0-1 線型計画法による小班別収穫予定

木　平　勇　吉
信州大学農学部　森林利用学研究室

目　　次

序論…… 2
1　研究の目的…… 2
2　研究の方法…… 3
3　研究の動機…… 4
第一章　収穫予定のための 0-1 線型計画法…… 4
1　収穫予定の制約条件と目的関数…… 4
2　小班の伐採分期決定問題の定式化…… 6
第二章　一般 0-1 線型計画法の解法… 7
1　線型計画法…… 7
2　0-1 線型計画法の解法…… 9
(1)　悉皆列挙法……10
(2)　グリーンバーグの解法……11
3　バラスの解法……12
(1)　問題の標準化……12
(2)　陰的列挙……13
(3)　Solution tree ……15
(4)　列挙順序……17
(5)　省略基準……20
(6)　計算手順……21
第三章　収穫予定のための 0-1 線型計画法の解法——加速列挙法のアルゴリズム——……21
1　バラスの問題との比較……21
2　Solution tree の比較……22
3　列挙順序の比較……25
4　省略基準の比較……26
5　計算手順の比較……28
第四章　加速列挙法の適用方法……28
1　問題サイズの縮小……28
2　小班の並べかえ……30
第五章　加速列挙法の処理能力……34
1　計算時間比較のための問題設定……34
2　バラスの解法との比較……37
3　問題のサイズと計算時間……37
4　陽的列挙数と計算時間……38
5　陽的列挙率と省略基準……38
(1)　陽的列挙率……38
(2)　省略基準……39
第六章　収穫予定への適用……41
1　事業団地の収穫予定……41
2　林道計画を取り入れた収穫予定……45
(1)　制約条件としての林道計画…45
(2)　適用事例……47
3　伐採林分の配置……49
(1)　最適解とその小班配置……49
(2)　森林の場所的秩序……50
第七章　まとめ……52
謝辞……53
引用文献……54
付録　(1)　入力データの様式……55
(2)　FOA プログラム ……56

序　　論

1　研究の目的

伐採を予定する林分を特定することにより収穫予定案に具体的な根拠を与え，林分相互の配置を規整する収穫予定の方法を見いだすことが本研究の目的である。

ところで，人間の手が入る以前の森林には，同じ林相の樹林が広がっていた。しかし，やがて人間はこの斉一な森林の一部分を伐採しはじめた。伐採した跡地には以前と同じ樹種を更新したり，あるいは，まったく異った樹種を人工的に植栽した。この伐採と更新とを年々，森林のあちこちで繰り返すことにより，斉一な林相であった天然林の中に林齢や樹種が異なった多くの小林分が混在するようになった。その結果，今日の森林は，それぞれ林齢，樹種，生育状態，保育方法あるいは施業目的の異なる小林分が隣接しあって出来た一つのモザイクと見なすことが出来る。そして，それら林分相互の配置は林木の生育，風・霜・雪などの気象被害，水や土壌の保全，森林作業，風景など森林の機能に大きな影響を与えている。

さて，現在のモザイクの形，配置は過去の伐採，更新の繰り返しの結果であり，現在の伐採，更新が将来のモザイクをきめる。したがって，目標とする望ましい林分配置を想定して，それを実現するような計画的な施業案を立てることが森林計画の課題である。この課題は林分の空間配置の規整とよばれている。

ところで，目標とする配置を実現させる唯一の手段は伐採であるから，それを実現するような伐採計画を見つけなければならない。その伐採計画はいかほど伐採するかという量だけでなく，どの林分を，いつ伐採するかという「場所」と「時間」を明らかにしなければならない。伐採林分の所在場所と伐採時期を特定した計画を立てれば，それらが実行された場合の将来の林分配置がわかり，それを事前に評価出来るからである[18)33)]。

この伐採量と同時に伐採林分の所在場所と伐採時期を特定した収穫予定を見つける手法を開発することが，この研究の課題である。

ところで，日本の森林は，すでに林分ごとに測量区画され，森林基本図にその場所，形が描かれ，面積，材積，樹種，林齢は森林簿に記述されている。地図や森林簿には森林区画の最小単位であり，施業の最小単位である小班（Sub-compartment）ごとに記載されている。そこで，収穫予定を立てる時に，伐採は小班を単位として行うと仮定すれば，伐採林分の所在場所を特定し，表示する手段としてこの「小班」を用いることが出来る。すなわち，伐採する小班の名前を決めることは，伐採林分の場所を特定することになる。

一方，森林計画では5年あるいは10年などを区切りにした期間を分期（Planning term）とよび，時間の単位としている。小班の伐採は一つの分期中に行うと仮定すれば，小班の伐採分期を決めることは伐採林分の伐採時期を特定することになる。

しかも，伐採する小班の名前と伐採分期が決まれば，それらを合計すれば全体としての伐採材積，面積も決まる。したがって，これら2つの仮定を用いることにより「伐採量と同時に伐採林分の所在場所と伐採時期を特定した収穫予定を見つける手法」という先に述べた研究課題は，「伐採を予定する小班の名前とその伐採分期を決定する手法」という具体的な課題に置き換えることが出来る。これを小班別収穫予定とよぶ。ところで，小班ごとに伐採分

期を決めるには，森林施業上の制約を考慮しなければならない。小班個別の制約条件として，まず，(i) その林分が伐期齢に達しているか，(ii) 林道が開設されているか，(iii) 法令により伐採が禁じられていないか，(iv) 風向を考えて隣接林分との位置や，伐採順序はよいか，(v) 集運材作業，林道開設の都合と伐採順序とが合っているかなどが挙げられる。他方，森林全体からみると年々の木材売上高や，労務量の変動を少なくするために，分期毎の伐採材積や伐採面積は一定の許容範囲内でなければならない。

そこで，これら林分個別の制約条件と森林全体としての制約条件の下で，森林経営上，設定される評価基準を満足させるように小班ごとの伐採分期を決定することが，小班別収穫予定の課題である。

さて，この小班ごとの最適伐採分期の決定問題は次のとおり，一つの数理問題として説明できる[31]。

今，n 個の小班からなる森林を対象に m 分期間の収穫予定を立てるとする。i 番目の小班の j 分期の伐採予定を変数 x_{ij} で定義する。伐採は小班ごとに，分期ごとに行うという仮定により，この x_{ij} の内容は「伐採しない」か「伐採する」かのいずれかである。「伐採しない」は0，「伐採する」は1と約束すると，x_{ij} は0か1かのいずれかの値を持つ変数と規定できる。この変数 x_{ij} を用いると，小班別収穫予定は小班個別の事情，分期ごとの伐採材積，伐採面積の許容範囲を制約条件，全体として達成すべき目標を目的関数とする線型計画となる。ことに $x_{ij}=0$ or 1 という制約をもつ特殊性からこれを0–1線型計画法あるいは0–1計画 (Zero–one integer linear programs, Binary problems) とよぶ。この0–1計画を解くことにより，小班の最適伐期を求めることが出来る[32]。

まとめると，森林全体の収穫予定を小班ごとの最適伐期決定問題とおきかえ，これを0–1線型計画法により数理的に解く。その結果から伐採林分の所在場所，相互の位置関係を明らかにして，将来の林分の空間配置を規整することが本研究の目的である。

2　研究の方法

小班の伐期決定問題を0–1計画により解く考え方はバラス (Egon Balas) が最初に提案した[24]。この考え方を応用すれば面積，材積あるいは折衷平分法，法正齢級法あるいは林分経済法などの従来の収穫予定法を0–1計画に帰着させて解くことが出来る[31]。あるいは林道計画と伐採順序の調整が出来ることが報告されている[24]。さて，収穫予定問題が0–1計画により解けることは，既に理論的には明快であるが，しかし，それを実際に解くのは困難である。扱う小班の個数が多く，計画期間が長くなると変数 x_{ij} の個数が多くなり，問題を解く計算量が厖大になる。そのため，現在のところ，この手法は収穫予定法として実際には使われていない。0–1計画を収穫予定に適用するためには，まず，0–1計画を解く解法の開発が必要である[31]。

そこで，本研究は，収穫予定に特有の性質を附帯条件とする収穫予定のための0–1計画 (Forestry oriented zero–one problem) を作り，このタイプの問題にしか適用できないが，この問題に限っては能率の良い専用的な解法 FOA (Forestry Oriented Algorithm) を開発することから始めた。

FOA はバラスの加法列挙法 (Additive algorithm) の考え方を基盤にした列挙法であり，附帯条件を活用して列挙の大部分を省略することにより計算を加速するアルゴリズムを持っ

ている。このアルゴリズムをプログラミングして電子計算機による計算時間を調べたところ，変数の数が多くなっても比較的短時間で解を見つけることがわかった。

次に，このプログラムを用いて，1つの森林地域を対象に制約条件を変えて，小班別の収穫予定を試算した。まず，第1段階は毎分期の伐採材積の変動を少なくし，安定した収穫予定案を求めた。第2段階は，ある与えられた林道計画のもとでの伐採小班の順序づけを試みた。これは林道を先に開設し，その後で伐採を行うという森林作業上の観点からの場所規整である。第3段階は隣接する伐採林分相互の位置関係を制約条件に含めて，林分配置の規整を試みた。これら3つの段階によって，収穫量が毎分期安定し，林道開設計画との順序性が正しく，隣接林分の配置を規整した小班別の収穫予定を見いだす方法を明らかにする。そして，いくつかの事例計算により0-1計画による小班別収穫予定の適用可能性とその限界を示す。

3　研究の動機

国有林の長期計画では，保続計算により標準伐採量を算出する。標準伐採量とは将来の収穫保続を確実なものにする観点から計算された現在の許容伐採材積である[30]。それに対し，毎年の事業を予定する短期計画が伐採林分を選定する。この選定は具体的な事業実行の都合で行う。

ところで，今日，この長期計画と短期計画との間に相当大きな齟齬が生じている[16]。その第1は許容伐採量に見合うだけの伐採林分を現実の森林の中に見つけることが困難になってきたという問題である。そして，将来にわたり保続的な収穫を保証している筈の標準伐採量が，5年毎の保続計算の都度，減少していることである。その結果，施業計画の中心になる標準伐採量への信頼がうすらいでいる[26]。この信頼を回復するには，現在の保続計算という収穫予定法の理論と手法を整備して，収穫予定林分という具体的な根拠を持った収穫予定法を採用しなければならないと考えた[14]。これが小班別収穫予定の研究の第1の動機である。

第2の動機は，収穫計画と林道計画との齟齬である。現在の齢級別面積表を基礎資料とする収穫予定では，個々の林分の所在場所，作業条件などを無視する。しかし今日，まず林道がなければ技術的にも，経済的にも収穫は行えないから，林分の所在場所のない資料による収穫予定は，伐採搬出作業の面から不確かなものである。林道開設費は高く，まとまった蓄積が乏しくなっている現況では，林道計画を収穫予定立案時にとり入れる手法が必要である[13]。さらに，林分の空間的な配置は森林の機能を維持する上で重要な条件である。保安林機能，風致維持，防風，土壌，生態系としての環境を維持するためには，森林の場所的な規整を行い，理想とする林分配置を実現しなければならない[7]。このために収穫予定は，収穫量，収穫時期と併せて，収穫場所を明示しなければならない[12][15]。

要約すれば，現在の収穫予定案に具体的な根拠を与え，林道計画とも調整され，森林の空間配置を規整する手法の開発が必要である。その一つの手法として小班別収穫予定を取りあげ，これを0-1計画により理論的に裏づけし，かつ，計算実務も容易な収穫予定法を見いだしたい。これが本研究の動機である。

第一章　収穫予定のための0-1線型計画法

1　収穫予定の制約条件と目的関数

n 個の小班について m 分期間の収穫予定を立てるとする。各小班の面積と，それぞれの分期における伐期材積は表—1.1のとおりである。各小班の各分期の伐採予定を変数 x_{ij} で定義する。

表—1.1　対象小班の面積，伐期材積

小班 \ 分期	面積	1	2 …… j …… m
		x_{11}	x_{12} …… x_{1m}
1	a_1	v_{11}	v_{12} …… v_{1m}
		x_{21}	x_{22} …… x_{2m}
2	a_2	v_{21}	v_{22} …… v_{2m}
⋮	⋮	……	x_{ij} ……
i	a_i	……	v_{ij} ……
⋮	⋮	x_{n1}	x_{n2} …… x_{nm}
n	a_n	v_{n1}	v_{n2} …… v_{nm}

a_i $(i=1, 2, \cdots\cdots, n)$; a_i は i 小班の面積
v_{ij} $(i=1, 2, \cdots\cdots, n\ j=1, 2, \cdots\cdots, m)$; v_{ij} は i 小班の j 分期の伐期材積
x_{ij} $(i=1, 2, \cdots\cdots, n\ \ j=1, 2, \cdots\cdots, m)$; x_{ij} は i 小班の j 分期における伐採予定

先に述べたが「伐採は小班を最小単位として，1分期内に行う」ことを前提条件にすると収穫予定に関する制約条件と目的関数は次のように変数 x_{ij} の1次式により表現することが出来る[19]。

制約条件

(i)　$x_{ij}=0$ or 1　　$(i=1, 2, \cdots\cdots, n\ \ j=1, 2, \cdots\cdots, m)$　　(1.1)

(ii)　各分期の総伐採材積の許容範囲

$$\left.\begin{array}{ll} 1\text{分期} & V_1 \leqq v_{11}x_{11}+v_{21}x_{21}+\cdots\cdots+v_{n1}x_{n1} \leqq V'_1 \\ 2\text{分期} & V_2 \leqq v_{12}x_{12}+v_{22}x_{22}+\cdots\cdots+v_{n2}x_{n2} \leqq V'_2 \\ \vdots & \vdots \qquad\qquad\qquad\qquad \vdots \\ m\text{分期} & V_m \leqq v_{1m}x_{1m}+v_{2m}x_{2m}+\cdots\cdots+v_{nm}x_{nm} \leqq V'_m \\ \multicolumn{2}{l}{V_j,\ V'_j\ (j=1, 2, \cdots\cdots, m);\ j\ \text{分期の伐採許容材積の下限および上限}} \end{array}\right\} \quad (1.2)$$

(iii)　各分期の伐採面積の許容範囲

$$\left.\begin{array}{ll} 1\text{分期} & A_1 \leqq a_1x_{11}+a_2x_{21}+\cdots\cdots+a_nx_{n1} \leqq A'_1 \\ 2\text{分期} & A_2 \leqq a_1x_{12}+a_2x_{22}+\cdots\cdots+a_nx_{n2} \leqq A'_2 \\ \vdots & \vdots \qquad\qquad\qquad\qquad \vdots \\ m\text{分期} & A_m \leqq a_1x_{1m}+a_2x_{2m}+\cdots\cdots+a_nx_{nm} \leqq A'_m \\ \multicolumn{2}{l}{A_j,\ A'_j\ (j=1, 2, \cdots\cdots, m);\ j\ \text{分期の伐採許容面積の下限および上限}} \end{array}\right\} \quad (1.3)$$

伐採後，伐跡地は全部，ただちに更新すると仮定すれば，(1.3) 式は将来森林の齢級別面積の許容範囲と解釈できる。

(iv)　各小班は計画期間中に1度だけ伐採する。

$$\left.\begin{array}{ll} 1\text{小班} & x_{11}+x_{12}+\cdots\cdots+x_{1m}=1 \\ 2\text{小班} & x_{21}+x_{22}+\cdots\cdots+x_{2m}=1 \\ \vdots & \vdots \\ n\text{小班} & x_{n1}+x_{n2}+\cdots\cdots+x_{nm}=1 \end{array}\right\} \tag{1.4}$$

(v)　小班個別の理由による伐採規制

$$\left.\begin{array}{ll} \text{伐期齢} & x_{ij}=0;\ i\text{小班は}j\text{分期には，まだ伐期齢に達していない。} \\ \text{林　道} & x_{ij}=0;\ i\text{小班は}j\text{分期には，林道が開設されていない。} \\ \text{法　令} & x_{ij}=0;\ i\text{小班は}j\text{分期には，法令により伐採が許されない。} \end{array}\right\} \tag{1.5}$$

(vi)　小班の伐採分期の特定化

$$x_{ij}=1;\ i\text{ 小班は}j\text{分期に必ず伐採する。} \tag{1.6}$$

(vii)　小班相互の伐採順序

h 小班は k 小班より後で伐採するとする[31]。

$$\left.\begin{array}{l} x_{h1}\leqq x_{k1} \\ x_{h1}+x_{h2}\leqq x_{k1}+x_{k2} \\ \quad\vdots \\ x_{h1}+x_{h2}+\cdots\cdots+x_{hm}\leqq x_{k1}+x_{k2}+\cdots\cdots+x_{km} \end{array}\right\} \tag{1.7}$$

目的関数

計画期間中の総伐採材積として，それを最大にする[1]。

$$Z=v_{11}x_{11}+v_{12}x_{12}+\cdots\cdots+v_{nm}x_{nm}\quad \max \tag{1.8}$$

2　小班の伐採分期決定問題の定式化

先に述べたが 0–1 計画は問題のサイズが大きくなると実際に解を求めることが困難になる。そこで，取り扱う問題の制約条件や，目的関数を一定の形式と内容に限定し，そのタイプの問題に専用的な解法を開発することにより，このサイズの障害の解決を試みた。したがって，上記の収穫予定問題に特有の条件の中から基本となる型の制約条件を選び出して，小班の伐採分期決定問題を定式化する。

制約条件 1

〔材積〕

$$\left.\begin{array}{ll} 1\text{分期} & v_{11}x_{11}+v_{21}x_{21}+\cdots\cdots+v_{n1}x_{n1}\geqq V_1 \\ 2\text{分期} & v_{12}x_{12}+v_{22}x_{22}+\cdots\cdots+v_{n2}x_{n2}\geqq V_2 \\ \vdots & \vdots \\ m\text{分期} & v_{1m}x_{1m}+v_{2m}x_{2m}+\cdots\cdots+v_{nm}x_{nm}\geqq V_m \\ \text{〔面積〕} & \\ 1\text{分期} & a_1x_{11}+a_2x_{21}+\cdots\cdots+a_nx_{n1}\geqq A_1 \\ 2\text{分期} & a_1x_{12}+a_2x_{22}+\cdots\cdots+a_nx_{n2}\geqq A_2 \\ \vdots & \vdots \\ m\text{分期} & a_1x_{1m}+a_2x_{2m}+\cdots\cdots+a_nx_{nm}\geqq A_m \end{array}\right\} \tag{1.9}$$

制約条件 2

〔材積〕

$$\left.\begin{array}{ll}
1\text{分期} & v_{11}x_{11}+v_{21}x_{21}+\cdots\cdots+v_{n1}x_{n1}\leqq V'_1 \\
2\text{分期} & v_{12}x_{12}+v_{22}x_{22}+\cdots\cdots+v_{n2}x_{n2}\leqq V'_2 \\
\vdots & \vdots \qquad\qquad\qquad \vdots \\
m\text{分期} & v_{1m}x_{1m}+v_{2m}x_{2m}+\cdots\cdots+v_{nm}x_{nm}\leqq V'_m \\
\text{〔面積〕} & \\
1\text{分期} & a_1x_{11}+a_2x_{21}+\cdots\cdots+a_nx_{n1}\leqq A'_1 \\
2\text{分期} & a_1x_{12}+a_2x_{22}+\cdots\cdots+a_nx_{n2}\leqq A'_2 \\
\vdots & \vdots \qquad\qquad\qquad \vdots \\
m\text{分期} & a_1x_{1m}+a_2x_{2m}+\cdots\cdots+a_nx_{nm}\leqq A'_m
\end{array}\right\} \tag{1.10}$$

制約条件 3

$$x_{ij}=0 \ or \ 1 \qquad (i=1,2,\cdots\cdots,n \quad j=1,2,\cdots\cdots,m) \tag{1.11}$$

制約条件 4

$$\left.\begin{array}{ll}
1\text{小班} & x_{11}+x_{12}+\cdots\cdots+x_{1m}=1 \\
2\text{小班} & x_{21}+x_{22}+\cdots\cdots+x_{2m}=1 \\
\vdots & \vdots \qquad\qquad\qquad \vdots \\
n\text{小班} & x_{n1}+x_{n2}+\cdots\cdots+x_{nm}=1
\end{array}\right\} \tag{1.12}$$

目的関数

$$Z=v_{11}x_{11}+v_{12}x_{12}+\cdots\cdots+v_{nm}x_{nm} \qquad \max \tag{1.13}$$

附帯条件 1

制約条件，目的関数の係数である $v_{ij}(i=1,\ 2,\cdots\cdots,n\ \ j=1,\ 2,\cdots\cdots,m)$,

$a_i(i=1,\ 2,\cdots\cdots,n)$ はすべて非負である。 (1.14)

附帯条件 2

林分は生長するから，各小班の伐期材積は分期につれて増加する。

$$\left.\begin{array}{ll}
1\text{小班} & v_{11}\leqq v_{12}\leqq\cdots\cdots\leqq v_{1m} \\
2\text{小班} & v_{21}\leqq v_{22}\leqq\cdots\cdots\leqq v_{2m} \\
\vdots & \vdots \qquad\qquad \vdots \\
n\text{小班} & v_{n1}\leqq v_{n2}\leqq\cdots\cdots\leqq v_{nm}
\end{array}\right\} \tag{1.15}$$

以上のとおり，制約条件 (1.9) (1.10) (1.11) (1.12) 式，目的関数 (1.13) 式と附帯条件 (1.14) (1.15) 式を小班別収穫予定のための 0-1 計画問題と規定する。

第二章　一般 0-1 線型計画法の解法

1　線型計画法

ある制約条件のもとで，目的関数を最大（または最小）にする問題を最適化問題という[8)]。ここでは最大化問題により説明する。生産活動を行う為に使われる資源，労働力，資金，機械などが有限で量的な制約がある場合に，それを数量的に表現したものが制約条件である。

そして，生産活動で達成しようと努力し，活動の成果を判定する尺度に用いられる事項を数量的に表現したものが目的関数である。それら，制約条件と目的関数がともに1次式である最適化問題が線型計画法である。線型計画法は扱う関数が線型1次式であるという特殊性があり，数学的に扱い易いので多くの具体的な問題に適用され，その有効なことが実証されている。森林計画の分野では，線型モデルによる収穫予定[4)22)23)28)]，全国森林資源の開発計画[9)]。資源配分計画[25)]。最適土地利用計画[10)]など，多くの事例が報告されている。

生産計画に関する線型計画の標準的な形式は次のとおりである[8)]。

制約条件

$$\left.\begin{array}{l} a_{11}x_1+a_{12}x_2+\cdots\cdots+a_{1n}x_n\leqq b_1 \\ a_{21}x_1+a_{22}x_2+\cdots\cdots+a_{2n}x_n\leqq b_2 \\ \quad\vdots \qquad\qquad\qquad\qquad \vdots \\ a_{m1}x_1+a_{m2}x_2+\cdots\cdots+a_{mn}x_n\leqq b_m \end{array}\right\} \tag{2.1}$$

$$x_1\geqq 0,\quad x_2\geqq 0,\quad \cdots, x_n\geqq 0 \tag{2.2}$$

目的関数

$$Z=c_1x_1+c_2x_2+\cdots\cdots+c_nx_n \quad \max \tag{2.3}$$

(2.1)，(2.2) 式の制約のもとでZを最大にする変数 x_1，x_2，……x_n の値を求める。

さて，条件式が上のように，すべて等不等式の場合を標準型とよび，それに対し，条件式がすべて等式の場合を正規型とよぶ[31)]。

$$\left.\begin{array}{l} a_{11}x_1+a_{12}x_2+\cdots\cdots+a_{1n}x_n=b_1 \\ a_{21}x_1+a_{22}x_2+\cdots\cdots+a_{2n}x_n=b_2 \\ \quad\vdots \qquad\qquad\qquad\qquad \vdots \\ a_{m1}x_1+a_{m2}x_2+\cdots\cdots+a_{mn}x_n=b_m \end{array}\right\} \tag{2.4}$$

制約条件が不等式 (2.5) で与えられる場合，スラック変数 (Slack variable) を導入することにより等式 (2.6) とすることが出来る。

$$a_{11}x_1+a_{12}x_2+\cdots\cdots+a_{1n}x_n\leqq b_1 \tag{2.5}$$

$$\left.\begin{array}{r} a_{11}x_1+a_{12}x_2+\cdots\cdots+a_{1n}x_n+x_{n+1}=b_1 \\ x_{n+1}\geqq 0 \end{array}\right\} \tag{2.6}$$

x_{n+1}；スラック変数

また，制約条件が等式 (2.7) の場合，これを等不等式 (2.8) に変換出来る。

$$a_{11}x_1+a_{12}x_2+\cdots\cdots+a_{1n}x_n=b_1 \tag{2.7}$$

$$\left.\begin{array}{l} a_{11}x_1+a_{12}x_2+\cdots\cdots+a_{1n}x_n\leqq b_1 \\ a_{11}x_1+a_{12}x_2+\cdots\cdots+a_{1n}x_n\geqq b_1 \end{array}\right\} \tag{2.8}$$

ところで，線型計画において変数が整数値しか取らないという制約条件がある場合，これを整数型線型計画法 (Integer linear programs) とよぶ。例えば，生産に使われる材料の大きさを変数とする場合，それが石油，水であれば随意に分割できるから変数は実数として扱える。これに対し，トラック，航空機の場合はそれらは分割できないから変数は整数でなければ意味がない。さらに，整数型線型計画法において，$x_i\leqq 1$ ($i=1$，2，……n) という条件

