2003年日本オペレーションズ・リサーチ学会
春季研究発表会

強化学習による最適チェックポイントの動的生成

岡村寛之 (01013754)†, 西村祐樹‡, 土肥正 (01307065)†
† 広島大学大学院工学研究科情報工学専攻
‡ 広島大学工学部第二類（電気系）

1. はじめに

データベースに代表されるファイルシステムでは，データ処理を完了するまでに計算コストを必要とする．反面，システム障害が発生するとかなりの計算ロスを被る可能性がある．そこで，主記憶から安定な二次記憶媒体にデータを保存するチェックポインティングと呼ばれる予防保全手続きと，障害発生後にシステムの状態を元の状態まで回復させるロールバックリカバリと呼ばれる事後保全手続きがなされる．

一般的に，チェックポインティングは次のように行われる．システム上でチェクポイントの生成が選択されると，計算のプロセスは親プロセスと子プロセスの二つに分岐される．分岐直後，親プロセスと子プロセスのデータは同一であり，親プロセスは計算を続行する．一方，子プロセスはチェックポイント以前までに蓄積されたデータを安定な二次記憶媒体へ保存する．つまり，チェックポインティングを行った後，親プロセスと子プロセスは平行して処理を行っている状態となる．

Vaidya [1] は，上述のチェックポイントモデルにおいて，計算ロスの最も小さい最適なチェックポイント間隔を解析的に導出している．本稿では，上述のチェックポイント生成モデルをセミマルコフ決定過程 (SMDP) によって再定式化し，強化学習によるアルゴリズムを適用することによって障害発生時間データの計測を行いながら動的にチェックポイントの生成を行うアルゴリズムを提案する．

2. SMDP によるモデル化

データ処理を行う集中型ファイルシステムを考える．システムは正常な状態から稼動を開始し，保存データ量がある一定量になった時点でチェックポイントを設定するかどうかの選択を行う．本稿ではこのような選択を実行する時点を決定点と呼ぶ．チェックポイントを生成しない場合，システムはデータ処理を継続する．一方，チェックポイントを生成する場合，計算のプロセスは二つに分岐される．親プロセスはそのまま計算を続け，子プロセスはチェックポイントを生成するための処理を始める．子プロセスによるチェックポイント生成が行われている間，親プロセスの処理率は低下する．システム障害はポアソン過程に従って発生すると仮定する．障害が発生した場合，一定期間中ロールバックリカバリが行われ，直前のチェックポイントの状態へ戻った後に処理の再実行を行う．このとき，障害が発生する直前の決定点までチェックポイントの生成を行わずに処理を実行し，障害発生直前の決定点で再びチェックポイントを生成するかどうかの選択を行う（図1参照）．

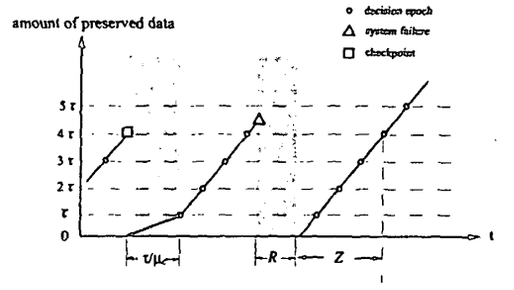

図 1: モデルの概念図．

前述したチェックポイント生成モデルに対して，セミマルコフ決定過程により問題の定式化を行う．具体的にシステム障害の発生がパラメータ $\lambda\,(>0)$ のポアソン過程に従う場合を考える．すなわち，障害発生時間間隔に対する確率分布は

