非同期通信モデルにおける 6F-05 分配・計算・収集操作のデータサイズ依存性*

渋沢 進†
茨城大学工学部‡

1 はじめに

プロセッサ間で1対多，多対1，多対多の通信を行うデータの分配，収集，ブロードキャスト，完全交換などの通信操作を行う集合通信は，並列分散処理において中心的な役割を果たす．集合通信は，複数のプロセッサを用いた時空間適応処理や仮想物体の衝突検出，データマイニングなどの各種応用にごく普通に現れる．

これまで，集合通信の並列アルゴリズムに関する研究では，しばしば通信ステップが同期的であると仮定して行われてきた．しかし，実際のプロセッサ間通信は非同期に近い状態で実行されており，同期的並列アルゴリズムと実際との間に大きな差異が存在する．このため，より実際に近い通信モデルを用いて並列アルゴリズムを設計し，評価することが求められている [1]．

プロセッサ間の非同期通信モデルとしては，これまで Log P モデルや BSP モデルなどが提案されてきた [2]．本報告では，これまでのモデルとは異なる通信パラメータを用いた非同期通信モデルを用いて，初期データが1プロセッサにあるときの分配・収集操作の実行時間を考察している．

2 プロセッサ間の非同期通信モデル

メッセージ通信において，1プロセッサから異なるメッセージを同時に複数のプロセッサに送ることはなく，1プロセッサで一度に受信できるメッセージは1つと仮定する．また，1プロセッサから他のプロセッサへの1対1通信の時間は，あるメッセージサイズに対して一定であると仮定する．

プロセッサ P_{send} から P_{recv} にメッセージ M を送信するとき，非同期通信モデルによるメッセージ通信の時間変化を図1のようにモデル化する [1]．図の矢印は送受信操作の開始と完了の対応を表す．図中の通信パラメータ δ, ε, ϕ をそれぞれ受信操作の開始遅れ時間，完了遅れ時間，送受信操作の重なり時間とよぶ．非同期通信の送受信操作時間を次のように定義する．

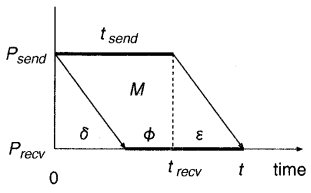

図 1: 非同期通信モデルでのメッセージ通信

[定義 1] 非同期通信モデルで，1プロセッサから他プロセッサにサイズ m のメッセージを送るのにかかる送受信操作時間 t_{send}, t_{recv} を次式で表す．

$$t_{send} = \sigma_s + m\tau_s, \quad t_{recv} = \sigma_r + m\tau_r \tag{1}$$

ここに，σ_s, σ_r は送受信データのサイズに依存しない定数であり，τ_s, τ_r はデータサイズの比例定数である．□

通信パラメータ δ について次のように仮定する．

[仮定 1] メッセージの送信中にネットワークの状態が変化しない定常状態では，受信操作の開始遅れ時間 δ はメッセージサイズに依らず一定とする．□

図1より，メッセージの通信時間とパラメータの間に次の関係が成り立つ．

[補題 1] 非同期通信モデルにおけるメッセージの通信時間 t，送受信操作時間 t_{send}, t_{recv}，通信パラメータ δ, ϕ, ε の間に次の関係がある．

$$t_{send} = \delta + \phi, \quad t_{recv} = \phi + \varepsilon \tag{2}$$

$$t = t_{send} + \varepsilon = \delta + t_{recv} = \delta + \phi + \varepsilon \tag{3}$$

特に $t_{send} = t_{recv}$ のとき，$\varepsilon = \delta$ である．□

3 データの多重分配

サイズ D のデータが初期プロセッサ P_0 に置かれているとき，これを異なるサイズに分割して，p プロセッサに分配する場合を考察する．分配において，異なるプロセッサ対で並列にデータを送受信する方法を多重分配とよぶ．以下では，2項木による多重分配を考察する．プロセッサ間結合がリング結合である場合，通信の衝突が生じないような多重分配を考えることができ，この場合を考察する．

