2004－HPC－99　(27)
2004／7／31

小規模 PC クラスタチューニング

福永　隆文*　　梅野　英典**　　芦原　評**

最近，低価格な PC クラスタが注目を浴びている．しかし，各 PC 間でやりとりする情報量が多い並列処理では，その通信遅延がボトルネックとなり，性能が向上しない．そこで，我々は PC 同士をクロスケーブルで接続するクロス接続方式を提案する．さらに，Gigabit Ethernet で用いられている受信割り込み遅延機能を無効化し，Channel Bonding を利用した．今回 4 台の PC クラスタ上にクロス接続方式を実装した．NAS 並列ベンチマーク IS(クラス A)の評価では 2.5 倍の性能向上を得ている．

Tuning of Small-scale PC-cluster

Takafumi Fukunaga,* Hidenori Umeno,** Hyo Ashihara**

Recently, a low-priced PC-cluster has attracted much attention. However, parallel processing that requires a lot of communication data can't improve its performance because of communication delay. In order to overcome such a bottleneck, we propose a cross connection method that connects PCs with cross cables. Furthermore, we ignored receive interrupt delay function that is used on Gigabit Ethernet and used Channel Bonding. This time, cross connection method has been implemented on a 4 node PC cluster. The results show that the IS (class A) performance on this cluster is 2.6 times faster than conventional system using switch.

１．はじめに

近年，PC/AT 互換機等（以後 PC）の高性能化及び通信性能の向上にともない多数の PC をクラスタ結合した並列システムが広まりつつある．この PC クラスタは超並列計算機に比べ，コスト，主記憶の拡張性，スケーラビリティ，新しい CPU 導入の容易さにおいて利点がある．また，普段使い慣れている環境なのでユーザが使いやすいという長所もある．ネットワーク機器においても，Switch を用いた Gigabit Ethernet がコモディティ LAN として定着しつつある．

しかし，Gigabit Ethernet を用いた PC クラスタにおいても，通信性能（遅延，バンド幅）が十分ではなく，高いバンド幅や低遅延を必要とする並列処理は台数効果が低い．そこで，Myrinet[1] を用いた AM[2]，FM[3]，PM[4), 5] などのように特殊なハードウェアに特化したネットワークプロトコルとデバイスドライバを用いて，通信性能向上を実現している．また，GigaE PM[6] では Gigabit Ethernet を用いているが，通信性能を引き出すために特化したプロトコルを用いている．実際に導入する場合は，既存機器を特殊なハードウエアへ切り替え，既存 OS へかなりの修正を加えなければならず，安定稼働するまでにはコストと時間を要する．

そこで，我々は小規模クラスタ環境において，特殊なハードウエアは用いず，さらに Switch を必要としないコストパフォーマンスが高いシステムを提案する．本提案方式（以後クロス接続方式）は２～８台の PC をクロスケーブルで直接接続し，Switch による遅延，帯域制限を除去するとともに，ネットワークインターフェースカード（以後 NIC）の受信割り込み遅延機能の無効化，Channel Bonding[7]（以後 Bonding）の利用というチューニングを行うものである．クロス接続方式の導入に必要な機器は，既存の環境から Switch を除いたものである．ただし，必要な NIC の枚数は接続を行う PC 台数分必要となる．

4 台の PC クラスタ上で NPB2.3 を用いて測定を行った結果，クロス接続を用いることにより，IS で 2.5 倍，CG で 1.4 倍の性能向上を示すことが分かっ

＊　熊本県立技術短期大学校
Kumamoto Prefectural College of Technology
＊＊　熊本大学
Kumamoto University

た．さらに，受信割り込み遅延機能を無効化することによりISで1.05倍の性能向上を示した. Channel Bonding 利用の測定は PC２台を用いて行ったが，1.21 倍の性能向上を示した．