$$F(t) \;=\; 1-\mathrm{e}^{-\lambda t}. \tag{1}$$

また，チェックポイントを生成するかどうかを選択する決定点において成立すべき最適性方程式を導出するため，以下の記号を定義する．

$R\,(>0)$：ロールバックリカバリに費やす時間

Z：ロールバックリカバリの後，障害直前の状態に移行するまでの処理時間（非負の確率変数）

$\alpha\,(>0)$：割引率

$\tau, 2\tau, 3\tau, \ldots$：チェックポイント生成可能な保存データ量

$\mu\,(>0)$：通常稼動中の処理率

$\mu_c\,(>0)$：チェックポイント生成中の処理率

チェックポイント生成に関する評価尺度として，総期待割引無駄時間を考える．これは時刻 t までの累積無駄時間に関する確率過程 $\{X(t);\, t>0\}$ を用いて

$$\mathrm{E}\left[\int_0^\infty \mathrm{e}^{-\alpha t}X(t)\mathrm{d}t\right] \tag{2}$$

と定義され，これを最小にする最適なチェックポイント生成アルゴリズムを構築する．

以下のような諸量を定義する．

$Q(n,\mathrm{cnt})$：n 番目の決定点においてチェックポイントを生成することを選択し，以後最適な方策を選択し続けた場合の総期待割引無駄時間

$Q(n,\mathrm{chk})$：n 番目の決定点においてチェックポイントを生成しないことを選択し，以後最適な方策を選択し続けた場合の総期待割引無駄時間

$V(n)$：n 番目の決定点における最小総期待割引無駄時間

このとき，以下の最適性方程式を得る．

$$V(n)=\min\{Q(n,\mathrm{cnt}),Q(n,\mathrm{chk})\}. \tag{3}$$

ここで

$$\begin{aligned}Q(n,\mathrm{cnt})=&\int_0^{\tau/\mu}\left\{\int_0^{t+R}\mathrm{e}^{-\alpha s}\,\mathrm{d}s+\mathrm{e}^{-\alpha(t+R)}\right.\\&\left.\times\mathrm{E}\left[\int_0^{Z}\mathrm{e}^{-\alpha s}\,\mathrm{d}s\right]+\mathrm{E}\left[\mathrm{e}^{-\alpha(t+R+Z)}\right]V(n)\right\}\mathrm{d}F(t)\\&+\int_{\tau/\mu}^{\infty}\mathrm{e}^{-\alpha\tau/\mu}V(n+1)\,\mathrm{d}F(t),\end{aligned} \tag{4}$$

$$\begin{aligned}Q(n,\mathrm{chk})=&\int_0^{\tau/\mu_c}\left\{\int_0^{t+R}\mathrm{e}^{-\alpha s}\,\mathrm{d}s+\mathrm{e}^{-\alpha(t+R)}\right.\\&\left.\times\mathrm{E}\left[\int_0^{Z}\mathrm{e}^{-\alpha s}\,\mathrm{d}s\right]+\mathrm{E}\left[\mathrm{e}^{-\alpha(t+R+Z)}\right]V(n)\right\}\mathrm{d}F(t)\\&+\int_{\tau/\mu_c}^{\infty}\left\{\int_0^{\tau/\mu_c}\left(1-\frac{\mu}{\mu_c}\right)\mathrm{e}^{-\alpha s}\,\mathrm{d}s\right.\\&\left.+\mathrm{e}^{-\alpha\tau/\mu_c}V(1)\right\}\mathrm{d}F(t),\end{aligned} \tag{5}$$

$$\begin{aligned}\mathrm{E}\left[\int_0^{Z}\mathrm{e}^{-\alpha t}\,\mathrm{d}t\right]=&\int_0^{n\tau/\mu}\left\{\int_0^{t}\mathrm{e}^{-\alpha s}\,\mathrm{d}s+\int_t^{t+R}\mathrm{e}^{-\alpha s}\,\mathrm{d}s\right.\\&\left.+\mathrm{e}^{-\alpha(t+R)}\mathrm{E}\left[\int_0^{Z}\mathrm{e}^{-\alpha t}\,\mathrm{d}t\right]\right\}\mathrm{d}F(t)\\&+\int_{n\tau/\mu}^{\infty}\left\{\int_0^{n\tau/\mu}\mathrm{e}^{-\alpha s}\,\mathrm{d}s\right\}\mathrm{d}F(t).\end{aligned} \tag{6}$$

