

No. **666**

多期間資産選択問題への内点法の適用

by

竹原 均

February 1996

# 多期間資産選択問題への内点法の適用

筑波大学社会工学系 竹原 均

October 20, 1994

Revised August 15, 1995

摘要 : 本研究においては, 離散近似された多期間資産選択問題をネットワーク構造を用いて表現する. その上でこの問題が結果として帰着される大規模線形計画問題に対して内点法を適用することにより, 最適ポートフォリオを求めることが可能であることを示す.

**Key Words:** Interior Point Algorithm, Downside Risk Model, Multi Period Portfolio Selection Model, Asset and Liability Management

## 1 対象問題と目的

市場には投資対象として  $n$  資産が存在し, 投資家は今後  $T$  期間自らの資産を運用するものとする. 初期資産として  $W_0$  を現金で保有し, 時点  $t = 0, 1, \dots, T - 1$  においてその資産構成を変更するものとする. 時刻  $T$  においては時刻  $T - 1$  の投資決定の結果を得て, その時点ですべての資産を現金化して清算するものとする. 各期における資産収益率は確率変数として与えられるため, この問題は確率計画問題としてのモデル化は可能では有るものの一般的な状況においてはその最適解を求めることは非常に困難である.

本研究においては Rockafellar and Wets [3] が対象とした多期間シナリオツリーが与えられたもとの資産選択問題について内点法を適用し, その有効性について検証する. Rockafellar and Wets においてはシナリオツリーの Root から Leaf へのパス上で得られる scenario subproblem への分解とそれら子問題の解の最適解への射影による解法を提案している. これに対して本論文においてはこうした分解算法を行わず, 与えられた問題を deterministic な大規模線形計画問題へと帰着し, 内点法をその問題に対して直接に適用する. 過去の多くの数値実験により, 内点法により問題の規模が大きいほど単体法と比較してより少ない計算時間で最適解が得られることが多いことが知られているが, こうした資産選択の現実問題に対して適用した場合にどのような結果が得られるかについては不明である. ここでは計算機上で実装を主体として, シナリオツリーと必要とされるデータを格納するデータ構造を決定する. そしてその構造を利用して Matrix Generator を作成した. これにより, 1) 多期間資産選択問題を計算機上で表現可能であること, そして 2) その最適ポートフォリオが実際に計算可能であること, 3) 内点法と単体法の比較を行い, 内点法の振る舞いについて分析を行うことの 3 点がこの研究の目的である.

## 2 多期間下方リスクモデルの記述

図1に与えられるような多期間シナリオが既に与えられているとする。現在の時刻は0として1期後( $t = 1$ )では複数の状態のうちいずれかが起こるものとする。それぞれの状態の生起確率、その状態が生起した場合の各資産の収益率は投資家には既知である。 $t = 0$ で投資家は各資産へ初期保有資産  $W_0$  を配分する。 $t = 1$ においてどの状態が起きたかにより  $t = 1$ での資産評価額は異なるため、当然その後の意志決定も異なることになる。そこで投資家は時刻  $t = 0, \dots, T - 1$ のすべての時点のすべての起こりうる状態での自らの投資行動を  $t = 0$ ですべて決定しなければならない。

シナリオツリーについて、以後ツリーの各結合点をノード (node) と呼ぶ。あるノードとそのノードが起きたときに次の期に起こりうるノードを parent と child と呼ぶ。ノードのうち child を持たないものを leaf, それ以外を internal と呼び、また親を持たないノードを root と呼ぶ。時刻  $t$  に対応して、各ノードはシナリオツリー上でのすべての枝の長さは1として root からの距離として height を持つ。実現確率、シナリオの実現値、最小許容収益率、資産評価額等はすべて各ノード情報として管理されるものとする。

また投資に関しては達成することが望まれる目標収益率が各ノードごとに既に与えられており、これもノード情報として管理されているものとする。ただしこの目標収益率は投資比率の決定とは独立に与えられているものとする。以後この値のことをフロアと呼ぶ。

さて internal node 数を  $k$  とすれば、資産数が  $n$  であるから少なくとも  $nk$  変数を決定しなければならない。仮に  $n = 9, T = 4$  として、Internal なノードが10個の Child へ接続しているとすると総ノード数は1111となるので  $nk = 9999$  となり、こうした問題が非常に大規模な問題に帰着されることは容易に推測される。