本報告の構成は，２章で従来の方式の問題点，３章で提案方式，４章で評価結果，５章で拡張性について考察し，６章でまとめる．７章で今後の課題について述べた．

２．従来方式の問題点

PC クラスタにおいて通信手段として高速な Gigabit Ethernet が使われる例が増えてきた．しかし，それでもなお，通信のオーバヘッドは計算処理に比べて高く，通信性能（遅延，バンド幅）は未だ十分ではない．その結果 PC クラスタで処理するより１台で処理した方が速いというケースも出てくる．ひとつの原因として TCP/IP プロトコル処理のオーバヘッドが考えられる．TCP/IP は信頼性，安定性のある通信プロトコルとして広く用いられている．しかし，PC クラスタにおいて TCP/IP を用いた場合，高いオーバヘッドにより通信に高い遅延が生じるとともに，NIC 本来のバンド幅性能が引き出せない場合もある．また，通信に CPU の多くの時間を割り当てると，計算処理へ費やす CPU アイドル時間が低下する．

そこで，従来，PC クラスタの性能向上のため低遅延，高バンド幅，低いオーバーヘッドを目的とした特殊なハードウェア，特化したネットワークプロトコル，デバイスドライバ，ライブラリが使われている．しかし，特殊なハードウェアを用いることは導入コストを増加させ，汎用性を低下させるとともに，新しい CPU 導入も容易ではなくなる．また，特殊なハードウェアを用いなくても，特化したネットワークプロトコル，デバイスドライバ，ライブラリを用いれば汎用性が低下するとともに，長年培われてきた TCP/IP の安定性の恩恵を受けることができない．そこで，近年普及してきた Gigabit Ethernet を用いて PC クラスタを構築するケースも多々見られるが, Switch を含めた通信オーバヘッドのため，高い台数効果が得られない現状である．

次に受信に伴う割り込み処理について述べる．本来 Ethernet 上の受信処理において，受信側はフレームを受信するたびに受信割り込み処理を行う．Fast Ethernet まではこの方法で問題はなかったが，Gigabit Ethernet の高速通信では，受信割り込みが頻繁に起こるために CPU に多大な割り込み処理負荷をかけることになる．そこで，最近の Gigabit Ethernet NIC では，複数のフレームが NIC に到着

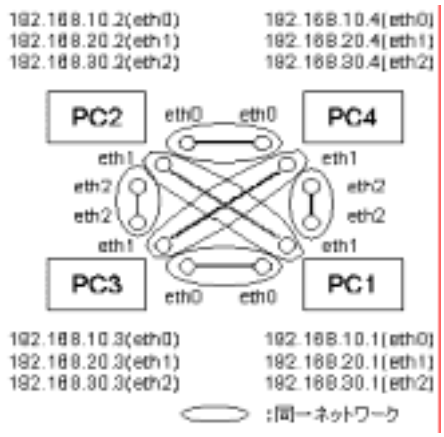


図１　クロス接続方式
（提案方式）

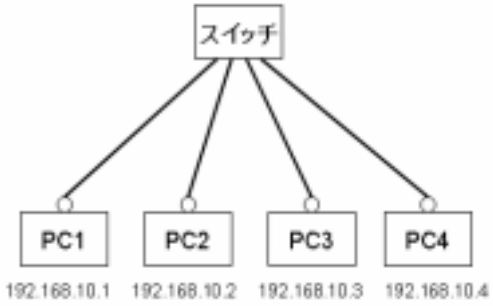


図２　Switch を用いた方式
（従来方式）

した後，割り込みを発生させ，複数のフレームをまとめて受信するという「受信割り込み遅延」を用いることで，割り込み処理による CPU への負荷を軽減させている．

この「受信割り込み遅延」はほとんどの場合, CPU の負荷を軽減させるという意味でパフォーマンス向上に寄与している．しかしながら，PC クラスタシステムにとって，並列性能のボトルネックは通信遅延にある．よって，高大域幅，低遅延を要求する並列処理にとって，フレームの受信が待たされることはパフォーマンスに多大な影響を与える．

３．提案する方式

今回我々は，小規模クラスタ環境上で，既存 NIC 及び TCP/IP をそのまま用いながら，通信負荷を複数 NIC に分散させるとともに Switch 仲介による通信性能低下を除外するクロス接続方式を提案する．更に，NIC の受信割り込み遅延機能を無効化し，Bonding を利用する．

３．１　クロス接続方式の概要

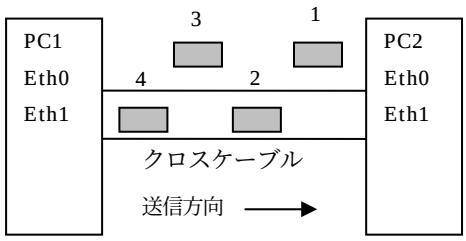


図３ Channel Bonding 接続例

クロス接続方式を用いた構成例を図１に示す．従来の Switch を用いた構成例を図２に示す．eth0～eth2 は NIC を表す.

