

# 琉球大学学術リポジトリ

## 教材構造を利用したランダム探索授業スケジュール 作成手法

メタデータ	言語: 出版者: 琉球大学教育学部 公開日: 2007-08-08 キーワード (Ja): キーワード (En): 作成者: 仲間, 正浩, Nakama, Masahiro メールアドレス: 所属:
URL	<a href="http://hdl.handle.net/20.500.12000/1373">http://hdl.handle.net/20.500.12000/1373</a>

# 教材構造を利用したランダム探索 授業スケジュール作成手法

仲間正浩

## Random Searching Lesson Scheduler by using Learning Element Structure

Masahiro NAKAMA\*  
(Received October 31, 1995)

### Abstract

To make good lesson schedule, we would better to take understanding of students and overlap of teaching aids and circumstances of teachers into consideration. Better teaching schedule in a subject could be made by using learning element structure and by applying sorting algorithm to it<sup>1),2)</sup>. But we could not satisfy variety needs in this way.

To develop lesson scheduler system that satisfy variety needs, lesson schedule estimation functions which estimate variety needs and a system which make better schedule by using learning structure element and by applying random searching method are considered in this paper.

### 1. はじめに

現実の教育活動における授業スケジュールは、生徒の学習理解の向上、教室や教具等の共有資源の重複使用回避、教師の都合等を十分に考慮した上で作成するのが望ましいと考えられる。単一教科に限れば、矛盾のない正しい指導順序は、教材構造に整列アルゴリズムを適用することによって容易に求められる<sup>1),2)</sup>。しかし、この方法だけでは、最初に述べた様々な要求を満足することはできない。

本研究では、様々な要求を同時に満たす授業スケジュールを自動的に生成するシステムを構築する試みを行っている。これまでの検討によって、①授業スケジュールに対する様々な要求を（授業スケジュールに対する）評価関数として定式化し、②この関数およびランダム探索法を用いることに

よってより良い授業スケジュールを導き出すシステムを構築したのでその内容について報告する。

### 2. 良い授業スケジュールを作成するための要件

教師が新しい単元の指導をする際にはそれを理解するために必要な単元をあらかじめ指導しておかなければならない。つまり、授業の準備段階において矛盾のない正しい指導順序を形成する必要がある。このような学習指導順序は一つの教科の狭い範囲に限定すれば、教材構造に整列アルゴリズムを適用することにより容易に求められる。しかし、現実の学校で行われる授業内容は、それぞれの教科が完全に独立しているのではなく、お互いに依存しあっている部分がかかなりあるはずである。このようなことを前提に考えれば全ての教科

---

\*Department of Computer Science, College of Education, University of the Ryukyus, Nishihara, Okinawa 903-01, Japan.

を包括的に捉えて全ての教師が連携しあってカリキュム運営を行うことが理想である。しかし、実際にこれを実行するには、自分の教科専門は言うまでもなく専門外の科目の内容も熟知し、多くの時間を費やして授業スケジュールを組み立てておく必要がある。また、これを、実行可能なスケジュールにするためには学校内にある共有資源の使用計画が重複しないようにしたり、一人の教師が同時に複数クラスの授業を受け持つような計画を避ける等の考慮が十分になされていなければならない。従って、上記のような理想的な授業スケジュールを現実のものとする為には、多くの教師がかなりの困難を強いられることになるであろう。

本研究では上記のような様々な要件を同時に満たす授業スケジュールをコンピュータによって自動的に生成するシステムの構築を試みている。このシステムは、教師の時間を犠牲にすることなく効率的なカリキュラム運営を行うための助けになると考える。

## 2.1. 生徒の理解を考慮した授業スケジュール

良い授業スケジュールを作成するためにまず考えなければならないのは、如何に生徒にとってわかりやすいものとするかであろう。以下では、この視点に立った良い授業スケジュールの要件を述べる。

### 2.1.1. 教材構造を基に得られる矛盾のない指導順序

生徒の理解を考慮して授業スケジュールを組み立てる際に最も重要なのは矛盾のない正しい指導順序が得られることであろう。矛盾のない正しい指導順序は、あらかじめ指導する範囲の教材構造を求めておき、それにソーティングアルゴリズムを施すことによって容易に得られる。しかし、この方法では一つの教材構造から複数の可能な指導順序を導き出すのは困難である。通常、一つの教材構造からは、矛盾のない正しい指導順序が複数求められる可能性がある。例えば、分数の足し算の指導を対象とした図2-1-aの様な教材構造

