

TR-AC-0014

002

Agentに基づくルーティングアルゴリズム

種田 和正 片岡 明 (実習生)

1998. 3. 2

ATR環境適応通信研究所

## 目次

1	はじめに-----	1
2	<i>AntNet</i> の振る舞い-----	1
3	Lock-freeアルゴリズム-----	1
4	ネットワーク故障に対する適応性-----	2
5	考察-----	3
6	おわりに-----	3
	参考文献-----	3
	図-----	4

## 1 はじめに

本実務訓練では、適応的なルーチングアルゴリズムの一つである *AntNet*<sup>[1]</sup> の振る舞いをシミュレーションにより評価する。*AntNet* が高学習率時でもトラヒック振動をおこさないように、ルーチング確率の Lock を解除する Lock-free アルゴリズムを考案した。また、それらの考案したアルゴリズムのネットワークの故障に対する適応性を検証した。

## 2 *AntNet* の振る舞い

### 2.1 学習率 $h$ の影響

*AntNet* における学習率  $h$  の影響を見るために、学習率  $h = 0.04, 0.1, 0.4$  としたときのシミュレーション結果を図 2-1 から図 2-3 に示す。

シミュレーション結果として、ネットワーク内パケット数 ( $N_p$ ) のダイナミクスから、トラヒックの安定状態を評価する。

トラヒック強度  $I$  をシミュレーション時間  $\text{time}=0-2000$  に  $I=0$ 、 $\text{time}=2000-3000$  に  $I=0.3$  とした。その他のシミュレーション条件を以下に示す。

- ネットワーク：図 1-1。すべてのリンクの伝送時間は単位パケット長当たり 1.0 simulation time、ノード処理時間は 0.0 とする。
- ルーチングテーブルの初期条件：ランダムな値
- パケットサイズ：平均値が  $1/\mu$  ( $\mu = 1$ ) の負の指数分布
- ソースと宛先ノードの分布：均一分布
- ノード当たりのパケット発生間隔：平均値が  $1/\lambda$  の負の指数分布
- トラヒックの強度  $I = \lambda/\mu$
- エージェントサイズ：0.1
- エージェントの発生間隔：50
- *AntNet* アルゴリズムの定数 ( $c = 2.0, a = 10.0, a' = 9.0, \epsilon = 0.25, t = 0.5$ )
- エージェントの探求確率  $P_e = 0.002$

$h = 0.04$  と低い学習率では  $I=0 \rightarrow 0.3$  の変化とともに  $N_p$  が大きく立ち上がるが、 $h = 0.4$  では立ち上がりが小さい。一般的に、学習率が高いほど収束は早くなる。

しかし、オリジナルの *AntNet* では、学習率  $h = 0.04$  と低く設定されている。これは、学習率  $h$  が高いと、安定状態への収束は早いですが、急激なトラヒック強度の増加時に振動がおきやすくなるからだと考えられる。

### 2.2 振動をおこすトラヒック

学習率  $h = 0.4$  と高く設定した *AntNet* が振動をおこす事例として、トラヒック強度を  $I=0.2 \rightarrow 0.5$  に変化させたシミュレーションを行った。

$h = 0.4$ 、時刻 0 にトラヒック強度  $I=0.2$  とし、時刻  $t_a=2000$  に  $I=0.5$  と変化させた結果を図 3-1 に示す。また、 $t_b=3000$  で  $I=0.5$  とした結果を図 3-2 に、時刻  $t_c = 5000$  で  $I=0.5$  とした結果を図 3 に示す。

### 2.3 振動と Lock の関係

図 3-2 はトラヒック強度の変化後、振動をおこしている。図 3-1 は、わずかに振動しているが振幅も小さく、収束も早い。図 3-3 は図 3-2 よりさらに大きく振動している。

これらの振動の原因としてルーチング確率の Lock があげられる。Lock とは、ルーチング確率が更新され続けた結果、1 に非常に近づき、1 から離れない現象をいう。ルーチング確率が Lock されると、パケットもエージェントもある特定の経路のみを選択することになり、ルーチングが固定され静的なルーチングアルゴリズムと同じ性能しかあげられない。Lock された経路がトラヒックの変化に対応しないので、待ちキュー長が急増し、それが振動を引き起こす原因となる。

図 4-1 に  $I=0.2$ 、 $h = 0.4$  での Lock 率の変化を示す。

$$\text{Lock率} = \frac{0.99 \text{ を超えるルーチング確率の数}}{\text{すべてのソースと宛先の組み合わせの数}} \quad (1)$$

