MPI/SPにおけるクラスタ統合管理方式の設計

5K−2

村山 和宏，落合 真一，山口 義一
三菱電機(株) 情報技術総合研究所

1. はじめに

我々はデータ処理・信号処理演算を行う大規模クラスタのためのMPI（Message Passing Interface）開発を行っている。現在開発中のMPI/SP（MPI for Signal Processing）では、従来のプロセッサ間通信だけでなく、システム制御情報の交換を行うことによる信号処理用の特殊な計算機配置でのシステム統合化を実現する。

本稿ではMPI/SPの機能の1つであるクラスタ統合管理方式について述べる。

2.背景

2.1. ターゲットシステムの特徴

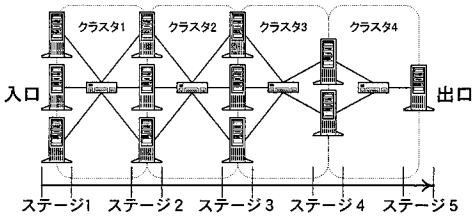

図1：クラスタ構成概要

本研究でターゲットとしているシステムは信号処理用に最適化された計算機配置構成となっており、以下の特徴を持つ。

- 膨大な計算機数による構成…信号処理では複雑なデータ処理を実行するため、数100台もの計算機によって構成されている。
- クラスタを鎖状に連結した（Chain of Clusters）構成…信号処理ではセンサから送信されるデータに対し、「入口」から「出口」に向けてパイプライン処理を行う。本システム構成ではシステムの計算機数が膨大であるため、全計算機を同一ネットワークで接続することができない。そこで、一部の計算機を複数のクラスタに所属させてクラスタを鎖状に連結し、システムを構築している（図1）。

The design of cluster-integration on MPI/SP
Kazuhiro MURAYAMA, Shinichi OCHIAI, Yoshikazu YAMAGUCHI
Mitsubishi Electric Corporation

2.2. 課題

2.1節で示した特徴を持つ計算機システムで、MPIを用いて並列信号処理を行う場合、以下の2つの課題が考えられる。

- 全クラスタの動作状態把握…本例のようにクラスタが鎖状に連結されている場合、各クラスタは隣接するクラスタの動作状態（正常／異常）しか把握することができない。各クラスタ上のアプリケーションがシステムの稼動状態に応じて適切な対応を取ることができるよう、各クラスタ上で全クラスタの動作状態を保持する必要がある。
- 隣接クラスタ上計算機への通信…パイプライン処理を行うためには独立して存在する隣接クラスタ上の計算機にデータを渡す必要があるが、従来のMPIは互いに独立したクラスタの計算機間で通信を行う機能を持たない。そこで、Chain of Clusters構成においても従来のMPIの枠組みを維持しつつ、異なるクラスタの計算機間メッセージ通信を実現することが課題となる。

3. MPI/SP設計

3.1. 課題の解決方法

本研究では信号処理用計算機システム上でメッセージ通信を行うためのライブラリ：MPI/SPを設計した。MPI/SPでは2.2節に示した課題である「全クラスタの動作状態把握」「クラスタ間通信」をそれぞれ以下のようにして実現している。

- Chain of Clusters構成での全クラスタの動作状態把握の実現：クラスタ鎖の端からもう片方の端に向かってクラスタの状態に関する情報を往復させることにより、全クラスタで各クラスタの動作情報を収集し、共有する。
- 隣接クラスタ間でのメッセージ受け渡しの実現：本研究では、クラスタ間でのメッセージの受け渡しは複数クラスタに属する計算機（クラスタゲートウェイ）にデータを通信することにより実現する。MPI/SPでは通信ド

メインの使い分けや、クラスタゲートウェイ取得のためのインタフェースを提供する。

以下の節で、課題を解決するために行った 2 点（MPI 環境初期化処理、インタフェース拡張）の拡張内容について述べる。

3.2. MPI/SP 設計内容

3.2.1. MPI 環境初期化処理の拡張

MPI/SP では、クラスタのマスタ（同一クラスタ上プロセスの管理を行う）はクラスタゲートウェイより選択する（図 2）。そして、以下の手順で MPI 環境の初期化を行い、全クラスタの動作状態に関する情報を収集・共有する。