を基にすれば図2-1-1 bの様な複数の指導順序が導き出せる。従って、よりわかりやすい授業スケジュールを作成するためには上記の複数の指導順序から最適なものを選び出す必要があるであろう。

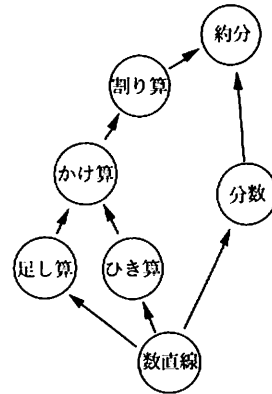


図2-1-1 a 単一教科内の教材構造

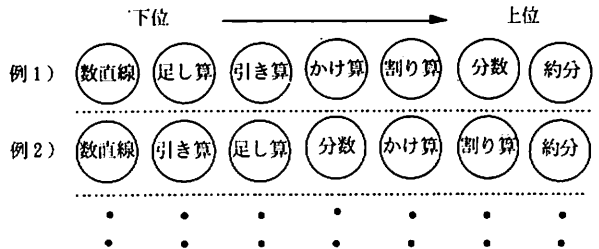


図2-1-1 b 一つの教材構造から導き出される複数の授業スケジュール

### 2.1.2. 複数教科の単元間でも矛盾を生み出さないスケジュール

単元間の順序関係は単一教科内にとどまらず複数教科にも存在する。従って、単一教科のみを考慮してスケジュールを作成すれば、異なる教科間で指導順序の矛盾が出てくる可能性がある(図2-1-2 a, 図2-1-2 b)。よって、授業スケジュールを作成する際には単一教科だけではなく全ての教科を統括した教材構造を基準に作成するのが望ましいと考える。

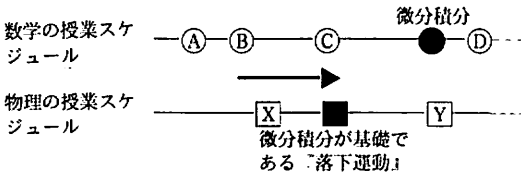
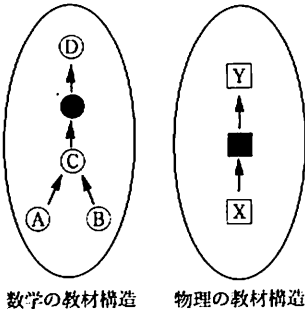


図 2-1-2 a 複数教科での授業順序

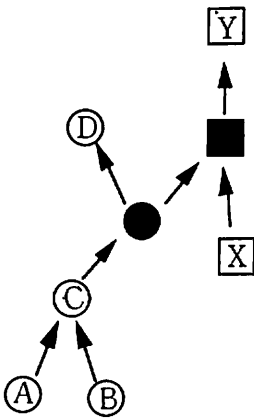


図 2-1-2 b 複数教科を考慮した教材構造

2.1.3. 教材構造中で隣り合った単元間の時間的距離を短くする

人間は、時間が経つにつれて学習したものを忘れてしまう。従って、たとえ矛盾のない正しい授業スケジュールが提供されたとしても教材構造中で隣り合った単元間の時間間隔を極端に長くしてしまうと、生徒にとっては、理解しがたいものとなることが考えられる。よって、スケジュールを作

成するにあたっては教材構造中矢印で直接結ばれた単元間の実際のスケジュール上での時間間隔を考慮する必要があると考える。例えば、図 2-1-3 の様な教材構造が存在し、一つの単元を指導するのに要する時間が同一であるとするならば [E A B C D F] というスケジュールではなく [A B C D E F] というスケジュールで指導するのが好ましいであろう。