先のシミュレーション結果と対応させると、時刻  $t_a, t_b, t_c$  の順で Lock 率が高くなっており、Lock 率が高いときにトラヒック強度を変化させると大きく振動していることが分かる。

図 4-2 に、 $h = 0.04, 0.4, I = 0.2, 0.4$  のときの Lock 率の推移を示す。 $h$  が高いほど、また  $I$  が高いほど早く Lock 率は上がる。

## 3 Lock-free アルゴリズム

学習率  $h$  が高いと Lock 率が高くなり、振動の原因となる。そこで、Lock 率を引き下げる 2 つの Lock-free アルゴリズムを考案した。

### 3.1 DCY(DeCaY)

ルーチング確率  $P_{d,n}^s(i)$  の  $\Delta d$  後の確率  $P_{d,n}^s(i+1)$  は次の式により計算する。 $(P_{d,n}^s)$ : 宛先がノード  $d$  のとき、ノード  $s$  からノード  $n$  へルーチングする確率)

$$P_{d,n}^s(i+1) = P_{d,n}^s(i) - (P_{d,n}^s(i) - \frac{1}{N})v \quad (2)$$

ただし、

$N$ : 隣接ノード数

$v$ : 減衰率

$$v = 1 - (1 - \frac{1-P_0}{P_0}(1 - 0.5^h))^{1/d}$$

減衰間隔  $\Delta d = (\text{エージェント送信間隔}) \div d$

初期値が  $P_0$  の確率が  $d$  回減衰されたのちに、エージェント送信間隔時間で戻ったエージェントに更新されると、ルーチング確率  $P_{d,n}^s$  が  $P_0$  でつり合うように  $v$  の式を設定した。 ( $h = 0.4, P_0 = 0.9, d = 10$  のとき、  $v = 0.0025$ )

$Trip_k(\mu_i, \sigma_i)$  はウィンドウサイズ 30 でエイジングを行う。

トラヒックが安定しているときは確率は初期値  $P_0$  でつり合い、トラヒックが混雑しエージェントが戻らないと、すべての確率は次第に  $\frac{1}{N}$  に近づいていく。

## 3.2 NFB(Negative FeedBack)

トリップ時間の長い経路に対して、負のフィードバックを行う。

ノード  $n$  からエージェントが到着したとすると、  $P_{d,n}^s(i) > 0.9$  かつ  $r' \geq 1$  かつ  $\frac{d|V_s(\bar{\mu}, \bar{\sigma})|}{dt} > \delta$  のとき、以下の式により次のルーチング確率  $P_{d,n}^s(i+1)$  を計算する。

$$P_{d,n}^s(i+1) = P_{d,n}^s(i) - (P_{d,n}^s(i) - \frac{1}{N})v' \quad (3)$$

ただし、

$N$ : 隣接ノード数

$\bar{\mu}, \bar{\sigma}$ : ノード  $s$  に最近到着した 30 個のエージェントのトリップ時間の平均 ( $\bar{\mu}$ ) と分散 ( $\bar{\sigma}^2$ )

$|V_s|$ :  $\bar{\mu}, \bar{\sigma}$  空間のベクトル ( $\bar{\mu}, \bar{\sigma}$ ) の長さ:  $\sqrt{\bar{\mu}^2 + \bar{\sigma}^2}$

負の強化率  $v' = 0.2$

スレッシュホールド  $\delta = 2$

ルーチング確率が Lock されている状態かつトラヒックが不安定な状態で、平均の 2 倍以上の遅延時間でエージェントが戻った場合、その Lock を解除する。

## 3.3 トラヒック強度の変化に対する適応性

2.3 節で示したトラヒック振動モデル ( $t_b = 3000$ ) を用いて各アルゴリズムの適応性を評価する。

DCY, NFB の 2 つの Lock-free アルゴリズムと、オリジナル *AntNet* の探求確率 ( $P_e$ ) を 0.3 に設定した ORG ( $P_e = 0.3$ ) に対して、図 3-2 と同様のシミュレーションを行った。

その結果、DCY, NFB, ORG ( $P_e = 0.3$ ) すべて振動を押さええることに成功した。

### 3.3.1 DCY

DCY に対してシミュレーションを行った結果を図 5-1 に示す。また、Lock 率を図 5-2 に、ルーチング確率  $P_{3,14}^s$  を図 5-3 に示す。

