論文 / 著書情報
Article / Book Information

題目(和文)	適応的探索による発電プラントの起動スケジューリング
Title(English)	
著者(和文)	神谷昭基
Author(English)	
出典(和文)	学位:博士(工学), 学位授与機関:東京工業大学, 報告番号:甲第3550号, 授与年月日:1997年3月26日, 学位の種別:課程博士, 審査員:
Citation(English)	Degree:Doctor (Engineering), Conferring organization: Tokyo Institute of Technology, Report number:甲第3550号, Conferred date:1997/3/26, Degree Type:Course doctor, Examiner:
学位種別(和文)	博士論文
Type(English)	Doctoral Thesis

博士（工学）論文

適応的探索による発電プラントの起動スケジューリング

Adaptive Search based Power Plant Start-up Scheduling

平成９年３月

指導教官　小林重信　教授
学籍番号　94D31020
氏　　名　神谷　昭基

目 次

1 序論 **1**

1.1 研究の背景 1

1.2 研究の目的及び意義 2

1.3 論文の構成 4

2 問題の設定と接近法 **5**

2.1 火力発電プラント起動スケジューリング問題 5

2.2 既存の研究 8

2.3 本研究の接近法 9

3 問題の特徴分析 **12**

3.1 目的関数と制約条件の性質 12

3.1.1 目的関数の強意単調性 12

3.1.2 制約関数の連続性 12

3.2 最適解の存在領域 13

3.3 境界領域近傍への強制 17

3.3.1 強制操作の概要 18

3.3.2 強制操作の詳細 21

4 GA による進化的探索 **24**

4.1 はじめに 24

4.2 コード化／交叉の方法 24

4.3 世代交替モデル 26

4.4 強制操作の導入 28

4.5 再利用機能とタブ戦略の導入 29

4.5.1 再利用機能の導入 29

4.5.2 タブサーチによる探索 30

4.5.3 タブ戦略を導入した GA による探索 32

4.6 実験 33

4.6.1 実験の目的と方法 33

4.6.2 実験結果 35

4.6.3 結果の考察 35
4.7 おわりに 40

5 GA と強化学習の融合による適応的探索 42

5.1 はじめに 42
5.2 複数境界に対する強制操作 42
5.3 複数境界探索戦略を導入した GA 44
5.4 強化学習の導入 46
5.5 GA と強化学習の融合 48
5.5.1 融合モデルによる探索 48
5.5.2 融合モデルによる学習 50
5.6 実験 52
5.6.1 実験の目的と方法 52
5.6.2 実験結果 53
5.6.3 結果の考察 54
5.7 おわりに 59

6 強化学習の報酬戦略に関する解析 61

6.1 はじめに 61
6.2 定義 61
6.2.1 出力空間 61
6.2.2 最適出力部分空間 62
6.2.3 学習環境と学習アルゴリズム 63
6.2.4 報酬戦略 64
6.3 報酬戦略に関する解析 65
6.3.1 正の報酬戦略 69
6.3.2 負の報酬戦略 69
6.3.3 正+負の報酬戦略 78
6.4 解析結果による報酬戦略の設計 86
6.5 実験 87
6.5.1 実験の目的と方法 87
6.5.2 実験結果 87

6.5.3 結果の考察 87
6.6 おわりに 89

7 結論 **90**
7.1 研究成果の要約 90
7.2 今後の研究課題 91

付録 **94**

謝辞 **97**

参考文献 **98**

1 序論

1.1 研究の背景

火力発電プラント起動スケジューリングとは，起動時間の最短化を図ると同時，起動過程に発生するタービンロータの最大熱応力を制限することである．昼夜間の電力需要の変動に対する負荷調整のため，近年，毎日起動停止の火力発電プラントが多くなってきている．起動時間の短縮は，起動損失の軽減，電力需要に対する供給側の追随性の向上，自然界の有限資源消費の節減や排出物の低減による環境汚染の抑制効果などが期待できる．さらに，任意に与えられる制限熱応力に対応した最適起動スケジュールをオンラインで提供することにより，より柔軟な運転が可能となる．

本問題に対して，古くから多くの研究が行われている．本問題は非線形性が強く多峰性であるため，従来の線形計画法や山登り法では，最適解を求めることが困難である．また，設計者のノウハウを利用した知識システムでは，最適解の質がシステムに導入される知識の質や量に左右され，知識収集コストがボトルネックであり，実世界問題に適用する場合，それに関する考慮が必要である．

自然界の進化は，集団を基本とした最適化過程である．進化型計算(Evolutionary Computation: EC)は，自然界の進化を模擬した工学的モデルであり，探索としての頑健性の側面を持ち，対象問題に関する専門知識や問題の線形性，微分可能性，単峰性という条件を必要とせず，困難な実世界の工学問題への適用に対する期待が大きい．進化型計算は，評価，選択，交叉と突然変異を繰り返しながら最適解の探索を行うが，本問題は，起動スケジュールの評価のためのタービンダイナミックシミュレーションによる最大熱応力計算は膨大な時間がかかり，さらにオンライン探索問題として空間サイズが大きい．進化型計算を本問題に適用した場合，プラント運転に要求されるオンライン探索性能を満たすため，効率的な探索モデルの構築が要請される．

学習により，問題解決能力の向上が期待できる．強化学習(reinforcement learning)は，与えられた環境において，環境に対する探索を行いながら，環境からの報酬(reward)を手掛かりに，環境に適応した行動を学習する機械学習モデルである．強化学習は，探索と学習能力を持ち合わせているので，強化学習を探索問題に適用した場合，学習効果による探索能力の向上が期待である．しかし，探索空間サイズの大きい実世界の工学問題において，効率的な学習モデルを構築するため，強化学習の探索効率の向上を図ることが要請される．

また，強化学習は環境から与えられる報酬により学習を行うので，強化学習を工学問題

に応用する場合，報酬の設計が重要であり，報酬の授与方法によって強化学習の学習性能が大きく影響される．しかし，従来では，強化学習の報酬の設計に関する研究がほとんど行われていない．

本論文では，工学応用という立場から，強化学習と進化型計算の融合による探索モデルに関する研究を行い，強化学習による学習効果と進化型計算による探索能力を融合することにより，効率のよい適応的探索モデルを実現し，探索空間サイズの大きい実世界問題である火力発電プラント起動スケジューリングへの適用を図る．さらに，報酬設計に関する理論的な解析を行い，その解析結果を高次元(high dimension)探索空間である火力発電プラント起動スケジューリング問題において実験による確認を行う．

進化型計算に関する研究は，遺伝的アルゴリズム(Genetic Algorithms: GA)，進化戦略(Evolution Strategies: ES)と進化的プログラミング(Evolutionary Programming: EP)がある[Fogel 94]が，GA は EP や ES に比べて，より一般的な枠組みを持つこと［小林 96］から，本論文では，進化型計算を GA により実現し，以下では GA という用語を使うことにする．

1.2 研究の目的及び意義

火力発電プラント起動スケジューリングは，制約付き最適化問題であるが，発生最大熱応力に対する制限が厳しい場合，起動時間を長くする必要があり，一方，制限が緩い場合，起動時間を短くすることが可能であるという経験的な事実により，本問題の最適解は実行可能解空間の境界付近に存在することが推定される．最適解が実行可能解空間の境界上に存在するので，解の全空間を探索することよりも境界近傍に沿った探索が効率的であると考えられる．制約付き最適化問題において，対象問題が単峰，微分可能で，最適解が実行可能解の境界上に存在する場合，従来より，有効制約戦略(active set strategy)と呼ばれる傾斜投射法(gradient projection method)があり[今野 78]，不等式制約条件に含まれる有効な制約式(active constraint)により定義された実行可能解空間の境界に沿った最適解の探索が可能である．しかし，本起動スケジューリング問題は，多峰であり，微分可能な式に変換することが困難であるため，従来の傾斜投射法による有効制約戦略の適用ができない．

本論文では，a) 本問題の最適解が実行可能解空間の境界上に存在することを明らかにし，b) 探索を実行可能解空間の境界近傍に限定する強制操作(enforcement operator)を提案し，c) 境界近傍探索のため，強制操作を組込んだ GA による効率的な探索モデルの構築を第一番目の研究目標とする．

近傍探索では，同じ解を繰り返して探索することがしばしば発生する．本論文では，繰

り返しに伴うコストを避けるため，強制操作を組込んだ GA に再利用機能とタブ戦略を導入することにより，効率のよい探索モデルが実現されることを実験によって確認する．ここで，再利用機能とは探索した解のシミュレーション計算結果を再利用することにより探索効率の向上を，タブ戦略とは最近探索した解への移動を禁止することにより探索効率の向上と局所的最適解からの早期脱出を図るものである．しかし，この探索を基本とした接近法だけでは，プラント運転に要求されるオンライン探索性能を充分満足できない．本問題のブレークスルーを図るため，強化学習と GA を融合したハイブリッド方式を提案することを第二番目の研究目標とする．

強化学習と GA の融合は，探索開始時と学習・探索過程に分けて，次の二つの効果が期待できる．

a) 強化学習は探索条件に対応する最適解を予め学習し，探索開始時では，学習効果により，任意に与えられる探索条件に対応する有望な解候補を生成し，最適解の探索を加速することが期待できる．
b) 学習または探索過程において，GA が強化学習を有望な領域で学習するようにガイドし，強化学習の学習を加速し，強化学習は，GA のガイドによる学習効果により，探索過程の序盤において，有望な解を生成することにより，GA の探索を加速する．このような相乗効果により，学習と探索の加速が同時に期待できる．

強化学習は環境から与えられる報酬が最大となるように，環境に対する自己適応を行うので，強化学習の工学応用において報酬の設計が重要な問題となる．一般に，報酬の種類として，正の報酬(positive reward)と負の報酬(negative reward)がある．強化学習がよい挙動が示された時に，正の報酬，悪い挙動の場合，負の報酬を与えることにより，強化学習がよい挙動を学習し，悪い挙動を避けることが期待できる．従来の強化学習に関する研究では，正の報酬，負の報酬，または両者の組合わせによる学習方法が用いられ，何れもよい学習結果が得られたことが報告されている．しかし，従来の研究の多くは探索空間が 1 次元である．実世界問題では，多次元の探索空間についても考慮する必要がある．低次元問題において，学習が成功したとしても，同じ接近法で高次元問題においても，成功するとは限らない．ここで，本論文は，探索空間の次元の観点から，強化学習の報酬に関する解析を行うことを第三番目の研究目標とする．これにより，探索空間次元と報酬の関係を明らかにし，報酬設計の指標の一つとする．

1.3 論文の構成

本論文は，「適応的探索による発電プラントの起動スケジューリング」と題し，7章より構成される．第1章は序論であり，研究の背景，目的と意義を述べる．第2章は，問題の設定と接近法であり，本研究の対象としている問題を概説し，既存の研究サーベイ及び本研究の接近法についてまとめる．第3章は，問題の特徴分析であり，本問題の目的関数と制約関数の性質から本問題の最適解が境界上に存在することを示し，境界近傍探索を行うための強制操作を提案する．第4章はGAによる進化的探索であり，GAのコード化／交叉と世代交替モデルを論じた後，探索の効率性の向上を図るため，強制操作，再利用機能とタブ戦略を導入したGAによる探索を提案し，その有効性について実験による評価を行なう．第5章は，GAと強化学習の融合による適応的探索であり，探索の頑健性の向上を図るため，複数境界探索戦略を導入したGAによる探索，さらに適応的な探索を図るため，GAと強化学習の融合モデルを提案し，実験による評価を行なう．第6章は，強化学習の報酬戦略に関する解析であり，ここでは，探索空間の次元と報酬の関係を明らかにし，その結果を発電プラントの起動スケジューリング問題において実験による確認を行う．第7章は結論であり，本研究の成果を総括し，今後の課題をとりまとめる．

2 問題の設定と接近法

2.1 火力発電プラント起動スケジューリング問題

火力発電プラント起動スケジューリングとは，ボイラ点火から発電機定格負荷到達までの起動時間の最短化にあるが，プラント起動過程に発生するタービンロータ最大熱応力を規定値内に抑えることが必要である．この問題は組み合わせ最適化問題として定式化されるが，以下のような特徴を持ち，効率のよい探索手法の確立が要請される．

a) 制約条件である最大熱応力を計算するためのタービンシミュレーション計算時間がかかり，SPARC station 20上では，1回あたり約1 CPU秒である．
b) オンライン探索問題として解空間サイズが大きく，約 $\Pi_{i=1}^{10} s_i \fallingdotseq 5.8\times10^8$ である（表4.1(p.36)を参照）．
c) いくつかの局所的最適解(local optima)を持ち，伝統的な傾斜探索法では，大域的最適解(global optimum)の探索が困難である．
d) プラント起動中に，プラントの運転状態変化や運転員の要求に応じて，オンラインで短時間内に最適解を探索する必要がある．

図2.1は，大容量な火力発電プラントの基本構成，図2.2は，火力発電プラント起動スケジュールを示す．プラント起動において，図2.2に示されるように，ボイラー点火後，主蒸気温度が通気目標温度 X_{10}(℃)に到達した時点で，タービン通気が行われる．タービン通気とは，タービンに蒸気を通し始めることである．通気後，タービン速度は加速率 X_1(rpm^2)で定格速度まで上昇する．その間，低速保持時間 X_2(min.)と高速保持時間 X_3(min.)により，速度が保持される．タービンにより駆動される発電機が電力系統に同期併入された後，発電機負荷は，増負荷率 X_4(%/min.)で初負荷まで上昇し，初負荷保持間 X_5(min.)経過後，負荷が再び上昇し，増負荷率 X_6，X_7，X_8 と X_9(%/min.)で定格負荷まで上昇する．スケジューリング問題とは，これらのスケジュールパラメータ X_i, i=1,...,10 の値を決定し，ボイラー点火から発電機定格負荷到達までの起動時間 T(min.)を最短化すると同時に，起動過程中に発生する最大熱応力 σ_{mj} を制限熱応力 σ_{lj} 内に抑えることである．添字 j=1,...,4 は高圧タービンロータ表面(j=1)，中圧タービンロータ表面(j=2)，高圧タービンロータボア（bore：中心孔）(j=3)，中圧タービンロータボア(j=4)を表わしている．

与えられる起動スケジュールパラメータ X_i, i=1,...,10 に対して，起動過程の時点 t, $0 \leqq t \leqq T$ の発生熱応力を σ_j, j=1,...,4 とすると，σ_j は次式により与えられる．

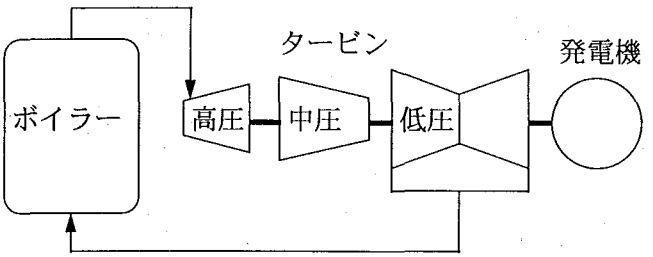


図 2.1: 発電プラントの基本構成

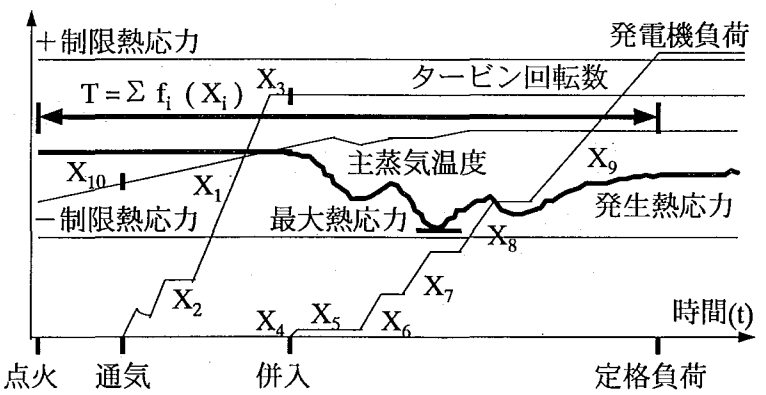


図 2.2: 火力発電プラント起動スケジュール

$$\sigma_j(X_1,...,X_i,...,X_{10},t)=g_j(X_1,...,X_i,...,X_{10}),\ j=1,...,4,\ 0\leqq t\leqq T \tag{2.1}$$

ここで，g_j はタービン起動過程をシミュレートするタービンダイナミックモデルに対応し，起動スケジュール$(X_1,...,X_i,...,X_{10})$と他の起動条件 τ_h を入力とし，発生熱応力 σ_j を出力とする関数である．起動条件 τ_h とは，プラント起動時のタービンロータ温度などによって構成される．本来発生熱応力 σ_j はこのような起動条件 τ_h によって変化するが，本最適化問題は τ_h が与えられる上，$(X_1,...,X_i,...,X_{10})$の値を決定するので，式(2.1)では τ_h の明記を省略した．発生熱応力 σ_j は圧縮または引張により負または正の値をとるが，その絶対値の最大値を制限する必要があるので，最大熱応力 σ_{mj} は次式により定義される．

$$\sigma_{mj}(X_1,...,X_i,...,X_{10})=\max_{0\leqq t\leqq T}\left|\sigma_j(X_1,...X_i,...,X_{10},t)\right|,\ j=1,...,4 \tag{2.2}$$

以上より，本最適化問題は下記のように定式化できる．

$$\min_{X_i,i=1,...,10} \quad T(X_1,...,X_i,...,X_{10}) \tag{2.3a}$$

$$\text{subject to} \quad \forall(i)\ X_i\in\{X_i^k \mid k=1,...,s_i\},\ i=1,...,10 \tag{2.3b}$$

$$\forall(j)\ \ \sigma_{mj}(X_1,...,X_i,...,X_{10})\leqq\sigma_{lj},\ j=1,...,4 \tag{2.3c}$$

ここで，式(2.3a)は目的関数，式(2.3b)は起動スケジュールパラメータ制約条件，式(2.3c)は最大熱応力制約条件を表わす．起動スケジュールパラメータ X_i, i=1,...,10 の制約集合$\{X_i^k \mid k=1,...,s_i\}$は表 4.1(p.36)に与えられ，$s_i$ は各起動スケジュールパラメータのサイズを表わす．表 4.1(p.36)により，例えば，$s_1=3$，$s_{10}=78$ である．式(2.3b)と(2.3c)は実行可能解の条件を表わす．以下の説明では，$(X_1,...,X_i,...,X_{10})$を起動スケジュールまたは解，X_i, i=1,...,10 を起動スケジュールパラメータまたは変数と呼び，起動スケジュール（解）は起動スケジュールパラメータ（変数）によって構成される．

式(2.3)はペナルティ法により次のような制約条件なしの問題に変換することができる．

$$\min_{X_i,i=1,...,10} \quad T'(X_1,...,X_i,...,X_{10})=T(X_1,...,X_i,...,X_{10})+[K_{Pj}]^{tr}\left(\left[\sigma_{mj}(X_1,...,X_i,...,X_{10})\right]-\left[\sigma_{lj}\right]\right) \tag{2.4}$$

where $K_{Pj}=0$, if $\sigma_{mj}\leqq\sigma_{lj}$

$K_{Pj}>0$, otherwise

K_{Pj}, j=1,...,4 はペナルティ定数，$[K_{Pj}]^{tr}$ は$[K_{Pj}]$の転置ベクトル，記号[]はベクトルを表わす．

2.2 既存の研究

本問題は古くから研究と解析が行われてきた．従来，起動スケジュールはタービン設計段階で専門家によってワーキングカーブとして設定され[Hanzalek 66], 自動化システムの一部として発電プラント制御用計算機に予め記憶されてきた[鈴木 80].

プラント起動過程における制御量と発生熱応力の入出力応答特性が線形でかつ時間的に不変(time invariant)であると仮定すると，最適起動スケジュールのアルゴリズム[Bednarski 73]や最適制御出力パターン[Domachowski 86]が解析用として提案された．実問題において，この入出力応答特性は非線形性が強く，しかも運転状態によって変化するので，提案されたアルゴリズムや制御出力パターンを実プラントの運転への適用が困難である．オンラインで周期的な最適制御を行うタービンロータ熱応力予測制御システムが開発された[Matsumoto 82]が，[Matsumoto 93]で指摘したように，この熱応力制御システムにより，起動完了時刻（起動スケジュール）を精度よく求めることが困難である．また，プラント運用中の起動スケジュールの最適化を目的に，専門家の知識を応用したファジィエキスパートシステムが提案された[Matsumoto 93]．知識システムを実問題に適用した場合，知識収集コストを考慮する必要がある．

以上の提案は何れも決定的な手法を用いているが，非決定的な手法を用いる研究として，モンテカルロ法による発電機群の起動停止計画の最適化が提案された[陳 89]．[陳 89]は，最小起動停止時間と電力需給バランスを満足する実行可能解を効率的に生成し、モンテカルロ法を用いて，十分な回数で一様な確率分布による試行点の抽出により，近似最適解を得る手法を提案した．[陳 89]の問題対象が本研究と異なるが，試行点の一様な抽出では，効率のよい最適解または近似最適解の探索が期待できない．

GA や強化学習を本問題に適用するのは，本研究が初めてである．GA を最適化問題に適用した場合，GA の設計要点は，コード化，交叉，突然変異と世代交替モデルであるが，具体的な設計は，対象問題に依存している．コード化に関しては，伝統的なバイナリコード GA はスキーマ理論[Holland 75]によって支持されるが，実世界の工学問題において，実数ないし整数コード GA([Eshelman 93], [Goldberg 91])がよい性能が示されている．また，世代交替モデルに関しては，エリート保存戦略[Goldberg 89a]の導入が多く行われている．

強化学習は，関数最適化や制御学習問題など幅広く応用できる[Williams 92]が，今まで提案された強化学習のほとんどは，ロボットのドッキング問題［Lin 93]など制御学習問題に関するものが多い．また，強化学習が真に有望な手法であると認知されるためには，いわゆる Toy Problem ではない実際的な問題に適用されることが望まれる［宮崎 96］.

強化学習の報酬方法として，正の報酬と負の報酬がある．[Barto 83]は，ポールバランス問題(pole-balancing problem)において，負の報酬だけを用いた学習を行う．ポールが倒れた時や，カートポールシステム(cart-pole system)がトラックの境界に衝突した時に，負の報酬“−1”が与えられる．[Lin 93]は，ロボットのドッキング問題において，正と負の報酬を用いた．衝突が発生した時，負の報酬“−10”，ドッキングに成功した時，正の報酬“+100”，その他の時，報酬“0”が与えられる．[Tan 91]は，グリッドワールド問題において，正の報酬だけを用いた学習を行った．ロボットがゴールに達した時に，正の報酬“+1”が与えられる．これらの問題において，強化学習の出力次元は何れも 1 である．ドメインサイズ，すなわち強化学習が取り得る行動の数は，[Barto 83]では 2，[Lin 93]では 6，そして[Tan 91]では 4 である．

しかし，強化学習を実問題に応用した場合，多次元出力の強化学習を考慮する必要がある．強化学習は，1 次元の問題に対する学習に成功したとしても，同様な接近法で多次元問題においても成功するとは限らないので，強化学習の報酬について理論的な解析が望まれる．

2.3 本研究の接近法

火力発電プラント起動スケジューリングは，制約付き最適化問題であるが，発生最大熱応力の制限が厳しければ，起動時間を長くする必要があり，制限が緩いなら，起動時間を短くすることが可能であるという経験的な観察により，本問題の最適解が実行可能解空間の境界またはその近傍に存在することが推定される[Kamiya 95]．本論文は，最適解が実行可能解空間の境界上に存在することを示す．最適解が実行可能解空間の境界上に存在することが保証されれば，実行可能解の全空間を探索するよりも，境界近傍に限定することにより，最適解を見逃すことなく効率的な最適解探索が期待できる．そこで，探索を境界近傍に限定する強制操作を提案する．

強制操作は探索を境界近傍に限定するだけであり，最適解の探索を行っているのではない．最適解の探索を行うため，強制操作と GA の組合わせによる境界近傍探索モデルを提案する．本問題はいくつかの局所的最適解を持っているので，確率探索法である GA の導入は，大域的最適解またはその近似最適解を探索することを目的としている．境界近傍探索のため，強制操作と GA の組合わせによる探索モデルに対して，以下のような探索戦略を導入する．

a) **GA の交叉モデル**　境界近傍探索のため近傍探索モデルが有効であるので, GA の交叉モデルは，探索ステップαを小さくする設定することにより，親の近傍に子供を生成することができる NDX－α(Normal Distribution Crossover－α)［Ono 96］を用いる．

b) **再利用機能とタブ戦略**　近傍探索では，同じ解を繰り返し探索することがしばしば生じる．再利用機能とは，探索した解のシミュレーション結果を記憶し，同じ解の評価のためのシミュレーション計算を省くことにより，探索時間の短縮を図るものである．タブ戦略[Glover 93]とは，最近探索した解への移動(move)を禁止することにより，探索効率の向上や局所的最適解からの早期脱出を図るものである．

c) **複数の境界に対する探索**　制約条件最適化問題において，実行可能解空間の外側に回りこんで探索することが効率よく最適解を探索できることが報告されている([Smith 93], [Glover 93])．本論文では境界の外側の探索を積極的に行うため，境界の外側に緩和境界を設けて，複数の境界に対する探索戦略を導入する．

しかし，以上のような探索を基本としたモデルでは，プラント運転に要求されるオンライン探索性能を充分満足することができない．ここで，要求されるオンライン探索性能を満たすため，本研究は学習と探索能力を持ち合わせている強化学習を導入し，学習効果による探索効率の向上を図る．本問題の探索条件である最大熱応力σ_{mj}やその他のプラント起動条件τ_hが連続値であるため, 強化学習モデルとしては汎化能力のあるニューラルネットワークを用いた確率傾斜法［木村 96］により実現される．本問題は非線形性が強いため，本研究では強化学習モデルに用いられるニューラルネットワークに対して中間層の導入を行い，非線形の学習モデルの構築を図る．強化学習と GA の融合は，

a) GA は強化学習を有望な領域で学習するようにガイドすることにより, 学習を加速する．
b) 強化学習は学習の加速効果により，探索の序盤より有望な解を生成し，それを GA の初期解として与えることにより，GA の探索を加速する．

というような相乗効果により，学習と探索効率の向上を図る．また，強化学習は探索条件に対応する最適解を予め学習し，プラント起動時では，学習効果により，任意に与えられる探索条件に対応する有望な解候補を生成し，最適解の探索を加速することが期待できる．

強化学習は，報酬を手掛かりに，与えられる入力に対する最適出力を学習することを目的としているので．報酬戦略の選択により，学習性能が大きく左右される．ここでは，次のような三つの報酬戦略を定義し，各報酬戦略に関する強化学習の学習性能に関する解析を行う．

a) **正の報酬戦略**　与えられる入力に対して強化学習が最適出力を行った時だけ，正の報酬，それ以外の時，報酬が与えられない．

b) **負の報酬戦略**　与えられる入力に対して強化学習が最適出力を行わなかった時だけ，負の報酬，それ以外の時，報酬が与えられない．

c) **正+負の報酬戦略**　与えられる入力に対して強化学習が最適出力を行なった時，正の報酬，最適出力を行わなかった時，負の報酬が与えられる．

3 問題の特徴分析

この章では，発電プラント起動スケジューリング問題の特徴を分析し，最適解が実行可能解空間の境界上に存在することを示した上，探索効率向上を図るため，探索を境界近傍に限定する強制操作を提案する．

3.1 目的関数と制約条件の性質

本問題は式(2.3)により定式化されるが，その特徴をまとめると，下記のとおりである．

a) 目的関数 $T(X_1,...,X_i,...,X_{10})$は強意単調(strongly monotone)である．

b) 目的関数 $T(X_1,...,X_i,...,X_{10})$と制約関数 $\sigma_{mj}(X_1,...,X_i,...,X_{10})$は連続であるが，決定変数 X_i, i=1,...,10 は運用上では表 4.1(p.36)に示されるように離散値 X_i^k, i=1,...,10, k=1,...,s_i をとる．

以下はこれらの特徴について説明する．

3.1.1 目的関数の強意単調性

加速率や増負荷率 X_i, i=1,4,6,7,8,9 が大きければ，起動時間が短くなるので，目的関数 $T(X_1,...,X_i,...,X_{10})$は加速率や増負荷率 X_i, i=1,4,6,7,8,9 に対して強意単調減少である．速度保持時間または負荷保持時間 X_i, i=2,3,5 が長ければ，起動時間が長くなるので，目的関数 $T(X_1,...,X_i,...,X_{10})$はこれらの保持時間 X_i, i=2,3,5 に対しては強意単調増加である．通気目標主蒸気温度 X_{10} が高ければ，ボイラ点火からタービン通気までの主蒸気温度上昇時間が長くなるので，目的関数 $T(X_1,...,X_i,...,X_{10})$は通気目標主蒸気温度に対しては強意単調増加である．以上より，目的関数 $T(X_1,...,X_i,...,X_{10})$は X_i, i=1,...,10 に対して強意単調である．

3.1.2 制約関数の連続性

発生熱応力 $\sigma_j(X_1,...,X_i,...,X_{10},t)$は，図 2.2 に示されるように時間 t に対して多峰関数であり，正または負の値を取るが，熱応力の発生現象は連続であり，すなわち発生熱応力 $\sigma_j(X_1,...,X_i,...,X_{10},t)$が連続関数である．この場合，制約関数である最大熱応力 $\sigma_{mj}(X_1,...,X_i,...,X_{10})$も連続関数であることを以下の定理により示す．

定理 3.1（制約関数の連続性）

任意な関数$\sigma_j(X_1,...,X_i,...,X_{10},t)$が連続なら，$\sigma_{mj}(X_1,...,X_i,...,X_{10})$も連続である．ただし，$\sigma_{mj}(X_1,...,X_i,...,X_{10})$は式(2.2)により与えられる．

証明：

$\sigma_j(X_1,...,X_i,...,X_{10},t)$が連続の場合，$\forall \varepsilon > 0$ に対して，$|X_i' - X_i| < \delta$ なら，常に

$$|\sigma_j(X_1,...,X_i',...,X_{10},t) - \sigma_j(X_1,...,X_i,...,X_{10},t)| < \varepsilon$$

となるような $\delta > 0$ が存在する．

$$\sigma_j(X_1,...,X_i',...,X_{10},t) - \sigma_j(X_1,...,X_i,...,X_{10},t)$$
$$\leqq |\sigma_j(X_1,...,X_i',...,X_{10},t) - \sigma_j(X_1,...,X_i,...,X_{10},t)| < \varepsilon$$

であるので，

$$\sigma_j(X_1,...,X_i',...,X_{10},t) < \sigma_j(X_1,...,X_i,...,X_{10},t) + \varepsilon$$

となる．従って，式(2.2)により以下の式が成立する．

$$|\sigma_{mj}(X_1,...,X_i',...,X_{10}) - \sigma_{mj}(X_1,...,X_i,...,X_{10})|$$
$$= \left| \max_{0 \leqq t \leqq T} |\sigma_j(X_1,...,X_i',...,X_{10},t)| - \max_{0 \leqq t \leqq T} |\sigma_j(X_1,...,X_i,...,X_{10},t)| \right|$$
$$< \left| \max_{0 \leqq t \leqq T} |\sigma_j(X_1,...,X_i,...,X_{10},t)| + \varepsilon - \max_{0 \leqq t \leqq T} |\sigma_j(X_1,...,X_i,...,X_{10},t)| \right| = \varepsilon$$

（証明終わり）

3.2 最適解の存在領域

最適解が境界上に存在することを示すため，ここでは，境界について定義する．

定義 3.1a（起動スケジュールパラメータが連続である場合の境界）

境界とは，下記の式(3.1a)または(3.1b)，及び制約条件式(2.3b)と(2.3c)を満足する実行可能解$(X_1,...,X_i,...,X_{10})$によって定義される．ただし，式(2.3b)において，X_iは連続であるとする．

$$(\exists(j)\ \sigma_{mj}(X_1,...,X_i,...,X_{10}) = \sigma_{lj},\ j=1,...,4) \text{ or} \tag{3.1a}$$
$$(\forall(i)\ (X_i = X_i^1 \text{ or } X_i = X_i^{si}),\ i=1,...,10) \tag{3.1b}$$

以下の定理が成立する.