3. 強化学習によるチェックポイント生成

強化学習 [2] とは，エージェント（学習と意思決定を行う者）が試行錯誤的に環境（制御する対象）との相互作用を学習して，環境に適応する（最適な制御を行う）ための方法論である．ニューラルネットワークのような教師付き学習と異なり，明示的な行動選択を示す教師が存在しない．その代わりエージェントは環境から報酬（費用）という情報を得ることができ，その情報を基にして環境を支配するパラメータを学習する．強化学習を適用できる環境は，一般に次の性質を持つ．(i) エージェントは環境に関する知識を予め持たない，(ii) 環境の状態遷移と報酬（費用）の発生は確率的である，(iii) 報酬（費用）は状態遷移を繰り返すことで発生する．これらの性質は，環境の動的な特性を SMDP によってモデル化可能であることを示唆しており，強化学習は SMDP によるモデル化と密接な関係がある．

ここではチェックポイント生成アルゴリズムに対して強化学習を適用する．具体的な強化学習による学習アルゴリズムとして，これまでに様々なものが提案されている．本稿では特に，代表的な Q 学習と呼ばれる強化学習アルゴリズムを適用する．Q 学習は強化学習の代表的な手法であり，SMDP による定式化と深く関連している．SMDP による定式化では，ある状態 s で行動 a を選択し，以降の選択では最適な行動を選択し続けたときの値 $Q(\cdot,\cdot)$ を基に最適性方程式を考える．この値は Q 値と呼ばれる．Q 学習は，試行錯誤を繰り返しながら Q 値の推定を行うアルゴリズムである．

具体的に，チェックポイント生成モデルに対する Q 学習のアルゴリズムは以下の通りである．

Step 1: 現在時刻 t (>0) においてシステムの状態 s_t（何番目の決定点であるか）を観測する．

Step 2: 任意の行動選択法に従って行動 a_t（処理の継続あるいはチェックポイント生成）を実行する．

Step 3: 実行から u (>0) 時間経過後，再びシステムの状態 s_{t+u}（何番目の決定点であるか）を観測する．このときの各時刻 v $(0\le v\le u)$ において無駄時間の発生記録 $X(v)$ を採取する（$X(v)$ は 0 か 1 の値）．

Step 4: 以下の更新式により Q 値を更新する．

$$\begin{aligned}Q(s_t,a_t)\ \leftarrow\ &(1-\beta)Q(s_t,a_t)\\&+\beta\left\{\int_0^{u}\mathrm{e}^{-\alpha v}X(v)\mathrm{d}v\right.\\&\left.+\mathrm{e}^{-\alpha u}\min_{a'}Q(s_{t+u},a')\right\}.\end{aligned} \tag{7}$$

ここで，$0<\beta\le 1$ は学習率と呼ばれる．

Step 5: 現在時刻を $t+u$ として Step 1 へ戻る．

上記のアルゴリズムにおける Step 2 では，Q 値に基づいて行動を選択する方法を定める必要がある．本論文では確率的な行動選択法として ε-greedy 選択を用いる [2].

ε-greedy 選択: $0<\varepsilon<1$ の確率でランダムに行動を選択する．それ以外では現在の状態 s に対応した Q 値が最小となる行動を選択する．

4. 今後の課題

本稿では強化学習により最適チェックポイントを動的に生成するためのモデル化を行った．また，導出した最適性方程式に対する強化学習のアルゴリズムを示した．今後は，システム障害の発生メカニズムが予測できないような状況において，最適チェックポイントを適応的に生成し，本手法の有効性を検証する．

参考文献

[1] N. H. Vaidya, Impact of checkpoint latency on overhead ratio of a checkpointing scheme, *IEEE Transactions on Computers*, vol. 46, no. 8 (1997).

[2] R. S. Sutton and A. G. Barto（三上貞芳，皆川雅章（共訳））：「強化学習」，森北出版 (2002).