クロス接続方式では各 NIC は別のネットワークセグメントに属する．よって送信時に経路制御により目的の PC へルーティングされる. 各 PC は IP フォワーディング機能を OFF にしておく必要がある．そうすることにより対向する PC への経路は１つに限定される．Switch を経由しないため，Switch 内での遅延，および高速通信時の Switch 内バンド幅による制限を受けることがない．また，Bonding 技術や Jumbo Frame 技術を用いる場合，経路上の Switch が当技術に対応している必要があるが, 今回 Switch を用いないためその必要はない. 後述するが今回 Bonding 技術を用い, クロス接続方式をチューニングした.

３．２　チューニング

Gigabit Ethernet 上のクロス接続方式において, 更に受信割り込み遅延（以後，割り込み遅延）機能の無効化，Bonding の利用という２点のチューニングを行った.

３．２．１　割り込み遅延機能の無効化

NIC の「割り込み遅延」を OFF にすることで, 遅延を縮小した. CPU の割り込み処理回数は増えることになるが, 遅延縮小による効果が高いと考えた.

今回 NIC として，Intel PRO/1000 MT Server Adapter を用いた. 当 NIC はモジュールを組み込む際にパラメータを指定することにより割り込み遅延時間を操作できる.

３．２．２　Channel Bonding の利用

Channel Bonding は Bonding と呼ばれるドライバを用い, 対向する PC との通信に複数の NIC と複数の通信線を用いることで通信のバンド幅を向上させる手法である．2 台の PC をクロスケーブルで接続する場合の Bonding 接続例を図３に示す. 送信データは交互に 2 枚の NIC に振り分けられる．そのため, 先行するフレームを NIC が送信中であっても別の NIC で送信することができる．受信側では 2 枚の NIC で受信することになる．Switch を用いた構成も可能である．Bonding ドライババージョンの違いにより方式に多少差があるが，今回は束ねる NIC の MAC アドレスを同一のものに書き換える Bonding ドライバを用いた. 今回の NIC は Gigabit Ethernet のため, 当方式でバンド幅を向上させるためには NIC を高速な PCI-X に接続する必要がある.

表１　測定環境

ハードウェア	Pentium 4 3.0GHz,Intel865G ﾁｯﾌﾟｾｯﾄ,1GB DDR SDRAM ﾒﾓﾘ, MSI 865G Neo2 ﾏｻﾞｰﾎﾞｰﾄﾞ
NIC	Intel PRO/1000 MT Desktop Adapter, ﾄﾞﾗｲﾊﾞﾊﾞｰｼﾞｮﾝ 4.3.15
OS	Redhat 8 Linux(2.4.18 ｶｰﾈﾙ)

表２　測定環境

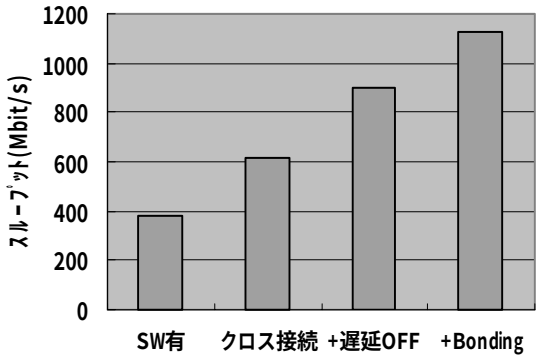
ハードウェア	Xeon 3.06GHz,IntelE7501 ﾁｯﾌﾟｾｯﾄ,2GB DDR SDRAM ﾒﾓﾘ, SUPERMICRO X5DPE-G2
NIC	Intel PRO/1000 MT Server Adapter, ﾄﾞﾗｲﾊﾞﾊﾞｰｼﾞｮﾝ 4.3.15
OS	Redhat 8 Linux(2.4.18 ｶｰﾈﾙ)

４．評価

この章では以下の 4 つの PC クラスタについて, 通信性能及び並列処理性能を評価したので，その結果を示す.

