

ファイル共有システムにおける 進化ゲーム理論を用いたキャッシングアルゴリズム

笹部 昌弘[†] 若宮 直紀^{††} 村田 正幸^{††}

[†] 大阪大学サイバーメディアセンター 〒560-0043 大阪府豊中市待兼山町 1-32

^{††} 大阪大学大学院情報科学研究科 〒565-0871 大阪府吹田市山田丘 1-5

E-mail: [†]m-sasabe@cmc.osaka-u.ac.jp, ^{††}{wakamiya,murata}@ist.osaka-u.ac.jp

あらまし P2P ファイル共有システムでは、ノードは自分の必要とするファイルを検索、取得する。複数のノードが同一ファイルをキャッシュし、他のノードに提供することによって、低遅延でファイル可用性の高いファイル共有が期待できる。しかしながら、ファイルのキャッシングには処理負荷、ストレージ資源などのコストがかかるため、ノードが利己的に振る舞うと、十分にファイルがキャッシュされず、特に人気の低いファイルがシステムから消失するなどの問題が発生する可能性がある。そこで本稿では、ノードの自律的、利己的な振る舞いによってシステム全体で適切なキャッシングが行われる機構の実現を目指し、進化ゲーム理論により、ノードの振る舞いがシステム全体のダイナミクスに与える影響について検証した。その結果、キャッシングに対するコストと需要のモデルによっては、ノードが利己的に振る舞ったとしてもファイルがシステムから消失することのない、ファイル共有が実現可能であることを示した。キーワード ファイル共有システム, P2P, 進化ゲーム理論, 利己的な制御

Caching Algorithms using Evolutionary Game Theory in a File-Sharing System

Masahiro SASABE[†], Naoki WAKAMIYA^{††}, and Masayuki MURATA^{††}

[†] Cybermedia Center, Osaka University Machikaneyamacho 1-32, Toyonaka-shi, Osaka, 560-0043 Japan

^{††} Graduate School of Information Science and Technology, Osaka University
Yamadaoka 1-5, Suita-shi, Osaka, 565-0871, Japan

E-mail: [†]m-sasabe@cmc.osaka-u.ac.jp, ^{††}{wakamiya,murata}@ist.osaka-u.ac.jp

Abstract In a P2P file-sharing system, a node finds and retrieves its desired file. If multiple nodes cache the same file to provide others, we can achieve a file-sharing system with low latency and high file availability. However, a node has to spend costs, e.g., processing load or storage capacity, on caching of a file. Consequently, a node may selfishly behave and hesitate to cache a file. In such a case, there is a possibility that unpopular files disappear from the system. In this paper, we aim to accomplish effective caching in the whole system that emerges from autonomous and selfish node behaviors. We first discuss relationship between selfish node behaviors and system dynamics according to evolutionary game theory. As a result, we show that a file-sharing system can be robust to file disappearance depending on a model of cost and demand for caching even if nodes behave selfishly.

Key words file-sharing system, Peer-to-Peer (P2P), evolutionary game theory, selfish control

1. はじめに

P2P ファイル共有システムでは、ノード同士が論理的なリンクを確立することによって、論理的なネットワークが構築され、その上でファイルや情報がやりとりされる。それぞれのノードは自分が持っていた、もしくは他のノードから取得したファイルをローカルなストレージに保持(キャッシュ)し、これを他の

ノードに提供する。複数のノードが協調し、あるファイルを複数個所にキャッシュすることができれば、ファイルをより確実に発見し、より早く取得できるようになり、システムの性能、可用性、信頼性などの向上が期待できる [1, 2]。

しかしながら、ファイルをキャッシュするためには処理負荷、ストレージ容量といったコストが発生する。ファイル共有システムを構成するノードはユーザ端末であり、それぞれが自身の判断に基づいて振る舞っているため、ファイルをキャッシュし

ないという非協力的な振る舞いをするノードも存在する [3] . その結果, 例えば, 需要の少ないファイルがシステムから消失し, ノードが利用できなくなるなどの問題が生じる. 適切なキャッシュ状態を保つためにすべてのノードの状態を監視, 制御するのは困難であるため, ノードの自律的, 利己的な振る舞いによって, システム全体として適切なキャッシングが行えるのが望ましい.

