特別研究報告書

コースツリーを考慮した
最適な履修計画の作成

指導教員　　　山下信雄　教授

京都大学工学部情報学科
数理工学コース
平成 22 年 4 月入学

竹村　健太朗

平成 27 年 1 月 30 日提出

摘要

組合せ最適化は, 様々な現実問題の意思決定を行うことに役立つ. そのため, 学校運営においても, クラス編成, スクールバスの運用などで用いられている. しかし, 学生の立場に焦点を当てた事例は少ない. そこで, 本報告書では学生主体の問題として, 履修計画を組合せ最適化問題を用いて解くことを考える.

履修計画を立てることは, 大学生活の学業に関して重要な事柄のひとつである. 特に, 単位修得につまづいてしまった学生は, 自分でしっかりと考えて履修計画を立てる必要がある. 何も考えないで計画を立てると, 高学年の科目を先に履修したり, 科目を過分に登録したりして, 単位修得がますます困難になってしまうことがある. そのような事態を防ぐための対策として, 各科目の履修順序を示したコースツリー, 履修科目の登録数を制限するキャップ制がある.

本報告書では, コースツリーとキャップ制を考慮して, 最適な履修計画を立てる組合せ最適化モデルを提案する. また, 数理工学コースが提供している実際の履修要覧, 時間割, コースツリーのデータを用いた数値実験を行うことによって, 提案した最適化モデルによって, コースツリーを考慮した履修計画が作成できる. さらに, 数理工学コースの実際の時間割はよく構成されていたため, 特殊な履修状況を除いて, コースツリー通りに無理なく履修することができることも確認できた.

目次

1 序論

数理最適化問題は, 様々な現実の問題を解決するのに利用されている．組合せ最適化は, 制約条件を満たす組合せの中で最も良いものを見つけ出す数理最適化問題である．. 組合せ最適化問題の代表的な応用例として, 最短路問題, 割当問題, クラス編成問題, ナップサック問題, スケジューリング問題, 巡回セールスマン問題などが挙げられる [6].

最近では, 学校の運営においても , 組合せ最適化モデルを用いた意思決定が行われている. 例えば, 新入生のクラス編成 [5] やスクールバスの配置とルート決めを行うプロジェクト [4] などが挙げられる．しかし, これらの事例のほとんどは, 学校運営および教員の観点による意思決定であり, 個々の学生に関する問題を取り扱っているものは少ない．そこで, 本報告書では, 学生の立場を主体とした問題を取り扱うことに焦点を当て, 履修計画を題材とした組合せ最適化問題について考える.

履修計画を立てることは, 大学生活の学業に関して重要な事柄のひとつである．毎年, 大学より提示された理想の履修計画に従って単位修得できれば問題はない．しかし, そのような計画通りの単位修得につまづいてしまった場合, それ以降の履修計画を立てる際に, 科目を過分に登録しすぎて無理な計画になってしまったり, 履修すべき科目の優先順位を誤ってしまったりすることが多い．その結果, 単位修得がますます困難になってしまい, 最悪の場合には多年度にわたり留年してしまうという事態にもなりかねない.

京都大学情報学科数理コースでは, 履修計画をたてるための助けとなるように, 各科目の履修順序についての相関関係を示したコースツリー案 (付録Ａの図 1) を学生に提供している. さらに, これまでの学生の履修状況に関する統計データから, 履修科目の登録数が多くなると, 成績がかえって下がってしまうということが示されている．そのデータに基づいて, 京都大学の多くの学部において, 履修科目数を制限するキャップ制が導入され始めている.

本報告書では, このコースツリー案およびキャップ制を導入して, 学生の現状の学年と不足単位数に応じて最適な履修計画を作成する数理モデルを提案する．さらに, 2014 年度の京都大学のシラバス [9] および履修要覧 [11] をデータとして用いて, 情報学科数理コースにおける履修計画を立てる.

本報告書の構成を以下に示す．2 節では本報告書で必要となる基礎事項を述べる．3 節では提案するコースツリーとキャップ制を考慮した組合せ最適化モデルを提案する．4 節では実際の数理工学コースのデータを用いた提案モデルのシミュレーションを行い, それについて考察する．最後に, 5 節ではまとめと今後の課題を述べる.

2　準備

この節では本報告書の基礎となる事項を述べる.

2.1　履修要件

平成 26 年度の京都大学情報学科数理工学コースの履修要項 [11] の内容について, 簡単に説明する.

工学部においては, 履修科目が次の表１のように大別されている [10].