(a) 通常の Gigabit Switch を用いた PC クラスタ
(b) クロス接続方式を用いた PC クラスタ
(c) (b)の割り込み遅延機能を無効化した PC クラスタ
(d) (c)に Bonding を利用した PC クラスタ

表１，表２に示す２つの測定環境を用いた．２つの測定環境を用いた理由を述べる．表１は標準的な PCI バスを用いた環境であり，表２は高速な PCI-X バスを用いた環境である．表２の測定環境は Bonding を利用した実験に必要である．表１環境の PCI バスでは 1Gbps を超える通信速度をだす Bonding を利用できないためである．しかし，表２の測定環境を満たすパソコンは２台しかないため, その他の実験は表１を満たすパソコン４台を用いて行った．２台より４台の方が多くの実験結果が得られるためである．なお，表２環境では CPU を 2 つ搭載しているが, 実験の際には 1CPU 用カーネルを利用した．よって計算処理能力は表１環境と表２環境は同程度といえる.



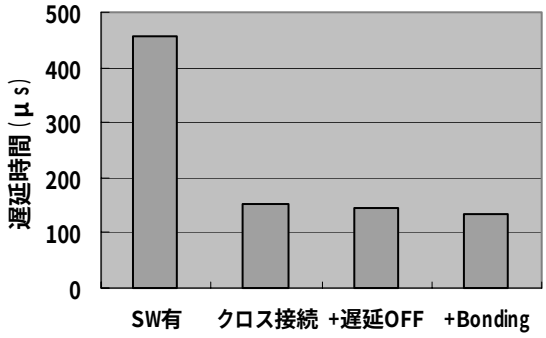


図5　遅延性能比較

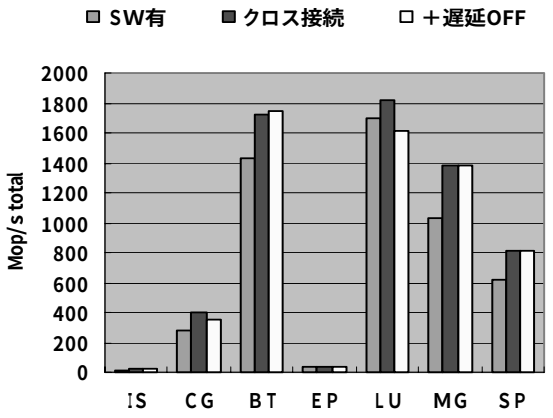


図6　並列処理性能比較（PC4台）

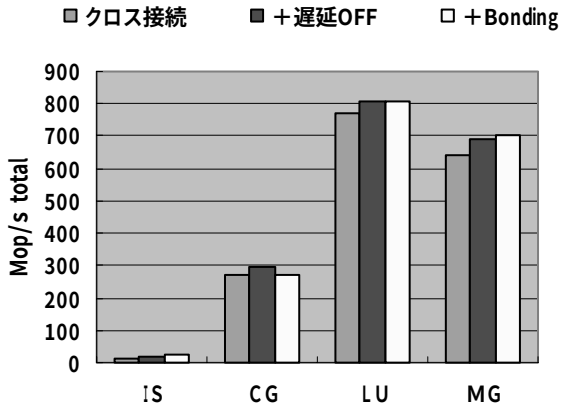


図7　並列処理性能比較（PC2台）

４．１　通信性能

バンド幅，遅延に関する測定結果を図４，図５に示す．バンド幅はメッセージ長1,024バイトで測定した．Switch はレイヤ３Switch である Cisco Catalyst 3550 12T を用いた．測定ツールは netperf-2.1pl3[8] を用いた．「SW有」は前述した(a)のクラスタであり，「クロス接続」は(b)，「+遅延OFF」は(c)，「+Bonding」は(d)の環境を示す．Bonding 利用時のみ表２仕様 PC であるが，CPU 性能，LAN 環境は同等のため，同一グラフに表記した．

図４の結果より，バンド幅性能はクロス接続，＋遅延 OFF,+Bonding の順に性能が飛躍的に伸びている．このことは高バンド幅を必要とする並列処理に有効である．ただし，バンド幅性能に比例して CPU 使用率は増大する．CPU 使用率の増大は計算処理負荷が大きい並列処理性能に悪影響を及ぼす．よって，必ずしも並列処理にプラスの面だけ与えるわけではない．