を加えることにより特別なタイプの問題となる。変数は0か1かのいずれかの値しか取りえない。これが0-1線型計画法である。

起こりうる可能性が2つしかない場合，例えば，Yes か No か，有か無かなどの二者択一の事象を変数 x_i の値，0と1とに対応させることにより，問題を数理的に表現する手法として使うことが出来る。線型計画の中でも，この0-1計画は，その適用分野，解法に特徴があるので，数理計画問題の1つの分野を形づくっている。典型的な0-1計画を歴史的に有名な事例により説明する[29]。

ナップザック問題

「ナップザックには重量30kgまでの荷物しか入らない。今，8種類の商品があり，それぞれの重さと価格は表—2.1のとおりである。それらの中から見本として幾種類かを選び，ナップザックに詰める場合，全体の価格が最大になるように商品を選べ」

表—2.1　商品の重さと価格

商品名	1	2	3	4	5	6	7	8
重さ(kg)	1	4	17	2	3	4	13	3
価格(円)	35	85	135	27	94	10	140	25

この問題は次のとおり線型計画法により定式化される。

まず，ナップザックに詰める8種類の商品のそれぞれの個数を x_1, x_2, ……, x_8 とする。

制約条件1　重量の制約

$$1x_1+4x_2+17x_3+2x_4+3x_5+4x_6+13x_7+3x_8 \leqq 30 \tag{2.9}$$

制約条件2　商品見本はたかだか1個しか持ってはいけない上，しかも端数は許されないから変数は0か1のいずれかの値しかとらない。

$$x_i=0 \text{ or } 1 \quad (i=1,\ 2, \cdots\cdots,\ 8) \tag{2.10}$$

この制約条件のもとで目的関数である商品見本の総価格を最大にする。

目的関数

$$Z=35x_1+85x_2+135x_3+27x_4+94x_5+10x_6+140x_7+25x_8 \quad \max \tag{2.11}$$

制約条件 (2.9), (2.10), 目的関数 (2.11) はいずれも変数 x_i の1次式であり，しかも変数 x_i のとる値は0か1かに限定されるから，これは0-1計画である。これを解くことにより選ぶべき商品は定まる。

2　0-1 線型計画法の解法

ナップザック問題で例示された0-1計画を一般式で表現すると，次の制約条件 (2.12), (2.13) 式のもとで，目的関数 (2.14) 式を最大（または最小）にする x_j の値を求めることになる。ここでは最大化問題をとりあげ，その解法を説明する。なお，最小化問題を最大化問題に変換する方法，あるいはその逆の変換方法は，次節で説明する。

制約条件

$$\sum_{j=1}^{n} a_{ij}x_j \leqq b_i \quad (i=1,\ 2, \cdots\cdots, m) \tag{2.12}$$

$$x_j=0 \text{ or } 1 \quad (j=1,\ 2, \cdots\cdots, n) \tag{2.13}$$

目的関数

$$Z=\sum_{j=1}^{n} c_j x_j \quad \max \tag{2.14}$$

n; 変数の個数

m; 制約条件式の個数

a_{ij}, b_i, c_j は係数

さて，この問題において変数 x_j が (2.13) 式を満足させる場合を解，(2.12)，(2.13) 式を満足させる場合を実行可能解，(2.12)，(2.13) および (2.14) 式を満足させる場合を最適解とよぶ。ところで，この問題の解の数，すなわち，x_j ($j=1, 2, \cdots\cdots, n$) が 0 か 1 かの値をとる組合せは 2^n であり，有限個である。したがって，解を網羅的にすべて列挙して，それぞれの解について実行可能性——(2.12) 式を満足させるか，および最適性——(2.14) 式を満足させるかを調べることにより最適解を（もし存在すれば）見つけることが出来る。このように，解を網羅して調べる解法を列挙法 (Enumeration) とよぶ。しかし，何の工夫もなく，悉皆的 (Exhaustive explicit enumeration) に列挙する方法では，変数の数が増加すると列挙すべき解の数が指数的に増加するので計算時間が長くなり，実用的には役に立たなくなる[5]。そこで，この考え方を基盤にした方法であるが，計算時間を短縮するために陰的列挙 (Implicit enumeration) とよぶアルゴリズムがある。1つの解の実行可能性，最適性を調べた結果，他のある解の実行可能性か，最適性かを否定出来るならば，それを列挙する必要はない。最適解でないことが事前に判明している全く魅力のない解の列挙を省略すれば，無意味な計算を避けることが出来る。省略した解は暗黙 (Implicit) の内に列挙したとよび，その解は最適解ではないと結論 (Fathom) する。これが陰的列挙の考え方である。以下に列挙法の解法として悉皆列挙法と，陰的列挙法であるグリーンバーグの解法，バラスの解法を概説する。

(1)　悉皆列挙法

悉皆列挙法は

(i)　全ての解を残らず列挙して調べ，

(ii)　その中から実行可能解を選び出し，

(iii)　実行可能解の中で目的関数の値が最大のものを最適解とする。

この方法は理論が単純明快であり，列挙法の基本構造を良く示している。前にも述べたが，この方法は変数の個数が多くなると計算量が厖大になるので実用的には使えない。しかし，これは陰的列挙法を理解するのに役立つ。今3個の変数を持つ 0-1 計画の解を列挙すると，それは 0 と 1 とからなる 3 個の数列で表現できる。この数列を 2 進数による記数法と考え，これを10進数に変換すれば 1 つの整数となるから，それぞれの解を 0 から 7 までの整数により番号づけることが出来る[31]。

$$\left.\begin{array}{l} 0,\ 0,\ 0=0 \\ 0,\ 0,\ 1=1 \\ 0,\ 1,\ 0=2 \\ \vdots \\ 1,\ 1,\ 1=7 \end{array}\right\} \tag{2.15}$$

これら 8 個の解を調べ，まず実行可能解を見つける。さらにそのうちで，目的関数が最大で

ある最適解を見つければよい。次の数値例により説明する。

制約条件

$$\left.\begin{array}{l} 5x_1+6x_3\leqq 9 \\ 4x_1+3x_2\leqq 6 \\ 4x_1+4x_2+3x_3\geqq 2 \\ 4x_1+3x_2\geqq 3 \\ x_1,\ x_2,\ x_3=0 \text{ or } 1 \end{array}\right\} \tag{2.16}$$

目的関数

$$Z=3x_1+6x_2+2x_3 \quad \max \tag{2.17}$$

変数 x_1, x_2, x_3 について0, 1の組合せを作り，すべてを列挙すると表—2.2のとおり，8個の解が出来る。それらの中で制約条件 (2.16) 式を満足させる実行可能解は No. 2, No. 3, No. 4の3個である。これら実行可能解の中で目的関数Zが最大である No. 3 が最適解である。

表—2.2 解の悉皆列挙

No.	x_1	x_2	x_3	実行可能性	Z	最適解
0	0	0	0	×	0	
1	0	0	1	×	2	
2	0	1	0	○	6	
3	0	1	1	○	8	◎
4	1	0	0	○	3	
5	1	0	1	×	5	
6	1	1	0	×	9	
7	1	1	1	×	11	

注1) 変数は3個で，解は8個 ($=2^3$)
注2) 制約条件を満足させる実行可能解は，No. 2, No. 3, No. 4
注3) その中で目的関数Zの値が最大である No. 3 が最適解。

(2) グリーンバーグの解法

H. Greenberg はこの解法を整数型線型計画の解法として提案したが，(2.19) 式において k を1とすると0-1計画の解法となる[6]。問題は制約条件 (2.18), (2.19) 式と目的関数 (2.20) 式である。

制約条件

$$\sum_{j=1}^{n} a_{ij}x_j \leqq b_i \quad (i=1,\ 2,\cdots\cdots,\ m) \tag{2.18}$$

$$\left.\begin{array}{ll} 0\leqq x_j\leqq k & (j=1,\ 2,\cdots\cdots,\ n) \\ k=1 & \end{array}\right\} \tag{2.19}$$

目的関数

$$Z=\sum_{j=1}^{n} c_j x_j \quad \min \tag{2.20}$$

x_j; 整数, m; 制約式の数, n; 変数の数

k; 変数の上限値, a_{ij}, b_i, c_j; 係数

(i)　まず，目的関数Zの小さな解から，単調増加列になる順序で解を列挙する。
(ii)　その列挙過程で，最初に見つかった実行可能解が最適解である。
(iii)　以下の列挙はすべて省略する。
この方法を最大化問題に置き換えて説明する。悉皆列挙法で扱った表―2.2の解を目的関数Zの値が単調減少列になるように並べかえたのが表―2.3である。そこで，最初の解から順次，その実行可能性を調べると，No. 3が初めての実行可能解であった。これが最適解である。この方法は，Zの値が単調減少列となるような順序で解を作り出す計算手順を必要とする。

表―2.3　グリーンバーグの解法

No.	x	x	x	Z	実行可能性	最適解
7	1	1	1	11	×	
6	1	1	0	9	×	
3	0	1	1	8	○	◎
2	0	1	0	6	↓	
5	1	0	1	5	以下の列挙は	
⋮		⋮		⋮	不要	

注1)　Zの値を大から小へ並べかえる。
注2)　No. 3が初めての実行可能解であり，それが最適解である。
注3)　以下の列挙は省略する。

3　バラスの解法

(1)　問題の標準化

バラスの解法（Additive algorithm）を適用するには一般0-1計画を次の型に標準化しなければならない[2)3)]。

制約条件

$$\sum_{j=1}^{n} a_{ij}x_j \geqq b_i \quad (i=1,\ 2,\cdots\cdots,m) \tag{2.21}$$

$$x_j=0 \text{ or } 1 \quad (j=1,\ 2,\cdots\cdots,n) \tag{2.22}$$

目的関数

$$Z=\sum_{j=1}^{n} c_j x_j \quad \min \tag{2.23}$$

$$\text{ただし } c_j \geqq 0 \ (j=1,\ 2,\cdots\cdots,\ n) \tag{2.24}$$

この標準化された問題の特徴は
(i)　目的関数の係数は，すべて非負であり，かつ，目的関数の最小化問題である。
(ii)　制約条件式がすべて≧向きの等不等式である，という点にある。
最大化問題を最小化問題に，そして目的関数の係数 c_j の非負への変換は次の(2.25)式による。

$$x'_j=\begin{cases} x_j & c_j \geqq 0 \quad \text{当初の問題が最小化問題の場合} \\ x_j & c_j \leqq 0 \quad 〃 \quad \text{最大化} \quad 〃 \\ 1-x_j & c_j < 0 \quad 〃 \quad \text{最小化} \quad 〃 \\ 1-x_j & c_j > 0 \quad 〃 \quad \text{最大化} \quad 〃 \end{cases} \tag{2.25}$$

x；変換前　　x'；変換後

なお，バラスの原論文(2)では制約条件式にスラック変数 y_i を導入して (2.26) 式のとおり正規化している。理論そのものは同じであるが，説明が容易になるので，ここでは標準化された制約式 (2.21) を用いる。

制約条件

$$\left.\begin{array}{l}\sum_{j=1}^{n} a_{ij}x_j+y_i=b_i \quad (i=1,\ 2,\cdots\cdots,m)\\ \qquad\qquad y_i\geqq 0\end{array}\right\} \tag{2.26}$$

制約条件 (2.21)，(2.22)，目的関数 (2.23) の形式の 0-1 計画を陰的列挙という考え方により解く方法を説明する。

(2)　陰的列挙

ある1つの解が最適解ではないという証拠を見つけると，その列挙を省略しても実際に列挙したと同じ結果となる。そして計算量を少なくすることが出来る。このように，解を列挙して調べる場合，最適解でないことが事前に判明している解の列挙を省略することを陰的列挙という。これに対し，実際に調べることを陽的列挙 (Explicit enumeration) とよぶ。次の問題により，陰的列挙の事例を示す[29]。

制約条件

$$\left.\begin{array}{ll}(1) & -x_1+2x_2+6x_3-2x_4-3x_5\geqq 0\\ (2) & -x_1\qquad\quad +3x_3+2x_4-2x_5\geqq 2\\ (3) & -x_1+5x_2-4x_3+x_4+2x_5\geqq 5\end{array}\right\} \tag{2.27}$$

目的関数

$$Z=3x_1+8x_2+x_3+16x_4+x_5 \quad \min \tag{2.28}$$

表—2.4　解の一覧

解の番号	x_1	x_2	x_3	x_4	x_5	解の番号	x_1	x_2	x_3	x_4	x_5
①	0	0	0	0	0	⑰	1	0	0	0	0
②	0	0	0	0	1	⑱	1	0	0	0	1
③	0	0	0	1	0	⑲	1	0	0	1	0
④	0	0	0	1	1	⑳	1	0	0	1	1
⑤	0	0	1	0	0	㉑	1	0	1	0	0
⑥	0	0	1	0	1	㉒	1	0	1	0	1
⑦	0	0	1	1	0	㉓	1	0	1	1	0
⑧	0	0	1	1	1	㉔	1	0	1	1	1
⑨	0	1	0	0	0	㉕	1	1	0	0	0
⑩	0	1	0	0	1	㉖	1	1	0	0	1
⑪	0	1	0	1	0	㉗	1	1	0	1	0
⑫	0	1	0	1	1	㉘	1	1	0	1	1
⑬	0	1	1	0	0	㉙	1	1	1	0	0
⑭	0	1	1	0	1	㉚	1	1	1	0	1
⑮	0	1	1	1	0	㉛	1	1	1	1	0
⑯	0	1	1	1	1	㉜	1	1	1	1	1

注）変数は5個であるから解は32（$=2^5$）個

この問題の解は表—2.4のとおり全部で32個である。これらの解のすべてを陽的列挙すれば，先に説明した悉皆列挙法となる。

まず，陰的列挙に必要な用語を説明する。

(i)　表—2.4の解のうち，変数 $x_1=1$ である解は⑰から㉜までである。これら⑰～㉜の16個の解は部分解 $x_1=1$ (Partial solution) の完備解 (Completion) とよぶ。

(ii)　変数 x_1 は1であるのに対し，残りの変数 x_2, x_3, x_4, x_5 は0か1かのいずれかの値を自由に取りうるので，これらを自由変数とよぶ。

(iii)　部分解 $x_1=1$ の自由変数 x_2, x_3, x_4, x_5 をそれぞれ0か1かに定めて完備解を作ることを完備化という。

(iv)　とりわけ，すべての自由変数を0とする解 $x_2=x_3=x_4=x_5=0$ を部分解 $x_1=1$ の0(ゼロ)完備解という。

さて，列挙は解①から始める。

(i)　解① $x_1=x_2=x_3=x_4=x_5=0$ が制約条件を満足させるかを調べると，次のとおり制約条件(2), (3)を満足させない。

$$\left.\begin{array}{ll} & \overset{x_1}{-1(0)}\overset{x_2}{+2(0)}\overset{x_3}{+6(0)}\overset{x_4}{-2(0)}\overset{x_5}{-3(0)}=0\geqq 0 \\ (1) & -1(0)\qquad\quad +3(0)+2(0)-2(0)=0\ngeqq 2 \\ (2) & -1(0)+5(0)-4(0)+1(0)+2(0)=0\ngeqq 5 \\ (3) & \end{array}\right\} \tag{2.29}$$

(ii)　制約条件(2)を満足させるには，その係数が正である x_3, x_4 を，制約条件(3)では，x_2, x_4, x_5 の少なくとも1つを1にしなければならない。そこで，例えば，$x_3=1$ として他の変数は0のままとする解⑤，$x_3=1$, $x_1=x_2=x_4=x_5=0$ が制約条件を満足させるかを調べる。

条件(3)を満足させない。

$$\left.\begin{array}{ll} & \overset{x_1}{-1(0)}\overset{x_2}{+2(0)}\overset{x_3}{+6(1)}\overset{x_4}{-2(0)}\overset{x_5}{-3(0)}=6\geqq 0 \\ (1) & -1(0)\qquad\quad +3(1)+2(0)-2(0)=3\geqq 2 \\ (2) & -1(0)+5(0)-4(1)+1(0)+2(0)=-4\ngeqq 5 \\ (3) & \end{array}\right\} \tag{2.30}$$

(iii)　制約条件(3)を満足させるには係数が正である変数 x_2, x_4, x_5 の少なくとも1個を1にしなければならない。しかし，x_2, x_4, x_5 をすべて1にしても制約条件(3)は満足されない。

$$(3)\quad \overset{x_1}{-1(0)}\overset{x_2}{+5(1)}\overset{x_3}{-4(1)}\overset{x_4}{+1(1)}\overset{x_5}{+2(1)}=4\ngeqq 5 \tag{2.31}$$

このことは，$x_3=1$ である限り，他の変数の値がいずれであっても制約条件(3)は満足されない。したがって，部分解 $x_3=1$ の完備解はすべて実行不可能であり，最適解ではないと結論できる。これを部分解 $x_3=1$ は終決 (Fathom) したという。部分解 $x_3=1$ の完備解⑥，⑦，⑧，⑬，⑭，⑮，⑯，㉑，㉒，㉓，㉔，㉙，㉚，㉛，㉜はいずれも実行不可能であるという証拠を得たので，これら15個の解を実行可能性がないことにより陰的列挙したという。

(iv)　次に，部分解 $x_3=1$ の完備解はすべて列挙され，実行不可能であるから，残りは，部分解 $x_3=0$ の完備解の中から最適解を見つける。解① $x_1=x_2=x_3=x_4=x_5=0$ は制約条件

(2), (3)を満足させない。そして，制約式(2)を満足させるためには x_3, x_4. 制約条件(3)については，x_2, x_4, x_5 の少なくとも1つの変数を1にしなければならないことは既に述べた。今，$x_3=0$ であるから，残りの変数 x_2, x_4, x_5 のうちで，例えば $x_4=1$ とする。そこで解③，$x_3=0$, $x_4=1$, $x_1=x_2=x_5=0$ が制約条件を満足させるかを調べる。制約条件(1)および(3)は満足されない。

$$\left.\begin{array}{ll} & \overset{x_1}{} \quad \overset{x_2}{} \quad \overset{x_3}{} \quad \overset{x_4}{} \quad \overset{x_5}{} \\ (1) & -1(0)+2(0)+6(0)-2(1)-3(0)=-2\ngeqq 0 \\ (2) & -1(0)\qquad\quad +3(0)+2(1)-2(0)=2\geqq 0 \\ (3) & -1(0)+5(0)-4(0)+1(1)+2(0)=1\ngeqq 5 \end{array}\right\} \tag{2.32}$$

(v)　制約条件(1)を満足させるには，係数が正である x_2，制約条件(3)では x_2, x_5 の少なくとも1つの変数を1にしなければならない。例えば x_2 を1とする。そこで，解⑪，$x_3=0$, $x_4=1$, $x_2=1$, $x_1=x_5=0$ は制約条件を満足させるので，これは実行可能解である。

$$\left.\begin{array}{ll} & \overset{x_1}{} \quad \overset{x_2}{} \quad \overset{x_3}{} \quad \overset{x_4}{} \quad \overset{x_5}{} \\ (1) & -1(0)+2(1)+6(0)-2(1)-3(0)=0\geqq 0 \\ (2) & -1(0)\qquad\quad +3(0)+2(1)-2(0)=2\geqq 2 \\ (3) & -1(0)+5(1)-4(0)+1(1)+2(0)=6\geqq 5 \end{array}\right\} \tag{2.33}$$

(vi)　部分解 $x_3=0$, $x_4=1$, $x_2=1$ の完備解で，自由変数 x_1, x_5 の少なくとも1個を1にした解は最適解とはなりえない。なぜなら，目的関数の係数はすべて非負であるから，自由変数 x_1, x_5 のいずれかでも1にすれば目的関数の値は，実行可能解⑪の目的関数を越えるからである。その結果，部分解 $x_3=0$, $x_4=1$, $x_2=1$ の完備解のうち解⑫，㉗，㉘，は最適解ではない証拠を得たので，これら3個の解は陰的列挙した。以上の列挙手順をまとめると表—2.5となり，4個の解を陽的列挙，18個の解を陰的列挙した。

表—2.5　陽的列挙と陰的列挙

手順	陽的列挙	陰的列挙
(i)	①	
(ii)	⑤	
(iii)		⑥ ⑦ ⑧ ⑬ ⑭ ⑮ ⑯ ㉑ ㉒ ㉓ ㉔ ㉙ ㉚ ㉛ ㉜
(iv)	③	
(v)	⑪	
(vi)		⑫ ㉗ ㉘

注1)　①～㉜は各手順で陽的列挙，あるいは陰的列挙された解。
注2)　⑪は実行可能解。
注3)　②，④，⑨，⑩，⑰，⑱，⑲，⑳，㉕，㉖はまだ列挙されていない。

(3)　Solution tree

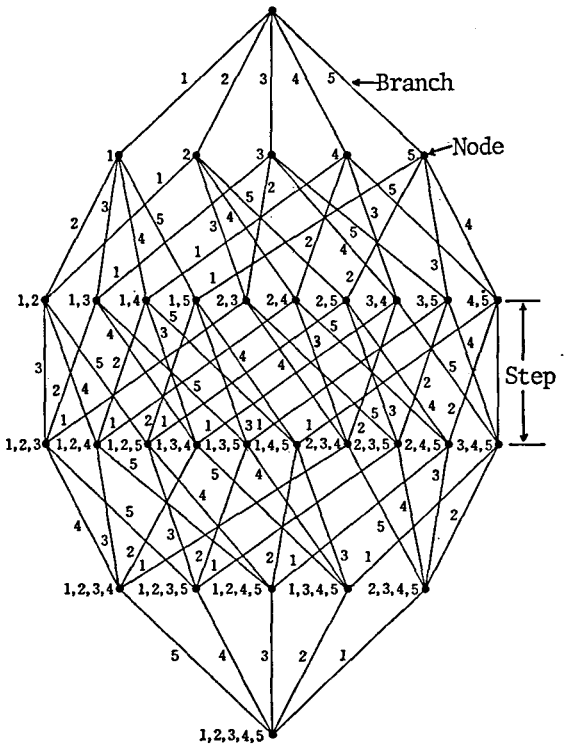
陰的列挙は次の3つの事項に整理できる。

(i)　列挙対象となる解の一覧の作成

(ii)　列挙順序を決める規則

(iii)　実行可能性，あるいは最適性がないと判断して列挙を省略する基準。

まず，列挙対象になる解を系統的に図示したものが Solution tree（解の木）である。5個の変数 x_1, x_2, x_3, x_4, x_5 の Solution tree は図—2.1である[29]。

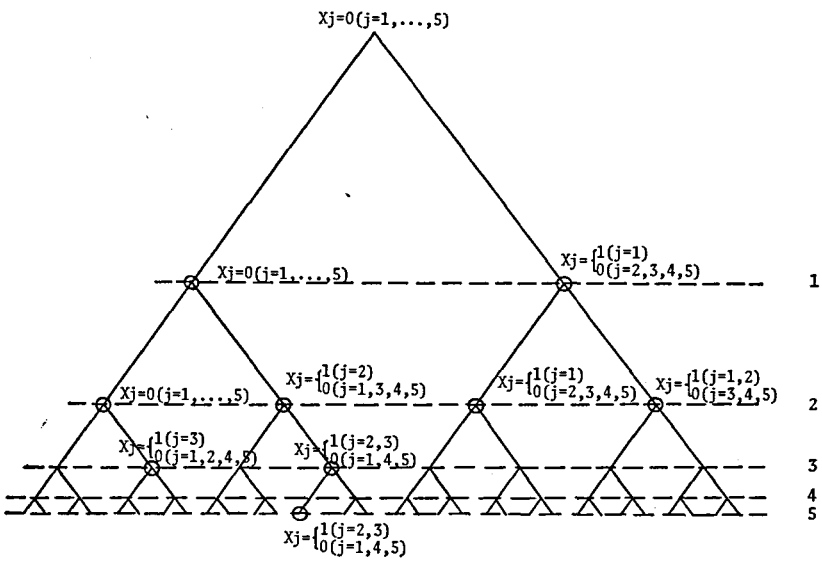


図—2.1　バラスの Solution tree

Solution tree の Node (結び目) は，解に対応し，各 Node に付された数値は，その解で値が1である変数名である。Branch (分枝) は次に，値を1にする変数名である。Node と Branch とは規則正しく階層的に組立てられており，その階層を Step という。5個の変数の Solution tree は5 Step からできている。

Solution tree はすべての解を網羅しているだけでなく，部分解と完備解との関係を示している。

Node (1, 3) は部分解 $x_1=x_3=1$ の0完備解，$x_1=x_3=1$ $x_2=x_4=x_5=0$ に対応し，この Node より下にあって，Branch で結ばれたすべての Node はこの部分解の完備解に対応する。バラスの Additive algorithm は，この Solution tree のすべての Node を陽的列挙か，陰的列挙かのいずれかにより調べ尽くすことにより最適解を求める解法である。そしてその列挙計算が加減算のみで行うので "Additive" という。ただし，バラスの原著(2)の Solution tree は図—2.2である。しかし，その理論を説明するには図—2.1が適しているの



図—2.2　バラスの Solution tree (原著)

でこれを用いる。

(4) 列挙順序

Solution tree の頂点の Node から列挙を始めることは既に述べた。各 Node から分岐する Branch は値を1にしうる自由変数に対応する。これら自由変数のうち，いずれを1にするかの選択基準により Node の列挙順序は決まる。そこで，まず列挙順序に必要な用語を説明する。

(i) 第 k 回目に列挙する部分解を S_k とする。これは，値1を与えられた変数の集合である。S_k に含まれない変数，すなわち，自由変数の集合を S'_k とする。図—2.1の Node (1,2) を例にするなら

$$\left.\begin{array}{l} S_k=(x_1=1,\ x_2=1)\ \text{or}\ (1,\ 2) \\ S'_k=(x_3=0,\ x_4=0,\ x_5=0)\ \text{or}\ (3,\ 4,\ 5) \end{array}\right\} \tag{2.34}$$