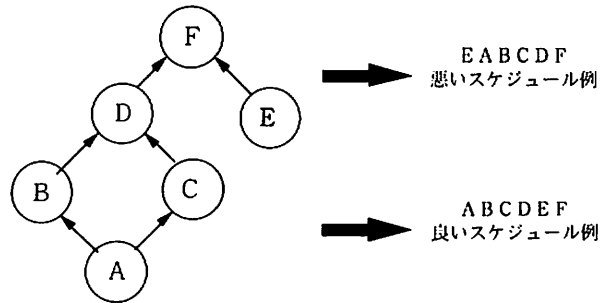


図 2-1-3 一つの教材構造から得られる良いスケジュールと悪いスケジュールの例

2.1.4. 授業スケジュールの時間割への当てはめ

実際問題として、並べられた単元列は時間割上に挿入しなければならない。図 2-1-4 のよう

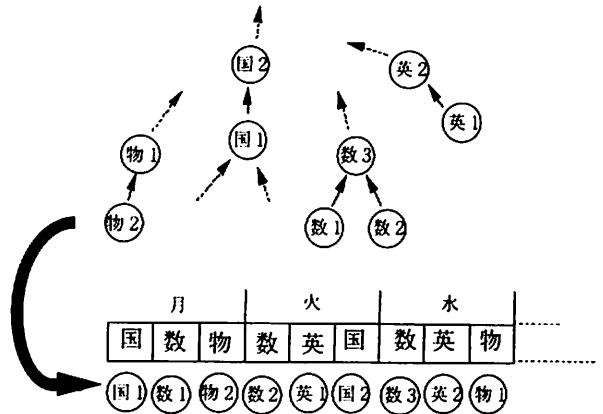


図 2-1-4 複数教科を対象とした教材構造から時間割への当てはめ

に、教材構造中の各単元を並べる場合には、その構造に従うばかりでなく、時間割中の指定科目時間中に挿入しなければならない。

## 2.2. 学校内にある共有資源の重複使用の回避

ある学校に OHP が 2 台しかなければ、OHP を利用した授業を同時間に 3 つ以上開講することは不可能である。従って、有限の共有資源を利用する際には、それらの利用時間帯をずらしておく必要がある。このような共有資源としては、各種教育機器、各種教室等が挙げられる。

## 2.3. 教師の記憶時間を考慮したスケジュールの作成

一人の教師が、同一の授業科目を複数のクラスに対して指導する形態は、しばしば見られるものであろう。このような授業形態において考慮すべき最も重要なことは一人の教師が同時に二つ以上のクラスで授業することができないことである。次に考慮すべきは、同一の単元を異なるクラスで指導する場合その時間間隔をあまり開けないことであると考えられる。つまり、教師があるクラスで行った同一単元の授業の教授内容、教授方法を忘れないうちにできるだけ早く別のクラスで指導できるようなスケジュールを作成すべきである。

## 2.4. 授業スケジュールに対するその他の要件

授業スケジュールを作成する際、上記以外に考えるべきことを列挙すると以下の通り

- ・複数の学年を考慮した授業スケジュール
- ・学校行事
- ・教師の休み

になるであろう。

本研究の現在の主な目的は、上記の様な複雑な要求に応えることのできる授業スケジュールの実現の可能性を探ることであると考え、また、問題を簡単にするために以下、

- ・単一の学年だけを考える

- ・教材構造中の一単元を指導するのに要する時間を一時限と仮定する

の様な制約条件の基で検討を行ってきた。以下では、この制約条件の基で開発した授業スケジュールの構築方法について説明する。

## 3. 授業スケジュールシステムの概要

本研究で制作している授業スケジュールシステムの概要を図 3 に示す。この授業スケジュールは、以下のデータ、

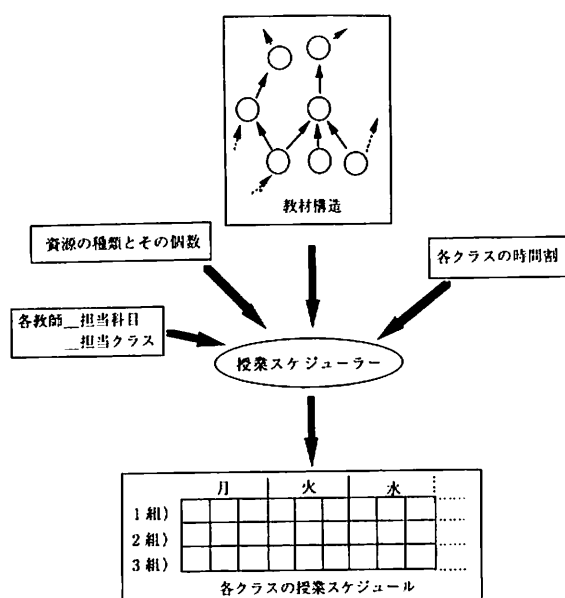


図 3 授業スケジュールシステムの概要

- ・各クラスの時間割、
- ・全ての授業科目の教材構造
- ・各教師の担当科目、担当クラス
- ・学校内にある共有資源の種類とその個数

を入力として受け取り、これらの情報を基にしてより良い授業スケジュールを作り出す。

このスケジュールが良い授業スケジュールを作り出す方法は、教材構造中の各単元を各クラスの時間割に無秩序に挿入する操作によって最初の授業スケジュールを作り出し、そのスケジュールを

