広く分散した動的な情報のための情報共有手法の最適化

八木 哲

NTT 未来ねっと研究所

本稿では、広く分散した多数の情報源において頻繁に変化する情報を観測し、その最新の情報を広く分散した多数の利用者に提供する動的な情報共有手法を提案する、最適化について述べる。まず、情報共有手法の課題を次のとおりである。(a) 広く分散した多数の情報源に起因する、多量の更新処理。(b) 広く分散した多数の利用者に起因する、多量の参照処理。(c) 頻繁に変化する情報に起因する、高い頻度の更新処理。(d) 最新情報の共有、これらの課題に対処するために、我々は次の特徴を持つ情報共有手法を提案している。「利用者は情報の詳細を段階的に参照する」という参照パターンを前提条件として、(1) 自律型モジュールを分散配置し、配置した地域で観測された情報の管理を担当させる。(2) 各自律型モジュールは、観測された情報から詳しく述びの複数の要約を動的に生成し、利用頻度の高い要約だけを複製して、観測された情報への索引として用いる。これらにより、参照処理と更新処理を自律型モジュールに分散・局所化する、自律型モジュールを駆る更新処理の頻度を削減する。換言すれば、自律型モジュール間の通信頻度を削減する。しかし我々の情報共有手法では、自律型モジュール間の通信頻度を削減できないことがある。本稿では、要約の冗長な転送を排除する方法を示す。

1 はじめに


(a) 広く分散した多数の情報源に起因する、多量の更新処理。
(b) 広く分散した多数の利用者に起因する、多量の参照処理。
(c) 頻繁に変化する情報に起因する、高い頻度の更新処理。
(d) 最新情報の共有。

従来、次のような情報共有手法が用いられている[10][11][12][13][14]。課題 (a) のために、自律型モジュールを分散配置し、配置した地域で観測された情報の管理を担当させる。これにより、観測された情報に対する更新処理を分散・局所化する。課題 (b) のために、自律型モジュールは、観測された情報を互いに複製あるいはキャッシュする。これにより、観測された情報に対する参照処理を分散・局所化する。このときの課題 (c) は「複製やキャッシュに対する高い頻度の更新処理」である。そこで課題 (c) のために、複製やキャッシュに対する更新処理の頻度を低減する。しかし複製やキャッシュが最新情報を持続し難くなるために、課題 (d) には対処し難しい。情報を時刻印や有効期限を付すすれば無効な情報を棄却できるが、課題 (d) に対処できるわけではない。

これに対して、我々は次の特徴を持つ情報共有手法 [15][16] を提案している。「利用者は情報の詳細を段階的に参照する」という参照パターンを前提条件として、(1) 自律型モジュールを分散配置し、配置した地域で観測された情報の管理を担当させる。(2) 各自律型モジュールは、観測された情報から詳しく述びの複数の要約を動的に生成し、利用頻度の高い要約だけ
を相互に複製して、観測された情報への索引として用いる。これにより、参照処理と更新処理を各自律型モジュールに分散・局所化する。自律型モジュールを跨る更新処理の順序を削減する。

両手法の方針は、複数の自律型モジュールの使用と、自律型モジュール間の依存関係の削減である。換言すれば、自律型モジュール間の通信順序の削減である。そこで「自律型モジュール間の通信順序」を尺度として両手法を比較すれば、次のようである。我々の手法では、自律型モジュール間の間の要約が冗長に転送される場合があり、この場合には従来手法よりも「自律型モジュール間の通信順序」を削減できないことがある。そこで、本稿では要約の冗長転送を排除する方法を示す。まず、我々の情報共有手法の枠組(2章)と実現方の概要(3章)を示す。次に、要約の冗長転送を排除する問題をモデル化(4章)、解法を示す(5章)。また、解法を考察する(6章)。最後に、本稿の内容をまとめ、今後の課題を示す(7章)。

2 情報共有手法の枠組

前提条件とする「利用者は情報の詳細を段階的に参照する」という参照パターンの詳細を示し、これに基づいて情報共有手法の枠組を示す。

- 各利用者は、第1段階では変数の情報の概要を参照する。第2段階では興味を持った情報の詳細を参照する。
- 各情報は、第1段階では変数の利用者から概要を参照される。第2段階では興味を持った利用者から詳細を参照される。