表 1　履修科目の種類

人文・社会科学系科目群 (以下, 人社)	哲学・思想系科目, 歴史・文明系科目, 芸術・言語文化系科目, 行動科学系科目, 地域・文化系科目, 社会科学系科目
自然・応用科学系科目群 (以下, 自然)	数学科目, 物理学科目, 化学科目, 生物学科目, 地学科目, 情報科目
外国語科目群 (以下, 語学)	英語および初修外国語
現代社会適応科目群 (以下, 現社)	情報系科目, 健康科学系科目, 環境系科目, 法・倫理コンプライアンス系科目
拡大科目群 (以下, 拡大)	スポーツ実習科目, 少人数教育科目, カルチャー一般科目, キャリア支援科目, 国際交流科目, 地域交流・貢献科目, 単位互換等科目
専門科目群 (以下, 専門)	工学部学生限定の科目

工学部情報学科では, 一回生, 三回生, 四回生の終了時に行われる, コース配属, 分野配属, 卒業という進級に関わる制限があり, それぞれ以下の通りに指定された単位が必要となる.

1. コース配属
   自然・専門科目の一回生配当科目のうち指定された科目の中から 15 単位以上 (自然 22 単位, 専門 8 単位)

2. 分野配属
   次の表２の科目および単位数

3. 卒業
   次の表３の科目および単位数

表 2　分野配属要件

人社	12 単位以上
自然	28 単位以上
語学	10 単位以上
現社	2 単位以上 (情報系科目は 8 単位まで)
専門	48 単位以上 (含, 選択必修 10 単位以上)
人社・現社・拡大	合計 17 単位以上 (22 単位まで)
総合計	108 単位以上

表 3　卒業要件

人社	12 単位以上
自然	28 単位以上
語学	10 単位以上
現社	2 単位以上 (情報系科目は 8 単位まで)
専門	66 単位以上 (含, 必修 5 単位, 選択必修 10 単位以上)
人社・現社・拡大	合計 17 単位以上 (22 単位まで)
総合計	134 単位以上

ただし, 表２と表３の専門科目に含まれる選択必修科目とは, 数理工学実験, 基礎数理演習, プログラミング演習, 数値計算演習, システム工学実験のことである. また, 必修科目とは, 卒業研究のことであり, 時間割への制約はない.

ここで, 上記の必要単位数の条件をふまえたうえで, 各要件を満たすことを目的として履修計画を立てることの重要度の違いについて述べる.

まず, コース配属されなければ, 基本的には二回生以上の科目を履修することはできない. また, コース配属前の履修計画では, クラス指定の一回生科目で時間割のほとんどが埋まってしまう. そのため, コース配属されるまでの履修計画はほぼ自動的に決まってしまう. よって, この期間の履修計画を立てることはあまり重要ではない.

また, 卒業要件は, 分野配属要件を満たしたうえでのものである. 最低の基準で分野配属された場合でも, その時点で必要な残りの単位数は, 必修科目 5 単位, 専門科目 13 単位を含んだ合計 26 単位となる. これを残りの一年で修得することになるが, 一日あたり一科目程度の履修計画で事足りるので, これはそれほど難しいことではない. よって, この期間の履修計画を立てることもあまり重要ではない.

したがって, コース配属されてから分野配属されるまでの履修計画を立てることが学生にとっての最も重要な事案となる.

2.2 数理最適化問題

数理最適化問題は, 以下のように表される.

$$\begin{array}{ll} \max & f(x) \\ \text{s. t.} & x \in F \end{array}$$

ただし, 決定変数 x は n 次元実ベクトル, 目的関数は n 次元実ベクトル空間上で定義された実数値関数である. さらに, $F \subseteq \mathbb{R}^n$ は実行可能領域である.

ここで, 個々の解 $x \in F$ を実行可能解と呼ぶ. f を最大にする実行可能解を最適解と呼ぶ.

数理最適化問題の中でも, f が線形な関数で, 実行可能領域 F が凸多面体と整数の集合の共通集合として表されているとき, 混合整数計画問題という. さらに, 整数のとりうる値が 0 か 1 の 2 値に限定されているものを混合 0-1 計画問題という. 何かを行うかどうかという決定を含んだ問題を定式化する際に, 0-1 の 2 値の値をとる変数が用いられる [3].

連続値をとる変数を含まない整数計画問題は, すべての実行可能解の組合せを列挙して, 目的関数の値を比較することで解くことができる. しかし, すべての実行可能解の組合せを列挙しようとすると, 計算時間が膨大なものとなってしまい, 解くことができない. このような問題をＮＰ困難であるという [1].

ＮＰ困難な問題は, すべての実行可能解を列挙する方法で解くことはできない. そこで, 効率的に解くためのいろいろなテクニックが研究されている. その中でも代表的な解法として分枝限定法が挙げられる. 分枝限定法とは, 問題を分割して場合分けを繰り返すことで, 最適解を見つける手法である. 一般に, 分枝限定法を用いれば変数が 20 から 100 程度の問題は解くことができる. なお, 現在では様々な分枝限定法の工夫が提案されていて, 変数が 10000 個にもおよぶ問題を１時間程度で解くこともできるようになっている [7].