定理 3.2（起動スケジュールパラメータが連続である場合の最適解条件）
実行可能解が存在し，目的関数と制約関数が連続で，目的関数が強意単調なら，式(2.3)に対する最適解は定義 3.1a の境界上に存在し，それ以外には存在しない.

証明：

最適解が定義 3.1a により定義される境界外にも存在，すなわち制約条件式(2.3b)と(2.3c)及び

$$(\forall(j)\ \sigma_{mj}(X_1',...,X_i',...,X_{10}') < \sigma_{lj},\ j=1,...,4) \text{ and } (\exists(i)\ X_i^1 < X_i' < X_i^{si},\ i=1,...,10)$$

を満足する最適解$(X_1',...,X_i',...,X_{10}')$が存在すると仮定する.

ここで，目的関数 $T(X_1,...,X_i,...,X_{10})$は X_i に対して強意単調減少とし，解$(X_1',...,X_i^{si},...,X_{10}')$について考える.

すべての j に対して$\sigma_{mj}(X_1',...,X_i^{si},...,X_{10}') \leqq \sigma_{lj}$である場合，$(X_1',...,X_i^{si},...,X_{10}')$は実行可能解である．$T(X_1,...,X_i,...,X_{10})$は X_i に対して強意単調減少であり，仮定により $X_i' < X_i^{si}$ であるため，$T(X_1',...,X_i^{si},...,X_{10}') < T(X_1',...,X_i',...,X_{10}')$となり，$(X_1',...,X_i',...,X_{10}')$が最適解ではない.

$\sigma_{mj}(X_1',...,X_i^{si},...,X_{10}') > \sigma_{lj}$であるような j が存在する場合，このような j を j^+とする．$\sigma_{mj^+}(X_1,...,X_i,...,X_{10})$は $X_i^1 \leqq X_i \leqq X_i^{si}$ において連続であるので，中間値の定理により，$\sigma_{mj^+}(X_1',...,X_i'+\delta_{j+},...,X_{10}') = \sigma_{lj^+}$かつ $X_1' < X_1' + \delta_{j^+} < X_i^{si}$ を満足する $\delta_{j^+} > 0$ が存在する．ここで，存在するすべての δ_{j^+}の中で，もっとも小さい δ_{j^+}を δ_mとすると，すべての j^+に対して δ_m は $\sigma_{mj^+}(X_1',...,X_i'+\delta_m,...,X_{10}') \leqq \sigma_{lj^+}$を満足し，$\sigma_{mj^+}(X'_1,...,X_i'+\delta_m,...,X'_{10}) = \sigma_{lj^+}$を満足する j^+が存在する.

j^+以外の j を j^-とすると，$\sigma_{mj^-}(X_1',...,X_i'+\delta_m,...,X_{10}') > \sigma_{lj^-}$であるような j^-が存在しない場合，$(X_1',...,X_i'+\delta_m,...,X_{10}')$は実行可能解である．$T(X_1,...,X_i,...,X_{10})$は X_i に対して強意単調減少であるので，$T(X_1',...,X_i'+\delta_m,...,X_{10}') < T(X_1',...,X_i',...,X_{10}')$となり，$(X_1',...,X_i',...,X_{10}')$が最適解ではない.

$\sigma_{mj^-}(X_1',...,X_i'+\delta_m,...,X_{10}') > \sigma_{lj^-}$であるような j^-が存在する場合，再び，中間値の定理により，すべての j^-に対して $\sigma_{mj^-}(X_1',...,X_i'+\delta_{m'},...,X_{10}') \leqq \sigma_{lj^-}$を満足する $\delta_{m'} > 0$ が存在し，$\sigma_{mj^-}(X_1',...,X_i'+\delta_{m'},...,X_{10}') = \sigma_{lj^-}$を満足する j^-が存在する．$(X_1',...,X_i'+\delta_{m'},...,X_{10}')$は実行可能解である．$T(X_1,...,X_i,...,X_{10})$は X_i に対して強意単調減少であるので，$T(X_1',...,X_i'+\delta_{m'},...,X_{10}') < T(X_1',...,X_i',...,X_{10}')$となり，$(X_1',...,X_i',...,X_{10}')$が最適解ではない.

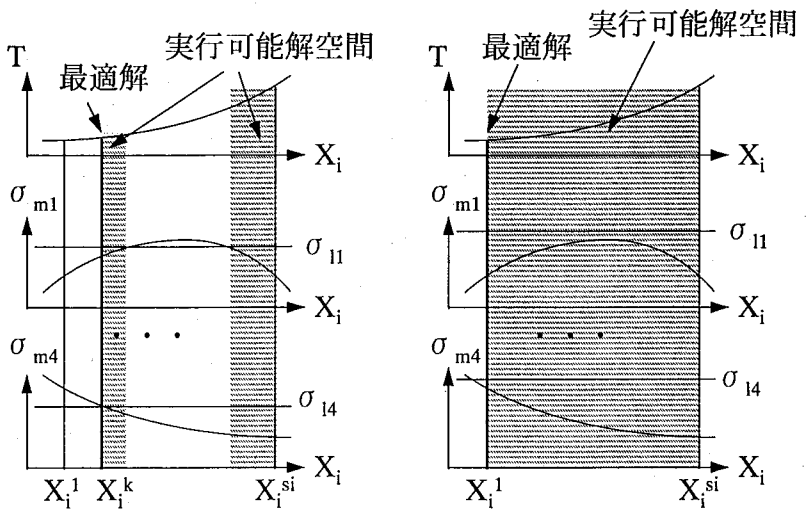
$T(X_1,...,X_i,...,X_{10})$は$X_i$に対して強意単調増加であっても，解$(X_1',...,X_i^1,...,X_{10}')$, $X_i^1<X_i'$について考えることにより，$(X_1',...,X_i',...,X_{10}')$が最適解ではないことが容易に証明できる．

以上より，$(X_1',...,X_i',...,X_{10}')$が最適解という仮定が正しくない．すなわち最適解は定義 3.1a により定義される境界以外では存在しないことが言える．定理の前提条件により実行可能解が存在するので，最適解の存在が保証される．最適解は境界以外に存在しないので，境界上のみ存在することになる．

（証明終わり）

ここで，注意されたいのは制約関数$\sigma_{mj}(X_1,...,X_i,...,X_{10})$が非単調であっても多峰であっても，最適解は境界に存在することである．本問題の制約関数の非単調性は表 3.1(p.20)によって示される．同表により差分関係$\frac{\Delta\sigma_{mj}}{\Delta X_i}$は，起動スケジュール$(X_1,...,X_i,...,X_{10})$または起動スケジュールパラメータの差分$\Delta X_i$の大きさによって正または負の値をとる．

最適解が境界上に存在する例として，図3.1 を示す．同図において，縦軸T及びσ_{mj}, j=1,...,4 はそれぞれ本問題の目的関数と制約関数，横軸は決定変数X_iを表わす．図 3.1 の例では，目的関数はX_iに対して強意単調増加である．図 3.1(a)では，最適解により与えれる 4 番目の最大熱応力σ_{m4}は制限熱応力σ_{j4}に等しい例である．また，制限熱応力σ_{lj}, j=1,...,4 を図3.1(b)のように充分緩めて設定すれば，最適解により与えられる起動スケジュールパラメー



(a) $\exists(j)\ \sigma_{mj}(X_1,...,X_i,...,X_{10})=\sigma_{lj}$　　(b) $\forall(i)\ X_i=X_i^1$ or $X_i=X_i^{si}$

図 3.1: 実行可能解空間の境界上に最適解の存在例

タ X_iは，強意単調増加なら下限値 X_i^1，強意単調減少なら上限値 X_i^{si} をとる．

系 3.1（起動スケジュールパラメータが連続である場合の最適解条件）

実行可能解が存在し，目的関数と制約関数が連続で，目的関数が単調なら，式(2.3)に対する最適解は定義 3.1a の境界上に存在する．

目的関数が単調なら，境界以外でも最適解が存在することがあるが，境界には必ず最適解が存在することに注意されたい．

目的関数 $T(X_1,...,X_i,...,X_{10})$は必ずしも強意単調増加ではないことについて説明する．発電機併入時の主蒸気温度は一定以上とする必要がある場合について考える．ここで，通気目標主蒸気温度 X_{10} 及び与えられる通気から併入時の主蒸気温度上昇曲線により決定される通気から併入までに必要な温度上昇時間を t_p, タービンの加速率 X_1 及び回転数保持時間 X_2 と X_3 により決定される通気から併入までに必要なタービン速度上昇と保持時間を t_s，必要とされる通気から併入までの時間を t_{sp} とすると，$t_{sp}=\max(t_s, t_p)$となる．$t_s<t_p$ の場合，加速率 X_1，回転数保持時間 X_2 と X_3 を起動時間の増加方向に変化させても，すなわち t_s を増大する方向に変化させても，$t_s=t_p$ なるまで，t_{sp} は変化がないので，目的関数 $T(X_1,...,X_i,...,X_{10})$ の値が変化しない．同様に，$t_p<t_s$ の場合，通気目標主蒸気温度 X_{10} を起動時間の増加方向に変化させても，$t_p=t_s$ なるまで，目的関数 $T(X_1,...,X_i,...,X_{10})$の値が変化しない．従って，$T(X_1,...,X_i,...,X_{10})$は X_1, X_2 と X_3 に対して強意単調増加ではなく，単調増加である．本研究の実験に使われる対象発電プラントモデルの目的関数は強意単調であるが，目的関数が単調であるような発電プラントもあるので，以上のような系を示した．

以上は，目的関数と制約関数が連続である場合の境界条件に関する最適解の存在について示したが，本問題の決定変数は，式 2.3(b)に示されるように離散値をとる．連続値の場合の定義 3.1a と対応して次のような離散値の場合の境界を定義する．

定義 3.1b（起動スケジュールパラメータが離散である場合の境界）

境界とは，下記の式(3.1c)と(3.1d)，及び制約条件式(2.3b)と(2.3c)を満足する実行可能解 $(X_1,...,X_i^k,...,X_{10})$によって定義される．

$$(\exists(j)\,(\sigma_{mj}(X_1,...,X_i^{k-1},...,X_{10})>\sigma_{lj},\ k=2,...,s_i) \text{ or } (\sigma_{mj}(X_1,...,X_i^{k+1},...,X_{10})>\sigma_{lj},\ k=1,...,s_i-1),\quad j=1,...,4) \qquad (3.1c)$$

or

$$(\forall(i)\,(X_i=X_i^1, \text{ or } X_i=X_i^{si}),\ i=1,...,10) \qquad (3.1d)$$

なお，連続の場合に対応して，離散の場合の最適解の境界上の存在に関する定理を以下に示す．

定理 3.3（起動スケジュールパラメータが離散である場合の最適解条件）

実行可能解が存在し，目的関数が強意単調なら，式(2.3)に対する最適解は定義 3.1b の境界上に存在し，それ以外の場所には存在しない．

証明：

目的関数 T は X_i に対して強意単調減少とする．ここで，最適解は定義 3.1b の境界以外の場所に存在，すなわち

$$(\forall(j)\,(\sigma_{mj}(X_1,...,X_i^{k'-1},...,X_{10}) \leqq \sigma_{lj} \text{ and } \sigma_{mj}(X_1,...,X_i^{k'+1},...,X_{10}) \leqq \sigma_{lj}),\ j=1,...,4) \text{ and}$$
$$(\exists(i)\ X_i=X_i^{k'},\ k'=2,...,s_i-1,\ i=1,...,10) \quad (3.2)$$

を満足するような最適解$(X_1,...,X_i^{k'},...,X_{10})$が存在すると仮定する．

式(3.2)より，$(X_1,...,X_i^{k'+1},...,X_{10})$は実行可能解である．$T(X_1,...,X_i,...,X_{10})$は X_i に対して強意単調減少であるとすると，$X_i^{k'}<X_i^{k'+1}$ であるので，$T(X_1,...,X_i^{k'},...,X_{10})>T(X_1,...,X_i^{k'+1},...,X_{10})$ となる．$(X_1,...,X_i^{k'},...,X_{10})$は最適解ではない．

また，目的関数 T は X_i に対して強意単調増加であっても，解$(X_1,...,X_i^{k'-1},...,X_{10})$について考えることにより，以上と同様に，$(X_1,...,X_i^{k'},...,X_{10})$が最適解でないことが容易に証明できる．

以上より，$(X_1,...,X_i^{k'},...,X_{10})$が最適解であるという仮定が正しくない．すなわち最適解は定義 3.1b により定義される境界以外では存在しないことが言える．定理の前提条件により，実行可能解が存在するので，最適解の存在が保証される．最適解が境界以外に存在しないので，最適解が境界上のみ存在することが証明された．

（証明終わり）

なお，連続の場合と同様に，目的関数が単調である場合に対応する系を以下に示す．

系 3.2（起動スケジュールパラメータが離散である場合の最適解条件）

実行可能解が存在し，目的関数が単調なら，式(2.3)に対する最適解は定義 3.1b の境界上に存在する．

3.3 境界領域近傍への強制

3.3.1 強制操作の概要

3.2 節において，本問題の最適解が境界上に存在することを示した．最適解は境界上に存在するので，境界に沿った探索は最適解を見逃すことなく効率的である．ここで，本研究はこのような考え方により，探索を境界近傍に限定する強制操作を提案する．強制操作は図 3.2 に示されるように，シミュレーション計算と線形近似計算という二つの計算を繰り返すことによって，与えられた解$(X_1,...,X_i,...,X_{10})$を漸近的に境界に移動させる．

図 3.2 において，シミュレーション計算では，与えられた解$(X_1,...,X_i,...,X_{10})$から，ダイナミックモデルを使って，四つの最大熱応力σ_{mj}を計算する（式(2.1)と(2.2)）．線形近似計算では，与えられた解$(X_1,...,X_i,...,X_{10})$が非実行可能解の場合と実行可能解の場合に分けて，線形近似係数a_{ij}を使って，下記の式(3.3a)と(3.3b)によりX_i^*を計算する．

$$X_i^* = \min_{X_i \mid \forall(j^-)\sigma_{mj^-} \le \sigma_{lj^-} \text{ かつ } \forall(j^+)\sigma_{mj^+} \text{ を増加させない}} \left| \sigma_{mj^+} - \sigma_{lj^+} \right|, \text{ if } \exists(j)\ \sigma_{mj} > \sigma_{lj} \tag{3.3a}$$

$$X_i^* = \min_{X_i \mid \forall(j)\sigma_{mj} \le \sigma_{lj}} T, \text{ if } \forall(j)\ \sigma_{mj} \leqq \sigma_{lj},\ j=1,...,4 \tag{3.3b}$$

$$X_i \leftarrow \underset{i=1,...,10}{\text{random}}\left(X_i^* \mid X_i^* \neq X_i \right) \tag{3.3c}$$

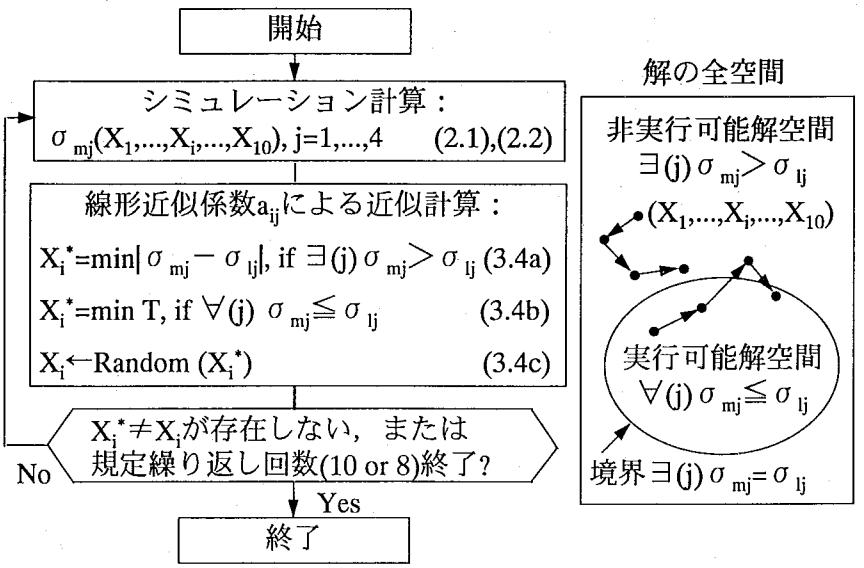


図 3.2: 強制操作アルゴリズム

ここで，$\sigma_{mj}(X_1,...,X_i,...,X_{10}) > \sigma_{lj}$ であるような j を j^+，$\sigma_{mj}(X_1,...,X_i,...,X_{10}) \leqq \sigma_{lj}$ であるような j を j^- としている．与えられた解$(X_1,...,X_i,...,X_{10})$が非実行可能解$(\exists(j)\ \sigma_{mj}(X_1,...,X_i,...,X_{10}) > \sigma_{lj})$なら，線形近似計算は式(3.3a)を使って，$X_i^*$を計算する．式(3.3a)は最大熱応力$\sigma_{mj}^+$が制限熱応力$\sigma_{lj}^+$との距離$|\sigma_{mj}^+ - \sigma_{lj}^+|$を最小化し，かつ最大熱応力$\sigma_{mj}^-$が制限熱応力$\sigma_{lj}^-$を超えないように，及び最大熱応力$\sigma_{mj}^+$が増加しないように，起動スケジュールパラメータ X_i^*を計算する．与えられた解$(X_1,...,X_i,...,X_{10})$が実行可能解$(\forall(j)\ \sigma_{mj}(X_1,...,X_i,...,X_{10}) \leqq \sigma_{lj})$なら，線形近似計算は式(3.3b)を使って，$X_i^*$を計算する．式(3.3b)は起動時間 T が最短となるように，かつ，任意な最大熱応力σ_{mj}が制限熱応力σ_{lj}を超えないように，起動スケジュールパラメータ X_i^*を計算する．このような計算は 10 の起動スケジュールパラメータ X_i, i=1,...,10 に対してそれぞれ独立に行う．

式(3.3c)において，式(3.3a)または(3.3b)を満足する X_i^*の中から，与えられる X_i に等しくない X_i^*をランダムに一つ選んで，与えられる X_i を更新する．記号←は代入を表わす．このような X_i^*が存在しない時，または，規定繰り返し回数を超えた時，強制操作が終了する．図 3.2 において，強制操作に最適解の探索機能を持たせるため，強制操作開始から終了までの操作を一強制操作過程と呼ぶと，一強制操作過程の中の最良解，すなわち目的関数値（式(2.4)）のもっとも低い解を強制操作を行った結果の解（これを強制最良解と呼ぶ）とする．

ここで，注意したいのは，強制操作は，起動スケジュールを境界近傍に移動させるだけであって，最適解を探索するため，最適解探索モデルと組み合わせる必要がある．なお，起動スケジュール$(X_1,...,X_i,...,X_{10})$が境界から離れれば離れるほど，境界に移動するための強制操作の繰り返し回数が多く必要となり，強制操作を繰り返すたびに，ダイナミックモデルによるシミュレーションを伴うので，探索性能が低下する．強制操作と組み合わせた場合，最適解探索モデルは，起動スケジュール$(X_1,...,X_i,...,X_{10})$を遠くに生成しないような近傍探索が望まれる．

以下は，強制操作に使用される線形近似係数 a_{ij} について説明する．線形近似係数 a_{ij}, i=1,...,10, j=1,...,4 は最大熱応力σ_{mj}, j=1,...,4 と起動スケジュールパラメータ X_i, i=1,...,10 の差分関係を表わし，式(3.4)により与えられる．

$$a_{ij} = \frac{\Delta\sigma_{mj}}{\Delta X_i} = \frac{\sigma_{mj}(X_1,...,X_i + \Delta X_i,...,X_{10}) - \sigma_{mj}(X_1,...,X_i,...,X_{10})}{(X_i + \Delta X_i) - (X_i)} \tag{3.4}$$

ここで，X_i は等価起動時間に換算された値である．本問題は非線形であるため，各線形近似係数 a_{ij}, i=1,...,10, j=1,...,4 は，それぞれが起動スケジュール$(X_1,...,X_i,...,X_{10})$と起動スケジ

ュールパラメータの差分Δ X_i に依存する．表 3.1 は，最適解探索過程において，ランダムに生成された起動スケジュール(X_1,...,X_i,...,X_{10})に対して，ランダムにある一つの起動スケジュールパラメータ X_i を選択し，変化させることによって得られた高圧タービンロータ表面熱応力(j=1)，中圧タービンロータ熱応力(j=2)と各起動スケジュールパラメータ X_i, i=1,...,10 に対応する線形近似係数 a_{ij}, i=1,...,10, j=1,2 の統計データである．本研究は，いくつかの代表的な制限熱応力 σ_{lj} について，最適解探索予備実験を行い，それぞれの代表的な制限熱応力 σ_{lj} に対して，表 3.1 のような線形近似係数 a_{ij} の統計データを予め求める．強制操作はこのように求められた統計データの平均値，すなわち表 3.1 の"ave"で表わされる値を用いる．強制操作に用いられる n 個の代表的な制限熱応力[σ_{lj}]をそれぞれ set 1,set 2,...,set n とし，対応する n 個の線形近似係数をそれぞれ[a_{ij}](set 1),[a_{ij}](set 2),...,[a_{ij}](set n)とすると，与えられた任意な制限熱応力 set x に対する最適解探索を行なう場合，強制操作は，代表的な制限熱応力 set 1,set 2,...,set n と対応する線形近似係数[a_{ij}](set 1),[a_{ij}](set 2),...,[a_{ij}](set n)から，内挿法により，任意に与えられる制限熱応力 set x に対する線形近似係数[a_{ij}](set x)を算出する．強制操作は，このような大まかな線形近似係数 a_{ij} を使うが，探索過程において生成された解を確率的に境界近傍に移動できれば，探索効率の向上が期待できると考えられる．

表 3.1: 最大熱応力 σ_{mj}(kg/mm^2)と起動スケジュールパラメータ X_i(min.)の差分関係

$$a_{ij}=\frac{\Delta\sigma_{mj}}{\Delta X_i},\ i=1,...,10,\ j=1,2\ (kg/mm^2/min.)$$

ただし，X_i は等価起動時間(min.)に換算された値である．

ｽｹｼﾞｭｰﾙﾊﾟﾗﾒｰﾀ X_i		X_1	X_2	X_3	X_4	X_5	X_6	X_7	X_8	X_9	X_{10}
サンプリング回数		173	596	569	601	221	64	93	194	128	559
高圧	**ave**	**-0.0125**	**-0.0252**	**-0.0037**	**-0.0246**	**0.0004**	**0.0037**	**-0.0928**	**-0.0117**	**-0.0003**	**-0.0277**
タービン	var	0.0235	0.0194	0.0198	0.0222	0.0226	0.0490	0.0801	0.0287	0.0020	0.0127
ロータ	max	0.0670	0.0503	0.0937	0.0906	0.0942	0.0682	0.0000	0.0000	0.0000	0.0430
表面 σ_{m1}	min	-0.0482	-0.0658	-0.0616	-0.0990	-0.0507	-0.1182	-0.2009	-0.0980	-0.0228	-0.0506
中圧	**ave**	**0.0068**	**0.0035**	**0.0057**	**0.0099**	**0.0056**	**-0.0204**	**-0.1057**	**-0.2588**	**-0.0808**	**0.0023**
タービン	var	0.0007	0.0008	0.0008	0.0017	0.0009	0.0031	0.0046	0.0056	0.0049	0.0004
ロータ	max	0.0091	0.0053	0.0078	0.0142	0.0079	-0.0171	-0.1005	-0.2456	-0.0682	0.0032
表面 σ_{m2}	min	0.0057	-0.0030	0.0034	-0.0098	0.0035	-0.0303	-0.1206	-0.2871	-0.0884	-0.0022

3.3.2 強制操作の詳細

図 3.2 の強制操作の詳細は図 3.3 に示す．図 3.3 において，点線で囲まれた部分は線形近似係数 a_{ij} を用いた式(3.3)の具体的な計算に対応し，X_i は等価起動時間に換算された値である．強制操作対象起動スケジュールパラメータ X_i に対して，図 3.3 の判定部(1)を満足する線形近似係数 a_{ij} 及び最大熱応力 σ_{mj} が存在する場合，式(3.3)を満足するような X_i^* が存在しないので，このような起動スケジュールパラメータ X_i を強制操作対象としない．図 3.3 の処理部(1)と(2)は，式(3.3a)と(3.3b)に対応した処理を行うものである．

本来，強制操作として、タービンロータ表面(j=1,2)とボア(j=3,4)に関して行う必要がある．しかし，ボア熱応力が制限値逸脱しないように表面熱応力より厳しく監視される必要があるので[Hanzalek 66]，運用上，ボア熱応力制限値のマージンが表面より大きく設定されている．そのため，通常，タービンロータ表面の発生熱応力が制限内であれば，ボアの発生熱応力も制限内にある．すなわち，ロータ表面(j=1,2)熱応力制約条件を満足する解は，ほとんどの場合，ロータボア(j=3,4)熱応力制約条件をも満足する．

本研究では，強制操作を効率よく，容易に構築するため，強制操作はタービンロータ表面における熱応力条件だけを含ませることとする($j \in \{1,2\}$)．ここで，式(2.3c)によって定義される制約条件に違反しないように，ボア熱応力の制限はペナルティ関数（式(2.4)）に含まれていることに注意されたい．図 3.4 は(j=1,2)の場合の強制操作を示す．同図の max と min はそれぞれ図 3.3 の処理部(1)と処理部(2)に対応する．

なお，パラメータ調整結果により，強制操作の最大繰り返し回数 n（図 3.3）は，ランダムに生成された初期解とそれ以降の解に対してそれぞれ 10 回と 8 回と設定した．ランダムに生成された初期解は，境界から離れることが多くあるので，境界に移動されるため，多くの強制操作が必要である．多くの回数を初期解に割り当てることにより，初期解を境界近傍に移動できる確率が高くなり，探索性能が向上されることが確認された．

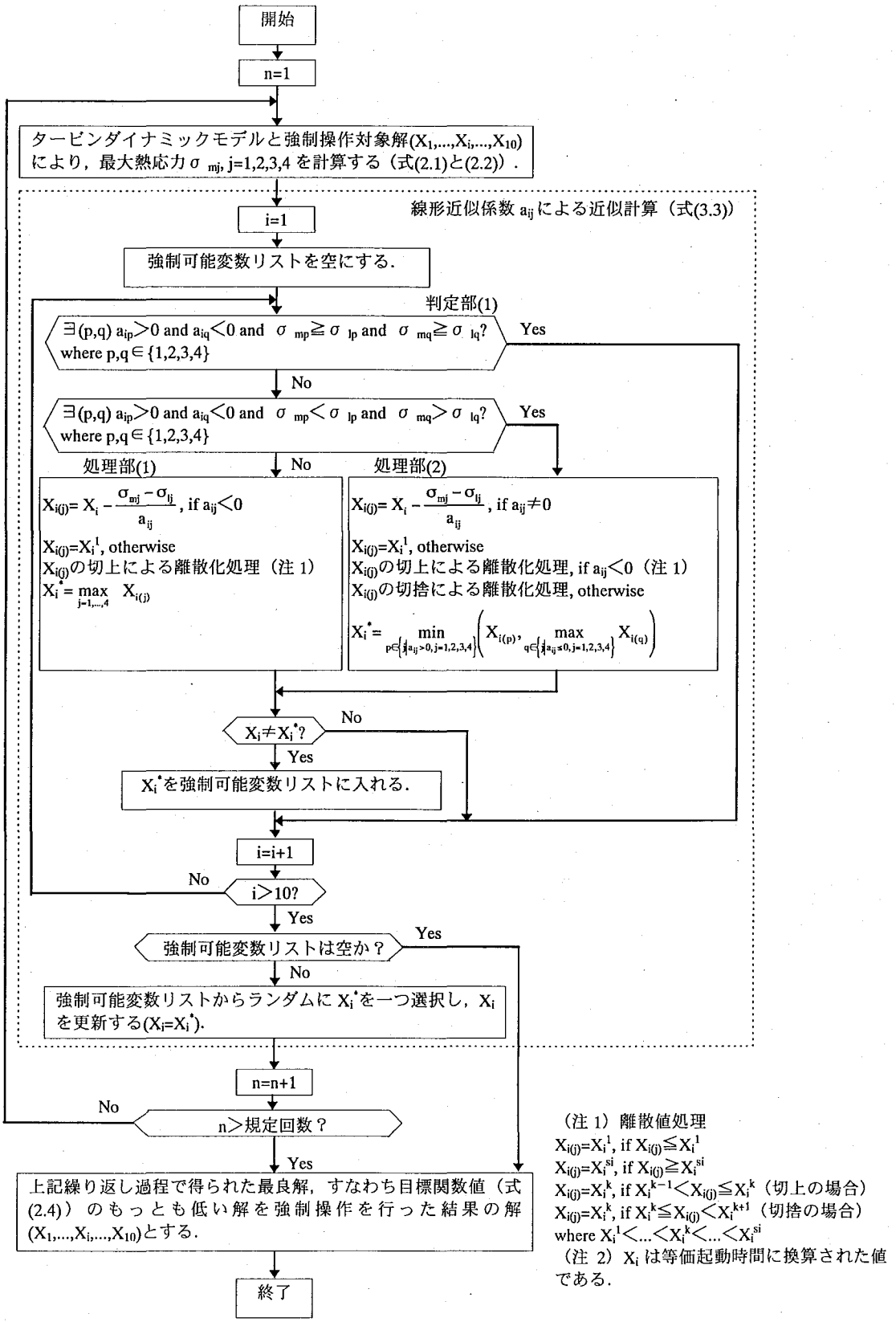
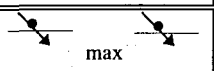
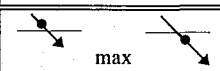
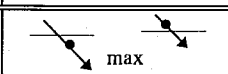
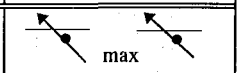
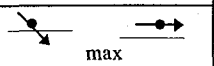
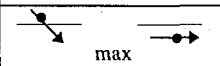
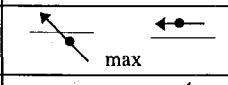
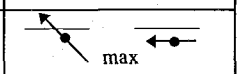
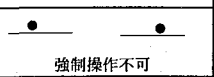
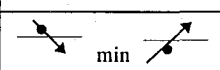
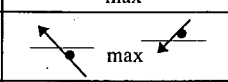
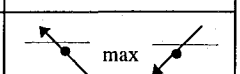
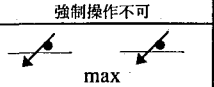
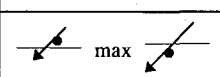
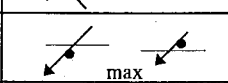
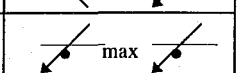
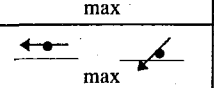
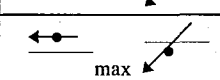
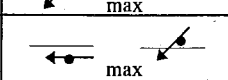
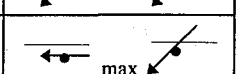
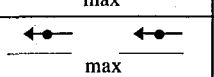
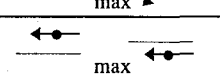
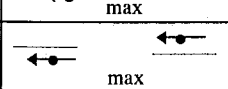
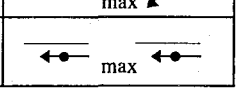


図 3.3: 強制操作アルゴリズムの詳細

		$\sigma_{m1} \geqq \sigma_{l1}$ $\sigma_{m2} \geqq \sigma_{l2}$	$\sigma_{m1} > \sigma_{l1}$ $\sigma_{m2} < \sigma_{l2}$	$\sigma_{m1} < \sigma_{l1}$ $\sigma_{m2} > \sigma_{l2}$	$\sigma_{m1} < \sigma_{l1}$ $\sigma_{m2} < \sigma_{l2}$
$a_{i1}<0$	$a_{i2}<0$	 max	 max	 max	 max
$a_{i1}<0$	$a_{i2}=0$	 max	 max	 max	 max
$a_{i1}<0$	$a_{i2}>0$	 強制操作不可	 min	 max	 max
$a_{i1}>0$	$a_{i2}>0$	 max	 max	 max	 max
$a_{i1}=0$	$a_{i2}>0$	 max	 max	 max	 max
$a_{i1}=0$	$a_{i2}=0$	 max	 max	 max	 max

（注）

●　：σ_{mj} を表わす.