参照パターンの第1段階では、各情報の概要が参照の対象である。利用者が関連的参照するために、各情報の参照順序は比較的高い。ここで、各情報源において観測された頻度に変化する情報から、次の特徴を持つ情報を動的に生成し、利用者に提供する。

(1) 視測した情報の特徴だけを反映する。
(2) 視測した情報と比べて時間的な変化が少ない。

この情報を「抽象化した情報」と呼ぶことにする。「抽象化した情報」の複製を利用者に近い距離に配置することで、「抽象化した情報」に対する参照処理を分散・局所化する。「抽象化した情報」の(1)(2)の特徴により、

「抽象化した情報」の複製に対する更新処理の順序を削減する。参照パターンの第2段階では、各情報の詳細が参照の対象である。利用者が選択的に参照するために、各情報の参照順序は比較的高い。そこで、各情報源において観測された頻度に変化する情報を、そのまま利用者に提供する。この情報を「索な情報」と呼ぶことにする。「索な情報」を情報源のある地域ごとに個別管理することで、「索な情報」に対する更新処理を分散・局所化する。「抽象化した情報」を対象として、参照する「索な情報」を利用者に選択させることにより、「索な情報」に対する参照処理を分散させる。

このような枠組を持つ情報共有手法を課題(a)(b)(c)(d)に照らし、次のようなである。課題(a)のためには、自律型モジュールを分散配置し、配置した地域の「索な情報」の管理を担当させる。これにより、「索な情報」に対する更新処理を分散・局所化する。課題(b)のために、各自律型モジュールは、「索な情報」から「抽象化した情報」を生成し、これを相互に複製して、「索な情報」への索引として用いる。これにより、「抽象化した情報」に対する参照処理を分散・局所化する。また「索な情報」に対する参照処理を分散する。課題(c)(d)のために、前記(1)(2)の特徴を持つ「抽象化した情報」を複製の対象とする。これにより、複製に対する更新処理の順序を削減した上で、複製が最新の情報を保持できるようにする。

3 情報共有手法の実現法の概要

3.1 構成

2章で示した情報共有手法の実現には、「抽象化した情報」を生成し、これを自律型モジュールの間で複製するための、体系的な仕組みが必要である。ここでは、2章で示した情報共有手法に基づいた情報共有基盤であるWISE (Wide-area Information Sharing Engine) [16]の要領を示す。まず、WISEの構成を図1に示す。地域ごとにセンサとRM(Resource Manager)
がある。センサは、情報源において情報を観測し、観測した情報を時刻印を押して、RMに渡す。RMは、2章で示した自律型モデルに相当する。利用者は、近傍のRMを介して、情報を参照する。RMには3つの中核的な機能要素がある。データベースは、‘素な情報’と‘抽象化した情報’を格納する。抽象化機能は、‘抽象化した情報’を生成し、これを他のRMに複製する。このとき、RMが相互に複製する‘抽象化した情報’が持つ情報量と、前提条件とする参照パターンの第1段階に必要な情報量とが一致しない場合がある。前者が少ない場合には、利用者は近傍のRMから保有する‘抽象化した情報’から他のRMが保有する‘素な情報’を参照する。この結果、参照処理を各RMに分散・局所化できない。前者が多い場合には、RMは利用されない‘抽象化した情報’を他のRMに複製する。この結果、複製に対する更新処理の頻度を削減できない。そこで学習機能は、他のRMに複製すべき抽象化した情報’を参照頻度に基づいて選定する。次節以降では、これら3つの機能要素の概要を示す。

3.2 ‘素な情報’と‘抽象化した情報’を格納するデータベース

まず、‘素な情報’と‘抽象化した情報’を、次に示す単位データdの集合として表す。

- d = (g, t, a, s, v)
  - g：地域、dが生成された地域を示す、識別名[12]を用いる。
  - t：時刻、dの生成時刻を示す、単位時間ごとに単調増加するシーケンス番号を用いる。
  - a：抽象度、値が大きいほど抽象化されており、よりいっそう、観測した情報の特微だけを反映し、観測した情報と比べて時間的な変化が少ないことを示す、‘素な情報’は0、‘抽象化した情報’は正の整数。
  - s：情報源、d生成した情報源を示す、gに対する相対識別名[12]を用いる。
  - v：値。

前提条件とする参照パターンに則して利用者が単位データを参照ように、次に示すWISE情報モデル(IMW:Information Model for WISE)と呼ぶグラフ構造をdの集合に与える。