3　コースツリーを考慮した履修計画の最適化モデル

この節では, 本報告書で提案する数理モデルに関する詳細について述べる.

3.1　問題の基本設定とデータ

第 2.1 節にて述べたように, コース配属されてから分野配属されるまでの履修計画を立てることが学生にとっての最も重要な事案である. よって, 本報告書では, この期間の履修計画を立てる数理モデルを考案する.

また, 簡単のため, 自然・専門の科目についてのみ考え, その他の科目は全て単位を修得できているものとする. 仮にそれらの科目を追加で導入する場合は, それらの科目の研究室配属要件にかかわる制約条件を加えるだけで問題の仕様の変更は容易に行える.

作成する履修計画は, 前期から始まる数年単位のものとする. 後期から始まる計画も簡単に拡張できるが, ここでは説明を簡単にするため省略する.

問題を解くためのデータとして, 以下のものを用いる.

- 科目の数：n
- 科目の集合：C= {1, 2, ..., n}
  - 自然科目の集合：C_1
  - 専門科目の集合：C_2
  - 選択必修科目の集合：C_3
- 分野配属に必要な単位数
  - 自然科目の必要単位数：p_1
  - 専門科目の必要単位数：p_2
  - 選択必修科目の必要単位数：p_3
- コースツリーのグラフ：$G(C, E)$
  ただし,$(a, b) \in E$ であるとき, a はコースツリーの後科目, b はコースツリーの前科目とする. また, コースツリーの枝の集合 E を以下の二つの集合 E_1 と E_2 に分割する.
  - 後科目 a と前科目 b が前期と後期で分かれている枝の集合：E_1
  - 後科目 a と前科目 b が前期と後期で分かれていない枝の集合：E_2
- 科目 $j \in C$ の単位数：m_j
- 科目 $j \in C$ の前科目の数：d_j

ここまでのデータは, 全ての学生に共通のものとなる. 一方, 次のデータは個々の学生の状況に依存するものとなる.

- 履修計画の年数：$\bar{y}$
- 履修計画年数の集合：$Y = \{1,...,\bar{y}\}$
- 修得済単位数
  - 自然科目の修得済単位数：q_1
  - 専門科目の修得済単位数：q_2
  - 選択必修科目の修得済単位数：q_3
- 科目 $j \in C$ の単位修得確率：r_j

3.2 履修計画の最適化モデル

3.2.1 基本モデル

この節では, 履修計画の最適化モデルに用いる決定変数, 目的関数, 制約条件について説明する.

1. 決定変数
   最適化モデルの決定変数として, 次のような x_{ij} を導入する. ここで, $i \in Y$ は計画年, $j \in C$ は科目を表している.

$$x_{i,j} = \begin{cases} 1 & (i \text{ 年に科目 } j \text{ を履修する}) \\ 0 & (i \text{ 年に科目 } j \text{ を履修しない}) \end{cases}$$

   この決定変数 $x_{i,j}$ により, どの科目を履修するのかどうかについて定める.

2. 制約条件
   最適化モデルの制約条件について説明する. まず, 曜日 (前期, 後期の情報を含む) の集合 K, 時限の集合 L を以下のように定義する.

$$K = \{1, 2, 3, 4, 5, 6, 7, 8, 9, 10\}$$
$$L = \{1, 2, 3, 4, 5\}$$

   ただし, 曜日の番号は, 月曜前期を 1, 火曜前期を 2, 水曜前期を 3, 木曜前期を 4, 金曜前期を 5, 月曜後期を 6, 火曜後期を 7, 水曜後期を 8, 木曜後期を 9, 金曜後期を 10, というふうに番号がふられている.

また, 時限の番号は, 1 限を 1, 2 限を 2, 3 限を 3, 4 限を 4, 5 限を 5, というふうに番号がふられている.
さらに, 時間割上の各コマ $(k,l)\in K\times L$ において履修できる科目の集合を $J(k,l)$ と定義する.

履修計画を最適化モデルとして定式化するにあたって, 必ず必要となる制約条件は, 次の三つである.
一つ目は, 同じコマに複数の科目を履修することはできないという原則に基づく制約である. 二つ目は, 同じ科目を二回以上履修することはできないという制約である. 三つ目は, 計画期の終わりに分野配属に必要な単位数を満たすことができるように履修するという制約である.
これらを数式で表すと, 以下のようになる.
(a) 同じコマには一つの科目しか履修できない

$$\sum_{j\in J(k,l)} x_{i,j} \leq 1, {}^{\forall}i \in Y, {}^{\forall}k \in K, {}^{\forall}l \in L$$

