Vol.2015-GN-93 No.21
Vol.2015-CDS-12 No.21
Vol.2015-DCC-9 No.21
2015/1/27

複数の Android タブレットを用いた並列 I/O 処理に関する一考察

濱中真太郎[†1] 中村優太[†2] 野村駿[†2] 坂本寛和[†2] 山口実靖[†1]

近年，Android OS を搭載したスマートフォンやタブレット PC が普及し，これらの端末の低価格化が進んでいる．Android タブレットは二次記憶装置としてフラッシュメモリを搭載している．フラッシュメモリはハードディスクと比べランダムアクセス性能が高い．Android タブレットをクラスタリングすることより高スループットのランダムアクセス処理システムの構築が可能であると期待できる．本稿では，単一の Android タブレットの I/O 性能の評価や Wi-Fi 接続の Android タブレットクラスタの並列 I/O 性能の評価を行い，タブレットクラスタにおける並列ランダムアクセス処理のボトルネックに関する考察を行う．

SHINTARO HAMANAKA[†1] YUTA NAKAMURA[†2] SHUN NOMURA[†2]
HIROKAZU SAKAMOTO[†2] SANEYASU YAMAGUCHI[†1]

1. はじめに

タブレット PC が普及し，安価でこれらの端末を入手できる環境が整いつつある．タブレット PC にはフラッシュメモリが搭載されており，フラッシュメモリはハードディスクと比較して，ランダムアクセス性能が高い．よって，フラッシュメモリが搭載されたタブレット PC を複数台用いてクラスタリングを行うことで，極めて高いスループットの I/O 処理環境を構築することが可能であると考えられる．また，Android OS はサーバ PC や PC クラスタで広く使用されている Linux OS を基に作成されているため，Android 端末内にサーバプロセスを起動し端末間で連携を行うクラスタ環境を構築することが容易にできると期待できる．

本稿ではまず，単一 Android のタブレットの I/O 性能の評価を行い，Android タブレットがハードディスク搭載 PC よりも高いランダムアクセス性能を有していることを示す．次に，Android タブレットのクラスタの構築方法を示し，性能評価により Android タブレットクラスタが非常に高いランダム I/O スループットを実現できることを示す．最後に，Android タブレットクラスタにおける並列ランダムアクセス処理のボトルネックに関する考察を行う．

2. 単一端末の I/O 性能

本章で単一の Android タブレット端末のランダム I/O 性能の評価を行い，HDD を搭載した PC と性能の比較を行う．

2.1 測定方法

ランダムアクセス要求を連続して発行するプログラムを C 言語で作成した．これを arm プロセッサ用にクロスコンパイルし，実験端末(Nexus7 2013)上で実行してその処理時間を測定することにより Android タブレットのランダム I/O 性能を評価した．また，同一プログラムをハードディスク搭載の PC 上で実行し，そのランダム I/O 性能の評価を行った．

本プログラムは，10GB のファイルの中からランダム(一様分布)に選択した位置(ファイル内のオフセット)に対して `read()` システムコールを発行することを繰り返す．10GB のファイルはタブレット端末，PC のファイルシステム上に作成され，物理的にはタブレット端末のフラッシュメモリ，PC のハードディスク内に保存される．システムコールのブロックサイズは，1B から 16KB とした．並列実行スレッド数は 1 から 32 とした．I/O 要求は読み込み要求のみ発行した．

2.2 実験環境

本実験で使用したタブレット端末と PC の仕様は表 1，表 2 の通りである．

表 1 使用タブレット

Device name	Nexus 7 (2013)
OS	Android4.4.2
CPU	Qualcomm Snapdragon S4 Pro,1.5 GHz
Memory	2GB
Wi-Fi	802.11a/b/g/n Dual Band
Storage	SK Hynix H26M51003EQR 16 GB eMMC NAND Flash
FileSystem	ext4
I/O Scheduler	CFQ

表 2 使用 PC

OS	Ubuntu 12.04.2 LTS
CPU	Intel Core i7-2700K CPU 3.50GHz
Memory	16GB
Storage	HDD ST3000DM001-1CH1 7200rpm
File System	ext4
I/O Scheduler	deadline