ここでは簡単化のため消費、計画期中での配当支払、新規資金の流入についてはこれを無視する。また、より複雑な形式を考慮することも可能であるが、ここではリスクについてその時間加法性を仮定することにする。この場合に以下のような目的関数の最小化を想定することができるであろう。

$$\alpha \sum_{v \in V} (1 + d)^{-h(v)} p_v (\max(\theta_v - W_s, 0))^k - \sum_{v \in L} p_v W_v \quad (1)$$

ここに

- $n$  資産数
- $V$  ノードの集合
- $L$  leaf であるノードの集合
- $W_v$  ノード  $v$  での資産評価額
- $d$  割引率
- $h(v)$  node  $v$  の height
- $\theta_v$  node  $v$  での資産評価に対するフロア一値
- $\alpha$  リスク回避度

上記の目的関数は MLPM Model (Mean Lower Partial Moment Model) を時間加法性の仮定のもとに多期間に拡張したものである。またこれ以降は LPM の次元を1に固定し線形計画問題へと問題を帰着させる。

次に制約条件は、一般化ネットワークフロー問題の流量保存則と通常のアセットアロケーション問題に対する制約条件の複合となる。ここでも簡単化のために、すべてのノードにおいて投資比

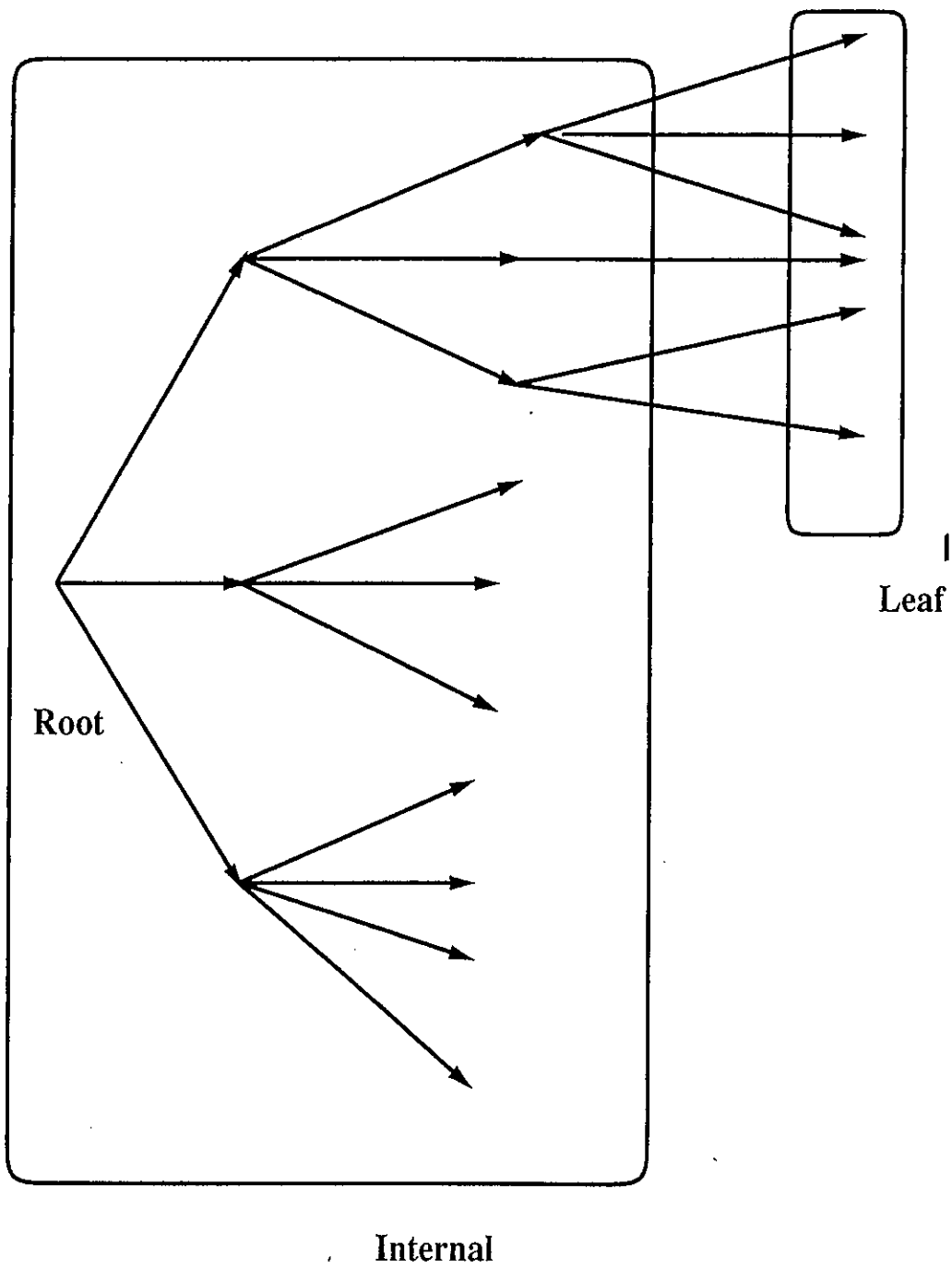


図 1: 多期間シナリオツリー

率  $y \in R^n$  が, 制約条件

$$\begin{aligned} e^t y &= 1 \\ b_u &\geq Ay \geq b_l \\ u &\geq y \geq l \end{aligned} \quad (2)$$

が満たすとする. ( $b_u, b_l \in R^m, A \in R^{m \times n}, u, l \in R^n$ .)

以後  $x_v \in R^n$  をノード  $v$  での投資額,  $x_p$  を parent での投資額,  $W_0$  を初期投資額とする. まず root においてはノード上での投資比率の制約のみを考慮すればよい. 従って,

$$\begin{aligned} e^t x_v &= W_0 \\ b_u W_0 &\geq Ax_v \geq b_l W_0 \\ u W_0 &\geq x_v \geq l W_0 \end{aligned} \quad (3)$$

