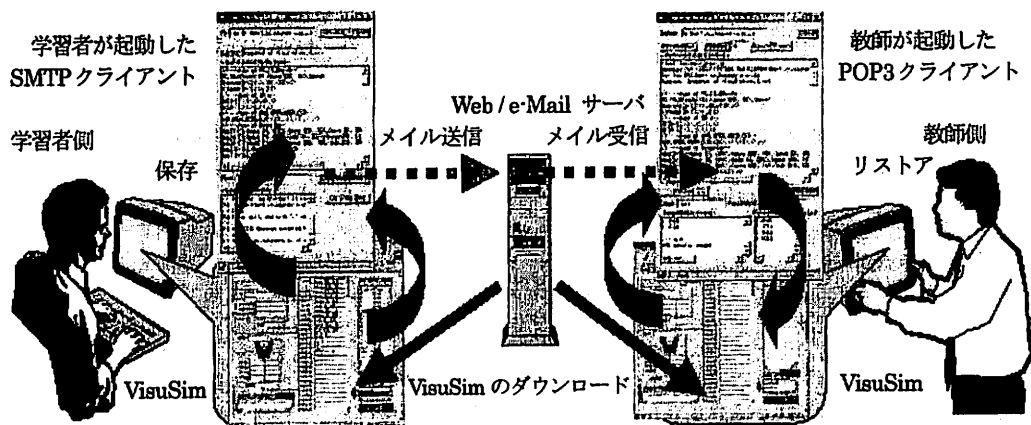


図 8 VisuSim が提供するメール送受信機能により学習者・教師間でのシミュレーション環境の非同期共有

メール送受信機能を VisuSim 本体に組み込むことで、ユーザ間において VisuSim 内部状態を非同期式ながら共有することが可能となった。これにより、学習者・教師間で適切な情報交換ができれば、教師が学習者の理解度をある程度的確にしかも、非同期という条件下で素早く把握することが可能となる。その意味で、ここで述べた機能を個別化支援として評価することができよう。VisuSim を用いた具体的な適応例は4節での詳述するが、このようなコラボレーション機能により、学習者の陥っている問題点や教師が当該学習者に対し今後どのような指導をすべきかといった課題を明確にする指針を得ることも可能となる

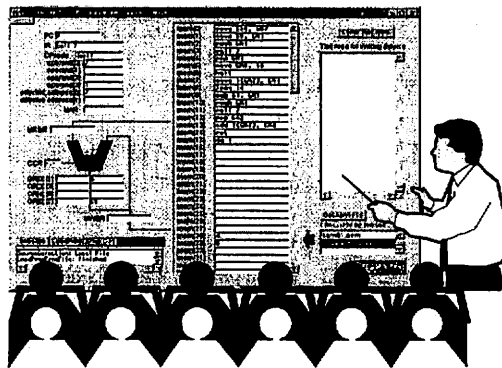


図9 VisuSim を利用した実際の講義イメージ

4. 計算機アーキテクチャ教育の実際

VisuSim の諸機能を活用し、如何にアーキテクチャ教育を実現しているか、あるいは今後実現しようとしているかについて、具体的に紹介する。

4.1 VisuSim を利用したアーキテクチャ講義とアセンブリ言語プログラミング演習

実際のアーキテクチャ教育では、ノイマン型計算機の理解を基礎として、布線論理とマイクロプログラミング、パイプライン制御、スーパーパイプライン・スーパースケラ・VLIW というプロセッサアーキテクチャに続き、参照の局所性に基づく記憶の階層構造、キャッシュ方式から仮想記憶方式へと話題が用意される。従って、ノイマン型計算機に関する教育はできるだけ効率よく実施されることが望ましい。VisuSim のような可視化機能を持ち、自学自習可能な教材を準備する必要性もこのような事情による。図9に、VisuSim を用いて、ノイマン型計算機の内部構造や動作を視覚化する講義イメージを示す。

演習では、短時間の講義形式で計算機の概要と共に VisuSim 自体の利用方法を示し、 $A+B=C$ のような簡単な例題で算術命令での処理の理解を促した後、いくつか例題、例えば、1からnまでの整数の総和を計算するプログラムなどを紹介し、具体的な機械語(アセンブリ言語)プログラミングに挑戦することになる。比較・条件分岐命令、繰返し構文を視覚的に理解するなど重要なテーマである。対象となる数列に規則性がない場合、インデックス修飾の効果を明示するテーマとなる。アドレス修飾を理解し、多重ループに挑戦すると任意の数値列を対象として、バブルソート程度のアプリが計算機内部でどのような実現されているか、どのような命令セットがあれば十分か、などが理解できる。

サブルーチン呼出しや復帰といったテーマも紹介可能となれば、スタックを操作し、引数を扱い、再帰的サブルーチン呼出しなどのノイマン型計算機としてはやや進んだ学習内容も視覚的に扱うことが可能となる。

図10に、昨年度取り上げた、VisuSim を使用した演習課題におけるレポート提出結果を掲載する。20名の学生に5テーマの演習課題(簡単な構造のプログラム、二重ループ、再帰的定義での階乗計算、フィボナッチ数列計算およびクイックソート;50点満点)を課した結果である。単純構造のプログラムや二重ループまでの理解は深まっており所定の成果を得た。再帰的定義を理解することが前提となる課題では45%の学生のみが回答を出すに留まった。やはり、再帰的サブルーチン呼出しなどを自力でアセンブリプログラミングすることは上級学年においても難しいテーマであり、クラスの半数以上が正解まで辿り着けない状況となった(前半成功、後半失敗)。