DCY では、定期的な高い確率は減衰され、相対的に低い確率は増加されるため、Lock は起こらない。図 5-3 では強度が変化した直後に確率が大きく変動し、ルーチングが移り変わっている様子が分かる。

### 3.3.2 NFB

NFB に対してシミュレーションを行った結果を図 5-4 に示す。また、Lock 率を図 5-5 に、  $\frac{d|V_{10}(\bar{\mu}, \bar{\sigma})|}{dt}$  を図 5-6 に示す。

Lock 率は 1 に近づいているが、強度の変化を  $\frac{d|V_s(\bar{\mu}, \bar{\sigma})|}{dt}$  により検出し、Lock 率を下げていることが分かる。

### 3.3.3 ORG ( $P_e = 0.3$ )

ORG ( $P_e = 0.3$ ) に対してシミュレーションを行った結果を図 5-7 に示す。また、Lock 率を図 5-8 に示す。

強度が変化した直後、Lock 率が下がり、トラヒックが安定後も Lock 率は 0.4 前後と安定し、1 に漸近していかない。

### 3.3.4 DCY ( $P_e = 0.3$ ), NFB ( $P_e = 0.3$ )

DCY, NFB の探求確率 ( $P_e$ ) を 0.3 と設定し、シミュレーションを行った結果を図 5-9 から図 5-12 に示す。

DCY, NFB ともに  $P_e = 0.3$  とすることで、トラヒック強度変化時の  $N_p$  の立ち上がり小さくなっている。

## 4 ネットワーク故障に対する適応性

図 1-1 のネットワークにリンクダウン、ノードダウン、ノードアップを起こし、ORG, DCY, NFB について適応性を検討する。リンクダウンとは、以下の処理を行う。

- ダウンしたリンクに接続されているノードのルーチングテーブルからそのリンクを使用するエントリを削除する。その他のリンクの確率の合計を 1 に正規化する。
- そのリンクを通ろうとするバックワードエージェントを廃棄する。

ノードダウンとは、以下の処理を行う。

- ダウンしたノードに接続するリンクはすべてダウン処理する。
- ダウンしたノードからはパケット、エージェントを発生しない。
- 宛先がダウンしたノードであるパケット、エージェントはダウンしたノードの隣接ノードが検知し廃棄する。ダウンしたノードに隣接しているノードのみの処理であるため、ノードダウンを通知するためのプロトコルは必要としない

ノードアップとは、以下の処理を行う。

- ノードダウン時に削除されたリンクのエントリをルーチングテーブルに追加する。
- アップ後、そのノードからパケット、エージェントを発生させる。

## 4.1 リンクダウン

図 6-1 から図 6-6 に ORG ( $P_e = 0.002$ ), ORG ( $P_e = 0.3$ ), DCY, NFB, DCY ( $P_e = 0.3$ ), NFB ( $P_e = 0.3$ ) について次の条件でシミュレーションを行った結果を示す。

- time=2000 から  $I=0.5$
- time=12000 にリンク 2-7 をダウン

また、収束時の  $N_p$  の平均値を表 1 に示す。

表 1: 収束時の  $N_p$  の平均値 (リンクダウン)

	ORG ( $P_e = 0.002$ )	ORG ( $P_e = 0.3$ )	DCY	NFB
$N_p(I=0.5)$	53.5	44.3	94.5	42.4

## 4.2 ノードダウン

ORG( $P_e = 0.3$ ), DCY, NFB について以下の条件でシミュレーションを行ったとき、収束時の  $N_p$  の平均値を表 2 に示す。

- time=2000 から  $I=0.3$
- time=7000 にノード 11 をダウン

表 2: 収束時の  $N_p$  の平均値 (ノードダウン)

	ORG ( $P_e = 0.3$ )	DCY	NFB
$N_p(I=0.3)$	19.2	30.7	21.2

図 7-1 から図 7-4 に ORG( $P_e = 0.3$ ), NFB, DCY( $P_e = 0.3$ ), NFB( $P_e = 0.3$ ) について以下の条件でシミュレーションを行った結果を示す。なお、DCY については  $v = 0.0025$  では発散したため、 $P_0 = 0.994, v = 0.00015$  とした。

