ネットニュースにおける投稿行動の解析

瀬尾 雄三

1996 年 1 月

修士 (経営学) 論文

指導教官

橋田 温 教授

木島 正明 助教授

久野 靖 助教授

筑波大学 経営・政策科学研究科

経営システム科学専攻

要旨

近年、コンピュータネットワークを介した人と人とのコミュニケーションが社会のさまざまな分野で利用されるようになった。コンピュータネットワークというコミュニケーション手段は伝統的なコミュニケーション手段と異なった特性を持つことから、これによって形成される社会もまた伝統的な社会とは異なった挙動を示すものと考えられる。一方、コンピュータネットワークで交わされる情報は、全てデジタル情報であり、コンピュータネットワーク上の社会からは、そのコミュニケーションに関する大量のデータを自動的に得ることができるという特徴がある。

本論文は、このようなコンピュータネットワーク上の情報を機械的に処理することにより、コンピュータネットワーク上の社会の挙動に関する知見を得る試みである。コンピュータネットワーク上の種々のコミュニケーション形態の内でも、特に情報の収集が容易なネットニュースにおける投稿行動を扱う。いくつかのニュースグループについて長期間にわたって収集されたネットニュースの各記事から、投稿の時刻と、記事の参照関係を抽出し、これらを統計的に処理することにより、投稿者の行動を規定する特徴的なパラメータを抽出することを試みる。

この研究の結果、投稿された記事に対するフォローの数の期待値は記事によって異なった値を持ち、一定の分布を示すこと、この分布はニュースグループの性質に対応することが明らかになった。また、遅れの分析により、投稿者が記事をチェックする頻度の分布も、概ね求めることができた。

ニュースグループに投稿される記事数の経時変化は分枝過程とみなすことができ、これらの特徴的パラメータにより、投稿される記事数を予測することもできる。本論文では、このための手法についても扱う。

Analysis of Posting Behavior to Net News

Yuzo Seo
Graduate School of Systems Management
University of Tsukuba, Tokyo
3-29-1 Otsuka, Bunkyo-ku, Tokyo, Japan

Abstract

In recent years, person-to-person communications through computer networks have been spreading over various social fields. The characteristics of computer networks as communication media are different from traditional communication media, so it is supposed that societies formed on computer networks have different phenomena. Furthermore as all informations carried by computer networks are digital, it is easy to obtain an ample quantity of data about societies on computer network.

This thesis presents an analytical method of article posting behaviors in news groups which are supposed to represent communication activities in societies on computer network. Network news groups are selected as objects of analysis for communication behavior. We extract data of posting times and reference relationships between articles from net news articles collected from several newsgroups for some periods and try to obtain typical parameter describing article posting behavior.

It is concluded that the number of followers for each article can be modeled by Poisson distribution with different parameters depending on values of new articles and the characters of newsgroups. Furthermore, the distribution of the intervals that posters check the newsgroup can be estimated by delay of followers posting.

The time series of articles numbers posted to a newsgroup is assumed as branching process and can be estimated by the parameters of the newsgroup. In this article, the methods for this purpose are also treated.

謝辞

この論文を書くに当たり、橋田先生には、研究の進め方から論文の作成に至るまでの全般にわたって、懇切丁寧な御指導を頂きました。また、木島先生と久野先生には論文全般にわたるご助言の他、特に木島先生には確率過程の扱いについて、久野先生には計算機に関する種々のことがらについてご指導頂きました。ここに感謝の意を表します。

本研究が対象とした、ネットニュース、特に fj のニュースグループは、多くのボランティアの活動によって支えられています。また、この研究には多くのフリーソフトウエア (Free BSD、linux、gcc、mule、gnus、LaTeX 等) を使用しました。これらのソフトウエアを作成、保守し、また、ネットワークコミュニティを支えるために費やされている多大な努力に対し敬意を表します。

本研究を進めるにあたり、著者の職場である、三菱化学総合研究所・磁気メディア研究所の上司、同僚の方々には、勤務時間、業務分担などの面で多岐にわたりご配慮頂きました。また、妻昌子をはじめとする家族の温かな協力と応援も、この研究を進める上で、欠かせないものでした。どうもありがとうございました。

目次

1. はじめに

1.1. 研究の背景

コンピュータネットワークは、近年社会の幅広い分野で利用されるようになった。この中で取り扱われる情報も、かつての無機的な数値情報に代って、人と人とがコミュニケートするための、言語や音声、映像などで表現された情報が大きな割合を占めるようになった。また、コンピュータネットワークの範囲も、かつては一つの組織で閉じた形態であったが、近年、異なる組織間でコンピュータネットワークの相互接続が行なわれるようになり、世界のコンピュータが単一のネットワーク — インターネット — に接続される時代となった。

コンピュータネットワークを介したコミュニケーションの特性、特にインターネットを介しての人と人とのコミュニケーションの特性は、伝統的なコミュニケーションの特性とさまざまな点で異なっている。まず第一にいえることは、コンピュータネットワークは広い意味での距離を無効にするコミュニケーション手段であるということ、即ち、コミュニケーションに関わるものの置かれた環境や状態の差がコミュニケーションの障害にならないという点である。コンピュータネットワークは、離れた場所に送信者のメッセージを瞬時に伝達し、受信者の側に保存する。受信者はこのメッセージをすぐに受けとることもできるし、自分の都合の良い時刻まで待たせて受けとることもでき、更には受信者側で受けとる情報を選択することも可能である。これらの特性によって、コンピュータネットワークは地理的、時間的壁を越えた伝達だけでなく、身体的、社会的壁を越えた伝達さえも可能とする。

第二の相違点は、コンピュータネットワークを介した情報伝達の自由度の高さである。扱われる情報は全てデジタル情報であり、文字情報だけでなく、音声、静止画像、動画像などの伝達も可能であり、これらを片方向に送ることも、相互に送り合うこともできる。また、受信した情報は保存され、任意の媒体に複製することも、また、これを再加工することも容易である。

コンピュータネットワークは、新しいコミュニケーションチャンネルであり、人と人との新しい関係のあり方、新しいタイプの社会を作り出す。この社会はコンピュータネットワークの内部に形成される社会であるが、コンピュータネットワークの利用が拡大するに従い、コンピュータネットワークを取り巻く外側の社会にも影響を与えると考えられている。例えば、経営組織の今後のあり方について、情報の流れを制御することによって成り立って

いたこれまでのヒエラルキー形社会に代わって、専門家集団をフラットな関係に接続したネットワーク形社会に移行するという指摘もなされている [1, 2]。コンピュータネットワークというコミュニケーション手段は、伝統的なコミュニケーション手段とさまざまな点で異なる特性を持つことから、コンピュータネットワークによってつくり出される新しい社会もまた、伝統的な社会とは異なった挙動を示すものと考えられる。この新しい社会の挙動に関する知識を深めることは、単にコンピュータネットワークの内部の社会だけでなく、将来の人間社会の方向性を知る上でも有益と思われる。

