社団法人 情報処理学会 研究報告
IPSJ SIG Technical Report

2008－MPS－72 (24)
2008－BIO－15 (24)
2008／12／17

大規模複雑なネットワーク分析に適した分散環境

小野泰正[†,?] 林 幸雄[†]

大規模な問題を分散処理で効率良く解くためには問題の分割粒度と相互依存関係によって分散処理システムを考えなくてはならない．ネットワークダイナミクスの問題の多くはパラメータ問題でありその相互依存度が低く粒度の細かい分割ができる．この特徴を使ったスケーラブルな分散環境を作り負荷の異なる問題を効率良くために動的負荷分散を行う．結果，簡潔な記述による分散処理環境の構築と効率の良いネットワークダイナミクスシミュレーションを提案し，そのシステムを使った結果，ASネットワークでのサイバーテロに相当する攻撃に対しての危険を示唆する．

Distributed processing environment for large, complicated network analyses

YASUMASA ONO[†,?] and YUKIO HAYASHI[†]

Dependence and granularity is important for task divide of distributed processing of large and complex network analysis. it almost is the issue of parameter, so it can be split to low interdependence and the small granularity. We make distributed computing suitable for such a lot of careful tasks. It promotes efficiency of network dynamics simulation. We examined about danger of the cyberterrorism by the experiment that became possible by the system. Result, We warn that, AS network are in a dangerous state for cyberterrorism.

1. はじめに

現在，通信環境の発達によりインターネットを始め，通信インフラは社会になくてはならない物になっている．一方，近年の計算機の性能向上により大規模ネットワークの構造分析が進み，インターネットを始め現実の多くのネットワークは，次数分布がべき乗則に従うScale-Free(SF) 構造を持つことが明らかになった．そして，この構造は少数のハブを持つことが特徴であるが，この構造に注目した様々な研究により，通信効率は良いがハブに高負荷が掛かり，渋滞現象が起りやすいことや，SF構造は不慮の事故には非常に強いが悪意のあるテロ攻撃等には脆い事が分かった[3]．この様な問題に対処するために，ネットワーク耐性を強化する対策としては，次世代無線技術を使った新しい構造や，新しいプロットコル等が，求められている．このような開発には数値実験が必要であるが，ネットワークのトポロジー構造，パケット転送のトラフィック特性，故障・攻撃に対する結合耐性等の分析には，様々な問題定義や性能の違いなどの組み合わせによって膨大な計算量が必要である．これを複数台での分散処理を行う際には，問題の分割粒度を捉え，相互依存度が少ない分割の仕方を考えることが重要である．本研究で扱うネットワークダイナミクスのシミュレーションはパラメータ問題であるため粒度が細かく相互依存が少ない問題に分化することが可能である．

† 北陸先端科学技術大学院大学
知識科学研究科
Japan Advanced Institute of Science and Technology
Shool of Knowledge Science

2. 分散環境の構築

Javaのオブジェクト指向によって分散処理を簡潔に書く手法は現在様々な分野で行われ，例えばRMI，CORBAやHOBEによる分散処理は，分散オブジェクトと呼ばれるネットワーク上に散らばったオブジェクトを利用する技術である．代表的な分散オブジェクト技術を表1に示す，RMIはJavaと共に自動的に導入されるメリットがあるがスタブ・スケルトンの生成・配置が必要だった．一方，CORBAはJavaのみならずCやFortran等の多言語での使用と，オーバーヘッドの少ない通信がメリットであるが，IDLによる複雑な分散処理用の設定をしなくてはならない．HORBはリモートサーバーのオブジェクトが自動生成される点が特徴であり，比較的簡単な導入ができるが，サポート体勢が弱い事が難点である．

2.1 RMIによる分散処理

J2SE 1.5までのJavaRMIは，Sun Javaを導入す

表 1 分散オブジェクト技術
Table 1 distributed object technology

	実行方式	言語	環境作成要項
RMI	接続型	java	java のインストール
CORBA	接続型	多種	CORBA のインストール
			IDL でのインターフェイスの記述
HORB	生成型	java	HORB のインストールと設定

接続型：リモートサーバーへホストサーバーが接続して使う
生成型：リモートサーバーにオブジェクトが自動生成されて使う.