- Nはノードの集合、N = {nf | df | dfは、地域がgj、抽象度がafである単位データ
  - AAは、ノードの抽象度の高低関係を表すアーカの集合、AA = {naf, nbf} | nafは、抽象度ahの単位データを要素として持つノード、nbfは、抽象度alの単位データを要素として持つノード、ah > al
  - AGは、ノードの地域の隣接関係を表すアーカの集合、AG = {naf, nbf} | nafは、抽象度alの単位データを要素として持つノード、nbfは、抽象度a2の単位データを要素として持つノード、a1 = a2}。

グラフ構造を与えられたdの集合を、IMWのインスタンスと呼ぶこととする。図2に示す。この例では、抽象度0、抽象度1、抽象度2の単位データを、それぞれノードを構成している。抽象度の大きい単位データから構成されるノードほど、地変的な地域情報を持つ、抽象化機能が、抽象度0の単位データから抽象度1以上の単位データを生成する(3.3節参照)。利用者は、次のように単位データを参照する。

1. 全地域の情報の概観を参照する。すなわち、ノードの地域の隣接関係を示すアーカをたどりながら、抽象度2の単位データを参照する。
2. 興味のある地域を選択する。すなわち、抽象度2の単位データを要素として持つノードのなかから、興味のある地域のノードを選択する。
3. 興味のある地域の情報の詳細を参照する。すなわち、選択したノードからノードの抽象度の高低関係を示すアーカをたどり、抽象度1の単位データを参照する。
4. 同様にして、抽象度0の単位データを参照する。

<table>
<thead>
<tr>
<th>抽象度2の単位データから構成されるノード群。</th>
<th>抽象度1の単位データから構成されるノード群。</th>
<th>抽象度0の単位データから構成されるノード群。</th>
</tr>
</thead>
<tbody>
<tr>
<td>RMが保存、RMが保存、RMが保存</td>
<td>保存、保存</td>
<td>保存、保存</td>
</tr>
</tbody>
</table>

図2：IMWのインスタンス
3.3 ‘抽象化した情報’を生成する抽象化機能

抽象化機能の動作を説明する。基本となる動作は、ノード \( n_{at} \) を構成する抽象度 \( al \) の単位データから、ノード \( n_{ah} \) を構成する抽象度 \( ah(>al) \) の単位データを生成する動作である。複数のノードを構成する単位データから、一つのノードを構成する単位データを生成することもある。この動作は次の特徴を持つ、

1. \( n_{at} \) を構成する単位データを表す情報の特徴だけを、\( n_{ah} \) を構成する単位データを表す情報に反映する。
2. \( n_{at} \) を構成する単位データの値が変化すると、\( n_{ah} \) を構成する単位データの値は変化しない。このようにする。以上の動作を抽象化動作と呼ぶことになる。具体的には、次のような動作である。

- 精度削減型の動作：単位データの値の有効桁数を削減することで、単位データを生成する。単位データの値を予め定めた種別に分類し、種別名を値とすることで、単位データを生成する。
- 要素選択型の動作：注目度の高い情報・一次判別に必要な情報・情報の概説・情報の定義情報など、特徴的な情報を表す単位データを選択することで、単位データを生成する。
- 要素集約型の動作：単位データの値の集合に対して統計関数・情報的な関数を適用することで、単位データを生成する。

抽象化機能は、IMW のインスタンスに対して抽象化動作を行う。すなわち、抽象度 0 の単位データの更新を契機として、更新された抽象度 0 の単位データから、ノードの抽象度の高精度を示すデータにより対応付けられた。抽象度 1 の単位データを生成する。生成した抽象度 1 の単位データから、ノードの抽象度の高精度を示すデータにより対応付けられた。抽象度 2 の単位データを生成する。これを繰り返し、連鎖反応的に単位データを生成する。したがって、生成した単位データを他の RM に複製する。

3.4 複製すべき‘抽象化した情報’を選定する学習機能

学習機能の動作を説明する。学習機能は、1 章で示した尺度である「自律型モデル間の通信頻度」に基づいて、RM が保有する単位データの原本のなかから、他の RM に複製すべき単位データを選定する。RM 1 の学習機能は、単位データ \( d \) を RM 1 から RM 2 に複製か否かを、次のように判断する。