本研究は、このような新しい社会の挙動に関する知識を深めるための試みの一つであり、コンピュータネットワークを介して多数の読者と意見を交換するネットニュースについて解析を行う。ネットニュースは、投稿者が記事を公開することによって情報の伝達がなされ、他の記事に対するフォロー記事の投稿が繰り返されることによって話題が発展し、議論が深まる。記事が投稿されただけでは、それが他者にどのような影響を与えたかは不明であるが、投稿された記事に対してフォロー記事が投稿された場合には、そこでコミュニケーションがなされ、何らかの人間関係が成り立っているということができる。このような観点からネットニュースの内部にみられる人間関係に注目した研究がいくつか行われている [7, 5]。

コンピュータネットワークによって形成される社会は、当然のことながらコンピュータや通信回線といったソフトウエア、ハードウエア資源によって支えられている。これらの資源の最適設計のためには、通信量の見積り、予測が欠かせない。これまでインターネット上で交わされるコミュニケーションに関して、いくつかの研究が行なわれている [3, 4]。これらは、情報量の推移を扱っているが、その情報量を決定する内部の構造の解析は行われていない。本研究はフォローによって示される記事の参照関係に注目し、ネットニュースというコミュニケーションの場を特徴付けるパラメータの抽出を行う。ついで、このようにして得られたパラメータを用いて、投稿される記事の量を予測することを試みる。

1.2. アプローチ

コンピュータネットワークの利用のあり方には様々な形態がある。人と人との間のコミュニケーションに関わる利用に限定しても、次のような利用方法が存在する。

- チャット (talk, phone を含む)
- 電子メール

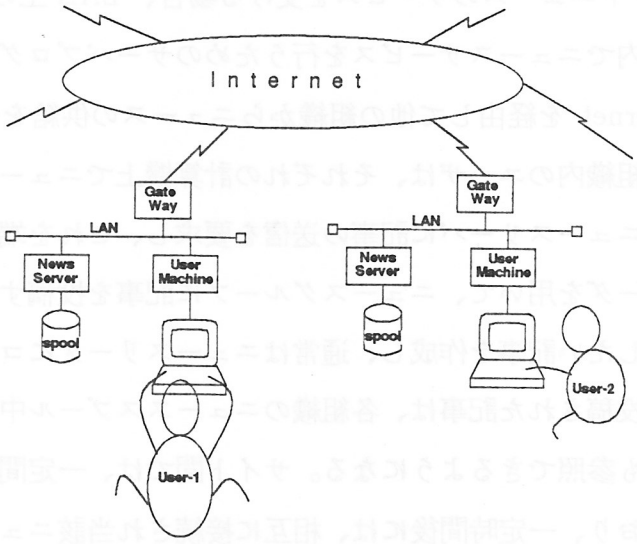

図 1: コンピュータネットワークとネットニュース

- ネットニュース (BBS を含む)
- WWW
- ...

これらの中で、ネットニュースはコミュニケーションの全てが公開されており、入手できるデータ量も多いことから解析に適していると考え、今回の研究の対象とした。ネットニュースについては、いくつかの文献 [8, 9, 10] に詳説されているが、以下、ニュースシステムの運用と投稿のなされ方について簡単に説明する。

図 1 に組織のコンピュータネットワークを Internet に接続してネットニュースに参加している様子を模式的に示す。現在、大学等の教育機関、国公立の研究機関、企業など、多くの組織がその内部にコンピュータネットワーク (LAN) を設置しており、それぞれの組織の LAN は Internet に接続され、相互に情報が交換される仕組みを形成している。Internet 上で行われているサービスの一つにネットニュースがある。ネットニュースは、話題によって階層的に分類されたニュースグループ名を持つ記事を相互に交換することによって成り立っているコミュニケーションの場であり、国際的には “comp” 等の Usenet ニュースグループが、日本国内では “fj” で始まるニュースグループ群[1]が活発な活動を続けている。

[1]1996 年 1 月 16 日現在、fj ニュースグループ管理委員会の公式に認めたニュースグループとして、291 個のニュースグループが存在する [12]。

組織がネットニュースのサービスを受ける場合、LAN 上の適当な計算機をニュースサーバとし、組織内でニュースサービスを行うためのサーバプログラムを実行させる。ニュースサーバは Internet を経由して他の組織からニュースの供給を受け、これをニュースプールに蓄える。組織内のユーザは、それぞれの計算機上でニュースリーダを実行し、ユーザの操作に従ってニュースサーバに記事の送信を要求し、これを端末画面上に表示する。ユーザはニュースリーダを用いて、ニュースグループに記事を投稿することができる。この場合、ユーザは投稿したい記事を作成し、通常はニュースリーダにコマンドを与えることにより投稿を行なう。投稿された記事は、各組織のニュースプール中に蓄えられ、そのサイトの他のユーザからも参照できるようになる。サイト間では、一定間隔でニュース、メイルの交換が行なわれており、一定時間後には、相互に接続され当該ニュースグループを購読している全てのサイトに投稿された記事が転送される。これらは、それぞれのサイトのディスク中に蓄えられ、そのサイト内のユーザの要求に応えて参照される。

スプール中に蓄えられた記事は、通常、1 ないし 2 週間後に削除される。しかし、ディスクスペースに余裕のあるサイトでは削除までの期間が長期にわたる場合もある。また、読者が興味を引いた記事は、その読者の個人的なディスクスペースに複写されて保存される場合もある。また、全ての記事を磁気テープ中に半永久的に保存するサイトもあり、CD-ROM に全ての記事をプレスする者も存在する。このような事情を考えれば、ネットニュースに対する投稿量を解析する際には、過去に投稿された記事の総量を扱うことが妥当であるともいえる。しかしながら、ネットニュースの運用においては、単位期間に投稿される記事の数と、それぞれの記事の大きさが問題となる。これは、通信経路の負荷を規定し、一定の期間内に投稿された記事を蓄積するためのディスクスペースを規定する。また、記事が少なければユーザもこれを読む意欲を失い、また、記事が多過ぎる場合はユーザが全ての記事を読むことが困難になり、有益な情報を見つけ出したり、適切な投稿を行うことが困難になる。そこで、本論文においては、単位時間に投稿される記事の量を解析の対象とした。