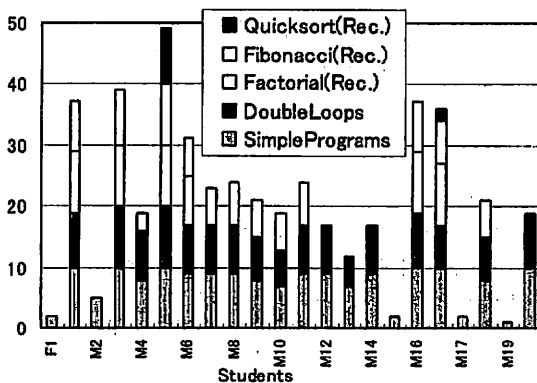


図10 VisuSim を使用した演習課題の評価グラフ

4.2 Web サーバとの連携による教材の適応化

ここでは、教材配布を行う Web サーバと連携することで、学習者の理解度に基づく個別化と適応化を実現する手順について述べる。

学習者の個別化および教材に適応性を持たせ、学習者に対応させること(適応化)は2節で述べた状態遷移モデルを用いる。自己申告や小テスト(プレテスト)などによって、学習者の理解度曲線を大づかみに把握し(パターン化し)、遷移系列のいずれかに対応させる(個別化)。これにより、学習者ごとに、推移行列が割り当てられ、VisuSim を教材として利用する際、ヘルプメッセージや選択される参照ファイル群の集合などが学習者の現状(状態遷移モデルの各ノードに対応)に基づき決定される(適応化)。

個別化は、学習者ごとの理解度を把握することが目的であり、理解度の高さに応じてより早く目標に到達する遷移系列(学習プロセスを決めるコースに対応)を割り当てることが教育効率の観点から望ましい。自己申告や小テスト結果により理解度の初期値を決定すると述べたが、VisuSim 自体を利用することも可能である。すなわち、動作に問題があるプログラムを複数本用意し、参照ファイル群に指定して、学習者に配布し、問題箇所を指摘させ、修正したプログラムを送信させる。教師は、受信した結果を確認し、学習者の理解度を予測し、状態遷移モデルの初期値を決定することも有効である。

次に、適応化の手順について述べる。教材 VisuSim は Java アプレットとして動作すると述べた。アプレットの場合、起動する HTML ファイルにパラメータを記載できる。Web サーバはアクセスしてきたユーザ識別により、推移行列を検索し、学習者の理解度に応じた(正確にはユーザに割り当てられた推移行列の要素に対応するデータベースより現在の状態を決定し)、適切なヘルプや参照ファイルを指定した HTML 記述をユーザに提供し、適応化した VisuSim を学習者ごとに起動させる。総て同じ操作であっても、ヘルプメッセージの内容や参照ファイルの種類やレベルが異なることになる。

現時点では、状態遷移は自動的に進まない。学習者が現在の状態に対応する課題をクリアした時点で教師が判断し(判断自体は機械的に可能)、学習者の状態を1つ進めるよう指示を送ることで、Web サーバが参照するデータベースの情報、すなわち推移行列の要素が更新される。その結果、学習者が次回アクセスする時点では、状態遷移が進み、ヘルプメッセージや参照ファイル(含む課題)も更新される。

5. おわりに

計算機アーキテクチャ教育を支援する教材の可視化機能の詳細を紹介した。学習者に対する個別化を図るため、状態遷移モデルの有効性を示し、定量的に扱うため推移行列の利用を提案した。また、教材の有する適応性を活用し、Web サーバと連携させる手順で、学習者の理解度に応じた教材提供を可能にする手法を述べた。これは、推移行列とデータベースを関連付ける手法で学習者の理解度を定量化し、教材 VisuSim に適切な情報をアクセスさせる方式であり、個別化・適応化を行う方法論の1つである。

今後は、提案した教材を活用した方法論が、具体的に教育効果を上昇させ、教育効率を改善しているかを示し、その有効性を示したい。また、VisuSim は既に Web 上で公開されているが、現時点で提供できる情報が多くない。これらを増強することでシステム全体の完成度・顧客満足度を高めたい。

謝辞：京都大学の富田眞治先生(大学院情報学研究所長)にはアーキテクチャ教育への示唆、VisuSim の初期版の開発支援など多くの援助をいただいた。香川大学の堀幸雄先生(総合情報基盤センター)には日頃の研究協力を願っている。記して感謝を表す。

参考文献

- [1] H.Liu and M.Yang, "QoL Guaranteed Adaptation and Personalization in E-Learning Systems," *IEEE Trans. Educ.*, Vol.48, No.4, pp.676-687, Nov. 2005
- [2] M.Grigoriadou, E.Kanidis, and A.Gogoulou, "A Web-based Educational Environment for Teaching the Computer Cache Memory," *IEEE Trans. Educ.*, Vol.49, No.1, pp.147-156, Feb. 2006
- [3] Y. Miura, K. Kaneko, and M. Nakagawa, "Development of an Educational Computer System Simulator Equipped with a Compilation Browser," *Proc. Int'l Conf. Computers in Education (ICCE2003)*, pp.140-143, 2003
- [4] A. Jutman, A. Sudnitson, R. Ubar, H-DWuttke, "E-learning Environment in the Area of Digital Microelectronics," *Proc. Int'l Conf. IT-based Higher Education & Training (IHHET2004)*, pp.28-33, 2004
- [5] Y.Imai, S.Tbmita, H.Niimi, and T.Kitamura, "Web-Based Computer Visual Simulator," *Technology Enhanced Learning (18th IFIP World Computer Congress)*, pp.111-120, 2004