プロセッサ P_0 にあるサイズ D のデータを分割して，P_0 からプロセッサ P_j $(0 \le j \le p-1)$ にサイズ d_j のデータを分配するとき，$D = \sum_{j=0}^{p-1} d_j$ である．正の整数 k に対して，$p = 2^k$, $1 \le i \le k$, $0 \le j_1, j_2 \le p-1$ とし，第 i 分配ステップで，プロセッサ P_{j_1} から P_{j_2} にサイズ $d^{(i)}_{j_1 j_2}$ のデータを送るときの通信時間，P_{j_1} での送信操作時間，P_{j_2} での受信操作時間，受信操作の完了遅れ時間を，それぞれ $t^{(i)}_{j_1 j_2}$, $t^{(i)}_{j_1 j_2 send}$, $t^{(i)}_{j_1 j_2 recv}$, $\varepsilon^{(i)}_{j_1 j_2}$ とおく．受信操作の開始遅れ時間 δ は，仮定1より一定とする．第 i 分配ステップで，プロセッサ P_{j_1} から P_{j_2} にサイズ $d^{(i)}_{j_1 j_2}$ のデータを送信するのにかかる両プロセッサでの時間 $s^{(i)}$ は，補題1より，$t^{(i)}_{j_1 j_2 send} = t^{(i)}_{j_1 j_2 recv}$ のとき，次の2通りの値をもつ．

- P_{j_1} で，$s^{(i)} = t^{(i)}_{j_1 j_2 send}$
- P_{j_2} で，$s^{(i)} = t^{(i)}_{j_1 j_2} = t^{(i)}_{j_1 j_2 send} + \varepsilon^{(i)}_{j_1 j_2} = t^{(i)}_{j_1 j_2 recv} + \delta$

プロセッサ P_0 がデータを送信し始めてから，プロセッサ P_j が最終ステップでサイズ d_j のデータを受け取るまでの時間を t_j とおけば，$t_j = \sum_{i=1}^{k} s^{(i)}$ である．プ

*The data-size dependancy of scatter, computation and gather in an asynchronous communication model

†Susumu Shibusawa
‡Ibaraki University, Hitachi, Ibaraki 316-8511, Japan

ロセッサ P_j に分配されるデータの経路係数を次のように定義する．

[定義 2] $1 \leq i \leq k$, $j_1 \neq j_2$ に対して，k ステップでプロセッサ P_j に分配されるデータが，第 i 分配ステップでプロセッサ P_{j_1} に保たれるか，または P_{j_1} から P_{j_2} に送信されるとする．このとき，P_j に分配されるデータの第 i 分配ステップでの経路係数 $\mu_j^{(i)}$ は，P_{j_1} に保たれるとき値 0 であり，P_{j_1} から P_{j_2} の送信されるとき値 1 であるとする．また，k ステップでプロセッサ P_j にデータを分配する経路を次のような 2 進数 μ_j で表す．

$$\mu_j = [\mu_j^{(1)}\ \mu_j^{(2)}\ \cdots\ \mu_j^{(k)}] \tag{4}$$

□

定義 2 より，第 i 分配ステップでの送信にかかる時間は，経路係数を用いて次のように表される．

$$s^{(i)} = t^{(i)}_{j_1 j_2 send} + \mu_j^{(i)} \varepsilon^{(i)}_{j_1 j_2} \tag{5}$$

第 i 分配ステップでプロセッサ P_{j_1} から P_{j_2} に送るサイズ $d^{(i)}_{j_1 j_2}$ のデータが，最終ステップで P_j に分配されるとき，番号 j の最小値 $j^{(i)}_{min}$ と最大値 $j^{(i)}_{max}$ は，次のように表される．

$$j^{(i)}_{min} = 2^{k-i+1} \lfloor \frac{j_1}{2^{k-i+1}} \rfloor + 2^{k-i} \tag{6}$$

$$j^{(i)}_{max} = 2^{k-i+1} (\lfloor \frac{j_1}{2^{k-i+1}} \rfloor + 1) - 1 = j^{(i)}_{min} + 2^{k-i} - 1 \tag{7}$$

プロセッサ P_j への分配時間 t_j は，$t^{(i)}_{j_1 j_2 send} = t^{(i)}_{j_1 j_2 recv}$ のとき，次のようになる．

$$t_j = \sum_{i=1}^{k} t^{(i)}_{j_1 j_2 send} + |\mu_j| \delta \tag{8}$$

ここに，$|\mu_j|$ は 2 進系列 μ_j の 1 の数である．

[補題 2] プロセッサ P_0 にあるデータを P_j にサイズ d_j ずつ多重分配するとき，第 i 分配ステップで，プロセッサ P_{j_1} から P_{j_2} にデータを送る送受信操作時間をそれぞれ $t^{(i)}_{j_1 j_2 send}$, $t^{(i)}_{j_1 j_2 recv}$ とする．プロセッサ P_j への分配時間 t_j は，$t^{(i)}_{j_1 j_2 send} = t^{(i)}_{j_1 j_2 recv}$ のとき，

$$t_j = k\sigma_s + (\sum_{i=1}^{k} \sum_{j=j^{(i)}_{min}}^{j^{(i)}_{max}} d_j)\tau_s + |\mu_j|\delta \tag{9}$$