である. root 以外の Internal なノードについては, ここでは配当と新規資金を考慮していないので自分の parent の投資結果としての資産評価額とノードでの再投資額が等しくなればよい. よって,

$$\begin{aligned} r_v^t x_p &= W_v \\ e^t x^s &= W_v \\ b_u &\geq A(x_v/W_v) \geq b_l \\ u &\geq x_v/W_v \geq l \end{aligned} \quad (4)$$

となる. ノードが leaf であればこの場合には再投資は必要無いため, 制約条件は

$$r_v^t x_p = W_v \quad (5)$$

である.

### 3 計算機上での実装

プログラムはC言語で記述されている. 各ノードは構造体により定義され, その node が実現したときの収益率, 生起確率, ノードでのポートフォリオ, 資産評価額等を構造体メンバーとして持つ. ただしポートフォリオ, 資産収益率は leaf においては必要としないため構造体には格納場所へのポインタのみを保有し, internal についてのみ領域を動的に確保する. また図2に示すように, parent と child の間には両方向の接続が与えられている. シナリオツリーが与えられた状況での線形計画問題へのモデル化は以下の手順による. まず root を開始点とする. root においては(3)式に基づく制約条件の処理と目的関数の処理を行う. 終了後 root の child のうちのどれか一つに移動する. その child が internal であれば(4)の処理を行い, さらにその child へと移動する. leaf に達した場合には(5)式の処理を行い, その leaf を探索済みとして leaf の parent へと戻り, child で探索済みでないものへ移動し処理を行う. 以後この処理を繰り返し, すべての child である leaf が探索済みとなった場合には, parent である internal node を探索済みとする. 以後このすべての child が探索済みとなった場合にその parent を探索済みとして, 1期前の node に戻るという手続きを続けることにより, 図3に示すように最終的に root が探索済みとなるのでこの段階で処理を終了とする. 従って root, root 以外の internal node, leaf での制約条件, 目的関数の処理を関数化してさらに再帰定義を用いることにより, シナリオから線形計画問題の係数行列等の生成は node を前処理, 深さ優先探索することにより実現できる.