(ii) 自由変数をすべて0とした部分解 S_k の0完備解が制約条件を満足させるかを調べ，満足されない制約条件の集合を V_k とする。制約条件は (2.27) 式とする。

制約条件

$$\left.\begin{array}{ll} & \overset{x_1}{}\quad\ \overset{x_2}{}\quad\ \overset{x_3}{}\quad\ \overset{x_4}{}\quad\ \overset{x_5}{} \\ (1) & -1(1)+2(1)+6(0)-2(0)-3(0)=1\geqq 0 \\ (2) & -1(1)\qquad\quad +3(0)+2(0)-2(0)=-1\not\geqq 2 \\ (3) & -1(1)+5(1)-4(0)+1(0)+2(0)=4\not\geqq 5 \end{array}\right\} \tag{2.35}$$

したがって，$V_k=(2,3)$ (2.36)

(iii) 自由変数 S'_k の中で，V_k の中の少なくとも1つの制約条件において正の係数を持つ変数の集合を T_k とする。

V_k に含まれる制約条件は，

$$\begin{array}{ll} (2) & -x_1\qquad +3x_3+2x_4-2x_5\geqq 2 \\ (3) & -x_1+5x_2-4x_3+x_4+2x_5\geqq 5 \end{array} \tag{2.37}$$

したがって，$T_k=(x_3,\ x_4,\ x_5)$ (2.38)

(iv) 現在までに見つけた実行可能解のうちで，その目的関数の値が最小の解を $\overline{X}$，その目的関数の値を $\overline{Z}$ とする。

列挙順序を説明する。

(i) Solution tree の頂点の Node にあたる解 $x_1=x_2=x_3=x_4=x_5=0$ が制約条件を満足させるかを調べる。これは制約条件(2)と(3)を満足させないから実行可能解ではない。

$$\left.\begin{array}{ll} & \overset{x_1}{}\quad\ \overset{x_2}{}\quad\ \overset{x_3}{}\quad\ \overset{x_4}{}\quad\ \overset{x_5}{} \\ (1) & -1(0)+2(0)+6(0)-2(0)-3(0)=0\geqq 0 \\ (2) & -1(0)\qquad\quad +3(0)+2(0)-2(0)=0\not\geqq 2 \\ (3) & -1(0)+5(0)-4(0)+1(0)+2(0)=0\not\geqq 5 \end{array}\right\} \tag{2.39}$$

$$\left.\begin{array}{l} S_1=(\phi)\quad \phi;\ \text{空集合} \\ S'_1=(1,\ 2,\ 3,\ 4,\ 5) \\ V_1=(2,\ 3) \\ T_1=(2,\ 3,\ 4,\ 5) \end{array}\right\} \tag{2.40}$$

T_1 に含まれない変数 x_1 は 制約条件を満足さすのに貢献しないから選択対象から除く。
T_1 に含まれる変数 x_2, x_3, x_4, x_5 のうち，いずれを1にするかの選択基準として，その変数について制約式の係数の和を用いる。

$$\text{係数和}\quad \left.\begin{array}{l} x_2: 2+0+5=7 \\ x_3: 6+3-4=5 \\ x_4: -2+2+1=1 \\ x_5: -3-2+2=-3 \end{array}\right\} \tag{2.41}$$

係数和が最大の x_2 を選択する。なぜなら，1つの変数の係数がある制約式で正であれば，その変数を1にすることは，その制約式を満足させるのに貢献するのに対し，負であれば貢献しないからである。T_1 に属する変数のうちで，すべての制約条件を満足させるのに貢献する可能性の大きさとして係数和を用いる。したがって，すべての制約条件を満足させる可能性の最も大きい変数を優先して選択することを意味する。

(ii)　次に列挙する解は，$x_2=1$, $x_1=x_3=x_4=x_5=0$ であり，図—2.1の Solution tree の Node (2)へ進む。この解は制約条件(2)を満足させない。

$$\left.\begin{array}{ll} & \overset{x_1}{-1(0)}+\overset{x_2}{2(1)}+\overset{x_3}{6(0)}-\overset{x_4}{2(0)}-\overset{x_5}{3(0)}=2\geqq 0 \\ (1) & -1(0)\qquad\quad +3(0)+2(0)-2(0)=0\ngeqq 2 \\ (2) & -1(0)+5(1)-4(0)+1(0)+2(0)=5\geqq 5 \\ (3) & \end{array}\right\} \tag{2.42}$$

$$\left.\begin{array}{l} S_2=(2) \\ S_2'=(1,\ 3,\ 4,\ 5) \\ V_2=(2) \\ T_2=(3,\ 4) \end{array}\right\} \tag{2.43}$$

$$\text{係数和}\quad \left.\begin{array}{l} x_3:\ \ 6+3-4=5 \\ x_4:\ -2+2+1=1 \end{array}\right\} \tag{2.44}$$

次に係数和が最大の x_3 を選ぶ。

(iii)　次に列挙する解は，$x_2=1$, $x_3=1$, $x_1=x_4=x_5=0$ であり，Node (2,3) へ進む。これは制約条件(3)を満足させない。

$$\left.\begin{array}{ll} & \overset{x_1}{-1(0)}+\overset{x_2}{2(1)}+\overset{x_3}{6(1)}-\overset{x_4}{2(0)}-\overset{x_5}{3(0)}=8\geqq 0 \\ (1) & -1(0)\qquad\quad +3(1)+2(0)-2(0)=3\geqq 2 \\ (2) & -1(0)+5(1)-4(1)+1(0)+2(0)=1\ngeqq 5 \\ (3) & \end{array}\right\} \tag{2.45}$$

$$\left.\begin{array}{l} S_3=(2,\ 3) \\ S'_3=(1,\ 4,\ 5) \\ V_3=(3) \\ T_3=(4,\ 5) \end{array}\right\} \tag{2.46}$$

さらに，T_3 に属するすべての変数 x_4, x_5 を1にしても制約条件(3)は満足されない。

$$(3)\quad \overset{x_1}{-1(0)}+\overset{x_2}{5(1)}-\overset{x_3}{4(1)}+\overset{x_4}{1(1)}+\overset{x_5}{2(1)}=4\ngeqq 5 \tag{2.47}$$

したがって，部分解 $x_2=x_3=1$ の完備解はすべて実行可能性がないので終決した。Node

(2,3) の下に連なるすべての Node (1, 2, 3), Node (2, 3, 4), Node (2, 3, 5), Node (1, 2, 3, 4), Node (1, 2, 3, 5), Node (2, 3, 4, 5), Node (1, 2, 3, 4, 5) は陰的に列挙した。

(iv) Node (2, 3) より下には，もはや列挙対象となる Node はないので Node (2) へ戻る。Node (2) で選択可能で残された Node は x_4 のみであり，Node (2, 4) へ進む。解は，$x_2=1$, $x_4=1$, $x_1=x_3=x_5=0$ である。

$$\left.\begin{array}{ll} (1) & -1\overset{x_1}{(0)}+2\overset{x_2}{(1)}+6\overset{x_3}{(0)}-2\overset{x_4}{(1)}-3\overset{x_5}{(0)}=0\geqq 0 \\ (2) & -1(0)\qquad\quad +3(0)+2(1)-2(0)=2\geqq 2 \\ (3) & -1(0)+5(1)-4(0)+1(1)+2(0)=6\geqq 5 \end{array}\right\} \tag{2.48}$$

$$\left.\begin{array}{l} S_4=(2,\ 4) \\ S'_4=(1,\ 3,\ 5) \\ V_4=\phi \end{array}\right\} \tag{2.49}$$

これは実行可能解であり，現在までに見つけた最良の解である。

$$\left.\begin{array}{l} \overline{X}=(x_1=0,\ x_2=1,\ x_3=0,\ x_4=1,\ x_5=0) \\ \overline{Z}=3(0)+8(1)+1(0)+16(1)+1(0)=24 \end{array}\right\} \tag{2.50}$$

部分解 $x_2=1$, $x_4=1$ の完備解は（0完備解を除いて）すべて最適性がないので終決した。

Node (2, 4) に連なるすべて Node (1, 2, 4), Node (2, 4, 5), Node (1, 2, 4, 5) は陰的に列挙した。

Node (2, 4) より下には列挙対象となる Node はないので Node (2) へ戻る。Node (2) にも列挙対象はないので頂点の Node (ϕ) へ戻る。

(v) Node (ϕ) で選択対象として残された変数は x_3, x_4, x_5 である。係数和の最大である x_3 を選び，Node (3) へ進む。

$$\text{係数和}\quad \left.\begin{array}{l} x_3=5 \\ x_4=1 \\ x_5=-3 \end{array}\right\} \tag{2.51}$$

$$\left.\begin{array}{ll} (1) & -1\overset{x_1}{(0)}+2\overset{x_2}{(0)}+6\overset{x_3}{(1)}-2\overset{x_4}{(0)}-3\overset{x_5}{(0)}=6\geqq 0 \\ (2) & -1(0)\qquad\quad +3(1)+2(0)-2(0)=3\geqq 2 \\ (3) & -1(0)+5(0)-4(1)+1(0)+2(0)=-4\ngeqq 5 \end{array}\right\} \tag{2.52}$$

$$\left.\begin{array}{l} S_5=(3) \\ S'_5=(1,\ 2,\ 4,\ 5) \\ V_5=(3) \\ T_5=(2,\ 4,\ 5) \end{array}\right\} \tag{2.53}$$

さらに，T_5 に属するすべての変数 x_2, x_4, x_5 を1にしても制約条件(3)は満足されない。

$$-1\overset{x_1}{(0)}+5\overset{x_2}{(1)}-4\overset{x_3}{(1)}+1\overset{x_4}{(1)}+2\overset{x_5}{(1)}=4\ngeqq 5 \tag{2.54}$$

部分解$x_3=0$ の完備解は終決した。

(vi) Node (ϕ) に戻り，選択対象として残された x_4, x_5 のうち，x_4 を選び，Node (4) へ

進む。

$$\left.\begin{array}{ll} & \overset{x_1}{-1(0)}\overset{x_2}{+2(0)}\overset{x_3}{+6(0)}\overset{x_4}{-2(1)}\overset{x_5}{-3(0)}=-2\not\geqq 0 \\ (1) & \\ (2) & -1(0)\qquad\quad +3(0)+2(1)-2(0)=2\geqq 2 \\ (3) & -1(0)+5(0)-4(0)+1(1)+2(0)=1\not\geqq 5 \end{array}\right\} \tag{2.55}$$

$$\left.\begin{array}{l} S_6=(4) \\ S'_6=(1,\ 2,\ 3,\ 5) \\ V_6=(1,\ 3) \\ T_6=(2,\ 3,\ 5) \end{array}\right\} \tag{2.56}$$

T_6 に属する x_2 はすでに (ii) において，x_3 は (v) において列挙済である。

(vii)　次に列挙する部分解は，$x_4=1$，$x_5=1$ であり，Node (4, 5) へ進む。

$$\left.\begin{array}{ll} & \overset{x_1}{-1(0)}\overset{x_2}{+2(0)}\overset{x_3}{+6(0)}\overset{x_4}{-2(1)}\overset{x_5}{-3(1)}=-5\not\geqq 0 \\ (1) & \\ (2) & -1(0)\qquad\quad +3(0)+2(1)-2(1)=0\not\geqq 2 \\ (3) & -1(0)+5(0)-4(0)+1(1)+2(1)=3\not\geqq 5 \end{array}\right\} \tag{2.57}$$

$$\left.\begin{array}{l} S_7=(4,\ 5) \\ S'_7=(1,\ 2,\ 3) \\ V_7=(1,\ 2,\ 3) \\ T_7=(2,\ 3) \end{array}\right\} \tag{2.58}$$

T_7 に属する x_2 はすでに (iv) で，x_3 は (v) で列挙済である。列挙対象がなくなったので Node (ϕ) へ戻る。

(viii)　次に列挙する部分解は $x_5=1$ であり，Node (5)へ進む。

$$\left.\begin{array}{ll} & \overset{x_1}{-1(0)}\overset{x_2}{+2(0)}\overset{x_3}{+6(0)}\overset{x_4}{-2(0)}\overset{x_5}{-3(1)}=-3\not\geqq 0 \\ (1) & \\ (2) & -1(0)\qquad\quad +3(0)+2(0)-2(1)=-2\not\geqq 2 \\ (3) & -1(0)+5(0)-4(0)+1(0)+2(1)=2\not\geqq 5 \end{array}\right\} \tag{2.59}$$

$$\left.\begin{array}{l} S_8=(5) \\ S'_8=(1,\ 2,\ 3,\ 4) \\ V_8=(1,\ 2,\ 3) \\ T_8=(2,\ 3,\ 4) \end{array}\right\} \tag{2.60}$$

T_8 に属する x_2 は (ii) で，x_3 は (v) で，x_4 は (vii) で列挙済である。以上の順序により Solution tree のすべての Node を陽的あるいは陰的に列挙して，次の最適解を得た。

$$\overline{X}=(x_1=0,\ x_2=1,\ x_3=0,\ x_4=1,\ x_5=0)$$

$$\overline{Z}=24$$

(5)　省略基準

列挙の省略基準はすでに陰的列挙および列挙順序の事例により説明した。それらを箇条的に要約する。

(i)　実行可能性がないことによる省略

部分解 S_k の0完備解が1つの制約条件を満足させない場合，自由変数 S'_k のいずれを1に

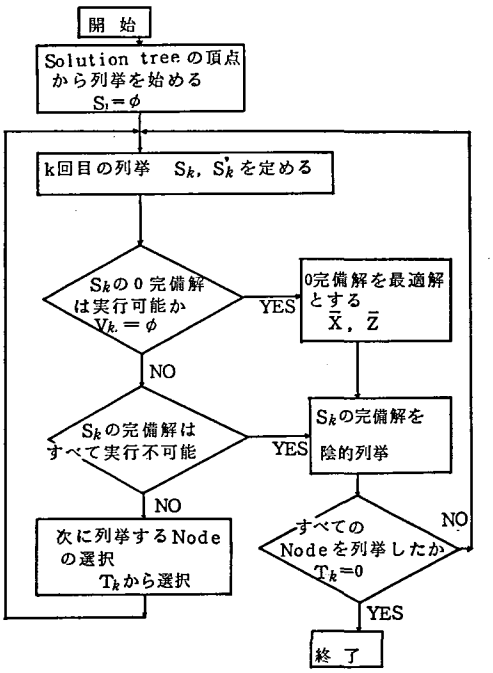
してもその制約条件を満足させないなら，部分解 S_k は実行可能性がないことにより終決する。S_k の完備解は省略する。

(ii)　最適性がないことによる省略

部分解 S_k の0完備解がすべての制約条件を満足させる場合，S_k は最適性がないことにより終決する。S_k の完備解は省略する。

(6)　計算手順

Solution tree，列挙順序および省略基準からなるバラスの解法の計算手順をフローチャートで示すと図—2.3となる。



図—2.3　バラスの解法のフローチャート

第三章　収穫予定のための0-1線型計画法の解法
——加速列挙法のアルゴリズム——

収穫予定のための0-1線型計画法は第一章で規定した。この形式の問題に対して専用的な解法が FOA (Forestry Oriented Algorithm) である[19]。

この解法は収穫予定問題に特有の附帯条件を利用して列挙の省略を行うことにより，計算時間の短縮を目的にして開発されたもので，バラスの Additive algorithm の考え方を基盤にしている。列挙の効率を高めて計算速度を速めるように改良したので，加速列挙法 (Accelerated enumeration) という。バラスの解法との対比によりその内容と特徴を説明する。

1　バラスの問題との比較

ＦＯＡの問題は第一章で，バラスの問題は第二章で規定したが，整理して対比すると次のとおりである。

〔制約条件〕

ＦＯＡの問題		バラスの問題	
$\left.\begin{array}{l}\sum_{i=1}^{n} v_{ij}x_{ij} \geqq A_j \\ \sum_{i=1}^{n} a_i x_{ij} \geqq V_j\end{array}\right\}$ $(j=1,2,\cdots\cdots,m)$	(1.9)	$\sum_{j=1}^{n} a_{ij}x_j \geqq b_i$ $(i=1,2,\cdots\cdots,m)$	(2.21)
$\left.\begin{array}{l}\sum_{i=1}^{n} v_{ij}x_{ij} \leqq V'_j \\ \sum_{i=1}^{n} a_i x_{ij} \leqq A'_j\end{array}\right\}$	(1.10)	$x_j=0$ or 1 $(j=1,2,\cdots\cdots,n)$	(2.22)

$x_{ij}=0$ or 1　(1.11)

$\sum_{j=1}^{m} x_{ij}=1$　(1.12)

$(i=1, 2, \cdots\cdots, n)$

〔目的関数〕

$Z=\sum_{j=1}^{m}\sum_{i=1}^{n} v_{ij}x_{ij}$　max　(1.13)

$Z=\sum_{j=1}^{n} c_j x_j$　min　(2.23)

〔附帯条件〕

$\left.\begin{array}{l} v_{ij}\geqq 0 \\ a_i\geqq 0 \end{array}\right\}$　(1.14)

$c_j\geqq 0$　(2.24)

$v_{ij}\leqq v_{i,j+1}$　(1.15)

バラスの問題とＦＯＡの問題との相異点は次の５点である。

(i)　ＦＯＡには制約式 (1.12) $\sum_{j=1}^{m} x_{ij}=1$ が常に存在する。これは計画期間内に各小班は１回だけ伐採することを意味する。

(ii)　制約条件 (1.9), (1.10) の係数 a_i, v_{ij} は小班の面積，材積であり，それらは非負である。

(iii)　バラスの問題の制約式 (2.21) はすべて≧向きの等不等式であるが，ＦＯＡの制約式 (1.10) は≦向きの等不等式である。不等号の向きを変えて統一することは出来るが，その結果，係数が負になる。係数がすべて非負であることを列挙の省略基準に利用するため，あえて不等号の向きの異なる制約条件を認めている。

(iv)　ＦＯＡには附帯条件 (1.15) $v_{ij}\leqq v_{i,j+1}$ がある。これは各小班の伐期材積は成長により分期とともに増加することを意味する。

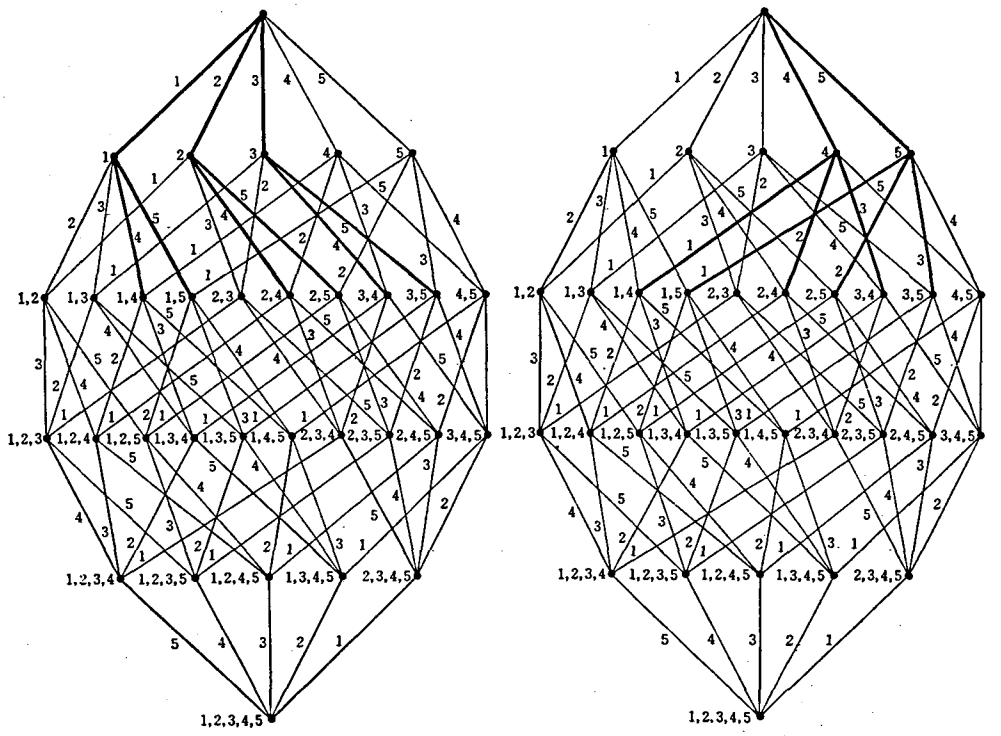
(v)　ＦＯＡは最大化問題である。

ＦＯＡはこれらの相異点を利用して列挙計算を加速するアルゴリズムを持っている。なお以下の説明では変数 x_{ij} $(i=1, 2, \cdots\cdots, n, j=1, 2, \cdots\cdots, m)$ に改めて一連番号をつけなおし，x_i と表示する。

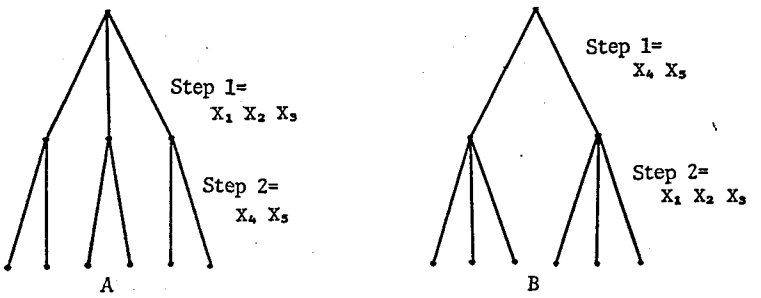
2　Solution tree の比較

変数が５個で x_1, x_2, x_3, x_4, x_5 の場合，バラスの Solution tree は32個の Node と５つの Step で出来ていることは，すでに図—2.1で示した。これを再度引用して図—3.1に示す。Solution tree の Node に相当するすべての解を陽的あるいは陰的に列挙して最適解を見つける場合，その列挙対象の Node が少なければ計算は加速される。そこで，ＦＯＡの問題が必ず備えている制約条件 (1.12) $\sum_{j=1}^{m} x_{ij}=1$ により実行不可能であることが明白である Node をあらかじめ除外することにより Solution tree のサイズを小さくする。例えば，図—3.1において制約条件 (1.12) は $x_1+x_2+x_3=1$, $x_4+x_5=1$ とする。Solution tree の中で Node (1, 4), Node (1, 5), Node (2, 4), Node (2, 5), Node (3, 4), Node (3, 5) の６個の Node のみが列挙対象になり，他の26個の Node はすべて実行可能性がない。そこで，頂点の Node からこれら６個の Node を結ぶ Branch の組合せは図—3.1. A と図—3.1. B の

バラスの Solution tree



FOA の Solution tree



図—3.1 Solution tree の対比

注1) 上のバラスの Solution tree から太線の部分を取り出したのが，下のＦＯＡの Solution tree である。

注2) ＦＯＡの Solution tree の端末の Node に実行可能解がある。

注3) 途中の Node は実行不可能な解であるが，列挙省略に使う。

二通りである。これらの図のどちらかを選びＦＯＡの Solution tree と定義する。これはバラスの Solution tree の一部を取り出したもので，サイズは著しく小さくなる。

ＦＯＡの Solution tree の各 Step は各小班に，Branch は小班の伐採分期に対応する。

図—3.1.A の場合，Step 1 は1小班に該当し，その3本の Branch は伐採分期1，2，3に該当する。Step 2 は2小班に該当する。これに対し，図—3.1.B の Step 1 は2小班に，Step 2 は1小班に該当する。いずれの Solution tree を用いても同じ結果を求めることが出来る。しかし，面積の大きな小班を上位の Step に割り当てることにより，計算を加速することが出来る。その理由は省略の基準の項で説明する。例えば，1小班面積＞2小班面積なら図—3.1.A をFOAの Solution tree と規定する。さて，新しい事例によりFOAの Solution tree の性質を説明する。

制約条件

$$8x_1+35x_2+9x_3+10x_4+12x_5+7x_6+8x_7+10x_8+12x_9 \geqq 41 \tag{3.1}$$

$$8x_1+35x_2+9x_3+10x_4+12x_5+7x_6+8x_7+10x_8+12x_9 \leqq 45 \tag{3.2}$$

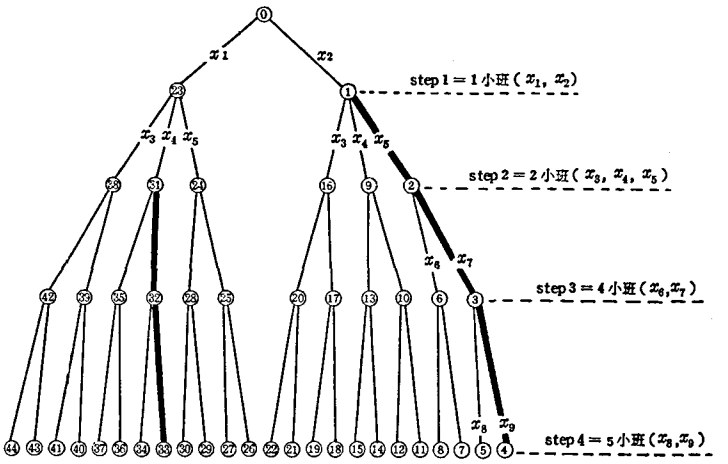
$$\left.\begin{array}{ll} 1\text{小班} & x_1+x_2=1 \\ 2\text{小班} & x_3+x_4+x_5=1 \\ 3\text{小班} & x_6+x_7=1 \\ 4\text{小班} & x_8+x_9=1 \\ x_i=0 \text{ or } 1 & (i=1,\ 2,\cdots\cdots,9) \end{array}\right\} \tag{3.3}$$