出発点としてそれに少しずつ変化を加えると同時に下記

- ・教材構造のルールに従ってならんでいるか？
- ・関連ある単元間の時間的距離は短いか？
- ・そのクラスが指定している科目の時間帯にあてはまっているか？
- ・一人の教師が同時に複数のクラスを担当していないか？
- ・共有資源の重複使用はないか？
- ・教師が複数のクラスで行う同一単元の授業間の時間間隔は十分短いか？

の通りの評価基準を基により良いスケジュールを探索していき、利用者が処理の中断を命じたときにこれまでに探索した中で最も良いスケジュールを出力する。

### 3.1. ランダム探索による一教科一クラス授業スケジュールの導出

様々な要求に応えられる授業スケジュールを容易に求める為に、本研究では、ランダム探索を用いることにした。この方法を図3-1 aに示す。このアルゴリズムでは、以下、

- ①最初に、教材構造中の全単元を無秩序に並べ、これを、古い単元列とし、
- ②この古い単元列からランダムに二つの単元を選び出し、それらの位置を交換して新たな単元列を生成する。この単元列を新しい単元列とする。
- ③授業スケジュールの善し悪しを判断する評価関数で二つの単元列を比較し、評価の良い方の単元を古い単元列とする。
- ④ ②、③を繰り返す。

の手順でスケジュールを作り出す。

本稿で述べる授業スケジュールは図3-1 bに示すように、図3-1 aで示したランダム探索アルゴリズムにこれから述べる評価関数を追加していき、それぞれの評価関数の総和を最終的な評価値とすることで構成する。この方法によって、教材構造ばかりでなく時間割、共有資源、時間間隔等を全て考慮に入れた複雑な要求に答えられるようなスケジュールをかなり簡単な操作によって導き出すことができるようになる。

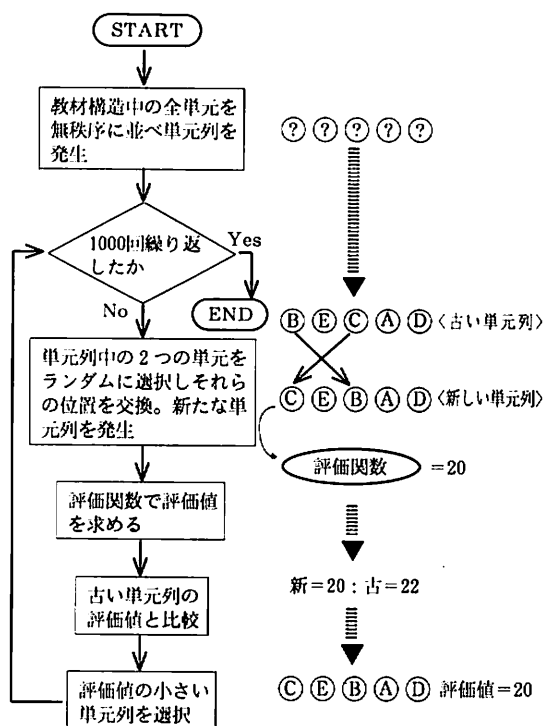


図3-1 a ランダム探索による授業スケジュールの導出

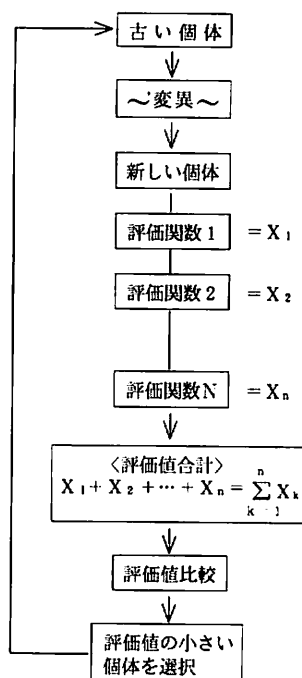
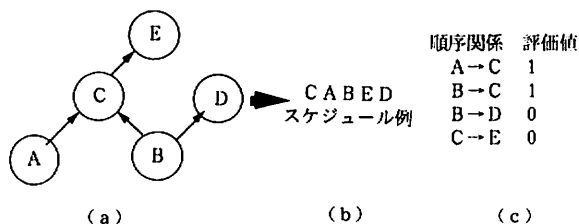