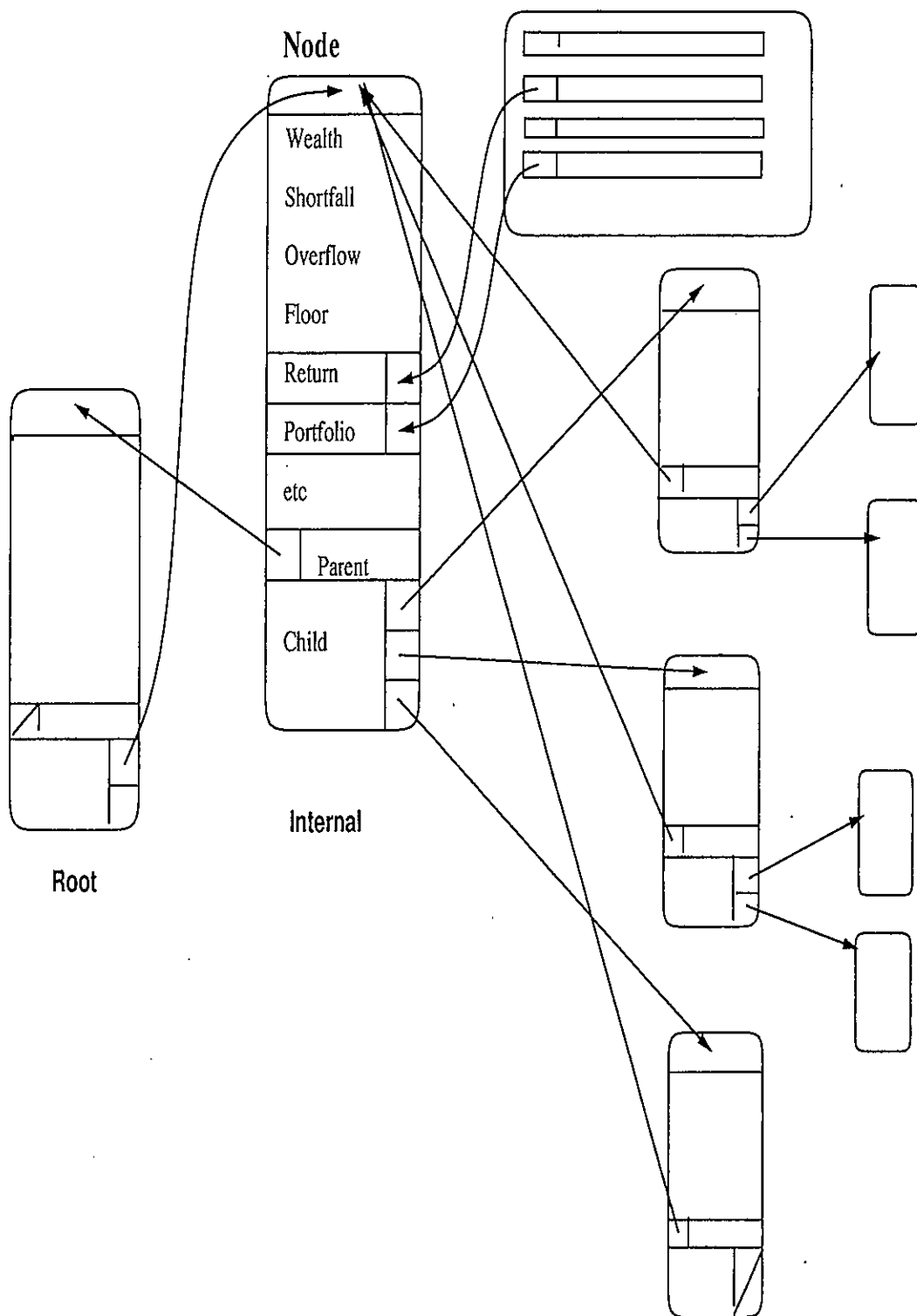


図 2: プログラムで用いたデータ構造

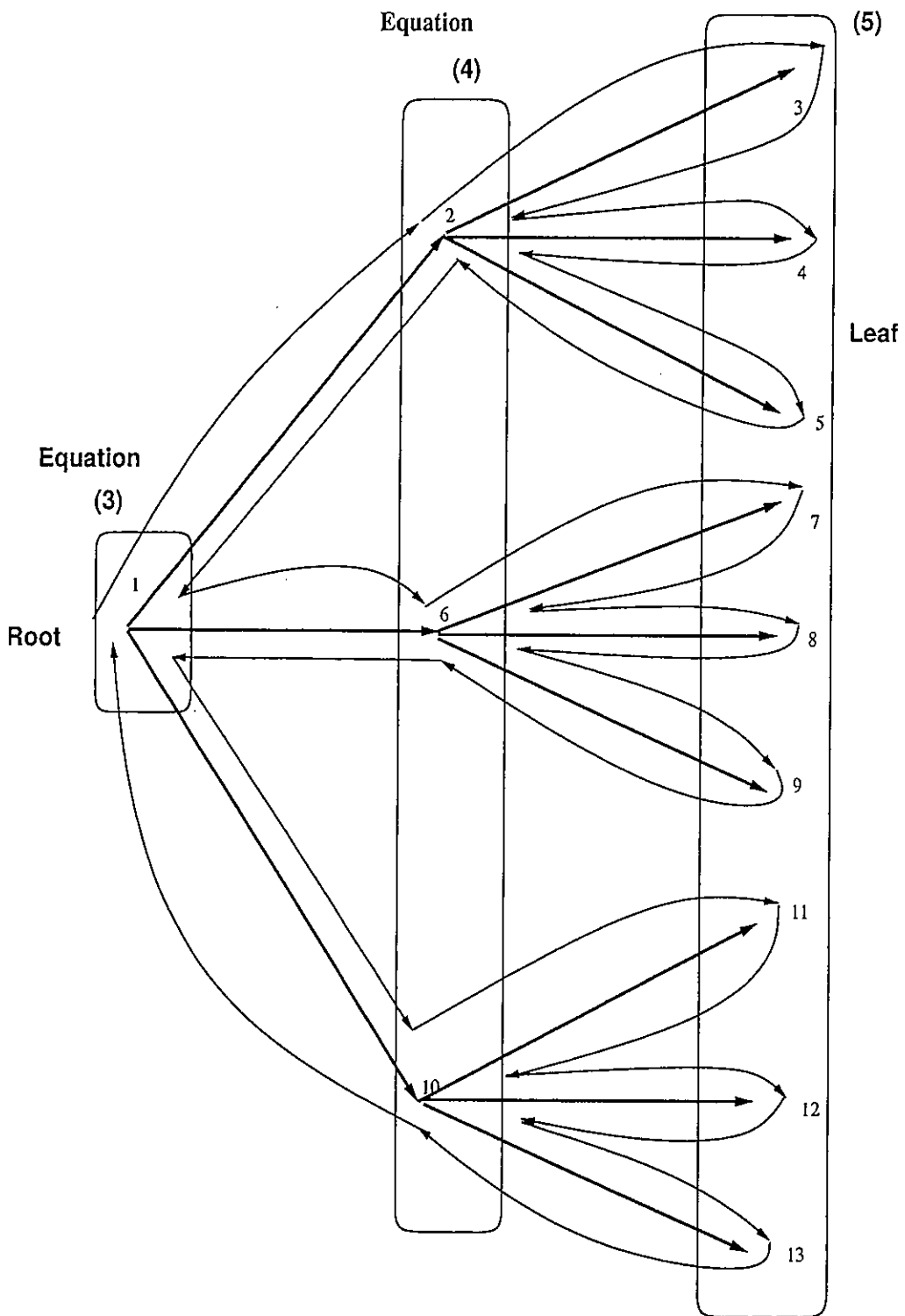


図 3: シナリオツリーとノードの探索と処理

## 4 計算機実験結果

結果として得られた線形計画問題は先にも述べたように非常に大規模となるため、最適ポートフォリオを求めることは困難であると考えられてきた。しかしながら、計算機の性能の向上と大規模数理計画問題に対するアルゴリズムの進歩、特に内点法アルゴリズムに対する研究の進展により、現在ではある程度の大きさの線形計画問題であれば、十分にこれを解くことが可能である。ここではこれまで説明したモデル、データ構造を用いて、計算機上で実現した結果について報告する。

テスト問題は以下のように設定した。まず投資対象資産は国内株、債券、転換社債、貸出金、安全資産、米国株式(ヘッジなし、フルヘッジ)、米国債券(ヘッジなし、フルヘッジ)の9資産とした。制約条件については空売りの禁止のみとした。シナリオツリーは最初にツリーの高さ(期間数)を3あるいは4と固定し、その上でツリーの分岐数を3~10と変化させて、比較的小規模の問題から大規模問題までに対応した。それぞれのツリーのノードに対応したシナリオの実現値は今回の場合は特定の予測モデルによらず、過去10年の実現収益率を埋めこんでいる。その他の設定として

- リスク許容度  $\alpha = 10.0$
- 初期資産  $W_0 = 1.0$
- フロアー  $\theta_v = W_0 1.055^h(v)$
- 割引率  $d = 0.02$  (2%)

としている。

使用計算機はIBM RS6000 Model 41T, CPU時間はシナリオツリーが完成している状態を仮定し、そこから行列生成、最適化、シナリオツリーへの各種情報のリストアまでを含むものとする。表5に期間数3,4の場合の実験結果を示す。図中において行数、列数、非零要素は帰着された線形計画問題の制約係数行列のサイズとそこに含まれる非零要素数を示している。またCPU時間の単位は秒である。