目的関数

$$Z=35x_1+45x_2+21x_3+25x_4+29x_5+11x_6+15x_7+6x_8+8x_9 \tag{3.4}$$

変数9個，小班数4でこの事例問題の Solution tree は図—3.2である。Node 番号は列挙順序である。端末の Node ④，⑤，⑦，……，㊹の中に実行可能解があり，途中の Node は列挙省略のために用いる。

ところで，この Solution tree の端末の Node に該当する解の目的関数Zを図示すると，



図—3.2　FOAの Solution tree

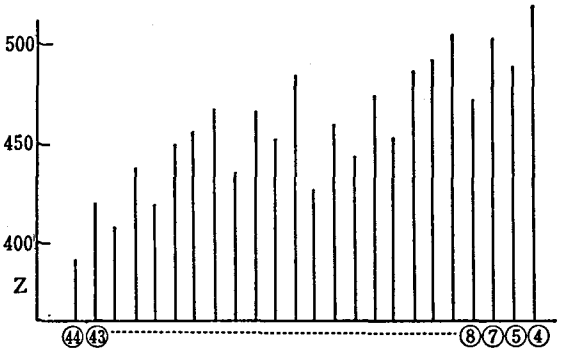
注1）Node ㉛の端末 Node ㉝，㉞，㊱，㊲の中で目的関数Zが最大になるのは㉝。

注2）Node ①の端末 Node ④，⑤，⑦，⑧，⑪，⑫，⑭，⑮，⑱，⑲，㉑，㉒の中で目的関数が最大になるのは最も右側の Node ④。

注3）小班面積は，1小班＞2小班＞3小班＞4小班とする。

図—3.3 となり，次の性質を持っている。

(i) 途中の Node ①の下で，Branch により結ばれた端末の Node ④，⑤，⑦，⑧，⑪，⑫，⑭，⑮，⑱，⑲，㉑，㉒のうち，最も右側に位置する Node ④の目的関数Zが最大である。一般に，1つの途中の Node の下に結ばれた端末の Node のうちでは，最も右側に位置する Node の目的関数Zが必ず最大である。これはＦＯＡの問題の附帯条件 (1.15)，$v_{ij} \leqq v_{i,j+1}$による。



図—3.3 端末の Node の目的関数Z
注） 端末の Node の目的関数Zは右上りの傾向を持つ。

(ii) 目的関数Zは図—3.3 のとおり右上りの傾向を持つ。列挙は右から行うから目的関数Zの大きな解から列挙を行う傾向を持つといえる。このことは，先に述べたグリーンバークの解法と共通点を持っている。

(iii) 面積の大きな小班から順次，上位の Step へ割り当てる。

ＦＯＡの Solution tree の性質を比較，要約する。

(i) ＦＯＡの Solution tree はバラスの Solution tree の一部分を取り出したものであり，サイズが小さい。

(ii) ＦＯＡの Solution tree は端末の Node に実行可能解がある。それらは右側の Node ほどZが大となる傾向があり，Solution tree の中での所在位置により Node の実行可能性，最適性に差がある。したがって列挙順序を事前に予定できる。

3 列挙順序の比較

ＦＯＡの列挙順序はバラスの方法に比べ，機械的であり，次の3つの規則による。

(i) Solution tree の頂点の Node から出発し，上から下への順序で列挙する。

(ii) 1つの Node から2つ以上の Branch があり，それらが列挙対象である時は，「最も右側の Branch」を選ぶ。

(iii) 1つの Node より下に列挙対象となる Branch がない場合は，1つだけ上位の Node へ戻る。この (i)，(ii)，(iii) の規則により決まる列挙順序が図—3.2の Node ⓪から Node ㊹である。

これに対し，バラスの列挙順序は，

(i) Solution tree の頂点の Node から出発し，上から下への順序で列挙する。

(ii) 1つの Node から2つ以上の Branch があり，それらが列挙対象である場合は「実行可能性の最も高い Branch を計算して選ぶ。」

(iii) ある Node より下に選ぶべき Branch がない場合は1つだけ上位の Node へ戻る。

したがって，列挙規則 (ii) に相異点がある。すなわち，バラスは各 Node で毎回最良と思われる Branch を計算により選びだす発見的方法 (heuristic procedure) であるが，ＦＯＡ

は，常に最も右側の Branch を選ぶ機械的方法である。図—3.3 のとおり，ＦＯＡの Solution tree では右側の解ほど目的関数 Z は大きい傾向があり，最も右側の Branch を選ぶことは最適性の大きな解を優先的に列挙することを意味する。ＦＯＡのこの列挙規則の特長は，

(i)　後で述べるが，最適性がないことによる省略基準による省略を多くできる。グリーンバークの解法の省略基準をゆるやかに適応している。

(ii)　次に，面積，材積の大きな小班から上位の Step に並べるから，制約条件に抵触するか否かの判定結果を早く発見できる。したがって，後で述べるが，実行可能性がないことによる省略基準による省略を多くすることが出来る。

(iii)　そして，機械的な順序は，列挙順序を決定するために必要な，繁雑な計算をなくして，計算時間を短縮する。

4　省略基準の比較

ＦＯＡは次の３つの省略基準を持っている。

(1)　下限制約条件に抵触する場合

これはバラスの実行可能性がないことによる省略基準に相当する。まず，比較のためにバラスの省略基準を第二章の事例 (2.52)，(2.53)，(2.54) 式を引用して再度説明する。

制約条件(3)

$$-\overset{x_1}{1}(0)+\overset{x_2}{5}(0)-\overset{x_3}{4}(1)+\overset{x_4}{1}(0)+\overset{x_5}{2}(0)=-4\ngeq 5 \tag{2.52}$$

$$\left.\begin{array}{l} S=(3) \\ S'=(1,\ 2,\ 4,\ 5) \\ V=(3) \\ T=(2,\ 4,\ 5) \end{array}\right\} \tag{2.53}$$

$$-1(0)+5(1)-4(1)+1(1)+2(1)=4\ngeq 5 \tag{2.54}$$

(2.52)，(2.53) 式は，「解 $x_3=1$，$x_1=x_2=x_4=x_5=0$ は制約条件(3)を満足させない。それを満足させる可能性をもつ変数は，x_2，x_4，x_5 である」ことを意味する。(2.54) 式は「そこで，x_2，x_4，x_5 のすべてを１にしたが，制約条件(3)を満足させない」したがって，部分解 $x_3=1$ の完備解はすべて実行可能性がないことにより省略する。これがバラスの省略基準であった。

これに対し，ＦＯＡの基準を第三章の事例 (3.1)，(3.3) 式を引用して説明する。

制約条件

$$8x_1+35x_2+9x_3+10x_4+12x_5+7x_6+8x_7+10x_8+12x_9\geqq 41 \tag{3.1}$$

$$\left.\begin{array}{l} x_1+x_2=1 \\ x_3+x_4+x_5=1 \\ x_6+x_7=1 \\ x_8+x_9=1 \\ x_i=0 \text{ or } 1\ (i=1,\ 2,\cdots\cdots,9) \end{array}\right\} \tag{3.3}$$

今，部分解 $S=(x_1=1,\ x_2=0)$ の０完備解は (3.1) 式を満足させない。

$$8(1)+35(0)+9(0)+10(0)+12(0)+7(0)+8(0)+10(0)+12(0)=8\ngeq 41 \tag{3.5}$$

(3.5) 式の係数はすべて非負であるから，自由変数を全部１にすれば，これを満足させる可

能性が最も高い。しかし，(3.3) 式により，$(x_3,\ x_4,\ x_5)$，$(x_6,\ x_7)$，$(x_8,\ x_9)$ はそれぞれ，いずれか1つの変数しか1に出来ない。そこで，それぞれの中で係数の最大のものを1にする。しかし，それは (3.6) 式のとおり，制約条件を満足しないから，部分解 S の完備解はすべて実行可能性がないとして省略する。

$$8(1)+35(0)+\max(9,\ 10,\ 12)+\max(7,\ 8)+\max(10,\ 12)=40\ngeqq 41 \tag{3.6}$$

このように，自由変数 S' について，同一小班に属する変数 $(x_3,\ x_4,\ x_5)$，$(x_6,\ x_7)$，$(x_8,\ x_9)$ の中で下限制約式の係数が最大のものを1としても，その制約式を満足しない場合，部分解 S の完備解は「下限制約条件に抵触する」として列挙を省略する。

(2) 上限制約式に抵触する場合

これはＦＯＡに特有の省略基準である。(3.2) 式の上限制約式を引用する。

制約条件

$$8x_1+35x_2+9x_3+10x_4+12x_5+7x_6+8x_7+10x_8+12x_9\nleqq 45 \tag{3.2}$$

部分解 $S=(x_1=0,\ x_2=1)$ 自由変数 $(x_3,\ x_4,\ x_5,\ x_6,\ x_7,\ x_8,\ x_9)$ において，自由変数 $x_5=1$ とした新しい部分解を調べる。新しい部分解 $S_2=(x_1=0,\ x_2=1,\ x_5=1)$ の完備解は (3.7) 式のとおり，上限制約条件を満足しない。

$$8(0)+35(1)+9(0)+10(0)+12(1)+7(0)+8(0)+10(0)+12(0)=47\nleqq 45 \tag{3.7}$$

(3.7) 式は，係数がすべて非負であるから自由変数 $(x_3,\ x_4,\ x_6,\ x_7,\ x_8,\ x_9)$ のいずれを1にしても満足しない。このように部分解 $S=(x_1=0,\ x_2=1,\ x_5=1)$ の0完備解が上限制約条件を満足しない場合，部分解 S の完備解は「上限制約条件に抵触する」として列挙を省略する。

(3) 最適性がない場合

制約条件

$$\left.\begin{aligned} &x_1+x_2=1\\ &x_3+x_4+x_5=1\\ &x_6+x_7=1\\ &x_8+x_9=1 \end{aligned}\right\} \tag{3.3}$$

目的関数

$$Z=35x_1+45x_2+21x_3+25x_4+29x_5+11x_6+15x_7+6x_8+8x_9 \tag{3.8}$$

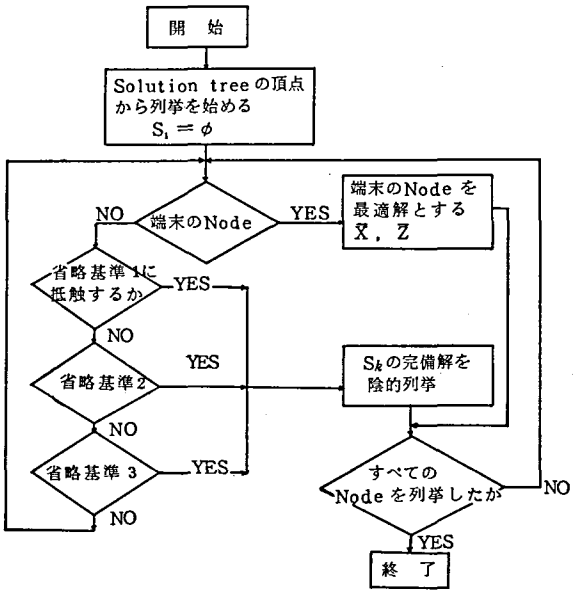
この問題については，次の実行可能解がすでに知られているとする。

$$\overline{X}=(x_1=0,\ x_2=1,\ x_3=0,\ x_4=0,\ x_5=1,\ x_6=1,\ x_7=0,\ x_8=1,\ x_9=0) \tag{3.9}$$

$$\overline{Z}=\overset{x_1}{35(0)}+\overset{x_2}{45(1)}+\overset{x_3}{21(0)}+\overset{x_4}{25(0)}+\overset{x_5}{29(1)}+\overset{x_6}{11(1)}+\overset{x_7}{15(0)}+\overset{x_8}{6(1)}+\overset{x_9}{8(0)}=91 \tag{3.10}$$

今，例えば，部分解 $S\ (x_1=1,\ x_2=0,\ x_3=1,\ x_4=0,\ x_5=0)$ 自由変数 $S'=(x_6,\ x_7,\ x_8,\ x_9)$ の完備解のうち，目的関数の最大値は (3.11) 式のとおり $Z=79$ であり，これは既知の実行可能解の $\overline{Z}$ におよばない。したがって，部分解は「最適性がない」として省略する。

$$\left.\begin{aligned} &Z=35(1)+45(0)+21(1)+24(0)+29(0)+\max(11,15)+\max(6,8)=79\\ &\quad\text{したがって，}\ Z<\overline{Z} \end{aligned}\right\} \tag{3.11}$$



図—3.4　ＦＯＡのフローチャート

これは，バラスの「最適性がないことによる省略」に相当する基準である。

5　計算手順の比較

ＦＯＡの計算手順をフローチャートで示す[19]。

ＦＯＡのフローチャートとバラス（図—2.3）とを比較すると，次のとおり要約できる。

(i)　いずれも Solution tree の頂点の Node から列挙をはじめる。

(ii)　次に値を１にする変数を選ぶのに，バラスは実行可能解を得る可能性が最も高い Branch を計算して選ぶ。ＦＯＡは機械的に最も右側の Branch を選ぶのであらかじめ列挙順序はきまっている。

(iii)　ＦＯＡでは端末の Node と途中の Node とを区別する。途中の Node は最適解となり得ないが，列挙の省略の判定に利用される。バラスは端末，途中の Node を区別しない。

(iv)　ＦＯＡは３つの省略基準を使って，最適性あるいは実行可能性がないことにより列挙を省略する。バラスは２つの省略基準を用いる。

(v)　いずれも，Solution tree のすべての Node を陽的列挙あるいは陰的列挙することにより終了する。

第四章　加速列挙法の適用方法

収穫予定を立てるには，まず，収穫予定問題を 0–1 計画そのものに定式化することからはじめる。その 0–1 計画は加速列挙法ＦＯＡのアルゴリズムを有効に利用出来るように設定されなければならない。ここで加速列挙法の適用方法を説明する。

1　問題サイズの縮小

小班数や分期数が多くなると，変数の数が多くなり，したがって列挙対象となる解の数が多くなる。列挙法にとって最も重要な評価基準である計算時間を検討するために，0–1 計画のサイズを測る尺度を規定する。

さて，バラスの解法では変数の個数が n であれば，列挙対象となる解の個数は 2^n となるので，n を問題のサイズとして用いる。これは先に述べたが，バラスの Solution tree の Node の数に相当する[2]。

これに対し，ＦＯＡでは変数の個数が同じでも次の事例，表—4.1 のように必ずしも解の個数は同じにはならない。

したがって，FOAでは列挙対象となる解の個数そのものを問題のサイズと規定する。これはFOAのSolution treeの端末のNodeの数に相当する[20]。図—4.1の場合，問題のサイズは24である。

表—4.1　変数の個数と解の個数

問　題	小班数	計画分期数	変数の個数	解の個数
(1)	10	2	20(=10×2)	1024(=2^{10})
(2)	2	10	20(=2×10)	100(=10^2)

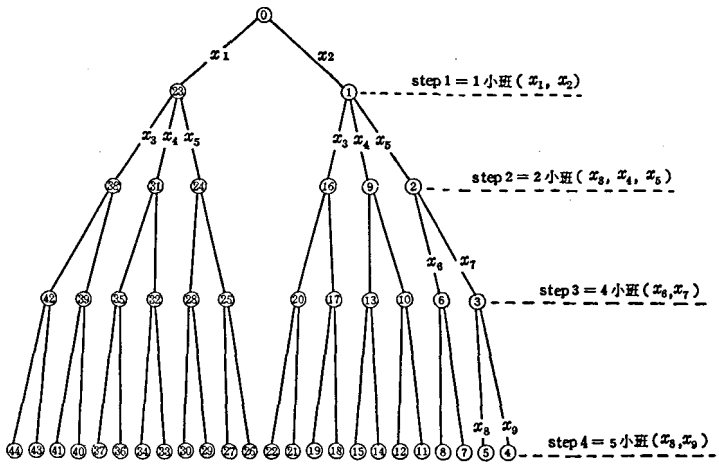


図4.1　問題のサイズ

注1)　これは，問題(3.1)，(3.2)，(3.3)のSolution treeである。端末のNode ④，⑤，……，㊸，㊹が列挙対象となる解であり，その個数が問題のサイズである。

注2)　その個数は各小班の変数の個数の積である。24＝2×3×2×2

さて，収穫予定を0-1計画そのものに定式化する時，問題サイズを出来る限り，小さくすることが必要である。このために，伐採することが事前に定まっている分期，あるいは逆に，決して伐採することが出来ない分期は変数から除外する。表—4.2を例題として，サイズの

表—4.2　当初の収穫予定

小班名＼分期	1	2	3
1	x_1	x_2	x_3
2	x_4	x_5	x_6
3	x_7	x_8	x_9
4	x_{10}	x_{11}	x_{12}
5	x_{13}	x_{14}	x_{15}

注1)　すべての小班はすべての分期で伐採の対象とする。

注2)　問題のサイズは，3^5=243

表—4.3　サイズを縮小した収穫予定

小班名＼分期	1	2	3
1	*	x_1	x_2
2	x_3	x_4	x_5
3	**	*	*
4	x_6	x_7	*
5	*	x_8	x_9

注1)　*　伐採対象とならない分期

注2)　**　必ず伐採しなければならない分期

注3)　問題のサイズは，2×3×2×2＝24

縮小方法を例示する。例えば次のような事情があるとする。

(i)　1小班の1分期は伐期に達していないので伐採対象としない。

(ii)　3小班は作業上の都合で第1分期に伐採する。

(iii)　4小班は労務上の都合で 1, 2 分期内に伐採する。したがって, 3分期は伐採対象とならない。

(iv)　5小班は林道計画上, 2分期以降に伐採する。

以上のように, 変数の値が前もって0か1かのいずれかに定まっている場合は, それらを変数から除き, 残りの変数に改めて一連番号を付けると表—4.3となる。その結果, 表—4.2の問題のサイズは243であるのに対し, 表—4.3のサイズは24となり, 大幅に縮小された。

2　小班の並べかえ

表—4.3について, 各小班の面積および伐期材積を記載したのが表—4.4である。表—4.4は小班の名前の順序により並べられているが, これを小班面積の大から小の順序により並べかえる。並べかえた後に, 改めて上から一連の変数名を与えた結果, 表—4.5が出来る。

表—4.4　小班の面積と伐採材積

小班名	面積 (ha)	1	2	3 (100m³)
1	14		x_1 29	x_2 38
2	23	x_3 40	x_4 53	x_5 64
4	80	x_6 154	x_7 185	
5	50		x_8 96	x_9 113

表—4.5　面積の大小による並べかえ

小班名	面積 (ha)	1	2	3 (100m³)
4	80	x_1 154	x_2 185	
5	50		x_3 96	x_4 113
2	23	x_5 40	x_6 53	x_7 64
1	14		x_8 29	x_9 38

このように面積の大小の順序により変数名を整理することが小班の並べかえである。並べかえは問題の内容を変えないが列挙計算を加速するのに非常に有効である。このことを, 表—4.4, 表—4.5の事例により説明する。

今, 第2分期の伐採材積の許容範囲は200から250であるとする。

表—4.4の場合の制約条件

$$\left.\begin{array}{ll}\text{上限制約式} & 29x_1+53x_4+185x_7+96x_8\leqq 250\\ \text{下限制約式} & 29x_1+53x_4+185x_7+96x_8\geqq 200\end{array}\right\}\qquad(4.1)$$

表—4.5の場合の制約条件

$$\left.\begin{array}{l}185x_2+96x_3+53x_6+29x_8\leqq 250\\ 185x_2+96x_3+53x_6+29x_8\geqq 200\end{array}\right\}\qquad(4.2)$$

この (4.1), (4.2) 式について, 列挙の省略基準にいかに抵触するかを比較する。

(1)　上限制約式に抵触する場合

第三章で述べた列挙順序により Solution tree の Step 1 に該当する変数 x_1, x_2 の値がまず定まり, 次に Step 2 の変数の値が定まる。上限制約式 (4.1) では (4.3) 式のとおり

Step 3 の段階で制約条件に抵触することが判明するので，以降の解の列挙は省略する。これに対し，(4.2) 式の場合は，(4.4) 式のとおり Step 2 の段階で制約条件に抵触するので以降の解の列挙を省略する。

上限制約式 (4.1) の場合

$$\left.\begin{array}{l}
\text{Step 1} \\
\quad \text{部分解 } S=(x_1=1,\ x_2=0) \text{ 自由変数 } S'=(x_3,\ x_4,\ x_5,\ x_6,\ x_7,\ x_8,\ x_9) \\
\qquad\qquad \overset{x_1}{29(1)}+\overset{x_4}{53(0)}+\overset{x_7}{185(0)}+\overset{x_8}{96(0)}=29\leqq 250 \\
\text{Step 2} \\
\quad S=(x_1=1,\ x_2=0,\ x_3=0,\ x_4=1,\ x_5=0)\quad S'=(x_6,\ x_7,\ x_8,\ x_9) \\
\qquad\qquad 29(1)+53(1)+185(0)+96(0)=82\leqq 250 \\
\text{Step 3} \\
\quad S=(x_1=1,\ x_2=0,\ x_3=0,\ x_4=1,\ x_5=0,\ x_6=0,\ x_7=1)\ S'=(x_8,\ x_9) \\
\qquad\qquad 29(1)+53(1)+185(1)+96(0)=267\nleqq 250
\end{array}\right\} \quad (4.3)$$

上限制約式 (4.2) の場合

$$\left.\begin{array}{l}
\text{Step 1} \\
\quad S=(x_1=0,\ x_2=1)\ S'=(x_3,\ x_4,\ x_5,\ x_6,\ x_7,\ x_8,\ x_9) \\
\qquad\qquad 185(1)+\overset{x_2}{96(0)}+\overset{x_3}{53(0)}+\overset{x_6}{29(0)}=185\leqq 250 \\
\text{Step 2} \\
\quad S=(x_1=0,\ x_2=1,\ x_3=1,\ x_4=0,\ x_5=0)\ S'=(x_6,\ x_7,\ x_8,\ x_9) \\
\qquad\qquad 185(1)+96(1)+53(0)+29(0)=281\nleqq 250
\end{array}\right\} \quad (4.4)$$

(2) 下限制約式に抵触する場合

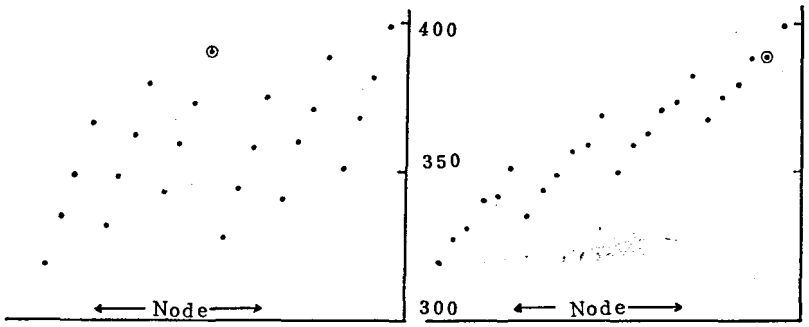
次の (4.5)，(4.6) 式のとおり，制約式 (4.1) の場合は Step 3 で，制約式 (4.2) では Step 1 で，それぞれ下限制約条件に抵触する。

下限制約式 (4.1) の場合

$$\left.\begin{array}{l}
\text{Step 1} \\
\quad S=(x_1=0,\ x_2=1)\ S'=(x_3,\ x_4,\ x_5,\ x_6,\ x_7,\ x_8,\ x_9) \\
\qquad\qquad \overset{x_1}{29(0)}+\max\overset{x_3\ x_4\ x_5}{(0,\ 53,\ 0)}+\max\overset{x_6\ x_7}{(0,\ 185)}+\max\overset{x_8\ x_9}{(96,\ 0)}=334\geqq 200 \\
\text{Step 2} \\
\quad S=(x_1=0,\ x_2=1,\ x_3=0,\ x_4=0,\ x_5=1)\ S'=(x_6,\ x_7,\ x_8,\ x_9) \\
\qquad\qquad 29(0)+53(0)+\max(0,\ 185)+\max(96,\ 0)=281\geqq 200 \\
\text{Step 3} \\
\quad S=(x_1=0,\ x_2=1,\ x_3=0,\ x_4=0,\ x_5=1,\ x_6=1,\ x_7=0)\ S'=(x_8,\ x_9) \\
\qquad\qquad 29(0)+53(0)+185(0)+\max(96,\ 0)=96\ngeqq 200
\end{array}\right\} \quad (4.5)$$

下限制約式 (4.2) の場合

$$\left.\begin{array}{l}
\text{Step 1} \\
\quad S=(x_1=1,\ x_2=0)\ S'=(x_3,\ x_4,\ x_5,\ x_6,\ x_7,\ x_8,\ x_9) \\
\qquad\qquad \overset{x_2}{185(0)}+\max\overset{x_3\ x_4}{(96,\ 0)}+\max\overset{x_5\ x_6\ x_7}{(0,\ 53,\ 0)}+\max\overset{x_8\ x_9}{(29,\ 0)}=178\ngeqq 200
\end{array}\right\} \quad (4.6)$$



