

DNAS2 による Ninf-G 資源管理システムの構築

山崎 弘貴

Hirota YAMAZAKI

1 はじめに

近年、豊富に存在する計算資源をネットワーク接続して仮想的に統合したグリッドが注目されている。現在ではグリッド技術を用いて、最適化問題などの科学技術計算を中心に用いられる通信ライブラリやグリッドミドルウェアの開発が盛んである。

年々規模が増加する最適化問題の状況にあわせ、数百台・数千台規模の計算資源を用いて並列分散計算を行うためのグリッドミドルウェアとして GridRPC がよく用いられている。GridRPC は多数の計算資源を同時に扱うため、資源の管理が重要となってくる。GridRPC は並列計算を行うための計算モデルであるため、資源管理を行うには別の仕組みが必要となってくる。P2P システムは、対等なマシン同士によるリアルタイムな通信を行って情報収集し、複数のマシンで情報管理できるため、資源管理に向いているシステムである。

そこで本報告では、GridRPC の 1 つである Ninf-G¹⁾ に本研究室で開発している P2P 指向型グリッドミドルウェア DNAS2²⁾ を用いて、資源情報を管理するシステムの構築を行う。

2 Ninf-G

2.1 Ninf-G の概要

Ninf-G は産業技術総合研究所を中心に開発された GridRPC で、遠隔手続き呼び出し (Remote Procedure Call:RPC) によるプログラミングモデルにより遠隔地に存在する計算資源を利用することが可能である。

Ninf-G は Fig. 1 に示すように Ninf サーバと Ninf クライアントから構成される。Ninf クライアントは前もって Ninf サーバにプログラム情報を登録しておく。そして Ninf クライアントが計算データを用意して Ninf サーバに計算の実行を依頼することで、登録したプログラムの計算を行いその結果を受け取る。1 つの Ninf クライアントが複数の Ninf サーバを用いることで、互いに独立した計算を並列実行することが可能である。

2.2 Ninf-G の問題点

Ninf-G は遠隔地に存在する数百台・数千台規模の計算資源を並列実行することで、問題を解くのに時間のかかる大規模・複雑な最適化問題を取り扱うことができるため、計算モデルに適している。

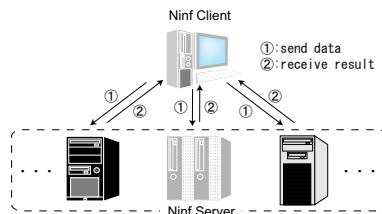


Fig. 1 Ninf-G のアーキテクチャ

しかし Ninf-G では Ninf クライアントが Ninf サーバに対して計算の実行を依頼するには、予め IP アドレスなどの資源情報を知っておく必要がある。現在では Ninf クライアントが Ninf サーバの資源情報を知る方法で通信を行っているが、他の方法で資源情報を知るには、資源情報を管理する仕組みが必要である。資源情報を管理する方法として P2P の特徴であるリアルタイムな情報収集があり、P2P 指向型グリッドミドルウェアの DNAS2 によって資源情報を管理することが可能であると考えられる。次章に DNAS2 についての概要および基本設計について述べる。

3 DNAS2

3.1 DNAS2 の概要

DNAS2 は、グリッドといった広域ネットワークにおいてトポロジを動的に変更できる P2P 指向のソリー型グリッドミドルウェアである。DNAS2 は DNAS を基に設計されており、DNAS で問題とされていたセキュリティや通信機構などを改善している。

DNAS2 はソケット通信により、各ノードで動作する DNAS デーモン同士で一定時間毎に情報通信を行っている。DNAS2 はクライアント・サーバといった構造ではなく、対等なマシン同士による通信のため、システム全体としては Fig. 2 によるアーキテクチャとなる。

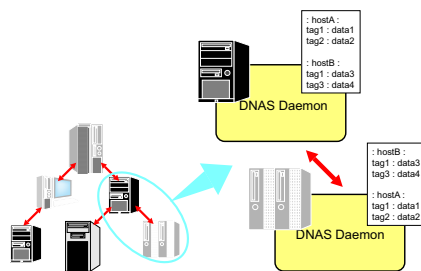


Fig. 2 DNAS2 のアーキテクチャ

3.2 DNAS2の基本設計

DNAS2は後述する3つの特徴を基に設計されており、負荷や耐故障性を考慮したシステムとなっている。

3.2.1 P2P 指向型システム

グリッド環境におけるノードの追加および削減に対応するため、任意のタイミングでのノードの参加または不参加を可能とするP2P指向型の仕組みを提供している。このためシステムを停止させることなく、残りのノードで動作を続行することができる。