読者が自分自身の記事を投稿する時、全く新規の記事を投稿する場合と、自分の読んだ他人の記事に対して自分の意見を述べる（フォロー）場合との二つの場合がある。新規の投稿は、それ自体が独立した現象であるのに対し、フォローは既に行なわれた投稿の影響を受けて行なわれたものであって、過去の投稿行動に依存する。そこで、本論文では、投稿を新規の記事の投稿とフォローの記事の投稿の二つにわけてモデルを構築した。フォローが投稿

量の経時変化に与える影響に関しては、二つの要素を考える必要がある。第一は、ある記事に対して発生するフォローの数で、第二はそれらがどの程度の遅れをもってなされるかである。これらはいずれもそのニュースグループの特性と密接に結び付いた確率分布であると推定される。これらと、ランダムな現象によって記事の流通量の時系列モデルが記述される。

1.3. 本研究の意義

本研究は、コンピュータネットワーク上に形成される新しい社会のダイナミズムを扱う。この社会は、未だ形成の段階にあって発展を続ける社会であり、この社会の特性に対する知識を深めることは興味深い。更に、コンピュータネットワークという新しいコミュニケーション技術は、経営組織の変革など、人間社会全体に対しても大きな影響を与えるとの指摘もなされており、コンピュータネットワーク社会のダイナミズムを知ることは、将来の社会のダイナミズムを探る上でも大きな意味があるだろう。

本研究においては、コンピュータネットワークから大量の情報を機械的に収集し、これを数理的に解析するという手法をとった。このようにして社会関係を解析することは、単にコンピュータネットワーク上の社会の解析にとどまらず、一般的な人間社会に対する理解を深める目的に利用することもできる。投稿量の経時変化を、記事が独立に発生する新規投稿と、過去の投稿に触発されてなされるフォローの数に分けて扱う方法は、新製品の普及速度が、自己の判断で購入を決定するイノベータと、他人の購入に触発されて購入するイミテータに区分することで推定されるとするバスモデル [11] の主張とも共通し興味深い。

ネットニュースに対する投稿行為が数理モデルによって記述できれば、そのモデルに現れるパラメータは、実際のニュースグループに投稿する記事の統計的解析によって求めることができ、ニュースグループの特性を評価する指標となる。一方、これらのパラメータが与えられれば、確率過程を記述する方程式を代数的、あるいは計算機シミュレーションにより解くことによって、ニュースグループに投稿される記事数を見積もることができる。こうして見積もられる投稿量の平均値や分散は、ネットワークを支える計算機資源、通信資源を見積る上で有用であるだけでなく、ネットニュースの参加者に対して提供されるサービスの品質という観点からも重要な指標である。

2. 解析理論

2.1. ネットニュースのダイナミックモデル

2.1..1 ネットニュースに投稿される記事

ネットニュースの記事の一例を以下に示す。

```
01 | Path: frontier.rc.m-kagaku.co.jp!nntpsrv!seo
02 | From: seo@media.rc.m-kasei.co.jp (Y. Seo)
03 | Newsgroups: fj.news.usage
04 | Subject: Re: Quotation of signature
05 | Date: 4 Dec 95 21:49:10
06 | Organization: Mitsubishi Kasei Research Center, Yokohama, Japan
07 | Lines: 17
08 | Message-ID: <SEO.95Dec4214910@media.rc.m-kasei.co.jp>
09 | References: <SEO.95Dec4111226@media.rc.m-kasei.co.jp>
10 |     <1995Dec4.063805.14256@merope.opus.or.jp>
11 | NNTP-Posting-Host: media
12 | In-reply-to: void@merope.opus.or.jp's message of Mon, 4 Dec 1995 06:38:05 GMT
13 |
14 | In article <1995Dec4.063805.14256@merope.opus.or.jp>
15 | void@merope.opus.or.jp (Kusakabe Youichi) writes:
16 |
17 |  | で、いったいそれのどこが猫に見えるのでしょうか?
18 |
19 | 以下の「#」で囲まれた部分。
      ..... 以下省略 .....
```

ネットニュースに投稿された記事には、全く新しい話題を提供する記事と、既に投稿された記事に対するフォローの二種類がある。上の例はフォロー記事である。これがフォロー記事であることは、一般に次の 3 通りの方法で識別できる。

1. 記事の内部で他の記事を引用している (行 14-17)
2. Subject の先頭に "Re:" が付いている (行 4)
3. Reference が指定されている (行 9)

これらの識別手段は完全ではない。引用はフォローにおいて必須ではなく、引用を行なっていないフォロー記事があることは何ら問題でない。また、Subject は、フォロー記事であっても、適宜内容にあった適切なものに変更することが推奨されている。

Reference 部については、フォローを重ねることによって参照元のメッセージ ID が次々と追加されて肥大化することから、この部分に記述されるメッセージ ID の数を制限するこ

とが推奨されている[2]。このため、ときには フォロー記事であるにも関わらず、Reference 部分が全て削除された記事が投稿される場合もある。しかしこれは誤った使い方であり、フォローの記事の場合は Reference 部を残し、少なくとも最後に記述されたメッセージ ID は残すべきとされている。このような背景から、記事相互の参照関係の機械的な抽出のために Reference 部を用いることは、完全ではないものの、妥当な選択であると考えられる。

```
435: [ 35:a94507@matsus] Re: 30.Nov night
    540: [ 21:nomimura@fjct]
        541: [ 22:kora@denebu.f]
            687: [ 21:watanabe@itc.]
        542: [ 31:maki@spica.ec]
            543: [ 17:mon@st.rim.or]
436: [ 31:hszk@warabi.c] Intenet television
437: [ 23:tce0461@ip.ku] K-A-I-B-U-N
    438: [  7:nayuta@tky.th]
    439: [ 16:faichan@aqu.b]
        567: [ 21:j1293065@ed.k]
    440: [ 30:h04785@cnts.h]
    484: [ 25:taira@strg.so]
        485: [ 14:jamada@calc.e]
            486: [  8:EH1T-KRD@j.as]
    501: [ 11:felix@csl.ogi]
    502: [ 17:tabuchi@obp.c]
        503: [ 11:c9307681@mn.w]
        504: [  6:tomato@srl.me]
        505: [ 14:miyano@sbl.cl]
    510: [ 17:shimz@qad.cpg]
    515: [ 16:takasi-i@is.a]
    516: [ 15:miyano@sbl.cl]
        517: [ 14:tamura@yuri.i]
            518: [ 32:KFQ02620@nift]
                519: [ 11:nayuta@tky.th]
    664: [ 11:ide@mtl.t.u-t]
441: [  4:kazuyuki@nwk.] Trouble on fast reactor "Monju"
    442: [  6:h_watana@news]
```

図 2: Subject 一覧

fj.jokes の記事一覧表を mule 上のニュースリーダ "GNUS" によって表示したものを図 2 に示す。GNUS は「スレッド機能」を持ち、記事の参照関係により記事の一覧表をインデント表示する[3]。このリストに示された記事の内、記事番号 (先頭の数字) 435、436、437、