——　：σ_{lj} を表わす.

●—　：$\sigma_{mj} \geqq \sigma_{lj}$ を表わす.

●—　：$\sigma_{mj} < \sigma_{lj}$ を表わす.

⟶　：下記のように強制操作による σ_{mj} と X_i の変化方向及び a_{ij} の正負を表わす. ただし, X_i は等価起動時間に換算された値である.

矢印が上向きの場合（↗ ↖），強制操作により σ_{mj} が増加することを表わす.

矢印が下向きの場合（↘ ↙），強制操作により σ_{mj} が減少することを表わす.

矢印が水平の場合（⇄），強制操作により σ_{mj} が変化しないことを表わす.

矢印が左向きの場合（↖ ⟵ ↙），強制操作により X_i が減少することを表わす.

矢印が右向きの場合（↗ ⟶ ↘），強制操作により X_i が増加することを表わす.

矢印が右上がり場合（↗↙），$a_{ij}>0$ を表わす.

矢印が右下がり場合（↖↘），$a_{ij}<0$ を表わす.

矢印が水平の場合（⇄），$a_{ij}=0$ を表わす.

max　：図 3.3 の処理部(1)に対応する処理を表わす.

min　：図 3.3 の処理部(2)に対応する処理を表わす.

図 3.4: (j=1,2)の場合の強制操作

4 GA による進化的探索

4.1 はじめに

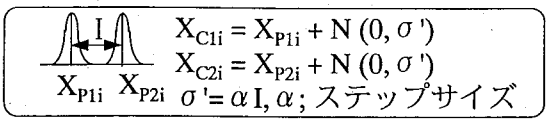
強制操作は，起動スケジュールを境界への移動を図ることであって，最適解の探索を行っているのではないので，最適解を探索するため，強制操作を最適解探索モデルと組み合わせる必要がある．ここでは，最適解探索モデルとしては，GA を使用し，GA をベースに強制操作，再利用機能とタブ戦略を導入した探索モデルを提案する．なお，有望な確率的探索モデルとして，GA の他にタブサーチ(Tabu Search: TS)[Glover 93]とシミュレーテッドアニーリング(Simulated Annealing: SA)[Kirtpatrick 83]がある．タブサーチとは解の探索履歴をタブリストに記憶し，最近探索した解への移動(move)を禁止することにより，探索効率の向上や局所的最適解からの脱出を図るものである．タブサーチは強力なメタ戦略であり，問題クラスによってはシミュレテッドアニーリングや GA の性能を凌ぐことが，種々の benchmark 問題に対する系統的な実験的解析で示されている[久保 95]．多点探索を基本とする GA と探索履歴を参照するタブサーチに対して，シミュレテッドアニーリングは一つだけの解を扱い，探索履歴を利用せず，温度パラメータによって定められたボルツマン分布によって近傍解への遷移を行う確率探索法である．アルゴリズムは極めて簡単で，枠組み自体が極めて汎用にできているので，広範囲の問題に適用できる[Rosen 94]．ここでは，GA の有効性を確認するため，GA をこれらのモデルないしこれらのモデルの組合わせと比較実験により，本研究の提案手法に関する評価を行なう［神谷 97a］．

以下では，タブサーチ及びタブ戦略を TS，タブ戦略を導入した GA を GA+TS，シミュレーテッドアニーリングを SA，タブ戦略を導入した SA を SA+TS と呼ぶ．

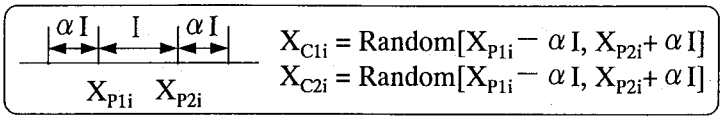
4.2 コード化／交叉の方法

GA のコード化／交叉として，「形質遺伝性に優れた交叉方法である整数コード NDX－α(Normal distribution crossover－α)[Ono 96]」，「多くの関数最適化問題において良好な性能を示した整数コード BLX－α(Blend crossover－α)[Eshelman 93]」及び「伝統的な 2 点交叉（2X；two point crossover）のバイナリコード GA[Goldberg 89a]」を用いて検討をし（図 4.1），実験により 3 者間の性能比較を行う．

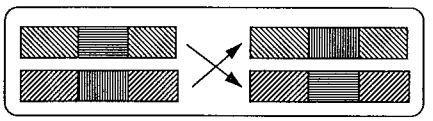
NDX－α と BLX－α は，インターバルスキーマ定理[Eshelman 93]に基づいて考案されたものである．NDX－α は BLX－α と共に，遺伝的な操作としての突然変異と交叉という二



(a) NDX-α（整数コード，集団サイズ2，エリート保存）



(b) BLX-α（整数コード，集団サイズ2，エリート保存）



(c) 2X（バイナリコード，集団サイズ20，エリート保存）

図 4.1: 本研究に用いられる GA のモデル

つの側面を持ち合わせている．突然変異としては，親に含まれないインターバルスキーマを生成することができ，交叉としては，ペアとなっている親に含まれる情報を使って，世代の進行に従い，漸次的に焦点を絞った探索を行うことができる．

NDX－αによる交叉モデルは図 4.1(a)に示される．同図において，ペアとなっている親の起動スケジュールを$(X_{P11},...,X_{P1i},...,X_{P110})$と$(X_{P21},...,X_{P2i},...,X_{P210})$とすると，NDX－$\alpha$は次のようなステップにより，子ペアの起動スケジュール$(X_{C11},...,X_{C1i},...,X_{C110})$と$(X_{C21},...,X_{C2i},...,X_{C210})$を生成する．

a) 10 ペアの親起動スケジュールパラメータの中から，ある一ペアのパラメータ（一点交叉）X_{P1i}とX_{P2i}（交叉サイト）をランダムに選択する．
b) 選択された親ペアの起動スケジュールパラメータ X_{P1i} と X_{P2i} を中心に子ペアのパラメータ X_{C1i} と X_{C2i} を正規分布 $N(0, \sigma')$に従って生成する．
c) 子ペア起動スケジュール $(X_{C11},...,X_{C1i},...,X_{C110})$ と $(X_{C21},...,X_{C2i},...,X_{C210})$ は，$X_{C11}=X_{P11},...,X_{C1i},...,X_{C110}=X_{P110},X_{C21}=X_{P21},...,X_{C2i},...,X_{C210}=X_{P210}$ とすることによって生成される．

ここで，σ'は親同士の距離に応じて適応的に調節される分散を表わし，αは探索ステッ

プサイズで，αを小さく設定することにより，近傍探索することができ，βは集団の早期収束を避けるためのバイアス定数である．パラメータ調整結果により，α=0.05，β=1 と設定した．

BLX－αは，選択された両親のある一ペアのパラメータを X_{P1i} と X_{P2i}，両親間の距離を $I=|X_{P1i}-X_{P2i}|$ とすると，子ペアのパラメータ X_{C1i} と X_{C2i} は区間$[X_{P1i}-\alpha I, X_{P2i}+\alpha I]$，$X_{P1i}<X_{P2i}$ 内で一様にパラメータ値の 2 回の独立抽出により生成される（図 4.1(b)）．子ペアの起動スケジュール$(X_{C11},...,X_{C1i},...,X_{C110})$と$(X_{C21},...,X_{C2i},...,X_{C210})$は，上記 NDX－$\alpha$のステップ c)と同様な処理により生成される．

2X はバイナリコード化された両親の一部を入れ替えることにより，子供を生成する（図 4.1(c)）［神谷 95］．2X は多点交叉の中で，もっともスキーマの破壊が少なく[Spears 91]，強制操作を組み合わせた場合，他の交叉点よりよい探索性能が期待である．突然変異操作は，一般的に低い突然変異率（例えば，0.001）で使われ，スキーマ破壊の元となる[Spears 91]ので，本研究では，突然変異操作の導入をしない．パラメータ調整結果，強制操作と組み合わせた場合，2X の集団サイズが大きく 20 である．この結果は[Spears 91]の提案に符合し，すなわちスキーマ破壊の小さい 2X は大きい集団サイズによりよい性能が得られる．なお，強制操作は生成された解を境界に移動させる時，集団内にない新しい対立遺伝子を生成することができるので，突然変異に相当する働きをすることが期待できる．

NDX－αと BLX－αの整数コード化においては，起動スケジュール$(X_1,...,X_i,...,X_{10})$が染色体に，起動スケジュールパラメータ X_i, i=1,...,10 が染色体を構成する遺伝子に，起動スケジュールパラメータの整数値 X_i^k, i=1,...,10, k=1,...,s_i が対立遺伝子にそれぞれ対応づけられている．表 4.1(p.36)は，起動スケジュールパラメータの表現型（工学単位）と整数コードの場合の遺伝子型（整数値）を示す．

2X のバイナリコード化においては，起動スケジュール$(X_1,...,X_i,...,X_{10})$が染色体に，起動スケジュールパラメータのバイナリ値 X_i, i=1,...,10 の各ビットが染色体を構成する遺伝子に，起動スケジュールパラメータの各ビットの値が対立遺伝子にそれぞれ対応づけられている．起動スケジュールパラメータ X_i, i=1,...,10 のバイナリ値とは，表 4.1(p.36)の整数値をバイナリ値に変換されたものである．

4.3 世代交替モデル

GA は個体の適応度に基づいた確率的な選択により，世代の交替を繰り返しながら，集団を進化させる過程の中で，最適解の探索を行なうモデルである．世代交替のための選択は，

reproduction（例えば，交叉による子の生成）のための選択と生き残るための選択という 2 種類のものにより構成される[Satoh 96]．この 2 種類の選択に対して，伝統的な SGA(Simple GA) [Goldberg 89a]は，次のような処理により行なわれる．a) reproduction のための選択としては，ルーレット選択，すなわち集団の中から，個体の適応度に比例した確率で，reproduction のための親個体を集団サイズと同じ数だけの復元抽出を行なう．b) 生き残るための選択としては，reproduction によって生成された子がすべて選択され，すなわちすべての親が一度に子によって置き換えられることによって，新しい世代が生成される．

[Satoh 96]は，世代交替モデルとして，SGA のような世代間のギャップの大きい世代交替モデルに対して，世代間のギャップの小さい世代交替モデル MGG (Minimal Generation Gap) を提案した．MGG は次のような世代交替のための選択を行なう．a) reproduction のための選択としては，集団の中から，ランダムに親個体を 1 ペアだけの非復元抽出を行なう．b) 生き残るための選択としては，reproduction によって生成された 1 ペアの子と対応する 1 ペアの親からなる 4 個体のファミリーの中から，エリート選択とルーレット選択により，1 ペアの個体を選択し，集団に加える．この場合のエリート選択とは，ファミリーの中で適応度のもっとも高い個体を選択することであり，ルーレット選択とは，残り 3 個体の中から，個体の適応度に比例した確率で一つの個体を選択することである．集団サイズ 20 と 100 の場合，SGA を含めたいくつかの代表的な世代交替モデルと比較した実験結果，MGG がもっともよい性能が示された[Satoh 96]．

本問題におけるパラメータ調整結果，強制操作を導入した場合の NDX－α の集団サイズは 2 であり，エリート保存戦略[Goldberg 89a]を導入した SGA と MGG の実験比較結果，両者はほぼ同様な性能が示された．ここで，本研究において，世代交替モデルとしては前者，すなわち伝統的なエリート保存戦略[Goldberg 89a]を導入した SGA を用いることとする．

なお，本研究において，エリート解とは今まで探索された実行可能解の中で，起動時間 T（式(2.3a)）の最短解である．ルーレット選択では，ペナルティを加算した起動時間 T'（式(2.4)）に基づいて行なう．ここで，個体$(X_1,...,X_i,...,X_{10})$の適応度を $F(X_1,...,X_i,...,X_{10})$とすると，ペナルティを加算した起動時間 $T'(X_1,...,X_i,...,X_{10})$の短い個体に対して高い適応度を与えるため，$F(X_1,...,X_i,...,X_{10})$は次の式により定義される．

$$F(X_1,...,X_i,...,X_{10}) = \overline{T'} + (\overline{T'} - T'(X_1,...,X_i,...,X_{10})) \tag{4.1a}$$

$$\overline{T'} = \frac{\sum_{\text{集団サイズ}} T'(X_1,...,X_i,...,X_{10})}{\text{集団サイズ}} \tag{4.1b}$$

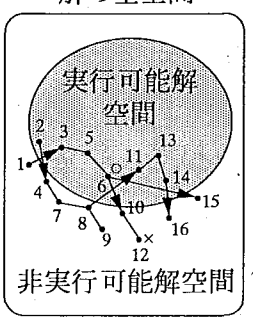
ただし，$\overline{T'}$ は集団内の T'に関する平均値または適応度の平均値である．

4.4 強制操作の導入

強制操作を導入した GA による探索の図式を図 4.2 に示す．同図において，矢印のある線は GA による操作を，矢印のない線は強制操作による操作を示す．強制操作を導入した GA はこの二つの操作の繰り返しにより最適解の探索を行う．強制操作に探索機能を持たせるため，各強制操作過程での強制最良解を強制操作によって生成された解とする．強制最良解とは，制約条件によるペナルティを加える起動時間 T'（式(2.4)）の最短であるような解である．

強制操作は繰り返し法により解を実行可能解空間の境界上に漸近的に移動させるが，生成された子が境界から離れれば離れるほど，その子を境界に戻すため，より多くの強制操作回数が必要とされるので，その分だけ探索効率が悪くなる．従って，強制操作と組み合わせた最適解探索モデルとして，近傍探索モデルが有利であると考えられる．

両親のパラメータが離れた場合，BLX－αにより生成された子が NDX－αよりも親より離れる機会が多い．すなわち，BLX－αにより生成される子が NDX－αよりも境界より離れることが多いと言えるので，強制操作と組み合わせた場合，その分だけ BLX－αの探索効率が悪くなる．伝統的なバイナリコードの 2 点交叉(2X)はその他の多点交叉よりもスキ



a) 解 4,7,8 と 9 からなるような一連続した強制操作過程を一強制操作過程，その過程の中の最良解（例えば解 8）を強制最良解と呼ぶ．

b) 親は強制最良解である．

c) 親 6 と 8 の交叉により子 10 と 11 が生成される．

d) 解 6 は初期解（例えば解 1）から解 14 までの最良解，すなわちエリート解（○）とし，強制最良解 12 が強制最良解 14 よりの改悪解とすると，エリート保存戦略により解 12（×）が淘汰される．

e) 解 6 は再び親となり，親 14 との交叉により，子 15 と 16 が生成される．

図 4.2: 強制操作を導入した GA による境界近傍探索概念図

ーマの破壊においては少ないとされている[Spears 91]が，両親間の一部を相互に入れ替えることにより子の生成を行うため，そのスキーマの破壊力は，探索ステップサイズαの小さいNDX－αやBLX－αと比べて大きいと考えられ，その分だけ，探索効率が悪化してしまう．以上より，強制操作と組み合わせた場合，探索ステップサイズαを小さく設定することにより，親の近傍に子を生成することができるNDX－αが有効であることが考えられるが，4.6節の実験でその効果について確認とする．

4.5 再利用機能とタブ戦略の導入

強制操作と組み合わせた時，近傍探索モデルが有効であるが，近傍探索の場合，最近探索された解に対する繰り返し探索がしばしば見られる．ここでは，この繰り返し探索に伴うコストを避けるため，本研究は，強制操作を組み合わせたGAに対して再利用機能とTSの導入を提案する．

4.5.1 再利用機能の導入

再利用機能は，過去探索された解に対応する最大熱応力のシミュレーション結果を記憶し，同じ解が再探索された時，過去記憶されたシミュレーション結果を再利用することにより，シミュレーションに必要な膨大な計算時間を省き，最適解探索時間の短縮を図るものである．再利用機能を実施するため，記憶すべき計算結果は，生成された各解に対して6点であり，すなわち最大熱応力4点，目的関数値1点とスカーラに変換された起動スケジュール1点である．最近，探索された約千回のシミュレーション計算結果を記憶すれば充分であるので，再利用機能を実現するためのメモリサイズが小さい．また，2分木探索法を用いれば，再利用するための探索コストはシミュレーション計算と比べて無視できるほど小さい．

同じ解を繰り返し探索するに伴うコストを軽減するという目的は同じであるが，TSと再利用機能は，異なった側面を持つ．すなわち，TSは局所的最適解から脱出する能力を持っているのに対し，再利用機能はTSにおける移動の禁止の副作用としての探索の手落ちを避けることが期待できる．

TSを導入するGAとは，TSとGAの融合による探索モデルであるが，以下では，強制操作を導入したTSによる探索モデルについて説明した後，TSとGAの融合による探索モデルについて説明する．

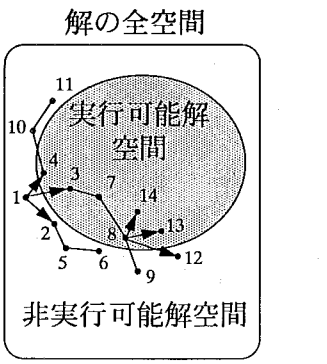
4.5.2 タブサーチによる探索

ここでは，強制操作を導入したTSによる最適解探索モデルについて説明する．TSとは，最近探索した解または移動の属性を有限サイズの先入れ先出し(first-in, first-out)リスト，すなわちタブリストに記憶しておき，最近探索した解への移動を禁止することにより，探索効率の向上や局所的最適解からの脱出を図るものである．

(1) タブリストの属性と種類

一般にタブリストの属性は主として「解の移動方向」または「解との禁止距離」によって定義される．本論文が対象とする多次元の整数最適化問題では，「解の移動方向」とは，例えば，起動スケジュールパラメータ X_i, i=1,...,10 の値の増加まはた減少，すなわち上げ移動（up move)または下げ移動(down move)によって定義できる[Glover 86]．この場合，タブリストに記憶される移動と逆の移動が禁止される．「解との禁止距離」とは，タブリストに記憶された解と類似な解または定義された距離内に解を生成する移動を禁止する[Glover 94].

TSに強制操作を導入した場合の探索の図式を図4.3に示す．解の移動は，同図に示されるように，近傍内の解の生成と生成される解の複数回の連続した強制操作により構成される．従って，「解の移動方向」によるタブリストの属性を定義することが難しく，ここでは，「解との禁止距離」によりタブリストの属性を定義することとする．なお，4.6節の実



a) 近傍へ遷移する元の解を元解と呼ぶ（例えば解1）．

b) 一つの元解より生成される強制最良解（例えば解6,8と11）の中の最良解（例えば解8）が元解となる．

c) 近傍サンプリングサイズを3とすると，元解8より近傍解12,13と14が生成される．

→ :TSによる近傍遷移
── :強制操作
数字 :解の生成順番

図4.3: 強制操作を導入したTSによるの境界近傍探索概念図

験では，「解との禁止距離」をパラメータ調整実験結果により 0 と設定した．最近探索された解に一致するときのみ対応する移動が禁止される．

最近探索された解を記憶するタブリストとして，本研究では次の 3 種類を用意する．図 4.3 を参照し，以下のタブリストを説明する．

a) **元解タブリスト**　これはサイズ有限の先入れ先出しリストで，最近探索された元解（解 1 と 8）を記憶する．本リストは，強制操作なしの TS のタブリストに相当し，最近探索された解への再探索を禁止することを目的としている．

b) **強制最良解タブリスト**　元解が生成されるたびに，本リストを空とする．本リストは元解（解 1）と次の元解（解 8）との間に生成されるすべての強制最良解（例えば，解 6，8 と 11）を記憶する．本リストは，同じ強制最良解の生成を避けることを目的としている．

c) **強制遷移解タブリスト**　各強制操作過程が開始されるたびに，本リストを空とする．本リストは，一強制操作過程内のすべての解（例えば解 3，7，8 と 9）を記憶することにより，一強制操作過程内でのサイクリング現象を避けることを目的としている．

元解から近傍解の生成過程では，強制最良解タブリストと強制遷移解タブリストは空リストとなり，元解タブリストだけが参照される．強制操作過程では三つのタブリスト，すなわち元解タブリスト，強制最良解タブリストと強制遷移解タブリストが参照される．

(2) タブリストのサイズ

タブリストのサイズが小さすぎると，サイクリング現象が発生し，大きすぎると，多くの移動が禁止されるため，解の質が落ちる[Glover 93]．4.6 節の実験では元解のサイズをパラメータ調整実験結果により 9 と設定した．また，上記のタブリスト属性の定義により，強制最良解タブリストと強制遷移解タブリストのサイズはそれぞれ近傍サンプリングサイズ（8（下記参照））と一強制操作過程の最大強制回数（8 ないし 10（3.3.2 節参照））に等しい．

(3) 近傍のサンプリングサイズ

相互間のハミング距離が 1 である起動スケジュールを近傍関係にあるという．表 4.1(p.36) により，起動スケジュール$(X_1,...,X_i,...,X_{10})$の近傍サイズを $N(X_1,...,X_i,...,X_{10})$とすると，起動スケジュールパラメータ X_i, i=1,...,10 がレンジの上限または下限にある場合（例えば X_1=120）とない場合（例えば X_1=180）によって，$N(X_1,...,X_i,...,X_{10})$は{10,...,16}の値を取る．

近傍サンプリングサイズを S とすると，$S \ll N(X_1,...,X_i,...,X_{10})$となることが推奨されている[Glover 93]．本研究では，パラメータ調整の結果，近傍サンプリングサイズは 8 と設定し，$N(X_1,...,X_i,...,X_{10})$の最大値，すなわち 16 の半分である．ここで，強制操作は非決定的であり，すなわち一つの近傍解に対して，可能な強制は何通りも存在するので，実際の近傍サイズは 16 よりも大きいことに注意されたい．

4.5.3 タブ戦略を導入した GA による探索

TS と GA の融合に関する[Glover 94]と[Fox 93]の提案は，何れも TS による探索の集中化(intensification)と GA 探索の多様化(diversification)という二つの側面を持ち合わせて探索を行わせることを目的としている．[Glover 94]では，TS により最近探索された複数解を GA の「交叉と突然変異」操作により，新しい解を散らして(scatter)生成し，探索の多様化を図っている．[Fox 93]では，「TS により最近探索された解」とある割合で「成層ランダムリスタート計画(stratified random restart scheme)により,解空間全体にわたって，まばらに生成される履歴なし(historyless)解」と組み合わせ，GA の「交叉」操作により新しい解を生成し，探索の多様化を図っている．

しかし，このような集中化と多様化という観点で融合された TS と GA の枠組みを本問題に適用すると，GA による多様化探索局面では，解が境界より遠く離れて生成されることがある．生成された子が境界から離れれば離れるほど，その子を境界に戻すため，より多くの強制操作回数が必要とされるので，このような探索では探索効率が悪化することが予想される．なお，TS の枠組み上に近傍探索 GA を導入することも考えられるが，このような融合はもはや本来の集中化・多様化という観点に基づいて行われるものではなくなる．また，4.6 節で説明されるように，再利用機能を導入した場合，サンプリングサイズの大きい TS は集団サイズの小さい近傍探索 GA より探索効率が劣っている．本研究では以上とは別な接近法をとり，近傍探索 GA の枠組み上に TS を導入し，近傍探索に見られるサイクリング現象を回避することにより，探索効率の向上を図る．GA+TS のタブリストは，TS に対応して以下の 3 種類があり，図 4.2 に基づいて説明する．

a) **親解タブリスト**　これは，TS の元解タブリストに対応するものであり，パラメータ調整結果により，サイズ 10 と設定した．最近までの親世代の解（例えば解 1，2，6，8）は，親解タブリストに記憶される．ただし，エリート保存戦略導入のため，エリート解（解 6）の再探索は禁止しないものとする．この戦略は，TS の加速基準（aspiration

criteria）[Glover 93]に準拠するものである．また，エリート解によって淘汰された強制最良解（解 12）は，探索されなかった解として，親解タブリストには登録しないものとする．

b) **強制最良解タブリスト**　このリストのサイズは GA の集団サイズと同じである．ある世代における子解がすべて生成されるたびに，本リストはこれらの子解により初期化される．本リストはある世代の子解（解 3，4）と次の世代の親解（解 6，8）との間に生成される強制最良解により更新される．本リストは，集団内で同じ強制最良解の生成を避けることを目的としている．

c) **強制遷移解タブリスト**　このリストは，TS の強制遷移解タブリストに対応し，一強制操作過程内でのサイクリング現象を避けることを目的としている．

4.6 実験

4.6.1 実験の目的と方法

起動スケジュールの最適解または近似最適解の探索を行なうため，本研究は強制操作，再利用機能と TS を導入した GA による探索モデルを提案した．実験の目的は，本提案の有効性を確認することであり，以下のような実験を行なう．

(1) 最適解の確認

人間により設計される起動スケジュールを本提案手法による探索された最適起動スケジュールと比較し，本提案手法による起動時間の短縮について確認する．

(2) 強制操作の有効性の確認

強制操作を導入する最適解探索モデルと強制操作を導入しない最適解探索モデルによる比較実験を行い，両者の探索性能の評価により強制操作の有効性を確認する．ここでは，最適解探索モデルとして GA を用いる．

(3) 近傍探索モデルと再利用機能の有効性の確認

強制操作を導入して，境界近傍探索を行なうため，近傍探索モデルが有効である．ここでは，この近傍探索モデルの有効性を確認するため，NDX－α，BLX－α と 2X を用いる．なお，ここで，各モデルに対して，再利用機能を導入する場合と導入しない場合による比較実験を行い，再利用機能を評価する．

(4) GA+TS 及び各種探索モデルの比較

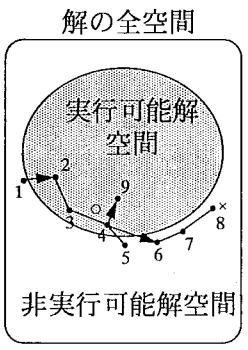
GA+TS の有効性を客観的的に評価するため，GA，TS，SA，SA+TS，HC(Hill Climbing) と RW(Random Walk)の比較実験を行なう．なお，これらのモデルはいずれも強制操作を導入したものである．GA+TS，GA と TS は 4.2 節から 4.5 節まで説明されているので，以下は SA，HC と RW について簡単に説明をする．

a) SA　多点探索を基本とする GA と探索履歴を参照する TS に対して，SA は一つの解だけを扱い，探索履歴を利用せず，温度パラメータによって定められたボルツマン分布（式(4.2)）によって近傍解への遷移を行う確率探索法である．

$$P_a = 1, \quad \text{if } T_n' < T_c' \tag{4.2a}$$

$$= \exp\left(\frac{-(T_n' - T_c')}{\tau}\right), \quad \text{otherwise} \tag{4.2b}$$

ここで，T_c'と T_n'はそれぞれ元解と近傍解に対応する目的関数値（式(2.4)），P_aは元解から近傍解への遷移が受理される確率，τは探索の進行に伴い，その値が漸次的に減少する温度パラメータである．本研究では運用上でよく使われる冷却式($\tau \leftarrow \gamma\ \tau$ [Rosen 94])を用いる．各温度において，n 回探索を実施した後，同冷却式により温度パラメータ τ の更新を行う．パラメータ調整実験結果により，γ=0.9，τ の初期値 τ_0=1,000，n=7 と設定した．SA と強制操作を組み合わせた場合の探索の図式を図 4.4 に



a) 近傍へ遷移する元の解を元解と呼ぶ（例えば解 1）．

b) 強制最良解 8 がその元解 4 よりの改悪解とし，ボルツマン分布に従って受理されなかったとすると，解 8（×）への遷移は取りやめとなる．

c) 解 4（○）は再び元解となり，近傍解 9 が生成される．

図 4.4: 強制操作を導入した SA による境界近傍探索概念図

示す.

b) **SA+TS**　SA に TS を導入する場合，上述の 4.5.2 節の TS または 4.5.3 節の GA+TS と同様に 3 種類のタブリストを用意するものとするが，ここでは説明を省略する.

c) **HC**　HC とは温度パラメータ τ を 0 に固定した SA に相当するものである．本問題はいくつかの局所的最適解を持つが，ここでは，強制操作を HC と組み合わせて，どこまで探索可能かについて確認をする.

d) **RW**　RW とは温度パラメータ τ を∞に固定した SA に相当するものである．RW も近傍探索モデルであるが，強制操作と組み合わせた場合，どこまでの探索性能が示されるかを確認する.

(5) 探索効率と解の質に関する評価方法

本問題の最適化計算において，目的関数を評価するため制約関数のシミュレーション計算はもっとも多くの計算時間を占めているので，ここでは，シミュレーション回数によって各最適化手法の探索効率の評価を行う．解の質に関しては，試行回数の中で最適解または近似最適解が得られた回数で評価する．近似最適解とは，最適解より起動時間が 1% を超えないような実行可能解である.