ノードの利己的な振る舞いがシステム全体のダイナミクスに与える影響は, ゲーム理論を用いることによって明らかにすることができ, キャッシングに対してもいくつかの検討がなされている. 例えば, 文献 [4] では, ファイルをキャッシュすることで消費するストレージ容量, 他のノードからファイルを取得する時に発生する遅延をそれぞれコストとみなし, ノードがファイルを利用するためのコストを最小化しようと振る舞った場合, システムの安定状態, すなわちナッシュ均衡が必ず存在することをゲーム理論により示している. また, ノードがファイルをキャッシュすることにより, そのファイルを利用するノードから報酬を得ることができるというインセンティブを導入することで, システム全体としての最適な状態がナッシュ均衡になることを明らかにしている. ただし, ナッシュ均衡は複数存在し, 必ずしもシステム全体として最適な状態になるとは限らない. また, 文献 [5] では複数人による囚人のジレンマゲームを用い, ファイル共有におけるノード間の協力的行動について論じており, ファイル共有に対するインセンティブがない場合には, 全員がフリーライダーとなる状態がナッシュ均衡となることを示している. そこで, ファイル共有によりノードが報酬, もしくは他のノードからの評価を得られるといった仕組みを導入することで, ファイルを共有するノードの割合を増加させられることを, 理論的解析とシミュレーションにより明らかにしている.

これらの文献では, ノードの利己的な振る舞いがファイル共有システムのナッシュ均衡における性能に与える影響を明らかにし, 協力的なキャッシングのためには他ノードからの報酬などのインセンティブが必要であるとしている. しかしながら, システムにインセンティブを導入することが難しい場合がある. 例えば, 匿名性を高めるためにファイルの所有者がわからないようになっているシステムでは, ファイルの取得者から所有者に対して報酬を与えることは難しい. そこで本稿では, こうしたインセンティブを導入することなく, より一般的なファイル共有システムにおいて, ノードがキャッシングに対する自分の需要やコストのみを考慮し, 自律的かつ利己的に振る舞った結果, システム全体で適切なキャッシングが行われる機構の実現を目指す. そのため, 特に進化ゲーム理論を用いて, 利己的なノードの局所的なやりとりがシステム全体としてのダイナミクスに与える影響について検証する [6-8] .

進化ゲーム理論は, 生物社会において様々な個体が相互に影響を与え合う環境の下, 生物の進化の過程で最適な行動が受け継がれるという考え方を, ゲーム理論を用いて説明しようとするものである. 生物に遺伝的にプログラムされた行動をゲーム理論における個体の取る戦略, また, ある行動を取る生物の子孫の数をそれぞれの戦略がもたらす利得とみなすことにより, 生物はゲーム理論における戦略的な相互依存関係にあるといえる. したがって, ゲーム理論により, 生物社会において優れた行動が子孫に受け継がれ, 広がるという現象を説明することができる. 例えば, 文献 [6, 9, 10] では, 進化ゲーム理論により, 利己的・非協力的に思われる個人の振る舞いが社会全体としての協力的行動の表れにどのような影響を与えるかを解明しようとしている. ファイル共有システムにおいても同様に, 局所的な情報に基づくノードの利己的な振る舞いがシステム全

表 1 payoff matrix (general)

自分の戦略	相手の戦略	
	協力	裏切り
協力	(R, R)	(S, T)
裏切り	(T, S)	(P, P)

体のキャッシング状態に与える影響を明らかにできるものと考えられる. そこで本稿では, 特にノードの振る舞いと安定状態におけるシステム全体でのキャッシュファイル数の関係を, 進化ゲーム理論におけるレプリケータダイナミクスとエージェントベースダイナミクスを用いて明らかにし, 利己的なノードの振る舞いによってシステム全体として適切なキャッシングが行われる仕組みについて検討する. レプリケータダイナミクスとエージェントベースダイナミクスは進化的に安定な状態における戦略分布を知るための手法である.

レプリケータダイナミクスは, 社会を構成する個体数が多く, かつ個体間の繋がりを表すネットワークが平均場としてみなせる場合に, 社会の平均よりも多くの利得を残すことのできる戦略が増えていく現象を数式としてモデル化したものである. これにより, メッシュネットワーク, 正則ネットワークにおいて, ノードがシステム全体のキャッシングの状態に基づいて利己的に振る舞う場合の特性を知ることができる. 一方, エージェントベースダイナミクスは, 優れた戦略が隣接する個体を通してネットワーク内に広まるという現象をモデル化したものである. エージェントベースダイナミクスでは, ある個体は, 隣接する個体とゲームを行い, その対戦結果から戦略の優劣を判断し, 次に自分の取る戦略を決定する. これにより, 様々なネットワークトポロジにおいて隣接ノードとのみゲームを行う場合の特性を知ることができる.