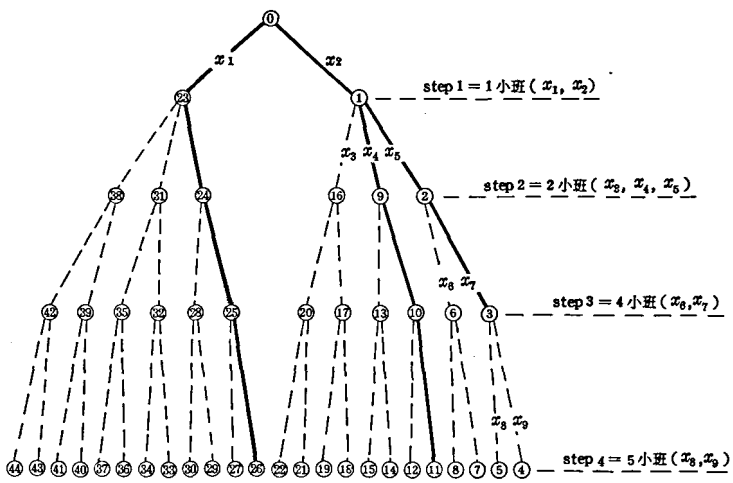
図—4.2　目的関数のZの並び（並べかえ前）　図—4.3　目的関数Zの並び（並べかえ後）

注1)　左図（並べかえ前）ではZはランダムに近いが，右図では右上りの傾向が明瞭である。

注2)　◉実行可能解の所在場所。

(3)　最適性がない場合

最適性がない場合とは，ある Node より先には既知の実行可能解の目的関数の最大値$\overline{Z}$より大きな目的関数をもつ解が存在しないことである。並べかえた後の問題（表—4.5）の Solution tree の端末の Node の目的関数Zは図—4.3である。これは並べかえ前の図—4.2に比べて，右上りの傾向がはっきりしている。例えば，今，図—4.3の右から2番目の Node が実行可能解であることを発見すると，それ以降の列挙はすべて省略できる。図—4.3のようにZの並びが単調減少列に近ければ，実行可能解を発見した場合には，多くの解を省略できる。



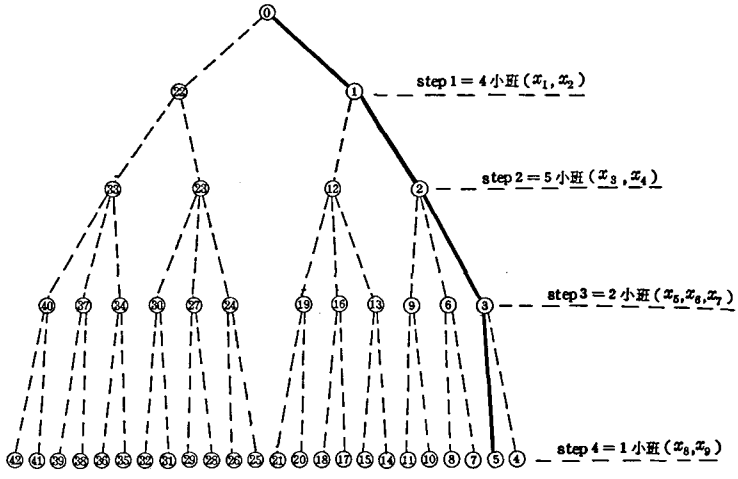
図—4.4　表4.4の列挙

注1)　実線は陽的列挙，点線は陰的列挙した部分である。

注2)　図—4.4の最適解は Node ㉖，図—4.5は Node ⑤である。しかし，収穫予定の内容は，どちらも同じである。

1小班の伐採分期＝2　　4小班の伐採分期＝2

2小班の伐採分期＝3　　5小班の伐採分期＝3



図—4.5 表—4.5の列挙

以上，並べかえが，それぞれの3つの省略基準に有効であり，列挙の省略により計算速度を加速することを明らかにした。そこで，表—4.4と表—4.5の列挙過程を Solution tree で図示すると図—4.4，図—4.5となる。図—4.5は図—4.4に比べて実際の陽的列挙にあたる実線部分が少なく，小班の並べかえにより計算が加速される状態を示している。

さて，このように問題のサイズを縮小し，並べかえることにより事前の準備が完了する。したがって，制約条件 (4.7)，(4.8)，(4.9)，(4.10) 目的関数 (4.11) がＦＯＡで解くべき問題である。

制約条件

各分期の伐採材積下限

$$\left.\begin{array}{ll} 154x_1+40x_5 & \geqq a_1 \\ 185x_2+96x_3+53x_6+29x_8 & \geqq a_2 \\ 113x_4+64x_7+38x_9 & \geqq a_3 \\ \text{各分期の伐採面積下限} & \\ 80x_1+23x_5 & \geqq a_4 \\ 80x_2+50x_3+23x_6+14x_8 & \geqq a_5 \\ 50x_4+23x_7+14x_9 & \geqq a_6 \end{array}\right\} \quad (4.7)$$

各分期の伐採材積上限

$$\left.\begin{array}{ll} 154x_1+40x_5 & \leqq b_1 \\ 185x_2+96x_3+53x_6+29x_8 & \leqq b_2 \\ 113x_4+64x_7+38x_9 & \leqq b_3 \\ \text{各分期の伐採面積上限} & \\ 80x_1+23x_5 & \leqq b_4 \\ 80x_2+50x_3+23x_6+14x_8 & \leqq b_5 \\ 50x_4+23x_7+14x_9 & \leqq b_6 \end{array}\right\} \quad (4.8)$$

$$x_i=0 \text{ or } 1 \quad (i=1,\ 2,\cdots\cdots,9) \quad (4.9)$$

$$\left.\begin{array}{l} x_1+x_2=1 \\ x_3+x_4=1 \\ x_5+x_6+x_7=1 \\ x_8+x_9=1 \end{array}\right\} \qquad (4.10)$$

目的関数

$$Z=154x_1+185x_2+96x_3+113x_4+40x_5+53x_6+64x_7+29x_8+38x_9 \quad \max \qquad (4.11)$$

ただし，a_i，b_i $(i=1,\ 2,\cdots\cdots,6)$ は各分期の伐採材積，伐採面積の下限と上限である。

第五章　加速列挙法の処理能力

第三章で説明したＦＯＡの計算手順を計算機プログラムとし，その処理能力を調べた[20]。プログラムは東京大学大型計算機センターを対象にして作った。入力データの様式とプログラムの内容は附録資料として末尾に添付する。

1　計算時間比較のための問題設定

ＦＯＡの処理能力を調べるために10個の事例問題を表—5.1のとおり設定し，計算時間を計った。問題のサイズ，制約条件の相異により生じる列挙の省略，および計算時間の差を検討できるように設定した。

表—5.1　処理能力測定のための事例問題

問題	小班数	分期数	変数の個数	制約式個数	サイズ	伐採材積許容範囲 (100m³)	伐採面積許容範囲 (ha)	計算時間 (秒)	総収穫量 (100m³)	陽的列挙数
1	8	8	53	16	2.151×10^6		47~ 87	2		358
2	25	4	54	8	1.699×10^8	1,200~2,500	280~470	5	6,349	18822
3	25	4	54	8	1.699×10^8	1,400~2,000	300~440	10	6,265	55115
4	25	4	54	8	1.699×10^8	1,500~1,700	350~420	24	6,162	168452
5	30	4	66	4	1.223×10^{10}	1,300~2,300		11	6,998	98617
6	30	4	66	4	1.223×10^{10}	1,500~2,100		50	6,903	531654
7	30	4	66	4	1.223×10^{10}	1,600~1,900		124	6,822	1347952
8	35	4	76	4	3.914×10^{11}		300~600	28	8,395	287404
9	35	4	76	4	3.914×10^{11}		390~560	214	8,292	2365803
10	35	4	76	4	3.914×10^{11}		430~540	275	8,247	3146317

問題１

これは，日林誌第56巻 p. 133の南雲の問題である[24]。表—5.2の10個の小班について８分期間の収穫予定を立てる。これは53個の変数をもち，問題のサイズは，2.151×10^6である。

問題2，3，4

表—5.3の25個の小班について４分期間の収穫予定を立てる。変数の数は54個，問題のサイズは$1,699\times10^8$である。問題２は各分期の伐採材積の許容範囲を1,200～2,500，伐採面積を280～470と広くとった。これに対し，問題３，４は伐採材積，面積の許容範囲を狭く設けた。

表—5.2 問題1——南雲の問題

小班名	面積 (ha)	各分期の伐期材積 (100m³)								変数の個数
		1	2	3	4	5	6	7	8	
1	64.0	*[1]	310	328	344	358	370	379	379	7
2	69.6	*[1]	*[4]	*[4]	*[4]	*[4]	*[4]	260	289	2
3	52.9	219	239	256	271	284	296	306	313	8
4	54.2	*[2]	*[3]	*[3]	*[3]	*[3]	*[3]	*[3]	*[3]	
5	45.8	*[4]	171	190	207	222	235	246	256	7
6	54.0	202	224	244	262	277	290	302	312	8
7	60.0	*[1]	*[4]	*[4]	*[4]	*[4]	*[4]	*[4]	*[5]	
8	38.5	*[1]	207	215	222	228	228	228	228	7
9	33.8	*[1]	182	189	195	200	200	200	200	7
10	61.3	*[1]	297	314	329	343	354	363	363	7

注1) *[1] 林道が開設されていないので伐採しない。
注2) *[2] 土場に隣接するので必ず伐採する。
注3) *[3] 注2の結果伐採できない。
注4) *[4] 伐期に達していないので伐採しない。
注5) *[5] 注4の結果必ず伐採する。
注6) 制約条件 各分期の伐採面積の許容範囲は67±20ha とする。
注7) 問題のサイズは，7×2×8×7×8×7×7×7＝2,151,291である。

表—5.3 問題2，3，4

小班	面積 (ha)	伐期材積 (100m³)			
		1	2	3	4
1	95	*	285	342	*
31	92	*	290	348	*
3	92	*	*	252	332
26	85	*	250	300	*
2	83	249	304	*	*
5	80	240	271	305	*
32	80	340	368	*	*
25	79	*	*	330	360
4	77	308	332	*	*
21	69	290	330	*	*
27	68	*	*	289	332
10	65	208	249	*	*
33	65	*	*	280	320
6	64	244	262	*	*
28	63	*	250	266	*
8	63	*	*	190	236
29	48	*	*	153	200
7	47	*	*	151	196
34	45	*	150	183	*
14	42	157	180	*	*
15	39	*	*	137	168
12	36	139	151	180	*
22	34	120	140	*	*
30	33	150	165	189	*
9	32	*	96	124	140

注1) *何らかの理由で伐採出来ない分期。
注2) 問題のサイズは，169,869,312＝($2^{21}\times3^{4}$)

表—5.4 問題5, 6, 7

小班	面積 (ha)	伐期材積 (100m³)			
		1	2	3	4
1	95	*	285	342	*
31	92	*	290	348	*
3	92	*	*	252	332
26	85	*	250	300	*
2	83	249	304	*	*
5	80	240	271	305	*
32	80	340	368	*	*
25	79	*	*	330	360
4	77	308	332	*	*
21	69	290	330	*	*
27	68	*	*	289	332
10	65	208	249	*	*
33	65	*	*	280	320
6	64	244	262	*	*
28	63	*	250	266	*
8	63	*	*	190	236
29	48	*	*	153	200
7	47	*	*	151	196
34	45	*	150	183	*
14	42	157	180	*	*
15	39	*	*	137	168
12	36	139	151	180	*
22	34	120	140	*	*
30	33	150	165	189	*
9	32	*	96	124	140
11	29	*	90	132	*
35	27	96	136	*	*
16	25	*	*	90	135
23	21	90	110	125	*
13	20	81	102	124	*

表—5.5 問題8, 9, 10

小班	面積 (ha)	伐期材積 (100m³)			
		1	2	3	4
17	98	330	372	*	*
1	95	*	285	342	*
31	92	*	290	348	*
3	92	*	*	252	332
26	85	*	250	300	*
24	84	*	260	314	*
2	83	249	304	*	*
5	80	240	271	305	*
32	80	320	368	*	*
25	79	*	*	330	360
4	77	308	332	*	*
18	71	290	324	*	*
21	69	290	330	*	*
27	68	*	*	289	332
10	65	208	249	*	*
33	65	*	*	280	320
6	64	244	262	*	*
28	63	*	250	266	*
8	63	*	*	190	236
19	55	*	*	172	199
29	48	*	*	153	200
7	47	*	*	151	196
34	45	*	150	183	*
14	42	157	180	*	*
15	39	*	*	137	168
20	37	*	*	145	164
12	36	139	151	180	*
22	34	120	140	*	*
30	33	150	165	189	*
9	32	*	96	124	140
11	29	*	90	132	*
35	27	96	136	*	*
16	25	*	*	90	135
23	21	90	110	125	*
13	20	81	102	124	*

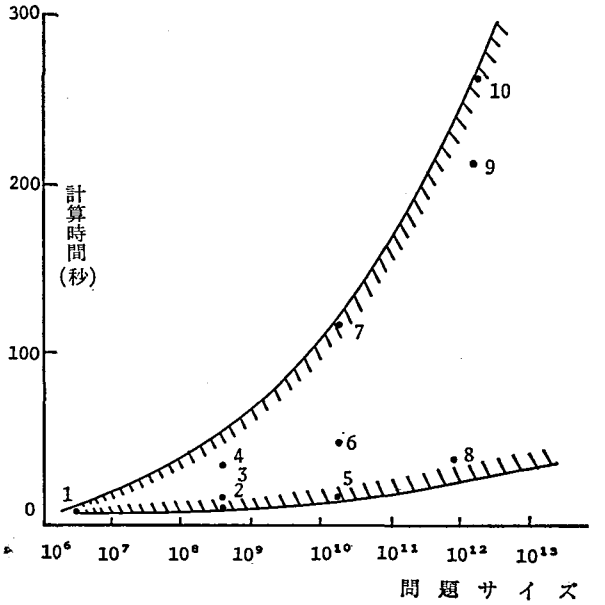
問題5, 6, 7

表—5.4の30個の小班について, 4分期間の収穫予定である。変数は66個, 問題のサイズは, 1.223×10^{10}である。各分期の伐採材積の許容範囲だけを制約条件とした。問題5は, その許容範囲を1,300〜2,300と広くしたのに対し問題6, 7は狭くした。

問題8，9，10

これらは表—5.5の35個の小班について，4分期間の収穫予定である。変数は76個，問題のサイズは，3.914×10^{11}である。各分期の伐採面積の許容範囲だけを制約条件とする。問題8は伐採面積の許容範囲が300～600と広い。それに対し問題9，10は390～560，430～540と狭い。

これら10個の問題を解くに要した時間は，表—5.6，図—5.1のとおりである。



図—5.1　計算時間

注）東大大型計算機センター　H-200Mによる

2　バラスの解法との比較

バラスの解法とＦＯＡとは，その対象とする0-1計画のタイプが異なる。したがって，両者の計算時間を直接的に比較することは適当ではないが，比較の参考にするために，同じ事例問題を二通りの解法で解いた。問題2をバラスの標準型0-1計画に変換して，それを解くのに要する時間の測定を試みた。バラスのプログラムは文献(28)に添付されたものを使用した。しかし，計算時間が長くかかり，最後まで解くことが不可能であった。途中の経過時間から推測した結果，ＦＯＡの計算時間の10,000～20,000倍の時間が必要であった。さらに，問題のサイズが大きくなるにつれて，その差は増加することが推測された。

収穫予定という特定のタイプの0-1計画を解くには，バラスの汎用的な解法を用いるよりも，専用的な解法ＦＯＡを開発することが有効であるといえる。

3　問題のサイズと計算時間

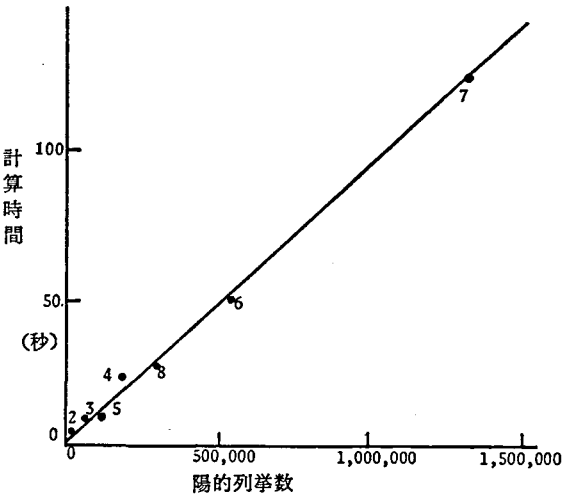
表—5.6　サイズと計算時間

問題	サイズ	計算時間（秒）	平均計算時間（秒）
1	2.151×10^{6}	2	2
2	1.699×10^{8}	5	
3	1.699×10^{8}	10	13
4	1.699×10^{8}	24	
5	1.223×10^{10}	11	
6	1.223×10^{10}	50	62
7	1.223×10^{10}	124	
8	3.914×10^{11}	28	
9	3.914×10^{11}	214	172
10	3.914×10^{11}	275	

表—5.6から，問題のサイズが大きくなるに従って，計算時間は増大するといえる。そこでは4通りのサイズの問題が試算され，それぞれのサイズ毎の平均計算時間はサイズの小さな順に，問題1で2秒，問題2，3，4では平均13秒，問題5，6，7では62秒，問題8，9，10では172秒となり，その増加傾向を示している。しかし，サイズが同じ問題であっても，その制約条件により計算時間は大幅に異なるといえる。問題2，3，4はすべて同一サイズの問題であるが，5，10，24秒と大きな差が生じている。このことは，問題5，6，7および問題8，9，10についても認められる。この理由は次のように考える。問題のサイズとは悉皆的に列挙する場合の列挙対象の数であるが，ＦＯＡはそのうち

表—5.7　陽的列挙数と計算時間

問 題	陽的列挙数	計算時間(秒)
1	358	2
2	18822	4
3	55115	10
4	168452	24
5	98617	11
6	531654	50
7	1347952	124
8	287404	28
9	2365803	214
10	3146317	275



図—5.2　陽的列挙数と計算時間

の一部を実際に列挙するだけで，他の大部分の列挙を省略する加速列挙法である。したがって，計算時間は問題のサイズより実際の列挙数，陽的列挙数に関係すると推定できる。

4　陽的列挙数と計算時間

各問題について，実際に列挙計算した Node の数と計算時間とを調べた結果，表—5.7と，図—5.2を得た。陽的列挙数と計算時間との間には明確な直線関係が認められた。したがって，計算時間は問題のサイズではなく，陽的列挙数によって決まると結論できる。陽的列挙数は問題のサイズおよび，問題自体の内容がいかに省略基準に抵触して，列挙を省略するかにより定まる。そこで，事例問題について省略基準への抵触について調べた。

5　陽的列挙率と省略基準

(1)　陽的列挙率

Solution tree における全 Node 数に対して，実際に陽的列挙した Node 数の割合を陽的列挙率とよび，列挙を省略して，陰的列挙した Node 数の割を陰的列挙率とよぶことにする。

$$\text{陽的列挙率}=\left(\frac{\text{実際に列挙した Node 数}}{\text{全 Node 数}}\right)\times 100$$

$$\text{陰的列挙率}=\left(\frac{\text{全 Node 数}-\text{実際に列挙した Node 数}}{\text{全 Node 数}}\right)\times 100$$

計算時間は陽的列挙数により定まることは前節で明らかにした。陽的列挙数は問題のサイズにより定まるところの全 Node 数と，陽的列挙率の積である。したがって，陽的列挙率をＦＯＡの処理能力と見なすことが出来る。事例問題の陽的列挙率は表—5.8 のとおりである。これは，問題の内容により陽的列挙率が相異することを示している。そこで，各問題の列挙過程で列挙省略の基準に如何に抵触するかを調べ，問題の内容と陽的列挙率との関係を検討する。

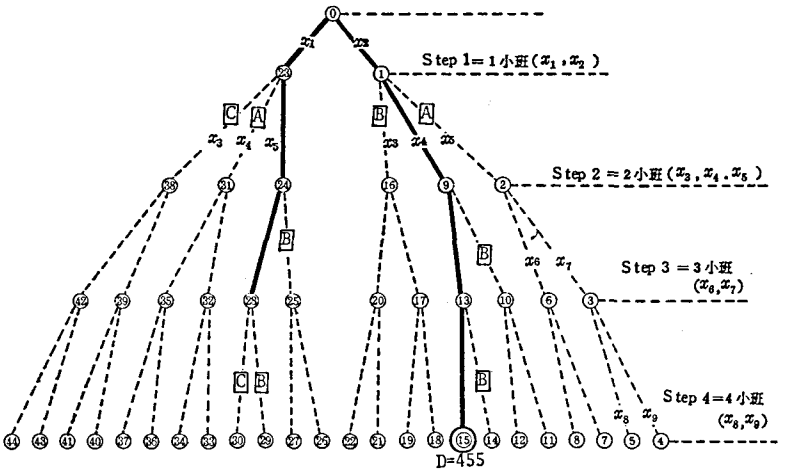
表—5.8　制約条件と陽的列挙率

問　題	サ イ ズ	制　約　条　件		陽的列挙率 (%)
		材 積 (m^3)	面 積 (ha)	
1	2.151×10^{6}		47~ 87	0.01664
2	1.699×10^{8}	1,200~2,500	280~470	0.01108
3	1.699×10^{9}	1,400~2,000	300~440	0.03245
4	1.699×10^{8}	1,500~1,700	350~420	0.09917
注) 4′	1.699×10^{9}	1,500~1,700	350~420	0.59433
5	1.223×10^{10}	1,300~2,300		0.00081
6	1.223×10^{10}	1,500~2,100		0.00435
7	1.223×10^{10}	1,600~1,900		0.01102
8	3.914×10^{11}		300~600	0.00007
9	3.914×10^{11}		390~560	0.00060
10	3.914×10^{11}		430~540	0.00080

注)　問題4′は，問題4を小班面積の小から大の順に並べかえて，試算した結果である。陽的列挙率，計算時間に大きな相異が生じた。
問題4（面積の大から小へ並べた場合）
陽的列挙率＝0.09917%　計算時間24（秒）
問題4′（面積の小から大へ並べた場合）
陽的列挙率＝0.59433%　計算時間129（秒）

(2)　省略基準

ＦＯＡの列挙の省略基準は先に述べたとおり，次の３つである。



図—5.3　列挙省略とその基準

注1)　総 Node 数は44である。
注2)　①-Ⓐ-②，Ａという基準により，②，③，④，⑤，⑥，⑦，⑧の７個の Node が省略された。
注3)　同様に，⑨-Ⓑ-⑩，Ｂにより，⑩，⑪，⑫の３個が省略された。
注4)　省略基準１をＡ，２をＢ，３をＣと表示した。

表—5.9　省略基準と省略された Node 数（比率）

問　題	総 Node 数 (%)	陽的列挙率 (%)	陰的列挙率 (%)	省　略　基　準　(%)		
				1	2	3
2	100	0.01108	99.98892	9.49	8.61	93.97
3	100	0.03245	99.96755	24.13	35.62	59.22
4	100	0.09917	99.90083	48.44	56.58	21.02

（単位%）

注1）Solution tree の総 Node 数を100とする。
注2）すべての Node は陽的列挙か，陰的列挙のいずれかにより列挙されている。その割合が陽的，陰的列挙率である。
注3）陰的列挙率について，その省略の根拠となる基準別の内訳をかぞえた。
注4）省略基準が重複する Node がある。そのため省略基準別内訳の合計は陰的列挙率を越える。

1　上限制約式に抵触
2　下限制約式に抵触
3　最適性がない

そこで，Solution tree をたどる列挙計算過程で，これらの基準に抵触する状態を図—5.3に，事例問題について省略した Node 数を表—5.9に示す。

これらの試算結果から，列挙の省略について次の傾向が推察される。

(i)　制約条件が厳しい問題では，省略基準1，2が陰的列挙率を高め，逆に制約条件の緩い問題では基準3が陰的列挙率を高める。

(ii)　制約条件の緩い問題は陰的列挙率が高く，したがって，計算時間は加速される。

(iii)　サイズが大きい場合，陰的列挙率は高くなる。

(iv)　面積の大から小への並べかえは，陰的列挙率を高めるのに有効である。

事例問題の試算により，問題サイズ，陽的列挙率，省略基準と計算時間との関係を検討した結果，ＦＯＡの能力は次のように結論できる。

(i)　まず，計算時間 t は図—5.2のとおり，問題の陽的列挙数 n にほぼ比例する。東大計算機センターによるとおよそ t (秒)$=10^{-4}\cdot n$ である。

(ii)　ところで，陽的列挙数は問題のサイズと陽的列挙率の積である。サイズは各小班・各分期に該当する変数の数とその構成という形式によりきまるのに対し，陽的列挙率は問題の制約条件，目的関数の内容によりきまる。

(iii)　陽的列挙率は試算事例によれば，0.099～0.00007（%）の範囲である。そして，制約条件が厳しく，あるいは問題サイズが小さい場合には陽的列挙率は大きくなる傾向がある。

(iv)　今，仮に，東大センターを対象に計算時間5分を目安とするなら，解ける問題のサイズはおよそ，10^{11}～11^{12}である。

(v)　このサイズは変数の個数に換算すると，70～80個，小班数では30～35個小班について，3～4分期間の収穫予定問題に相当する。

(vi)　このサイズは，国有林や，まとまった事業団地の現実の収穫対象小班の数に比べて極めて小さい。したがって，ＦＯＡは対象森林全体についての最適収穫予定を計算する能力は

ない。部分での最適解を提供する機能を持っているにすぎない。
(vii) 収穫予定を0–1計画そのものに定式化した段階で，それを解くのに必要な計算時間は，図—5.1を参照すれば，およそ事前に予測できる。