- \( r \) は、RM 2 の近傍の利用者による、RM 1 が保有する \( d \) の原本と RM 2 が保有する \( d \) の複製に対する参照頻度。
- \( u \) は、RM 1 による、\( d \) に対する更新頻度。
- \( r \) と \( u \) を用いて次のように判断する。厳密には通信プロトコルに依存した重み付けが \( r \) と \( u \) に必要であるが、ここでは単純化のために省略した。

- \( r \geq u \) の場合、\( d \) を RM 1 から RM 2 に複製する。
- \( r < u \) の場合、\( d \) を RM 1 から RM 2 に複製しない。

4 問題のモデル化

学習機能は、単位データごと、あるいはノードごとに、複製するか否かを選定する。従って、同じ抽象度 0 の単位データから生成された抽象度の異なる複数の単位データが、同一の RM に複製されたり同一の RM を介して参照されることがある。このとき、抽象度の高い単位データは抽象度の低い単位データから生成可能であるという意味において、単位データは単純に転送されている。しかし、特定の抽象度の単位データだ
けを複製し、より抽象度の高い単位データを複製先のRMにおいて生成すれば、単位データの冗長的な転送を排除できる。以上の問題をモデル化する。

(1) IMWのインスタンスから、ノードの抽象度の高低関係を示すアーケとノードから構成される木を抽出する。図3に例示する。ここでは、RM1においてRM1からRM2への単位データの冗長な転送を排除する場合を取り上げる。

(2) RM1が原本を保有する単位データから構成されるノードに、属性として「参照頻度」・「更新頻度」・「転送状態」を設ける。

(3) 「参照頻度」は、RM2の近傍の利用者による、ノードを構成する単位データの参照頻度。更新頻度は、RM1による、ノードを構成する単位データの更新頻度。「転送状態」は「参照」・「複製」・「生成」の3値を取る。「通信頻度」は「参照頻度」に等しい。

(4) 「転送状態」が「参照」であるとき、ノードを構成する単位データをRM1に複製しない、RM2の近傍の利用者は、RM1が保有する単位データの原本を参照する。「通信頻度」は「参照頻度」に等しい。

(5) 「転送状態」が「複製」であるとき、ノードを構成する単位データをRM2に複製する。RM2の近傍の利用者は、RM2が保有する単位データの複製を参照する。「通信頻度」は「更新頻度」に等しい。

(6) 「転送状態」が「生成」であるとき、このノードの子であるノードの「転送状態」が「複製」か「生成」である。このノードを構成する単位データはRM2において生成される。RM2の近傍の利用者は、RM2において生成された単位データを参照する。「通信頻度」は0である。

(7) 「通信頻度」の合計値を最小にする、各ノードの「転送状態」を求める。

5 解法

5.1 概要

解法の概要を、単純な例題を用いて示す。図4に例題を示す。「参照頻度」は、葉から葉へ均一に減少している。すなわち、单位時間あたりに一定数の利用者が、葉から葉へ均一に分散しながら単位データを参照している。「更新頻度」は、葉から根へ均一に減少している。すなわち、抽象化機能が、葉から根へ単位データの時間的な変化が均一に減少するように、単位データを生成している。このとき、「通信頻度」の合計値を最小にする、各ノードの「転送状態」を求めめる。この例題では、抽象度が等しいノードの「参照頻度」が「更新頻度」が等しいために、抽象度が等しいノードを一つの仮想的なノードと見なすことで（単純仮想ノードと呼ぶことにする）、問題を単純化できる。次のパラメータを用いて解法の概要を示す。

・ $r_0$ ($a = 0, 1, 2, 3$) は、抽象度が$a$である単純仮想ノードの「参照頻度」、単純仮想ノードを構成するノードの「通信頻度」の合計値である。

・ $u_a$ ($a = 0, 1, 2, 3$) は、抽象度が$a$である単純仮想ノードの「更新頻度」、単純仮想ノードを構成するノードの「更新頻度」の合計値である。

まず、抽象度が$a$である単純仮想ノードを複製することにすれば、この単純仮想ノードの「転送状態」は「複製」であり、抽象度が$a$より高い単純仮想ノードの「転送状態」は「生成」になる、換言すれば、抽象度が$a$より低い単純仮想ノードの「転送状態」は「参照」になる。このときの「通信頻度」の合計値 $T_{ar}$ は、次のとおりである。

$$ T_{ar} = \left\{ \begin{array}{ll} u_0 & a = 0 \\ \sum_{i=0}^{a-1} r_i + u_a & a = 1, 2, 3 \end{array} \right. \quad (1) $$