本稿では, まず, キャッシングに対するコストと需要に基づいてファイル共有システムをゲームとしてモデル化する. モデル化したゲームに対して, ノードが大域的な情報を得られるという理想的な環境における, システム全体のキャッシュファイル数を, レプリケータダイナミクスを用いて理論的に明らかにし, モデルの基本的な特性を評価する. さらに, エージェントベースダイナミクスに基づくシミュレーションにより, より現実的な論理ネットワークにおいてノードが局所的な情報に基づいて振る舞う場合の, システム内のキャッシュファイル数, 及びノードからファイルの所有者までの距離を評価する. 評価結果から, ファイル可用性, 検索遅延の観点で適切なキャッシング実現に適したキャッシングのモデルについて述べる.

以下, 2章で本稿で扱うキャッシングゲームのモデルについて述べる. 3章ではレプリケータダイナミクスを, 4章ではエージェントベースダイナミクスを用いて, モデルに基づくノードの振る舞いとキャッシング状態との関係について検証する. 最後に, 本稿のまとめと今後の課題について 5章で述べる.

2. キャッシングゲーム

進化ゲーム理論では, 2者間において, それぞれがある戦略を選択した際に得られる利得を, 表 1 に示すような利得行列で与え, ゲームをモデル化する. 例えば, 右上の (S, T) は自分が「協力」, 相手が「裏切り」の戦略をとったとき, それぞれ S, T の利得を得ることを示す. 一般に良く知られるゲームとしては, 表 1 におけるパラメータが $T > R > P > S$ となる囚人のジレンマゲーム, $T > R > S > P$ となる雪の吹き溜まりゲームが挙げられる.

本稿では, ファイル共有システムにおけるキャッシングを隣接する 2 ノード間でのキャッシングゲームとしてモデル化する. ノードのストレージ容量に制限がない場合には, 異なるファ

表 2 payoff matrix (caching, cost: load)

自分の戦略 \ 相手の戦略	キャッシュする	キャッシュしない
キャッシュする	$(b - c_l, b - c_l)$	$(b - 2c_l, b)$
キャッシュしない	$(b, b - 2c_l)$	$(0, 0)$

表 3 payoff matrix (caching, cost: storage)

自分の戦略 \ 相手の戦略	キャッシュする	キャッシュしない
キャッシュする	$(b - c_s, b - c_s)$	$(b - c_s, b)$
キャッシュしない	$(b, b - c_s)$	$(0, 0)$

イル間でのストレージ容量の競合を考慮する必要がないため、キャッシングゲームは単一ファイルを扱うゲームが複数集まって構成されているとみなすことができる。そこで、以降では簡単化のため単一ファイルの場合のキャッシングゲームについてモデル化し、利得行列を定める。

あるファイルに対して、ノードの取りうる戦略はファイルをキャッシュする、キャッシュしないのどちらかとなる。ノードはファイルを利用することによってそのファイルに対する自分の需要を満足できることから、ファイルの利用によって得られる利益を需要 b とする。一方、キャッシングにおけるノードにとってのコストは、自分と他のノードからのファイルへのアクセスによる処理負荷 c_l 、ファイルをキャッシュすることで必要となるストレージ容量 c_s によって定められるものとする。なお、ファイル取得の際にかかるネットワーク帯域、遅延といった動的に変化するコストを導入した場合の特性評価については今後の課題とする。また、一般的なファイル共有システムにおけるキャッシングを対象としていることから、本稿では、キャッシュに対するインセンティブメカニズム [4, 5] は考慮しない。

コストとして処理負荷のみを用いた場合の利得は、 $R = b - c_l$ 、 $T = b$ 、 $S = b - 2c_l$ 、 $P = 0$ となる (表 2)。ただし、 $b - 2c_l \geq 0$ である。表 2 は $T > R > S > P$ となり、パラメータの関係は雪の吹き溜まりゲームの場合と等しい。コストとしてストレージ容量のみを用いた場合には、 $R = b - c_l$ 、 $T = b$ 、 $S = b - c_l$ 、 $P = 0$ となる (表 3)。ただし、 $b - c_s \geq 0$ である。表 3 は $T > R = S > P$ となり既存のゲームには一致するものが存在しない。

以降の章では、上記の利得行列を対象に、パラメータとキャッシュファイル数、すなわちキャッシュするという戦略を取るノード数の関係を明らかにし、ファイル可用性の高いファイル共有の実現について検討する。

3. レプリケータダイナミクスによる分析

本章では、表 2、表 3 のパラメータ設定と、ファイル共有システムの安定状態においてファイルをキャッシュするという戦略を選ぶノード数との関係を、レプリケータダイナミクス [6-8] を用いて理論的に明らかにする。より多くのノードがキャッシュするという戦略をとるような利得行列を与えることができれば、利得行列に基づき利己的に振る舞うノードからなるファイル共有システムにおけるファイル可用性を高められるものと考えられる。