表より明らかにわかるように、ノード数で1000、列数で数1000程度の問題であれば数秒で最適な資産構成を求めることが可能である。シナリオ数が10000の場合、行数は20000、列数は40000を超えるがその場合でも4~5分程度のCPU時間で解が得られている。空売り制約のみを与えている点、資金の計画期間中での流出入が無い点など対象問題としては構造が単純であるから、この結果が実務的な状況に必ずしも対応しているとは言えないものの、シナリオに基づく多期間モデルは実用の範囲にあると考えてよいのではないか。

単体法との比較した場合に、特にシナリオ数が増加した場合に内点法の適用が非常に有効であることがわかる。4期間モデル、分岐数が9,10の場合には単体法を用いた場合には計算時間が1時間を越えるためCPU時間の計測を行っていない。しかしながらそれまでの内点法と単体法のCPU時間の比率の推移が示すように、より大規模な問題に関しては内点法の方が単体法と比較して数10倍高速であることが推測される。

## 5 結論

本研究においては、多期間シナリオツリーが与えられたもとの資産選択問題を一般化ネットワークフロー問題の流量保存と下方リスクモデルの考え方をもとに記述した。そしてこのモデルを表現するためのデータ構造を決定し、また分解算法を用いず、資産選択問題を決定的な大規模線形計画問題に帰着し、この問題に対して内点法を適用することの有効性を検証した。ここでのモ



分岐	ノード	行	列	非零	内点法	単体法	比率
3	40	91	234	636	0.72	0.72	1.00
4	85	189	441	1301	1.11	1.20	1.08
5	156	341	744	2324	1.31	2.19	1.67
6	259	559	1161	3783	1.55	4.17	2.69
7	400	855	1710	5756	2.48	9.37	3.78
8	585	1241	2409	8321	3.39	18.70	5.52
9	820	1729	3276	11556	5.82	35.40	6.08
10	1111	2331	4329	15539	8.84	67.68	7.66

(a) 3 期間モデルの場合

分岐	ノード	行	列	非零	内点法	単体法	比率
3	121	280	720	1959	1.16	1.73	1.49
4	341	765	1785	5269	2.82	8.43	2.99
5	781	1716	3744	11699	5.65	35.48	6.28
6	1555	3367	6993	22791	12.37	134.41	10.87
7	2801	6000	12000	40399	25.29	428.09	16.93
8	4681	9945	19305	66689	51.97	1115.23	21.46
9	7381	15580	29520	104139	122.68	—	—
10	11111	23331	43329	155539	262.38	—	—

(b) 4 期間モデルの場合

表 1: 資産選択問題のサイズと最適解を得るのに要した時間

デルの記述は1期間のMLPMモデルの時間加法性の仮定のもとでの拡張であり、年金ALM等の実用上の問題に対しても適用可能である。また計算機実験の結果が示すようにこの問題に対しては内点法が非常に有効であり、実際に許容される時間内に最適な投資戦略を見出すことが可能である。以上の点からこのモデルとアルゴリズムの実用性が十分に期待できる。

## 参考文献

- [1] J. R. Birge and L. Qi, "Computing block angular Karmarkar projections with applications to stochastic programming," *Management Science* 34 (1988) 1472-1479.
- [2] G. Infanger, "Monte Carlo (importance) sampling within a Benders' decomposition for stochastic linear programs," Technical Report, SOL 89-13, Department of Operations Research, Stanford University. (Stanford CA, 1989).
- [3] Rockafellor and R. J. Wets, "Scenarios and policy aggregation in in optimization under uncertainty," *Mathematics of Operations Research* 16 (1987) 32-49.
- [4] R. Van Slyke and R. J. Wets, "L-Shaped linear programs with application to optimal controll and stochastic programming," *SIAM Journal on Applied Mathematics*, 17 (1969) 638-663.
- [5] H. Takehara, "A penalty function approach for portfolio selection," *MTEC Working Paper No. T926, MTB Investment Technology Institute, 2-6-2 Ohtemachi, Chiyoda-ku, Tokyo 100 Japan* (1992).
- [6] H. Takehara, "An interior point algorithm for large scale portfolio optimization," *Annals of Operations Research* 45 (November 1993) 373-386.