第六章　収穫予定への適用

1　事業団地の収穫予定

ある事業団地の収穫予定を0–1計画とし，それを加速列挙法ＦＯＡにより解いた。ところで，森林の収穫予定の問題は，その森林の状況と経営者の考え方に強く依存し，その目標も著しく多様な領域である。したがって，この数理計画問題の最適解は，収穫予定問題全体の中での部分の最適化，あるいは設定した1つの評価基準に対する最適解を現わしているにすぎない[31]。この0–1計画が扱う収穫材積の最大化という考え方は森林施業にとって，基本的で，相当に普遍的な評価基準ではあるが，それ以外にも考慮されるべき施業上の要因は多い。したがって，ここでの収穫予定とは森林経営者が施業計画を決定するための判断材料であり，それを提供することである。さて，対象森林は木曾谷国有林内の1つの事業区で全森林面積は6,900haである。そこは3つの施業団地に分れているので，それぞれの団地で伐採，育林事業が安定して継続できるように，将来4分期間の収穫予定を立てた。各団地について，施業対象となる小班は表—6.1のとおりである。林齢により3グループに分けた。

表—6.1　対象森林の小班数

団地名	小班数	(A)若齢人工林	(B)老齢天然林	(C)壮齢人工林
Ⅰ	265	163	88	14
Ⅱ	111	80	9	22
Ⅲ	307	213	79	15
計	683	456	176	51

注）収穫予定の対象になる小班は(B)，(C)の227個である。しかし，0–1計画による最適化問題の対象になるのは(C)の51個である。

(A)　計画期間中には伐期齢に達しないので，収穫の対象とならない若齢人工林。
(B)　成長の停止した老齢天然林。
(C)　伐期齢に達する壮齢人工林。

ここでは，(B)および(C)が収穫予定の対象である。しかし，(B)は成長が停止しているから，どの分期で伐採しても総伐採材積の最大化には何の影響も与えない。したがって，(C)を対象にして0–1計画を適用した。小班の面積，伐期材積は表—6.2のとおりである。各団地ごとに壮齢人工林の伐採材積が漸増するように各分期の伐採材積の上限，下限を設定した。

団地ごとに0–1計画を次の方法により設定した。以下，団地Ⅰについて説明する。
(i)　面積の大小により並べかえたあと，それぞれの小班について伐採可能な分期に変数 x_1 から x_{44} を表—6.3のとおり割り当てた。

表—6.2 計 算 条 件

団地名	小班名	面積 (ha)	伐期材積 (m³) 1	2	3	4
I	1	30.39	•8,732	9,343	9,717	10,008
	2	15.22	•4,551	4,870	5,064	5,267
	3	14.46	*	*	3,955	•4,231
	4	12.79	3,573	•3,966	4,244	4,414
	5	1.83	*	*	*	•198
	6	21.85	5,014	5,616	6,233	•6,670
	7	1.83	146	156	162	•169
	8	22.91	•6,196	6,630	6,895	7,171
	9	34.81	*	•9,246	9,894	10,289
	10	3.74	*	830	•888	924
	11	16.82	*	4,986	•5,335	5,548
	12	28.05	*	7,627	8,466	•9,059
	13	38.98	*	13,341	•14,275	14,846
	14	17.34	*	•6,041	6,283	6,534
	伐採材積の許容範囲		20,000 ~ 17,000	20,000 ~ 17,000	21,000 ~ 18,000	21,000 ~ 1,8000
II	1	21.70	4,763	•5,096	5,300	5,512
	2	30.44	*	*	*	•7,875
	3	20.98	*	*	*	•5,334
	4	19.90	•6,107	6,351	6,605	6,803
	5	29.64	•9,164	9,531	9,912	10,209
	6	38.56	*	*	•13,703	14,662
	7	24.16	*	*	6,221	•6,656
	8	18.47	*	*	•4,942	5,288
	9	7.08	*	*	•1,941	2,076
	10	22.40	*	*	*	•5,815
	11	27.43	*	*	*	•7,089
	12	10.88	*	*	•2,164	2,316
	13	10.88	*	*	•2,705	2,895
	14	35.65	*	*	•9,982	10,680
	15	14.22	•4,224	4,393	4,569	4,706
	16	19.81	6,422	•6,872	7,146	7,432
	17	4.10	•1,312	1,364	1,419	1,462
	18	25.99	7,511	•8,037	8,358	8,609
	19	8.15	2,508	•2,684	2,791	2,903
	20	11.43	*	*	*	•4,117
	21	8.21	*	*	*	•2,368
	22	16.28	*	*	*	•5,640
	伐採材積の許容範囲		25,000 ~ 20,000	27,000 ~ 22,000	38,000 ~ 25,000	45,000 ~ 30,000

	1	15.12	4,448	•4,765	4,951	5,160
	2	3.66	•821	870	918	962
	3	4.06	*	*	1,301	•1,359
	4	14.21	*	3,848	•4,166	4,384
	5	18.77	*	*	4,993	•5,364
	6	7.01	*	*	*	•1,956
	7	3.04	801	896	985	•1,047
Ⅲ	8	35.99	9,862	•10,560	11,124	11,683
	9	18.91	*	*	*	•4,982
	10	38.04	*	14,038	•14,887	15,530
	11	1.88	*	*	*	•340
	12	19.20	•6,178	6,457	6,684	6,894
	13	20.44	*	*	*	•5,065
	14	8.15	2,407	•2,579	2,685	2,800
	15	29.88	•9,058	9,424	9,810	10,101
	伐採材積の許容範囲		19,000 ~ 16,000	20,000 ~ 17,000	21,000 ~ 18,000	22,000 ~ 19,000

注1) ＊伐期齢に達しないので，伐採対象にならない。
注2) ・ＦＯＡにより得られた最適伐採分期。

表—6.3 変数の定義

団地	小班	面積 (ha)	伐期材積 (m³) と変数 1	2	3	4
	13	38.98	*	13,341 x_1	14,275 x_2	14,846 x_3
	9	34.81	*	9,246 x_4	9,894 x_5	10,289 x_6
	1	30.39	8,732 x_7	9,343 x_8	9,717 x_9	10,008 x_{10}
	12	28.05	*	7,627 x_{11}	8,466 x_{12}	9,059 x_{13}
	8	22.91	6,196 x_{14}	6,630 x_{15}	6,895 x_{16}	7,171 x_{17}
	6	21.85	5,014 x_{18}	5,616 x_{19}	6,233 x_{20}	6,670 x_{21}
I	14	17.34	*	6,041 x_{22}	6,283 x_{23}	6,554 x_{24}
	11	16.82	*	4,986 x_{25}	5,335 x_{26}	5,548 x_{27}
	2	15.22	4,551 x_{28}	4,870 x_{29}	5,064 x_{30}	5,267 x_{31}
	3	14.46	*	*	3,955 x_{32}	4,231 x_{33}
	4	12.79	3,573 x_{34}	3,966 x_{35}	4,244 x_{36}	4,414 x_{37}
	10	3.74	*	830 x_{38}	888 x_{39}	924 x_{40}
	7	1.83	146 x_{41}	156 x_{42}	162 x_{43}	169 x_{44}

(ii)　各分期の伐採材積の許容範囲を制約条件とした。それは (6. 1) 式である。

(iii)　目的関数は (6. 2) 式である。

制約条件

伐採材積の上限，下限

$$
\left.\begin{array}{l}
\text{第1分期} \\
\quad 8732x_7+6196x_{14}+\cdots\cdots\cdots\cdots+146x_{41}\geqq 17000 \\
\quad 8732x_7+6196x_{14}+\cdots\cdots\cdots\cdots+146x_{41}\leqq 20000 \\
\text{第2分期} \\
\quad 13341x_1+9246x_4+\cdots\cdots\cdots\cdots+156x_{42}\geqq 17000 \\
\quad 13341x_1+9246x_4+\cdots\cdots\cdots\cdots+156x_{42}\leqq 20000 \\
\text{第3分期} \\
\quad 14275x_2+9894x_5+\cdots\cdots\cdots\cdots+162x_{43}\geqq 18000 \\
\quad 14275x_2+9894x_5+\cdots\cdots\cdots\cdots+162x_{43}\leqq 21000 \\
\text{第4分期} \\
\quad 14846x_3+10289x_6+\cdots\cdots\cdots\cdots+169x_{44}\geqq 18000 \\
\quad 14846x_3+10289x_6+\cdots\cdots\cdots\cdots+169x_{44}\leqq 21000
\end{array}\right\} \qquad (6.1)
$$

目的関数

$$Z=13341x_1+14275x_2+\cdots\cdots\cdots\cdots+169x_{44} \quad \max \qquad (6.2)$$

これをＦＯＡプログラムにより解き，最適解を得た。団地Ⅱ，Ⅲについても同様の方法で最適解を得た。各小班の伐採分期は表―6. 2 に示す。問題のサイズ，変数の数，分期別伐採材積，総伐採材積，計算時間は表―6. 4 に示す。この壮齢人工林の最適収穫予定に，老齢天然林の収穫を加えることにり，団地毎の収穫予定は完成した[20]。

表―6. 4　団地別収穫予定

団地	小班数	変数個数	サイズ	伐採材積 (m³)					計算時間 (秒)
				1	2	3	4	計	
Ⅰ	14	45	5.972×10^6	19,479	19,253	20,498	20,327	79,557	2
Ⅱ	22	53	8.389×10^6	20,807	22,689	35,437	44,894	123,827	3
Ⅲ	15	42	5.898×10^5	16,057	17,904	19,053	20,113	73,127	2

この適用事例の結果は次のように要約できる。

(i)　この問題のサイズは，いずれの団地も 10^6～10^7 程度で，前章で明らかにしたＦＯＡの処理能力の範囲内である。したがって，容易に最適解を求めることが出来た。その結果，各分期の伐採材積を一定の許容範囲に規制し，しかも，総収穫材積を最大にする小班毎の伐採分期を決めることが出来た。

(ii)　伐採を予定する林分の所在地を小班名により特定した。この小班別収穫予定は，森林の空間配置の問題を検討する有効な資料である。これについては，次節の林道問題，配置問題により検討する。

(iii)　小班別収穫予定は，伐採を予定する林分そのものを現場で指差し，示すことが出来る。したがって，その林分の作業の難易，販売，造林技術などを，施業上の観点から検討して収穫予定案を評価することが出来る。このことは伐採すべき材積，面積の総量だけを示す現在の収穫予定にない特長である。

(iv)　限られたサイズの問題に対しではあるが，ＦＯＡは小班別収穫予定を立てるのに必要な計算技術的な裏づけとなることを，この事例は示している。

2　林道計画を取り入れた収穫予定

(1)　制約条件としての林道計画

収穫予定の課題は長期的で，対象森林全体としての最適化と，短期的で林分という部分の最適化という２つの評価基準を調和させることにあり，歴史的にも収穫予定論の争点であった。例えば，「それまでの経理方式が全森林の立場から各林分の取扱い方をきめたのに対し，林分施業法にあっては逆に各林分自体の要求をまず考慮し，しかる後に全体に及ぼすという順序をとる」[27]。

さて，0-1 計画では全体としての最適化は目的関数として扱う。それに対し，小班の個別の事情や分期毎の伐採量の要求は制約条件として扱う。小班の個別の事情は雑多でその種類も多いが，次の３つの範疇に整理できる。

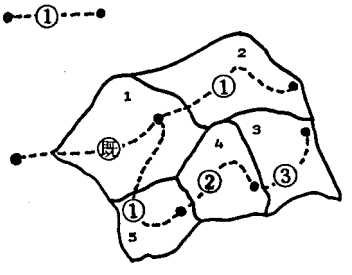
(i)　林齢　林分が伐期齢に達しているか否かという条件。

(ii)　林道　林分の伐出作業が行えるように林道が開設されているか否かという条件。

(iii)　施業　法令による施業規制，保全上の規制，作業上の要求，経営上あるいは政策上の要求など，あらゆる雑多な個別の事情を含めて，施業が規制されているか否かという条件。

今まで，これらのうち林齢に関する事項が収穫予定論の中心であったが，林道および施業に関する事項もそれと同様に重要な要件であり，特に「林道計画を考慮しない計画は全く無意味なもの」となる[24]。

ところで，0-1 計画による小班別収穫予定においては小班という場所を特定した林分を扱うので，林道路線計画ような空間配置を扱う計画事項と伐採計画との対応が可能となる[11]。ここでは，林道計画を 0-1 計画の制約条件として数式化する方法を表—6.5，図—6.1 の事



図—6.1　小班と路線との対応

表—6.5　対　象　森　林

小班名	林齢(年)	伐期齢(年)	路線	開設分期	面積(ha)	伐期材積 (m³) 1	2	3
1	45	50	既設		8.5	2,490 x_1	2,680 x_2	2,850 x_3
2	55	60	1		8.2	3,020 x_4	3,180 x_5	3,330 x_6
3	40	50	3		8.1	(2,240) x_7	(2,450) x_8	2,640 x_9
4	45	45	2		7.6	(2,330) x_{10}	2,500 x_{11}	2,660 x_{12}
5	60	50	1		6.4	2,750 x_{13}	2,880 x_{14}	2,970 x_{15}

注）（　）は林道が開設されていないので，伐採しない分期。

例を用いて説明する。

まず，林道が開設されている小班は伐採が出来るが，未開設の小班は伐採が出来ないとする。次に小班と林道路線との対応を規定するため，森林図の上に林道網を描く。林道網は既設路線と将来の開設予定路線とからなる。図—6.1により集運材方法，地形から判断して，それぞれの小班とその搬出に利用される路線とを対応させる。ところで，それぞれの路線の開設予定分期は林道計画により，既に定められているとする。この小班と林道との対応を規定した結果を表—6.5に示す。

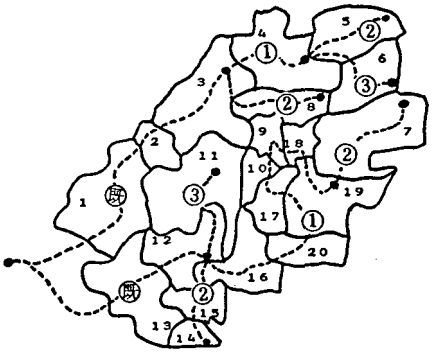
表—6.6　小班と林道路線の対応

小班名	林齢(年)	伐期齢(年)	林道の開設分期	面積(ha)	伐期材積(m³) 分期 1	2	3
11	45	50	3	8.5	2,490	2,680	2,850
7	50	60	2	8.2	3,020	3,180	3,330
1	40	40	既設	8.1	2,240	2,450	2,640
3	45	50	既設	7.6	2,330	2,500	2,660
4	30	40	1	7.0	1,700	1,970	2,200
13	60	40	既設	6.4	2,750	2,880	2,970
19	50	50	1	6.0	2,160	2,300	2,420
12	30	40	既設	5.3	1,230	1,420	1,590
5	45	50	2	4.8	1,440	1,550	1,650
6	50	55	3	4.2	1,330	1,410	1,480
8	45	50	2	3.6	1,110	1,200	1,270
16	40	45	1	3.1	970	1,070	1,150
20	50	45	1	2.8	880	940	990
17	30	40	1	2.3	540	620	700
15	50	40	2	1.9	720	760	800
2	40	50	既設	1.6	460	500	540
14	50	50	2	1.6	500	530	560
9	35	45	1	1.5	360	400	440
10	45	50	1	1.3	420	450	480
18	60	50	1	1.1	410	430	450

表—6.7　伐期齢と林道の制約

小班名	分期 1	2	3
11	[(x_1)]	[x_2]	x_3
7	[(x_4)]	(x_5)	x_6
1	x_7	x_8	x_9
3	(x_{10})	x_{11}	x_{12}
4	(x_{13})	(x_{14})	x_{15}
13	x_{16}	x_{17}	x_{18}
19	x_{19}	x_{20}	x_{21}
12	(x_{22})	(x_{23})	x_{24}
5	[(x_{25})]	x_{26}	x_{27}
6	[(x_{28})]	[x_{29}]	x_{30}
8	[(x_{31})]	x_{32}	x_{33}
16	(x_{34})	x_{35}	x_{36}
20	x_{37}	x_{38}	x_{39}
17	(x_{40})	(x_{41})	x_{42}
15	[x_{43}]	x_{44}	x_{45}
2	(x_{46})	(x_{47})	x_{48}
14	[x_{49}]	x_{50}	x_{51}
9	(x_{52})	(x_{53})	x_{54}
10	(x_{55})	x_{56}	x_{57}
18	x_{58}	x_{59}	x_{60}

(　) 伐期齢に達しない分期
[　] 林道が開設されていない分期



図—6.2　林道開設予定図

さて，これら5個の小班について林道計画を取り入れて，将来3分期間の収穫予定をたてるには，既に説明したが各小班の各分期に変数を割り当てる。そして，伐採材積・面積の許容範囲を制約条件，総伐材採積を目的関数とする0-1計画を設定する。ここではそれに加えて，林道が開設されていない小班は伐採出来ないことを制約条件 (6.3) 式として追加する。

$$\left.\begin{array}{ll}\text{小班3} & x_7=x_8=0\\ \text{小班4} & x_{10}=0\end{array}\right\} \qquad (6.3)$$

(2) 適用事例

林道計画を取り入れた収穫予定の適用事例として20個の小班からなる森林について，将来3分期間の収穫予定を立てた。各小班の林齢，伐期齢，面積，伐採材積，対応路線の開設分期は表—6.6，林道計画は図—6.2のとおり与えられるとする[21]。

表—6.7のとおり変数を各小班，各分期に割り当て，この収穫予定問題を0-1計画として数式化した。

制約条件

1

$$\left.\begin{array}{l} 2490x_1+3020x_4+\cdots\cdots\cdots\cdots+410x_{58}\geqq V_1 \\ 2680x_2+3180x_5+\cdots\cdots\cdots\cdots+430x_{59}\geqq V_2 \\ 2850x_3+3330x_6+\cdots\cdots\cdots\cdots+450x_{60}\geqq V_3 \end{array}\right\} \quad (6.4)$$

2

$$\left.\begin{array}{l} 2490x_1+3020x_4+\cdots\cdots\cdots\cdots+410x_{58}\leqq V'_1 \\ 2680x_2+3180x_5+\cdots\cdots\cdots\cdots+430x_{59}\leqq V'_2 \\ 2850x_3+3330x_6+\cdots\cdots\cdots\cdots+450x_{60}\leqq V'_3 \end{array}\right\} \quad (6.5)$$

V_j, V'_j ; j分期の伐採材積の下限，上限

3 $x_i=0$ or 1 ($i=1, 2, \cdots\cdots, 60$) (6.6)

4

$$\left.\begin{array}{l} x_1+x_2+x_3=1 \\ \vdots \\ x_{58}+x_{59}+x_{60}=1 \end{array}\right\} \quad (6.7)$$

5 伐期齢の制約

$$\begin{array}{l} x_1=x_4=x_5=x_{10}=x_{13}=x_{14}=x_{22}=x_{23}=x_{25}=x_{28}=x_{31} \\ \qquad\qquad =x_{34}=x_{40}=x_{41}=x_{46}=x_{47}=x_{52}=x_{53}=x_{55}=0 \end{array} \quad (6.8)$$

6 林道の制約

$$x_1=x_2=x_4=x_{25}=x_{28}=x_{29}=x_{31}=x_{43}=x_{49}=0 \quad (6.9)$$

目的関数

$$Z=2490x_1+2680x_2+\cdots\cdots\cdots\cdots+450x_{60} \quad \max \quad (6.10)$$

さて，林齢あるいは林道の制約条件を収穫予定に含める場合と，含めない場合とを比較するために[17]，次の3つの問題A，B，Cを設けた。

問題A，B，Cを解くために，各分期の伐採材積の下限，上限として次の三通りを設けた。

表—6.8 設定された問題

問題名	問題の内容	目的関数	制約式
A	小班個別の事情を無視した場合	(6.10)	(6.4) (6.5) (6.6) (6.7)
B	林齢条件を取り入れた場合	(6.10)	(6.4) (6.5) (6.6) (6.7) (6.8)
C	林齢条件と林道条件を取り入れた場合	(6.10)	(6.4) (6.5) (6.6) (6.7) (6.8) (6.9)

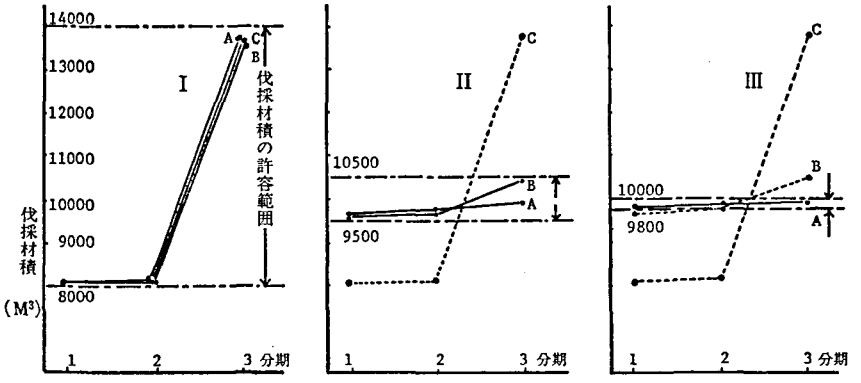
表—6.9　最適解の総伐採材積（Z）

問題＼条件	(I)	(II)	(III)
A	29,920	29,610	29,540
B	29,710	29,400	*
C	29,680	*	*

注）＊実行可能解なし　　(単位m³)

	伐採材積の下限V	上限V' (m³)
(I)	8,000	14,000
(II)	9,500	10,500
(III)	9,800	10,000

以上の問題A，B，Cそれぞれについての制約条件I，II，IIIをFOAプログラムにより解き，表—6.9，図—6.3の解を得た。



図—6.3　伐採材積の許容範囲と最適解

•——実行可能な最適解。
•……許容範囲をはみ出す。実行可能解なし。
A，B，Cは表—6.8に示す計画問題である。

この結果を検討する。

(i)　Iの場合は，伐採材積の許容範囲が広いので，問題A，B，Cのいずれにも最適解が存在する。そして，分期毎の伐採量のパターンは類似している。目的関数である総伐採材積は問題A，B，Cについて，29920>29710>29680 (m³) の順序となる。個別の小班に関する制約条件がない問題Aが最大の収穫を得るのに対し，林齢と林道という制約条件を課した問題Cは最小の収穫しか得ることが出来ない。個々の小班に関する要求と全体にとっての最適化とは一致しないことを示している。

(ii)　次に，分期毎の伐採材積を均等化するために伐採材積の許容範囲を狭くしたのが条件(II)，(III) である。問題Bの (III)，問題Cの (II)，(III) には実行可能解が存在しない。小班個別の制約条件を考慮すると保続的な収穫予定は立て難くなるといえる。すなわち，林齢と林道の制約を課した問題Cでは保続的な収穫予定を立てることは出来ないのに対し，問題Aは毎分期均等な材積収穫の計画を得ることが出来る。

(iii)　林道計画を取り入れた0-1計画は収穫予定に対する林道計画の制約を明らかにするとともに，小班ごとに伐採搬出を確実に保証するので，実行可能な収穫予定を立案することを示している。事例問題を解くのに要した計算時間を表—6.