3.1 レプリケータダイナミクス

レプリケータダイナミクスでは、ある戦略を取るノードの増加数はその戦略を取ることで得られる利得に比例する。したがって、システム全体の平均利得よりも多くの利得をもたらす戦略を選ぶノード数はシステム内で増加し、そうでなければ減少する。ただし、ゲームの対戦相手は自分以外のすべてのノード、もしくは全ノードから均一に選ばれた複数ノードであることを前提としている。

まず、表 1 の一般的な利得行列で表されるゲームにおけるレ

プリケータダイナミクスの導出過程について説明する。ある時刻において協力戦略を選ぶノードの割合を x とすると、裏切り戦略を選ぶノードの割合は $1 - x$ であり、協力戦略を選んだノードが得る平均利得は、

$$Rx + S(1 - x) \quad (1)$$

となる。一方、裏切り戦略を選んだあるノードが得る平均利得は、

$$Tx + P(1 - x) \quad (2)$$

となる。したがって、ノード全体としての平均利得は、

$$x\{Rx + S(1 - x)\} + (1 - x)\{Tx + P(1 - x)\} \quad (3)$$

となる。協力戦略を選んだノードの平均利得の全体の平均利得との差は次式で表される。

$$\begin{aligned} & \{Rx + S(1 - x)\} \\ & - \{x(Rx + S(1 - x)) + (1 - x)(Tx + P(1 - x))\} \\ & = \{(R + P - T - S)x + S - P\}(1 - x) \end{aligned} \quad (4)$$

したがって、 x の変化を表すレプリケータダイナミクス \dot{x} は次式で表される。

$$\dot{x} = \{(R + P - T - S)x + S - P\}(1 - x)x \quad (5)$$

3.2 システムの安定状態におけるキャッシュファイル数
システムが均衡状態となるのは $\dot{x} = 0$ となる x の値である。すなわち、 $x = 0, 1, \frac{P - S}{R + P - T - S}$ が均衡点となる。

次に各均衡点の安定性について述べる。式 (5) より、 $0 \leq x \leq 1$ のとき、 $(1 - x)x \geq 0$ である。一方、 $(R + P - T - S)x + S - P$ は、2 章でのゲームの定義より、 $x < \frac{P - S}{R + P - T - S}$ では正、 $x > \frac{P - S}{R + P - T - S}$ であれば負となる。したがって、 x は初期値によらず $\frac{P - S}{R + P - T - S}$ に近づくため、安定した均衡点は以下の場合のみである。

$$x = \frac{P - S}{R + P - T - S} \quad (6)$$

以上のことから、システムの安定状態における協力戦略をとるノードの割合 x は利得行列のパラメータに依存することが分かる。ここで、自分と相手が共に協力戦略をとる場合に生じる利得に対するコストの比を r と定義する [9-11]。 r は協力戦略をとる場合のリスクを表しており、 $(0, 1]$ の値となる。 r が 0 に近いほどノードは協力戦略を取りやすいことを意味する。本稿では、 r は b と c_l 、もしくは b と c_s によって決定されるため、 r はキャッシングの需要に対するコストとみなせる。

まず、コストが処理負荷のみの場合には、式 (6) と表 2 より、安定状態における協力戦略を選ぶノードの割合 x は次式で表される。

$$x = \frac{b - 2c_l}{b - c_l} \quad (7)$$

一方、両者が協力戦略をとる場合の利得に対するコストの比 r は次式となる。

$$r = \frac{c_l}{b - c_l} \quad (8)$$

したがって、 x と r の間には以下のような関係が成り立つ。

$$x = 1 - r \quad (9)$$

コストがストレージ容量のみの場合には、式 (6) と表 3 より、安定状態における協力戦略を選ぶノードの割合 x は次式で表される。

$$x = \frac{b - c_s}{b} \quad (10)$$

一方、両者が協力戦略をとる場合の利得に対するコストの比 r は次式となる。

$$r = \frac{c_s}{b - c_s} \quad (11)$$

したがって、 x と r の間には以下のような関係が成り立つ。

$$x = \frac{1}{1 + r} \quad (12)$$

式 (9), (12) より、いずれのモデルにおいても r の増加、すなわちファイルに対する需要 b の減少にともない、キャッシュするという戦略を選ぶノードの割合 x も低下することがわかる。本稿では、特定のファイルが消失することのないファイル可用性の高いファイル共有システムの実現を目指しているが、コストを処理負荷とすると、式 (9) より r が 1 に近く、ファイルに対する需要が低い場合には、ファイルがシステムから消失することがわかる。一方、コストをストレージ容量とした場合には、 r が 1 の場合でもキャッシュファイル数が 0 になることはなく、ファイル可用性が高いといえる。