- time=2000 から  $I=0.5$
- time=12000 にノード 11 をダウン
- time=35000 にノード 11 をアップ

## 5 考察

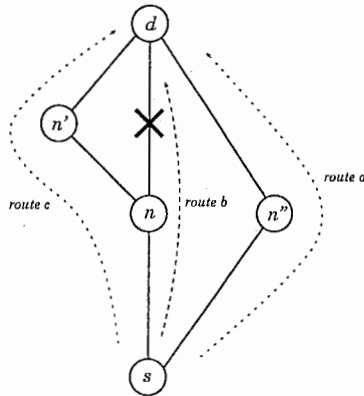


図 8-1: リンクダウン

ORG, NFB, DCY アルゴリズムの動作を図 8-1 を用いて説明する。3つのアルゴリズムの探求確率は同じとする。 $P_{d,n}^s = 1$  のとき、リンク  $(n, d)$  が切断すると仮定する。

### (i) $n'$ が存在しない場合

$s$  は  $d$  宛パケットを  $n$  へ送り続ける。 $n$  はリンク断を検知し、 $d$  宛パケットを  $s$  へ送り返す。従って、 $s$  と  $n$  の間にループパケットが存在する。このパケットのピンポン伝送は  $P_{d,n}^s$  が小さくなるまで続く。

この場合、時間とともに  $P_{d,n}^s$  が減少する DCY が最も早く  $P_{d,n}^s$  を小さくする。NFB と ORG の動作は全く同じである。リンクを 1 本しか持たないノードをスタブノードと呼ぶ。ネットワーク故障でスタブノードが発生すると  $N_p$  が急増する。(ノード 11 ダウン時のノード 12 はスタブノードである。)

### (ii) $n'$ が存在する場合

$s$  は、 $d$  宛パケットを  $n$  へ送り続ける。 $n$  はリンク断を検知、 $d$  宛パケットを確率  $P_{d,n'}$  で  $n'$  へ送る。また、確率  $P_{d,s}^n$  で  $s$  へ送り返す。従って、 $s$  と  $n$  の間にループパケットが発生する。

このパケットのピンポン伝送は、 $P_{d,n}^s$  または  $P_{d,s}^n$  が小さくなるまで続く。

#### (a) ルート $c$ の遅延が小さいとき

すべてのアルゴリズムが素早く  $P_{d,n'}$  を大きくするため、相対的に  $P_{d,s}^n$  は小さくなる。アルゴリズムの差はない。

#### (b) ルート $c$ の遅延が大きいとき

NFB はパス  $c$  を通るエージェントのトリップ時間によって、 $P_{d,n}^s$  を小さくできる。従って、ORG よりも  $P_{d,n}^s$  を早く小さくできる。DCY は、時間とともに  $P_{d,n}^s$  を小さくするため、ORG よりも早く小さくできる。

一方、ノードダウンは多くのエージェントロスとパケットループを引き起こす。このため DCY の減衰速度が最短パスルーチングへ自己組織化する速度を上回ると、均一ランダムルーチングになり、 $N_p$  が無限に発散する危険性を持つ。よって、減衰率を小さくする必要がある。減衰率の減少を補うために探求確率を増加することによって、DCY の適応性は維持できる。

## 6 おわりに

本実務訓練でお世話になりました担当の種田さんをはじめ、ATR 環境適応通信研究所の皆様にお礼を申し上げます。

## 参考文献

- [1] G.D.Caro and M.Dorigo. Mobile Agents for Adaptive Routing.

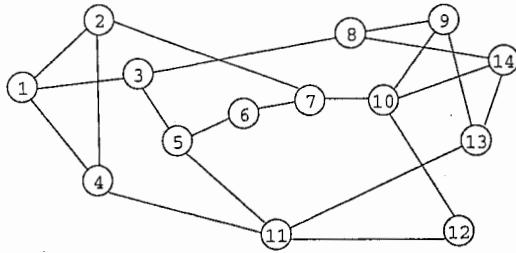


図 1-1: ネットワーク図

## 2 AntNet の振る舞い

### 2.1 学習率 $h$ の影響

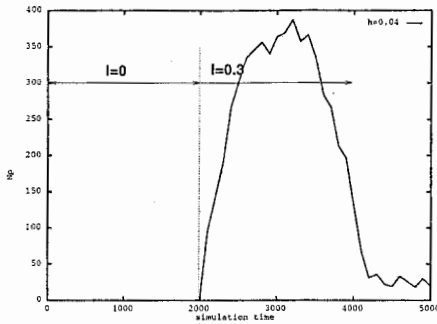


図 2-1:ORG( $h = 0.04$ )