Vol.2015-GN-93 No.21
Vol.2015-CDS-12 No.21
Vol.2015-DCC-9 No.21
2015/1/27

2.3 測定結果

測定結果を図 1 から図 4 に示す．両図の横軸は並列実行スレッド数であり，両図の縦軸は得られた I/O スループットである．単位は[Transaction/sec]および[KByte/sec]であり，システムコール `read()` 1 回を 1 トランザクションとしている．各線は，システムコール `read()` のブロックサイズを表している．

図より，フラッシュメモリを搭載したタブレットのランダム I/O 性能は，ハードディスク搭載の PC よりも高いことが確認できる．また，タブレットにおいてはスレッド数を一定数(4 スレッド)まで増加させると総 I/O 性能は向上し，それ以上はスレッド数を増加させても総 I/O 性能の増加は無いことが分かる．

ブロックサイズに着目すると，ファイルシステムのブロックサイズが 4KB であるためシステムコールブロックサイズを 4KB 未満にしてもトランザクションスループットの向上が無いことが分かる．すなわち，ユーザー空間プロセスが 4KB 未満の I/O 要求を発行してもファイルシステム実装がブロックデバイスに対して 4KB の要求を発行している．

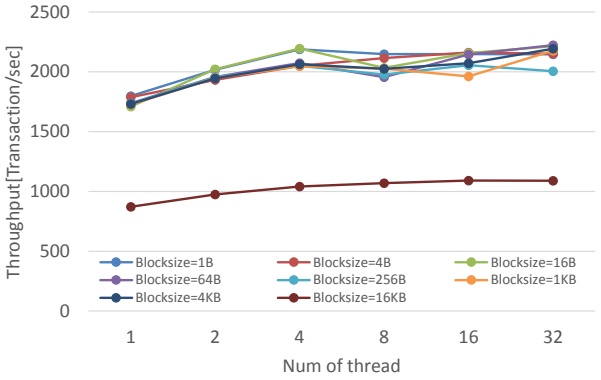

図 1　I/O 性能評価結果[Transaction/sec] (タブレット)

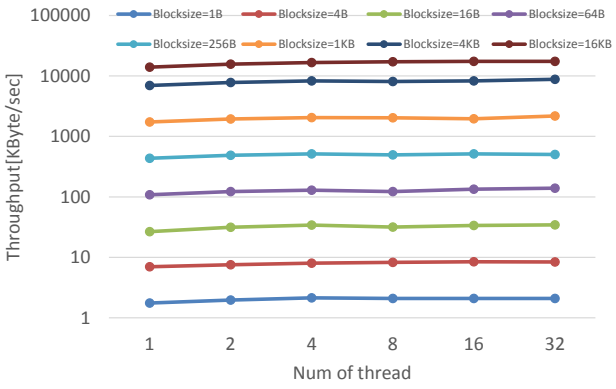

図 2　I/O 性能評価結果[KByte/sec] (タブレット)

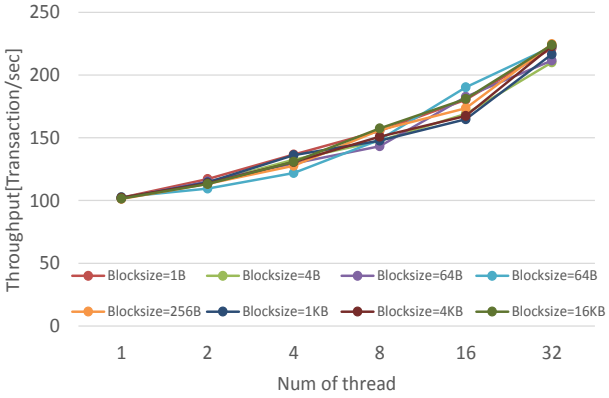

図 3　I/O 性能評価結果[Transaction/sec] (PC)