4. エージェントベースダイナミクスによる分析

エージェントベースダイナミクス [6, 9-11] を用いることで、ノード間の局所的なやり取りに基づくキャッシングがシステム全体におけるキャッシュファイル数に与える影響を明らかにできる。さらに、それぞれのノードの選ぶ戦略、すなわちキャッシュファイルの位置がわかるため、ファイルの検索遅延についても評価が可能となる。

4.1 エージェントベースダイナミクス

エージェントベースダイナミクスにおいては、ノードは隣接ノードと利得を比較し、戦略を決定する。初期状態ではノードはある確率でランダムに戦略を割り当てられる。従来研究では、協力と裏切りの戦略をそれぞれ 50%ずつとしている [9-11]。戦略が決定されると、あるノード i は隣接するすべてのノードとゲームを 1 回ずつ行う。これを 1 世代とする。隣接するすべてのノードとのゲームを終えると、得られた利得の総和 S_i もしくは平均 A_i を基に、ノード i は次の世代の戦略を以下の手順によって決定する。

(1) 利得の比較相手の選出

隣接ノードからランダムに一つのノード j を選択する。

(2) 利得の比較に基づく次の戦略の決定

- 利得の総和を用いる場合

$S_j > S_i$ である場合には、

$$P_S(i, j) = \frac{S_j - S_i}{(T - P) \max(k_i, k_j)} \quad (13)$$

の確率で j の戦略を採用し、そうでなければ自分の戦略を採用する。ただし、 k_i, k_j は i, j の隣接ノード数を表す。

- 利得の平均を用いる場合

$A_j > A_i$ である場合には、

$$P_A(i, j) = \frac{A_j - A_i}{T - P} \quad (14)$$

の確率で i は j の戦略を採用し、そうでなければ自分の戦略を採用する。

いずれも、自分より多くの利得を得たノードの戦略を真似しようとすることを意味しており、利得の差が大きいほど戦略を模倣する確率は増加する。

フルメッシュネットワークや正則ネットワークではノードごとの隣接ノード数はすべて等しいため、利得の総和と平均のどちらを用いても結果は同じとなる。しかしながら、スケールフリーネットワークやランダムネットワークでは、ノードごとの隣接ノード数にばらつきがあるため、結果が異なる。例えば、文献 [9, 10] ではスケールフリーネットワーク上で囚人のジレンマゲーム、雪の吹き溜まりゲームを行う場合には、利得の総和を用いた方がノードが協力戦略を取る傾向が高くなることを示している。利得の総和は隣接ノード数に依存するため、次数の高いノードはゲームの対戦回数から次数の低いノードよりも多くの利得を得る傾向がある。その場合、次数の低いノードが次数の高いノードの戦略を模倣したとしても、元の戦略に比べてより多くの利得を得られるとは限らない。したがって、隣接ノード数よりも戦略の優劣の方が戦略の選択には重要であると考えられ、本稿では、ノードは利得の平均を比較することによって戦略を決定するものとする。

なお、エージェントベースダイナミクスでは安定状態を定めることは困難であるため、十分な世代を繰り返した時点の状態について評価を行う [9-11]。

4.2 シミュレーション評価

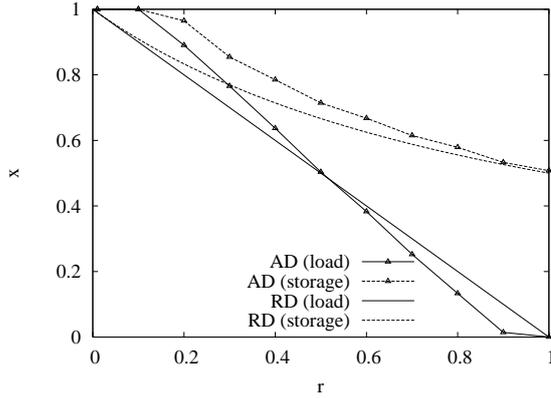
スケールフリーネットワークとランダムネットワークにおいて、キャッシングゲームにおけるそれぞれの利得行列に基づくノードの振る舞いが、システム全体のファイル可用性、検索遅延に与える影響をシミュレーション評価を通して検証する。ファイル可用性は、キャッシュするという戦略を選ぶノードの割合 x で評価する。あるファイルに着目した場合、ノード数と x の積がキャッシュファイル数となる。また、ノードから自分も含めたファイルの所有者までの平均最小ホップ数を検索遅延とする。ただし、ファイルがシステムから消失している場合には、任意の 2 ノード間の最大ホップ数を検索遅延とする。