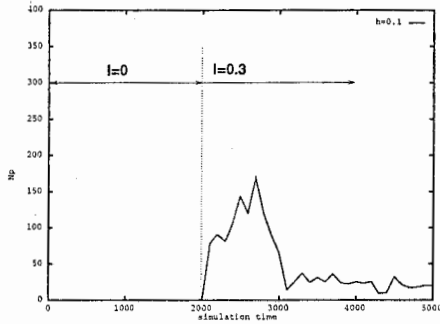


図 2-2:ORG( $h = 0.1$ )

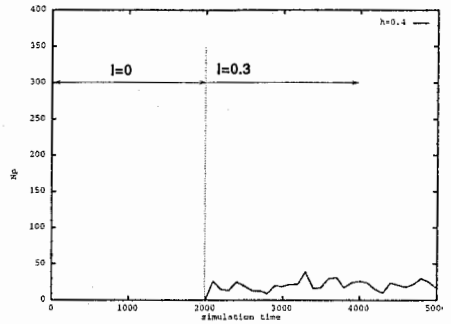


図 2-3:ORG( $h = 0.4$ )

### 2.2 振動をおこすトラヒック

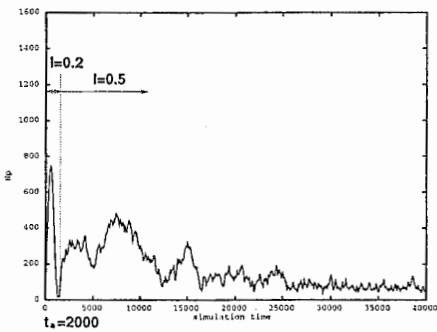


図 3-1:ORG( $h = 0.4$ )  $t_a = 2000$

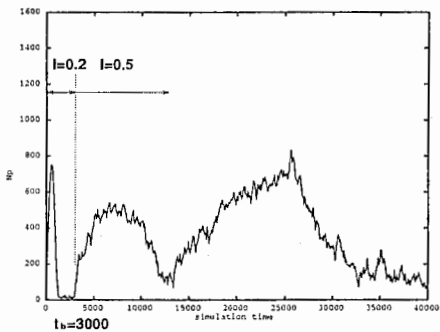


図 3-2:ORG( $h = 0.4$ )  $t_b = 3000$

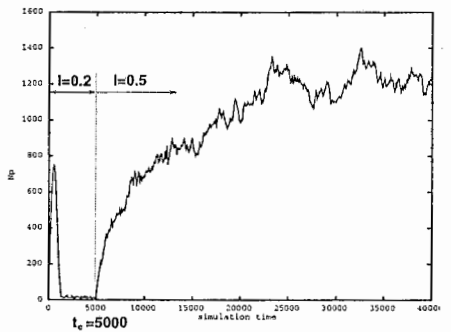


図 3-3:ORG( $h = 0.4$ )  $t_c = 5000$

### 2.3 振動と Lock の関係

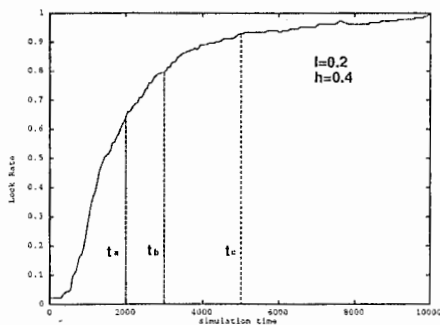


図 4-1:Lock Rate( $h = 0.4, I=0.2$ )

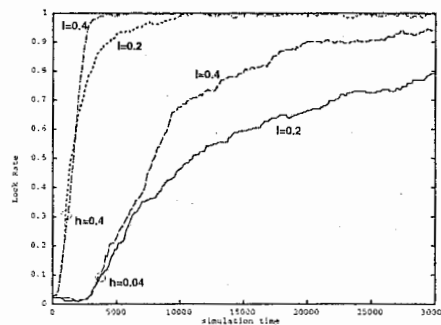


図 4-2:Lock Rate( $h = 0.04, 0.4, I=0.2, 0.4$ )