3.2.2 ツリートポロジ

DNAS2はFig. 2に示すようなツリートポロジを構成する。隣接する一部のノードとのみ直接通信するため、ある一部のノードへの多大な負荷を防いでいる。

3.2.3 トポロジの動的再構成

グリッド環境におけるノードの追加および削減、またはネットワークの不通による障害が生じた場合、トポロジを動的再構成する機能を有しており、以下に示す3通りにおいて、Fig. 3に示すような動的再構成が行われる。

- 上位ノードへの接続が不可である場合
- 下位ノード数の接続が指定値より超える場合
- ノードのロードアベレージが閾値を超える場合

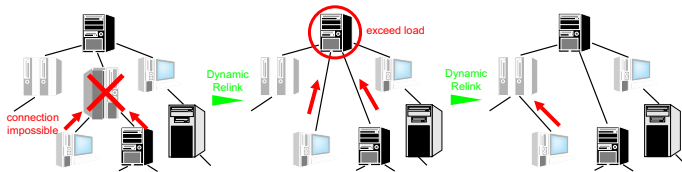


Fig. 3 トポロジの動的再構成の様子

4 DNAS2によるNinf-G 資源管理システム

Ninf-Gを用いて計算を行うときの資源情報をDNAS2によって管理するシステムの構築を考える。Ninf-GではNinfクライアントがNinfサーバに計算の実行を依頼する際、予めIPアドレスなどの資源情報を知っておく必要があり、現在ではNinfクライアントがNinfサーバの資源情報を知る方法で通信を行っている。

Ninf-Gには独自に資源管理を行う仕組みが存在しないため、DNAS2を用いて資源情報を扱う枠組みを定義すれば、自動的に資源情報を知ることが可能になると考えられる。

そこで構築するシステムではNinfクライアントが資源情報を知る必要なく通信が行えるように、NinfサーバからNinfクライアントに対して通信を行う方法を取る。またDNAS2をNinfサーバ上で動作させることで、Ninfクライアントの資源情報を管理することができる。

今回構築するシステムの構造をFig. 4に示す。

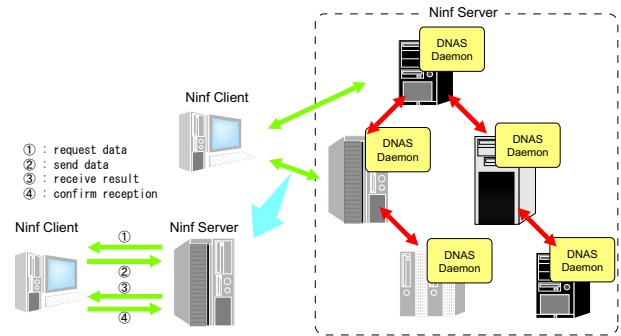


Fig. 4 DNAS2によるNinf-G 資源管理システム

構築するシステムでのNinfサーバとNinfクライアントの通信は以下の手順で行われる。

1. NinfServerがNinfClientに対して情報を要求
2. NinfClientはNinfServerに対して情報を送信
3. NinfServerは計算モジュールで計算を行い、結果をNinfClientに返信
4. NinfClientは受信したことをNinfServerに通知

この通信手順によって、NinfクライアントはNinfサーバへの計算依頼を自主的に行う必要はなく、NinfサーバがNinfクライアントから計算に必要な情報を自動的に取り、計算処理を行ってNinfクライアントに返すことが可能であると考えられる。つまりNinfクライアントは計算に用いるデータを用意するだけで、通信はNinfサーバが中心に行う。

5 まとめ

本報告では、GridRPCシステムNinf-Gを用いて計算するときの資源情報を本研究室で開発しているDNAS2を用いて管理するシステムの構築について述べた。Ninf-Gには独自に資源管理を行う仕組みが存在しないので、DNAS2を用いて資源情報を扱う仕組みを提供すれば、自動的に資源情報を知ることが可能になると考えられる。構築するシステムでは、NinfクライアントがNinfサーバの資源情報を取得する必要なしに通信を行うことができ、Ninfクライアントの情報はNinfサーバで動作するDNAS2で管理する。

今後の課題としては、Ninf-Gを抜いてNinf-Gの構造を明らかにした後、構築するシステムの実装方法について考える必要がある。

参考文献

- 1) Ninf Project Home Page
<http://ninf.apgrid.org/>
- 2) 谷口 義樹, 廣安 知之, 三木 光範: Javaによる階層型グリッドミドルウェアDNAS2の設計と実装, 同志社大学大学院修士論文, 2005