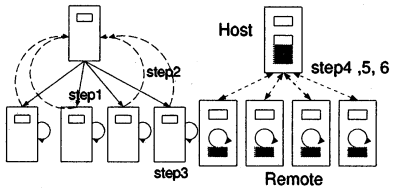

図 1 ダイナミックコードローディングを使った分散環境の構築
Fig. 1 Dynamic code loading for Distributed computing

ると同時に使用できるメリットがあるが, スタブ及びスケルトンの生成・配置や, rmiregistry というリポジットの常駐が必要であることなどから, 分散処理を行う言語としては扱いにくかった. しかし, その後の RMI の発展により J2SE 1.5 からは, スタブ, スケルトンの動的生成, 及び rmiregistry の呼出, ダイナミックコードローディングの実装などで, リモートサーバーは RMI の呼出窓口である rmid の設置と使用するポートを開放しておけば, HORB と同様にクライアントから動的にサーバーを起動し使用できる様になった. この仕組みを使うと, ホストサーバーがパラメータをプールしておくだけでリモートサーバーが必要に応じてパラメータを取得する様に, リモートサーバからのコールバック機能にパラメータ取得機能を追加するだけで, ワークプール型の動的分散処理手法が実現できる. これは, 以下のような手順で実行できる (図 1). (STEP.1) ホストサーバーがリモートサーバを呼び出す. (STEP.2) リモートサーバーがタスクオブジェクトを取得. (STEP.3) リモートサーバーが起動し rmiregistry を起動しタスクを登録する. (STEP.4) リモートサーバーがコールバックしホストサーバーへの通信を開く. (STEP.5) プールされているパラメータを取得しタスクを実行する. (STEP.6) プールされているパラメータがなくなるまで STEP. 6 を繰り返す. STEP1〜4 で, RMI における分散オブジェクトの生成型を実装し, STEP4〜6 でワークプール方式の分散処理を行っている. このように, 簡潔な記述で分散処理環境が構築でき, かつタスクスケジュールをリモートサーバーが行うことで, 通信オーバヘッドを少なくしている. この分散処理環境のでは, 例えば本研究の実験例のように, ネットワーク構造や攻撃手法などに関するパラメーターの組合せに応じてサーバがオブジェクトを呼だし実行する事で, 様々な実験が動的に行えるメリットを生む.

2.2 問題と改善

本研究が提案するシステムでは, タスク実行の終えたリモートサーバーがタスクを取得することで, ホストサーバーの問い合わせ処理を軽くすることが可能である. しかし, この方式は, ホストサーバーはタスクの監視を行っていないため, 故障したり通信が切れた場合にタスクの終了を永遠に待つことが問題となる. そこで, ホストサーバーは定期的にリモートサーバに問い合わせをし, もし実行が確認できなかった場合, パラメータをプールに再度入れ直す事で再実験を促すようにした. また, タスク実行中の暴走を想定し, 同じ負荷の処理を複数回試行したり, 大幅に超す場合にはパラメータを再度プールに投入して, リセットする機構を導入した.

3. 実 験 例

3.1 結合耐性のシミュレーション

インターネットは, ハブが除去されるとネットワークが極度に分断してしまうことが明らかになり [3], いくつかの対策が検証されている [4] が, 本研究では, このハブ攻撃をサイバーテロの観点から考えてみる. サイバーテロの基本的な手口はホームページのバナーやウィルス感染によるバックドアによってゾンビ PC を作りだし, その PC から悪意のあるプログラムを接続関係のある PC に送りつける事が報告されている [1]. 最近は, 金銭の詐取を目的としたものが多く通信被害にはあまり至っていなが, 同様の手口で PC やシステムをダウンさせることができ, 非常に危険である. このような不正アクセスの対策に関する研究は, 個々のセキュリティーレベルを上げることを目的とするミクロレベルの危険性の研究と, ネットワーク構造に注目したネットワークの耐性 [3] や頑健なネットワーク構造や改善方法を探るマクロレベルの研究があり [4], 本稿では後者を扱う. ネットワークの耐性については, 実測された接続関係を元に, 不慮な故障や悪意のあるテロ攻撃を想定し, 被害規模をシミュレーションによって探る研究が広く行われている. ネットワーク全体が一つにまとまった, 連結成分である初期状態から頂点を段々除いていくと, 途中で孤立した島のような部分クラスターができてくる. その中で最大の連結成分を持つも

表 2 AS 基本データ
Table 2 Basic data of AS

	N	M	$\langle k \rangle$	$\langle L \rangle$
2004	17509	35829	4.092	3.77
2005	19846	40485	4.079	3.78
2006	22456	45050	4.012	3.86

のが, 分離した他の小さな成分と同程度のサイズになると, 完全に機能不全なバラバラな状態となる. 連結成分のサイズは接続関係にある頂点数を数えることで求められため, 最大連結成分 (GC:Giant Compornent) を除いた島の平均サイズ $\langle s \rangle$ は, 島の数を N_{island}, 島の大きさを S_i とすると,

$$\langle s \rangle = \frac{\sum_{i=1}^{N_{island}} S_i - GC}{N_{island} - 1} \quad (1)$$

で求められる.