### 3 Lock-free アルゴリズム

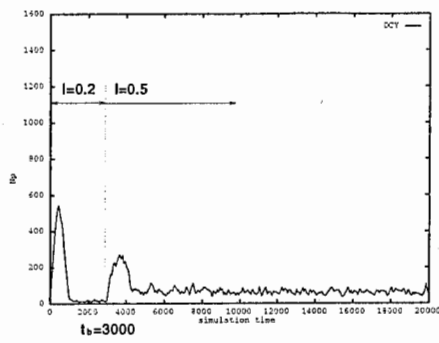


図 5-1:DCY

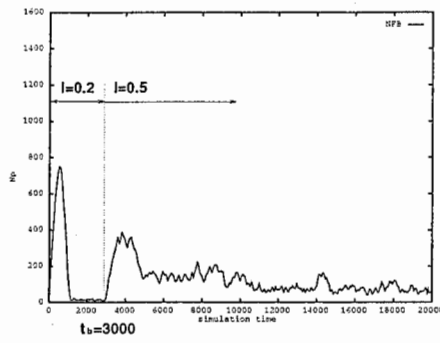


図 5-4:NFB

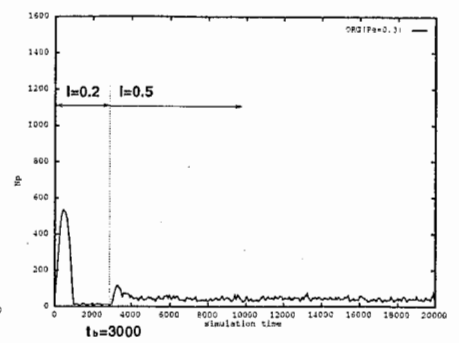


図 5-7:ORG( $P_e = 0.3$ )

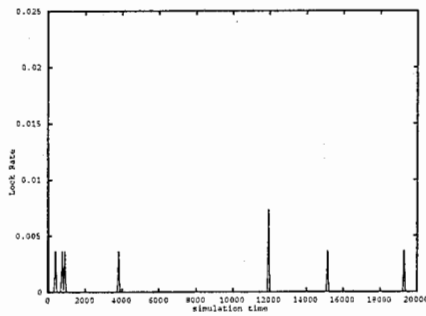


図 5-2:DCY,Lock Rate

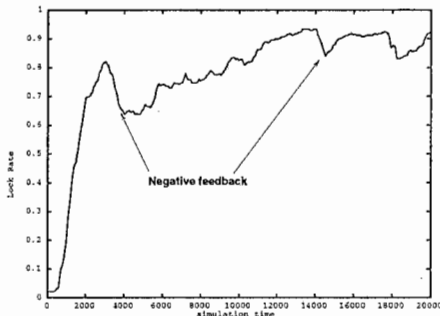


図 5-5:NFB,Lock Rate

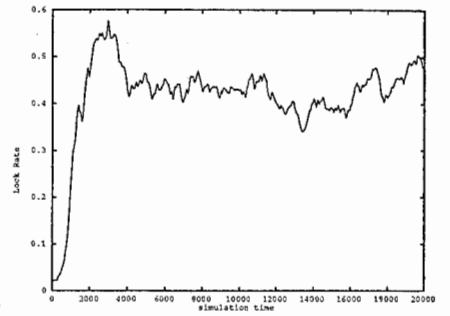


図 5-8:ORG( $P_e = 0.3$ ),Lock Rate

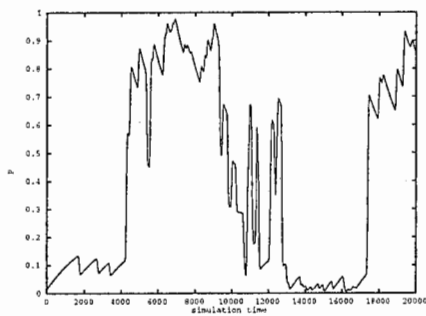


図 5-3:DCY, $P_{2,14}^{10}$

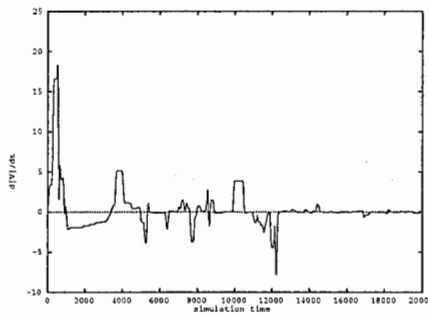
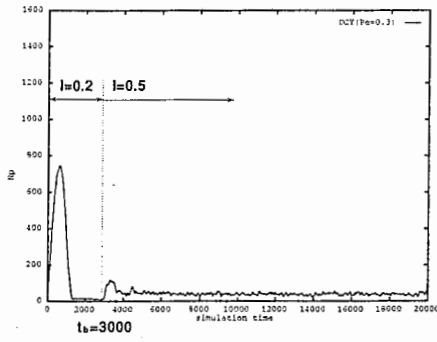
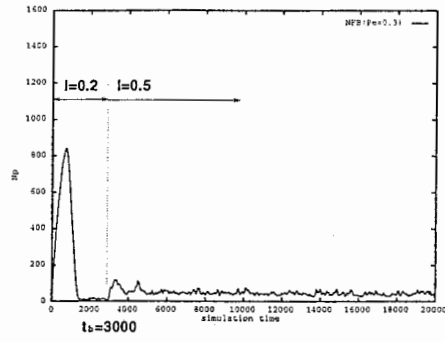