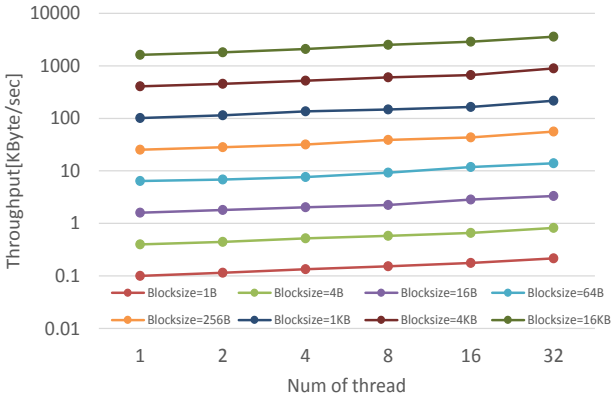

図 4　I/O 性能評価結果[KByte/sec] (PC)

3. タブレットクラスタの I/O 性能

本章にて，Android タブレットクラスタの構築手法と実装手法，そして同クラスタのランダム I/O 性能評価について述べる．

3.1 システム概要

本クラスタは 1 台のマスタ端末と，*n* 台のスレイブ端末で構成される．各スレイブ端末は *s* バイトのファイルを保持し，クラスタ全体で *n***s* バイトのデータ保存空間を有している．図 5 の様にクラスタ全体で仮想的に *n***s* バイトの 1 つのストレージ空間(図 5 内の”virtual strage”)をクライアントに提供する．I/O 要求はマスタ端末内のマスタプロセスに与えられ，応答はマスタプロセスから返される．I/O 要求がマスタプロセスに与えられると，I/O 要求は担当のスレイブ端末に転送され，受信したスレイブ端末中のスレイブプロセスがその I/O 要求をローカルファイルを用いて処理する．そして，スレイブプロセスが I/O 要求に対する応答をマスタプロセスに送り返す．実験に使用した無線 LAN ルータの仕様は表 3 の通りである．

Vol.2015-GN-93 No.21
Vol.2015-CDS-12 No.21
Vol.2015-DCC-9 No.21
2015/1/27

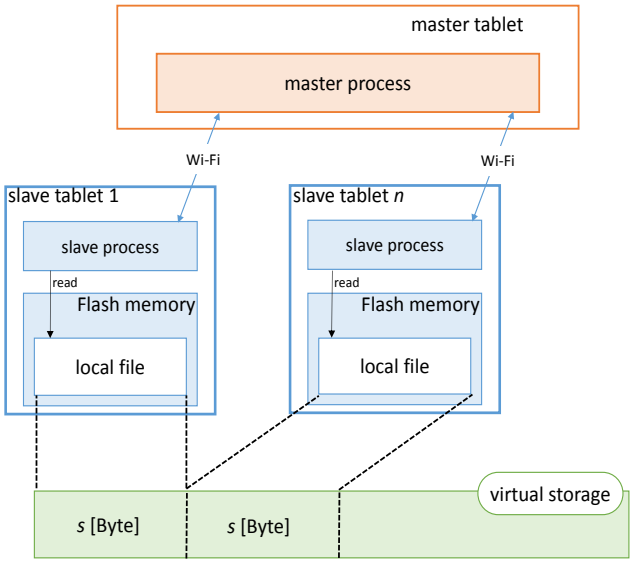

図 5 システム概要

表 3 使用無線 LAN ルータ

品番	PA-WR9500N-HP
規格	IIEEE802.11n/a/g/b

3.2 実装

前節で述べたように本クラスタは 1 台のマスタ端末と，*n* 台のスレイブ端末で構成される．図 4 に実装の詳細を示す．本稿の構築例では，*n*=8[台]，*s*=10[GB]であり，クラスタは合計 9 台のタブレットで構成され，仮想的に 80GB のストレージ空間が存在する．

マスタ端末は要求の発行，要求の振り分け，要求の転送，応答の受信を行う．スレイブ端末は 10GB のファイルを有しており，マスタ端末から受け取った要求アドレスに対して指定サイズの I/O 読み込みを行いマスタ端末へ応答送り返す．