3.2 ネットワーク

インターネットに関する実データは, Cooperative Associatin for Internet Data Analysis(CAIDA) によって収集されている. 本研究では, AS レベル (大学やプロバイダー等の拠点) での近年の 2004,2005,2006 年のそれぞれ 7 月に収集されたデータを使う. 各年の基本特性として, 頂点数, 総辺数, 平均次数, 最短経路のホップ数での平均経路長を表 2 に示す. このように, 年を重ねる毎に頂点数や総辺数の増加が見られネットワークは成長している. しかし, 構造的な変化はあまり見られない.

3.3 攻撃手法

不慮な故障や悪意のあるテロ攻撃を想定し, 被害規模をシミュレーションによって探ることが広く行われているが, 図 6 の様に, 以下の 4 つの場合に分けインターネットの AS レベルの実データ [CAIDA] に対する結合耐性を調べる. (Case1:強サイバーテロ攻撃) ランダムに頂点を選択し隣接する頂点を全て除去する. (Case2:弱サイバーテロ攻撃) ランダムに頂点を選択し隣接する頂点の中からランダムに一頂点を選んで除去する. (Case3:不慮の事故) ランダムに選択された頂点を除去する. (Case4:テロ攻撃) 次数に即した確率で頂点を選択して除去する.

3.4 タスクの分割化

本研究では, テロ攻撃や不慮の故障などに対する AS ネットワークの結合耐性を調べるには, 上記した 4 つの攻撃方法と, その対象となる 3 つのネットワークにより, 12 の組合せにおいて実験を行う. 本研究では, 確率的な頂点選択を行うが, 確率を使う実験では複数回実行されることが一般的であるので, それぞれ１００回試行した. 分散処理のタスクの分割では, できるだけ粒度の細かく依存度の低い分割が求められる. 本実験は, 攻撃度合に則して徐々に変化したネットワーク構造の連結成分を計測することが主なプロセスとなる. このうち, 攻撃度合を増やすプロセスは依存度が高いため分割しない. しかし, 各組合せに対し 100 回試行を行った後に集計処理を行うプロセスは, 組合せ間に依存関係がないため, 分割して処理を行う. 従って, 本実験ではネットワークデータと攻撃手法の組合せによる 12 種を各 100 回試行することから, タスクは 1200 分割された.

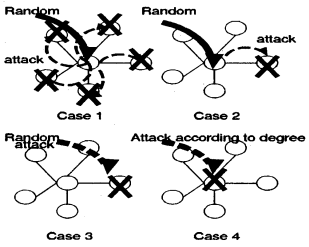

図 2 攻撃手法
Fig. 2 attack method

4. AS ネットワークの頑健性

本実験の結果を述べる. 図 3 は 4 つの攻撃手法による攻撃率に即したノード総数に対する最大連結成分の比と故障率に対する分断された島の平均サイズを示している. インターネットが不慮の故障に対して非常に頑健な構造になっている事や, SF の特徴である少数のハブが除去されるとネットワークが極度に分断してしまうことが指摘されているが [3], 同様な結果が本実験において AS ネットワークにも確認された. 次に, サイバーテロによる攻撃を想定した頂点除去によっても, 早い段階で AS ネットワークの崩壊を引き起こされる事が分かる. 注目すべきは, この結果がハブ攻撃を想定した頂点除去による結果に近似していることである. Case1 の頂点を除去率が次数に即した確率と等価になることは, L.K.Gallos らにより指摘されている [5] ことから同様の被害が出たことは理解できる. しかし, Case1 よりも制限を厳しくした Case2 においても同様に被害が拡大している事は, 少しのセキュリティーホールでも甚大な被害が引き起こされる可能性を示唆している. $\langle s \rangle$ は GC が維持できなくなり, 分断した瞬間 $\langle s \rangle$ が増えるため, 崩壊現象が起こされた臨界値をより明確に見ることができる. この臨界値 f_s を表 3 に示す. 臨界値や図 3 からは, サイバーテロの様な,

表 3 臨界値 f_s 臨界故障数 f_n
Table 3 f_s or f_n

	f_s04	f_s05	f_s06	f_n04	f_n05	f_n06
Case1	0.12	0.11	0.13	2182	2205	3101
Case2	0.14	0.13	0.14	2485	2657	3188
Case3	0.74	0.72	0.72	13059	14330	16242
Case4	0.08	0.08	0.09	1568	1762	2161