$T_{ar}$ の最小値は、$T_0$ から $T_3$ へ昇べきの順に探索することで、求めることができる。こうして求めた $T_{ar}$ の最小値と、全てのノードの「転送状態」を「参照」にした場合の「転送頻度」の合計値とで、値が小さい方における「転送状態」が解である。
5.2 アルゴリズム

一般には、前節の例題のように「参照頻度」と「更新頻度」が均一に変化とは限らない。そこで、木を
振りかけ順になぞりながら、段階的に解を求める。まず、葉を含む高さ 2 の部分木 \( T_{ij} \) \((i = 0, 1, 2, ..., n)\) における解を求める。次に示す \( T_{ij} \) \((i = 0, 1, 2, ..., n)\) の
状態 \((1)(2)(3)\) から、「通信頻度」の合計値が最小の状態を選択することで、5.1 節の単純な例題と同様に解
を求めることができる。

- 状態 (1) : 子であるノードを複製する状態、親であ
るノードの「転送状態」は「生成」、子であるノードの「転送状態」は「複製」。
- 状態 (2) : 親であるノードを複製する状態、親であ
るノードの「転送状態」は「複製」、子であるノードの「転送状態」は、通信頻度が小さくなるように「複
製・参照」から個別に選択。
- 状態 (3) : ノードごとに複製するか否かを選択した
状態、各ノードの「転送状態」は、「通信頻度」が小さ
くなるように「複製・参照」から個別に選択。

次に、\( T_{ij} \) \((i = 0, 1, 2, ..., n)\) を仮想的なノードと
見なし（仮想ノードと呼ぶことにする）、仮想ノードと
その親であるノードから構成される高さ 2 の部分木 \( T_{2j} \)
\((j = 0, 1, 2, ..., m)\) における解を求める。次に示す仮
想ノードの状態 \((a)(b)\) を求めれば、\( T_{2j} \) \((j = 0, 1, 2, 
..., m)\) における、状態 \((1)(2)(3)\) に関する状態を求
めることができる。状態 \((1)(2)(3)\) に関する状態か
ら、通信頻度の合計値が最小の状態を選択することで、\( T_{ii} \) \((i = 0, 1, 2, ..., n)\) の場合と同様に解を求める
ことができる。

- 状態 (a) : 「転送状態」が「複製」であるノードに相当
する状態、これは、仮想ノードの親であるノードを
構成する単位データが、複製される RM において生
成可能な状態である。仮想ノードとみなした部分木
の状態が状態 \((1)(2)\) であるとき、部分木の根である
ノードを構成する単位データは複製される RM にあ
り、条件を満たす。そこで、部分木の状態 \((1)(2)\) か
ら、「通信頻度」の合計値が小さい方の状態を選択す
る。このとき、部分木を構成する各ノードの「転
送状態」を「複製転送状態」、「通信頻度」の合計値を
「複製通信頻度」と呼ぶこととする。
- 状態 (b) : 「通信頻度」が小さくなるように「複製・参照」から「転送状態」を選択したノードに相当する

5.3 適用例

課題 (a)(b)(c)(d) を含む分かりやすい例題として、次のような広域ネットワーク監視サービスを取り上げ
る。監視対象は 4 つの組織のネットワークである。各
ネットワークには 8 つのサブネットがある。このとき、
各サブネットにおけるネットワーク・ルータ・サーバ
の負荷情報を監視し、サービスのユーザビリティを示す
情報、あるいは性能管理や障害管理のための情報と
して、各組織の利用者と管理者に提供する。

我々の情報共有手法を用いる場合には、負荷情報
を定期的に収集するセンサをサブネットごとに置き、
収集した負荷情報を管理する RM をネットワークごと
に置く。これは図 1 と同様の構成である。各 RM を RM_1・RM_2・RM_3・RM_4 と呼ぶことにする。RM_1・RM_2・RM_3・RM_4 は、図 3 と同様のLMW のインタース
スを用いて、抽象度の高い単位データをサービスのユー
ザビリティを示す情報として提供する。抽象度の低い
単位データを性能管理や障害管理のための情報として
提供する。ここでは、RM_1 において RM_1 から RM_2
への単位データの冗長な転送を排除する場合を取り上
げ、付録に示した疑似コードの実行過程を示す（図5参
照）。段階 0 では、問題を未で表している。サービスの
ユーザビリティを示す情報が「高負荷」を示したため
に、普段は参照されない性能管理や障害管理のための
情報も参照されているという、参照頻度が比較的高く
なる状況を想定した。段階 1 では、葉を含む高さ 2 の
部分木を仮想ノードとして、5.2 節で示した状態 (a)(b)
6 考察