遅延は小さい方が望ましいが，図５の結果より，クロス接続は SW 有に対して大きく性能が向上している．これは Switch が存在しないので，その分遅延が小さくなるのは当然の結果である．ただし，Switch によっては今回用いた装置より遅延が小さいものがある．我々の測定ではレイヤ２Switch である NETGEAR GS516T を用いた場合，遅延は今回の結果より良好であった．ただ，いずれにしてもクロス接続方式が遅延は小さい．割り込み遅延機能を OFF にしても，Bonding を用いても遅延性能に違いはなかった．この２つは遅延を向上させる手法ではないので，当然の結果である．Bonding を用いた場合，わずかに向上しているように見えるが，これは PC 環境の違いであると予測できる．

４．２　並列処理性能

並列処理性能比較を図６，図７に示す．ベンチマークとして NAS Parallel Benchmarks(NPB) V2.3[9] を用いた．NAS Parallel Benchmarks は８つのベンチマークがあるが，今回は IS(Integer Sort), CG(Conjugate Gradient)，EP(Embarrassingly Parallel), MG(Multigrid), LU(LU Simulated CFD Application), BT(BT Simulated CFD Application), SP(SP Simulated CFD Application)について測定した．また，ベンチマークは測定対象システムの規模に応じたクラスがある．主なクラスは小規模ワークステーションクラスタ用（クラス W），中規模ワークステーションクラスタ用（クラス A），中規模並

列計算機用（クラス B），大規模並列計算機用（クラス C）である．
当システムは小規模クラスタであるが，通信負荷が高い並列処理の効果を測定するため，今回は中規模クラスタ用ベンチマークであるクラス A を用いた．表１仕様の PC４台及び Switch を用いて，クロス接続の場合，更に割り込み遅延機能を OFF にした場合の結果を図６に示す．図７は表２仕様の PC を２台用いて，クロス接続の場合，更に割り込み遅延機能を OFF にした場合，更に NIC2 枚を用いた Bonding の場合の測定結果である．Bonding の効果については図７より考察する．

図６よりクロス接続方式は Switch を用いた構成よりよい結果を示している．IS では 2.5 倍，CG では 1.4 倍の性能向上を示した．クロス接続に割り込み遅延機能無効化を追加した場合は，IS で 1.05 倍を示したが，CG,LU では性能が低下した．これは，割り込み遅延機能を OFF にしたために，CPU に対する割り込み処理負荷が増大したため，計算処理性能が低下したためと考えられる．高バンド幅と低遅延を必要とする IS にのみ効果を見ることができた．

図７は PC２台を用いた結果である．BT，SP は２台では実行できず，EP はほとんど通信を行ず変化は認められないので結果は省略した．＋遅延 OFF と＋Bonding を比較して Bonding の効果を考察する．IS において Bonding を用いた場合 23.8Mop/s を示した．＋遅延 OFF の場合が 19.6Mop/s であるので，Bonding により 1.21 倍の性能向上が達成できたことになる．その他のベンチマークでは性能向上はみられなかった．CG では 10%の性能低下の結果となった．これは，CPU 使用率が 100%を示す瞬間が複数見られたことから，Bonding の利用がもたらす通信性能向上より CPU の負荷向上による計算能力の低下が大きく CG の並列処理に影響を与えたためと思われる．しかし，高バンド幅，低遅延を必要とする IS において効果がみられたことは，並列処理の種類よっては Bonding を組み込む効果があることを示す．ただし，IS はその通信負荷の高さのため，台数効果が得られにくい．今回の環境では PC を 4 台用いるよりも 1 台の場合が高い性能を示した．表２の環境で PC１台の時，IS は 30.8Mop/s を示した．IS で並列処理性能を向上させるためには，さらなるチューニングが必要である．