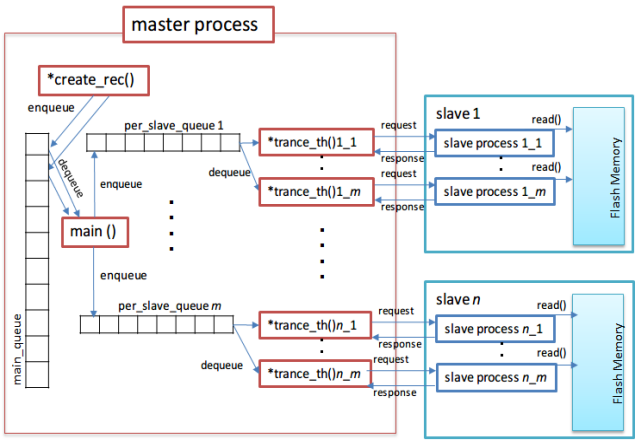

図 6 タブレットクラスタのシステム構成

- マスタプロセス

  マスタプロセスはマスタ端末上で動作し，1 本の `main_queue` と *n*(=8)本の `per_slave_queue`，1 個の main スレッドと *n***m* 個の転送スレッドを有している．すなわち，`per_slavequeue`1 本あたり *m* 個の転送用スレッドが存在する．`main_queue` にはクライアントプログラムが発行した virtual storage への I/O 要求が格納される．通常，I/O 要求発行は別プロセスにて行われるが，実験用の本実装ではマスタプロセス内で発行を行った．`per_slave_queue` には，スレイブ端末ごとの I/O 要求が格納され，`main_queue` から `per_slave_queue` への移動は main スレッドにより行われる．

  転送用スレッドは `per_slave_queue` から I/O 要求を取り出し(デキューし)，それをスレイブ端末のスレイブプロセスに転送する．転送用スレッドは後述する各スレイブ端末のスレイブプロセスと TCP コネクションを確立する．転送は TCP コネクションを用いて行われる．`main_queue` に対して発行される I/O 要求には virtual storage 全体におけるアドレス(本実装の例では 5 バイト長のアドレスであり，0~80GB の範囲を取る)が格納されている．例えば，要求アドレスが 20GB 以上 30GB 未満のアドレスである場合は，当該 I/O 要求は 3 本目の `per_slave_queue` に移動され，スレイブ端末 3 に転送される．

- スレイブプロセス

  スレイブプロセスはマスタ端末のマスタプロセスから I/O 要求を受け取り，送られてきた要求アドレスに対して I/O システムコールを行う．そして，応答をマスタプロセスに送信する．スレイブプロセスは，マスタ端末の転送用スレッドと同数だけ起動され，転送用スレッド 1 個とスレイブプロセス 1 個で TCP コネクションが 1 個確立される．

3.3 測定結果

`read()` システムコールのブロックサイズを 1 バイト，スレイブ端末 1 台ごとの転送用スレッドの数を 1~32 個として並列実行したときの I/O 性能測定結果を図 7 に示す．図の横軸はスレイブ端末の台数で，縦軸は得られたスループットである．各線は，スレイブ端末 1 台あたりの転送用スレッドの数である．図より，並列実行スレッド数を増加させることによりスループットが向上することが分かる．スレイブ端末の台数に着目すると，スレイブ端末数を増やすに伴い I/O スループットが向上すること，2 台目以降はスレイブ端末を追加することによる I/O 性能の向上が少ないことが分かる．スレイブ端末 1 台 32 スレッド実行時のスループットに着目すると，前章で得たフラッシュストレージの限界性能(2100[transaction/sec])に近い性能が達成できていることが分かる．しかし，スレイブ端末が 2 台以上

Vol.2015-GN-93 No.21
Vol.2015-CDS-12 No.21
Vol.2015-DCC-9 No.21
2015/1/27

のスループットに着目すると，スレイブ端末 1 台あたりの I/O 性能は上記性能より低いことが分かる．このことより、マスタプロセスからスレイブプロセスに送られる I/O 要求の量が十分でなく，フラッシュメモリの性能を飽和させることができていないと予想できる．また，スレイブ端末ごとのスレッド数を増加させても十分な性能が得られないことから．タブレット間の Wi-Fi 通信におけるパケット数が処理のボトルネックとなっていることも原因の 1 つとして考えることができる．

そこで，1 回(1 パケット)あたりの転送 I/O 要求数を 10 個，100 個と増加させて I/O スループットを測定した(パケット数の Wi-Fi 通信性能への影響については次節で考察する)．

