1J-06

Android OSにおけるMPI並列処理アプリケーション実行環境の検討

新里 将大 † 大津 金光 †† 大川 猛 †† 横田 隆史 ††
† 宇都宮大学工学部情報工学科 †† 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

近年のモバイル端末は高性能化が進み，クアッドコアやオクタコアのプロセッサが標準的に搭載されている．さらに，メモリやストレージも高性能化・大容量化してきており，大規模なプログラムをモバイル端末上で実行することが可能となってきている．また，モバイル端末は広く普及しているため，モバイル端末の処理性能を並列処理の計算資源として利用することが期待されている．我々は，複数台のAndroid OS搭載端末を使用してMPIによる並列処理を行うクラスタ計算機システムを開発している[1]．本システムは，MPI並列処理の実行環境をAndroid端末のシステム領域に配置している．Android端末のシステム領域は一般ユーザの権限では操作することができないため，本システムで使用するAndroid端末はスーパーユーザ化を行っている．しかし，Android端末のスーパーユーザ化作業は端末が起動不可能になるなどの危険があり，基本的にメーカの保証対象外の操作である．そこで，本稿ではAndroid端末のスーパーユーザ化を行わずにMPI並列処理を行うための実行環境について検討する．

2 Android クラスタシステム

本システムでは，Android OSを搭載した端末を使用してMPI並列処理を行う．ノード間通信には，Wi-FiやEthenetなどのAndroid端末に標準搭載されているネットワーク通信機能を使用する．

MPI並列処理はOpen MPIライブラリを使用して実現する．Open MPIライブラリはC言語で記述されており，Android端末上で実行するためにはAndroid用ネイティブ開発環境であるAndroid Native Development Kit(Android NDK)を用いてビルドする必要がある．Open MPIライブラリはAndroid OSをサポートしていないため，本システムでは，Android NDKを用いてOpen MPIライブラリをビルドする環境の構築を行っている．また，MPIアプリケーションはCまたはC++言語で記述したものを使用する．MPIアプリケーションについても，Android NDKを用いてビルドするためにビルド環境の構築を行っている．

本システムでは，MPI並列処理を起動するためのAndroidアプリケーションを実装していない．Androidアプリケーションを使用せずにAndroid端末上でMPI並列処理を実行する場合，Open MPIライブラリおよびMPIアプリケーションの実行ファイルをAndroid端末内に格納し，Androidのコマンドラインツールから直接起動するしか方法はない．Android端末内において，実行権限を付与した実行ファイルを配置することができるストレージ領域はシステム領域のみであり，システム領域へのアクセスにはスーパーユーザ権限が必要である．そのため，従来手法では，Android端末のスーパーユーザ化を行い，システム領域にOpen MPIライブラリおよびMPIアプリケーションを配置し，Androidのコマンドラインツールから MPIアプリケーションを起動する．

3 Android OSにおけるMPIアプリケーションの実行環境

Android端末のスーパーユーザ化を行わずにノード内でMPI並列処理を行うための実行環境について検討する．

Android OSカーネルはカスタマイズしたLinuxカーネルを使用しており，Linuxの機能の多くがAndroid OSにおいても使用されている．特に，Android端末内のファイルにはLinuxと同様にユーザID，ファイルパーミッションなどが付与されている．また，ライブラリや実行ファイルは実行権限が付与された状態でストレージに格納されていなければ使用することができない．

Androidは外部ストレージと内部ストレージという2つのストレージ領域を備えている．

外部ストレージ領域はどのようなアプリケーションからでもアクセス可能な共有のストレージである．外部ストレージはFATファイルシステムを使用しており，FATはLinuxのファイルパーミッションをサポートしていないため，外部ストレージに保存されたファイルには実行権限を付与できない．そのため，Open MPIライブラリおよびMPIアプリケーションは内部ストレージ領域に格納しなければならない．

内部ストレージ領域はAndroidのシステムライブラリなどを保存するために使用するストレージである．通常，内部ストレージへのアクセスにはスーパーユーザ権限が必要である．ただし，Androidアプリケーションには，各Androidアプリケーションに固有の内部ストレージ領域が与えられている．このストレージ領域はアプリケーションのデータ保存用領域として使用することができる．また，Androidアプリケーションが自身の固有の内部ストレージ領域にアクセスする場合にはスーパーユーザ権限が不要である．

提案手法では，外部ストレージに保存されたOpen

Consideration on runtime environment of MPI parallel processing application in Android OS
†Masahiro Nissato, ††Kanemitsu Ootsu, ††Takeshi Ohkawa and ††Takashi Yokota
Department of Information Science, Faculty of Engineering, Utsunomiya University (†)
Department of Information Systems Science, Graduate School of Engineering, Utsunomiya University (††)

MPI ライブラリおよび MPI アプリケーションを，自身の固有の内部ストレージ領域に格納する Android アプリケーションを作成する．Android アプリケーションが自身の固有の内部ストレージ領域にアクセスする場合にはスーパーユーザ権限を必要としないため，Android 端末をスーパーユーザ化せずに内部ストレージ領域に Open MPI ライブラリおよび MPI アプリケーションを格納できる．ただし，Android アプリケーション固有の内部ストレージ領域に格納した MPI アプリケーションにアクセスできるのは，MPI アプリケーションの格納を行った Android アプリケーションのみである．従って，MPI アプリケーションの起動に関しても同一の Android アプリケーション上から実行する必要がある．