(b) 同じ科目は年度をまたいで履修できない

$$\sum_{i\in Y} x_{i,j} \leq 1, {}^{\forall}j \in C$$

(c) 必要単位数を満たすように履修する

$$\sum_{i\in Y}\sum_{j\in C_\alpha} m_j x_{i,j} \geq p_\alpha - q_\alpha, (\alpha = 1,2,3)$$

3. 目的関数

最適化モデルの目的関数について説明する. 前述の制約条件を満たしているならば, 計画通りに単位を修得できる確率は 0 ではないことがわかる. しかし, 単位修得確率は計画に依存して変化するのものである. そこで, 単位修得数の期待値を最大化することを目的とする.
これを数式で表すと, 以下のようになる.

$$\sum_{i\in Y}\sum_{j\in C} m_j r_j x_{i,j}$$

3.2.2　コースツリー

基本モデルの問題点の一つとして, コースツリーの履修順序を考慮に入れていないということが挙げられる. そこで, コースツリーに従って前科目を履修していれば, 後科目の単位修得確率が上がる仕組みを導入する. ただし, ここでは前科目を履修状況についてのみ考え, 実際に前科目の単位を修得できるのかどうかについては考えないものとする. これによって, コースツリーの履修順序に従った履修計画を立てるようになる.

ここで, コースツリーは $G(C, E)$ で与えられていたことに注意する. また, $(a, b) \in E$ であるとき, a はコースツリーの後科目, b はコースツリーの前科目というように表されていた. さらに, $E_1 \subseteq E$ は後科目 a と前科目 b が前期と後期で分かれている枝の集合, $E_2 \subseteq E$ は後科目 a と前科目 b が前期と後期で分かれていない枝の集合と定義されていた.

コースツリーのための決定変数として, 次の $y_{i,a,b}$ を導入する. ここで, $i(\in Y)$ は計画年を表している.

$$y_{i,a,b} = \begin{cases} 1 & (i\ \text{年に後科目}\ a\ \text{を履修するとき, すでに前科目}\ b\ \text{を履修済}) \\ 0 & (\text{それ以外}) \end{cases}$$

さらに, 後科目 a と前科目 b をコースツリーの順番通りに履修したときの, 後科目 a の単位修得確率を $r^t_{a,b}(>0)$ とする. そして, 基本モデルの目的関数に

$$\sum_{i \in Y} \sum_{(a,b) \in E} r^t_{a,b} y_{i,a,b}$$

を加える. このようにするだけでは, $y_{i,a,b}$ の値は常に 1 になってしまう. そこで, 次のような制約を加える.

- コースツリーの前科目と後科目が前期と後期にわかれているとき

$$2y_{i,a,b} \leq x_a + \sum_{\hat{i}=1}^{i} x_{\hat{i},b}, {}^\forall(a, b) \in E_1, {}^\forall i \in Y$$

- コースツリーの前科目と後科目が前期と後期にわかれていないとき

$$2y_{i,a,b} \leq x_a + \sum_{\hat{i}=1}^{i-1} x_{\hat{i},b}, {}^\forall(a, b) \in E_2, {}^\forall i \in Y/\{1\}$$

この制約によって, $y_{i,a,b}$ の値が 1 となるのは, 後科目 a を履修しており, かつ後科目 a を履修するより前に前科目 b を履修しているときとなる.

3.2.3　キャップ制

基本モデルの問題点の一つとして, 履修可能な科目を可能な限り履修すれば目的関数値が最大になってしまうということが挙げられる．それは, 履修科目の登録数が多くなると成績が下がるという現実の統計データと食い違ってしまう．そこで, 一日に履修する科目数を制限するキャップ制を導入する．本来のキャップ制は, 履修科目数の上限を超えて登録できないようにするものであるが, そのようにすると, 計画年数での実行可能解が得られないことがある．そこで, ここでは履修科目数の上限を越えたときに, 総修得単位数の期待値が下がるというペナルティを加えることにしている．これによって, 一日に履修する科目数ができる限り一定数以内になるような履修計画を立てるようになる．

キャップ制のための決定変数として, 次の $z^u_{i,k}$ を導入する．ここで, u は履修科目の登録数の上限を表している．さらに, $i{\in}Y$ は計画期, $k{\in}K$ は曜日を表している．

$$z^u_{i,k} \geq 0$$

さらに, 履修科目の登録数が u を越えたときの総修得単位数の期待値の減少値を $r^c_u({<}0)$ とする．そして, 基本モデルの目的関数に

$$\sum_{i\in Y}\sum_{k\in K} r^c_u z^u_{i,k}$$

を加える．このようにするだけでは, $z^u_{i,k}$ の値は常に 0 になってしまう．そこで, 次のような制約を加える．

$$z^u_{i,k} \geq \sum_{l\in L}\sum_{j\in J(k,l)} x_{i,j} - u, {}^{\forall}i \in Y, {}^{\forall}k \in K$$