複数の I/O 要求を 1 回(1 パケット)に集約して転送する手法の実装は以下の通りである．マスタプロセス内の転送用スレッドは，最大集約数(10 または 100)分の I/O 要求の `per_slave_queue` からのデキューを試みる．そして，デキューできた分を，1 回のソケット `write` システムコールを用いて転送する．

スレイブプロセスは，ソケットより読み取ることが可能である全てのデータを読み込み，読み込みが行うことができた全ての I/O 要求を処理した後，全ての応答を 1 回のソケット `write` システムコールによりマスタ端末に転送する．Nagle のアルゴリズムは有効にしていないため，Nagle アルゴリズムに起因する Ack 待ちで Wi-Fi 通信の RTT だけ TCP 通信が遅延されることはない．TCP 層においてフラグメントが生じるため，マスタプロセスが一度に送信した I/O 要求の数とスレイブプロセスが一度に読み込む I/O 要求数が必ず一致するとは限らない．

最大集約数 10 の結果を図 8 に，最大集約数 100 の結果を図 9 に示す．スレイブ端末数と合計スループットに着目すると，両図ともスレイブ端末数に比例に近い合計 I/O スループットが実現されていることが分かる．このことから I/O 要求および応答を集約して転送することにより Wi-Fi 通信を用いるタブレットクラスタにおいて非常に高いランダム I/O スループットを実現できることが分かる．得られた I/O スループットは最大で約 16,000[Transaction/sec]であり，ハードディスクと比較すると大幅に高いスループットを実現できていることが分かる．スレッド数と合計スループットの関係に着目すると，スレッド数 8(図 8)や，4(図 9)で飽和を迎えている事がわかる．

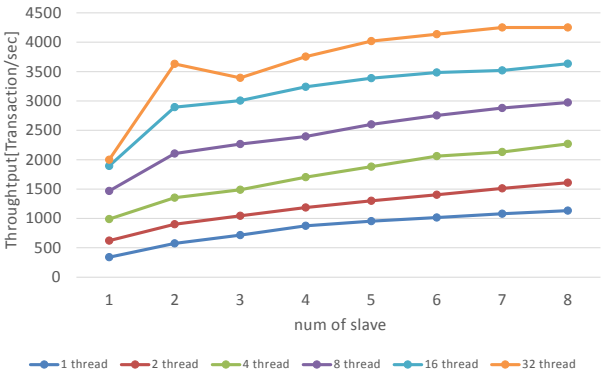

図 7　I/O 性能評価結果(タブレットクラスタ)

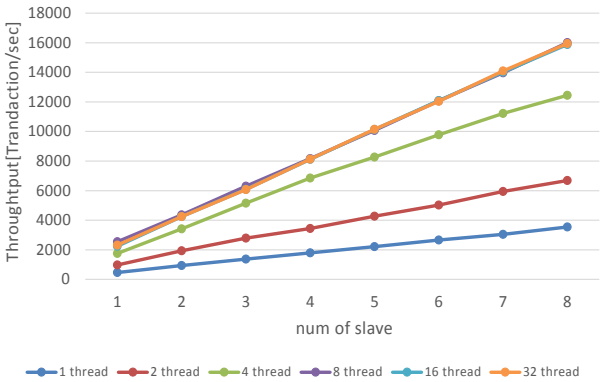

図 8　I/O 性能評価結果
(タブレットクラスタの最大集約数=10)

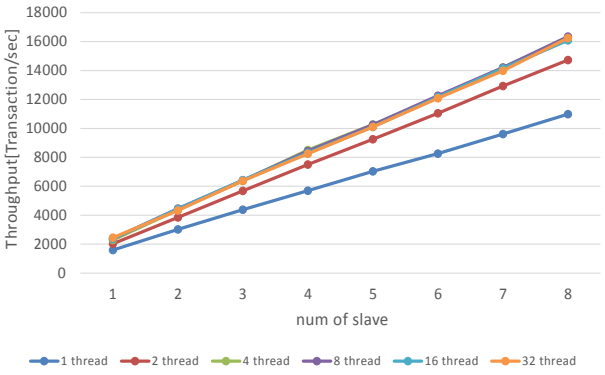

図 9　I/O 性能評価結果
(タブレットクラスタの最大集約数=100)

3.4　測定結果(I/O 無し)