図3-1 b 複数の評価関数を用いた授業スケジュールの導出

これから述べる評価関数のある授業スケジュールに適用した場合、その関数によって得られる評価値の低い方を良いスケジュールとし、評価値の高い方を悪いスケジュールとする。つまり、悪いスケジュールに対してペナルティを課していくという考え方で評価関数を構成していく。



### 3.2. 単一教科単一クラス授業スケジュール作成のための評価関数

#### 3.2.1. 授業スケジュール中の指導順序の矛盾を取り除く為の評価関数

授業スケジュールが生徒にとって理解しやすいものであるためには、教材構造を基に判断して指導順序の矛盾がない方が望ましい。これを達成するために以下の評価関数を用意した。

〈〈評価関数 1：授業スケジュール中の指導順序の矛盾を取り除く〉〉

この評価関数では、個々の順序関係が与えられた単元列中で正しい順序になっているかどうかを教材構造中の順序関係に従って一組ずつ検査していく。順序が逆転している場合には評価値にペナルティを追加し、正しければ評価値にペナルティは加えない。全ての順序関係について検査し終えた時点での評価値をこの評価関数で与えられる評価値とする。

つまり、この評価関数で与えられる評価値を

評価関数 1 で与えられる評価値  
= 授業スケジュール中での順序関係の矛盾の総数

の通りとする。

例えば、図 3-2-1 (a) の様な教材構造があれば、この教材構造中に存在する順序関係は (A-C, B-C, B-D, C-E) の通りとなる。図 3-2-1 (b) の単元列の評価値 = 1 + 1 + 0 + 0 = 2 (図 3-2-1 (a)) となる。

図 3-2-1 単一教科の教材構造を基に順序の正誤を評価

#### 3.2.2. 各単元の授業時間をその教科の時間割に合致させる為の評価関数

複数教科の単元から構成される教材構造の各単元をその順序関係に従って並べ、時間割上の該当する教科の時間帯に当てはめなければならない。この条件を満足させる為に次のような評価関数を追加した。

〈〈評価関数 2：時間割上の教科の合致〉〉

本研究では、時間割上の指定された教科の欄に授業時間を直接割り当てるのではなく、一週間に割り当てられた時間数を時間割上の任意の場所に割り当てる方法を採用した。これは、時間割に強く制約されない柔軟な授業スケジュールを指向しているためである。

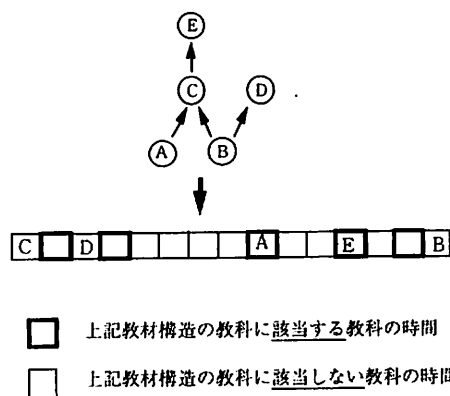


図 3-2-2 単一教科教材構造中の単元を時間割へ割り当てる

各単元が時間割上の指定する教科の時間に当てはまっているかどうかを調べ、当てはまっていれば評価値に0を加え、当てはまっていなければ評価値に1を加える。例えば、図3-2-2上の教材構造を図3-2-2下の様に時間割に当てはめた場合、

- ・単元Cはその時間割に当てはまっていない→部分評価値=1
- ・単元Dもその時間割に当てはまっていない→部分評価値=1
- ・単元Aは、時間割に当てはまっている→部分評価値=0

の様に部分評価値を求めていき、以下同様にして単元E、Bの部分評価値を求め、全ての部分評価値を加えたのがそのスケジュールの評価値である。

### 3.2.3. 教材構造中で隣接した単元間の時間の短縮の為の評価関数

関連性が強い内容の授業はなるべく時間を開けない様にスケジュールリングするために以下の評価関数を用意した。

〈〈評価関数3：教材構造中隣接する単元間の時間間隔を短くする〉〉

教材構造中順序関係で直接関係づけられている(矢印で直接結ばれている)それぞれの2つの単元が授業スケジュール上でどれだけの間隔を持っているかを調べ、全ての順序関係について求めその総和を求める。