ここに，経路係数 μ_j の第 i 桁 $\mu_j^{(i)}$ は次のように表される．

$$\mu_j^{(i)} = w_{k-i} \tag{10}$$

□

4 データの多重通信操作

P_0 を根とする 2 項木による P_j から P_0 への収集について，次の性質が成り立つ．

[補題 3] プロセッサが P_1, P_2, $\cdots$, P_{p-1}, P_0 の順に並び，P_0 を根とする 2 項木によって，P_j から P_0 にデータを多重収集するとき，収集データサイズが小さく，収集の送受信操作時間がそれぞれ一定値 σ_s, σ_r とみなすことができ，かつ $\sigma_s = \sigma_r$ ならば，P_j からの収集時間 t_j は，

$$t_j = k\sigma_s + |\mu_j|\delta \tag{11}$$

ここに，$w' = ((j-1) \bmod k) = [w'_{k-1}\ \cdots\ w'_1 w'_0]$ とするとき，μ_j の第 i 桁 $\mu_j^{(i)}$ は次のように表される．

$$\mu_j^{(i)} = \overline{w'_{k-i}} \tag{12}$$

□

各プロセッサへの分配データサイズが等しい場合の単一方向の例を図 2に示す．図において，水平方向の破線はプロセッサが待ち状態にあることを示す．

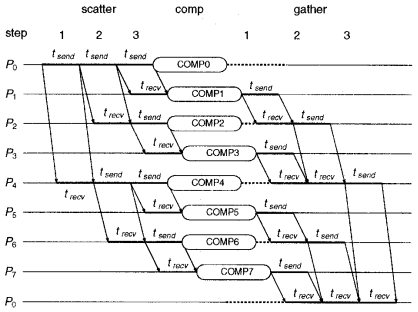

図 2: 非同期通信による単一方向の操作の例

[定理 1] [1] 非同期通信による多重分配・計算・収集の操作において，各プロセッサに分配するデータサイズが等しく，$t^{(i)}_{j_1 j_2 send} = t^{(i)}_{j_1 j_2 recv}$ のとき，分配と収集のデータの流れが単一方向の操作は，双方向の操作よりつねに高速である．□

各プロセッサへの分配データサイズを任意とした場合に，次の性質が成り立つ．

[補題 4] P_0 を根とする 2 項木によって，P_j から P_0 にデータを多重収集するとき，収集データサイズが小さく，収集の送受信操作時間がそれぞれ一定値 σ_s, σ_r とみなすことができ，かつ $\sigma_s = \sigma_r$ とする．P_1 での収集開始時刻を g_1 とするとき，P_j での収集開始時刻 g_j は，

$$g_j = g_1 + (k - |\mu_j|)\delta \tag{13}$$

ここに，μ_j は式 (12) を満たす．□

5 まとめ

本報告では，全データが 1 プロセッサにある場合に，各プロセッサへの分配データサイズを任意にしたときの多重分配と収集の時間について考察した．単一方向の操作は，双方向の操作に較べて，初期プロセッサ以外のプロセッサでの待ち時間の差が小さく，このため分配データのサイズの差を小さくできる．今後，計算時間も含めた全実行時間のさらに詳しい考察が必要である．また，非同期通信モデルにおける異なるプロセッサ間での計算と通信の重ね合わせの一般的な考察も重要である．

謝辞 ご討論頂く研究室の皆様に感謝致します．

参考文献

[1] 渋沢進, “非同期通信モデルを用いた分配・収集操作の一評価,” 信論 (D), Vol.J82-D-I, No.12, pp.1403-1407, 1999.

[2] D.E Culler, et al., “Log P: towards a realistic model of parallel computation,” *Proc. 4th ACM SIGPLAN Symp. on Principles and Practice of Parallel Programming*, pp.1–12, 1993.