本節で，前節の実験を I/O 処理を行わずに実行し，Wi-Fi 通信の性能の評価およびボトルネックに関する考察を行う．実験は同クラスタのスレイブプロセスの `fseeko()`，`fread()` システムコールをコメントアウトして実行を行った．Wi-Fi 通信は前節と同量だけ行われ，I/O 要求への応答としてスレイブプロセスのソケット `write` 用に用意されたバッファに偶然格納されているデータが転送される．測定は，スレッド数 32 集約無し，スレッド数 32 最大集約

Vol.2015-GN-93 No.21
Vol.2015-CDS-12 No.21
Vol.2015-DCC-9 No.21
2015/1/27

数 10，スレッド数 32 最大集約数 100 で行った．非集約は最大集約数 1 と同義である．結果を図 10 に示す．図の横軸はスレイブ端末の台数を，各線は非集約，集約数 10 個，集約数 100 個のそれぞれを表している，縦軸は得られたスループットである．図から，集約数を 10 倍にすると Wi-Fi 通信の速度が 10 倍に近い程度で上昇していることが分かる．次に，I/O 要求を行った場合と行わない場合両方の各最大集約数における 1 秒辺りの平均 TCP パケットサイズを図 9 に，1 秒辺りの転送パケット数を図 10 に示す．図より，集約を行わない状態では非常に小さいパケットが多数転送されていることが分かる．Wi-Fi は CSMA 方式の一つである CSMA/CA 方式を用いて動作をしており，多端末環境で平均フレームサイズが小さい通信を行うと高い性能が得られない[1][2]．よって集約により性能向上が実現されたと考えられる．

以上より，タブレットクラスタを工夫なく構築してもハードディスクと比較して十分に高いランダム I/O スループットが得られること，その場合 Wi-Fi 通信がボトルネックとなりマルチスレッド化による遅延の隠蔽を試みても効果に限界があること，パケットの集約により，スレイブ端末数に比例する性能が得られる(本稿の範囲では 1 台から 8 台までにおいて)ことが分かった．