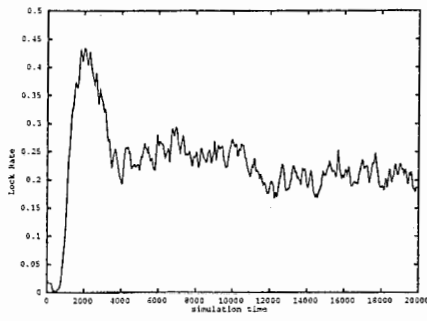
図 5-6:NFB, $\frac{d|V_s(\bar{\mu}, \bar{\sigma})}{dt}$



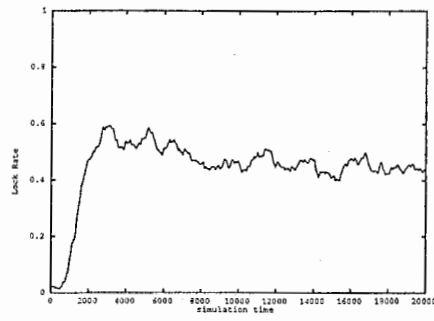
☒ 5-9:DCY( $P_e = 0.3$ )



☒ 5-11:NFB( $P_e = 0.3$ )



☒ 5-10:DCY( $P_e = 0.3$ ),Lock Rate



☒ 5-12:NFB( $P_e = 0.3$ ),Lock Rate



## 4 Network 故障に対する適応性

### 4.1 リンクダウン

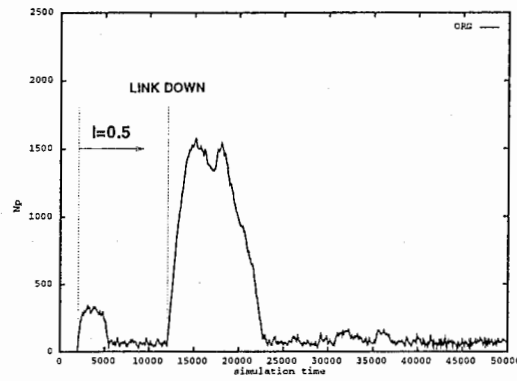


図 6-1:ORG( $P_e = 0.002$ )

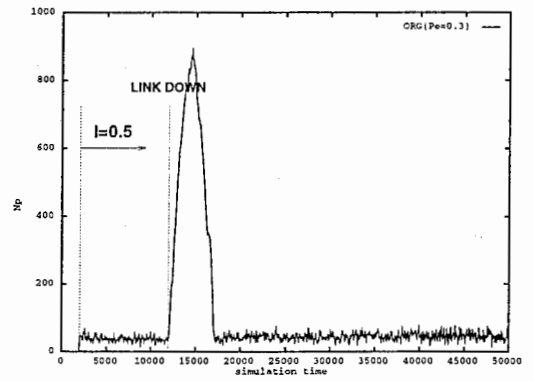


図 6-2:ORG( $P_e = 0.3$ )