4.6.2 実験結果

表 4.1 は，起動スケジュールパラメータの設計値と探索により確認されている最適値を示す．同表において，制限熱応力条件 1 とはタービンウォーム起動に対応した制限熱応力の設計値，制限熱応力条件 2 とは，設計値より厳しく設定した制限値を表わす．表 4.2 は，強制操作に対して，近傍探索 GA 及び再利用機能の導入が有効であることを確認する実験結果である．表 4.3 は，強制操作と再利用機能を導入した場合の各最適化手法の性能比較に関する実験結果である.

4.6.3 結果の考察

(1) 最適解に関する実験結果の考察

表 4.1 において，設計制限熱応力条件（制限熱応力条件 1）では，本研究で提案した手法 (GA+TS)によって最適化された起動スケジュールの起動時間は 211.6 分で，設計値の起動時間 243 分より約 32 分改善された．本実験は制限熱応力に焦点を当てて行ったが，実際には，起動時の振動発生を代表とする経験的な制約条件を考慮すると，タービンウォーム起動に

表 4.1: タービンウォーム起動における制限熱応力条件 1 と条件 2 に対応する最適起動スケジュール

$X_i \in \{X_i^k \mid k=1,...,s_i\}$, i=1,...,10

s_i は X_i のサイズを表わし，例えば，$s_1=3$, $s_{10}=78$ である．

X_i	起動スケジュールパラメータX_iの名称	レンジ		制限熱応力		
		工学単位(表現形)	整数値(遺伝子形)	条件1		条件2
				設計	最適	最適
X_1	加速率(rpm/min.)	120,180,360	0,1,2	180	360	360
X_2	低速保持時間(min.)	0,1,...,28	0,1,...,28	0	0	0
X_3	高速保持時間(min.)	5,6,...,37	0,1,...,32	9	5	5
X_4	増負荷率 1 (%/min.)	0.5,1.0	0,1	1.0	1.0	1.0
X_5	初負荷保持時間(min.)	0,1,...,40	0,1,...,40	12	0	25
X_6	増負荷率 2 (%/min.)	0.5,1.0	0,1	1.0	1.0	1.0
X_7	増負荷率 3 (%/min.)	0.5,1.0	0,1	1.0	0.5	0.5
X_8	増負荷率 4 (%/min.)	1.0,1.5	0,1	1.0	1.0	1.0
X_9	増負荷率 5 (%/min.)	1.5,2.0,2.5,3.0	0,1,2,3	2.0	3.0	2.0
X_{10}	通気主蒸気温度(℃)	349,350,...,426	0,1,...,77	376	349	380
T	起動時間(min.)	---	---	243	211.6	257.1

おいて，加速率は $360rpm^2$ ではなく $180rpm^2$ とする方が好ましい．この場合，最適解は 221.6 分となり，最適化された起動時間の短縮は 22 分となる．ここで，表 4.1 に示される加速率パラメータ X_1 のレンジ 120，180，360 を単に 180 に設定するだけで，このような振動制約条件を容易に最適化モデルに加えることができることに注意されたい．

設計値より厳しく設定された制限熱応力（制限熱応力条件 2）では，最適化された起動時間は 257.1 分である．制限熱応力が厳しいほど，実行可能解空間が狭くなり，しかも最適解が実行可能解空間の境界上に存在するので，最適解の起動時間が長くなる．制限熱応力条件 2 の最適起動時間 257.1 分はこの実験で示されるように制限熱応力条件 1 の最適値 211.6 分より長くなっている．なお，この条件のもとでの設計は行われていないことを付記する．

(2) 強制操作，再利用機能と近傍探索 GA の有効性

表 4.2 の実験結果により，強制操作と組み合わせた場合，NDX－αがもっとも探索性能がよい．NDX－α は BLX－α よりわずかならが，探索に必要な平均シミュレーション回数が少なく，2X よりは大幅に少なく，強制操作と組み合わせた場合，近傍探索 GA が有効であると言える．また，NDX－α に対する強制操作の導入前後の比較では，後者は前者より

表 4.2: 強制操作，再利用機能と近傍探索 GA の有効性に関する実験結果

(1) NDX－α: Normal Distribution Crossover-α

(2) BLX－α: Blend Crossover-α

(3) 2X: Two-point Crossover

(4) 再利用率$=\dfrac{\text{再利用回数}}{\text{シミュレーション回数}+\text{再利用回数}}\times 100\,(\%)$

(5) タービン起動モード:ウォーム起動

(6) シミュレーション回数:最適解探索されるまでの回数

(7) 制限熱応力条件 1

(8) 全試行に対して，最適解が得られている．

探索モデル	強制操作	試行回数	シミュレーション回数				
			ave	var	max	min	再利用率(%)
NDX－α	**あり**	**50**	**73**	**30.0**	**135**	**34**	**31**
BLX－α	あり	50	74	31.6	145	18	27
2X	あり	50	285	98.0	687	155	7
NDX－α	なし	8	392	183.3	719	176	17

平均シミュレーション回数が大幅に少なく，強制操作が探索効率の向上に有効であることを示している．さらに，強制操作の導入前後により探索された最適解の起動スケジュールパラメータの値は同様であり，境界近傍に探索を限定する強制操作の導入は最適解を見逃すことなく，効率的に探索を行えることが言える．

近傍探索モデルの場合，同じ解の繰り返し生成がしばしば発生するので，表 4.2 に示されるように，探索ステップサイズαを小さく設定された近傍探索モデルである NDX－αの再利用率が大きく，31%である．両親のパラメータが離れた場合，BLX－αにより生成された子が NDX－αよりも親から離れる機会が多いので，BLX－αの再利用率は NDX－αより少なく，27%である．また，両親間の一部を相互に入れ替えることにより子の生成を行う 2X は，子を親より遠くに離れて生成することが多いので，再利用率がもっとも少なく，わずか 7%である．これにより，再利用機能は近傍探索に有効であり，強制操作を導入した NDX－αに対して探索効率が 31%の向上となる．

(3) 強制操作を導入した場合の各探索モデルの比較

表 4.3 に強制操作を導入した各種最適化手法の性能比較を示す．表 4.2 と表 4.3 の比較により，制限熱応力条件 2 の最適解または近似最適解探索に必要なシミュレーション回数が制限熱応力条件 1 より多いので，ここでは，探索が難しい条件 2 を使って各種最適化手法の性能比較を行うこととした．

a) 強制操作を組み込んだ HC と RW の性能について　HC は現在の解の改善方向に遷移するだけなので，初期局面では効率のよい探索が期待できるが，探索過程で一度，局所的最適解に陥いると，脱出できなくなる．表 4.3 の実験結果により，HC のシミュレーション回数はもっとも少なく，100 回の試行で最適解または近似最適解が得られた回数は 92 回である．ただし，HC は強制操作と組み合わせて使われていることから，強制操作によって局所的最適解からの脱出が行われた場合が少なくないことを付記する．すなわち，HC を強制操作と組み合わせることにより，近傍解が現在の解より改悪であっても，強制最良解が現在の解より改善される場合，改悪である近傍解を迂回し

表 4.3:　強制操作を導入した場合の各最適化手法の性能比較実験結果

(1) タービン起動モード:ウォーム起動

(2) 試行回数:100

(3) 制限熱応力 2

(4) 再利用率$=\dfrac{\text{再利用回数}}{\text{シミュレーション回数}+\text{再利用回数}}\times 100\ (\%)$

(5) GA の交叉は NDX－αを用いる．

探索モデル	シミュレーション回数									起動時間(min.)				(近似)最適解回数
	再利用機能なし				再利用機能あり				再利用率(%)					
	平均	最大	最小	分散	平均	最大	最小	分散		平均	最大	最小	分散	
GA	702	2,248	9	524	428	1,144	9	266	39	258.8	259.7	257.1	0.7	100
TS	539	1,809	7	322	449	1,180	7	244	16	259.0	259.7	257.3	0.5	100
SA	1,287	2,082	8	639	892	1,470	8	424	36	259.7	265.8	257.2	1.4	72
GA+TS	**462**	**1,571**	**9**	**334**	**385**	**1,287**	**9**	**254**	**16**	**258.8**	**259.7**	**257.1**	**0.7**	**100**
SA+TS	1,242	2,071	11	501	1,134	1,844	11	440	8	259.4	261.7	257.3	1.0	85
HC	623	2,098	8	426	291	694	8	137	53	259.7	270.2	257.2	2.3	92
RW	937	2,269	6	590	695	1,667	6	439	27	261.4	267.4	257.3	2.2	27

て，強制最良解への遷移が期待できる場合がしばしば生じた．

RW (Random Walk)は生成された近傍解はすべて受理される．RW は予測の通りに性能がよくなく，100 回試行で得られた最適解または近似最適解の回数はわずか 27 回である．

以上より，本最適化問題においては，強制操作を導入することにより，HC や RW であっても効率のよい探索およびそこそこの質の解が期待できることが確認されたが，より質の高い解を効率よく見出すためには，さらに高度な戦略的探索法を確立する必要があると帰結される．

b) GA，TS 及び SA の性能について　表 4.3 に示されるように，GA，TS 及び SA をそれぞれ単独に適用し，再利用機能を導入しなかった場合には，起動時間では GA が TS よりやや短いが，シミュレーション回数では TS が GA に比べて平均 20 数%少なくなっている．一方，再利用機能を導入した場合，GA は TS より平均 5%程度少ないシミュレーション回数となっている．再利用機能を導入しない場合でも，GA+TS は，GA と TS のそれぞれの単独よりよい性能が示される．さらに，GA+TS に再利用機能を導入した場合にもっとも高い性能が確認されている．

以下では，本実験結果を踏まえて，各解法の特徴を対比的に考察する．

b-1) GA 対 SA　エリート保存戦略を導入した集団サイズ 2 の近傍探索 GA は，あたかもエリート個体による HC とその近傍にエリートでない個体による RW を組み合わせたような探索になっている．HC は効率よく山登りをしながら，ときどき RW の助けにより局所的最適解を迂回することができる．また，HC と RW 両者の役割は両者間の相対位置によって，ときどき入れ代わる．一方，1 個体の SA は温度パラメータだけを頼りに前半では RW，後半では HC のような探索をしている．局所的最適解に捕えられないため，前半の RW のような探索に費やす期間を後半の HC よりも十分長くとるように，温度パラメータの設定を行う必要がある．本実験結果は，HC にガイドされながら，RW により局所的最適解を迂回するような近傍探索 GA が，長い時間で RW として広範囲に徘徊する SA より有利であることを示している．

b-2) GA 対 TS　近傍探索 GA は，親の近くに子を生成するので，同じ場所を何度も訪ねることがある．再利用機能を導入しなかった場合，このような繰り返しを戦略的に禁止する TS と比べて，GA の探索効率が劣っていることが確認された．

一方，再利用機能を導入すると，同じ場所を繰り返して訪ねても，記憶されるシミュレーションの計算結果の再利用によりむだな解の評価のための再計算を省くこ

とができる．さらに，TS は，近傍の解が元解より改悪であっても遷移を行うので，RW のような探索を避けるため，近傍サンプリングサイズをある程度以上の大きさとする必要がある．本問題ではパラメータ調整結果，TS の近傍サンプリングサイズが 8 であり，集団サイズ 2 である近傍探索 GA よりサイズが大きい分だけ，GA は TS より探索性能が若干優れているといえる．

b-3) GA+TS 対 GA と TS　再利用機能ありとなしにかかわらず，GA+TS は GA 単独と TS 単独より探索性能がよい．GA+TS の集団サイズが 2，TS の近傍サンプリングサイズが 8 であるため，前記 b-2)と同様にこのサイズの相違分だけ，GA+TS の探索効率がよいと考えられる．再利用機能を導入した場合，集団サイズ 2 の GA+TS が同じ集団サイズである GA よりよい理由は TS の導入により，サイクリング現象が発生しがちな局所的最適解領域より，脱出しやすいからである．局所的最適解領域で，長く滞在すればするほど，再利用機能により記憶されていない新しい解が生成する機会が増えるので，その分だけ GA は GA+TS より探索効率が落ちると考えられる．

4.7 おわりに

本章では，多くの局所的最適解を持ち，オンライン探索問題として解空間サイズの大きい発電プラント起動スケジューリング問題における最適解または近似最適解を効率よく見出す探索手法を提案し，性能の比較・評価を行った．

本研究は問題固有の特徴にあわせて種々な戦略の導入を行った．本問題の最適解は実行可能解空間の境界上に存在するという考察結果から，探索効率向上のため，探索空間を境界の近くに限定する強制操作に関する提案を行った．生成される解が境界より遠く離れないように，近傍探索 GA を採用した．近傍探索法では，同じ解を何度も繰り返して探索してしまうので，同じ解に対する目的関数の評価に必要な膨大なシミュレーション計算を省くため，過去の解の計算結果を記憶し，再利用する機能の導入を行なった．

さらに，再利用機能とは対照的な接近法であるが，最近探索した解の履歴を記憶し，同じ解への移動(move)を禁止することにより，探索効率の向上や局所的最適解からの脱出を可能とする TS を導入し，実験結果により効率のよい探索を示した．GA の遺伝子型表現においては，連続な実数値であるような起動スケジュールパラメータを許容される精度範囲で整数コードで表現することにより，解全空間サイズの縮小，再利用機能の導入が可能となり，効率のよい探索手法の実現に寄与した．

SPARC station 20 において，GA+TS に基づく近似最適解探索に必要な CPU 時間は，平均

約 6 分である．この性能は，オフラインでの起動スケジュール設計では許容されると考えられる．しかし，オンライン運転での使用のためには，数十秒以下での処理時間が要求される．第 5 章では，強化学習を導入し，学習効果による処理速度の向上を図る探索モデルを提案する．

5 GA と強化学習の融合による適応的探索

5.1 はじめに

第 4 章では，強制操作，再利用機能と TS を導入した GA による起動スケジューリングを提案した．しかし，この接近法では，最適解または近似最適解探索に必要な平均 CPU 時間は，SPARC station 20 上で，約 6 分かかり，目標とする 30 秒内に最適解または近似最適解を探索するというオンライン探索性能を満足していない．ここでは，本問題のブレークスルーを図るため，GA と強化学習を融合したハイブリッド方式を提案する．強化学習は予め代表的な制限熱応力に対応する最適解を学習し，探索開始時では学習効果により任意に与えられる熱応力制限値に対する有望な解候補を生成し，最適解の探索を加速する．また，強化学習の学習過程において，強化学習と GA を融合し，GA が強化学習を有望な領域で学習するようにガイドし，学習の加速を図る．また，強化学習は，GA のガイドによる学習効果により，探索過程の序盤において，有望な解を生成することにより，GA の探索の加速を図る．このような相乗効果により，探索と学習の加速が期待できる．さらに，任意に与えられる種々な制限熱応力に対して頑健性のある探索モデルを構築するため，第 4 章で提案したモデルを拡張して，複数境界条件に対する強制操作と複数エリート保存戦略を導入する［神谷 97b］．

5.2 複数境界に対する強制操作

以下の議論に必要な用語を図 5.1 と対比して定義しておく．

定義 5.1（緩和実行可能解）

緩和実行可能解とは，次の式を満足する解である．

$$\forall(i)\ X_i \in \{X_i^k \mid k=1,...,s_i\},\ i=1,...,10 \tag{5.1a}$$

$$\forall(j)\ \sigma_{mj}(X_1,...,X_i,...,X_{10}) \leqq \sigma_{lj} + \delta_j,\ j=1,...,4 \tag{5.1b}$$

定義 5.2（起動スケジュールパラメータが連続である場合の緩和境界）

緩和境界とは，下記の式(5.2)，及び緩和制約条件式(5.1)を満足する緩和実行可能解$(X_1,...,X_i,...,X_{10})$によって定義される．ただし，式(5.1a)において，X_iは連続であるとする．

$$(\exists(j)\ \sigma_{mj}(X_1,...,X_i,...,X_{10}) = \sigma_{lj} + \delta_j,\ =1,...,4)\ \text{or} \tag{5.2a}$$

$$(\forall(i)\ (X_i=X_i^1 \text{ or } X_i=X_i^{si}),\ i=1,...,10) \tag{5.2b}$$

定義 5.3（起動スケジュールパラメータが連続である場合の境界の外側近傍）

境界の外側近傍とは，下記の式(5.3)，及び緩和制約条件式(5.1)を満足する緩和実行可能解$(X_1,...,X_i,...,X_{10})$によって定義される．ただし，式(5.1a)において，X_iは連続であるとする．

$$(\exists(j)\ \sigma_{lj} < \sigma_{mj}(X_1,...,X_i,...,X_{10}) \leqq \sigma_{lj} + \delta_j,\ =1,...,4) \text{ or} \tag{5.3a}$$

$$(\forall(i)\ (X_i=X_i^1 \text{ or } X_i=X_i^{si}),\ i=1,...,10) \tag{5.3b}$$

以上の定義において，$\delta_j>0$は近傍サイズを決定する定数で，本研究では設計制限熱応力の約 4%とした．以下では，起動スケジュールパラメータが連続であるとして説明を行い，離散の場合の説明や，定義 5.2 と 5.3 に対応する離散の場合の定義を省略とする．

複数境界に対する強制操作とは，与えられた解を定義 3.1 の境界または定義 5.2 の緩和境界に移動させることである．後者に対する線形近似計算式は，式(3.3)と対応して，次式により与えられる．

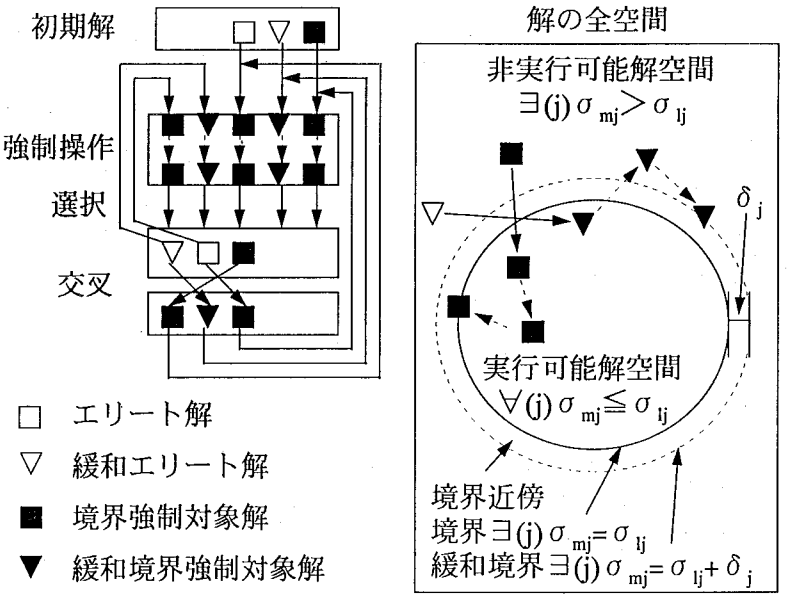


図 5.1: 複数境界探索戦略を導入した GA

$$X_i^* = \min_{X_i \mid \forall(j^-)\sigma_{mj^-} \le \sigma_{lj^-}+\delta_{j^-} \text{ かつ} \forall(j^+)\text{て}\sigma_{mj^+} \text{を増加させない}} \left| \sigma_{mj^+} - \left(\sigma_{lj^+} + \delta_{j^+} \right) \right|, \text{ if } \exists(j)\, \sigma_{mj} > \sigma_{lj} + \delta_j \quad (5.4a)$$

$$X_i^* = \min_{X_i \mid \forall(j)\sigma_{mj} \le \sigma_{lj}+\delta_j} T, \text{ if } \forall(j)\ \sigma_{mj} \leqq \sigma_{lj} + \delta_j,\ j=1,\ldots,4 \quad (5.4b)$$

$$X_i \leftarrow \underset{i=1,\ldots,10}{\text{random}} \left(X_i^* \mid X_i^* \neq X_i \right) \quad (5.4c)$$

制約条件つき探索問題において，実行可能解空間境界の内側に限定することなく，外側も積極的に探索することにより，探索の頑健性と性能向上が期待できる[Smith 89],[Glover 93]．本論文では，GA が集団の進化を基本とした複数解による探索モデルであることに着目して，図 5.1 に示されるような複数境界に対する強制操作と以下に説明する複数エリート保存戦略との組合わせにより，境界の両側を同時に探索する方法を提案する．以降では，この二つの戦略を合せて，複数境界探索戦略と呼ぶ．

5.3 複数境界探索戦略を導入した GA

境界の両側の探索を行うため，境界の外側近傍（定義 5.3）の解も保存する必要があるので，ここで，従来のエリート保存戦略[Goldberg 89a]を拡張して，複数エリート保存戦略を提案する．

図 5.1 は，この複数エリート保存戦略及び 5.2 節で提案した複数境界に対する強制操作を導入した GA による探索モデルを示す．複数エリート保存戦略により保存される解は，エリート解“□”と緩和エリート解“▽”である．エリート解とは，今まで探索された実行可能解(式(2.3b)と(2.3c))の中で，起動時間 T（式(2.3a)）の最短な解である．緩和エリート解“▽”とは，今まで探索された緩和実行可能解(式(5.1))の中で，起動時間 T（式(2.3a)）の最短な解である．

図 5.1 において，本モデルは，初期解生成，強制操作，選択と交叉により構成される．

(1) 初期解生成

初期解生成では，三つの起動スケジュールがランダムに生成される．後で説明する強化学習と組み合わせた場合，初期解集団は，強化学習により生成される解“■”及び複数エリート保存戦略により選ばれたエリート解“□”と緩和エリート解“▽”から構成される．境界（定義 3.1）に対する探索回数を増やすため，エリートでも緩和エリートでもない解は境界強制対象解“■”とする．

(2) 複数境界に対する強制操作

境界（定義 3.1）に対する強制操作とは，エリート解“□”や境界強制対象解“■”を境界に移動し，一方，緩和境界（定義 5.2）に対する強制操作とは，緩和エリート解“▽”や緩和境界強制対象解“▼”を緩和境界に移動することである．

(3) 選択

選択では，強制操作によって生成された集団の中から，三つの解を選択する．二つは，複数エリート保存戦略により選ばれたエリート解“□”と緩和エリート解“▽”である．残りの一つは，境界（定義 3.1）に対する探索回数を増やすため，境界条件に基づくペナルティ加算した起動時間 T'（式(2.4)）により，ルーレット選択された境界強制対象解“■”である．

(4) 交叉

近傍探索を実現するため，交叉は NDX－α を用いる．本提案モデルでは，集団サイズを小さく設定している．集団は基本的には三つの解により構成されるが，エリート解や緩和エリート解による探索回数を増やすため，強制操作では，さらに，エリート解と緩和エリート解を集団に加えている．パラメータ調整結果，強制操作と組み合わせた場合，このような小集団構成による探索効率がもっともよいと確認されている．

初期収束を避けるため，GA の集団サイズは，ある程度大きく取る必要があるが，実用問題において，集団サイズの大きい GA は，シミュレーテッドアニーリングや TS に遜色を示すことがある[Reeves 93]．また，エリート保存のないバイナリコード GA に対して，直列計算によるスキーマ探索効率が最大となるための集団サイズの最適値は小さく 3 であることを示し，初期収束を避けるため，周期的なリスタート法を導入することが提案されている[Goldberg 89b]．

本研究において，NDX－α と強制操作は突然変異の働きを持ち合わせている[Kamiya 95]ので，小集団でも初期収束の回避が期待できる．

ここで，三つの親解から三つの子解を生成するため，次のような交叉モデルを用いた．

a) 三つの親解をランダムに並べ，親解 1, 2, 3 とする．
b) 親解 1 と 2 を交叉し，親解 1 を中心に子解 1 を生成する．
c) 親解 2 と 3 を交叉し，親解 2 を中心に子解 2 を生成する．
d) 親解 3 と 1 を交叉し，親解 3 を中心に子解 3 を生成する．
e) 親解 i がエリート解“□”や境界強制対象解“■”なら，対応する子解 i を境界強制対

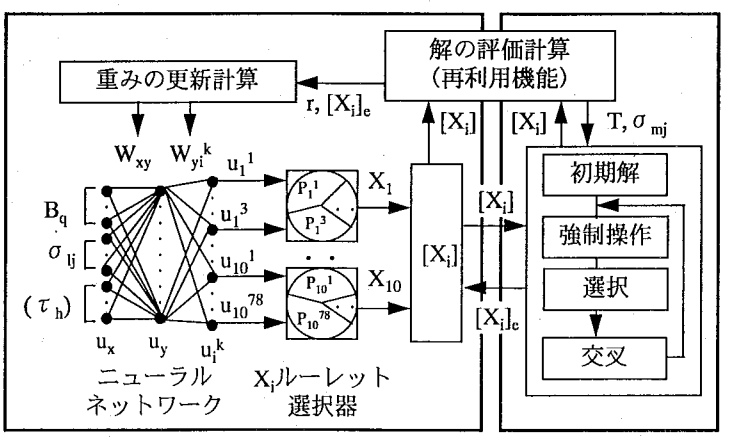
象解“■”とする.

f) 親解 i が緩和エリート解“▽”なら，対応する子解 i を緩和境界強制対象解“▼”とする.

5.4 強化学習の導入

強化学習とは，環境から感覚入力を受け，行動の候補となる競合集合から選択された行動を環境へ出力し，環境からの報酬を手掛かりに，環境に適応する機械学習である［山村 94］．本最適化問題において，制限熱応力 σ_{lj} と起動条件 τ_h を環境からの入力，起動スケジュールの解候補の集まり $\{X_i^k \mid i=1,...,10, k=1,...,s_i\}$ を環境への出力競合集合，起動スケジュールの評価結果によって，環境から報酬 r が与えられるとする（図 5.2(a)）．

［木村 96］は報酬の遅れのある問題及び汎化を必要とする問題に対して，ニューラルネットワークを用いた強化学習モデルを提案した．本研究は遅れのない問題であるが，非線形で，入出力パラメータが多次元であるような問題である．ここでは，下記のように[Williams 92]と[Rumelhart 86]を参考に学習信号に相当する characteristic eligibility を逆伝播する中間層を設け，さらに，学習の安定化を図るため，momentum 項[Rumelhart 86]を導入する.



(a) 強化学習　　(b) GA（図 5.1）

図 5.2: GA と強化学習の融合モデル

(1) 重みの更新

ニューラルネットワークの重みを $\mathbf{W}=[W_{xy}], [W_{yi}^{k}]$, x=1,...,6, y=1,...,14, i=1,...,10, k=1,...,s_i とすると，$\mathbf{W}$ の更新則は次式により与えられる．

$$\triangle \mathbf{W}(z)=\alpha'(r-b)\mathbf{D}(z)+\eta \triangle \mathbf{W}(z-1) \tag{5.5a}$$

$$\mathbf{D}(z)=\mathbf{e}(z)+\gamma\ \mathbf{D}(z-1) \tag{5.5b}$$

$$\mathbf{e}(z)=\nabla_{\mathbf{W}} \ln P_i^k(z) \tag{5.5c}$$

$$P_i^k(z)=\frac{\exp\left(u_i^k(z)\right)}{\sum_{n=1}^{s_i}\exp\left(u_i^n(z)\right)} \tag{5.5d}$$

$$[u_i^k(z)]=[W_{yi}^k(z)][u_y(z)] \tag{5.5e}$$

$$[u_y(z)]=[W_{xy}(z)][u_x(z)] \tag{5.5f}$$

$$[u_x(z)]=\left[\left[B_q(z)\right]^{tr}\left[\sigma_{lj}(z)\right]^{tr}\left[\tau_h(z)\right]^{tr}\right]^{tr} \tag{5.5g}$$

ここで，u_x は入力層ユニット，u_y, y=1,...,14 は中間層ユニット，u_i^k, i=1,...,10, k=1,...,s_i は出力層ユニット，z は第 z 探索ステップ，α'は学習率，r は報酬，b は reinforcement base line，η は学習安定係数，$\mathbf{D}=[D_{xy}]$, $[D_{yi}^k]$は報酬の遅れのある問題に対応する割引 characteristic eligibility，γ は割引率，$\mathbf{e}=[e_{xy}]$, $[e_{yi}^k]$は characteristic eligibility，P_i^k は起動スケジュールパラメータ X_i が値 X_i^k を取る確率を表わす．

入力層ユニット u_x は，バイアスユニット B_q, q=1,2，制限熱応力ユニット σ_{lj}, j=1,...,4 と起動条件ユニット τ_h, h=1,...,4 により構成されるが，本研究の現状では，タービンウォーム起動に限定して，τ_h を構成するタービンロータ温度等の条件を一定としているため，起動条件ユニットを 5.6 節の実験には導入していなく，u_x, x=1,...,6 とする．タービンウォーム起動モードとは，タービン起動時にタービンロータ温度がウォーム状態である．σ_{lj} は，スケール変換され，区間[1,2]内の値を取る．出力層ユニット u_i^k, i=1,...,10, k=1,...,s_i は，起動スケジュールパラメータ X_i の値 X_i^k, i=1,...,10, k=1,...,s_i に対応としているので，表 4.1 により，その数は $\sum_{i=1}^{10} s_i$ =197 である．

パラメータ調整結果により，α'=0.062, b=0, η=0.8, γ=0, B_q=1, q=1,2，中間層ユニット数=14，エリート解$[X_i]_e$が更新された時，正の報酬 r=1 与えられることとした．強化学習は正の報酬と負の報酬があるが，本問題においては，正の報酬を用いるべきであることを第 6 章「強化学習の報酬戦略に関する解析」で改めて論じることとする．

(2) 中間層～出力層の$[e_{yi}^{k}]$の計算

式(5.5c)～(5.5g)により，起動スケジュール$[X_i]=[X_i^a]$, i=1,...,10, a=1,...,s_iが選択されたとすると，

$$e_{yi}^{k}=\frac{\partial}{\partial W_{y_i}^{k}}\ln P_i^k=u_y(\delta_{k,a}-P_i^k) \tag{5.6}$$

となり，ただし，k=a なら$\delta_{k,a}=1$，k≠a なら$\delta_{k,a}=0$である．

(3) 入力層～中間層の$[e_{xy}]$の計算

式(5.5c)は，合成関数の微分定理(chain rules)により，

$$e_{xy}=\frac{\partial}{\partial W_{xy}}\ln P_i^k=\sum_{i=1}^{10}\left(\sum_{k=1}^{s_i}\left(\frac{\partial}{\partial X_i^k}\ln P_i^k\times\frac{\partial}{\partial W_{xy}}X_i^k\right)\right) \tag{5.7}$$

となり，さらに，式(5.5d)～(5.5g)を用いて，式(5.7)を展開すると，

$$e_{xy}=\frac{u_x}{u_y}\sum_{i=1}^{10}\sum_{k=1}^{s_i}W_{y_i}^{k}e_{y_i}^{k} \tag{5.8}$$

となり，学習信号に相当する characteristic eligibility は，出力層から入力層に逆伝播される．

5.5 GA と強化学習の融合

発電プラント運転操作に要求されるオンラインの最適解探索性能を満たすため，図 5.2 に示される枠組みで，GA と強化学習の融合を提案する．強化学習は探索と学習能力を持ち合わせているので，ここでは，GA と強化学習を融合し，最適解の探索と学習に適用することを図る．

5.5.1 融合モデルによる探索

探索空間サイズの大きい問題において，強化学習の加速を図るため，[Lin 93]は，人間の教示(teaching)により，強化学習を有望な領域で学習する提案を行った．本研究が対象としている問題領域では，探索過程のエリート解や探索の目標である最適解または近似最適解が未知のため，人間の教示による強化学習の加速ができない．本研究では，GA の教示によ