例えば、図3-2-2上のA~Dの教材構造が同図中下の様にスケジュールされた場合このスケジュールの評価値は次の様に求める。

- ・矢印で直接結ばれた単元A Cの授業スケジュール上での距離は8
- ・矢印で直接結ばれた単元B Cの授業スケジュール上での距離は14

以下同様にして、BD=12, CER=11を求め、これらの総和8+14+12+11=45を求める。これが評価関数3の評価値となる。

### 3.3. 複数教科単一クラス授業スケジュールの為の評価関数

この場合、教材構造は、複数教科の単元を含み、また、時間割には複数教科の時間が指定されるものとする。つまり、上記の評価関数3を単一ではなく複数教科に当てはめることが必要である。この場合の処理手続きは、例えば図3-3(a)の教材構造を図3-3(c)の様に時間割に当てはめた場合、

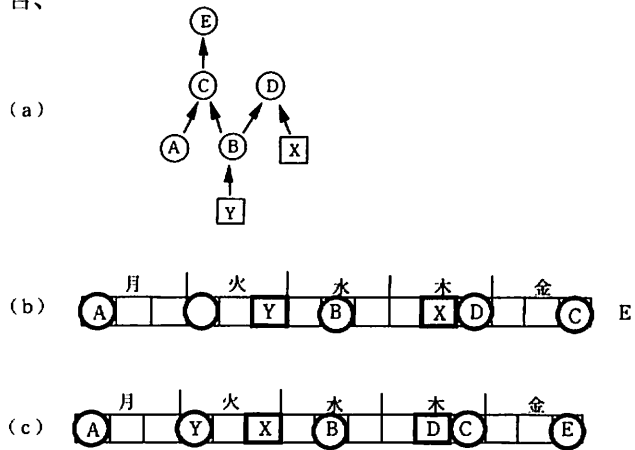


図3-3 複数教科教材構造の時間割への当てはめ

- ・単元Dは、その該当する教科の時間(丸印)に当てはまっていない。よって部分評価値=1
  - ・単元Xは、その該当する教科の時間(四角印)に当てはまっている。よって部分評価値=0
- 以下同様にして、各単元の部分評価値を求め、全ての部分評価値の総和を求めたものがこの評価関数で与えられる評価値となる。ここでは評価値=2となる。

#### ※空き時間の問題

図3-3(a)の様な教材構造を図3-3(b)のように時間割へ割り当てる場合、該当する科目を合致させたどのような授業スケジュールを考慮しても全ての時間を埋められない。これに対処するためには次の2つの方法、

- ・空き時間は、復習や豆テスト、または独立した単元の学習時間を割り当てる(図3-3(b))。
- ・指定された教科科目を無視して挿入可能な他



の教科を割り当てる。本来指定されているのは'○'教科であるが、'□'教科の授業を行う(図3-3(c))。

が考えられるが、本研究では、授業に要する総時間が短くなる後者の方を選択している。

### 3.4. 複数教科-複数クラスの場合の授業スケジュールの作成

単一クラスの場合は授業スケジュールを1次元で取り扱っていたが、複数クラスに対処するには授業スケジュールを図3-4-1のように2次元のデータとして扱う。

	月	火	水	木	金
1組)	B A A B	B		B A	
2組)	A	B A		B A	A B
3組)	B	B	A A	B A B	
		*		*	*

図3-4-1 複数教科複数クラスの時間割

この場合の授業スケジュールに対する評価は複数教科の単元からなる教材構造を上で述べた単一クラスの場合と同じ評価関数を全てのクラスに対して適用し、これによって得られる評価値の総和を求めて総合的な評価値を得る。

複数教科-複数クラスの場合に授業スケジュールを作成する際に利用するデータは、複数教科の単元を含んだ教材構造、複数クラスの時間割、複数の教師、共有資源となる。

#### 3.4.1. 教師が同時に重複してクラスを受け持つことを回避する評価関数

空き時間が生じた場合に授業交換を行った場合に一人の教師が同じ時間に複数のクラスを受け持たないようにする為に次の評価関数を用意した。

<<評価関数4：一人の教師が複数クラスを同時に重複して受け持つのを防ぐ>>

本研究で作成する授業スケジュールシステムは、まず、教材構造中の各単元を全てのクラスの時間割に割り当てていき、次いで、各授業に対し

て担当教師を割り当てる。その後評価関数4によって同一教師の複数クラスでの授業担当の重複を検査していく。

この評価関数は、それぞれの教師が同時間に複数のクラスで重複して授業が行われるように計画されていた場合、もし、2クラスで重なってれば1を、3クラスで重なってれば2をNクラスで重なってればN-1を評価関数に加える。