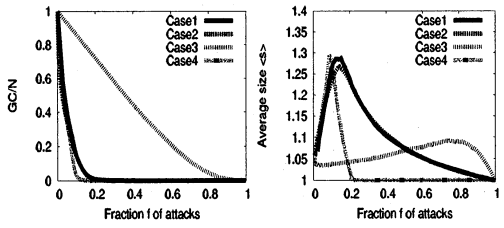

図 3 GC $\langle s \rangle$ 2004 年度の AS
Fig. 3 GC or average size $\langle s \rangle$ of AS(2004)

俯瞰的情報を必要としない攻撃でも非常に大きな被害がもたらされる事が指摘され, かつ Case2 の様に少数の攻撃しか成功しないケースでも被害が拡大することが分かる. しかし, 機能不全に陥いるまでに必要な攻撃数を鑑みると, 表 4 に示されるように, ネットワーク全体からみれば少数ではあるが, 数千の頂点への攻撃が必要である.

5. 分散処理結果

最後に, 本研究におけるシミュレーション実験に掛かった時間の計測によって性能評価をする. 実験環境は CPU Pentium4 2.4GHz MEMORI 512MB をリモートサーバ用 8 台, ホストサーバー 1 台の構成である. 実験では, 前記した方法で 1200 にタスクを分割したが, 同 PC にて 1 タスクの実行に掛かる時間を表 5 に示す. 表 5 の様に, ネットワークデータや手法によって負荷は異なり, 特に手法によっては, 負荷の軽いものに比べ 10 倍もの時間が掛かることもある. 本システム全体の計算のスケーラビリティは, 台数ごとに,1 台:87.08 時間, 2 台:43.17 時間, 4 台:21.58 時間, 8 台:10.91 時間と時間が短縮されることが分かった. 一方, 本システムではワークプール方式による分散処理を行っているがこの方式の問題点として, 計算の最終段階において残存するタスクが少ない状態では, 処理を行わないリモートサーバーが出てくる. しかし, 処理を行わない PC が存在する時間は全体の 1.1 %程度と少なく, 通信オーバーヘッドも, 1 回のタスクオブジェクト転送に対し 1.6ms と非常に小いさく無視できる時間であった. しかし, 本実験が学内 LAN 環境での高速通信網下の実験であることには注意が必要である.

表 4 タスク処理に掛かる時間
Table 4 time of task

	04	05	06
Case1	136027ms	172560ms	218874ms
Case2	147710ms	190437ms	242638ms
Case3	169774ms	214153ms	275954ms
Case4	1409795ms	1756361ms	2233161ms

6. おわりに

ネットワークダイナミクス解析の多様な負荷で膨大な計算量が必要となる. これを効率欲解くために分散処理によって計算資源を確保することは一般的に行われているが, そこで, 重要となるのは問題の分割粒度と相互依存関係である. ネットワークダイナミクスの問題は多くがパラメータ問題である特徴を使い粒度の細く依存度の少ないタスクに分割できる. こうした特徴から本研究では JavaRMI におけるワークプール方式の分散処理環境を構築し, 複雑なタスクスケジューリングを行わずとも, 効率の良い処理が行えることを確認した. また, 分散実験として AS ネットワークの結合耐性をサイバーテロを想定した手法で計測し, その結果, サイバーテロの危険性は, ハブ攻撃に近い被害をもたらすことが分かった. その結果は, AS ネットワークの構造に起因する現象であるので, サイバーテロによる攻撃が実際行われた場合の危険性を示唆すると共に早急な対策が必要な事を訴えたい.

参考文献

1) 情報処理振興事業協会セキュリティーセンター: コンピューターウィルス, 不正アクセスの届け出状況 2008 年 10 月,(2008)
2) A.Wollrath and J.Waldo,R.Riggs:Java-Centric Distributed Computing, IEEE micro Vol17 pp44-53,(1997)
3) R.Albert, H.Jeong, and A.-L.Barabasi,:Error and attack tolerance of complex networks, Nature, Vol.406 pp.378-382,(2000)
4) Y. Hayashia and J. Matsukubob.:Improvement of the robustness on geographical networks by adding shortcuts,Physica A, 380 pp552-562,(2007)
5) L.K.Gallos, F.Liljeros, P.Argyrakis, A.Bunde, S.Havlin:Improving immunization strategies, Phys.Rev.E 75,045104,(2007)
6) CAIDA:http://www.caida.org/home/