表—6.10　計算時間

問題＼条件	(I)	(II)	(III)
A	23	25	71
B	2	2	*
C	2	*	*

(単位秒)

注）東大計算機センターM200Hによる

10に示す。

3　伐採林分の配置

小班は地図上にその位置，形が規定されているから，小班を単位とする収穫予定は，前にも述べたが，伐採林分の所在場所を特定した収穫予定法である。この方法は森林構成の空間配置を規整する手段となる。ここでは，伐採小班の配置を0-1計画の制約条件として取り扱う方法を事例問題により検討する。

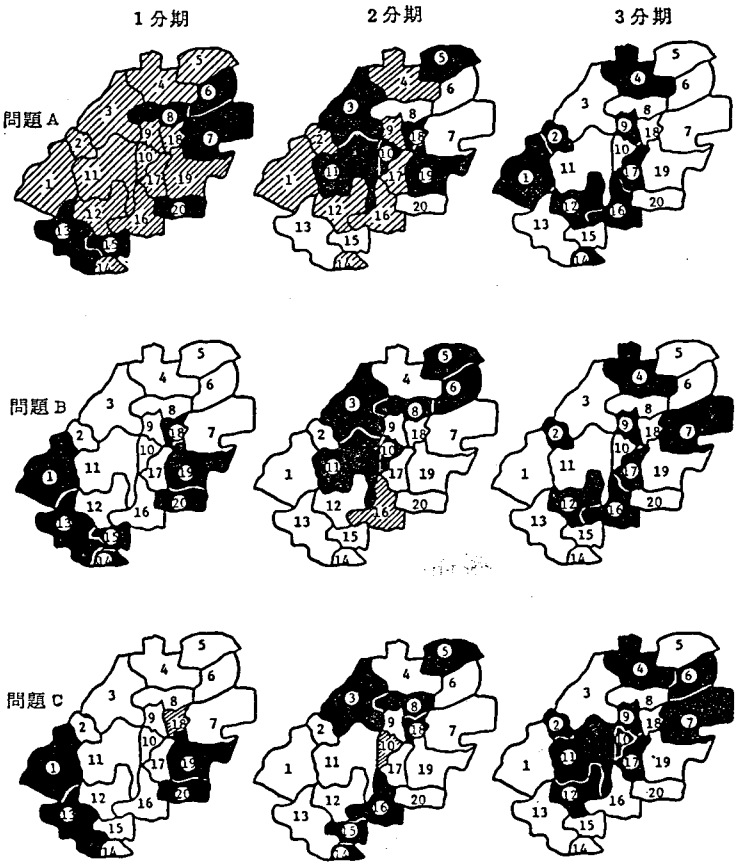
(1)　最適解とその小班配置

前節の問題A—(Ⅲ)，B—(Ⅱ)，C—(Ⅰ)の最適解により決る小班別の伐採分期は，表—6.11である。これらを地図上に示すと，図—6.4となる。

ところで，将来の森林の空間配置はこれら伐採小班の配置によってきまる。問題Aを例にすると，5年後，10年後，15年後の齢級別林分配置図は，1分期，2分期，3分期の伐採予定小班図を重ね合せると出来る。図—6.5は15年後の林分の配置図である。当然ではあるが，このように小班別収穫予定では結果として，将来の森林の空間配置を知ることが出来る。こ

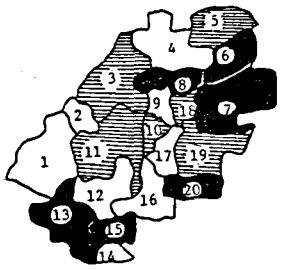
表—6.11　小班の伐採分期

小班名	問題		
	A	B	C
11	2	2	3
7	1	3	3
1	3	1	1
3	2	2	2
4	3	3	3
13	1	1	1
19	2	1	1
12	3	3	3
5	2	2	2
6	1	2	3
8	1	2	2
16	3	3	2
20	1	1	1
17	3	3	3
15	1	1	2
2	3	3	3
14	3	1	2
9	3	3	3
10	2	2	3
18	2	1	2



図—6.4　伐採予定小班

注）黒い部分：伐採する小班。
斜線部分：伐採可能な小班。



図—6.5　15年後の林分配置
注）　黒 ……Ⅲ齢級
　　斜線……Ⅱ齢級
　　白 ……Ⅰ齢級

れは従来の齢級表による収穫予定では出来ないことであり，この 0-1 計画による収穫予定の大きな特長である。

しかし，第二章で規定した 0-1 計画は，伐採材積，面積を制約条件，目的関数とした最適化問題であり，伐採林分の所在場所については何も考えていない。したがって，この 0-1 計画は林分の配置，隣接関係，順序性を規制する機能は持っていない。

(2)　森林の場所的秩序

伐採する林分の場所を規制し，目標とする林分配置を実現することにより，森林に場所的な秩序を与えようとする考え方は，歴史的には伐採列区により示された[37]。「19世紀のドイツでは大面積な皆伐が広く行われたが，そのための針葉樹，特にトウヒ林の風害が甚大なものとなった」[31]。この冬期の強烈な風に対する対策として，伐区の大きさと共に，その伐採林分の配置が，順序性が提案された。一連の伐区を主風の方向に林齢の勾配をつけて配列させるものである。

しかし，日本では，地形，気象，植栽樹種が異なり，この風向に対する林分配置の問題は施業として定着していない[35][36]。とりわけ，戦後，架線運材法が普及した結果，伐出作業に適した伐採順序，伐区選定が行われている。また，林道が普及すると路線の開設位置にそって伐区が選定されたため，伐採後に再生する森林の配置を規整する考え方は希薄であった。

昭和30年代の大面積皆伐，一斉造林に対し，崩壊地発生，土壌流失，保水機能の減退，病虫害・気象害の発生などで森林の不健全さが指摘され，森林保護，地域保全の観点から林分配置は課題となってきた。特に，昭和40年代後半，環境保全や風致のために伐区を分散，そして小面積とし，保残帯設定などの施業が指導されている[34]。

しかし，森林の場所規整は，木材生産，森林保全のため理念的に論議されているが，具体的な森林施業技術として定着していない[38]。しがって，空間配置についての理論研究と実践が必要であり，今後の課題である。現在のところ，空間配置の目標を明示出来ないが，この節では，地図を参照して，伐採小班の場所を変更，あるいは限定することを，0-1 計画の制約条件に加えることを事例問題により示す。

(i)　まず，0-1 計画の最適解により定まる小班の伐採分期を森林図上に描く。ここでは，配置はまったく考慮されていない。

(ii)　空間配置の目標あるいはモデルを想定した上で，地図上の各小班の伐採分期を概観する。小班相互の隣接関係，伐採の順序性について想定した目標に抵触するか，より望ましい配置を指摘する。

(iii)　指摘した事項を 0-1 計画の制約条件として追加する。伐採分期を特定するか，特定の分期の伐採を禁止するかのいずれかによる。

$$\left.\begin{array}{ll}\text{伐採分期の特定} & x_i=1\\ \text{特定分期の伐採禁止} & x_i=0\end{array}\right\} \qquad (6.1)$$

図—6.6 の上段が，表—6.6，問題Aの最適解の伐採小班の配置である。これについて次のような指摘があったとする。

表—6.12　伐採小班を特定した場合の伐採材積

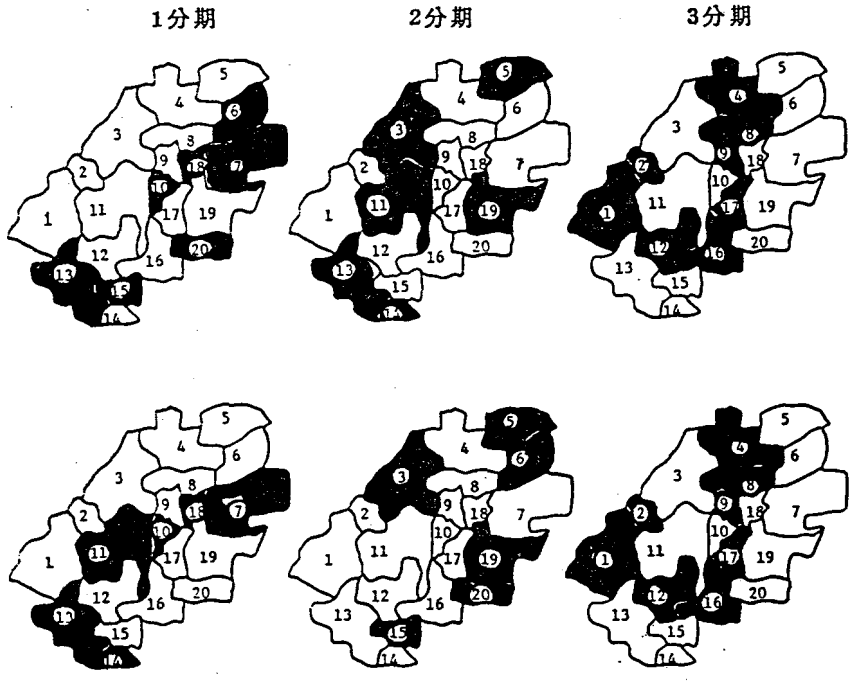
分期	1	2	3	合計（Z）
特定しない計画	9,530	9,560	10,530	29,620
特定した計画	9,590	9,460	10,530	29,580
伐採材積の許容範囲	10,600 ～ 9,400	10,600 ～ 9,400	10,600 ～ 9,400	

（単位m³）

$$\left.\begin{array}{lll} 11\text{小班} & 2\text{分期} \longrightarrow 1\text{分期} & x_1=1 \\ 6\text{小班} & 1\text{分期} \longrightarrow 2\text{分期} & x_{29}=1 \\ 15\text{小班} & 1\text{分期} \longrightarrow 2\text{分期以降} & x_{43}=0 \end{array}\right\} \qquad (6.2)$$

(6.2) 式を制約条件に追加して，再計算の結果得た最適解が図—6.6 の下段である。

各分期の伐採材積の許容範囲，伐採材積は表—6.12 である。小班の伐採分期を制約することにより目的関数Zは小さくなる。小班の配置という部分の要求が，全体の最適化に与える負の影響である。簡易な事例であるが，伐採林分の配置の制約のもとで収穫材積を最大にする小班別収穫予定を見つけることが出来た。



図—6.6　伐採林分の配置を規制した収穫予定
注1）　上段：配置を規制しない計画
　　下階：配置を規制した計画

第七章　ま　と　め

(1)　収穫予定は各小班の伐採分期を決める問題に置き換えることが出来る。すなわち，小班伐採分期が決れば，それらを合計することにより全体の伐採材積・面積が決り，しかも，伐採林分自身も決っているから極めて具体的な収穫予定案となる。伐採林分の所在場所を特定することにより林道計画や将来の森林の配置問題を規整できることが，この論文の主題である小班別収穫予定の特長である。

この小班別収穫予定は1つの数理問題で表現できる。今，i 小班の j 分期の伐採予定を二値変数 x_{ij} で定義する。各分期の伐採材積・面積の許容範囲，伐期齢，林道開設の有無，施業要件などの制約条件のもとで，計画期間中の総伐採材積を最大にする変数 x_{ij} を求める問題は，制約条件 (1.9)，(1.10)，(1.11)，目的関数 (1.13) により 0–1 線型計画となる。

(2)　この 0–1 計画を解くことにより，各小班の最適伐採分期を得ることが出来る。そして，列挙法により，この問題を解くことが出来ることは理論的には明らかである。しかし，問題のサイズが大きくなると計算量が厖大になるので実用的には使えないのが現状である。

そこで収穫予定に特有の附帯条件である (1.12)，(1.14)，(1.15) 式をもつ特定のタイプの 0–1 計画を設け，このタイプの問題に専用的な解法ＦＯＡ (Forestry Oriented Algorithm) を開発した。これはバラスの Additive Algorithm を基盤にしており，附帯条件を利用して計算時間を短縮することを目的として改良された加速列挙法である。

(3)　列挙法では解を系統的に網羅した Solution tree (図—2.1)，(図—4.1) を描き，Node のすべてを列挙して調べることにより最適解を求めることが出来る。しかし，調べるまでもなく最適解ではないことが事前に判明する Node の列挙は省略する。これが陰的列挙であり，これにより実際に調べる陽的列挙を少なくして，計算時間を短縮することが出来る。

ＦＯＡはバラスの解法と比べて，まず (i) Solution tree 自体に特徴がある。サイズが小さく，かつ，Solution tree の中での所在位置より Node の実行可能性，最適性 (図—3.2) に差がある。(ii) したがって，Node の列挙順序は機械的に決る。(iii) 3つの省略基準をもち，それにより陰的列挙を多くすることが出来る (図—4.4)，(図—4.5)。このアルゴリズムをプログラミングして事例問題 (表—5.1) により計算時間を計った。

(4)　その結果，計算時間は問題のサイズではなく，陽的列挙数に比例することがわかった。(表—5.7)。陽的列挙数は問題のサイズと陽的列挙率との積であるから，この陽的列挙率 (表—5.8) を小さくすることがＦＯＡのアルゴリズムの能力であるといえる。

陽的列挙率は問題の制約条件に左右されるが，計算事例では0.099～0.00007％ (表—5.8) であった。すなわち，全体の Node のうち99.406～99.99993％の Node が省略された。東大大型計算機センターを対象として，計算時間5分程度で解きうる問題のサイズは30～35個小班について，3～4分期間の収穫予定問題である。

(5)　団地毎の収穫予定を立案するために，ＦＯＡを適用した。(表—6.1) 木曾谷国有林の1つの事業区を3つの団地に分け，団地ごとに伐採，造林事業量が安定して，継続出来るように，将来4分期間の小班別収穫予定を試算した。(表—6.2) ここで得られた解は設定された

制約条件，目的関数に対する最適解にすぎないが，計画立案者の意志決定のための判断材料となる。

(6) 得られた最適解は伐採材積，面積の総量と同時に伐採を予定する小班が特定されている。(図—6.4) したがって，林道計画との対応，林分相互の隣接関係や伐採順序という森林の空間配置に関する資料を得ることが出来た。これは齢級表による収穫予定法では得ることの出来ない資料である。そこで，林道の開設状況を 0-1 計画の制約条件とすることにより，収穫計画と林道計画との結合を試みた。

地図上で路線と小班とを対応させて，林道の開設計画により，対応する小班の伐出の可否を判定した。その結果を 0-1 計画の制約条件 (6.3) 式として追加した。事例の試算結果は，林道計画を収穫予定の条件とすることによる収穫量水準の減少量を明らかにした。(図—6.3)

(7) 小班の伐採分期を地図上に描くことにより，将来の森林の配置を予測することが出来る。(図—6.5) したがって，森林の配置の観点から，描かれた収穫予定案を評価することが出来る。そこで，林分の配置図を検討した結果，伐採林分の変更を指示するため，それらを 0-1 計画の制約条件として (6.1) 式の形により追加する。この問題を再計算することにより林分配置について，計画者の意向を取り入れた最適解を得ることが出来る。しかし，この方法では，制約条件が厳しく，選択できる小班が限られる場合は実行可能解を得られるとは限らない。あるいは1つの条件を追加することにより，伐採林分の配置が大幅に変るため，期待した配置が得られるとは限らない。

0-1 計画の制約条件として伐期齢，林道計画，林分相互の配置を追加することにより林分個別の事情は満足される反面，目的関数である総収穫量は減少した (表—6.9)。全体としての最適化と個々の要求とは一致しないことが示された。

(8) 目標とする望ましい林分配置を想定して森林に空間的な秩序を与えるために，伐採林分の場所規制と長期間の計画的な施業が必要である。この論文で提示した 0-1 計画は伐採林分の配置を目的関数として取り上げることは出来なかった。また制約条件としても，事後的に，しかも部分にしか扱っていない。したがって，森林の場所的規整を目的とした収穫予定法としては，まったく不十分であるといえる。ここで報告された小班別収穫予定は，伐採予定林分を指差し示すことにより保続的な収穫量計画に具体性を与えたこと，および将来の林分配置についての資料を提供したという役割を果しているにすぎない。

数理計画の最適解はあくまで森林計画の立案者のための判断資料である。したがって，この小班別収穫予定は不十分ではあるが，立案者の意志決定に有用な，今までにない資料を提供できると思われる。

謝　辞

この論文の作成にあたり，御指導をいただいた名古屋大学農学部鈴木太七教授に心から感謝します。鈴木教授には，この論文の基礎となった関連報文の作成段階から長期間にわたり，終始，懇切な助言をいただき，原稿校閲のため，貴重な多くの時間をさいていただいた。

また，研究資料として森林簿，森林図を貸していただいた前橋営林局，長野営林局，長野県庁の方々，本文のまとめに適切な助言と叱咤激励をいただいた信州大学農学部菅原聰教授，井上裕教官，資料整理の労をお願いした細井敏代嬢に厚くお礼を申し上げます。

データ処理には信州大学データステーション，東大大型計算機センターを利用させていただいた。

引 用 文 献

1) 天野正博・西沢正久：パラメトリック線型計画による保続計算の事例 86回日林講 45～46, 1975

2) Balas, E.: An Algorithm for Solving Linear Programs with Zero-one Variables. Opns Res. 517～546, July-August 1965

3) Balas, E.: Bivalent Programming by Implicit Enumeration. Encyclopedia of Computer Science and Technology, Vol. 2, 479～494. New York. 1975

4) Field, R. C., Dress, P. E., Fortson, J. C.: Complementary Linear and Goal Programming Procedures for Timber Harvest Scheduling. Forest Sci. 26, 121-133, 1980

5) 後藤元規・田中和博・鈴木太七：0-1 計画法としての収穫予定（II）小班群のグループ化 92回日林論 141～142, 1981

6) Greenberg, H.（真鍋龍太郎訳）：整数計画法 158 pp, 培風館，東京，1976

7) 井上由扶：森林経理学 298 pp, 地球社，東京，1974

8) 伊理正夫：線型計画法 現代数学全書3 147 pp, 白日社，1973

9) 科学技術庁資源局:木材の生産予測について(II)林業における収穫予定の数学的研究 54 pp, 1963

10) Kent, B. M.: Linear Programming in Land-Management Planning on National Forests. J. For. 78, 469～471, 1980

11) Kirby, M.: Land Use Planning, Transportation Planning, and Integer Programming. Proc. of Workshop System Analysis and Forest Resource Management. 271～284, University of Geogia, Athens, USA, 1975

12) 木平勇吉・井上裕：場所づけを考慮した保続計算 86回日林講 47～48, 1975

13) 木平勇吉：林道計画を含めた保続計算方法（予報）26回日林中支講 303～308, 1978

14) 木平勇吉：現行保続計算の問題点 信大農演報15 37～63, 1978

15) Konohira, Y.: A System for In-place Inventory and Sustained Yield Planning. Proc. of Workshop Forest Resource Inventories, 482～485, Colorado State University, Fort Collins, USA, 1979

16) Konohira, Y.: Feasible and Optimal Sustained-Yield Planning for National Forests in Japan. Proc. of IUFRO S4.4, Symposium on Forest Management Planning: Present Practice and Future Decision. 61～67, Blacksburg, Virginia, USA, 1980

17) 木平勇吉：国有林における林道計画と標準伐採量 91回日林論 107～108, 1980

18) 木平勇吉：場所づけ保続計算 28回日林中支講 193～194, 1980

19) 木平勇吉：収穫予定のための 0-1 線型計画の解法（I）加速列挙法の開発 日林誌63 345～353, 1981

20) 木平勇吉：収穫予定のための 0-1 線型計画の解法（II）適用方法と処理能力 日林誌63 392～399, 1981

21) 木平勇吉：収穫予定のための 0-1 線型計画の解法（III）林道計画を取り入れた収穫予定法と適用事例 日林誌64 27～31, 1982

22) 南雲秀次郎：線型計画法による資源計画（II）81回日林講 72～74, 1970

23) 南雲秀次郎：線型モデルによる収穫予定理論に関する研究 名大演報5 138～235, 1970

24) 南雲秀次郎：線型モデルによる収穫予定法の研究（Ⅰ）0-1計画法の適用　日林誌56 128～135, 1974
25) Navon, D. I.: TimberRAM a Long-Range Planning Method for Commercial Timber Lands under Multiple-Use Management. USDA Forest Service Research Paper PSW-70, 22 pp, 1971
26) 野村進行：森林経理考　250 pp, 農林出版, 東京, 1975
27) 岡崎文彬：森林経営計画　282 pp, 朝倉書店, 東京, 1955
28) Omi, P. N., Murphy, J. L., Wensel, L. C.: A Linear Programming Model for Wildland Fuel Management Planning. Forest Sci. Vol. 27, 81～94, 1981
29) Plane, D. R., McMillan Jr. C.（黒田充・豊田吉顕・田部勉・馬渡鎮夫訳）：整数計画法入門 223 pp, 培風館, 東京, 1976
30) 林野庁監修：国有林野経営規程の解説　477 pp. 地球出版, 東京, 1970
31) 鈴木太七：森林経理学　197 pp, 朝倉書店, 東京, 1979
32) 鈴木太七・田中和博：0-1計画法としての収穫予定　日林誌63 94～96, 1981
33) Teeguarden, D. E.: Forest Regulation 3, the Geographic Basis. J. For., 72, 217～220, 1974
34) 魚住侑司：林分配置に関する研究（Ⅰ）国有林野鳥取事業区における皆伐区の分析　鳥大農研報31巻, 208～213, 1979
35) 魚住侑司：森林施業法と林分の配置序列（Ⅰ）国有林野鳥取事業区の事例 31回日林関西支講 19～22, 1980
36) 魚住侑司：林分配置と森林の場所的規制との関連　国有林野山崎事業区における事例の分析　日林誌62 190～194, 1980
37) 魚住侑司：森林経理学における場所的規制理論の展開（Ⅰ）森林経理学論争以前の考察　91回日林講　105～106, 1980
38) 魚住侑司：森林経理学における場所的規制理論の展開（Ⅱ）森林経理学論争期の考察　92回日林論　109～110, 1981

附　　録

附録1　入力データの様式

① 小班数　分期数
10　4

② 小班名	面積 (ha)	伐期材積 (100m³) 1分期	2分期	3分期	4分期
1	95.		285.	342.	369.
31	92.	268.	290.	348.	
3	92.			252.	332.
26	85.		250.	300.	344.
2	83.	249.	304.		
5	80.	240.	271.	305.	
31	80.	320.	368.		
25	79.			330.	360.
4	77.	308.	332.	350.	
21	69.	250.	285.	342.	391.

③　制約条件の種類　　(注) 1……材積と面積　2……材積のみ　3……面積のみ
　1

④

伐採材積上限				伐採面積上限			
1分期	2分期	3分期	4分期	1分期	2分期	3分期	4分期
1,000	1,100	1,200	1,200	250	250	250	270

⑤

伐採材積下限				伐採面積下限			
1分期	2分期	3分期	4分期	1分期	2分期	3分期	4分期
700	700	700	800	160	170	180	190