この制約と目的関数によって, $z^u_{i,k}$ の値が 0 となるのは, 一日に履修する科目数が u 以下になるときとなる．

ここでは, 基本モデルの目的関数に次式を加えることで, キャップ制を $u{=}2,\ 3,\ 4$ で適用する．

$$\sum_{4}^{2}\sum_{i\in Y}\sum_{k\in K} r^c_u z^u_{i,k}$$

このため, 履修科目の登録数が多くなればなるほど大きなペナルティがかかる．

3.2.4　提案モデル

この節で説明した決定変数, 目的関数, 制約条件をまとめると, 履修計画の最適化モデルは以下のように表される.

$$
\begin{aligned}
\max \quad & \sum_{i \in Y} \sum_{j \in C} m_j r_j x_{i,j} + \sum_{(a,b) \in E} r^t_{a,b} y_{i,a,b} + \sum_{4}^{2} \sum_{i \in Y} \sum_{k \in K} r^c_u z^u_{i,k} \\
\text{s. t.} \quad & \sum_{j \in J(k,l)} x_{i,j} \leq 1, {}^\forall i \in Y, {}^\forall k \in K, {}^\forall l \in L \\
& \sum_{i \in Y} x_{i,j} \leq 1, {}^\forall j \in C \\
& \sum_{i \in Y} \sum_{j \in C_\alpha} m_j x_{i,j} \geq p_\alpha - q_\alpha, (\alpha = 1,2,3) \\
& 2y_{i,a,b} \leq x_a + \sum_{\hat{i}=1}^{i} x_{\hat{i},b}, {}^\forall (a,b) \in E_1, {}^\forall i \in Y \\
& 2y_{i,a,b} \leq x_a + \sum_{\hat{i}=1}^{i-1} x_{\hat{i},b}, {}^\forall (a,b) \in E_2, {}^\forall i \in Y, i \geq 2 \\
& x_{i,j} \in \{0,1\} \\
& y_{i,a,b} \in \{0,1\} \\
& z^u_{i,k} \geq 0
\end{aligned}
$$

この問題の決定変数の数は, 以下のようになる.

- 科目の決定変数 $x_{i,j}$ の数：$n\bar{y}$ 個
- コースツリーの決定変数 $y_{i,a,b}$ の数：$\{\bar{y}|E_1|$ + $(\bar{y}$-1)$|E_2|\}$ 個
- キャップ制の決定変数の数：$30\bar{y}$ 個

いま, 数理工学コースの科目数は n=69 であり, コースツリーの辺の数は $|E|$=150 である. よって, 二年計画 $\bar{y}$=2 のときのこの問題の変数の個数は高々 500 程度である. そのため, 市販の 0-1 整数計画問題に対するソルバーで十分解くことができる.

4　数値実験

この節では, 前節で説明したモデルに基づいた数値実験を行う．全ての実験に共通のデータとして, 2014 年度の京都大学の履修要覧およびコースツリー表 (付録Ａの図 1) を用いた．実験は CPU が Intel(R)Core(TM)i5CPU 2.40GHz, メモリが 4.0GB の計算機上で行い, プログラムの実装には MATLAB8.3.0.532(R2014a) を用いた．整数計画問題は, MATLAB に付属しているコマンド intlinprog で解いた．

また, 単位修得確率に関する各変数の値は, 以下のように定めた．

$$
\begin{aligned}
r_j &= 0.8 - 0.2d_j \\
r^t_{a,b} &= 0.4 \\
r^c_u &= 0.5
\end{aligned}
$$

この実験では, 三回生になってようやくコース配属された学生が, 分野配属を目的として履修計画を立てる場合を想定する．そのため, 二回生および三回生の配当科目は全て履修できるものとする．

4.1　実験 1

履修計画の最適化モデルの, コースツリーの制約およびキャップ制の制約の有無による違いを比較する実験を行う．ここで, 立てる計画は二年計画であり, 学生は一回生配当科目をすべて修得しているものとする．結果は, 次の表 4 のようになる．

表 4　実験 1

コースツリーの制約	キャップ制の制約	登録科目数	総単位数の期待値	登録科目あたりの単位数の期待値	コースツリーの前科目を未履修な登録科目の数	キャップ制の制限を越えている科目の数	時間割
有	有	41	63.80	1.56	3	6	表 6
有	無	43	66.80	1.55	3	8	表 7
無	有	37	50.60	1.37	9	6	表 8
無	無	39	53.20	1.36	10	8	表 9

コースツリーの制約の有無を比較すると, コースツリーの制約がある方が, 総単位数の期待値および登録科目科目あたりの単位数の期待値が大きくなっている．よって, コースツリーの制約があることで, より単位修得のしやすい履修計画を立てることができるといえる．

キャップ制の制約の有無を比較すると, キャップ制がある方が総単位数の期待値が小さくなっている. しかし, 登録科目科目あたりの単位数の期待値はほとんど変化していない. これは, キャップ制がある場合に履修しなくなった科目分, 総単位数の期待値が下がっているだけである. よって, キャップ制の制約があることで, 必要性の低い科目を履修しなくてもいいことがわかり, その分の労力を削減することができるといえる.

