5X-4

無線接続型 Android クラスタシステムの自動構築制御手法

荒井 裕介 † 大津 金光 † 横田 隆史 † 大川 猛 †
† 宇都宮大学大学院工学研究科情報システム科学専攻

1 はじめに

近年，マルチコアを搭載した高性能な Android OS 搭載の携帯端末 (Android 端末) が普及している．Android 端末は Bluetooth や Wi-Fi といった無線通信機能を利用することで，端末間の無線相互通信が可能となっている．この特徴に着目し，我々はいつでもどこでも即時構築可能なクラスタ計算機システムとして，Android 端末を用いた無線接続型クラスタシステムの開発を行っている [1]．しかし，これまでは新たな端末がクラスタに参入する際，自動で検知することができず，新規参入の端末毎に IP アドレスや使用可能 CPU 数といった，システム内の端末情報を手動で更新する必要があった．

そこで今回，無線接続型クラスタシステムを自動で構築し，また，端末の新規参入があった際にそれを検知し，システムに参加している端末情報の自動更新を行う機能を実装した．本稿ではそれについて述べる．

2 無線接続型 Android クラスタシステム

我々が開発を行っている無線接続型 Android クラスタシステムの概要を述べる．まず，無線通信手段について述べる．Android 端末には，Wi-Fi，Bluetooth，3G/4G の 3 種類の通信手段がある．複数の端末を用いてクラスタ計算機を構築する場合，端末間の通信性能がクラスタとしての実行性能に大きな影響を与える．そのため，できるだけ高速な通信手段を用いることが望ましい．本システムでは，手軽に利用でき，かつ通信速度が高速な Wi-Fi をシステムの無線通信手段として使用する．

本システムは，複数の Android 端末上での並列分散処理を想定する．そのため，並列分散処理で最も利用される Message Passing Interface (MPI) の実装の一つである，Open MPI[2] を並列分散処理の基礎として用いる．また，リモートプロセスの起動には OpenSSH を用いる．

以上の Open MPI と OpenSSH，およびクラスタ上で実行する MPI 並列プログラムは，Android Native Development Kit (NDK) を用いてビルドする．通常，Android 上で動作するアプリケーションは Dalvik VM と呼ばれる仮想マシン上で動作する為，動作が遅くなる．しかし，NDK を用いて開発を行えば CPU が直接実行する機械命令プログラムとなるため，高速な動作が可能になる．

Automatic Construction Control Method of Wireless Connected Android Cluster System
†Yusuke Arai, Kanemitsu Ootsu, Takashi Yokota and Takeshi Ohkawa
Department of Information Systems Science, Graduate School of Engineering, Utsunomiya University (†)

3 自動構築制御手法

新たな端末がクラスタシステムに参入する場合，参入する端末は今までシステムに参加していた端末に自分の参入を通知し，システム内の端末は新規参入端末に自分たちの情報を伝える必要がある．我々はこの一連の処理を，ブロードキャスト通信を行うことを前提とした，独自のプロトコルにより実現した．ここで，端末の新規参入の自動検出，さらに構築したクラスタシステムで自動的に MPI 並列処理を行う制御について述べる．

クラスタシステム自動構築の制御は，端末の新規参入を検知し IP アドレスと並列分散処理に提供可能な CPU 数 (以下これらをまとめてホスト情報) を記録するサーバ側と，クラスタシステム内の各端末に新規参入を通知するクライアント側に分けられる．以下にそれぞれの制御方法について述べる．また，図 1 に制御全体の流れを示す．

クライアント側制御

クライアント側の制御では，新規参入端末は図 1 のように，ブロードキャストアドレスによりローカルネットワーク内にメッセージを送信する．通知が完了したら，自分自身のホスト情報を記録した情報ファイル (以下ホストファイル) を生成し，他の端末からの受信待ち状態に移行する．この時，自身の IP アドレスは「localhost」と記録する．ホストファイルはシステムに参加している端末の管理，および MPI 並列処理において，各端末に割り当てる並列プロセス数を決定する際に用いられる．この状態で一定時間 (現在は 5 秒)，いずれかの端末から応答が無かった場合，クラスタが存在していないと判断し，メッセージを送信した端末がクラスタの最初の端末となる．その後，サーバ側の制御へと移行する．クラスタが既に存在する場合は，クラスタ内のすべての端末から送信される応答メッセージを受信し，端末の情報をホストファイルに記録する．最後に応答を受け取ってから一定時間 (5 秒) 他の端末から応答が無かった場合，クライアント側制御を終了しサーバ側の制御へと移行する．

サーバ側制御

サーバ側の制御では，図 1 のように新規参入端末から新規参入要求のッセージを受け取った場合，クラスタシステム内のサーバ側制御を行っているすべての端末は，各々が持っているホストファイルに新規参入の通知があった端末のホスト情報を記録する．その後，参入を要求してきた端末へと

参入を許可した旨のメッセージを返信し，再度待ち受け状態に戻る．この状態で一定時間端末の新規参入が検知できなかった場合，それ以上の端末の参入は無いと判断し，通信待ち状態を解除する．待ち受け状態解除時にクラスタに参加している端末で MPI 並列処理を行う．