前提条件とする「利用者は情報の詳細を階段的に参照する」という参照パターンを利用して、「自律型モデル間の通信頻度」を削減するには、利用者が必要とする詳しく応じた情報だけを、自律型モデル間の間で転送すれば良い。従来の情報共有手法では、SNMP マネージャが予め定められた情報の相互に複製するために、前記のような転送はできない。我々の情報共有手法では、RM が抽象度の異なる単位データを動的に生成し、利用頻度の高い単位データだけを複製するために、前記のような転送が可能になる。しかし利用者が詳しい情報を頻繁に必要とする場合には、本稿で示したアルゴリズムを適用しても、RM は抽象度の低い単位データを相互に複製するために、従来手法との差は減少する。この場合には、共有する情報の詳細さを制限する方法の利用が考えられる。たとえば「自律型モデル間の通信頻度」に対する RM の処理能力やネットワークの帯域の余力を考慮したことと契機として、抽象度の低い単位データから順に共有を中断し、各 RM に関連した局所的な利用に切り替えられる。

7 おわりに

本稿では、広く分散された多数の情報源において頻繁に変化する情報の観察、その最新の情報を広く分散された多数の利用源に提供するための情報共有手法における、最適化について述べた。本稿では、RM の間の単位データの冗長な転送を排除する方法を示した。今後の課題として、RM の間で単位データを交換するためには、即時性・信頼性・転送効率などの要件を満たす通信プロトコルの確立がある。

参考文献

付録 疑似コード

「」で囲った文字列は各ノードに付属する変数である。「」で囲った文字列は定数である。

単位データの冗長な転送を排除する（）
{ 状態 (a) (b) を求めること（根であるノード）。

状態 (a) (b) を求めること（ノード）
{ // 通信頻度が小さくなるように "参照" と
//"複製"から転送状態を選択
if ( 通信頻度 < 更新頻度 ) {  
  通信頻度 = "参照";
  転送状態 = "参照";
} else {  
  通信頻度 = "更新頻度";
  転送状態 = "複製";
  }

if ( ノードは葉である ) {  
// このノードだけでは仮想ノードを作る場合。
// 状態 (a) (b) における各値を求める。
  最善通信頻度 = '通信頻度';
  最善転送状態 = '転送状態';
  複製通信頻度 = '更新頻度';
  複製転送状態 = '複製';
} else {  
// 真の子孫とともに仮想ノードを作る場合。
// 子であるノードを移動。
  子であるノードに対して順に「状態 (a) (b) を
求める ()」を適用；

// 状態 (a) (b) における各値を求める。
  最善通信頻度の合 =
  子であるノードの最善通信頻度の合;
  複製通信頻度の合 =
  子であるノードの複製通信頻度の合;
if ( 複製転送頻度の合 ≤
  最善転送頻度の合 + 通信頻度 )
{  // 状態 (b) として状態 (1) を選択する場合。
    最善通信頻度 = '複製転送頻度の和';
    最善転送状態 = '生成';
  // 状態 (a) として状態 (1) を選択。
    複製通信頻度 = '複製転送頻度の和';
    複製転送状態 = '生成';
}

// 真の子孫であるノードにおいて、複製転送
状態 を、最善転送状態 に代入する;
}
else {  
// 状態 (b) として状態 (2) か状態 (3) を
// 選択する場合。
  最善通信頻度 = '複製転送頻度の和 + 通信頻度';
  最善転送状態 = '転送状態';
if ( 複製転送頻度の合 ≤
  最善転送頻度の合 + 更新頻度 )
{  // 状態 (a) として状態 (1) を選択する場合。
    複製通信頻度 = '複製転送頻度の和';
    複製転送状態 = '複製';
  // 状態 (a) として状態 (2) を選択する場合。
    最善通信頻度 = '複製転送頻度の和 + 更新頻度';
    複製転送状態 = '複製';
}

// 真の子孫であるノードにおいて、最善転送
状態 を、複製転送状態 に代入する;
}
}