５．拡張性

クロス接続方式は PC 同士を直接クロスケーブルで接続するため，台数に制限がある．一部に Switch を用いることで何台も接続することはできる（図８）

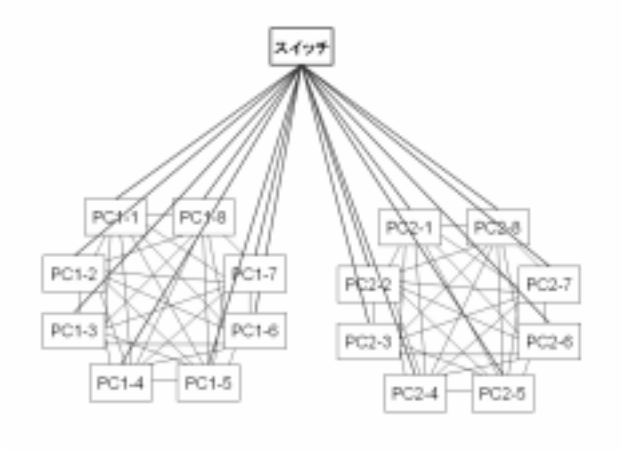


図８　拡張したクロス接続

が，当接続方式ではクロス接続による効果は小さくなる．　拡張性を必要とする場合には，クロス接続方式を用いず，割り込み遅延機能無効化，Bonding の利用に絞ってチューニングする必要がある．

６．まとめ

小規模 PC クラスタにクロス接続方式，割り込み遅延機能無効化，Bonding 方式を導入し，並列処理性能のチューニングを行った. NPB を用いて測定した結果，クロス接続方式は通信負荷が高いすべてのベンチマークについてよい性能を示した．IS で 2.5 倍，CG で 1.4 倍の性能向上である．割り込み遅延機能無効化及び Bonding 方式の利用では共に高バンド幅，低遅延を必要とする IS でのみ，効果が得られた．それ以外のベンチマークでは性能に変化がみられないか, 性能低下の結果となった. これは，チューニングにより CPU のアイドル率が低下し，計算処理がボトルネックとなったためと考えられる．後者２点のチューニングについては動的な制御が今後の課題となる．

７．今後の課題

今後の課題として，割り込み遅延機能の動的切り替えがまず挙げられる．この場合，単に機能の ON,OFF でなく，遅延時間の長さを動的に切り替えることが並列処理向上につながると考えられる．切り替えのタイミングとして CPU 使用率の利用が挙げられる．受信割り込み禁止間隔を広くすれば，実際の割り込み回数は少なくなり, CPU 使用率は低くなる．逆に割り込みを頻繁に許せば CPU 使用率が高くなる．しかし，過度に CPU 使用率を低くする必要はないと考える．計算処理に悪影響を及ぼさな

い範囲内で割り込みを頻繁に許した方が並列処理速度向上につながると思われる．よって，CPU 使用率が十分低い時は割り込みを頻繁に許し，CPU 使用率が高いときは割り込み禁止間隔を広くする方式が考えられる．現在この点についてドライバなどの修正を行い検証している．

参考文献

1) Boden, N.J., Cohen, D., Felderman, R.E., Kulawik, A.E., Seitz, C.L., Seizovic, J.N. and Su, W.-K.: Myrinet – A Gigabit-per-Second Local-Area Network, IEEE MICRO, Vol.15,No.1, pp.29-36 (Feb. 1995).
2) http://now.cs.berkeley.edu/AM/lam_release.html.
3) Pakin, S., Lauria, M. and Chein, A.: High Performance Messaging on Workstations: Illinois Fast Messages(FM) for Myrinet, Proc. Supercomputing '95, San Diego, California (1995).
4) Tezuka, H., Hori, A., Ishikawa, Y. and Sato,M.:PM: An Operating System Coordinated High Performance Communication Library, Sloot, P. and Hertzberger, B. (Eds.),*High-Performance Computing and Networking,*Vol.1225, Lecture Notes in Computer Science,pp.708-717, Springer-Verlag(Apr. 1997).
5) Tezuka, H., O'Carroll, F., Hori, A. and Ishikawa, Y.: Pin-down Cache: A Virtual Memory Management Technique for Zero-copy Communication, *IPPS/SPDP'98,* pp.308-314,IEEE(Apr. 1998).
6) 住元真司，堀敦史，手塚宏史，原田浩，高橋俊行，石川裕: GigaE PM: Gigabit Ethernet を用いた高速通信機構の設計と評価，情報処理学会論文誌，Vol. 41, No. 5, pp. 1390-1399 (2000).
7) T.Sterling, D.Savarese, D.J.Becker, B.Fryxell, K.Olson. Communication Overhead for Space Science Applications on the Beowulf Parallel Workstation.In Proceedings of the Fourth IEEE Symposium on High Performance Distributed Computing(HPDC-95),August 1995.
8) http://www.netperf.org/
9) http://www.nas.nasa.gov/Software/NPB/