[2]多くのニュース投稿ソフトウエアには、ヘッダー部を編集する機能を備えており、投稿者による Reference 部や Subject 部の変更が可能である。

[3]GNUS の変数 "gnus-thread-ignore-subject" を "t" とした場合、参照関係は Reference 部の情報のみに基づいて決定される。この変数を "nil" とした場合は、Reference で参照関係が定義されており、かつ、Subject も一致することをフォローとみなす条件とする。

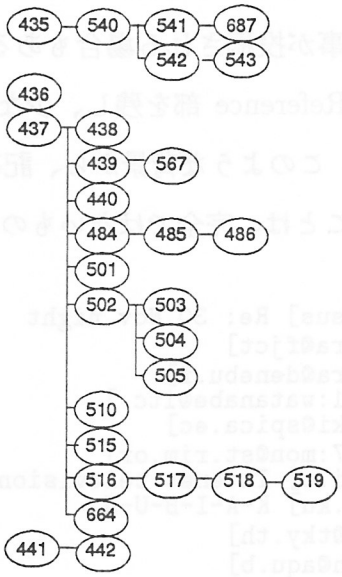

図 3: Subject の木構造

441 は、このニュースグループに関して、新規の投稿と考えられる。記事番号 435 は、Subject の先頭に "Re: " が付されていることからもわかるように、フォロー記事の形態をとっているが、この元記事は fj.rec.tv に投稿された記事であり、fj.jokes のニュースグループにおいては新規の話題となる[4]。

このような参照関係により、ネットニュースに投稿された記事は木構造を形成する (図 3)。記事の参照関係のつくる木構造は、図 2に示した Subject 一覧のインデント表示からも読みとることができる。前の行に対してインデントが深い記事は子に相当する。また、ある記事のインデント深さに対し、これより浅いインデントの記事が現れる以前に現れた同じインデント深さの記事は、その記事の兄弟である。ネットニュースの記事が形成する木構造において、ルートのノードは新規に投稿される記事であり、ルート以外のノードはフォローの記事である。それぞれのノードの子の数は、それぞれの記事に対するフォローの個数であり、フォローが付かなかった記事は末端ノードとなる。

ネットニュースに投稿された記事が読者に与える影響を評価するという観点から、それぞれの記事をルートノードとするサブツリーを評価するパラメータを考えると、その記事に

[4]異なるニュースグループへのフォローは、投稿者が Newsgroup フィールドを書き換えることによって行なうことができる。

対する直接のフォローの数、即ち子の数 k と、その記事をルートノードとするサブツリーの大きさ m の二つのパラメータが考えられる。また、子を持つノードをルートとするサブツリーに対しては、子孫の平均長さ ℓ を $\ell = (m-1)/k$ で定義することもできる。子の数は、フォローの数が記事固有の特性によって決定される場合に良い評価尺度になろう。一方、フォローを通して引き継がれていく「話題」の特性がフォローの数を決定する要因に含まれる場合は、サブツリーの大きさ m もしくは平均長さ ℓ についても合わせて評価する必要があろう。この論文では、子の数に着目した解析を中心に行ない、この結果に基づいてフォローを通して引き継がれるパラメータについて考察した。

2.1..2 モデル

ネットニュースの木構造の性質を調べることは、それ自体、ネットニュースの特徴を知る上で有用であろう。このような木構造の特性は、ネットニュースの投稿量の動きにも影響を与えるものと考えられる。

新規の話題を提供する記事は日々多数投稿され、これらをルートノードとする木構造が、日々多数形成される。木構造のそれぞれのノードには、一定の確率分布に従う数の、子のノードにあたるフォローの記事が連なる。フォローは、その元となる記事の投稿時刻に対して時間遅れを持ち、木構造の深さ方向は時間の経過に対応する。ネットニュース上に現れる記事は、このような木構造の多数の重ね合わせで表現することができ、それぞれの時間間隔に含まれるノードの総数の推移として全投稿量の推移が与えられる。

ルートに相当する新規の話題は、他の投稿と無関係に、ランダムに投稿される。一方、フォローの投稿は既に投稿されている記事の影響を直接に受けてなされる。ネットニュースの特定のニュースグループにこれから投稿される記事の数は、ランダムに発生する新規の記事と、これまでの投稿の結果ニュースグループ上におかれた記事に対するフォローの数から求めることができるが、後者は、それぞれの記事に対して何件のフォローが、どの程度の遅れを持ってなされるかに依存する。新規の投稿量は、非常に多数の参加者が独立に投稿することから、ポアソン分布に従うと考えられる。従って、新規の投稿量の確率分布を求めるためには、新規の投稿量の平均値を求めれば良い。一方、フォローの投稿量は、それぞれの記事に対するフォローの投稿数の確率分布と、フォローの遅れの確率分布によって規定される。これらは、ネットニュースの記事と投稿者の特性を反映しているものと考えられる。

ネットニュースに投稿された記事は、それが新規の話題を提供する投稿であるか他の記事に対するフォローの投稿であるかを問わず、フォローの対象となり得る。そこで、全ての記事の各々をルートとするサブツリーを考え、その特性を解析することによってニュースグループ全体の挙動を明らかにすることを試みる。

2.2. フォローの数

2.2..1 平均フォロー数の上限

記事が読まれると、それに対してフォローの記事が投稿される可能性がある。フォローが行なわれるか否かは確率的な現象であり、それぞれの記事に対してなされるフォローの数 K はランダムな値となる。一つの記事に対してなされるフォローの数 K の平均値を μ_K、一日あたりの記事の総投稿量の平均を A、新規の記事の投稿量の平均を N とすると、

$$A = N + \mu_K A$$

が成り立つことから、

$$A = \frac{N}{1 - \mu_K}$$

となる。ここで、$0 \leq \mu_K < 1$ でなければならず、これ以上の場合は記事数が発散する。以下の議論においては、$0 \leq \mu_K < 1$ の場合について解析を進める。