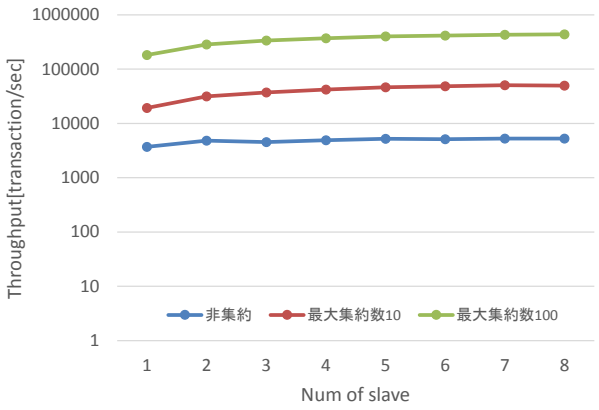

図 8　Wi-Fi 転送速度

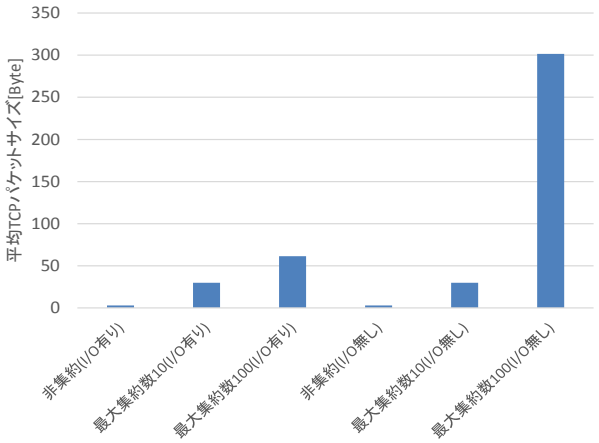

図 9　平均 TCP パケットサイズ

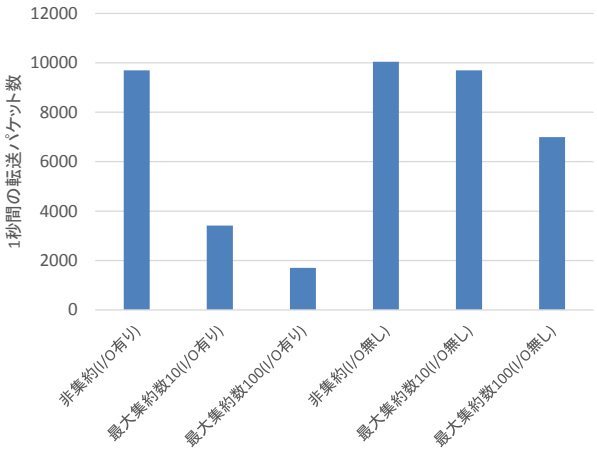

図 10　1 秒間の転送パケット数

4. 関連研究

早水らは，非常に多く(40 台)のフラッシュメモリを搭載したシステムを構築し，並列 I/O 処理性能の評価を行っている[3]．そして，同システムにより非常に高い I/O 性能(454,000[iops])を達成できることを示している．当該研究はハイエンドシステムにおける I/O 性能の向上を目指しており，安価なタブレット PC を活用して安価に高スループット I/O システムを構築することを目指す本稿の研究とは研究の目的が異なっている．よって，当該研究においては本稿で行われたような Wi-Fi 通信を用いての評価や，通信フレームサイズを含む考察はなされていない．

スマートフォンやタブレット PC のフラッシュメモリの I/O 性能に関する研究としては，文献[4]の I/O 性能の評価，I/O スケジューラの改変による I/O 性能向上手法の提案がある．当該研究では，フラッシュメモリを搭載したスマートフォンやタブレット PC に着目し，I/O 性能評価などを行ってる．しかし，それらをクラスタリングして高い I/O スループットを実現することに関しては考察がされていない．

5. おわりに

本稿では，高いランダムアクセス性能を持つフラッシュメモリを搭載しており，かつ安価である Android タブレットに着目し，Android タブレットをクラスタリングして高スループットランダムアクセス処理環境を構築することについての考察を行った．単一 Android タブレットと Wi-Fi 接続の Android クラスタのランダムアクセス処理性能の評価を行ったところ，I/O 要求の集約などにより Android タブレットクラスタにより高いスループットのランダムアクセス処理システムの構築が可能であることがわかった．

今後は，write 処理に関する考察，CPU を用いる処理の性能に関する考察を行っていく予定である．

Vol.2015-GN-93 No.21
Vol.2015-CDS-12 No.21
Vol.2015-DCC-9 No.21
2015/1/27

謝辞

本研究は JSPS 科研費 24300034, 25280022 の助成を受けたものである.

参考文献

1) Bianchi, G., "Performance analysis of the IEEE 802.11 distributed coordination function," Selected Areas in Communications, IEEE Journal on , vol.18, no.3, pp.535,547, March 2000

2) Tourrilhes, J., "Packet frame grouping: improving IP multimedia performance over CSMA/CA," Universal Personal Communications, 1998. ICUPC '98. IEEE 1998 International Conference on , vol.2, no., pp.1345,1349 vol.2, 5-9 Oct 1998

3) 早水悠登，合田和生，喜連川優,"フラッシュストレージ環境におけるアウトオブオーダ型データベースエンジン OoODE の実験的クエリ処理性能評価",
電子情報通信学会第 6 回データ工学と情報マネジメントに関するフォーラム(DEIM2014)， A2-3, 2014

4) Yuta Nakamura, Kyosuke Nagata, Shun Nomura, Saneyasu Yamaguchi, “I/O Scheduling in Android Devices with Flash Storage”, ICUIMC '14 Proceedings of the 8th International Conference on Ubiquitous Information Management and Communication
Article No. 83

正誤表

下記の通り，誤記がありましたので訂正いたします。

正誤箇所	誤	正
P5 図番号	図 8，図 9，図 10	図 10，図 11，図 12
P5 3.5 説中の文章	次に，I/O 要求を行った場合と行わない場合両方の各最大集約数における 1 秒辺りの平均 TCP パケットサイズを図 9 に，1 秒辺りの転送パケット数を図 10 に示す.	次に，I/O 要求を行った場合と行わない場合両方の各最大集約数における 1 秒辺りの平均 TCP パケットサイズを図 11 に，1 秒辺りの転送パケット数を図 12 に示す.