提案モデルと基本モデルを比較すると, 提案モデルの方が総単位数の期待値も科目あたりの単位数の期待値も大きいことから, 提案モデルの方がより単位修得のしやすい履修計画を立てることができるといえる.

4.2 実験２

提案モデルによって作成した履修計画の, 一年計画と二年計画の場合の違いを比較する実験を行う. ここで, 学生は一回生配当科目をすべて修得しているものとする. 結果は, 次の表 5 のようになる.

表 5　実験２

計画年数	登録科目数	総単位数の期待値	登録科目あたりの単位数の期待値	コースツリーの前科目を未履修な登録科目の数	キャップ制の制限を越えている科目の数	時間割
2	41	63.80	1.56	3	6	表 6
1	32	35.40	1.11	4	12	表 10

二年計画と一年計画の場合を比較すると, 総単位数の期待値も, 科目あたりの単位数の期待値も, 二年計画の場合のほうが一年計画の場合より明らかに大きい.

この状況で, 分野配属に必要な単位数は 42 単位である. 一年計画の場合, 総単位数の期待値が必要単位数を下回っている. また, ここでは, 自然・専門の科目以外はすべて修得済であると仮定しているが, もしそうでなければ, さらに留年する確率が高くなる. よって, 単位修得状況が厳しい場合は, 一年で無理な計画を実行しようとするよりは, 初めから二年で計画を行う方が効率的であるといえる.

また, 二年計画と一年計画で, コースツリーの前科目が未履修な科目の数にはあまり差がない. このことから, 情報学科数理コースの時間割は, コースツリー通りに無理なく履修できるようによく構成されているということがいえる.

5 まとめと今後の課題

本研究では, コースツリーを考慮した最適な履修計画を作成する数理モデルを考案した. また, 実際にこのモデルを用いて履修計画を作成する実験を行った. その結果, 提案したモデルを用いることで, より効率的な履修計画を作成することができることが確認された. また, 既存の時間割はよく構成されていたため, 特殊な履修状況を除いて, コースツリー通りに無理なく履修できることも確認できた.

本報告書で行った実験では, 各科目の単位修得確率を一律に定めていたが, 実際の各科目の単位修得確率に関しては, 個別の内容や, 個々人の得手不得手によって変化するものである. もし単位修得確率についての正確なデータを得ることができれば, また別の結果が期待できるかもしれない.

謝辞

本研究に取り組みにあたって多大なる助言をいただき, 本報告書の作成にあたっては細部に至るまでに様々なご指摘や適切なご指導を賜った山下信雄教授に深く感謝致します. また, 福田助教ならびに最適化数理研究室の皆様に厚く御礼申し上げます.

参考文献

[1] Ｂ. コルテ, Ｊ. フィーゲン, 組合せ最適化　第２版, 丸善出版, 2009
[2] 藤沢克樹, 梅谷俊治, 応用に役立つ５０の最適化問題, 朝倉書店, 2009
[3] 福島雅夫, 新版数理計画入門, 朝倉書店, 2011
[4] 五十嵐歩美, 高校生が挑む「快適！スクールバスプロジェクト」, 2014
[5] 今野浩, 数理決定法入門　キャンパスのＯＲ, 朝倉書店,1992
[6] 久保幹雄, 松井知己, 組合せ最適化 [短編集], 朝倉書店, 1999
[7] 久保幹雄, 組合せ最適化とアルゴリズム, 共立出版, 2000
[8] 松井泰子, 根本俊男, 宇野毅明, 入門オペレーションズ・リサーチ, 東海大学出版会, 2008
[9] 京都大学シラバス集, http://ocw.kyoto-u.ac.jp/syllabuses, 2015
[10] 全学共通科目履修の手引き, http://www.z.k.kyoto-u.ac.jp/zenkyo/guidance, 2015
[11] 京都大学情報学科履修要項, http://www.s-im.t.kyoto-u.ac.jp/ja/curriculum/youran, 2015
[12] MATLAB2014, MathWorks, http://jp.mathworks.com/