2.2..2 記事が等しい性質を持つ場合

それぞれの記事に対して投稿されるフォロー記事の数 K の期待値が全ての記事で一定値 p をとると仮定すると、記事は多くの（無限大で近似される数の）サイトで独立に読まれるため、フォローの有無は独立の試行と考えられ、k 個のフォロー記事が投稿される確率はポアソン分布で与えられ、次式に示す値をとる。

$$\Pr(K = k) = \frac{p^k}{k!} e^{-p} \tag{2.1}$$

2.2..3 記事の性質が異なる場合

上に示した (2.1) 式では、p を全ての記事に対して一定として扱ったが、一般に、興味を引く記事に対しては p は大きく、あまり興味を引かれないような記事に対しては p は小さい値をとるものと考えられる。

p の大きさが記事によって異なることを計算に反映させるため、記事のフォロー数の期待値が値 p をとる確率密度 $w(p)$ を想定する。この時、記事が k のフォローを受ける確率 $\Pr(K=k)$ は次式で与えられる。

$$\Pr(K=k) = \int_0^\infty w(p)\frac{p^k}{k!}e^{-p}dp \tag{2.2}$$

$w(p)$ の分布は未知であるが、種々の分布形状の表現ができる Γ 分布

$$w(p) = \frac{\theta^\alpha}{\Gamma(\alpha)}p^{\alpha-1}e^{-\theta p} \tag{2.3}$$

に従うと仮定する。この場合、積分は以下のように計算される[5]。

$$\begin{aligned}
\Pr(K=k) &= \int_0^\infty \frac{\theta^\alpha}{\Gamma(\alpha)}p^{\alpha-1}e^{-\theta p}\frac{p^k}{k!}e^{-p}dp \\
&= \frac{\theta^\alpha}{\Gamma(\alpha)k!}\int_0^\infty p^{\alpha+k-1}e^{-(1+\theta)p}dp \\
&= \frac{\Gamma(\alpha+k)}{\Gamma(\alpha)k!}\theta^\alpha(1+\theta)^{-\alpha-k} \\
&= \frac{\Gamma(\alpha+k)}{\Gamma(\alpha)k!}(\frac{\theta}{1+\theta})^\alpha(\frac{1}{1+\theta})^k
\end{aligned} \tag{2.4}$$

即ち、$\Pr(K=k)$ は負の二項分布となり、K の平均 μ_K と分散 σ_K^2 は次式で与えられる。

$$\begin{aligned}
\mu_K &= \frac{\alpha}{\theta} \\
\sigma_K^2 &= \frac{\alpha(1+\theta)}{\theta^2}
\end{aligned}$$

これらは観測値から計算されるので、次式により、α および θ を逆算することができる。

$$\theta = \frac{\mu_K}{\sigma_K^2-\mu_K} \tag{2.5}$$

$$\alpha = \frac{\mu_K^2}{\sigma_K^2-\mu_K} \tag{2.6}$$

なお、記事数が発散せず定常過程が実現されるという前提のもとでは、Γ 分布の要請を考慮して、$0<\alpha<\theta$ の条件が成り立つ。

[5]なお、Γ 関数の比は、$\frac{\Gamma(\alpha+k)}{\Gamma(\alpha)} = (\alpha+k-1)(\alpha+k-2)\cdots(\alpha+1)(\alpha)$ によって計算される。

2.3. 分枝過程による解析

元記事の投稿日の翌日に全てのフォローがなされるものと仮定すると、ネットニュースへの投稿量の問題は、移入を伴う分枝過程の問題と見做すことができる。

日 t の投稿量を A_t とし、各記事に対して、翌 $t+1$ 日に ξ_j 件のフォローが行なわれるとする。$t+1$ 日には、新規の投稿も N_{t+1} 件なされるとすれば、日 $t+1$ の投稿量 A_{t+1} は次式で与えられる。

$$A_{t+1} = \sum_{j=1}^{A_t} \xi_j + N_{t+1} \tag{2.7}$$

A_{t+1} の母関数 $\mathcal{A}_{t+1}(z)$ は $z^{A_{t+1}}$ の期待値で与えられ、次のように計算される。ここで $\mathcal{P}(z)$ はフォロー数分布の、$\mathcal{N}_{t+1}(z)$ は日 $t+1$ における新規投稿量の確率母関数である。

$$\begin{aligned}
\mathcal{A}_{t+1}(z) &= \mathrm{E}\left[z^{A_{t+1}}\right] \\
&= \mathrm{E}\left[z^{\sum_{j=1}^{A_t}\xi_j}\right]\mathrm{E}\left[z^{N_{t+1}}\right] \\
&= \{\sum_{k=0}^{\infty}\mathrm{E}\left[z^{\sum_{j=1}^{A_t}\xi_j}|A_t=k\right]\mathrm{Pr}(A_t=k)\}\mathrm{E}\left[z^{N_{t+1}}\right] \\
&= \left[\sum_{k=0}^{\infty}\{\mathcal{P}(z)\}^k Pr(A_t=k)\right]\mathrm{E}\left[z^{N_{t+1}}\right] \\
&= \mathcal{A}_t\{\mathcal{P}(z)\}\mathcal{N}_{t+1}(z)
\end{aligned} \tag{2.8}$$

よって、投稿量の定常確率分布の母関数 $\mathcal{A}(z)$ は次式で与えられる。

$$\mathcal{A}(z) = \mathcal{A}(\mathcal{P}(z))\mathcal{N}(z) \tag{2.9}$$

$\mathcal{A}(z)$ の一次および二次の導関数は次のように計算される。

$$\begin{aligned}
\mathcal{A}'(z) &= \mathcal{A}'(\mathcal{P}(z))\mathcal{P}'(z)\mathcal{N}(z) + \mathcal{A}(\mathcal{P}(z))\mathcal{N}'(z) \\
\mathcal{A}''(z) &= \mathcal{A}''(\mathcal{P}(z))\{\mathcal{P}'(z)\}^2\mathcal{N}(z) + \mathcal{A}'(\mathcal{P}(z))\mathcal{P}''(z)\mathcal{N}(z) + \\
&\quad 2\mathcal{A}'(\mathcal{P}(z))\mathcal{P}'(z)\mathcal{N}'(z) + \mathcal{A}(\mathcal{P}(z))\mathcal{N}''(z)
\end{aligned} \tag{2.10}$$

新規話題の投稿量がポアソン分布に従い、記事に対するフォロー数の分布が負の二項分布に従うとすると、それぞれの確率母関数とその一次および二次の導関数は次式で与えられる。

$$\begin{aligned}
\mathcal{N}(z) &= e^{-\lambda(1-z)} \\
\mathcal{N}'(z) &= \lambda e^{-\lambda(1-z)}
\end{aligned}$$

$$
\begin{aligned}
\mathcal{N}''(z) &= \lambda^2 e^{-\lambda(1-z)} \\
\mathcal{P}(z) &= \left(\frac{\theta}{1+\theta-z}\right)^{\alpha} \\
\mathcal{P}'(z) &= \frac{\alpha\theta^{\alpha}}{(1+\theta-z)^{\alpha+1}} \\
\mathcal{P}''(z) &= \frac{\alpha(\alpha+1)\theta^{\alpha}}{(1+\theta-z)^{\alpha+2}}
\end{aligned}
$$