例えば、各クラスに対して図3-4-1のような教師の割り当てが与られているとすると。

- A教師については、火曜日の1時間目と木曜日の3時間目に2つのクラスを受け持つことになっているので、部分評価値=1(火の1)+1(木の3)=2
- B教師については、木曜日の1時間目に3つ重なっているため、部分評価値=2

以下同様にして全ての教師について重なり具合を調べ、部分評価値を求め、それらの総和をこの評価関数の評価値とする。

#### 3.4.2. 共有資源の重複使用を回避する為の評価関数

複数クラスの授業スケジュールを作成すると限りある共有資源が重複して使用される計画が発生させられる可能性がある。従って、それを避けるための評価関数が必要になる。

<<評価関数5：共有資源重複使用の回避>>

この評価関数で行う処理手続きは

図3-4-2中の英大文字は資源の種類を表し、その文字の下にならんでいる数字は、各クラスで該当する資源を使用する個数を表している。この例では、月曜日の1時間目にAの資源が各クラスで使用されている個数は、0+1+2=3

	月の1			月の2			
	A	B	C	?	A	B	C
1組)	0	0	1				
2組)	1	1	0				
3組)	2	0	1				

図3-4-2 共有資源の授業スケジュールへの割り当て

(個)である。仮に資源Aが2個資源Bが1個資源Cが1個あるとすると。この評価関数で与えられる評価値は以下の手続きで与えられる。

- ・月曜日の1時間目Aの資源の使用総数は、2組の1個+3組の2個=3個。これは、最大個数の2個を1個オーバーしているので部分評価値=1
- ・月曜日の1時間目Bの資源は、2組の1個だけである。これはその限度個数以下なので、部分評価値=0。

以下同様にして全ての共有資源について重複使用を調べていき。全ての部分評価値の総数をこの評価関数の評価値とする。

### 3.4.3. 教師が同一単元の授業を短期間の間に行う為の評価関数

同一単元の授業が複数のクラスで予定されている場合、最初に行う授業と最後に行う授業の時間間隔が長くなれば、一度行った授業内容を忘れてしまい、復習する無駄を生じる恐れがある。このような無駄を少なくするためにこの評価関数を設定することにした。

〈〈評価関数6：教師が行う同一内容の授業を短期間の間に行う〉〉

その手続きは、一人の教師が同一内容の単元の授業を最初に行ってから、最後に行うまでの時間間隔を全ての単元について求めこれを部分評価値とする。全ての教師の授業に対してそれぞれ部分評価値を求め、その総和を評価値とする。

例えば、図3-4-3の様な授業スケジュールでは、A教師が、単元1の授業を最初に月曜日の2時間目に行い、最後に水曜日の2時間目に行うことになっているので部分評価値=7(スケ

	月	火	水	木	金
1組)	A1		A2		A3
2組)		A2	A1 A3		
3組)		A1	A2		A3

図3-4-3 一教師が複数クラスを受け持つ授業スケジュール

ジュール上でのコマ数の差)になる。以下同様にして教授単元2、3について部分評価値を求め、これらの部分評価値の総和を求める。この手続きを全ての教師に対して行い、全ての部分評価値の総和を求めたものがこの評価関数の評価値となる。

### 3.5. それぞれの評価関数の重みづけ

本研究で試作した授業スケジュールシステムは、上記に示したランダム探索アルゴリズムに加え、これまで示してきた全ての評価関数の総和を最終的な評価関数として使用して授業スケジュールを作り出す。このスケジュールの性質は評価関数の与え方に依存することになるが、実際にどのようなスケジュールが求められるかは、このスケジュールを使用する側の都合によって決定されるべきであり、一意に決められるものではないと考えられる。従って、このシステムの利用者の要求に柔軟に応えられるようにするためにそれぞれの評価関数によって得られる評価値をそのまま加えるのではなく、最終的な評価値は、以下の式

$$\begin{aligned}
 \text{最終的な評価値} = & \text{評価関数1} * p_1 \\
 & - \text{評価関数2} * p_2 \\
 & - \text{評価関数3} * p_3 \\
 & \dots \\
 & \dots \\
 & + \text{評価関数n} * p_n
 \end{aligned}$$