る強化学習の加速を提案する．GA が，強化学習を有望な領域で学習するように教示することにより，学習を加速することが期待できる．

GA と強化学習の融合による探索アルゴリズムについて，図 5.3 により説明する．図 5.3 において，探索は，重み及びエリート解と緩和エリート解の初期化から始まる（ステップ(1)と(2)）．エリート解または緩和エリート解の初期化とは，それぞれの解に対応する起動時間を十分大きい値に設定することである．強化学習は，与えられる重みに基づく計算により，解候補を生成する（ステップ(4)）．強化学習により，エリート解の更新があった場合，報酬が与えられ，学習が行われる（ステップ(9)）．強化学習により，エリート解の更

```
        /*探索開始*/
(1)     重み[W]の要素を±0.05 内にランダム生成;
(2)     エリート解と緩和エリート解を初期化;
(3)   ┌ (4)～(18)を 100 回繰り返して実行{
      │   /*強化学習による探索*/
(4)   │   強化学習は 1 回探索を行う;
      │   /*学習終了条件*/
(5)   │   if (探索された解が最適解または近似最適解)
(6)   │     go to (19);
(7)   │   else {
(8)   │     if (エリート解の更新があった)
      │       /*学習*/
(9)   │       報酬を与え，更新されたエリート解により重みを更新;
(10)  │     else {
      │       /*GA による探索*/
(11)  │       GA はエリート解更新まで最大 300 世代の探索を行う;
      │       /*学習終了条件*/
(12)  │       if (探索された解が最適解または近似最適解)
(13)  │         go to (19);
(14)  │       if (エリート解の更新があった)
      │         /*学習*/
(15)  │         報酬を与え，更新されたエリート解により重みを更新;
(16)  │     }
(17)  │   }
(18)  └ }
        /*探索終了*/
(19)    exit;
```

図 5.3: GA と強化学習の融合による探索アルゴリズム

新がなかった場合，GA は，強化学習により生成された解と保存されたエリート解と緩和エリート解を初期解として，エリート解の更新まで，探索を行う（ステップ(11)）．GA の探索の結果，エリート解の更新があった時，強化学習に報酬が与えられ，学習が行われるものとする（ステップ(15)）．最適解または近似最適解が探索される（ステップ(5)と(12)）まで，以上の探索と学習が繰り返して行われる．

5.5.2 融合モデルによる学習

GA と強化学習の融合による学習とは代表的な制限熱応力に対応する最適解を学習することである．ここで，学習の加速を図るため，「GA の教示による強化学習の加速」と「最適解の再現による学習の加速」を提案する．前者は上述のように，GA が強化学習を有望な領域で学習するように教示することにより学習を加速することである．後者について説明をする．従来，探索問題における強化学習アルゴリズムの一般的な枠組みは，

a) 初期状態をランダムまたはその他の定義により生成する．
b) 目標状態まで探索を行なう．
c) 探索途中または目標状態に到達した時点で報酬が与えられ，学習が行われる．
d) 学習が収束するまでにステップ a)から c)を繰り返す．

により構成される．しかし，多くの場合において，最適解が探索された時点でも，学習がまだ収束していないことが見られる．ここで，最適解の再現による学習とは，すべての最適解が探索された時点以降では，「ステップ b)目標状態まで探索を行なう」において，実際の探索を行なうことなく，最適解の再現により目標状態を達成することである．最適解の再現による学習を実現するための最適解の検出は，例えば，ステップ a)から d)の繰り返し回数が既定値以上になっても，今まで探索された最良解の更新がなかったことにより行なうことができる．最適解の再現による学習により，最適解が探索された以降では，ステップ b)の実際の探索を省くことにより，学習の加速が期待できる．

図 5.4 は，この最適解の再現による学習を導入した GA と強化学習の融合による学習アルゴリズムである．図 5.4 において，ステップ(6)は，強化学習により解候補を生成し，ステップ(7)は，過度適合(overfitting) [Stork 95]を回避する条件である．ステップ(6)の強化学習によって，確率的に生成された解候補が最適解であった場合，対応する学習用制限熱応力の学習が十分行われたとして，過度適合とならないため，その回に限って重みの更新をせず，次の制限熱応力の学習処理に移行する．

ステップ(9)～(12)は最適解の再現による学習に関する処理である．再現された最適解と与えられる報酬により，学習が行われる（ステップ(11)）．制限熱応力によって，最適解の探索が容易の場合と困難な場合がある．容易とは少ない探索ステップ数で最適解を獲得で

```
        /*学習開始*/
(1)     重み[W]の要素を±0.05 内にランダム生成;
(2)     各学習用制限熱応力 set のエリート解と緩和エリート解を初期化;
(3)   ┌(4)～(23)を 10,000 回繰り返して実行 {
(4)   │  (5)～(22)を各 set に対して，1 回ずつ実行 {
(5)   │    実行中の set を set x とし，強化学習の入力として与える;
      │    /*強化学習による探索*/
(6)   │    強化学習は set x について 1 回探索;
      │    /*過度適合回避条件*/
(7)   │    if (強化学習により探索された set x の解が最適解)
(8)   │      go to (4);
(9)   │    if (set x の最適解がすでに探索された) {
      │      /*学習の進度を合せる条件/*
(10)  │      if (全 set の最適解が探索された)
      │        /*最適解の再現による学習*/
(11)  │        報酬を与え，set x とその最適解により重みを更新;
(12)  │    }
(13)  │    else {
(14)  │      if (set x のエリート解[X_i]_e の更新があった)
      │        /*エリート解による学習*/
(15)  │        報酬を与え，set x と更新されたエリート解[X_i]_e により重みを更新;
(16)  │      else {
      │        /*GA による探索*/
(17)  │        GA は set x についてエリート解[X_i]_e 更新まで最大 300 世代の探索を行う;
(18)  │        if (set x のエリート解[X_i]_e の更新があった)
      │          /*エリート解による学習*/
(19)  │          報酬を与え，set x と更新されたエリート解[X_i]_e により重みを更新;
(20)  │      }
(21)  │    }
(22)  │  }
(23)  └}
(24)    重み[W]を保存;
        /*学習終了*/
(25)    exit;
```

図 5.4: GA と強化学習の融合による学習アルゴリズム

きることであり，困難とは多い探索ステップ数で最適解を獲得できることである．最適解が探索された時点で，該当する制限熱応力に対する最適解の再現による学習をすぐ開始とすると，学習が，その容易に最適解が探索された制限熱応力に特化され，その結果，最適解の探索が困難な制限熱応力に対する最適解探索がますます困難となり，全体の学習効率が低下してしまう．ここで，各学習用制限熱応力の学習進度を合せ，最適解が容易に探索できる制限熱応力に対する学習が突出して行われないため，スプップ(10)のように，最適解の再現による学習開始は，全制限熱応力の最適解が探索された時点以降とする．

強化学習の探索結果により，エリート解の更新があった場合，報酬が与えられ，更新されたエリート解により，学習が行われる（ステップ(15)）．強化学習の探索結果により，エリート解の更新がなかった場合，GA が探索を行い（ステップ(17)），その結果，エリート解の更新があった場合，強化学習に報酬が与えられ，学習が行われる（ステップ(19)）．

5.6 実験

5.6.1 実験の目的と方法

本研究は，探索の頑健性とオンライン探索性能の向上のため，以下のような戦略に関する提案を行なった．

a) 複数境界探索戦略による探索の頑健性の向上
b) 強化学習と GA の融合による学習と探索の加速
c) 最適解の再現による学習の加速

実験の目的は，これらの戦略の有効性を確認することである．これの戦略に関する実験を行うため，表 4.1 の制限熱応力の条件を増やして，表 5.1 に示される set 1〜set 11 まで合計 11 set の制限熱応力の条件を用いる．なお，表 5.1 において，set 0 は第 6 章の報酬の解析に関する実験に使われる制限熱応力であるが，ここでは，各制限熱応力に対する最適解の比較のため，set 0 の制限値もこの表に示す．表 5.1 の制限熱応力 set 2 と set 10 はそれぞれ表 4.1 の制限熱応力条件 1 と条件 2 に対応する．表 5.1 の set 1〜set 11 までの各制限熱応力値の間隔は一定であり，set 1 の制限値が緩く（制限値が高く），set 2〜set 11 の順に制限値が厳しく（制限値が低く）設定されている．set 0 は熱応力の制限のない条件であり，制限値が∞に相当する．表 5.1 に各 set の制限値に対して探索された最適解が示されている．これらの最適解は境界条件（定義 3.1b（離散の場合））を満足している．set 0 に対応する最

表5.1: タービンウォーム起動条件における各制限熱応力の最適起動スケジュール

$X_i \in \{X_i^k \mid k=1,....,s_i\}$, i=1,...,10

s_i は X_i のサイズを表わし，例えば，$s_1=3$, $s_{10}=78$ である．

X_i	起動スケジュール パラメータX_iの名称	レンジ		（値が大）← 制限熱応力set→ （値が小）					
		工学単位 (表現型)	整数値 (遺伝子型)	(set 0)	set 1	set2...8	set 9	set 10	set 11
				最適	最適	最適	最適	最適	最適
X_1	加速率(rpm/min.)	120,180,360	0,1,2	360	360	360	360	360	360
X_2	低速保持時間(min.)	0,1,...,28	0,1,...,28	0	0	0	0	0	0
X_3	高速保持時間(min.)	5,6,...,37	0,1,...,32	5	5	5	5	5	5
X_4	増負荷率 1 (%/min.)	0.5,1.0	0,1	1.0	1.0	1.0	1.0	1.0	1.0
X_5	初負荷保持時間(min.)	0,1,...,40	0,1,...,40	0	0	0	23	25	28
X_6	増負荷率 2 (%/min.)	0.5,1.0	0,1	1.0	0.5	1.0	1.0	1.0	1.0
X_7	増負荷率 3 (%/min.)	0.5,1.0	0,1	1.0	1.0	0.5	0.5	0.5	0.5
X_8	増負荷率 4 (%/min.)	1.0,1.5	0,1	1.5	1.0	1.0	1.0	1.0	1.0
X_9	増負荷率 5 (%/min.)	1.5,2.0,2.5,3.0	0,1,2,3	3.0	3.0	3.0	2.0	2.0	1.5
X_{10}	通気主蒸気温度(℃)	349,350,...,426	0,1,...,77	349	349	349	374	380	382
T	起動時間(min.)	---	---	189.9	210.6	211.6	252.8	257.1	269.3

適解は式(3.1d)を満足し，すなわちすべての起動スケジュールパラメータ X_i, i=1,...,10 は最短起動時間に等価な値をとり，$X_i=X_i^1$ or $X_i=X_i^{si}$ である．set 1～set 11 の最適解は式(3.1c)を満足し，すなわち何れかの起動スケジュールパラメータ X_i を等価起動時間の低い値に変化させた時，制限熱応力を超過する最大熱応力が存在する．

複数境界探索戦略を評価するため，複数戦略を導入した強化学習と GA の融合による探索モデルと複数戦略を導入しない強化学習とGAの融合による探索モデルを用いて実験し，比較を行なう．強化学習と GA の融合による学習と探索の加速に対しては，強化学習単独による探索モデルと強化学習と GA の融合による探索モデルを用いて，両者の実験比較により評価を行なう．最適解の再現による学習は，最適解の再現を導入した強化学習と GA による融合モデル及び最適解の再現を導入しないモデルを用いた実験により，再現機能の効果について評価する．

5.6.2 実験結果

表5.2は強化学習とGAの融合及び最適解再現戦略の導入による学習効率に関する実験結果であり，表 5.3 と図 5.5 は強化学習と GA の融合に関する探索効率に関する実験結果である．表 5.4 は学習を行なった後の各制限熱応力に対する実験結果である．表 5.5 は複数境界探索に関する実験結果である．

5.6.3 結果の考察

(1) GA と強化学習の融合による学習効率

表 5.2 は制限熱応力 set 2, 6, 10 を使った学習に関する実験結果である．表 5.2 において，No.1 は GA を導入しない，No.2 は GA を導入した強化学習である．同実験結果により，No.2 は，No.1 より，学習に必要なシミュレーション回数が約 1/10 低減する．GA と強化学習の融合は，GA が強化学習を有望な領域に学習を教示することにより，学習効率が大きく向上されることが確認された．

No.3 は，GA と強化学習の融合モデルに対して，最適解の再現による学習戦略を導入した学習モデルである．表 5.2 の実験結果に示されるように，最適解の再現による学習戦略の導入により，シミュレーション回数は No.3 が No.2 より約 73%減少し，本戦略が有効であることが言える．なお，最適解の再現による学習戦略の導入により，最適解による学習を行う頻度が多くなるため，学習が収束するまでの重み更新回数も減少し，193 回から 70 回となった．

No.4 は GA を使って，制限熱応力 set 2, 6, 10 のそれぞれの最適解に対する探索結果である．各制限熱応力に対する試行回数は 100 回である．同表には，各制限熱応力の最適解が探索されるまでのシミュレーション回数の合計が示されている．No.3 と No.4 は共に，最適解が探索されるまでのシミュレーション回数を表わしているが，No.3 の平均シミュレーション回数は No.4 より約 6%少ない．この結果により，GA と強化学習の融合は学習と探索が共に加速されることが確認され， 最適解の探索を例えば GA を使って先に行なってから，

表 5.2: GA と強化学習の融合による学習に関する実験結果

(1) 最適解の再現：最適解の再現による学習戦略

(2) 制限熱応力：set 2, 6 と 10

No	学習モデル	試行回数	シミュレーション回数				重み更新回数
			ave	var	max	min	
1	強化学習	2	60,006	18,795	73,297	46,716	245
2	GA＋強化学習	8	5,431	2,504	8,788	1,576	193
3	**GA＋強化学習＋最適解の再現**	**100**	**1,453**	**886**	**3,431**	**109**	**70**
4	GA	100	1,546	814	3,891	40	---

強化学習またはその他の学習モデル，例えば誤差逆伝播モデルのような教師付き学習を使って，学習を行なうよりも，探索と学習が同時に行なう方が効率的であることが示される．

(2) GA と強化学習の融合による探索効率

表 5.3 は，GA と強化学習の融合による探索効率に関する実験結果である．表 5.3 において，No. 1 の GA とは，GA 単独のモデル，No. 2 の GA＋強化学習とは，GA と強化学習を融合した探索モデルを表わす．実験結果により，No. 2 の GA＋強化学習は，No. 1 の GA より，平均シミュレーション回数が約 13%少ない．以下は，この理由に関する考察をまとめる［神谷 97b］．

a) GA の教示により，強化学習がむだな探索を省くことができる．
b) GA の教示により，学習が加速され，強化学習が有望な起動スケジュールパラメータを探索の序盤より推定することができる．
c) 強化学習により推定された有望な起動スケジュールパラメータは，GA に与えられ，GA は，有望な起動スケジュールパラメータを初期解として使うことにより，探索が加速される．

ここで，最後の c)について，もう少し説明を加える．強化学習は，エリート解が更新された時，エリート解による重み更新，すなわちエリート解による学習が行われる．一方，生成されるエリート解に高い頻度で出現する起動スケジュールパラメータ値 X_i^k は，最適解の構成要素，すなわち building block[Goldberg 89a]である可能性が高いと考えられる．エリート解による学習の進行に伴い，有望なパラメータ値は，高い確率で，強化学習によって生成される起動スケジュールに含まれ，building block が形成される．探索の進行に伴い，GA は，強化学習の学習効果により，探索の焦点を有望な領域（building block によって定義される領域）に移行し，また，強化学習は，GA の教示により，学習が加速される．このような，相乗効果により，GA と強化学習の融合は，効率のよい探索モデルを与えていると考えられる．

強化学習との融合による過去のエリート解情報の利用は，集団サイズの小さい GA に特に有効であると考えられる．集団サイズの大きい GA は，探索の進行に従い，building block が集団に蓄積され，集団の収束に伴い，集団内に蓄えられる情報により，有望な領域に探索を絞ることが期待できる．

最適解または近似最適解探索過程に生成されたエリート解による学習を行った強化学習がどの程度探索効率に寄与するかを確認するため，図 5.5 の実験結果を示す．GA と強化学

表 5.3: GA と強化学習の融合による学習に関する実験結果

(1) 試行回数：100

(2) 制限熱応力：set 10

No	探索モデル	シミュレーション回数			
		ave	var	max	min
1	GA	385	254	1,287	9
2	**GA+強化学習**	**334**	**245**	**1,276**	**12**

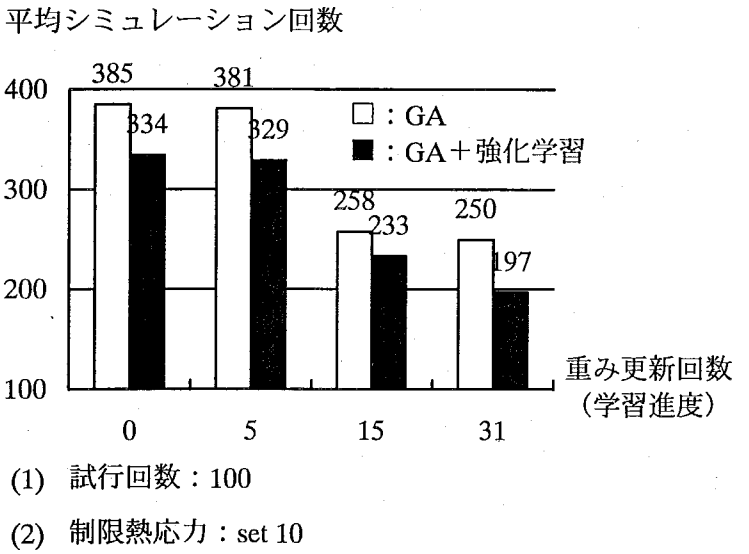


図 5.5: 探索効率と学習進度の関係

習の融合モデルによる最適解または近似最適解が探索されるまでの重み更新回数は，100 回試行に対して，最小 5，平均 15，最大 31 である．ここで，100 回試行結果から重み更新回数がちょうど 5, 15, 31 であるようなニューラルネットワークをランダムに一つずつ選択し，それぞれ net 5, net 15 と net 31 とする．図 5.5 において，GA とは，net 5, 15, 31 により，確率的に生成される解を初期解とした探索結果である．GA＋強化学習とは，net 5, 15, 31 を初期ニューラルネットワークとした探索結果である．図 5.5 の横軸の数字 5, 15, 31 は net 5, 15, 31 を用いた GA 及び GA＋強化学習に，0 は初期解をランダムに生成する GA 及び初期重みをランダムに生成する GA＋強化学習にそれぞれ対応している．図 5.5 の横軸の数字 0

に対応する実験結果は，表 5.3 から引用されていることに注意されたい．

図 5.5 の実験結果により，重み更新回数が多ければ，多いほど，GA 及び GA＋強化学習は，何れも，より少ないシミュレーション回数で最適解または近似最適解を探索できた．また，すべての重み更新回数の場合において，GA＋強化学習が GA よりよい探索効率が示されている．探索過程において，生成されたエリート解による強化学習は，有望な解を推定し，探索の加速に寄与していると考えられる．

GA と強化学習の融合による探索モデルは，探索と学習を同時に行うので，プラント起動過程の起動スケジュール再計算［鈴木 80］において，前回の起動スケジュール計算で行った探索の学習効果により，再計算の場合の運転状態が，前回計算で予測された運転状態とそれほど変化がなかった場合，最適解または近似最適解をより容易に探索できることが期待できる．すなわち，GA と強化学習の融合による探索モデルは，学習効果により，プラント起動過程に適応して，効率的な探索を行うことが期待できる．

なお，ここで，本論文の表 5.3 の GA＋強化学習による最適解または近似最適解が探索されるまでの平均シミュレーション回数は 334 であるのに対して，従来研究[Kamiya 96]では 361 であることを付記しておく．本論文では，ニューラルネットワークの重み計算において，学習の安定化を図るための momentum 項(η=0.8)（式(5.5a)）を導入している．その結果，本論文では大きい学習率(α'=0.062)に設定することが可能となり，学習が加速され，従来研究[Kamiya 96](α'=0.03)よりよい探索効率が得られている．

(3) 学習後の最適解探索結果

学習後の最適解または近似最適解の探索結果を表 5.4 に示す．同表において，set 2, 6, 10 は学習に使われた制限熱応力であり，他の set は，学習に使われなかった制限熱応力である．学習後では，学習効果とニューラルネットワークの汎化能力により，平均シミュレーション回数が 1 回～8 回となり，探索効率は著しく向上された．

(4) 複数境界探索戦略

表 5.5 は，学習前の複数境界探索戦略の導入に関する実験結果を示す．同表において，単一とは，一境界条件に対する強制操作とエリート保存戦略を導入するモデル，複数とは複数の境界条件に対する強制操作と複数エリート保存戦略を導入するモデルを表わすものである．複数境界条件に対する強制操作と複数エリート戦略の導入により，制限熱応力 set 8 に対する最適解または近似最適解探索の平均シミュレーション回数は，約 3,000 から約 80 回となり，大幅に低減することが示された．

表 5.4:　学習後の近似最適解探索結果

(1)　試行回数：100 回

(2)　学習用データ：set 2,6,10

制限熱応力	シミュレーション回数			
	ave	var	max	min
set 1	1	0	1	1
set 2	**1**	**0**	**1**	**1**
set 3	1	0	1	1
set 4	1	0	1	1
set 5	1	0	1	1
set 6	**1**	**0**	**1**	**1**
set 7	2	1	8	2
set 8	6	3	13	2
set 9	5	12	58	2
set 10	**1**	**0**	**1**	**1**
set 11	8	5	49	3

制限熱応力 set 8 によって，定義される解分布は凡そ図 5.6(a)の通りであることが見られる．複数エリート保存戦略の導入がなかった場合，境界または緩和境界に対する強制操作によって境界の外側近傍に有望な解“×”が生成されても，その有望な解“×”は，エリート解“●”により淘汰され，探索山の傾斜が急で，かつ山登りの道が狭くなり，探索が困難となる．一方，複数エリート保存戦略の導入があった場合，境界の外側であっても，緩和境界内の解“×”が生き残り，山登りの道が広くなり，探索が容易になる．

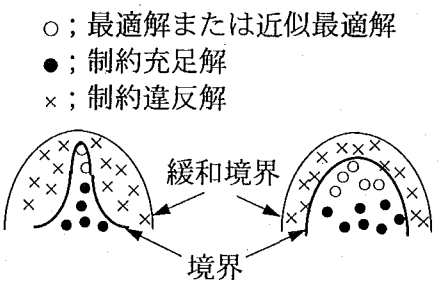
ところで，表 5.5 により，探索条件によっては，複数の境界条件に対する強制操作と複数エリート保存戦略の導入により，平均シミュレーション回数が set 2 では 26 から 30 回，また，set 10 では 334 から 418 回と増えた．制限熱応力 set 2,10 によって，定義される解分布は凡そ図 5.6(b)の通りであることが見られる．この場合，複数の境界条件に対する強制操作と複数エリート保存戦略の導入により，探索空間が広がった分だけ，探索効率が低下していると考えられる．しかし，制限熱応力 set 8 のような極端に探索の困難な山に対して，探索回数を大幅に減らすことがきるので，本提案戦略は探索の頑健性の向上に有効であるこ

表 5.5: 複数境界探索戦略の実験結果

(1) 探索モデル：GA と強化学習の融合

(2) sim：シミュレーション

境界	制限熱応力 set 2			**制限熱応力 set 8**			制限熱応力 set 10		
	試行回数	近似最適解回数	平均sim回数	試行回数	近似最適解回数	**平均sim回数**	試行回数	近似最適解回数	平均sim回数
単一	100	100	26	4	3	**3,043**	100	100	334
複数	100	100	30	100	100	**82**	100	100	418



(a) 制限熱応力：set 8　　(b) 制限熱応力：set 2,10

図 5.6: 境界条件により定義される解分布

とが言える.

なお，この制限熱応力 set 2, 8, 10 は，学習実験のために用意した 11 sets の制限熱応力の中で，探索が容易(set 2)，困難(set 8)，その中間(set 10)であるような制限熱応力である．ここでは，この三つを代表として実験を行った.

5.7 おわりに

本章は，火力発電プラント起動スケジューリング問題において，GA と強化学習の融合に

よる学習と探索モデル及び複数境界探索戦略の提案とその有効性に関する考察を行った．GA は，強化学習を有望な領域で学習するように教示することによって，強化学習の学習を加速する．強化学習は，building block が含まれる確率が比較的に高いエリート解を用いた学習を行うことにより，探索の進行に伴い，building block を形成し，GA の探索を加速する．このような GA と強化学習の相乗効果により，探索と学習が同時に加速される．GA と強化学習の融合による探索モデルは，学習前の状態から探索を開始しても，探索過程のエリート解による学習効果により，強化学習を導入しない GA より，13%の探索効率の向上が確認された．最適解の再現による学習戦略の導入により，学習効率が大きく向上され，学習に必要な CPU 時間は，SPARC station 20 上では，約 16 時間から約 24 分となった．また，学習効果により，最適解または近時最適解探索に必要な CPU 時間は，SPARC station 20 上で，制限熱応力によっては，平均 1 ないし 8 秒，最大 1 ないし 58 秒となった．プラント運転に要求されるオンライン探索性能，すなわち 30 秒という研究の当初目標をほぼ達成できた．

本問題の最適解は境界の内側近傍に存在するが，頑健性のある探索モデルを構築するため，境界の内側に限定することなく，外側も積極的に探索することを図った．ここで，境界の外側近傍に，緩和境界を設け，第 4 章で提案した一つの境界に対する強制操作を拡張して，複数境界に対する強制操作，及び従来の一種類の解だけを保存するエリート保存戦略を拡張して，境界の外側近傍にある制約違反解も第 2 種類のエリート解として保存する複数エリート保存戦略を導入することにより，頑健性のある探索モデルを構築することができ，実験結果によりその有効性が確認された．

6 強化学習の報酬戦略に関する解析

6.1 はじめに

第 5 章では，強化学習の報酬の授与方法として，エリート解が更新された時，正の報酬 r=1 が与えられる（5.4(1)項を参照）．パラメータ調整結果により，本発電プラント起動スケジュールング問題において，このような正の報酬戦略がもっとも学習効率がよいと確認されている．

一方，[Barto 83]のポールバランス問題や[Lin 93]のロボットのドッキング問題の強化学習において，それぞれ負の報酬戦略，または正+負の報酬戦略が用いられて，良好な学習結果が得られている．[Barto 83]と[Lin 93]の問題において，強化学習器が失敗した行動出力をした時，負の報酬が与えられる．学習過程において，学習が最適出力に収束していない時，強化学習器が成功する行動出力よりも失敗した行動出力の回数が多いので，失敗した行動出力に対して，負の報酬を与えることにより，強化学習器が学習を行う機会が増え，学習が加速されることが期待できる．

[Barto 83]と[Lin 93]の問題における強化学習の出力次元は 1 で，本発電プラント起動スケジュールング問題における強化学習の出力次元は 10 である．ここで，本章では，報酬戦略に関する解析を行い，その結果を本発電プラント起動スケジュールング問題に対する報酬戦略の設計に適用する．なお，本章の解析結果により，強化学習の出力が低次元の場合，負の報酬または正の報酬により，学習が最適解に収束するが，高次元の場合，負の報酬により学習が最適解に収束しないか，学習が不安定になることが示され，高次元の問題に対して，強化学習を用いた場合，正の報酬を与えることを提案する．

6.2 定義

6.2.1 出力空間

強化学習は，与えられる報酬により最適入力－出力写像(optimal input-output mapping)を学習するものである．ここで，図 6.1 はある入力[Y]に対応する強化学習の出力空間を表わす．同図において，強化学習の入力－出力空間に関する記号と用語を下記のように定義する．

a) [Y]：入力

b) $(X_1[Y],...,X_i[Y],...,X_n[Y])$：出力

c) $X_i[Y] \in (X_1[Y],...,X_i[Y],...,X_n[Y])$：出力要素

d) $X_i^k[Y] \in X_i[Y]$, i=1,...,n, k=1,...,s：出力ドメイン

e) “n”：出力の次元数

f) “s”：各出力次元における出力ドメインサイズ

g) $(X_1^*[Y],...,X_i^*[Y],...,X_n^*[Y]) \subset (X_1[Y],...,X_i[Y],...,X_n[Y])$：最適出力

h) $X_i^*[Y] \in (X_1^*[Y],...,X_i^*[Y],...,X_n^*[Y])$：最適出力要素

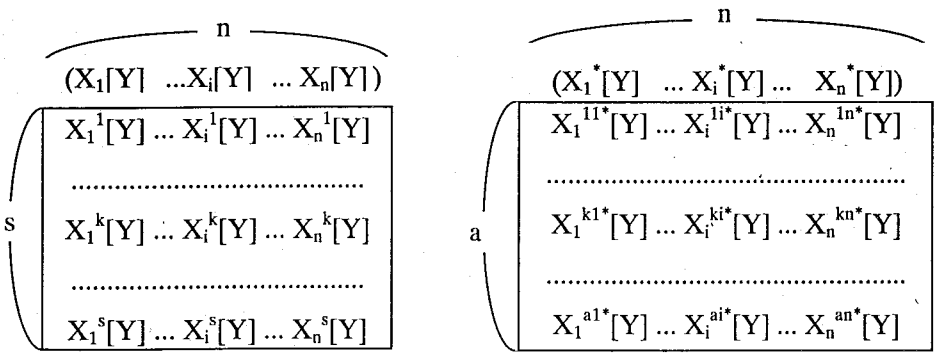
i) $X_i^{ki*}[Y] \in X_i^*[Y]$, i=1,...,n, k_i=1,...,a_i：最適出力ドメイン

j) “a”, where a＜s：各出力次元における最適出力ドメインサイズ

多次元問題は，各出力状態に対して，それぞれ一つの出力ドメインを割り当てることにより，1 次元問題として表現することができる．しかし，このように表現された 1 次元問題の出力ドメインサイズが実用的に大きくなりすぎることがある．例えば，n=10，s=20 のような問題に対して，10 次元問題表現では，合計出力ドメインサイズが n×s=200 であるが，1 次元問題表現とした時，出力ドメインサイズが $n^s=20^{10}$ となる．従って，強化学習は多次元表現問題にも応用できるように考慮される必要がある．

6.2.2 最適出力部分空間

最適出力部分空間は出力全空間を構成する部分空間である．ここで，図 6.1(b)に示される



(a)出力全空間　　(b) 最適出力部分空間

注：s＞a≧1, n≧1

図 6.1: ある入力[Y]に対応する強化学習の出力空間

ように，ある入力状態に対して，複数の最適出力が許容されるものとする．この場合，ある入力[Y]に対して，各最適出力要素 $X_i^*[Y]$, i=1,...,n に含まれる最適出力ドメイン $X_i^{ki*}[Y]$の数は同じであるとして，“a”とする．最適出力($X_1^*[Y]$,...,$X_i^*[Y]$,...,$X_n^*[Y]$)は，各出力要素に含まれる任意な最適出力ドメイン $X_i^{ki*}[Y]$の組合わせによって構成されるとする．従って，ある入力[Y]に対する最適出力の数は“a^n”である．ここで，比“$\frac{a}{s}$, $1 \leqq a < s$”は，強化学習によって生成される最適出力の許容誤差を表わすものとして考えることができる．多くの最適化問題では，a=1 であり，すなわちある入力状態に対する最適出力は一通りである．