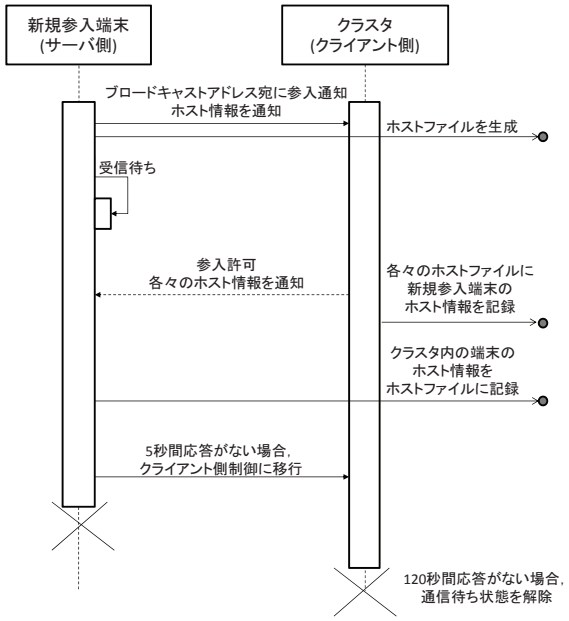

図 1: 制御全体のシーケンス

クラスタシステムの構築後，システム内の端末間で並列処理を行う．

まず，処理を行いたいプログラムの実行可能バイナリファイルを，scp コマンドでホストファイルに記録されている各端末へと転送する．転送はファイルを持っている端末が行う．処理を行う MPI プログラムは，事前にビルドしたものをシステム内のいずれかの端末に持たせておく．

システム内の各端末へと実行可能バイナリファイルの転送が完了したら，MPI 並列処理を開始する．

4 機能検証

今回実装したシステムで，正常にクラスタを構築できるか，また構築したクラスタシステムで正しく並列処理を行うことができているかの検証を N クイーン問題 (クイーン数: 16) プログラムを使用して行った．プログラムを Android 端末 1 台から 3 台で実行した．検証に使用した環境を表 1 に示す．各端末への並列プロセスの最大割り当て数は 1 台につき 4 プロセスとした．

クラスタに端末が参入するまでの様子を図 2 に示す．図 2 の (a) がクラスタのサーバ側，(b) が新規参入しようとしている端末の様子となる．(a) の四角で囲った部分が参入端末からの参入通知メッセージ，(b) の四角で囲った部分がクラスタ内端末からのメッセージとなる．それぞれの画像を見ると，クラスタ内の端末

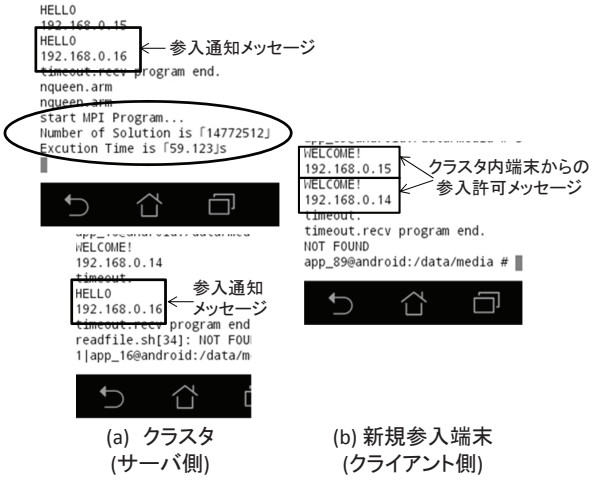

図 2: 端末がクラスタに参入する様子

表 1: 評価環境

使用端末	ASUS 社 TF201
CPU	NVIDIA Tegra3 モバイルプロセッサ (4 コア)
動作周波数	1.4GHz(クアッドコア稼働時 1.3GHz)
メモリ	1GByte
Wi-Fi	規格: IEEE 802.11 b/g/n (ストリーム数: 1) 帯域: 18Mbps (Iperf による測定)
ルータ	NEC PAWG600HP

は新規参入端末からのメッセージを，新規参入端末はクラスタ内の各端末からのメッセージを正しく受信していることがわかる．また，(a) の丸で囲った部分は並列処理の結果を出力している部分である．正しい結果を出力していることが確認できたので，実装したシステムが図 1 のシーケンス通りに動作し，また，クラスタとして正しく動作していることも確認できた．

5 おわりに

本稿では，我々が開発している無線接続型クラスタシステムにおいて，クラスタシステムの自動構築のための制御方法について述べた．今後の課題として，現状のシステムでは端末の途中脱退には対応できないので，こちらにも対応できるような制御を実装する必要がある．

謝辞

本研究は，一部日本学術振興会科学研究費補助金 (基盤研究 (C)24500055，同 (C)24500054，同 (C)25330055，若手研究 (B)25730026) の援助による．

参考文献

[1] 荒井 裕介ほか: “Android 端末を使用したクラスタ計算機システムの構築”, 情報処理学会第 75 回全国大会講演論文集, pp.1-219〜1-220, 2013.

[2] Edgar Gabriel et al.: “Open MPI: Goals, Concept, and Design of a Next Generation MPI Implementation”，Proceedings, 11th European PVM/MPI Users' Group Meeting，2004.