本手法では，MPI アプリケーションを Android アプリケーション上から起動するために Java API Framework の ProcessBuilder クラスを利用する．ProcessBuilder クラスを用いることにより，Android アプリケーション上から Android OS のコマンドや，MPI アプリケーションを起動することができる．

以上の検討に基づいて実装した，MPI アプリケーションの実行環境では，MPI 並列処理を起動する Android アプリケーションから mpirun を実行するプロセスが fork し，mpirun から MPI 並列処理プロセスが fork されることにより MPI 並列処理を実行する．この方法では，Android アプリケーションとは別のプロセスとして MPI 並列処理を実行するため，MPI 並列処理中に Android アプリケーションのプロセスが Andoid 端末上に常駐する．これは，ProcessBuilder クラスを用いて実行したコマンドは別プロセスとして実行されるためである．

4　動作テスト

MPI 並列処理を起動する Android アプリケーションのテスト実装を行い，Android アプリケーションを介して MPI 並列処理を起動した場合について動作テストを行う．動作テストでは，テスト実装した Android アプリケーションから MPI 版の N クイーンプログラムを起動可能であるかテストする．動作テストに使用した Android 端末は Nexus7(2013) タブレットである．Nexus7(2013) の構成は，動作周波数：1.5GHz，コア数：4，メモリ：2GB，OS：Android 5.1.1 であり，動作テストでは 1 ノード内で 4 プロセスの MPI 並列処理を行う．

実装した Android アプリケーションは，MPI 並列処理の結果を端末画面に表示する．図 1 は，端末に表示された MPI 並列処理結果のスクリーンショットである．図 1 から，クイーン数 17 のときの N クイーンプログラムの出力結果を得られていることが分かる．

MPI 並列処理実行時の Android 端末のプロセス状況を図 2 に示す．図 2 は，MPI 並列処理実行時に top コマンドを使用した際のスクリーンショットである．赤枠で囲われた部分が MPI 並列処理プロセスである．

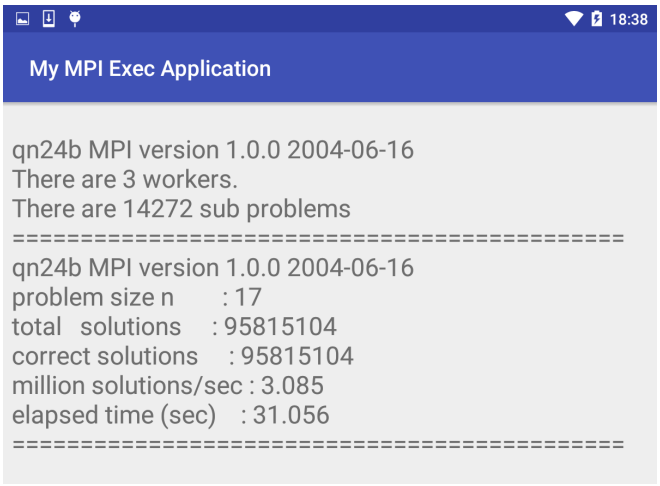

図 1: 動作テスト時のスクリーンショット

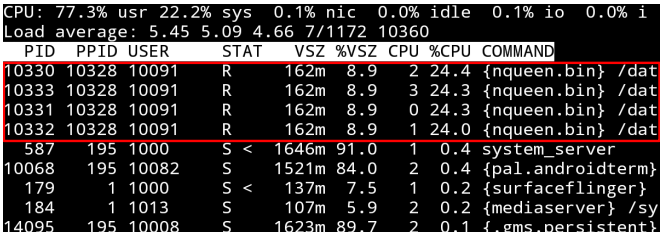

図 2: 動作テスト時のプロセス状況

nqueen.bin は MPI 並列処理の実行ファイルであり，4 つの MPI 並列処理プロセスが起動していることが分かる．

以上から，MPI 並列処理の実行結果を得られており，MPI 並列処理プロセスが複数立ち上がっていることから，Android アプリケーション上から MPI 並列処理が起動可能であることを確認できた．

5　おわりに

本稿では，Android 端末のスーパーユーザ権限を必要としない形で MPI による並列処理を実行する方法について検討し，テスト実装を行った．今後の課題は Android アプリケーションから起動した MPI アプリケーションによるノード間の MPI 並列処理を実現することである．

謝辞

本研究は一部 JSPS 科研費 15K00068，16K00068，17K00072 の助成による．

参考文献

[1] Yuki Sawada, Yusuke Arai, Kanemitsu Ootsu, Takashi Yokota, Takeshi Ohkawa, “Performance of Android Cluster System Allowing Dynamic Node Reconfiguration”, Wireless Personal Communication, Vol.93, Issue 4, pp.1067-1087, Feb. 2017.