これらの $z=1$ における値は次のようになる。

$$
\begin{aligned}
\mathcal{N}(1) &= 1 \\
\mathcal{N}'(1) &= \lambda \\
\mathcal{N}''(1) &= \lambda^2 \\
\mathcal{P}(1) &= 1 \\
\mathcal{P}'(1) &= \frac{\alpha}{\theta} \\
\mathcal{P}''(1) &= \frac{\alpha(\alpha+1)}{\theta^2}
\end{aligned}
$$

これらを用いて $z=1$ における $\mathcal{A}$ の導関数の値を計算すると以下のようになる。

$$
\begin{aligned}
\mathcal{A}'(1) &= \frac{\lambda\theta}{\theta-\alpha} \\
\mathcal{A}''(1) &= \frac{\alpha(\alpha+1)\theta\lambda}{(\theta-\alpha)^2(\theta+\alpha)} + \frac{\theta^2\lambda^2}{(\theta-\alpha)^2}
\end{aligned}
$$

これらの値を用いれば、投稿量の定常確率分布の平均値 μ_A およびその分散 σ_A^2 は次のように計算される。

$$
\begin{aligned}
\mu_A &= \mathcal{A}'(1) \\
&= \frac{\lambda\theta}{\theta-\alpha} \\
\sigma_A^2 &= \mathcal{A}''(1) + \mathcal{A}'(1) - \{\mathcal{A}'(1)\}^2 \\
&= \frac{\theta\lambda(\theta^2+\alpha)}{(\theta-\alpha)^2(\theta+\alpha)}
\end{aligned}
\tag{2.11}
$$

2.4. フォローの遅れ

前節では、フォローは元記事の投稿の翌日になされるものと仮定したが、実際のフォロー記事の投稿の元記事の投稿に対する遅れは分布を持つ。投稿量の経時変化には、フォローの

遅れの分布が影響するだろう。本論文における投稿量の確率分布の推定は、実際のネットニュースにおいて測定された、フォローが遅れ d を持つ確率密度 $Q(d)$ を用いて行なった。

フォローの遅れが発生する原因として、参加者のネットニュースに対するアクセスが間欠的であること、ネットワーク内での転送に遅れがあること、フォロー記事の作成に時間を要することなどがあげられる。ここでは、一つの仮説として、フォロー投稿者がネットニュースにアクセスする間隔がフォローの遅れを決定すると仮定してみる。ある読者がフォローを行なおうとするとき、前回にネットニュースにアクセスした時点で存在する全ての記事は読み捨てられており、フォローの対象となる記事が $d_{\max}$ 日前から現在に至る記事であると考える。即ち、フォローのチェック間隔が $d_{\max}$ 日であるとき、0 日から $d_{\max}$ 日までの遅れが生じると考える。このような前提の下では、チェック間隔の分布を遅れの分布より推定することが可能である。即ち、最大の遅れ日数 $d_{\max}$ の頻度割合を $P(d_{\max})$ とすると、チェック間隔が $d_{\max}$ である頻度割合はその $d_{\max}+1$ 倍あったはずである。また、チェック間隔 $d_{\max}$ は遅れ 0 から $d_{\max}$ の全てに対して $P(d_{\max})$ だけ寄与するので、遅れの分布からこれを差し引く。残った遅れの分布は、最大の遅れ日数が多くても $d_{\max}-1$ であるので、同様の手続きを繰り返すことによりチェック間隔の分布を求めることができる。

なお、単純にフォローの投稿日と元記事の投稿日から遅れを求めると遅れ 0 が他の遅れ日数と異なる評価がなされるという問題が生じる。これは、遅れ 0 は元記事とフォローが同一の日付であることを意味するが、フォローを行なう時点において全ての記事が投稿されているわけではなく、フォローより先に記事が投稿される確率は 1/2 と考えられるためである。日で区分した遅れの分布から求めることのできるチェック間隔の分布は、同じく、日で区分されたものであるため、上の手続きで求まるチェック間隔 0 の頻度は、チェック間隔 0 から 24 時間のものを含めたい。しかし、遅れの分布には上のような問題があるため、前記の手続きを単純に測定値に適用しただけでは、このような結果は得られない。そこで、補正した測定値に対して前記の手続きを行なうことによりチェック間隔を求める。補正の手続きはもっとも単純な方法によった。遅れ 0 には遅れ時間が 0 から 24 時間のフォローを含めたいが、遅れ 0 と観測されるのはこの内の半分であり、残りは遅れ 1 と観測される。24 時間以内のフォローであっても、元記事の投稿よりも早い時刻にフォローが行なわれる場合はそれは翌日のフォローになる。よって、遅れ 1 の頻度の 1/2 を遅れ 0 に移し、以下同様に、遅れ $d+1$ の頻度の 1/2 を遅れ d に移せば良い。

本論文では、投稿量の推定はフォローの遅れの分布形状を与えて行ない、チェック間隔の分布はデータの分析にとどめたが、チェック間隔の分布は読者のネットニュースに対するアクセス間隔の分布と同様と考えられ、読者の行動様式の一端を知る手がかりとなろう。

2.5. 確率分布による投稿量の推定

節2.3. では、フォローの遅れを一定として扱ったが、ここではフォローの遅れが確率分布で表される場合について考える。フォローが時間 d の遅れをもってなされる確率密度を $Q(d)$ とするとき、特定の記事に合計 k 件のフォローがある場合、d 日後に i 個の記事が投稿される確率 $P_{k,d}(i)$ は、次式で与えられる。ここで、$\bar{Q}(d) = 1 - Q(d)$ である。

$$P_{k,d}(i) = \binom{k}{i} Q(d)^i \bar{Q}(d)^{k-i}$$

一つの記事に k 件のフォローがなされる確率を $P(k)$ とすると、一つの記事に対して d 日後に i 件のフォローがなされる確率 $P_d(i)$ は次式で与えられ、

$$P_d(i) = \sum_{k=i}^{\infty} \binom{k}{i} Q(d)^i \bar{Q}(d)^{k-i} P(k)$$

確率母関数 $\mathcal{I}_d(z)$ は次のようになる。

$$\begin{aligned}
\mathcal{I}_d(z) &= \sum_{i=0}^{\infty} z^i \sum_{k=i}^{\infty} \binom{k}{i} Q(d)^i \bar{Q}(d)^{k-i} P(k) \\
&= \sum_{k=0}^{\infty} P(k) \sum_{i=0}^{\infty} z^i \binom{k}{i} Q(d)^i \bar{Q}(d)^{k-i} \\
&= \sum_{k=0}^{\infty} P(k) \{\bar{Q}(d) + zQ(d)\}^k \\
&= \mathcal{P}\{\bar{Q}(d) + zQ(d)\} \\
&= \mathcal{P}\{\mathcal{D}_d(z)\} \qquad (2.12)
\end{aligned}$$