4.2.1 シミュレーションモデル

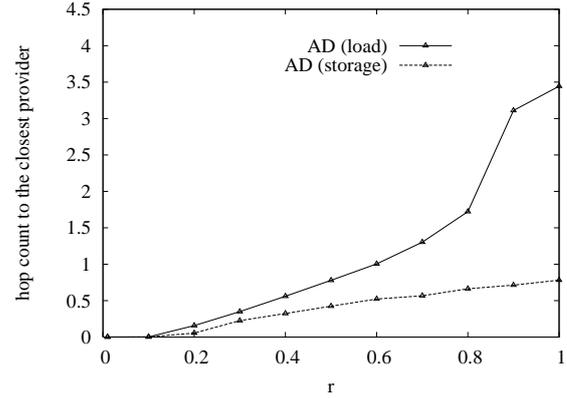
シミュレーションには NetLogo [12] を用い、シミュレーション条件については文献 [9-11] を参考に以下のように設定した。ノード数 1,000 の Barabási-Albert (BA) モデル [13] に基づくスケールフリーネットワークと Waxman アルゴリズム [14] に従うランダムネットワーク ($\alpha = 0.15, \beta = 0.2$) をトポロジ生成ツール BRITe [15] を用いて作成した。なお、新規参加ノードの接続数 m は 2, 4 とした。それぞれのネットワークトポロジについて、任意の 2 ノード間の最大ホップ数の 10 個のネットワークにおける平均を表 4 に示す。ファイルをキャッシュするコストは、 c_l, c_s とともに 1 と設定した。したがって、式 (8), (11) からコストの種類によらず $b = \frac{1+r}{r}$ となる。シミュレーション開始時のノードの戦略分布はキャッシュする、しないともに 50%ずつとなるようにランダムに設定した。システムの安定状態における特性を調べるため、シミュレーション開始後 10,000 世代までを過渡状態とみなし、その後 1,000 世代の平均値を用いる。なお、以降の結果ではそれぞれ 10 回のシミュレーションの平均値を示す。図中の RD, AD はそれぞれレプリケータダイナミクス、エージェントベースダイナミクスを用いた場合の結果を表す。

4.2.2 利得行列とファイル可用性・検索遅延の関係

図 1(a) は、 $m = 4$ のスケールフリーネットワークにおいて



(a) fraction of cooperators



(b) hop count to the closest provider

図 1 payoff matrix vs. file availability and search latency (scale-free: $m = 4$)

表 4 maximum hop count between two arbitrary nodes

	$m = 2$	$m = 4$
scale-free network	7.4	5
random network	8.7	6

エージェントベースダイナミクスを用いた場合の、 r に対するキャッシュするという戦略を取るノードの割合 x を示している。コストの種類によらず、 r の増加、すなわち需要の減少とともに x も低下することがわかる。また、処理負荷をコストとした場合には、 r の大きいファイルが消失しやすくなっている。一方、ストレージ容量をコストとした場合には、 r が 1 の場合でもキャッシュファイル数が 0 になることはなく、ファイルの可用性が高いと言える。また、図 1(a) には、レプリケータダイナミクスにより導出された x もあわせて示しているが、処理負荷をコストとした場合には、需要の高いファイルはより多くキャッシュされるのに対し、需要の低いファイルは消失しやすくなっていることがわかる。一方、処理負荷をコストとした場合には、 r によらず、システム全体の情報ではなく局所的な情報のみを基にノードが振る舞う方がファイル可用性が高くなることからわかる。

図 1(b) はファイルの所有者までの最小ホップ数の平均を示している。図より、コストの種類によらず r の増加とともにファイルの所有者までの最小ホップ数も増加していることがわかる。これは図 1(a) で示されるようにファイルの所有者の減少によるものである。ストレージ容量をコストとした場合は、図 1(a) より、需要の低いファイルでも所有者が全体の 5 割を超えるため、処理負荷をコストとする場合に比べてより近くにファイルの所有者を見つけることができ、検索遅延が小さい。

4.2.3 ネットワーク構造の影響

図 2 はスケールフリーネットワークにおける m を 2, 4 とした場合のキャッシュするという戦略を取るノードの割合 x 、及びファイルの所有者までの最小ホップ数を表している。なお、ノードの平均次数は $2m$ となるため、 m が大きい方がネットワークはより密となる。図 2(a) より、コストの種類によらず、 m が小さいと需要の高いファイルがより多くキャッシュされる傾向が見られる。これは、高次数ノードの影響によるものである。高次数ノードは他のノードに比べてより多くの利得を獲得する傾向があり、さらに、隣接ノードから利得の比較相手として選ばれる可能性も高い。そのため、高次数ノードの選ぶ戦略はネットワーク内に広がりやすい。 m が小さいと低次数ノードが多くなるため、高次数ノードの影響がより大きくなる。さらに、処理負荷をコストとした場合には、 m が小さいと需要の低