6.2.3 学習環境と学習アルゴリズム

報酬を手掛かりに最適入力－出力写像を学習する強化学習アルゴリズムは，報酬の遅れのない場合[Williams 92]とある場合[Barto 83][木村 96]によって，多く提案されているが，ここでは，多次元出力空間における強化学習の報酬戦略を解析するため，次のような「理想的な学習環境」と「学習アルゴリズム」を定義する．

(1) 学習環境

強化学習は，与えられる各入力に対する入力－出力写像を行い，出力を決定する．この場合，各入力－出力ペアに与えられる報酬は，次のように a)一貫性と b)マルコフ性が保たれ，さらに学習は，c)充分な入力提示回数が確保されるものと仮定する．

a) **報酬の一貫性の仮定**　正の報酬戦略を用いた場合，最適入力－出力ペアは常に正の報酬が与えられ，非最適入力－出力ペアには報酬が与えられない．負の報酬戦略を用いた場合，非最適入力－出力ペアは常に負の報酬が与えられ，最適入力－出力ペアには報酬が与えられない．正+負の報酬戦略を用いた場合，最適入力－出力ペアは常に正の報酬が与えられ，非最適入力－出力ペアには常に負の報酬が与えられる．

b) **報酬のマルコフ性の仮定**　報酬の授与は，現在の入力－出力ペアのみに依存して，過去のペアに依存しない．

c) **充分な入力提示回数の仮定**　各入力状態は，学習に必要な充分な回数だけ学習器に繰り返し提示される．

報酬の遅れのない問題では，強化学習に与えられる報酬は，通常，仮定 b)マルコフ性を持つものであり，すなわち報酬は現状の入力－出力ペアにのみ依存する．報酬の遅れのある問題では，強化学習に与えられる報酬は過去の入力－出力ペアにも依存する．しかし，

ここでは，報酬の遅れのある問題においても，理想的な学習環境を仮定し，入力－出力ペアが生成された時，そのペアが最適かどうかが判定できて，その時点でそのペアに対して報酬が与えられるものとする．

(2) 学習アルゴリズム

a) 確率パラメータ $u_i^k[Y]$　強化学習アルゴリズムは，探索のランダム性が伴われる必要がある[Williams 92]．ここで，ランダム性を伴った入力－出力写像が可能とするため，さらに，与えられる報酬を基本とした最適入力－出力ペアの学習が可能とするため，各入力－出力ドメインペア $X_i^k[Y]$に対して，独立に一つの確率パラメータ $u_i^k[Y]$を割り当てる．このように割り当てられた確率パラメータ $u_i^k[Y]$の数は“n×s×入力状態数”である．学習の開始時，すべての確率パラメータ $u_i^k[Y]$に対して，同じ値を設定するものとし，例えば，$\forall(Y)\ u_i^k[Y]=0,\ i=1,...,n;\ k=1,...,a$ である．ある入力－出力ペア[Y]－$(X_1,...,X_i,...,X_n)$に報酬 r が与えられると，その報酬 r はその入力－出力ペア[Y]－$(X_1,...,X_i,...,X_n)$に属するすべての確率パラメータ $u_i^k[Y]$に伝播され，これらの確率パラメータ $u_i^k[Y]$は，伝播される報酬 r によって次式のように更新される．

$$u_i^k[Y]\leftarrow u_i^k[Y]+r \tag{6.1}$$

ここで，記号“←”は代入を表わす．

b) 出力ドメインの選択確率 $P(X_i=X_i^k|\ u_i^k[Y])$　入力[Y]と対応する確率パラメータ $u_i^k[Y]$が与えられた時，出力ドメイン X_i^kが出力要素 X_i として選択される確率 $P(X_i=X_i^k|\ u_i^k[Y])$は，下記のようなロジック関数によって与えられるものとする．

$$P(X_i=X_i^k|\ u_i^k[Y])=\frac{\exp\left(u_i^k[Y]\right)}{\sum_{j=1}^{s}\exp\left(u_i^j[Y]\right)} \tag{6.2}$$

従って，以上の仮定により，学習の開始時点では，ある入力[Y]が与えられた時，対応すべての出力ドメイン X_i^kは，ランダムにある出力要素 X_i として選択されることになる．学習の進行に伴い，同じ出力要素 X_iの中で，それまでの学習過程の中でより多くの報酬 r（より多い正の報酬，より少ない負の報酬）を貰えた出力ドメイン X_i^kは，その出力要素 X_i として選択される確率が高くなる．

6.2.4 報酬戦略

(1) 初期設定$\forall(Y)\ u_i^k[Y]=0,\ i=1,...,n;\ k=1,...,s$

(2) [Y]を入力する.

(3) 式(6.2)により出力$(X_1,...,X_i,...,X_n)$を生成する.

(4) 報酬 r を受け取る.

(5) 式(6.1)により確率パラメータ $u_i^k[Y]$を更新する.

(6) 終了条件が成立するまで，(2)～(5)を繰り返す.

図 6.2: 強化学習アルゴリズム

図 6.2 は，以上の定義に基づいて構築された強化学習アルゴリズムである．同図において，報酬 r（step(4)）は次の三つの戦略によって与えられるものとする．

戦略 1（正の報酬戦略）

最適入力－出力ペアなら，正の報酬 $r=r^+>0$，それ以外の時，r=0 を与える．

戦略 2（負の報酬戦略）

非最適入力－出力ペアなら，負の報酬 $r=r^-<0$，それ以外の時，r=0 を与える．

戦略 3（正+負の報酬戦略）

最適入力－出力ペアなら，正の報酬 $r=r^+>0$，非最適入力－出力ペアなら，負の報酬 $r=r^-<0$ を与える．

6.3 報酬戦略に関する解析

6.2 節の定義により，以下が導かれる．

(1) 入力[Y]の省略

報酬の一貫性とマルコフ性及び充分な入力提示回数という理想的な学習環境（6.2.3(1)項）の仮定により，出力ドメインの選択確率 $P(X_i=X_i^k|\ u_i^k[Y])$に関する解析は，各入力状態

に対して独立行うことができる．従って，解析のための一般性を失うことなく，“入力”や記号“Y”の陽的な記述を省略できる．以下では，簡単のため，例えば，最適入力－出力ペアを単に最適出力，$X_i^*[Y]$を単にX_i^*と表現することとする．

(2) 最適出力ドメインと非最適出力ドメインの生成期待確率

a) 各確率パラメータu_i^kの初期値が同様に0と設定されて，かつb) 出力に与えられる報酬rは一様にその出力に含まれる出力ドメインに伝播され，対応する確率パラメータu_i^kが一度に更新されるので，学習過程において，各最適出力ドメインの生成確率の期待値と各非最適出力ドメインの生成確率の期待値は，それぞれ同じであると仮定できる．ここで，次のような記号を付加する．

a) $\langle u^+ \rangle$：各最適確率パラメータの期待値$\langle u_i^k \rangle = \langle u^+ \rangle$, $i=1,...,n$; $k \in \{k_i^* | k_i = 1_i,...,a_i\}$

b) $\langle u^- \rangle$：各非最適確率パラメータの期待値$\langle u_i^k \rangle = \langle u^- \rangle$, $i=1,...,n$; $k \notin \{k_i^* | k_i = 1_i,...,a_i\}$

c) $\langle P^* \rangle$：ある出力要素に任意な最適出力ドメインが出現する期待確率

d) $\dfrac{\langle P^* \rangle}{a}$：ある出力要素にある特定な最適出力ドメインが出現する期待確率

e) $\langle 1-P^* \rangle$：ある出力要素に任意な非最適出力ドメインが出現する期待確率

f) $\dfrac{\langle 1-P^* \rangle}{s-a}$：ある出力要素にある特定な非最適出力ドメインが出現する期待確率

(3) 解析結果のまとめ

6.3.1節以降の解析結果を，次のようにまとめることができる．

結果1（正の報酬戦略）

強化学習の出力次元に関係なく，最適出力ドメインの期待生成確率$\langle P^* \rangle$は，学習の進行に伴い，1に収束する．

結果2（負の報酬戦略で，低出力次元の場合：$n \leqq \dfrac{s}{s-a}$）

最適出力ドメインの期待生成確率$\langle P^* \rangle$は，学習の進行に伴い，1に収束する．

結果3（負の報酬戦略で，高出力次元の場合：$n > \dfrac{s}{s-a}$）

最適出力ドメインの期待生成確率$\langle P^* \rangle$は，学習の進行に伴い，P_bに収束するか，学習が不安定になる．ただし，P_bは下式により与えられる．

$$\frac{a}{s} < P_{b(lower)} < P_b < P_{b(upper)} < 1 \tag{6.3a}$$

$$P_{b(lower)} = g + \frac{g^n}{\frac{1}{1-g} - n \times g^{n-1}} \tag{6.3b}$$

$$P_{b(upper)} = g + \frac{g^n}{\frac{1}{1-g} - \frac{h^n - g^n}{h-g}} \tag{6.3c}$$

$$g = \frac{a}{s} \tag{6.3d}$$

$$h = \left(\frac{1}{n} \times \frac{s}{s-a}\right)^{\frac{1}{n-1}} \tag{6.3e}$$

多くの応用問題において，s と比べて a が小さく，すなわち $a \ll s$ である．$a \leqq 0.5s$，または $n \gg 1$ である場合，P_b は，

$$P_{b(lower)} \fallingdotseq P_b \fallingdotseq P_{b(upper)} \fallingdotseq \frac{a}{s} \tag{6.4}$$

となる．学習開始時点では，式(6.1)と(6.2)により最適出力ドメインの生成確率は $\frac{a}{s}$ である．従って，この結果により，負の報酬戦略で，強化学習の出力次元が高い場合，多くの応用問題において，学習によって最適出力ドメインの生成確率が改善ができなく，最適出力はランダムにしか生成されないことを意味する．

結果 4（正+負の報酬戦略で，低出力次元の場合：$n \leqq \frac{s}{s-a}$）

$\frac{r^+}{|r^-|}$ の大きさに関係なく，最適出力ドメインの期待生成確率<P^*>は，学習の進行に伴い，1 に収束する．ただし，$r^+ > 0$ と $r^- < 0$ はそれぞれ正の報酬と負の報酬である．

結果 5（正+負の報酬戦略で，高出力次元の場合：$n > \frac{s}{s-a}$）

$\frac{r^+}{|r^-|}$ は，

$$\frac{r^+}{|r^-|} > L \tag{6.5a}$$

$$L = \frac{1}{n} \times \frac{s}{s-a} \times \left(\frac{s}{a} \times \frac{n-1}{n} \right)^{n-1} - 1 \tag{6.5b}$$

である時，最適出力ドメインの期待生成確率<P^*>は，学習の進行に伴い，1 に収束するが，$\frac{r^+}{|r^-|} \leqq L$ である時は，最適出力ドメインの期待生成確率<P^*>は，学習の進行に伴い，P_b に収束するか，学習が不安定になる．ただし，$r^+ > 0$ と $r^- < 0$ はそれぞれ正の報酬と負の報酬で，P_b は以下の式により与えられる．

$$\frac{a}{s} < P_{b(lower)} < P_b < P_{b(upper)} < 1 \tag{6.6a}$$

$$P_{b(lower)} = g + \frac{f \times g^n}{\frac{1}{1-g} - n \times f \times g^{n-1}} \tag{6.6b}$$

$$P_{b(upper)} = g + \frac{f \times g^n}{\frac{1}{1-g} - \frac{h^n - f^{\frac{n}{n-1}} \times g^n}{h - f^{\frac{1}{n-1}} \times g}} \tag{6.6c}$$

$$f = 1 + \frac{r^+}{|r^-|} \tag{6.6d}$$

$$g = \frac{a}{s} \tag{6.3d}$$

$$h = \left(\frac{1}{n} \times \frac{s}{s-a} \right)^{\frac{1}{n-1}} \tag{6.3e}$$

式(6.5b)により，n が大きい場合，$\frac{a}{s}$ が小さいほど，L が指数的に増大する．多くの強化学習アルゴリズムにおいて，正の報酬は負の報酬より極端に大きく，すなわち $r^+ \gg |r^-|$ とすることができない．$r^+ \gg |r^-|$ の場合，a) 学習過程において，負の報酬が有効でないか，または，b) 負の報酬が有効である場合，r^+ が大きいため，学習が不安定になる．また，多くの応用問題において，s と比べて a が小さく，すなわち $a \ll s$ である．$a \leqq 0.5s$，または $n \gg 1$ である場合，P_b は，

$$P_{b(lower)} \fallingdotseq P_b \fallingdotseq P_{b(upper)} \fallingdotseq \frac{a}{s} \tag{6.4}$$

となる．この結果により，正+負の報酬戦略で，強化学習の出力次元が高い場合，上記の結果 3（負の報酬戦略）と同様に，多くの応用問題において，学習は最適出力ドメインの生成確率を改善しなく，最適出力はランダムにしか生成されないことを意味する．

以上の結果 1 から 5 までにより，強化学習の出力が高次元(n≫1)である場合，正の報酬戦略だけが有効であることが言える．

6.3.1 正の報酬戦略

この戦略では，最適出力に対して正の報酬が与えられ，非最適出力に対しては報酬が与えられない．図 6.2 の学習アルゴリズムの第 z 回繰返しにおける$<u^+(z)>$，$<u^-(z)>$と$<P^*(z)>$は次式により与えられる．

$$<u^+(z+1)>=<u^+(z)>+\frac{\langle P^*(z)\rangle^n}{a}\times r^+,\ \text{ただし，}\ u^+(1)=0,\ r^+>0 \tag{6.7a}$$

$$<u^-(z+1)>=<u^-(z)>,\ \text{ただし，}\ u^-(1)=0 \tag{6.7b}$$

$$<P^*(z)>=\frac{a\times\exp\langle u^+(z)\rangle}{(s-a)\times\exp\langle u^-(z)\rangle+a\times\exp\langle u^+(z)\rangle} \tag{6.7c}$$

ここで，$<P^*(z)>^n$は第 z 回繰返しにおける最適出力の期待生成確率，$\frac{\langle P^*(z)\rangle^n}{a}$は第 z 回繰返しにおけるある特定な最適出力ドメインに正の報酬 r^+が与えられる期待確率を表わす．$<P^*(1)>=\frac{a}{s}$であるため，z が∞に近づく時，$<u^+(z)>$が∞に，$<P^*(z)>$が 1 に近づく．結果 1 が得られる．

6.3.2 負の報酬戦略

この戦略では，非最適出力に対して負の報酬が与えられ，最適出力に対しては報酬が与えられない．しかし，この場合，非最適出力に最適出力ドメインが含まれることがあるため，最適出力ドメインは負の報酬を受け取ることがある．ここで，$<R_o^->$をある特定な最適出力ドメインに与えられる負の報酬の期待値，$<R_n^->$をある特定な非最適出力ドメインに与

えられる負の報酬の期待値とすると，$<R_o^->$と$<R_n^->$は次式により与えられる．

$$<R_o^->=\frac{\langle P^*\rangle\left(1-\langle P^*\rangle^{n-1}\right)}{a}\times r^- \tag{6.8a}$$

$$<R_n^->=\frac{1-\langle P^*\rangle}{s-a}\times r^- \tag{6.8b}$$

ただし，$r^-<0$．ここで，$<\triangle R>$を最適出力ドメインと非最適出力ドメインに与えられる期待報酬の差とすると，式(6.8a)と(6.8b)より，$<\triangle R>$は，次式により与えられる．

$$<\triangle R>=<R_o^->-<R_n^->=\frac{1}{a}\times\left\{\langle P^*\rangle^n-\frac{s}{s-a}\times\langle P^*\rangle+\frac{a}{s-a}\right\} \tag{6.9}$$

ただし，$r^-=-1$．図 6.3 は，a=1 とし，$<P^*>$を 0～1 に変化させた時に対する$<-R_o^->$ (n=1,2,3,5,10)と$<-R_n^->$ (s=2,3,5,10)の変化を示す．同図により，a=1 の場合，$<\triangle R>$は下記となる．

ケース 1： if “n=1” or “n=2 and s=2”,

then $<\triangle R>\succ 0$ for $<P^*>=[0, 1)$ (6.10)

ケース 2： if “n≧3” or “n=2 and s≧3”,

then $<\triangle R>\succ 0$ for $<P^*>=[0, P_b)$; $<\triangle R>=0$ at $<P^*>=P_b$; $<\triangle R>\prec 0$ for $<P^*>=(P_b, 1)$ (6.11)

ここで，$P_b=(0, 1)$は，式(6.12)で定義され，記号“(,)”は開空間，“[,]”は閉空間を表わす．

$$P_b\equiv<P^*>\Big|_{<\Delta R>=0} \tag{6.12}$$

式(6.1)と(6.2)に示されるように，出力ドメインX_i^kに与えられる負の報酬r^-が多ければ，多いほど，対応する出力ドメインX_i^kの生成確率P_i^kが減少するので，上記の二つのケースに対して，以下が得られる．

a) ケース 1 では，$<P^*>=[0,1)$において，最適出力ドメインよりも，非最適出力ドメインに与えられる負の報酬が多いので，学習の進行に伴い，最適出力ドメインの生成期待確率$<P^*>$は増加し，初期値$\frac{a}{s}$から 1 まで近づくことが期待できる．従って，負の報酬戦

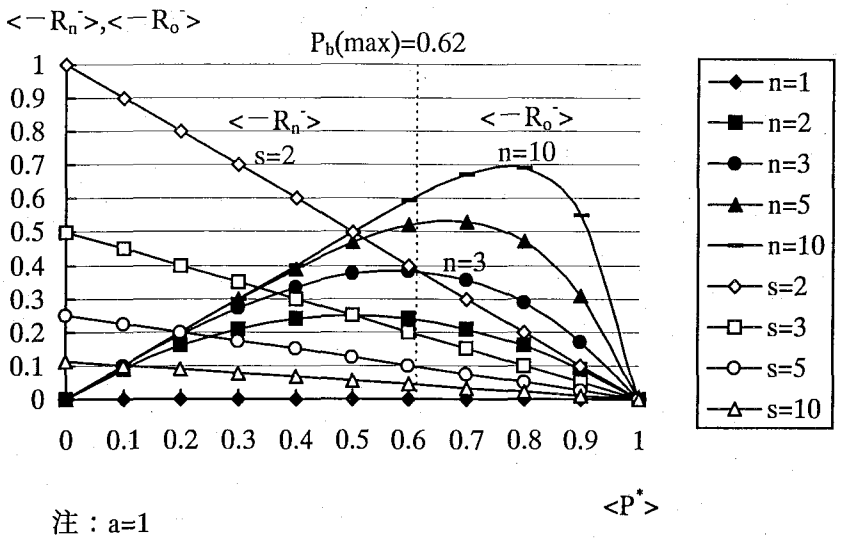


図 6.3: 負の報酬戦略による最適出力ドメインと非最適出力ドメインに与えられる負の報酬期待値曲線

略により，強化学習は最適出力に収束することが期待できる．

b) ケース 2 では，$\langle P^* \rangle=[0,P_b)$において，最適出力ドメインよりも，非最適出力ドメインに与えられる負の報酬が多いので，この区間では，最適出力ドメイン生成期待確率$\langle P^* \rangle$は，学習の進行に伴い増加する．$\langle P^* \rangle=(P_b,1)$において，最適出力ドメインよりも，非最適出力ドメインに与えられる負の報酬が少ないので，この区間では，最適出力ドメイン生成期待確率$\langle P^* \rangle$は，学習の進行に伴い減少する．従って，このケースでは，学習の進行に伴い，最適出力ドメイン生成期待確率$\langle P^* \rangle$は，P_b に収束するか，P_b を中心に振動し，学習が不安定になる．さらに，図 6.3 により，P_b の最大値 $P_b(max)=0.62$ である．

以下の定理では，$s>a\geqq 1$ の場合と P_b の下界と上界を示す．

定理 6.1 （負の報酬戦略）

式(6.9)により，

a) $n \leqq \dfrac{s}{s-a}$ なら，$\langle P^* \rangle=[0, 1)$では$\langle R \rangle>0$，$\langle P^* \rangle=1$ では$\langle R \rangle=0$ が成立する．

b) $n > \frac{s}{s-a}$ なら，$\langle P^* \rangle = [0, P_b)$では$\langle R \rangle > 0$，$\langle P^* \rangle = P_b$では$\langle R \rangle = 0$，$\langle P^* \rangle = (P_b, 1)$では$\langle R \rangle < 0$，$\langle P^* \rangle = 1$では$\langle R \rangle = 0$が成立する．ここで，$P_b$は式(6.3)により与えられる．

証明：

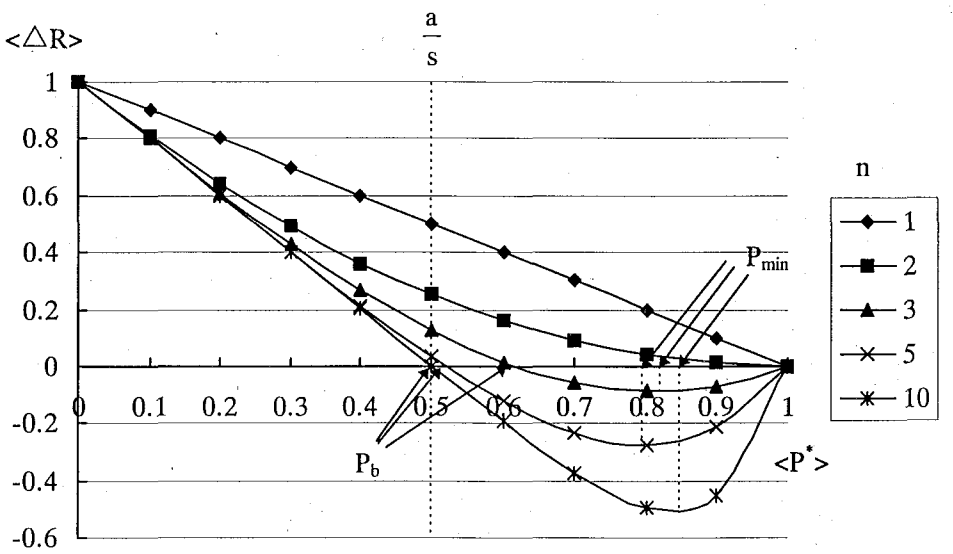
定理証明の参考のため，図 6.4 は，s=2, a=1, n=1,2,3,5,10，及び$\langle P^* \rangle$を 0 から 1 に変化させた時の$\langle \Delta R \rangle$（式(6.9)）の変化を示す．

式(6.9)により，$\langle P^* \rangle = 1$ または n=1 なら，定理 6.1 が成立することが明らかである．ここで，$n \geqq 2$ と$\langle P^* \rangle = [0, 1)$について考える．

式(6.9)により，以下の式が得られる．

$$\frac{d\langle \Delta R \rangle}{d\langle P^* \rangle} = \frac{1}{a} \times \left(n \times \langle P^* \rangle^{n-1} - \frac{s}{s-a} \right) \tag{6.13a}$$

$$\frac{d^2\langle \Delta R \rangle}{d\langle P^* \rangle^2} = \frac{n(n-1)\langle P^* \rangle^{n-2}}{a} \tag{6.13b}$$



注：s=2; a=1

図 6.4: 負の報酬戦略による最適出力ドメインと非最適出力ドメインに与えられる期待報酬の差

$n \geqq 2$，$a \geqq 1$ と$\langle P^* \rangle \geqq 0$ では$\dfrac{d^2\langle \Delta R \rangle}{d\langle P^* \rangle^2} \geqq 0$ であるので，$\langle R \rangle$は$\langle P^* \rangle \geqq 0$ に対して凹関数である．ここで，式(6.9)と(6.13a)により，$\langle P^* \rangle = 0, \dfrac{a}{s}, 1$ に対応する$\langle \triangle R \rangle$と$\dfrac{d\langle \Delta R \rangle}{d\langle P^* \rangle}$のそれぞれの式を表 6.1 に示す．表 6.1 により，$n \leqq \dfrac{s}{s-a}$ の場合，$\left.\dfrac{d\langle \Delta R \rangle}{d\langle P^* \rangle}\right|_{\langle P^* \rangle = 1} \leqq 0$ である．さらに，$\langle P^* \rangle > 0$ に対して$\langle R \rangle$は凹関数であり，$\langle P^* \rangle = 1$ に対して$\langle R \rangle = 0$ であるため，$\langle P^* \rangle = [0, 1)$に対して$\langle R \rangle > 0$ となる．定理 6.1a)が証明された．

以下は，$n > \dfrac{s}{s-a}$ について考える．$n > \dfrac{s}{s-a}$ の場合，

$$\left.\frac{d\langle \Delta R \rangle}{d\langle P^* \rangle}\right|_{\langle P^* \rangle = \frac{a}{s}} < 0 \tag{6.14}$$

$$\left.\frac{d\langle \Delta R \rangle}{d\langle P^* \rangle}\right|_{\langle P^* \rangle = 1} > 0 \tag{6.15}$$

が成立する．表 6.1 により，式(6.15)が成立することが明らかであるが，ここでは，式(6.14)の成立について示す．

表 6.1 により，$\dfrac{a}{s} < 1$ のため，$n = \dfrac{s}{s-a}$ の場合，

表 6.1: 負の報酬戦略の場合の$\langle P^* \rangle = 0, \dfrac{a}{s}, 1$ に対応する$\langle \triangle R \rangle$と$\dfrac{d\langle \Delta R \rangle}{d\langle P^* \rangle}$の式

	$\langle P^* \rangle = 0$	$\langle P^* \rangle = \dfrac{a}{s}$	$\langle P^* \rangle = 1$
$\langle \triangle R \rangle$	$\dfrac{1}{s-a}$	$\dfrac{1}{a} \times \left(\dfrac{a}{s}\right)^n$	0
$\dfrac{d\langle \Delta R \rangle}{d\langle P^* \rangle}$	$-\dfrac{1}{a} \times \dfrac{s}{s-a}$	$\dfrac{1}{a} \times \left(n \times \left(\dfrac{a}{s}\right)^{n-1} - \dfrac{s}{s-a}\right)$	$\dfrac{1}{a}\left(n - \dfrac{s}{s-a}\right)$

$$\left.\frac{d\langle\Delta R\rangle}{d\langle P^*\rangle}\right|_{\langle P^*\rangle=\frac{a}{s}}=\frac{1}{a}\times\left(n\times\left(\frac{a}{s}\right)^{n-1}-\frac{s}{s-a}\right)=\frac{1}{a}\times\frac{s}{s-a}\times\left(\left(\frac{a}{s}\right)^{\frac{a}{s-a}}-1\right)<0 \tag{6.16}$$

が成立する．次に $\left.\frac{d\langle\Delta R\rangle}{d\langle P^*\rangle}\right|_{\langle P^*\rangle=\frac{a}{s}}=\frac{1}{a}\times\left(n\times\left(\frac{a}{s}\right)^{n-1}-\frac{s}{s-a}\right)$ が $n\geqq\frac{s}{s-a}$ に対して単調減少である

ことを示す．ここで，$\left.\frac{d\langle\Delta R\rangle}{d\langle P^*\rangle}\right|_{\langle P^*\rangle=\frac{a}{s}}$ を n で微分すると，以下の式が得られる．

$$\frac{d}{dn}\left(\left.\frac{d\langle\Delta R\rangle}{d\langle P^*\rangle}\right|_{\langle P^*\rangle=\frac{a}{s}}\right)=\frac{d}{dn}\left(\frac{1}{a}\times\left(n\times\left(\frac{a}{s}\right)^{n-1}-\frac{s}{s-a}\right)\right)=\frac{1}{a}\times\left(\frac{a}{s}\right)^{n-1}\times\left(1+n\times\log\left(\frac{a}{s}\right)\right) \tag{6.17}$$

以下は一般に成立する公式である．

$$\log x=2\sum_{i=0}^{\infty}\frac{1}{2i+1}\left(\frac{x-1}{x+1}\right)^{2i+1},\ 0<x<\infty \tag{6.18}$$

ここで，$x=\frac{a}{s}$ を式(6.18)に代入し，$0<\frac{a}{s}<1$ であることに注意すると，$n\geqq\frac{s}{s-a}$ に対して，

式(6.17)の右辺にある $\left(1+n\times\log\left(\frac{a}{s}\right)\right)$ は以下となる．

$$1+n\times\log\left(\frac{a}{s}\right)=1+n\times2\sum_{i=0}^{\infty}\frac{1}{2i+1}\left(\frac{\frac{a}{s}-1}{\frac{a}{s}+1}\right)^{2i+1}$$

$$<1+\frac{s}{s-a}\times2\times\left(\frac{\frac{a}{s}-1}{\frac{a}{s}+1}\right)$$

$$=1-\frac{2s}{s+a}<0$$

以上より，式(6.17)は，

$$\frac{d}{dn}\left(\left.\frac{d\langle\Delta R\rangle}{d\langle P^*\rangle}\right|_{\langle P^*\rangle=\frac{a}{s}}\right)<0$$

となり，$\left.\frac{d\langle \Delta R\rangle}{d\langle P^*\rangle}\right|_{\langle P^*\rangle=\frac{a}{s}}=\frac{1}{a}\times\left(n\times\left(\frac{a}{s}\right)^{n-1}-\frac{s}{s-a}\right)$ が $n\geqq\frac{s}{s-a}$ に対して単調減少であることが示された．なお，式(6.16)により $n=\frac{s}{s-a}$ なら $\left.\frac{d\langle \Delta R\rangle}{d\langle P^*\rangle}\right|_{\langle P^*\rangle=\frac{a}{s}}<0$ であり，$\left.\frac{d\langle \Delta R\rangle}{d\langle P^*\rangle}\right|_{\langle P^*\rangle=\frac{a}{s}}$ は $n\geqq\frac{s}{s-a}$ に対して単調減少のため，$n\geqq\frac{s}{s-a}$ に対して $\left.\frac{d\langle \Delta R\rangle}{d\langle P^*\rangle}\right|_{\langle P^*\rangle=\frac{a}{s}}<0$ となる．式(6.14)の成立が示された．

$\langle P^*\rangle>0$ に対して$\langle R\rangle$は凹関数であるため，式(6.14)と(6.15)により，区間$\langle P^*\rangle=(\frac{a}{s},1)$に$\langle \Delta R\rangle$の極小値が存在する．さらに，$\langle P^*\rangle=\frac{a}{s}$ では$\langle \Delta R\rangle=\frac{1}{a}\times\left(\frac{a}{s}\right)^n>0$，$\langle P^*\rangle=1$ では$\langle \Delta R\rangle=0$であるため，$\langle \Delta R\rangle$の極小値が負で，かつ

$$\frac{a}{s}<P_b \tag{6.19}$$

が成立する．ただし，P_bは区間$\langle P^*\rangle=(0,1)$にける$\langle \Delta R\rangle=0$ となる点で，式(6.12)により定義される．