で与えられるようにした。ここで、 $p_1 - p_n$ は、利用者が好ましいと考える数値を適当に代入するものと想定している。

### 3.6. 局所的な解へ陥ることを回避するための探索点接近法

これまでに述べた探索法では、授業スケジュールに少しの変異を与え、徐々に良い評価値へ近づけていく方法を採用していた。しかし、この方法では、局所的な解に陥りそこから抜け出せなくなることがしばしば起こる。従って、本研究では、複数回変異を与え評価値を求めることを繰り返

し、評価値が局所解へ達した（一定回数の試行を経過しても評価値の改善が見られない）と判断できた場合に全てのクラスのスケジュールに変異を与える（これまでは、一つのクラスを選択して変異を与えていた）大きな変異を与えることにより局所解へ陥ることを回避する方法を採用した。

本研究ではこの探索法を利用し、これまで述べた全ての評価関数を用いた授業スケジュールプログラムを作成した。このプログラムを用いて計算機上でスケジューリングを実行し大きな変異を3回与えた場合の結果例を図3-6に示す。この時の各評価関数の重み付けは、 $p_1 = 500$ ,  $p_2 = 1$ ,  $p_3 = 50$ ,  $p_4 = 50$ ,  $p_5 = 200$ ,  $p_6 = 1$ である。このグラフより、大きな変異を与える前（グラフ上A点）よりも大きな変異を数回与えた場合（グラフ上C点）の方が良い結果が得られているのがわかる。

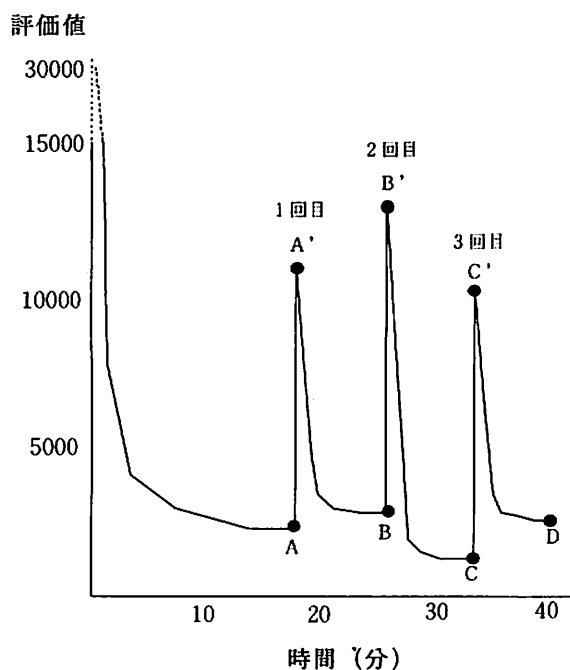


図3-6 探索点変動と評価値の推移

#### 4. まとめ

本稿では、ランダム探索法を利用した授業スケジュールと、この探索で使用する各評価関数の構成法について述べた。ここで述べたスケジュールは、実現の可能性を確認することを意図してきたものであるため、必ずしも利用者の要求に十分にえられるものではないであろう。しかし、これまでの検討によって、様々な要求に柔軟に応えられるスケジュールを作成するシステムが現実に構築できることが確認できたものとする。

今後、実際にこのようなシステムを簡単に利用できるようにするためには、更に次のような事柄

- ① 全ての教科に対する教材構造を容易に構築するためのシステムの実現
- ② 現場の様々な要求に応えられるような新たな評価関数の追加
- ③ より高速にかつ広範囲にわたる探索を可能にするための新たな探索手法の開発
- ④ 実際の教育現場に容易に導入できるようにするための戦略の検討

等を検討していく必要があると考える。

#### 謝辞

本研究の検討に熱心に協力された平成6年度卒業生の田本智幸君に感謝いたします。

#### 参考文献

- 1) 佐藤 隆博：“教育情報工学入門”、コロナ社、1989.
- 2) 米澤 宜義：“教材構造と学習履歴に基づく復習課題系列の生成”、電子情報通信学会論文誌A、Vol.J75-A No 2、pp.382-389、1992
- 3) 赤堀 侃司：“あいまいな教授方略を反映した学習課題系列の生成”、電子情報通信学会論文誌A、Vol.J75-A No 2、pp.362-370、1992