いファイルがより消失しやすくなっていることがわかる。

一方、図 2(b) より、処理負荷をコストとした場合について、 r が 0.4、ストレージ容量をコストとした場合には 0.7 より低い場合には、 $m = 2$ の方がファイルの所有者までの最小ホップ数が小さくなっていることが分かる。これは図 2(a) で示されるように $m = 2$ の場合には r の小さいファイルが多くキャッシュされているためである。 r がさらに増加するとファイルの所有者が減るため、 $m = 4$ の場合の方が優れた結果となる。ただし、ストレージ容量をコストとした場合には m の違いによる影響は小さい。

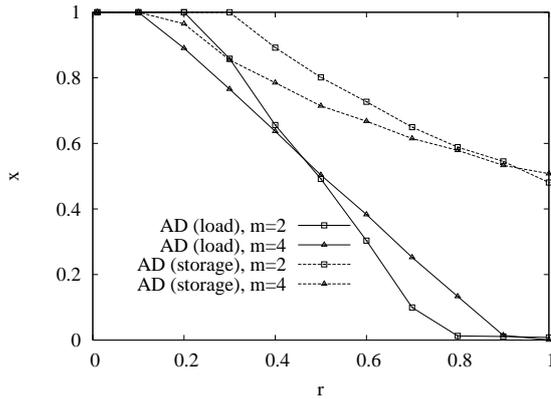
以上のことから、コストにストレージ容量を用いることにより、ノードの平均次数にあまり依存せず、ファイル可用性が高く、検索遅延の小さいファイル共有が実現できると言える。

図 3 はスケールフリーネットワークとランダムネットワークにおける m を 4 とした場合の、キャッシュするという戦略を取るノードの割合 x 、ファイルの所有者までの最小ホップ数を表している。図 3(a) より、コストの種類によらずスケールフリーネットワークの方がランダムネットワークより x が小さくなることから分かる。そのため、表 4 に示すようにスケールフリーネットワークの方がランダムネットワークよりノード間距離が近いにもかかわらず、図 3(b) に示されるように、ランダムネットワークよりもファイル所有者までの最小ホップ数が大きくなる。ただし、ストレージ容量をコストとした場合にはトポロジ構造の違いが与える影響は小さい。

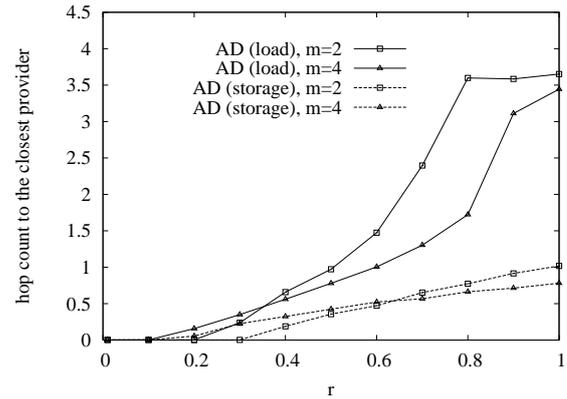
以上のことから、コストをストレージ容量とした場合には、スケールフリーネットワーク、ランダムネットワークいずれにおいても、ファイル可用性が高く、検索遅延の小さいファイル共有が実現できると言える。

5. おわりに

本稿では、ノードが利己的かつ自律的に振る舞ったとしても、システム全体で適切なキャッシングが可能でファイル共有システムを実現するため、進化ゲーム理論を用いてノードの振る舞いとキャッシングとの関係について明らかにした。まず、2 ノード間でのキャッシングゲームとしてファイル共有システムをモデル化し、レプリケータダイナミクスによりシステム内のファイル数を解析的に導出することで、モデルの基本的な特性を明らかにした。さらに、ノードの局所的な振る舞いがシステム性能に与える影響を、エージェントベースダイナミクスに基づくシミュレーションにより評価した。シミュレーション評価により、キャッシングに関するコストとしてストレージ容量を用いた