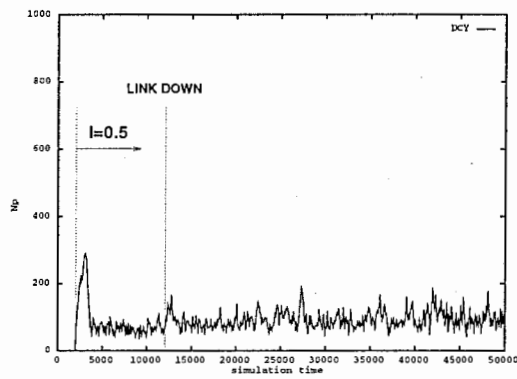


図 6-3:DCY

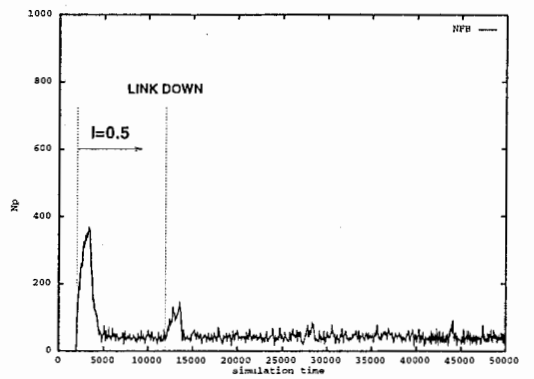


図 6-4:NFB

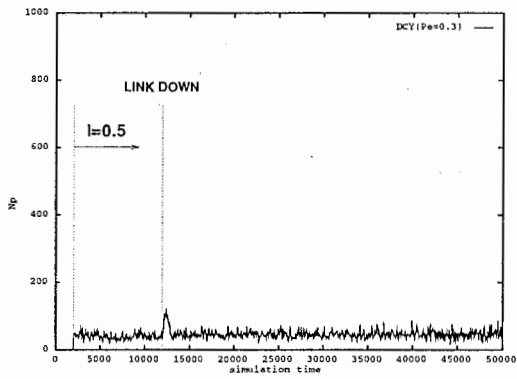


図 6-5:DCY( $P_e = 0.3$ )

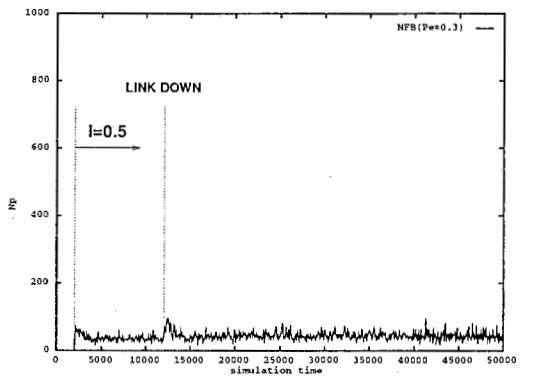


図 6-6:NFB( $P_e = 0.3$ )

## 4.2 ノードダウン

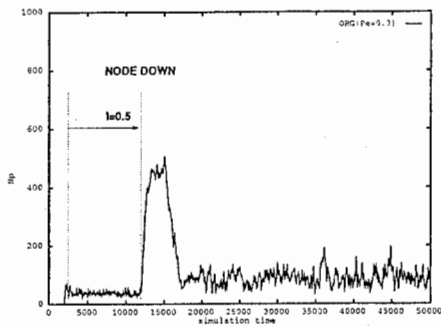


図 7-1:ORG( $P_e = 0.3$ )

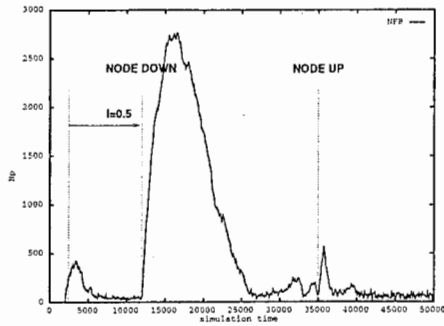


図 7-2:NFB

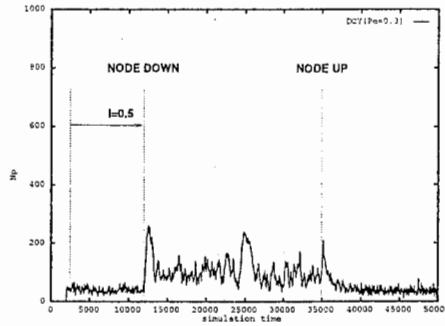


図 7-3:DCY( $v = 0.00015, P_e = 0.3$ )

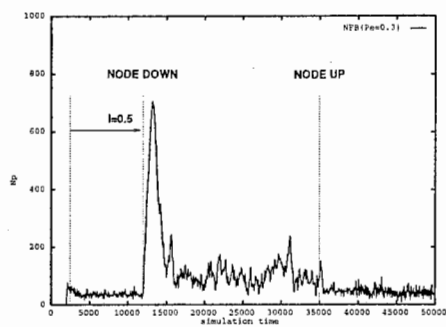


図 7-4:NFB( $P_e = 0.3$ )