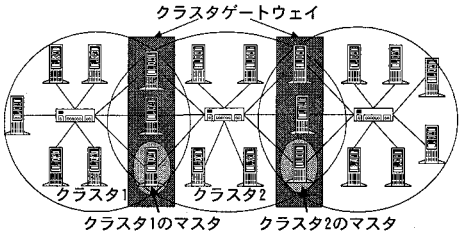

図 2：マスタ選択の例

1. **スレーブからマスタへの正常通知：**
   各スレーブは、マスタに「プロセス ID」「計算機名」「通信ポート」を示した表を送信することにより自プロセスの正常通知を行う。
2. **隣接するクラスタの正常動作を確認：**
   各マスタは隣接するクラスタが正常に動作していることを確認する。
3. **クラスタ情報のフォワード：**

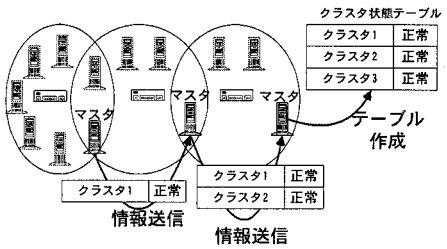

図 3：クラスタ情報のフォワード

マスタは正常通知をパイプライン処理の「出口」に近いマスタに向けて送信する。各マスタで 3.の処理を行うことにより「出口」にあるクラスタのマスタには全クラスタの正常通知が届く（図 3）。

4. **全クラスタ情報の配信：**
   パイプラインの「出口」にあるクラスタのマスタは、全クラスタの情報をまとめ、「クラスタ状態テーブル」を作成し、このテーブルを「入口」に近いクラスタのマスタに通知する。テーブルを受信したマスタは、さらに「入口」に近いマスタに向けてテーブルを配信する。これを繰り返すことにより、全マスタがテーブルを受信する。
5. **スレーブにシステム正常動作の通知：**
   4.でクラスタ状態テーブルを受信したマスタは、全スレーブに全クラスタが正常であることを通知する。この通知により MPI の初期化が終了し、プロセッサ間通信が可能となる。

3.2.2.インタフェースの拡張

クラスタ間通信を実現するために、通信ドメインの使い分け、クラスタゲートウェイの取得に関するインタフェースの拡張を行った。

1. クラスタ毎の MPI_COMM_WORLD の実現
   本研究における信号処理では各クラスタ毎に処理が独立しているため、MPI/SP ではクラスタごとにデフォルトのコミュニケータ（MPI_COMM_WORLD）を存在させている。これにより、アプリケーションで各クラスタ内での処理が簡潔に記述できる。
2. 仮想クラスタ ID の付与
   クラスタゲートウェイが通信を行う場合には通信ドメインを識別する必要があるため、各クラスタに番号を付与することとした。
3. MPI 関数の追加
   各クラスタ上に存在するコミュニケータの使い分けは、以下の 2 関数を使用することにより実現できる。
   - MPISP_get_communicator
     機能：指定されたクラスタにおけるデフォルトのコミュニケータ（MPI_COMM_WORLD）を取得
     入力：仮想クラスタ ID
     戻り値：デフォルトのコミュニケータ
   - 関数 MPISP_get_clustergw
     機能：指定した 2 つのクラスタのクラスタゲートウェイを取得
     入力：仮想クラスタ ID
     出力：クラスタゲートウェイのランク一覧

4.まとめ

本稿では、信号処理用大規模クラスタ上で MPI によるメッセージ交換を可能とするため、従来の MPI の拡張を行った。今後、この設計を実システムに適用し、評価を行う予定である。

参考文献

MPI Forum . ”MPI : A Message-Passing Interface Standard“. Jun 12, 1995.