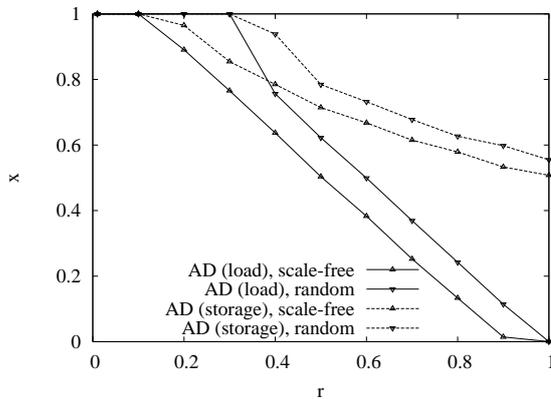


(a) fraction of cooperators

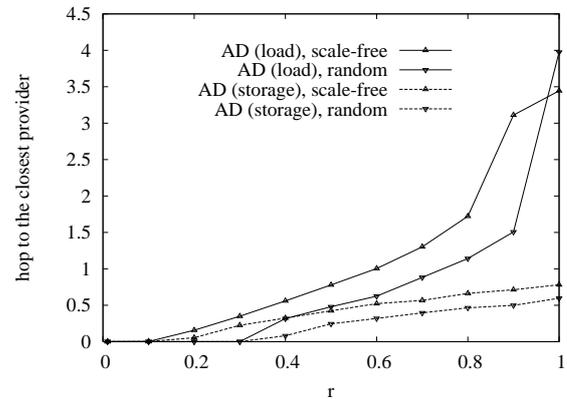


(b) hop count to the closest provider

図 2 impact of network density (scale-free: $m = 2, 4$)



(a) fraction of cooperators



(b) hop count to the closest provider

図 3 impact of node degree distribution (scale-free vs. random)

場合には、ネットワーク構造によらず、ノードが利己的に振る舞ったとしてもファイルがシステムから消失することのない、ファイル共有が実現可能であることを示した。

今後は、動的に変化するコストを含め、複数種類のコストを考慮することで、より適切なキャッシングが可能なキャッシングゲームについて検討する。また、ファイル共有システム以外の情報ネットワークにおけるダイナミクスの分析と制御手法の検討にも取り組みたい。例えば、アプリケーションレベルマルチキャストなどオーバーレイネットワークを利用した情報配信やセンサネットワークにおける情報拡散・収集においては、情報は中継ノードを介して送信元から受信先へと転送される。情報の転送には帯域、電力といったコストが発生することから、情報の重要性などを考慮した適切な情報転送の仕組みが必要となる。進化ゲーム理論を用いることで、適切な情報転送に必要なノードの振る舞いを明らかにすることができる。

謝辞 本研究の一部は、科学研究費補助金若手研究 (B) No.17700058 及び文部科学省 21 世紀 COE プログラム (研究拠点形成費補助金) によるものであり、ここに謝意を表す。

文 献

[1] K. P. Gummadi, R. J. Dunn, S. Saroiu, S. D. Gribble, H. M. Levy and J. Zahorjan: "Measurement, Modeling, and Analysis of a Peer-to-Peer File-Sharing Workload", SOSP 2003, New York, pp. 314-329 (2003).
 [2] A. Rowstron and P. Druschel: "Storage Management and Caching in PAST, a Large-Scale, Persistent Peer-to-Peer Storage Utility", SOSP 2001, New York, pp. 188-201 (2001).
 [3] E. Adar and B. A. Huberman: "Free Riding on Gnutella",

Technical Report Xerox PARC (2000).

[4] B.-G. Chun, K. Chaudhuri, H. Wee, M. Barreno, C. H. Papadimitriou and J. Kubiatowicz: "Selfish Caching in Distributed Systems: A Game-Theoretic Analysis", Proceedings of PODC 2004, Aveiro, Portugal, pp. 21-30 (2004).
 [5] K. Ranganathan, M. Ripeanu, A. Sarin and I. Foster: "Incentive Mechanisms for Large Collaborative Resource Sharing", Proceedings of CCGRID 2004, Washington, DC, USA, pp. 1-8 (2004).
 [6] S. Gyorgy and F. Gabor: "Evolutionary Games on Graphs", ArXiv Condensed Matter e-prints, cond-mat/0607344 (2006).
 [7] 生天目: "ゲーム理論と進化ダイナミクス", 森北出版 (2004).
 [8] 佐伯, 亀田: "進化ゲームとその展開", 共立出版 (2003).
 [9] J. M. Pacheco and F. C. Santos: "Network Dependence of the Dilemmas of Cooperation", Proceedings of CNET 2004, Aveiro, Portugal, pp. 90-100 (2004).
 [10] F. C. Santos and J. M. Pacheco: "A New Route to the Evolution of Cooperation", Journal of Evolutionary Biology, **19**, 3, pp. 726-733 (2006).
 [11] C. Hauert and M. Doebeli: "Spatial Structure Often Inhibits The Evolution of Cooperation in The Snowdrift Game", Nature, **428**, (2004).
 [12] "NetLogo". available at <http://ccl.northwestern.edu/netlogo/>.
 [13] A.-L. Barabási and R. Albert: "Emergence of Scaling in Random Networks", Science, **286**, (1999).
 [14] B. M. Waxman: "Routing of Multipoint Connections", IEEE Journal on Selected Areas in Communications, **6**, 9, pp. 1617-1622 (1988).
 [15] "BRITE". available at <http://www.cs.bu.edu/brite/>.