付録 A　コースツリー表

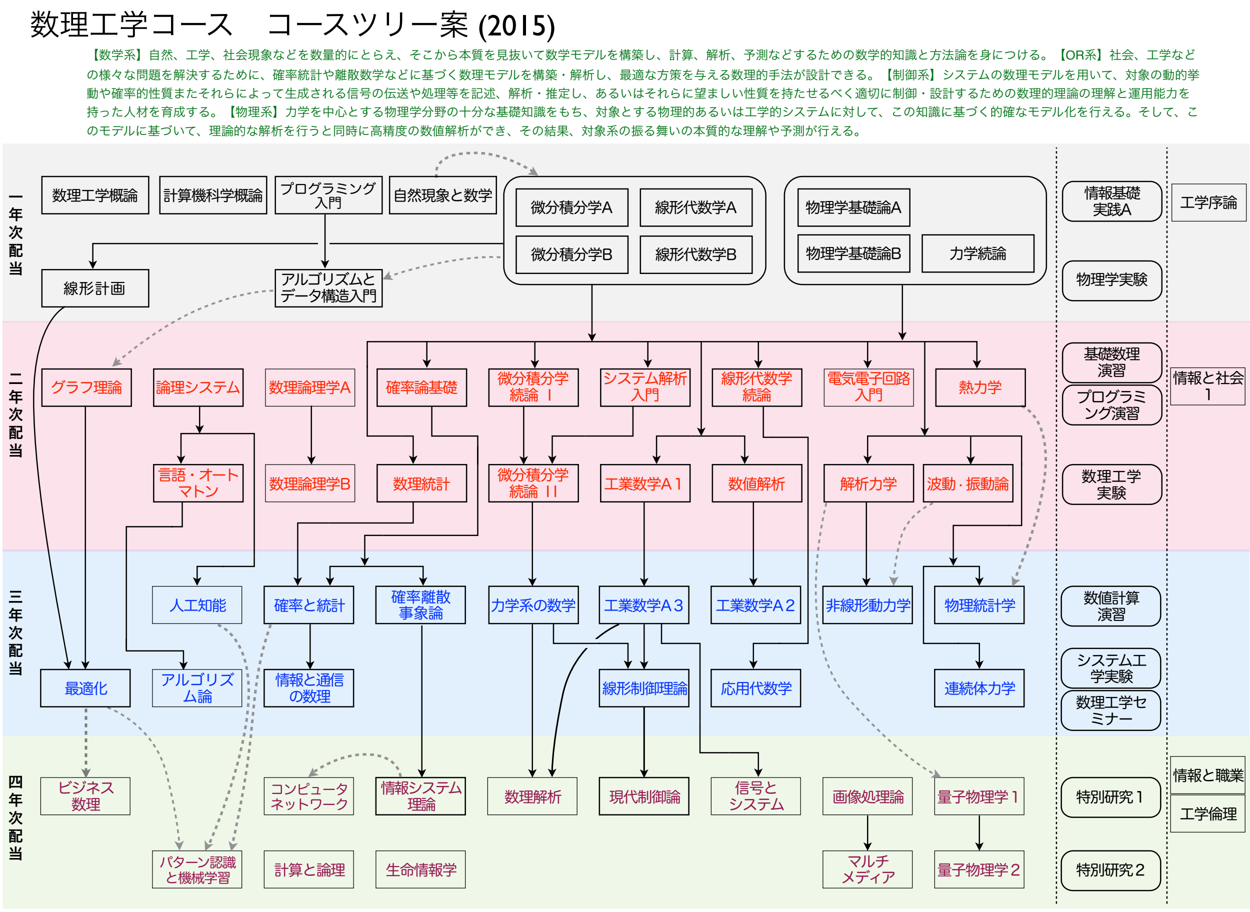

図 1

付録 B　時間割表

実験で出力された時間割

表 6　コースツリー：有　キャップ制：有

1年目前期	1限	2限	3限	4限	5限
月曜日			数値計算演習	数値計算演習	
火曜日		線形代数学続論	基礎数理演習	基礎数理演習	
水曜日	工業数学Ａ３	システム解析入門			
木曜日	確率論基礎		微分積分学続論Ｉ		
金曜日		論理システム	プログラミング演習	プログラミング演習	

1年目後期	1限	2限	3限	4限	5限
月曜日			数理工学実験	数理工学実験	
火曜日			数理工学実験	数理工学実験	
水曜日	言語・オートマトン	解析力学	数値解析		
木曜日		熱力学	微分積分学続論ＩＩ		
金曜日	数理統計	数理工学セミナー			

2年目前期	1限	2限	3限	4限	5限
月曜日		工業数学Ａ２			
火曜日		確率離散事象論			
水曜日		確率と統計	人工知能		
木曜日		物理統計学			
金曜日	線形制御理論	振動・波動論	非線形動力学		

2年目後期	1限	2限	3限	4限	5限
月曜日		応用代数学			
火曜日		連続体力学			
水曜日	情報と通信の数理	最適化			
木曜日	アルゴリズム論	工業数学Ａ１	システム工学実験	システム工学実験	
金曜日			システム工学実験	システム工学実験	