ここで、$\mathcal{D}_d(z) = \bar{Q}(d) + zQ(d)$ は、確率 $Q(d)$ のベルヌイ分布の確率母関数である。

現時点 t より d だけ過去の全投稿数を $A(t-d)$、その確率母関数を $\mathcal{A}_{t-d}(z)$ とすると、これ等に対する t におけるフォローの数の確率母関数 $\mathcal{I}_{t,d}(z)$ は次式で与えられる。

$$\begin{aligned}
\mathcal{I}_{t,d}(z) &= \sum_{n=0}^{\infty} [\mathcal{P}\{\mathcal{D}_d(z)\}]^n \Pr\{A(t-d) = n\} \\
&= \mathcal{A}_{t-d}[\mathcal{P}\{\mathcal{D}_d(z)\}]
\end{aligned}$$

時点 t におけるフォローは、過去の全ての投稿に対するフォローの和であるため、その確率母関数 $\mathcal{I}_t(z)$ は次のようになる。

$$\begin{aligned}\mathcal{I}_t(z) &= \prod_{d=0}^{t} \mathcal{I}_{t,d}(z) \\ &= \prod_{d=0}^{t} \mathcal{A}_{t-d}\left[\mathcal{P}\{\mathcal{D}_d(z)\}\right]\end{aligned}$$

長時間の経過後フォロー数が安定すると考えると、その確率母関数が $\mathcal{I}(z)$ は次式となる。

$$\mathcal{I}(z) = \prod_{d=0}^{\infty} \mathcal{A}\left[\mathcal{P}\{\mathcal{D}_d(z)\}\right]$$

全投稿量は、新規の投稿量 N とフォローの投稿量の和で与えられるため、上の式は次のように変形される。

$$\mathcal{I}(z) = \prod_{d=0}^{\infty} \mathcal{N}\left[\mathcal{P}\{\mathcal{D}_d(z)\}\right] \mathcal{I}\left[\mathcal{P}\{\mathcal{D}_d(z)\}\right]$$

新規の投稿量は、平均 λ のポアソン分布に従うものと仮定し、それぞれの記事に対するフォローの個数は、パラメータを α および θ とする負の二項分布に従うと仮定すると、それぞれの確率母関数は次のように与えられる。

$$\begin{aligned}\mathcal{N}(z) &= e^{-\lambda(1-z)} \\ \mathcal{P}(z) &= \left(\frac{\theta}{1+\theta-z}\right)^{\alpha}\end{aligned}$$

これより、

$$\mathcal{P}\{\mathcal{D}_d(z)\} = \{\frac{\theta}{1+\theta-\mathcal{D}_d(z)}\}^{\alpha}$$

これらを先の式に代入すると次を得る。

$$\begin{aligned}\mathcal{I}(z) &= \prod_{d=0}^{\infty} \exp\left[-\lambda + \lambda\{\frac{\theta}{1+\theta-\mathcal{D}_d(z)}\}^{\alpha}\right] \mathcal{I}\left[\{\frac{\theta}{1+\theta-\mathcal{D}_d(z)}\}^{\alpha}\right] \\ &= \exp\left[-\lambda \sum_{d=0}^{\infty}\{1-\mathcal{P}(\mathcal{D}_d(z))\}\right] \prod_{d=0}^{\infty} \mathcal{I}\{\mathcal{P}(\mathcal{D}_d(z))\} \qquad (2.13)\end{aligned}$$

また、一次及び二次の導関数は次のようになる。

$$\begin{aligned}\mathcal{I}'(z) &= \mathcal{I}(z) \sum_{d=0}^{\infty} \mathcal{P}'(\mathcal{D}_d(z)) Q(d) \left[\lambda + \frac{\mathcal{I}'\{\mathcal{P}(\mathcal{D}_d(z))\}}{\mathcal{I}\{\mathcal{P}(\mathcal{D}_d(z))\}}\right] \\ \mathcal{I}''(z) &= \mathcal{I}'(z) \sum_{d=0}^{\infty} \mathcal{P}'(\mathcal{D}_d(z)) Q(d) \left[\lambda + \frac{\mathcal{I}'\{\mathcal{P}(\mathcal{D}_d(z))\}}{\mathcal{I}\{\mathcal{P}(\mathcal{D}_d(z))\}}\right] +\end{aligned}$$

$$\mathcal{I}(z)\sum_{d=0}^{\infty}\mathcal{P}''(\mathcal{D}_d(z))Q(d)^2\left[\lambda+\frac{\mathcal{I}'\{\mathcal{P}(\mathcal{D}_d(z))\}}{\mathcal{I}\{\mathcal{P}(\mathcal{D}_d(z))\}}\right]+$$
$$\mathcal{I}(z)\sum_{d=0}^{\infty}\mathcal{P}'(\mathcal{D}_d(z))Q(d)\frac{\mathcal{I}''(z)\{\mathcal{P}(\mathcal{D}_d(z))\}\mathcal{P}'(\mathcal{D}_d(z))Q(d)\mathcal{I}\{\mathcal{P}(\mathcal{D}_d(z))\}}{\mathcal{I}\{\mathcal{P}(\mathcal{D}_d(z))\}^2}-$$
$$\mathcal{I}(z)\sum_{d=0}^{\infty}\mathcal{P}'(\mathcal{D}_d(z))Q(d)\frac{\mathcal{I}'\{\mathcal{P}(\mathcal{D}_d(z))\}^2\mathcal{P}'(\mathcal{D}_d(z))Q(d)}{\mathcal{I}\{\mathcal{P}(\mathcal{D}_d(z))\}^2} \tag{2.14}$$

$z=1$ における導関数は次のように計算される。

$$\begin{aligned}\mathcal{I}'(1) &= \frac{\alpha\lambda}{\theta-\alpha}\\ \mathcal{I}''(1) &= \frac{\alpha^2\theta^2\lambda^2+\alpha\theta\lambda(\alpha+1)(\theta-\alpha)\sum_{d=0}^{\infty}Q(d)^2-\alpha^4\lambda^2\sum_{d=0}^{\infty}Q(d)^2}{(\theta^2-\alpha^2\sum_{d=0}^{\infty}Q(d)^2)(\theta-\alpha)^2}\end{aligned} \tag{2.15}$$

これより、フォロー投稿量の平均 μ_I および分散 σ_I^2 は以下となる。

$$\begin{aligned}\mu_I &= \mathcal{I}'(1)\\ &= \frac{\alpha\lambda}{\theta-\alpha}\\ \sigma_I^2 &= \mathcal{I}''(1)+\mathcal{I}'(1)-\{\mathcal{I}'(1)\}^2\\ &= \frac{\alpha\theta\lambda(\alpha+1)\sum_{d=0}^{\infty}Q(d)^2}{(\theta^2-\alpha^2\sum_{d=0}^{\infty}Q(d)^2)(\theta-\alpha)}+\frac{\alpha\lambda}{\theta-\alpha}\end{aligned} \tag{2.16}$$