次に，P_bの下界を求める．$\langle P^*\rangle>0$ に対して$\langle R\rangle$は凹関数であるため，次式が成立する．

$$\left.\frac{d\langle \Delta R\rangle}{d\langle P^*\rangle}\right|_{\langle P^*\rangle=\frac{a}{s}}<\frac{\left.\langle \Delta R\rangle\right|_{\langle P^*\rangle=P_b}-\left.\langle \Delta R\rangle\right|_{\langle P^*\rangle=\frac{a}{s}}}{P_b-\frac{a}{s}} \tag{6.20}$$

式(6.12)の定義により$\left.\langle \Delta R\rangle\right|_{\langle P^*\rangle=P_b}=0$ である．さらに，表 6.1 により

$$\left.\langle \Delta R\rangle\right|_{\langle P^*\rangle=\frac{a}{s}}=\frac{1}{a}\times\left(\frac{a}{s}\right)^n$$

$$\left.\frac{d\langle \Delta R\rangle}{d\langle P^*\rangle}\right|_{\langle P^*\rangle=\frac{a}{s}}=\frac{1}{a}\times\left(n\times\left(\frac{a}{s}\right)^{n-1}-\frac{s}{s-a}\right)$$

が成立するので，式(6.20)は下記となる．

$$\frac{1}{a}\times\left(n\times\left(\frac{a}{s}\right)^{n-1}-\frac{s}{s-a}\right)<\frac{-\frac{1}{a}\times\left(\frac{a}{s}\right)^{n}}{P_b-\frac{a}{s}}$$

$$P_b>P_{b(lower)}=g+\frac{g^n}{\frac{1}{1-g}-n\times g^{n-1}} \tag{6.21}$$

ここで，g は式(6.3d)により与えられ，$P_{b(lower)}$は P_b の下界である．

次に P_b の上界を求める．$<P^*>>0$ に対して<R>は凹関数であるため，次式が成立する．

$$\frac{\langle\Delta R\rangle\Big|_{\langle P^*\rangle=P_{min}}-\langle\Delta R\rangle\Big|_{\langle P^*\rangle=\frac{a}{s}}}{P_{min}-\frac{a}{s}}>\frac{\langle\Delta R\rangle\Big|_{\langle P^*\rangle=P_b}-\langle\Delta R\rangle\Big|_{\langle P^*\rangle=\frac{a}{s}}}{P_b-\frac{a}{s}} \tag{6.22}$$

ここで，$\langle\Delta R\rangle\Big|_{\langle P^*\rangle=P_{min}}$ は<R>の極小値である．式(6.9)により，$\langle\Delta R\rangle\Big|_{\langle P^*\rangle=P_{min}}$ は，

$$\langle\Delta R\rangle\Big|_{\langle P^*\rangle=P_{min}}=\frac{1}{a}\times\left\{P_{min}{}^{n}-\frac{s}{s-a}\times P_{min}+\frac{a}{s-a}\right\} \tag{6.23}$$

となり，P_{min} は次のように $\frac{d\langle\Delta R\rangle}{d\langle P^*\rangle}\Big|_{\langle P^*\rangle=P_{min}}=0$ とすることにより求められる．式(6.13a)により，

$$\frac{d\langle\Delta R\rangle}{d\langle P^*\rangle}\Big|_{\langle P^*\rangle=P_{min}}=\frac{1}{a}\times\left\{n\times P_{min}{}^{n-1}-\frac{s}{s-a}\right\}=0$$

$$P_{min}=\left(\frac{1}{n}\times\frac{s}{s-a}\right)^{\frac{1}{n-1}} \tag{6.24}$$

が得られる．

式(6.23)，表 6.1 と式(6.24)により，式(6.22)は下記となる．

$$P_b<P_{b(upper)}=g+\frac{g^n}{\frac{1}{1-g}-\frac{h^n-g^n}{h-g}} \tag{6.25}$$

ただし，g と h はそれぞれ式(6.3d)と(6.3e)により与えられ，$P_{b(upper)}$は P_b の上界である．な

お，式(6.22)から式(6.25)への変換を付録 A を参照してください．

以上により，定理 6.1 が証明された．

（証明終わり）

ここで，式(6.9)では負の報酬 $\bar{r}$ を $\bar{r}=-1$ としているが，$\bar{r}$ が任意な負の実数値($\bar{r}<0$)でも，定理 6.1 が成立することを付記する．なお，$a \ll s$，または $n \gg 1$ の時，式(6.21)と(6.25)により $P_b \fallingdotseq P_{b(lower)} \fallingdotseq P_{b(upper)} \fallingdotseq \frac{a}{s}$ となる．図 6.5 は，$\frac{a}{s}$ を 0.05 から 0.55 に変化させ，及び n=3,5,10 とした時の P_b の上界 $P_{b(upper)}$（式(6.25)）と下界 $P_{b(lower)}$（式(6.21)）の変化を示す．図 6.5 に示されるように，$n \gg 1$，または $a \leqq 0.5$ の時，$P_b \fallingdotseq P_{b(lower)} \fallingdotseq P_{b(upper)} \fallingdotseq \frac{a}{s}$ となる．以上より，結果 2 と 3 が得られた．

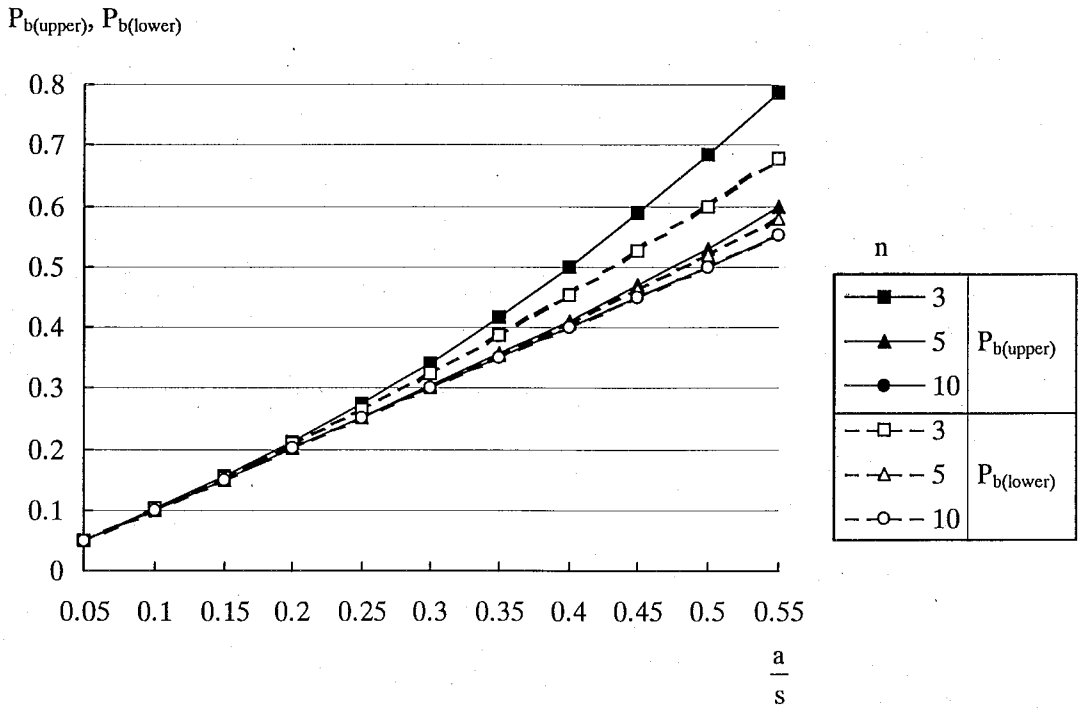


図 6.5: 負の報酬に関する P_b の上界 $P_{b(upper)}$ と下界 $P_{b(lower)}$曲線

6.3.3 正+負の報酬戦略

この戦略では，最適出力に対して正の報酬 $r^+>0$ が与えられ，非最適出力に対して負の報酬 $r^-<0$ が与えられる．ここで，$<R_o^+>$をある特定な最適出力ドメインに与えられる正の報酬の期待値とすると，$<R_o^+>$は次式により与えられる．

$$<R_o^+>=\frac{\langle P^*\rangle^n}{a}\times r^+,\ \text{ここで，}\ r^+>0 \tag{6.26}$$

正+負の報酬戦略により，最適出力ドメインに与えられる期待報酬と非最適出力ドメインに与えられる期待報酬の差$<\triangle R>$は，式(6.8a)，(6.8b)と(6.26)により，次式が得られる．

$$<\triangle R>=<R_o^+>+<R_o^->-<R_n^-> \tag{6.27a}$$

$$<\triangle R>=\frac{|r^-|}{a}\times\left\{\left(1+\frac{r^+}{|r^-|}\right)\times\langle P^*\rangle^n-\frac{s}{s-a}\times\langle P^*\rangle+\frac{a}{s-a}\right\} \tag{6.27b}$$

定理 6.2 （正+負の報酬戦略）

式(6.27b)により，

a) $n\leqq\frac{s}{s-a}$ なら，$<P^*>=[0,1]$では$<\triangle R>>0$ が成立する．

b) $n>\frac{s}{s-a}$ なら，

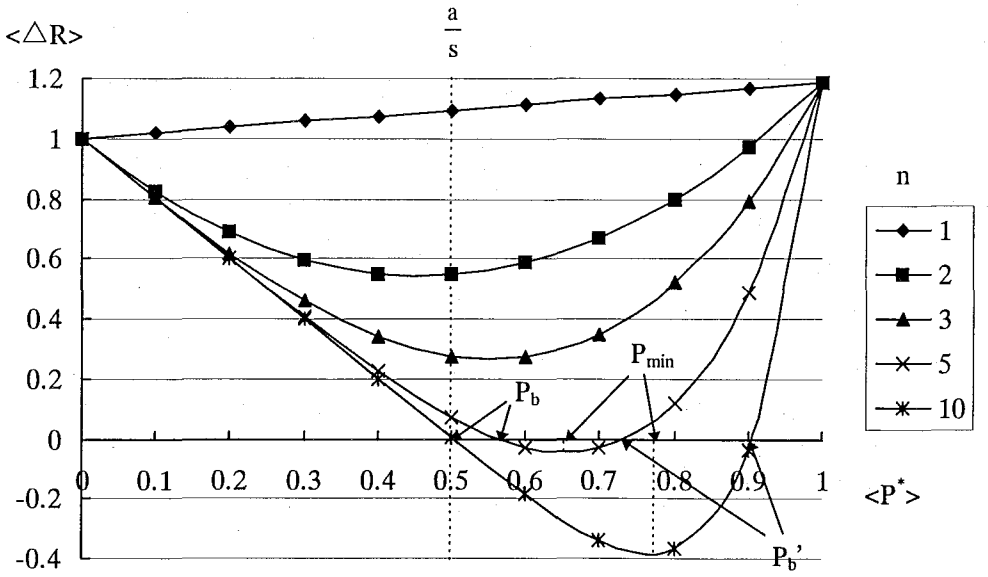
i) $\frac{r^+}{|r^-|}>L$ の場合，$<\triangle R>>0$ が成立する．

ii) $0<\frac{r^+}{|r^-|}\leqq L$ の場合，$<P^*>=[0,\ P_b)$では$<R>>0$，$<P^*>=P_b$ では$<R>=0$，$<P^*>=(P_b,\ P_b')$では$<R><0$，$<P^*>=P_b'$では$<R>=0$，$<P^*>=(P_b',\ 1]$では$<R>>0$ が成立する．

ただし，L と P_b はそれぞれ式(6.5b)と(6.6)により与えられ，$P_b\leqq P_b'<1$ である．

証明：

定理の証明の参考のため，図 6.6 は，s=2, a=1, n=1,2,3,5,10, $\frac{r^+}{|r^-|}$=1. 185 及び$<P^*>$を 0 から 1 に変化させた時の$<\triangle R>$（式(6.27b)）の変化を示す．



注：s=2; a=1; $\frac{r^+}{|r^-|}$=1. 185

図 6.6: 正+負の報酬戦略による最適出力ドメインと非最適出力ドメインの期待報酬の差

定理 6.1a)により，$n \leqq \frac{s}{s-a}$ なら，$\langle P^* \rangle$=[0, 1)に対して以下の式が成立する．

$$\frac{1}{a} \times \left\{ \langle P^* \rangle^n - \frac{s}{s-a} \times \langle P^* \rangle + \frac{a}{s-a} \right\} > 0$$

$$\langle P^* \rangle^n - \frac{s}{s-a} \times \langle P^* \rangle + \frac{a}{s-a} > 0$$

また，$1 + \frac{r^+}{|r^-|} > 1$ であるので，$n \leqq \frac{s}{s-a}$ に対して，

$$\left(1 + \frac{r^+}{|r^-|} \right) \times \langle P^* \rangle^n - \frac{s}{s-a} \times \langle P^* \rangle + \frac{a}{s-a} > 0$$

$$\frac{|r^-|}{a}\times\left\{\left(1+\frac{r^+}{|r^-|}\right)\times\langle P^*\rangle^n-\frac{s}{s-a}\times\langle P^*\rangle+\frac{a}{s-a}\right\}>0$$

が成立する．

さらに，式(6.27b)により，$\langle P^*\rangle=1$ なら，$\langle\triangle R\rangle=\frac{r^+}{a}>0$ が成立する．定理 6.2a)が証明された．

以下は，$n>\frac{s}{s-a}$ について考える．式(6.27b)により，以下の式が得られる．

$$\frac{d\langle\Delta R\rangle}{d\langle P^*\rangle}=\frac{|r^-|}{a}\times\left\{n\times\left(1+\frac{r^+}{|r^-|}\right)\times\langle P^*\rangle^{n-1}-\frac{s}{s-a}\right\} \tag{6.28a}$$

$$\frac{d^2\langle\Delta R\rangle}{d\langle P^*\rangle^2}=\frac{|r^-|}{a}\times n\times(n-1)\times\left(1+\frac{r^+}{|r^-|}\right)\times\langle P^*\rangle^{n-2} \tag{6.28b}$$

$n\geqq 2$，$a\geqq 1$ と$\langle P^*\rangle\geqq 0$ では$\frac{d^2\langle\Delta R\rangle}{d\langle P^*\rangle^2}\geqq 0$ であるので，$\langle R\rangle$は$\langle P^*\rangle\geqq 0$ に対して凹関数である．$\frac{d\langle\Delta R\rangle}{d\langle P^*\rangle}=0$ と式(6.27b)より，$\langle P^*\rangle$を消去すると，$\langle\triangle R\rangle$の極小値$\langle\Delta R\rangle\Big|_{\frac{d\langle\Delta R\rangle}{d\langle P^*\rangle}=0}$は下式により与えられる．

$$\langle\Delta R\rangle\Big|_{\frac{d\langle\Delta R\rangle}{d\langle P^*\rangle}=0}=\frac{|r^-|}{a}\times\left\{\left(\frac{1}{n}-1\right)\times\frac{s}{s-a}\times\left(\frac{s}{n\times\left(1+\frac{r^+}{|r^-|}\right)\times(s-a)}\right)^{\frac{1}{n-1}}+\frac{a}{s-a}\right\} \tag{6.29}$$

ここで，$\langle\Delta R\rangle\Big|_{\frac{d\langle\Delta R\rangle}{d\langle P^*\rangle}=0}>0$ とすると，式(6.29)により式(6.5)が得られる．定理 6.2b)i)が証明された．以下は，$0<\frac{r^+}{|r^-|}\leqq L$（L は式(6.5b)により与えられる）について考える．

式(6.27b)と(6.28a)より，$\langle P^* \rangle = 0, \frac{a}{s}, 1$ に対応する$\langle \Delta R \rangle$と$\frac{d\langle \Delta R \rangle}{d\langle P^* \rangle}$のそれぞれの式は表 6.2 により与えられる．

ここで，$0 < \frac{r^+}{|r^-|} \leqq L$，$n > \frac{s}{s-a}$ の場合，

$$\left.\frac{d\langle \Delta R \rangle}{d\langle P^* \rangle}\right|_{\langle P^* \rangle = \frac{a}{s}} < 0 \tag{6.30}$$

$$\left.\frac{d\langle \Delta R \rangle}{d\langle P^* \rangle}\right|_{\langle P^* \rangle = 1} > 0 \tag{6.31}$$

が成立する．表 6.2 により，式(6.31)が成立することが明らかであるが，ここでは，式(6.30)の成立について示す．

表 6.2 により，

$$\left.\frac{d\langle \Delta R \rangle}{d\langle P^* \rangle}\right|_{\langle P^* \rangle = \frac{a}{s}} = \frac{|r^-|}{a} \times \left\{ n \times \left(1 + \frac{r^+}{|r^-|} \right) \times \left(\frac{a}{s} \right)^{n-1} - \frac{s}{s-a} \right\} \tag{6.32}$$

表 6.2: $\langle P^* \rangle = 0, \frac{a}{s}, 1$ に対応する$\langle \Delta R \rangle$と$\frac{d\langle \Delta R \rangle}{d\langle P^* \rangle}$のそれぞれの式

	$\langle P^* \rangle = 0$	$\langle P^* \rangle = \frac{a}{s}$	$\langle P^* \rangle = 1$
$\langle \Delta R \rangle$	$\frac{\|r^-\|}{s-a}$	$\frac{\|r^-\|}{a} \times \left(1 + \frac{r^+}{\|r^-\|} \right) \times \left(\frac{a}{s} \right)^n$	$\frac{r^+}{a}$
$\frac{d\langle \Delta R \rangle}{d\langle P^* \rangle}$	$-\frac{\|r^-\|}{a} \times \frac{s}{s-a}$	$\frac{\|r^-\|}{a} \times \left\{ n \times \left(1 + \frac{r^+}{\|r^-\|} \right) \times \left(\frac{a}{s} \right)^{n-1} - \frac{s}{s-a} \right\}$	$\frac{\|r^-\|}{a} \times \left\{ n \times \left(1 + \frac{r^+}{\|r^-\|} \right) - \frac{s}{s-a} \right\}$

である．$\frac{r^+}{|r^-|} \leqq L$ であるため，式(6.32)は，

$$\left.\frac{d\langle \Delta R\rangle}{d\langle P^*\rangle}\right|_{\langle P^*\rangle=\frac{a}{s}} \leqq \frac{|r^-|}{a} \times \left\{ n \times (1+L) \times \left(\frac{a}{s}\right)^{n-1} - \frac{s}{s-a} \right\} \tag{6.33}$$

となる．式(6.5b)により与えられる L の式を式(6.33)に代入すると，

$$\left.\frac{d\langle \Delta R\rangle}{d\langle P^*\rangle}\right|_{\langle P^*\rangle=\frac{a}{s}} \leqq \frac{|r^-|}{a} \times \frac{s}{s-a} \left\{ \left(\frac{n-1}{n}\right)^{n-1} - 1 \right\} < 0$$

となり，式(6.30)の成立が示された．

式(6.28b)により<△R>は<P^*>≧0 に対して凹関数であるため，式(6.30)と(6.31)により，<P^*>≧0 に対する<△R>の極小値 $\langle \Delta R\rangle\Big|_{\frac{d\langle \Delta R\rangle}{d\langle P^*\rangle}=0}$ は，区間<P^*>=($\frac{a}{s}$, 1)に存在する．

L は式(6.29)において $\langle \Delta R\rangle\Big|_{\frac{d\langle \Delta R\rangle}{d\langle P^*\rangle}=0}$ =0 とすることによって求められたので，$\frac{r^+}{|r^-|}$=L の時，$\langle \Delta R\rangle\Big|_{\frac{d\langle \Delta R\rangle}{d\langle P^*\rangle}=0}$ =0 である．さらに，式(6.29)により $\langle \Delta R\rangle\Big|_{\frac{d\langle \Delta R\rangle}{d\langle P^*\rangle}=0}$ は $\frac{r^+}{|r^-|}$ に対して単調増加であるので，$\frac{r^+}{|r^-|} \leqq L$ の場合，<△R>の極小値 $\langle \Delta R\rangle\Big|_{\frac{d\langle \Delta R\rangle}{d\langle P^*\rangle}=0} \leqq 0$ となる．また，表 6.2 により，$\langle \Delta R\rangle\Big|_{\langle P^*\rangle=\frac{a}{s}} > 0$ と $\langle \Delta R\rangle\Big|_{\langle P^*\rangle=1} > 0$ であるため，

$$\frac{a}{s} < P_b \leqq P_b' < 1 \tag{6.34}$$

が成立する．

次に，P_b の下界を求める．<P^*>>0 に対して<R>は凹関数であるため，次式が成立する．

$$\left.\frac{d\langle\Delta R\rangle}{d\langle P^*\rangle}\right|_{\langle P^*\rangle=\frac{a}{s}} < \frac{\langle\Delta R\rangle\Big|_{\langle P^*\rangle=P_b} - \langle\Delta R\rangle\Big|_{\langle P^*\rangle=\frac{a}{s}}}{P_b - \frac{a}{s}} \tag{6.35}$$

ここで，$\langle\Delta R\rangle\Big|_{\langle P^*\rangle=P_b}$ =0 と表 6.2 により

$$\langle\Delta R\rangle\Big|_{\langle P^*\rangle=\frac{a}{s}} = \frac{|r^-|}{a} \times \left(1 + \frac{r^+}{|r^-|}\right) \times \left(\frac{a}{s}\right)^n$$

$$\left.\frac{d\langle\Delta R\rangle}{d\langle P^*\rangle}\right|_{\langle P^*\rangle=\frac{a}{s}} = \frac{|r^-|}{a} \times \left\{n \times \left(1 + \frac{r^+}{|r^-|}\right) \times \left(\frac{a}{s}\right)^{n-1} - \frac{s}{s-a}\right\}$$

が成立するので，式(6.35)は

$$\frac{|r^-|}{a} \times \left\{n \times \left(1 + \frac{r^+}{|r^-|}\right) \times \left(\frac{a}{s}\right)^{n-1} - \frac{s}{s-a}\right\} < \frac{-\frac{|r^-|}{a} \times \left(1 + \frac{r^+}{|r^-|}\right) \times \left(\frac{a}{s}\right)^n}{P_b - \frac{a}{s}} \tag{6.36}$$

$$P_b > P_{b(lower)} = g + \frac{f \times g^n}{\frac{1}{1-g} - n \times f \times g^{n-1}} \tag{6.37}$$

となる．ここで，f と g はそれぞれ式(6.6d)と(6.3d)により与えられ，$P_{b(lower)}$は P_b の下界である．

次に P_b の上界を求める．<P^*>>0 に対して<R>は凹関数であるため，次式が成立する．

$$\frac{\langle\Delta R\rangle\Big|_{\langle P^*\rangle=P_{min}} - \langle\Delta R\rangle\Big|_{\langle P^*\rangle=\frac{a}{s}}}{P_{min} - \frac{a}{s}} > \frac{\langle\Delta R\rangle\Big|_{\langle P^*\rangle=P_b} - \langle\Delta R\rangle\Big|_{\langle P^*\rangle=\frac{a}{s}}}{P_b - \frac{a}{s}} \tag{6.38}$$

式(6.27b)より，$\langle\Delta R\rangle\Big|_{\langle P^*\rangle=P_{min}}$ は，

$$\langle \Delta R \rangle \Big|_{\langle P^* \rangle = P_{min}} = \frac{|r^-|}{a} \times \left\{ \left(1 + \frac{r^+}{|r^-|} \right) \times {P_{min}}^n - \frac{s}{s-a} \times P_{min} + \frac{a}{s-a} \right\} \tag{6.39}$$

となり，P_{min}は$\frac{d\langle \Delta R \rangle}{d\langle P^* \rangle}\Big|_{\langle P^* \rangle = P_{min}}$ =0 とすることにより求められる．式(6.28a)により，

$$\frac{d\langle \Delta R \rangle}{d\langle P^* \rangle}\Big|_{\langle P^* \rangle = P_{min}} = \frac{|r^-|}{a} \times \left\{ n \times \left(1 + \frac{r^+}{|r^-|} \right) \times {P_{min}}^{n-1} - \frac{s}{s-a} \right\} = 0 \tag{6.40}$$

$$P_{min} = \left(\frac{\frac{s}{s-a} \times \frac{1}{n}}{1 + \frac{r^+}{|r^-|}} \right)^{\frac{1}{n-1}} \tag{6.41}$$

が得られる．以上より，式(6.38)は，

$$P_b < P_{b(upper)} = g + \frac{f \times g^n}{\frac{1}{1-g} - \frac{h^n - f^{\frac{n}{n-1}} \times g^n}{h - f^{\frac{1}{n-1}} \times g}} \tag{6.42}$$

となる．f，g と h はそれぞれ式(6.6d)，(6.3d)と(6.3e)により与えられ，$P_{b(upper)}$は P_b の上界である．式(6.38)から式(6.42)への変換を付録 B を参照してください．

以上により，定理 6.2 が証明された．

（証明終わり）

図 6.7 は，$\frac{a}{s}$ を 0.05 から 0.55 に変化させ，n=3,5,10 とした時の $\frac{r^+}{|r^-|}$ の下限値 L（式(6.5b)）の変化を表わす．図 6.7 に示されるように，n が大きい時，$\frac{a}{s}$ が小さいほど，$\frac{r^+}{|r^-|}$ の下限値 L が指数的に増大する．

a≪s，または n≫1 の時，式(6.37)と(6.42)により $P_b \fallingdotseq P_{b(lower)} \fallingdotseq P_{b(upper)} \fallingdotseq \frac{a}{s}$ となる．図 6.8 は，$\frac{a}{s}$ を 0.05 から 0.55 に変化させ，及び $\frac{r^+}{|r^-|}$=1.185，n=5,7,10 とした時の P_b の上界 $P_{b(upper)}$

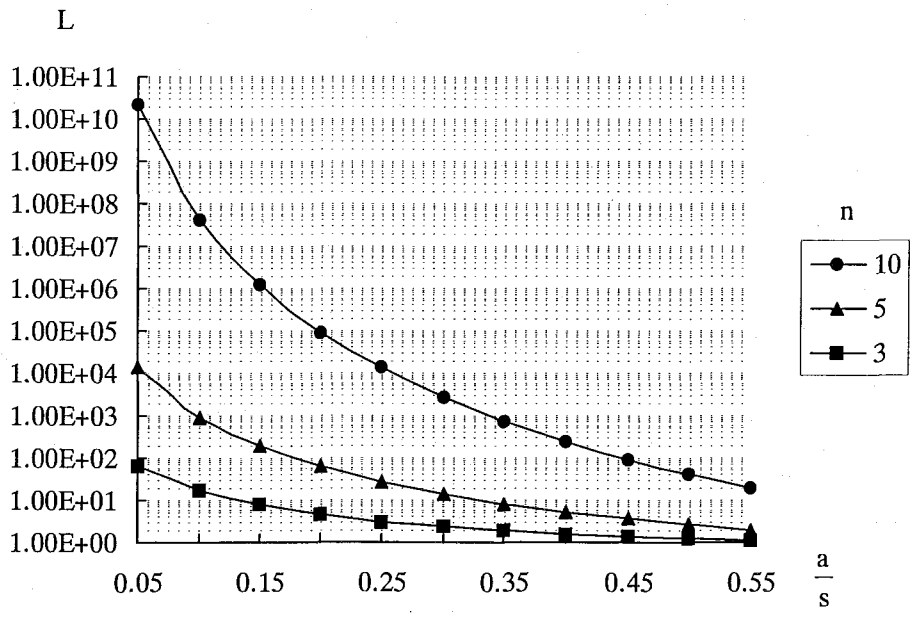
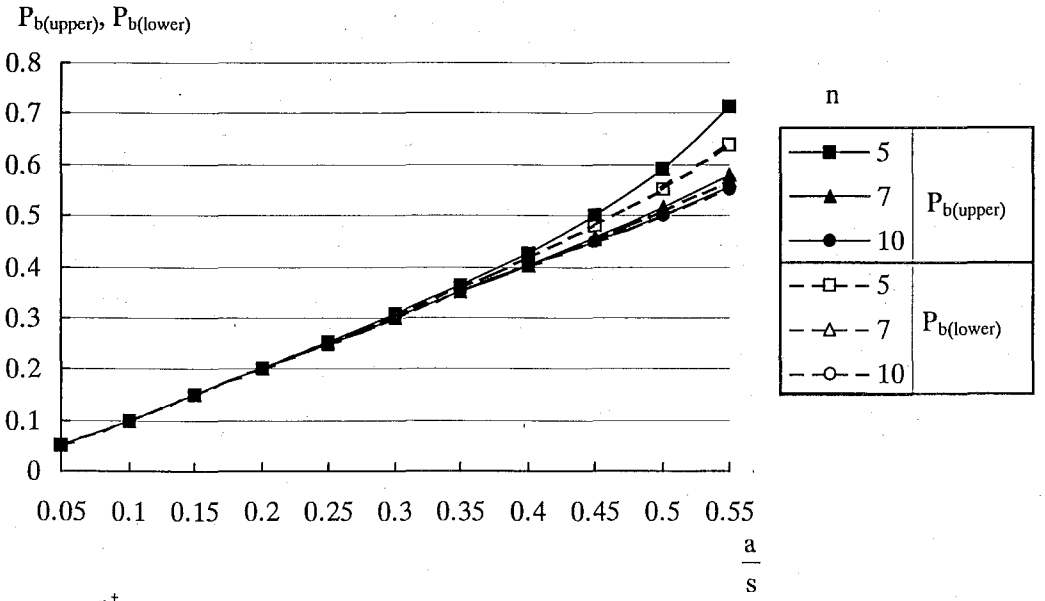


図 6.7: $\frac{r^+}{|r^-|}$の下限値 L の曲線



注：$\frac{r^+}{|r^-|}$=1.185

図 6.8: 正+負の報酬に関する P_b の上界 $P_{b(upper)}$ と下界 $P_{b(lower)}$曲線

（式(6.42)）と下界 $P_{b(lower)}$（式(6.37)）の変化を示す．図 6.5 に示されるように，$n \gg 1$，または $a \leqq 0.5$ の時，$P_b \fallingdotseq P_{b(lower)} \fallingdotseq P_{b(upper)} \fallingdotseq \frac{a}{s}$ となる．学習開始時では $<P^*> = \frac{a}{s}$ であるため，定理 6.2 により，$n > \frac{s}{s-a}$ かつ $0 < \frac{r^+}{|r^-|} \leqq L$ の場合，学習の進行に伴い，$<P^*>$は初期値から(1) P_b に収束するか，(2) P_b を中心に振動するか，または(3) 1 に収束することになる．しかし，(3) が可能とするためには，1 学習ステップで $<P^*>$が $<P^*> < P_b$ から $<P^*> > P_b$'と大きく変化できるように学習ステップを大きく設定する必要がある．一般に学習ステップが大きい場合，学習が不安定になることが多い．

以上より，結果 4 と 5 が得られた．

6.4 解析結果による報酬戦略の設計

ここでは，以上の解析結果を発電プラント起動スケジューリング問題における報酬戦略の設計に適用する．本問題の制限熱応力 σ_{lj}, j=1,...,4，起動スケジュール（または解）$(X_1,...,X_i,...,X_{10})$，起動スケジュールパラメータ X_i, i=1,...,10 及び起動スケジュールパラメータの値 X_i^k, i=1,...,10, k=1,...,s_i はそれぞれ前節までの報酬戦略の解析で定義した用語，すなわち入力[Y]，出力$(X_1,...,X_i,...,X_n)$，出力要素 X_i, i=1,...,n 及び出力ドメイン X_i^k, i=1,...,n, k=1,...,s に対応している．