表７　コースツリー：有　キャップ制：無

1年目前期	1限	2限	3限	4限	5限
月曜日			数値計算演習	数値計算演習	
火曜日		線形代数学続論	基礎数理演習	基礎数理演習	
水曜日	工業数学Ａ３	システム解析入門			
木曜日	確率論基礎	グラフ理論	微分積分学続論Ｉ		
金曜日		論理システム	プログラミング演習	プログラミング演習	

1年目後期	1限	2限	3限	4限	5限
月曜日			数理工学実験	数理工学実験	
火曜日		連続体力学	数理工学実験	数理工学実験	
水曜日	言語・オートマトン	解析力学	数値解析		
木曜日		熱力学	微分積分学続論II		
金曜日	数理統計	数理工学セミナー			

2年目前期	1限	2限	3限	4限	5限
月曜日		工業数学Ａ２			
火曜日		確率離散事象論			
水曜日		確率と統計	人工知能		
木曜日		物理統計学			
金曜日	線形制御理論	振動・波動論	非線形動力学		

2年目後期	1限	2限	3限	4限	5限
月曜日		応用代数学			
火曜日		数理論理学B			
水曜日	情報と通信の数理	最適化			
木曜日	アルゴリズム論	工業数学Ａ１	システム工学実験	システム工学実験	
金曜日			システム工学実験	システム工学実験	

表８　コースツリー：無　キャップ制：有

1年目前期	1限	2限	3限	4限	5限
月曜日		工業数学Ａ２	数値計算演習	数値計算演習	
火曜日		確率離散事象論	基礎数理演習	基礎数理演習	
水曜日	工業数学Ａ３		人工知能		
木曜日	確率論基礎		微分積分学続論Ⅰ		
金曜日		論理システム	非線形動力学		
1年目後期	1限	2限	3限	4限	5限
月曜日		応用代数学	数理工学実験	数理工学実験	
火曜日			数理工学実験	数理工学実験	
水曜日	情報と通信の数理		数値解析		
木曜日	アルゴリズム論	熱力学			
金曜日	数理統計	論理システム			
2年目前期	1限	2限	3限	4限	5限
月曜日					
火曜日		数理論理学A			
水曜日		確率と統計			
木曜日		物理統計学			
金曜日		振動・波動論	プログラミング演習	プログラミング演習	
2年目後期	1限	2限	3限	4限	5限
月曜日					
火曜日		連続体力学			
水曜日	言語・オートマトン	最適化			
木曜日		工業数学Ａ１	システム工学実験	システム工学実験	
金曜日		数理工学セミナー	システム工学実験	システム工学実験	

表 9　コースツリー：：無　キャップ制：無

1年目前期	1限	2限	3限	4限	5限
月曜日		工業数学Ａ２	数値計算演習	数値計算演習	
火曜日			基礎数理演習	基礎数理演習	
水曜日	工業数学Ａ３		人工知能		
木曜日	確率論基礎	グラフ理論	微分積分学続論 I		
金曜日	線形制御理論		非線形動力学		
1年目後期	1限	2限	3限	4限	5限
月曜日		応用代数学	数理工学実験	数理工学実験	
火曜日		連続体力学	数理工学実験	数理工学実験	
水曜日	情報と通信の数理	最適化			
木曜日	アルゴリズム論		システム工学実験	システム工学実験	
金曜日	数理統計	数理工学セミナー	システム工学実験	システム工学実験	
2年目前期	1限	2限	3限	4限	5限
月曜日					
火曜日		確率離散事象論			
水曜日		確率と統計			
木曜日		物理統計学			
金曜日		論理システム	プログラミング演習	プログラミング演習	
2年目後期	1限	2限	3限	4限	5限
月曜日					
火曜日		数理論理学Ｂ			
水曜日	言語・オートマトン	解析力学	数値解析		
木曜日		工業数学Ａ１	微分積分学続論 II		
金曜日					

表 10　一年計画

1年目前期	1限	2限	3限	4限	5限
月曜日		工業数学Ａ２	数値計算演習	数値計算演習	
火曜日		線形代数学続論	基礎数理演習	基礎数理演習	
水曜日	工業数学Ａ３	システム解析入門	人工知能		
木曜日	確率論基礎	グラフ理論	微分積分学続論Ⅰ		
金曜日		振動・波動論	プログラミング演習	プログラミング演習	

1年目後期	1限	2限	3限	4限	5限
月曜日		応用代数学	数理工学実験	数理工学実験	
火曜日		連続体力学	数理工学実験	数理工学実験	
水曜日	情報と通信の数理	最適化	数値解析		
木曜日	アルゴリズム論	熱力学	システム工学実験	システム工学実験	
金曜日	数理統計	数理工学セミナー	システム工学実験	システム工学実験	