⑥　データ終了
　99999

附録 2

FOA (Forestry Oriented Algorithm) Program リスト

東大大型計算機センター H200M 用

```
CARD      ISN        SOURCE STATEMENT

 1                 C     FORESTRY ORIENTED ALGORITHM FOR ZERO-ONE PROGRAMMING PROBLEMS
 2          2             DIMENSION AU(50,500),BU(50),AL(50,500),BL(50),C(500),BUW1(50),
 3                       1 IV(100,10),IVN(100),IC(100,10),ICN(100),IA(100),IAANS(100),
 4                       2 BLW1(50),BLW2(50),CMAX(100),ALL(50,501),CMAXX(101)
 5                       3,KA(100),AA(100),AB(100,10),KAB(100,10)
 6          6             COMMON IS,LEVEL,N,M,ZMAX,ZIS,NU,NL,NT,CW1,CW2,BUW1,BLW1,BLW2,ISMAX
 7          7             COMMON AU,BU,AL,BL,C,IV,IVN,IC,ICN,IA,IAANS,CMAX,ALL,CMAXX
 8          8             COMMON KA,AA,AB,KAB
 9          9             CALL DATA
10         10             CALL INITIA
11         11             CALL REPEAT
12         12             STOP
13         13             END


14          1             SUBROUTINE DATA
15          2             DIMENSION AU(50,500),BU(50),AL(50,500),BL(50),C(500),BUW1(50),
16                       1 IV(100,10),IVN(100),IC(100,10),ICN(100),IA(100),IAANS(100),
17                       2 BLW1(50),BLW2(50),CMAX(100),ALL(50,501),CMAXX(101)
18                       3,KA(100),AA(100),AB(100,10),KAB(100,10)
19          6             COMMON IS,LEVEL,N,M,ZMAX,ZIS,NU,NL,NT,CW1,CW2,BUW1,BLW1,BLW2,ISMAX
20          7             COMMON AU,BU,AL,BL,C,IV,IVN,IC,ICN,IA,IAANS,CMAX,ALL,CMAXX
21          8             COMMON KA,AA,AB,KAB
22          9             READ(5,100)N,M
23         10             DO 1 I=1,N
24         11             READ(5,101)KA(I),AA(I),(AB(I,J),J=1,M)
25         12           1 WRITE(6,220)KA(I),AA(I),(AB(I,J),J=1,M)
26         13         101 FORMAT(I5,7F10.0)
27         14         100 FORMAT(16I5)
28         15         220 FORMAT(1H ,I5,F8.2,5X,10F10.2)
29         16         200 FORMAT(1H ,13H SYOHAN-KOSU=,I4,12H   BUNKI-SU=,I3,4H NT=,I5)
30                  C
31                  C    SET  AND IV
32         19             NT=0
33         20             DO 10 I=1,N
34         21             JJ=0
35         22             DO 11 J=1,M
36         23             IF(AB(I,J))12,11,12
37         24          12 JJ=JJ+1
38         25             IVN(I)=IVN(I)+1
39         26             NT=NT+1
40         27             IV(I,JJ)=NT
41         28             KAB(I,J)=NT
42         29             C(NT)=AB(I,J)
43         30          11 CONTINUE
44         31          10 CONTINUE
45         32             WRITE(6,200)N,M,NT
46         33             WRITE(6,201)
47         34         201 FORMAT(1H...,6H C(I)=)
48         35             WRITE(6,202)(C(I),I=1,NT)
49         36         202 FORMAT(1H ,10F13.2)
50         37             TOTAL=0
51         38             DO 4 J=1,NT
52         39           4 TOTAL=TOTAL+C(J)
53         40             ZMAX=TOTAL/FLOAT(NT)*FLOAT(N)*0.9
54         41             WRITE(6,202)ZMAX
```

```
55                  C
56        43              READ(5,100)KWHICH
57                  C
58        45              IF(KWHICH-1)25,600,25
59        46           25 IF(KWHICH-2)800,700,800
60        47          600 CONTINUE
61                  C       SET UPPER LIMIT
62        49              NU=2*M
63        50              NL=2*M
64                  C       VOLUME LIMIT
65        52              DO 20 J=1,M
66        53              DO 21 I=1,N
67        54              IF(AB(I,J))22,21,22
68        55           22 AU(J,KAB(I,J))=AB(I,J)
69        56              AL(J,KAB(I,J))=AB(I,J)
70        57           21 CONTINUE
71        58           20 CONTINUE
72                  C       AREA LIMIT
73        60              DO 30 J=1,M
74        61              JJ=J+M
75        62              DO 31 I=1,N
76        63              IF(AB(I,J))32,31,32
77        64           32 AU(JJ,KAB(I,J))=AA(I)
78        65              AL(JJ,KAB(I,J))=AA(I)
79        66           31 CONTINUE
80        67           30 CONTINUE
81        68              GO TO 900
82        69          700 CONTINUE
83                  C VOLUME LIMIT ONLY
84                  C       SET UPPER LIMIT
85        72              NU=M
86        73              NL=M
87                  C       VOLUME LIMIT
88        75              DO 720 J=1,M
89        76              DO 721 I=1,N
90        77              IF(AB(I,J))722,721,722
91        78          722 AU(J,KAB(I,J))=AB(I,J)
92        79              AL(J,KAB(I,J))=AB(I,J)
93        80          721 CONTINUE
94        81          720 CONTINUE
95        82              GO TO 900
96        83          800 CONTINUE
97                  C AREA LIMIT ONLY
98        85              NU=M
99        86              NL=M
100                 C       AREA LIMIT
101       88              DO 830 J=1,M
102       89              JJ=J
103       90              DO 831 I=1,N
104       91              IF(AB(I,J))832,831,832
105       92          832 AU(JJ,KAB(I,J))=AA(I)
106       93              AL(JJ,KAB(I,J))=AA(I)
107       94          831 CONTINUE
108       95          830 CONTINUE
109       96              GO TO 900
110       97          900 CONTINUE
111                 C
112       99          203 FORMAT(1H ,15H AU(I,J),BU(I)=)
113      100          102 FORMAT(4(I5,I5,F10.0))
114      101              READ(5,110)(BU(I),I=1,NU)
115      102          110 FORMAT(8F10.0)
116      103              WRITE(6,203)
117      104              DO 9 I=1,NU
118      105            9 WRITE(6,202)((AU(I,J),J=1,NT),BU(I))
119      106          206 FORMAT(1H ,40I3)
120                 C
121      108              READ(5,110)(BL(I),I=1,NL)
122      109              WRITE(6,204)
123      110          204 FORMAT(1H ,15H AL(I,J),BL(I)=)
124      111              DO 2 I=1,NL
125      112            2 WRITE(6,202)((AL(I,J),J=1,NT),BL(I))
126                 C
127      114              DO 40 I=1,N
128      115              CMAX(I)=0
129      116              DO 41 J=1,M
130      117              IF(CMAX(I)-AB(I,J))42,41,41
131      118           42 CMAX(I)=AB(I,J)
132      119           41 CONTINUE
133      120           40 CONTINUE
134      121              DO 5 I=1,NL
135      122              ALL(I,NT+1)=0
136      123              ALL(I,NT)=AL(I,NT)
```

```
      DO 6 J=2,NT
      JJ=NT-J+1
      ALL(I,JJ)=AL(I,JJ)+ALL(I,JJ+1)
    6 CONTINUE
      WRITE(6,202)(ALL(I,JJ),JJ=1,NT)
    5 CONTINUE
C
       WRITE(6,206)(IVN(I),I=1,N)
      WRITE(6,207)
  207 FORMAT(1H ,9H IV(I,J)=)
      WRITE(6,206)((IV(I,J),J=1,M),I=1,N)
C
C     WRITE(6,209)
C     WRITE(6,202)(CMAX(I),I=1,N)
  208 FORMAT(1H ,10H ALMAX(I)=)
  209 FORMAT(1H ,9H CMAX(I)=)
C
      CMAXX(N+1)=0
      CMAXX(N)=CMAX(N)
      DO 7 I=2,N
      II=N-I+1
      CMAXX(II)=CMAXX(II+1)+CMAX(II)
    7 CONTINUE
      WRITE(6,202)(CMAXX(I),I=1,N)
      RETURN
      END
```

```
      SUBROUTINE INITIA
      DIMENSION AU(50,500),BU(50),AL(50,500),BL(50),C(500),BUW1(50),
     1 IV(100,10),IVN(100),IC(100,10),ICN(100),IA(100),IAANS(100),
     2 BLW1(50),BLW2(50),CMAX(100),ALL(50,501),CMAXX(101)
     3,KA(100),AA(100),AB(100,10),KAB(100,10)
      COMMON IS,LEVEL,N,M,ZMAX,ZIS,NU,NL,NT,CW1,CW2,BUW1,BLW1,BLW2,ISMAX
      COMMON AU,BU,AL,BL,C,IV,IVN,IC,ICN,IA,IAANS,CMAX,ALL,CMAXX
      COMMON KA,AA,AB,KAB
      IS=1
      LEVEL=1
C     SELECT CANDIDATES
      IWORK=IVN(LEVEL)
      DO 50 I=1,IWORK
      DO 10 II=1,NU
      IF(AU(II,IV(LEVEL,I))-BU(II))10,10,50
   10 CONTINUE
C
      DO 20 II=1,NL
      TOTAL=ALL(II,1)
      IF(TOTAL-BL(II))50,20,20
   20 CONTINUE
C
   30 ICN(LEVEL)=ICN(LEVEL)+1
      IC(LEVEL,ICN(LEVEL))=IV(LEVEL,I)
   50 CONTINUE
      CALL PRINT1
      IF(ICN(LEVEL))60,60,70
   60 WRITE(6,200)
  200 FORMAT(1H ,16H STOP AT ATEP-60)
      RETURN
C     SET SOLUTION IA(LEVEL)
   70 IA(LEVEL)=IC(LEVEL,ICN(LEVEL))
C     CANCEL
      IC(LEVEL,ICN(LEVEL))=0
      ICN(LEVEL)=ICN(LEVEL)-1
   90 RETURN
      END
```

```
      SUBROUTINE REPEAT
      DIMENSION AU(50,500),BU(50),AL(50,500),BL(50),C(500),BUW1(50),
     1 IV(100,10),IVN(100),IC(100,10),ICN(100),IA(100),IAANS(100),
     2 BLW1(50),BLW2(50),CMAX(100),ALL(50,501),CMAXX(101)
     3,KA(100),AA(100),AB(100,10),KAB(100,10)
      COMMON IS,LEVEL,N,M,ZMAX,ZIS,NU,NL,NT,CW1,CW2,BUW1,BLW1,BLW2,ISMAX
      COMMON AU,BU,AL,BL,C,IV,IVN,IC,ICN,IA,IAANS,CMAX,ALL,CMAXX
      COMMON KA,AA,AB,KAB
 1000 CONTINUE
      LEVEL=LEVEL+1
      LEVEL1=LEVEL-1
      LEVEL2=LEVEL+1
      IF(LEVEL-N)1002,1003,1001
 1001 LEVEL=LEVEL-1
      GO TO 80
```

```
215        16          1003 CONTINUE
216        17               DO 5 I=1,NU
217        18               BUW1(I)=BUW1(I)+AU(I,IA(LEVEL1))
218        19             5 CONTINUE
219        20               DO 6 I=1,NL
220        21               BLW1(I)=BLW1(I)+AL(I,IA(LEVEL1))
221        22               BLW2(I)=0
222        23             6 CONTINUE
223        24               CW1=CW1+C(IA(LEVEL1))
224        25               CW2=0
225        26               GO TO 1004
226        27          1002 CONTINUE
227                   C    SELECT CANDIDATES
228        29               DO 1 I=1,NU
229        30               BUW1(I)=BUW1(I)+AU(I,IA(LEVEL1))
230        31             1 CONTINUE
231        32               DO 4 I=1,NL
232        33               BLW1(I)=BLW1(I)+AL(I,IA(LEVEL1))
233        34             4 CONTINUE
234                   C
235        36               CW1=CW1+C(IA(LEVEL1))
236                   C
237        38          1004 CONTINUE
238        39               DO 10 I=1,IVN(LEVEL)
239                   C    UPPER LIMIT
240        41               DO 11 II=1,NU
241                   C    WRITE(6,500)I,II,BUW1(II),AU(II,IV(LEVEL,I)),BU(II)
242        43           500 FORMAT(1H ,1HUPPER-CHECK,8H I=,II=,,2I5,3F10.0)
243        44               IF(BUW1(II)+AU(II,IV(LEVEL,I))-BU(II))11,11,15
244        45            11 CONTINUE
245                   C    LOWER LIMIT
246        47               DO 12 II=1,NL
247                   C    WRITE(6,501)I,II,BLW1(II),BLW2(II),AL(II,IV(LEVEL,I)),BL(II)
248        49           501 FORMAT(1H ,11H LOWERCHECK,2I5,4F10.0)
249        50               IF(BLW1(II)+ALL(II,IV(LEVEL2,1))+AL(II,IV(LEVEL,I))-BL(II))16,12,
250                           112
251        52            12 CONTINUE
252                   C    ZMAX LIMIT
253                   C    WRITE(6,502)CW1,CW2,C(IV(LEVEL,I)),ZMAX
254        55           502 FORMAT(1H ,10HZMAX-CHECK,4F10.0)
255        56               IF(CW1+CMAXX(LEVEL2)    +C(IV(LEVEL,I))-ZMAX)17,13,13
256                   C   SET CANDIDATE
257        58            13 ICN(LEVEL)=ICN(LEVEL)+1
258        59               IC(LEVEL,ICN(LEVEL))=IV(LEVEL,I)
259        60               GO TO 10
260        61            15 CONTINUE
261                   C    WRITE(6,210)LEVEL,I,II
262        63               GO TO 10
263        64            16 CONTINUE
264                   C    WRITE(6,211)LEVEL,I,II
265        66               GO TO 10
266        67            17 CONTINUE
267                   C    WRITE(6,212)LEVEL,I,II
268        69           210 FORMAT(1H ,23HREJECTED BY UPPER LIMIT,3I5)
269        70           211 FORMAT(1H ,24H REJECTED BY LOWER LIMIT,3I5)
270        71           212 FORMAT(1H ,24H REJECTED BY Z-MAX LIMIT,3I5)
271        72            10 CONTINUE
272                   C    CALL PRINT3
273                   C
274        75            29 IF(ICN(LEVEL))30,30,40
275        76            30 LEVEL=LEVEL-1
276        77               IF(LEVEL)31,31,32
277        78            32 CONTINUE
278        79               DO 33 I=1,NU
279        80               BUW1(I)=BUW1(I)-AU(I,IA(LEVEL))
280        81            33 CONTINUE
281        82               DO 34 I=1,NL
282        83               BLW1(I)=BLW1(I)-AL(I,IA(LEVEL))
283        84            34 CONTINUE
284        85               CW1=CW1-C(IA(LEVEL))
285        86               IA(LEVEL)=0
286        87               GO TO 29
287        88            31 WRITE(6,201)ISMAX,ZMAX,IS
288        89               WRITE(6,202)(IAANS(I),I=1,N)
289        90           201 FORMAT(1H ,7H ISMAX=,I5,6H ZMAX=,F12.2,8HLAST-IS=,I5)
290        91           202 FORMAT(1H ,60I2)
291        92               RETURN
292                   C    SET SOLUTION
293        94            40 IA(LEVEL)=IC(LEVEL,ICN(LEVEL))
294        95               IS=IS+1
295                   C    CANCEL
296        97               IC(LEVEL,ICN(LEVEL))=0
297        96               ICN(LEVEL)=ICN(LEVEL)-1
```

```
      IF(IS-IS/5555*5555-1)800,801,800
  801 CONTINUE
C     CALL PRINT3
      WRITE(6,400)IS,LEVEL,(IA(I),I=1,LEVEL)
  400 FORMAT(1H ,2I6,40I3)
  800 CONTINUE
      GO TO 1000
C     CHECK FEASIBILITY
   80 CONTINUE
C     NEW OPTIMAL SOLUTION
      CWORK=CW1+C(IA(LEVEL))
      IF(CWORK-ZMAX)70,70,66
   66 ZMAX=CWORK
      ZIS=CWORK
      ISMAX=IS
      DO 67 I=1,N
   67 IAANS(I)=IA(I)
      CALL PRINT4
C     CHECK ZMAX TEST
      CALL ZTEST
C
   70 CONTINUE
      GO TO 29
   90 CONTINUE
      RETURN
      END
```

```
      SUBROUTINE ZTEST
      DIMENSION AU(50,500),BU(50),AL(50,500),BL(50),C(500),BUW1(50),
     1 IV(100,10),IVN(100),IC(100,10),ICN(100),IA(100),IAANS(100),
     2 BLW1(50),BLW2(50),CMAX(100),ALL(50,501),CMAXX(101)
      COMMON IS,LEVEL,N,M,ZMAX,ZIS,NU,NL,NT,CW1,CW2,BUW1,BLW1,BLW2,ISMAX
      COMMON AU,BU,AL,BL,C,IV,IVN,IC,ICN,IA,IAANS,CMAX,ALL,CMAXX
      WRITE(6,100)
  100 FORMAT(1H ,5HZTEST)
C     CALL PRINT3
      CWP=0
      DO 60 LEVEL=2,N
      CWP=CWP+C(IA(LEVEL-1))
      IF(ICN(LEVEL))60,60,1
    1 ICNWOR=ICN(LEVEL)
      DO 40 J=1,ICNWOR
      JJ=ICNWOR+1-J
      IF(CWP+C(IC(LEVEL,JJ))+CMAXX(LEVEL+1)-ZMAX)2,2,40
    2 DO 30 JJJ=1,JJ
      IC(LEVEL,JJJ)=0
      ICN(LEVEL)=ICN(LEVEL)-1
   30 CONTINUE
      GO TO 41
   40 CONTINUE
   41 CONTINUE
C  NARABIKAE
      IVNWOR=IVN(LEVEL)
      DO 50 J=1,IVNWOR
      IF(IC(LEVEL,J))51,51,50
   51 DO 55 JJJ=J,IVNWOR
      IC(LEVEL,JJJ)=IC(LEVEL,JJJ+1)
   55 CONTINUE
   50 CONTINUE
   60 CONTINUE
      LEVEL=N
C     CALL PRINT3
      WRITE(6,101)
  101 FORMAT(1H ,10H ZTEST END)
      RETURN
      END
```

```
      SUBROUTINE PRINT1
      DIMENSION AU(50,500),BU(50),AL(50,500),BL(50),C(500),BUW1(50),
     1 IV(100,10),IVN(100),IC(100,10),ICN(100),IA(100),IAANS(100),
     2 BLW1(50),BLW2(50),CMAX(100),ALL(50,501),CMAXX(101)
     3,KA(100),AA(100),AB(100,10),KAB(100,10)
      COMMON IS,LEVEL,N,M,ZMAX,ZIS,NU,NL,NT,CW1,CW2,BUW1,BLW1,BLW2,ISMAX
      COMMON AU,BU,AL,BL,C,IV,IVN,IC,ICN,IA,IAANS,CMAX,ALL,CMAXX
      COMMON KA,AA,AB,KAB

      WRITE(6,200)IS,LEVEL,ZMAX,ZIS
  200 FORMAT(1H ,4H IS=,I5,8H  LEVEL=,I4,11H  ZMAX,ZIS=,2F14.2)
      RETURN
      END
```

```
      SUBROUTINE PRINT2
      DIMENSION AU(50,500),BU(50),AL(50,500),BL(50),C(500),BUW1(50),
     1 IV(100,10),IVN(100),IC(100,10),ICN(100),IA(100),IAANS(100),
     2 BLW1(50),BLW2(50),CMAX(100),ALL(50,501),CMAXX(101)
     3,KA(100),AA(100),AB(100,10),KAB(100,10)
      COMMON IS,LEVEL,N,M,ZMAX,ZIS,NU,NL,NT,CW1,CW2,BUW1,BLW1,BLW2,ISMAX
      COMMON AU,BU,AL,BL,C,IV,IVN,IC,ICN,IA,IAANS,CMAX,ALL,CMAXX
      COMMON KA,AA,AB,KAB
      DO 1 I=1,NU
    1 WRITE(6,200)BU(I),BUW1(I)
  200 FORMAT(1H ,7H BU(I)=, F12.2,10H  BUW1(I)=,F12.2)
      DO 2 I=1,NL
    2 WRITE(6,201)BL(I),BLW1(I),BLW2(I)
  201 FORMAT(1H ,7H BL(I)=,F12.2,19H  BLW1(I)=,BLW2(I)=,2F12.2)
      WRITE(6,202)CW1,CW2
  202 FORMAT(1H ,9H CW1,CW2=,2F12.2)
      RETURN
      END
```

```
      SUBROUTINE PRINT3
      DIMENSION AU(50,500),BU(50),AL(50,500),BL(50),C(500),BUW1(50),
     1 IV(100,10),IVN(100),IC(100,10),ICN(100),IA(100),IAANS(100),
     2 BLW1(50),BLW2(50),CMAX(100),ALL(50,501),CMAXX(101)
     3,KA(100),AA(100),AB(100,10),KAB(100,10)
      COMMON IS,LEVEL,N,M,ZMAX,ZIS,NU,NL,NT,CW1,CW2,BUW1,BLW1,BLW2,ISMAX
      COMMON AU,BU,AL,BL,C,IV,IVN,IC,ICN,IA,IAANS,CMAX,ALL,CMAXX
      COMMON KA,AA,AB,KAB
      DO 1 I=1,LEVEL
    1 WRITE(6,200)(ICN(I),IA(I),(IC(I,J),J=1,M))
  200 FORMAT(1H ,8H ICN(I)=,I3,8H  IA(I)=,I3,10H  IC(I,J)=,10I4)
      RETURN
      END
```

```
      SUBROUTINE PRINT4
      DIMENSION AU(50,500),BU(50),AL(50,500),BL(50),C(500),BUW1(50),
     1 IV(100,10),IVN(100),IC(100,10),ICN(100),IA(100),IAANS(100),
     2 BLW1(50),BLW2(50),CMAX(100),ALL(50,501),CMAXX(101)
     3,KA(100),AA(100),AB(100,10),KAB(100,10)
      COMMON IS,LEVEL,N,M,ZMAX,ZIS,NU,NL,NT,CW1,CW2,BUW1,BLW1,BLW2,ISMAX
      COMMON AU,BU,AL,BL,C,IV,IVN,IC,ICN,IA,IAANS,CMAX,ALL,CMAXX
      COMMON KA,AA,AB,KAB
      WRITE(6,200)IS,LEVEL,ZIS
      WRITE(6,201)(IA(I),I=1,N)
  200 FORMAT(1H ,3HIS=,I5,7H LEVEL=,I5,14H FEASIBLE-ZIS ,F12.2,5H ANS=)
  201 FORMAT(1H ,60I2)
      RETURN
      END
```

In-place timber harvest scheduling by zero-one integer program

By Yukichi KONOHIRA
Laboratory of Forest Utilization, Fac. Agric., Shinshu Univ.

Summary

The purposes of this paper are: (1) to identify the individual forest compartments that are scheduled for cutting in each planning period so that harvest scheduling plans can be clear and concrete, (2) to find an efficient method for harvest scheduling that can regulate the spacial order of forest compartment allocation.

Soundness and function of forests can be increased when individual forests are well allocated and an ideal spacial order is attained. The only means whereby spacial order for the forest management is realized is the harvest. Therefore, the harvest scheduling must indicate not only the cutting volume or the cutting area as a whole, but also each individual forest location scheduled for cutting.

On the assumption that a compartment is the unit of harvest, scheduling problems can be defined as the problem of deciding the cutting period for each compartment, and it can be treated as a zero-one integer linear program. Because, once the cutting period of each compartment is determined, the total cutting volume and area can be easily calculated by summing up the individual compartments. Further, this type of plan itself gives extreamly concrete information to the planners, because the location of individual forests are already clarified.

This method is called "in-place" timber harvest scheduling.

This forest management problem is formulated as a mathematical problem and a solution is sought using a zero-one integer program:

Find the set of binary variables x_{ij} corresponding to the cutting of compartment i in planning period j. If x_{ij} is 1, compartment i should be cut in period j, and if x_{ij} is 0, compartment i should not be cut in period j.

The objective function: total yield over the whole planning period should be maximized. To find the most desirable yield plan under some managerial constraints such as the limit of cutting volume or area in each period, cutting age of each species, forest road construction and allocation of compartments, the following zero-one integer program is used:

Find x_{ij} $\begin{Bmatrix} i=1,\ 2,\cdots\cdots,\ n.\ \ n: \text{number of compartment} \\ j=1,\ 2,\cdots\cdots,\ m.\ \ m: \text{number of planning period} \end{Bmatrix}$ such that

$$Z=\sum_{i=1}^{n}\sum_{j=1}^{m} v_{ij}\cdot x_{ij} \quad \max$$

Subject to

(1) $x_{ij}=0$ or 1

(2) Limit of total cutting volume in each period

$$V_j \leqq \sum_{i=1}^{n} v_{ij}\cdot x_{ij} \leqq V'_j \quad (V_j,\ V'_j: \text{lower and upper limit of total cutting volume})$$

(3) Limit of total cutting area in each period

$$A_j \leqq \sum_{j=1}^{n} a_i\cdot x_{ij} \leqq A'_j \quad (A_j,\ A'_j: \text{lower and upper limit of total cutting area})$$

(4) Each compartment should be cut only once during a rotation

$$\sum_{j=1}^{m} x_{ij}=1$$

(5) All coefficents : v_{ij} (Volume of compartment i in period j), a_i (: Area of compartment i) are nonnegative

$$0 \leqq v_{ij},\ 0 \leqq a_i$$

(6) Stands grow and the volume does not decrease.

$$v_{ij} \leqq v_{i,j+1}$$

An application of a zero-one integer algorithm to the forestry problem was reported by Egon Balas in 1964.

The zero-one integer algorithm is clear in logic and the number of solutions are limited, therefore an optimal solution can be solved by enumeration procedure.

However, as the number of variables x_{ij} become numerous, it is very difficult to find an optimal solution because of the tremendous computation load, and thus it has not been applied to practical problems yet.

An efficient algorithm appliable to forestry problems was developed and is explained in this paper. A forestry oriented algorithm (FOA) which is based on Balas' additive algorithm is designed for solving harvest scheduling zero-one problems mentioned above, using its unique characteristics. That is,
FOA can accelerate the computation time on the assumption that (1) every compartment should be cut only once during a rotation, (2) all coefficients are nonnegative, and (3) stands grow and the volume does not decrease.

A systematic enumeration procedure is repeated successively according to the nodes shown on the solution tree. When any evidence is found on any node that no feasible or optimal solution exists, that node is abandoned, and the computation is abbreviated. This procedure is called "implicit" enumeration.

The characteristics of FOA which differ from Balas' additive algorithm are: (1) the solution tree itself is smaller in size and each node has different possibilities for optimality and feasiblity of the solution according to the location on the tree, therefore, (2) the enumeration sequence of nodes can be determined automatically, (3) the three standards that may become the evidence for abbreviation of enumeration are established so that the number of exhaustive enumerations (enumeration that is procedured in practice) is eliminated.

Before applying this program, meaningless variables must be rejected to reduce the size of zero-one problems, and subcompartments must be arranged in the decreasing order of their area in order to accelerate the computation.

This algorithm was translated into a computer program and the computation time is examined by using numerical examples. It becomes clear that the computation time lengthens in proportion to the problem's size, however, (1) only a short time is needed when the constraints are lax, that is, there is a wide range of allowable cutting volume or cutting area in each period, and (2) on the contrary, when the constraints are limited, the computation time becomes notably longer. Further, it is clear that the computation time is strictly proportioned to the number of exhaustive enumerations. Define the ratio r : r=(the number of exhaustive enumerations) / (the number of all nodes on the solution tree), the ratio r can be seen as the efficiency of FOA program.

Although the ratio r changes mainly depend on the constraints as mentioned

Example	Size	Constraints		Ratio r (%)
		Limit of cutting Volume (m^3)	Limit of cutting area (ha)	
1	2.151×10^6		47~ 87	0.01664
2	1.699×10^8	1,200~2,500	280~470	0.01108
3	1.699×10^8	1,400~2,000	300~440	0.03245
4	1.699×10^8	1,500~1,700	350~420	0.09917
5	1.223×10^{10}	1,300~2,300		0.00081
6	1.223×10^{10}	1,500~2,100		0.00435
7	1.223×10^{10}	1,600~1,900		0.01102
8	3.914×10^{11}		300~600	0.00007
9	3.914×10^{11}		390~560	0.00060
10	3.914×10^{11}		430~540	0.00080

above, r is between 0.099～0.00007 % by the numerical examples as shown in the table. That is, from 99.9 % to 99.99993 % of whole nodes are abandoned and the computation is abbreviated.

As a result, FOA can become practical and efficient for forestry oriented zero-one problems. It can be concluded that the timber harvest scheduling problem by individual compartments, although the number of compartments are limited, can be solved by a zero-one integer linear program.

This FOA program was applied to Kiso national district forest to predict a sustained yield plan, and a plan was formed. The forest is divided into three working sections, and in each section, the cutting period of each compartment is scheduled so that logging and cilviculture can be sustained continually for 20 years.

The optimal solution indicates not only the total cutting volume and area, but also clarifies the cutting period of the individual compartments. Therefore, each compartment is evaluated as to whether it can be logged or not in each of the planning periods. The cutting sequence or neighbouring relationship of each cutting compartment can be discussed.

As an element of harvest scheduling, forest road planning is formulated to the constraints in a zero-one problem and some examples are tested. A forest road plan is drawn on a map and the roads are difinded as to which compartments they intersect. When schedule of road construction is recieved, then the compartments which are not ready to transport are rejected from the immediate cutting schedule.

This harvesting schedule was applied to a forest of 20 compartments and the results follow : By comparing the problem with road planning and the problem without road planning, it becomes clear that (1) if the limits of the cutting volume or cutting area in each period are wide, feasible and optimal solutions can be found in both problems. (2) On the contrary, when the limits are narrow, a feasible solution does not exist in the problem with road planning.

Forest allocation in the future are visualized and discussed. A disirable cutting period of each compartment indicated by the optimal solution is drawn on the map, and the spacial order is evaluated from managerial and/or environmental view points. Requirements of any compartment that need its cutting schedule changed, are added to the constraints of the zero-one problem. Then the revised problem is recalculated and the new optimal solution is shown on the map. By this repetition, a final optimal solution that can satisfy the spacial order intended by planners is found.

In the examples the results show that the total cutting volume decreases when the constraints are limited, that is, the demands of the individual compartments and the demands of optimization of the whole are opposed to each other.

It can be concluded that to attain an ideal spacial order in a management

forest, a strict scheduling of cutting compartments is required. The zero-one problem discussed in this paper can contribute to this requirement. The auther believes an optimal solution of a mathematical problem is always information that may assist in the better decision making of planners, and the in-place harvest scheduling method surely gives useful information that was not previously available to the forest management planner.