発電プラント起動スケジューリング問題の場合，次元 n=10，1 次元当たりの平均出力ドメインサイズ $s = \frac{\sum_{i=1}^{10} s_i}{n} = \frac{197}{10} \fallingdotseq 20$，最適出力ドメインサイズ a=1 である．これにより $n > \frac{s}{s-a}$ であるので，前節の解析結果 3 により，本問題は負の報酬戦略を用いることができない．また，最適解に収束するための正の報酬対負の報酬の比，$\frac{r^+}{|r^-|}$ の下限値 L は前節の解析結果 5 により $L \fallingdotseq 2 \times 10^{10}$ となる．従って，正+負の報酬戦略を用いる場合，$\frac{r^+}{|r^-|}$ を大きく（2×10^{10} 以上）設定する必要がある．しかし，$\frac{r^+}{|r^-|}$ を大きく設定すれば，負の報酬 r^- の効果がないか，学習が不安定になるので，本問題は正+負の報酬戦略を用いることができない．

以上より，本問題は正の報酬戦略を用いる．なお，強化学習は本問題の最適解を探索・学習することを目的とするが，最適解が未知であるため，ここでは，学習のサブ目標とし

て，エリート解が更新された時，正の報酬を強化学習に与えることとする．

6.5 実験

6.5.1 実験の目的と方法

強化学習は報酬戦略によって，学習性能が大きく影響されるので，ここでは，報酬戦略の解析結果を確認するため，発電プラント起動スケジューリング問題を用いて報酬の授与方法に関する実験を行なう．なお，本実験は本研究の予備実験として，最大熱応力の計算を必要としない制限熱応力 set 0（表 5.1）を使い，報酬は次の 4 種類の方法を用いる．

a) エリート解が更新された時，正の報酬を与え，それ以外の時，報酬を与えない．
b) エリート解が更新された時，正の報酬を与え，エリート解より悪い解が探索された時，負の報酬を与え，エリート解と同様な解が探索された時，報酬を与えない．
c) 前回より改善解が探索された時，正の報酬を与え，それ以外の時，報酬を与えない．
d) 前回より改善解が探索された時，正の報酬を与え，前回より改悪解が探索された時，負の報酬を与え，前回と同様な解が探索された時，報酬を与えない．

6.5.2 実験結果

表 6.3 は，制限熱応力 set 0 を用いて，上述の四つの報酬授与方法 a)，b)，c)と d)に関する実験結果である．表 6.3 の実験結果により，報酬授与方法 a)による学習は，最適解に収束するが，それ以外の報酬授与方法では何れも最適解に収束しない．以下はこれに関する考察を行なう．

6.5.3 結果の考察

(1) 報酬授与方法 a)の場合

この報酬の授与方法では，エリート解が更新された時に，正の報酬が与えられ，それ以外の時，報酬が与えられない．この場合，更新されたエリート解は，最適解の構成要素でない起動スケジュールパラメータを含んでいることがあるので，最適解の構成要素でない起動スケジュールパラメータにも正の報酬が与えられることがある．

6.3 節の報酬戦略の解析では，最適解の構成要素でない起動スケジュールパラメータに正の報酬を与える解析について行わなかったが，表 6.3 の実験結果により，最適解の構成要素でない起動スケジュールパラメータに正の報酬が与えられても，学習が最適解に収束する．

表 6.3: 報酬授与方法に関する実験結果

(1) 試行回数：4, (2) 最適解：189.9 min., (3) 制限熱応力：set 0

(4) $r^+>0$：正の報酬, (5) $r^-<0$：負の報酬

報酬方法				起動時間(min.)			
	正の報酬	負の報酬	$\frac{r^+}{\lvert r^- \rvert}$	ave	var	max	min
a	**最良解更新**	**なし**	**---**	**189.9**	**0**	**189.9**	**189.9**
b	最良解更新	最良解より改悪	1	235.3	8	245.9	227.8
			2	227.0	7	237.5	218.8
			4	219.3	3	224.3	215.5
c	改善解	なし	---	216.9	11	225.2	200.3
d	改善解	改悪解	1	217.8	7	229.0	213.4
			2	194.3	3	199.7	190.9
			4	207.1	8	218.5	200.5

この理由は，エリート解の更新回数の増加に伴い，エリート解に最適解の構成要素である起動スケジュールパラメータが含まれる確率が高くなり，最適解の構成要素である起動スケジュールパラメータに与えられる正の報酬が最適解の構成要素でない起動スケジュールパラメータよりも多いからと考えられる．

(2) 報酬授与方法 b)の場合

この報酬の授与方法では，エリート解が更新された時に，正の報酬が与えられ，エリート解より悪い解が探索された時，負の報酬が与えられる．この場合，表 6.3 の実験結果に示されるように，学習が最適解の収束しない．以下はその理由を述べる．

6.4 節の報酬戦略の設計結果により，最適解に収束するため正の報酬対負の報酬の比 $\frac{r^+}{\lvert r^- \rvert}$ の下限値 L は，$L \fallingdotseq 2\times10^{10}$ となる．表 6.3 の実験では，$\frac{r^+}{\lvert r^- \rvert}$ が 1,2,4 と設定されているため，学習が最適解に収束しない．また，6.3 節の結果 5（正+負の報酬戦略で，高出力次元の場

合）に説明されるように，$\frac{r^+}{|r^-|}$を大きく設定すれば，負の報酬の効果がないか，学習が最適解に収束しないか，学習が不安定になる．以上により，本起動スケジューリングのような高次元問題において負の報酬は用いられないことが言える．

(3)　報酬授与方法 c)と d)の場合

学習過程において，エリート解より低い評価値の改善解とエリート解より高い評価値の改善解と比較して前者の生成回数が多い．改善解が生成されるだけで，正の報酬を与えると，エリート解より低い評価値の解に対して多くの報酬を与えることになり，学習が最適解に収束しない．

6.6 おわりに

強化学習は，報酬を手掛かりに学習を行うので，報酬戦略の戦略によって，学習性能が大きく影響される．報酬には正の報酬と負の報酬がある．正の報酬戦略が用いられた時，強化学習のよい挙動に対して正の報酬が与えられ，負の報酬戦略が用いられた時，悪い挙動が観察される時，負の報酬が与えられる．本章では，強化学習の出力次元，出力ドメインサイズ，最適出力ドメインサイズと上記のような報酬戦略の関連について解析を行い，解析結果を本問題に対する報酬戦略の設計に用いた．本問題の場合，強化学習の出力次元，平均出力ドメインサイズと最適出力ドメインサイズは，それぞれ 10，20 と 1 である．報酬戦略の設計により，本問題に対して，正の報酬では学習は最適解に収束するが，正+負の報酬や負の報酬では学習が収束しないか，学習が不安定になる結果が得られた．また，この設計結果は本問題に対する報酬戦略の実験により確認できた．従来，強化学習の報酬戦略の設計に関する研究がほとんど行われなかったが，ここで，本研究結果を報酬戦略の設計指標の一つとして提案する．なお，本研究結果により，強化学習の出力が高次元である問題では，負の報酬戦略が用いられないことが結論付けられた．高次元問題において，強化学習を適用する場合，正の報酬を用いることを提案する．

7 結論

7.1 研究成果の要約

火力発電プラント起動スケジューリング問題は，多峰性で，オンライン探索として解空間サイズが大きく，解の評価のためのタービンダイナミックシミュレーションが膨大な計算時間を必要とする困難なオンライン探索問題である．本問題の目的関数である起動時間関数は単調であることから，境界条件上に最適解が必ず存在する，すなわち不等式により構成される本問題の制約条件に活性な制約式(active constraint)が含まれることに着目して，探索を境界近傍に限定する強制操作を提案した．境界近傍探索を行うため，強制操作を組込んだ近傍探索 GA を用いて，最適解探索モデルを構築した．近傍探索では，同じ解を繰り返し探索することがしばしば発生するので，最適解探索モデルに再利用機能とタブ戦略を導入した．再利用機能とは，探索した解に対応するシミュレーション結果を記憶し，同じ解が探索された時，記憶されたシミュレーション結果を再利用し，評価のためのシミュレーション計算を省くことにより，探索効率の向上を図るものである．タブ戦略とは最近探索された解の再探索を禁止することにより，探索効率の向上と局所的最適解の早期脱出を図ることである．このような戦略を導入したことにより，探索の繰り返しに伴うコストを大幅に軽減し，効率的な最適解探索モデルを構築することができた．実験結果により，強制操作と再利用機能を導入した場合，GA とタブ戦略の融合は GA 単独，TS 単独，SA 単独ないし SA とタブ戦略の融合より探索効率がよいと確認された．

タービンウォーム起動に対して，探索された最適解の起動時間は設計値より 22 分または約 10%改善された．

探索条件として与えられる種々な制限熱応力に対して，頑健性のある探索モデルを構築するため，複数境界探索戦略を導入した．複数境界探索戦略とは，実行可能解空間の境界に対して緩和境界を設定し，解を境界ないし緩い境界に移動させたり，実行可能解空間のエリート解と緩和実行可能解空間のエリート解という 2 種類のエリート解を保存することにより，実行可能解空間の外側近傍を積極的に探索する戦略である．このような戦略の導入により，頑健性のある探索モデルを得ることができ，探索ステップ数を多く必要とする制限熱応力に対する探索回数を大幅に減少することができた．

しかし，以上のような探索だけの接近法では，最適解または近似最適解探索に必要な平均 CPU 時間は SPARC station 20 上で約 6 分かかり，プラント運転に必要なオンライン探索性能を満たせなかった．オンライン探索性能を満たすため，強化学習と GA を融合するハ

イブリッド方式を提案した．探索過程において，GA は強化学習を有望な領域で学習するようにガイドし，強化学習の学習を加速する．強化学習は学習効果により，探索過程の序盤に，有望な解候補を生成することができ，GA の探索を加速する．このような相乗効果により，学習と探索が加速され，強化学習を導入しない場合と比べて，約 13%の探索効率の向上が実験結果により示された．さらに，代表的な制限熱応力について学習を行った結果，未学習の制限熱応力に対する最適解または近似最適解探索に必要な CPU 時間は，平均 1 ないし 8 秒，最大 1 ないし 58 秒となり，目標とする性能をほぼ達成できた．

強化学習は報酬を手掛かりに学習を行い，報酬の与え方により学習性能が大きく影響される．学習に用いられる報酬の戦略として，正の報酬（強化学習のよい挙動に対して，正の報酬を与える）と負の報酬（強化学習の悪い挙動に対して，負の報酬を与える）がある．本研究は強化学習の出力次元，出力ドメインサイズと最適出力ドメインサイズにより，報酬戦略に関する解析を行い，その結果を本問題の報酬戦略の設計に適用した．従来の研究では，報酬戦略の設計についてほとんど行われなかった．ここで，本研究結果を強化学習の報酬の設計の一つの指標として提案する．なお，本研究結果により，高次元問題では負の報酬が用いられないという結論が導かれ，高次元問題では正の報酬を使うことを提案する．

7.2 今後の研究課題

本研究は，タービンウォーム起動モードについて実験を行った．実用化のため，他の起動モード，すなわちコールド起動やホット起動モードをも考慮する必要がある．起動モードによって，すなわちプラント起動時のタービンロータ温度，主蒸気温度や再熱蒸気温度によって最適解が変化するため，実用化の場合，強化学習の学習条件にこれらの起動時の温度条件を追加する必要がある．このような起動条件の追加によって，強化学習に用いられるニューラルネットワークの学習能力を超える場合，以下の対策が考えられる．

a) ニューラルネットワークの学習能力の限界に合せ，学習条件の値のレンジを分割し，複数のニューラルネットワークを使う．例えば，学習条件の値のレンジを 3 分割する場合，コールド，ウォームとホットという 3 個のニューラルネットワークを用意し，それぞれのニューラルネットワークが対応する起動モードの最適起動スケジュールの学習を行う．すなわち，例えばコールドニューラルネットワークは，低い温度レンジ，ホットニューラルネットワークは，高い温度レンジ，ウォームニューラルネットワークはその中間の温度レンジに対する最適解の学習を個別に行なう．プラント起動時で

は，起動モードによって，対応するニューラルネットワークを用いて，与えられる起動時の温度条件に対応する有望な初期解候補を生成し，最適解または近似最適解の探索を行なう．

b) 本研究は，プラント起動時における有望な初期解候補の生成として，学習を行なったニューラルネットワークを用いることを提案したが，このような有望な初期解候補の生成として，従来の設計手法[Hanzalek 66]を用いることも考えられる．この場合，プラント起動スケジュールの最適解探索アルゴリズムの枠組みは，次のように構成される．

ステップ 1)　与えられる起動条件に対して従来の設計手法[Hanzalek 66]により，初期解候補の生成を行なう．

ステップ 2)　生成された初期解候補を用いて，GA と強化学習の融合による適応的探索モデルにより最適解または近似最適解の探索を行なう．

このような枠組みでは，従来のような起動スケジュール設計コストが発生するが，利点としては，上述のような起動条件の追加によるニューラルネットワークの学習能力問題が解消できると共に，従来の設計手法で生成された起動スケジュールを改善しながら，最適解または近似最適解の探索を行なうので，従来の設計手法よりよい解，すなわち従来の設計手法より起動時間の短い起動スケジュールの生成が期待できる．

c) 発電プラントの起動スケジュールはタービン通気時点で最終決定される[Hanzalek 66]．ここで，タービン通気までに起動スケジューリングを周期的に，例えば 5 分周期で行ない，各周期の起動スケジューリングは，前周期で探索された解を初期解として使う．各周期の最適解は前周期のそれと近似するので，前周期の探索された解を初期解として使うことにより，最適解または近似最適解の探索時間の短縮が期待できる．

実用化の場合，実機の挙動に近似したダイナミックシミュレーションモデルの構築が要請される．この場合，シミュレーションモデルのパラメータ調整として，GA ないし強化学習の適用が考えられる．すなわち，シミュレーションにより計算される熱応力と実機の発生熱応力の誤差が最小化となるように，GA ないし強化学習を適用して，シミュレーションモデルの最適パラメータを探索する．

以上は発電プラント起動スケジューリングに関する研究課題であるが，本研究結果の一般問題への適用として以下の研究課題がある．

a) 工学分野における最短時間問題は，最適化の対象となる時間を表わす目的関数が単調である場合が多い．本研究で示されるように，目的関数が単調である場合，最適解は

実行可能解空間の境界上に存在する．また，複数の目的関数が競合する問題において，一つの目的関数だけを残して，残りの目的関数を制約関数とした場合，最適解は実行可能解の境界に存在することが知られている．ここで，本研究結果をベースに適応的境界探索モデルの一般的なわく組みを構築し，最適解が境界上に存在するような一般問題への適用を図る．

b) 本研究では，強化学習の出力が高次元である問題に対して，負の報酬を用いた場合，学習が学習が最適解に収束しないか，学習が不安定になる条件を示した．ここで，「学習が最適解に収束しない」と「不安定になる」を判別する条件，及び不安定の場合の学習の挙動に関する解析を行ない，多次元問題における強化学習の挙動をより明らかにする．

付録

A　式(6.22)から式(6.25)への変換

式(6.24)，表 6.1 と式(6.23)により，式(6.22)は次式となる．

$$\frac{\frac{1}{a}\times\left(\left(\frac{1}{n}\times\frac{s}{s-a}\right)^{\frac{n}{n-1}}-\frac{s}{s-a}\times\left(\frac{1}{n}\times\frac{s}{s-a}\right)^{\frac{1}{n-1}}+\frac{a}{s-a}\right)-\frac{1}{a}\times\left(\frac{a}{s}\right)^{n}}{\left(\frac{1}{n}\times\frac{s}{s-a}\right)^{\frac{1}{n-1}}-\frac{a}{s}}>\frac{0-\frac{1}{a}\times\left(\frac{a}{s}\right)^{n}}{P_b-\frac{a}{s}}$$

さらに式の変換を行なうと，下記の式が得られる．

$$P_b<\frac{a}{s}+\frac{\left(\frac{a}{s}\right)^{n}\times\left(\left(\frac{1}{n}\times\frac{s}{s-a}\right)^{\frac{1}{n-1}}-\frac{a}{s}\right)}{-\left(\frac{1}{n}\times\frac{s}{s-a}\right)^{\frac{n}{n-1}}+\frac{s}{s-a}\times\left(\frac{1}{n}\times\frac{s}{s-a}\right)^{\frac{1}{n-1}}-\frac{a}{s-a}+\left(\frac{a}{s}\right)^{n}}$$

$$=\frac{a}{s}+\frac{\left(\frac{a}{s}\right)^{n}\times\left(\left(\frac{1}{n}\times\frac{s}{s-a}\right)^{\frac{1}{n-1}}-\frac{a}{s}\right)}{\frac{s}{s-a}\left(\left(\frac{1}{n}\times\frac{s}{s-a}\right)^{\frac{1}{n-1}}-\frac{a}{s}\right)-\left(\frac{1}{n}\times\frac{s}{s-a}\right)^{\frac{n}{n-1}}+\left(\frac{a}{s}\right)^{n}}$$

$$=g+\frac{g^{n}}{\frac{1}{1-g}-\frac{h^{n}-g^{n}}{h-g}}$$

ただし，g と h はそれぞれ式(6.3d)と(6.3e)により与えられ，式(6.25)が得られる．

B　式(6.38)から式(6.42)への変換

式(6.39)，表 6.2 と(6.41)により，式(6.38)は，

$$\frac{\frac{|r^-|}{a}\times\left\{\left(1+\frac{r^+}{|r^-|}\right)\times\left(\frac{\frac{s}{s-a}\times\frac{1}{n}}{1+\frac{r^+}{|r^-|}}\right)^{\frac{n}{n-1}}-\frac{s}{s-a}\times\left(\frac{\frac{s}{s-a}\times\frac{1}{n}}{1+\frac{r^+}{|r^-|}}\right)^{\frac{1}{n-1}}+\frac{a}{s-a}\right\}-\frac{|r^-|}{a}\times\left(1+\frac{r^+}{|r^-|}\right)\times\left(\frac{a}{s}\right)^n}{\left(\frac{\frac{s}{s-a}\times\frac{1}{n}}{1+\frac{r^+}{|r^-|}}\right)^{\frac{1}{n-1}}-\frac{a}{s}}$$

$$>\frac{0-\frac{|r^-|}{a}\times\left(1+\frac{r^+}{|r^-|}\right)\times\left(\frac{a}{s}\right)^n}{P_b-\frac{a}{s}}$$

さらに式の変換を行なうと，下記の式が得られる．

$$P_b<\frac{a}{s}+\frac{f\times\left(\frac{a}{s}\right)^n\times\left\{\left(\frac{1}{f}\times\frac{s}{s-a}\times\frac{1}{n}\right)^{\frac{1}{n-1}}-\frac{a}{s}\right\}}{-f^{\frac{-1}{n-1}}\left(\frac{s}{s-a}\times\frac{1}{n}\right)^{\frac{n}{n-1}}+\frac{s}{s-a}\times\left(\frac{1}{f}\times\frac{s}{s-a}\times\frac{1}{n}\right)^{\frac{1}{n-1}}-\frac{a}{s-a}+f\times\left(\frac{a}{s}\right)^n}$$

$$=\frac{a}{s}+\frac{f\times\left(\frac{a}{s}\right)^n\times\left\{\left(\frac{1}{f}\times\frac{s}{s-a}\times\frac{1}{n}\right)^{\frac{1}{n-1}}-\frac{a}{s}\right\}}{\frac{s}{s-a}\times\left\{\left(\frac{1}{f}\times\frac{s}{s-a}\times\frac{1}{n}\right)^{\frac{1}{n-1}}-\frac{a}{s}\right\}-f^{\frac{-1}{n-1}}\times\left(\frac{s}{s-a}\times\frac{1}{n}\right)^{\frac{n}{n-1}}+f\times\left(\frac{a}{s}\right)^n}$$

$$=\frac{a}{s}+\frac{f\times\left(\frac{a}{s}\right)^n}{\frac{s}{s-a}-\frac{f^{\frac{-1}{n-1}}\times\left(\frac{s}{s-a}\times\frac{1}{n}\right)^{\frac{n}{n-1}}-f\times\left(\frac{a}{s}\right)^n}{\left(\frac{1}{f}\times\frac{s}{s-a}\times\frac{1}{n}\right)^{\frac{1}{n-1}}-\frac{a}{s}}}$$

$$=g+\frac{f\times g^n}{\frac{1}{1-g}-\frac{h^n-f^{\frac{n}{n-1}}\times g^n}{h-f^{\frac{1}{n-1}}\times g}}$$

ただし，f, g と h はそれぞれ式(6.6d), (6.3d)と(6.3e)により与えられる．以上より，式(6.42)が得られる．

謝辞

本研究にあたって，始終熱心な御指導と多大なる御教示を頂きました小林重信教授に心より深く感謝の意を表わします．また，研究を進める上で，多大な御助言を頂きました山村雅幸助教授と宮崎和光助手にお礼を申し上げます．研究室の計算機の使い方に関していろいろと教えて頂きまして，進化型計算研究ついて助言をくださった佐藤 浩様と小野 功様，強化学習研究について助言を頂きました木村 元様を始め，研究室の皆様に感謝致します．

この貴重な研究機会を与えて頂きました東芝システムテクノロジー 小林 進様に心より深く感謝致します．

本研究に対して許可をくださった東芝エンジニアリング 鈴木 茂様（元 東芝 技師長），同じく本研究に対して許可をくださって，研究期間中に多くの助言を頂きました東芝 田中俊太郎様，入社以来，多くの研究機会を与えて頂きました東芝 河井研介様，研究途中，通信部門に転籍となったにもかかわらず，本研究の継続に対して許可をくださった東芝 情報通信部門 畑 浩靖様，高岡博史様，発電部門 根田利勝様，柳沢昭男様に厚く感謝致します．本研究用のタービンダイナミックシミュレーションモデルを提供して頂きました東芝 松浦泰則様，中本政志様，タービン専門家として助言を頂きました東芝 桃枝克郎様，篠崎幸雄様にお礼を申し上げます．本研究に対して多くの支援を頂きました東芝システムテクノロジー 前大道正博様，東芝 稲葉幸夫様，青木滋夫様，増山秀夫様，戸根洋一様，酒井敏夫様，一色利朗様を始め，東芝並びに東芝システムテクノロジー関係者の皆様に深く感謝致します．

研究期間中，英語論文の校正をしてくださった兄 神谷昭広，多くの理解と支援を頂きました妻 方 幼娜に謝意を述べ，研究期間中に，一緒に遊ぶ時間がほとんどなかった長女 薫，次女 希にありがとう．最後に本論文を父と母に捧げたい．

参考文献

[Barto 83] Barto, A. G., Sutton, R. S. and Anderson, C. W.: "Neuronlike Adaptive Elements That Can Solve Difficult Learning Control problems," IEEE Transactions on Systems, Man, and Cybernetics, SMC-13, No. 5, pp. 834-846 (1983).

[Bednarski 73] Bednarski, S. and Shen, C. N.: "ANALYSIS AND ALGORITHM FOR A MINIMAX PROBLEM WITH THERMAL STRESS APPLICATIONS," 14th Joint Automatic Control Conference, Vol. 20, No. 22, pp. 765-775 (1973).

[陳 89]陳 洛南，豊田淳一：「モンテカロル法による火力発電機の起動停止計画」，電気学会論文誌 B，Vol. 109B，No. 2，73-80 (1989).

[Domachowski 86] Domachowski, Z.: "STRESS OPTIMUM CONTROL OF A START-UP AND SHUT-DOWN OF CYCLIC DUTY TURBINE," Proceedings of American control conference, pp. 1954-1959 (1986).

[Eshelman 93] Eshelman, L. J. and Schaffer, J. D.: "Real-Coded Genetic Algorithms and Interval-Schemata," Foundations of Genetic Algorithms, Morgan Kaufmann, pp. 187-202 (1993).

[Fogel 94] Fogel, D. B.: "An Introduction to Simulated Evolutionary Optimization," IEEE Transactions on Neural Networks, Vol. 5, No. 1, pp. 3-14 (1994).

[Fox 93] Fox, B. L.: "Integrating and accelerating tabu search, simulated annealing, and genetic algorithms," Annals of Operations Research 41, pp. 47-67 (1993).

[Glover 86] Glover, F.: "Future paths for integer programming and links to artificial intelligence," Computers & Operations Research, Vol. 13, pp. 533-549 (1986).

[Glover 93] Glover, F., Taillard, E. and De Werra, D.: "A user's guide to tabu search," Annals of Operations Research 41, pp. 3-28 (1993).

[Glover 94] Glover, F.: "Tabu search for nonlinear and parametric optimization (with links to genetic algorithms)," Discrete Applied Mathematics 49, pp. 231-255 (1994).

[Goldberg 89a] Goldberg, D. E.: Genetic Algorithms in Search, Optimization & Machine Learning, Addison Wesley (1989).

[Goldberg 89b] Goldberg, D. E.: "Sizing Populations for Serial and Parallel Genetic Algorithms," Proceedings of 3rd International Conference on Genetic Algorithms, pp. 70-79 (1989).

[Goldberg 91] Goldberg, D. E.: "Real-coded Genetic Algorithms, Virtual Alphabets, and Blocking," Complex Systems 5, pp. 139-167 (1991).

[Hanzalek 66] Hanzalek, F. J. and Ipsen, P. G.: "Thermal Stresses Influence Starting, Loading of Bigger Boilers, Turbines," Electrical World, pp. 58-62 (1966).

[Holland 75] Holland, J. H.: Adaptation in Natural and Artificial Systems, The MIT Press (1975).

[神谷 95]神谷昭基，山村雅幸，小林重信：「進化型計算による発電プラント起動スケジューリング」，知能シンポジウム(1995).

[Kamiya 95] Kamiya, A., Ono, I., Yamamura, M. and Kobayashi, S.: "Thermal Power Plant Start-up Scheduling with Evolutionary Computation by Using an Enforcement Operator," Proceedings of IEEE International Conference on Systems, Man, and Cybernetics, pp. 1372-1379 (1995).

[Kamiya 96] Kamiya, A., Yamamura, M. and Kobayashi, S.: "Optimal Power Plant Start-up Scheduling: A Reinforcement Learning Approach Combined with Evolutionary Computation," Proceedings of 4th International Conference on Soft Computing, pp. 519-524 (1996).

[神谷 97a]神谷昭基，小野功，山村雅幸，小林重信：「強制操作とタブ戦略を導入した進化型計算による発電プラント起動スケジューリング」，人工知能学会誌，Vol. 12，No. 1，100－110 (1997).

[神谷 97b]神谷昭基，小野功，小林重信：「適応的探索による火力発電プラント起動スケジューリング」，電気学会論文誌 C(1997).

[木村 96]木村元，山村雅幸，小林重信：「部分観測マルコフ決定過程下での強化学習：確率傾斜法による接近」，人工知能学会誌，Vol. 11，No. 5，761－768 (1996).

[Kirtpatrick 83] Kirtpatrick, S., Gelatt, C. D. and Vecchi, M. P.: "Optimization by Simulated Annealing," Science, Vol. 220, pp. 671-680 (1983).

[小林 96]小林重信，小野功：「進化型計算に基づくシステム最適化」，計測と制御，Vol. 35，No. 7，508－513 (1996).

[今野 78]今野浩，山下浩：非線形計画法，日科技連出版社(1978).

[久保 95]久保幹雄:「Generic Local Search と Life Span Method」,計測と制御，Vol. 34，353-357 (1995).

[Lin 93] Lin, L. J.: "Scaling Up Reinforcement Learning for Robot Control," Proceedings of the tenth International Conference on Machine Learning, pp. 182-189 (1993).

[Matsumoto 82] Matsumoto, H., Kato, F., Eki, Y., Hisano, K., Fukushima, K. and Sato, Y.: "TURBINE CONTROL SYSTEM BASED ON PREDICTION OF ROTOR THERMAL STRESS," IEEE Transactions on Power Apparatus and Systems, Vol. PAS-101, No. 8, pp. 2504-2512 (1982).

[Matsumoto 93] Matsumoto, H., Eki, Y., Kaji, A., Nigawara, S., Tokuhira, M. and Suzuki, Y.: "AN OPERATION SUPPORT EXPERT SYSTEM BASED ON ON-LINE DYNAMICS SIMULATION AND FUZZY REASONING FOR STARTUP SCHEDULE OPTIMIZATION IN FOSSIL POWER PLANTS," IEEE Transactions on Energy Conversion, Vol. 8, No. 4, pp. 674-680 (1993).

[宮崎 96] 宮崎和光：離散マルコフ決定過程における強化学習，東京工業大学学位論文 (1996).

[Ono 96] Ono, I., Yamamura, M. and Kobayashi, S.: "A Genetic Algorithm with Characteristic-preserving for Function Optimization," Proceedings of the 4th International Conference on Soft Computing, pp. 511-514 (1996).

[Reeves 93] Reeves, C. R.: "USING GENETIC ALGORITHMS WITH SMALL POPULATIONS," Proceedings of 5th International Conference on Genetic Algorithms, pp. 92-99 (1993).

[Rosen 94] Rosen, B. E.，中野良平：「シミュレーテッドアニーリング—基礎と最新技術—」，人工知能学会誌，Vol. 9，No.3，365-372 (1994).

[Rumelhart 86] Rumelhart, D. E., McClelland, J. L., et al: PARALLEL DISTRIBUTION PROCESSING, Vol. 1, The MIT Press (1986).

[Satoh 96] Satoh, H., Yamamura, M. and Kobayashi, S.: "Minimal Generation Gap Model for GAs Considering Both Exploration and Exploitation," Proceedings of the 4th International Conference on Soft Computing, pp. 493-497 (1996).

[Smith 93] Smith, A. E. and Tate, D. M.: "Genetic Optimization Using A penalty Function," Proceedings of 5th International Conference on Genetic Algorithms, pp. 499-505 (1993).

[Spears 91] Spears, W. M. and De Jong, K. A.: "An Analysis of Multi-Point Crossover," Foundations of Genetic Algorithms, Morgan Kaufmann, pp. 301-315 (1991).

[Stork 95] Stork, D. G.: "NEURAL NETWORKS AND PATTERN RECOGNITION," Tutorial #6 of IEEE International Conference on Systems, Man, and Cybernetics (1995).

[鈴木 80]鈴木孝雄，鈴木 実，茂在哲雄，田中俊太郎：「広野火力１，２号変圧運転プラントの超自動化」，火力原子力発電，Vol. 31，No. 10，1090-1106 (1980).

[Tan 91] Tang, M.: "Cost-Sensitive Reinforcement Learning for Adaptive Classification and Control," Proceedings of 9 th National Conference on Artificial Intelligent, pp. 774-780 (1991).

[Williams 92] Williams, R. J.: "Simple Statistical Gradient-Following Algorithms for Connectionist Reinforcement Learning," Machine Learning ,Vol. 8, pp. 229-256 (1992).

[山村 94]山村雅幸，小林重信：「強化学習と人工生命」，人工知能学会全国大会（第８回），チュートリアル講演テキスト，I-1 (1994).