また、全投稿量の平均 μ_A および分散 σ_A^2 は以下のように計算される。

$$\begin{aligned}\mu_A &= \mu_I+\lambda\\ &= \frac{\theta\lambda}{\theta-\alpha}\\ \sigma_A^2 &= \sigma_I^2+\lambda\\ &= \frac{\theta\lambda\{\alpha\sum_{d=0}^{\infty}Q(d)^2+\theta^2\}}{(\theta^2-\alpha^2\sum_{d=0}^{\infty}Q(d)^2)(\theta-\alpha)}\end{aligned} \tag{2.17}$$

(2.17) 式を 2.3.節で導いた (2.11) 式を比較すると、平均値は同一の式で与えられ、分散は $\sum_{d=0}^{\infty}Q(d)^2$ が 1 のとき双方の式は同一となる。$\sum_{d=0}^{\infty}Q(d)^2$ の値が小さくなると分散は低下し、$\sum_{d=0}^{\infty}Q(d)^2$ が 0 に近付くに従って平均値 $\frac{\theta\lambda}{\theta-\alpha}$ に近付く。

3. 実データの解析

3.1. 解析に用いたデータ

解析は、日本で公開されたネットニュース "fj" のニュースグループより、"fj.jokes"、"fj.jokes.d"、"fj.sys.mac" および "fj.sys.ibmpc" の 4 グループを選び、190 日から 300 日間のデータ収集期間内に到着したすべての記事を解析の対象とした。また、約 50 日間のデータ収集期間で、"fj.books"、"fj.soc.media"、"fj.misc"、"fj.rec.games.video.home.superfamicom"、"fj.news.usage" および "fj.soc.men-women" に投稿された記事についても解析を行なった。

各ニュースグループの性格は次の通り [12]。

fj.jokes： ジョークを投稿するためのニュースグループであり、フォローは、主に、先に出たジョークの改良版、パロディーである。

fj.jokes.d： fj.jokes に関する論評であり、時として議論の応酬になる。

fj.sys.mac： Machintosh に関する質問と、それに対する答が中心となっている。

fj.sys.ibmpc： IBM-PC とその互換機に関する質問とその答が中心。

fj.books： 本に関する話題

fj.soc.media： マスメディアに関する話題

fj.misc： 他のグループに分類されない話題

fj.rec.games.video.home.superfamicom： ファミリーコンピュータに関する話題

fj.news.usage： ネットニュースの使い方に関する議論

fj.soc.men-women： 男女関係のあり方に関する話題

データはニューススプールに存在する全ての記事を定期的に採集し、採集開始初期に見られる、投稿日の異常に早い記事を除外した。このような記事は、ニュース配送系のトラブルなどにより大きな遅れを生じたものと思われる。こうして得られた一連の記事から、更に、先頭から 7 日間に投稿された記事と、末尾から 7 日間に投稿された記事を除外した。これは、

採集開始直後の記事数が他に比べて少ない傾向を示すこと、期間終了の記事に関してはフォロー記事が採集されていない可能性があることから解析対象外とした。それぞれのニュースグループにおける解析期間と投稿量の単純統計量を表1に示す。日付は全て1995年1月1日からの経過日数である。

ニュースグループに投稿された記事の一日あたりの投稿量の経時変化の代表例として、ニュースグループ fj.jokes に投稿された記事の一日あたりの等講料の経時変化を図4に示す。他のニュースグループに対する投稿量の推移は AppendixA に示した。この図において、上側の折れ線は全投稿量の推移を、下側の折れ線は新規話題の投稿量の推移を示す。フォローの投稿量の推移は、これら二つの折れ線の間の幅で読み取ることができる。

この図を眺めると、ネットニュースに対する投稿行動の、種々の特徴が読みとれる。投稿量の経時変化は、ニュースグループそれぞれで、多少の違いが見られる。fj.jokes、fj.sys.mac、fj.sys.ibmpc の投稿量は比較的安定して推移するのに対し、fj.jokes.d の投稿量は不安定である。データ数は少ないが、fj.soc.media、fj.misc、fj.news.usage、fj.soc.men-women の投稿量も同様の不安定さを示す。このような違いは、そのグループで交わされるコミュニケーションのあり方が、主として、作品の発表であるのか、質問とそれに対する応答であるのか、それとも特定の話題に対する議論であるのかといった、ニュースグループの性質を反映しているものと考えられる。

周期を7日間とする明らかな周期変動も明瞭である。投稿量は、新規、フォロー共に7日毎に大きな落ち込みを示す。これは、投稿者が学校や勤務先から主に投稿していることによると思われる。

3.2. フォロー数の分布と分布曲線へのあてはめ

フォローの数の分布を、記事の強さの分布をΓ分布と仮定した場合(負の二項分布)と、記事が等しい性質を持つと仮定した場合(ポアソン分布)について、実測値と比較した結果を、各ニュースグループについて図5、図6、図7および図8に示す。また、それぞれの分布と実測値との差の χ^2 を表2に示す。いずれのニュースグループにおいても、負の二項分布が実測値に良く一致した。これは、それぞれのニュースグループに投稿される記事が均質ではなく、期待されるフォロー数が多い記事と、フォロー数の期待値が少ない記事が混在している結果であると考えられる。

上段: 平均、下段: 分散

News Group	解析期間	新規話題投稿数	フォロー投稿数	全投稿数
fj.jokes	125 - 349	12.44 44.09	14.76 69.86	27.20 184.70
fj.jokes.d	60 - 349	3.17 6.44	4.48 21.40	7.65 36.71
fj.sys.mac	165 - 349	16.45 81.59	35.83 406.89	52.28 772.75
fj.sys.ibmpc	170 - 349	5.19 11.91	12.53 104.64	17.72 159.59
fj.books	312 - 349	3.92 13.65	4.21 14.47	8.13 33.39
fj.soc.media	317 - 348	1.00 1.43	1.85 7.00	2.85 9.74
fj.misc	311 - 347	3.02 13.63	7.49 57.74	10.51 103.35
fj.rec.games.video. home.superfamicom	311 - 349	9.36 43.70	24.26 275.18	33.62 470.39
fj.news.usage	309 - 349	4.75 19.61	19.64 115.61	24.38 149.91
fj.soc.men-women	312 - 349	2.13 3.27	9.29 63.55	11.42 79.78

表 1: 各ニュースグループの解析期間および投稿量

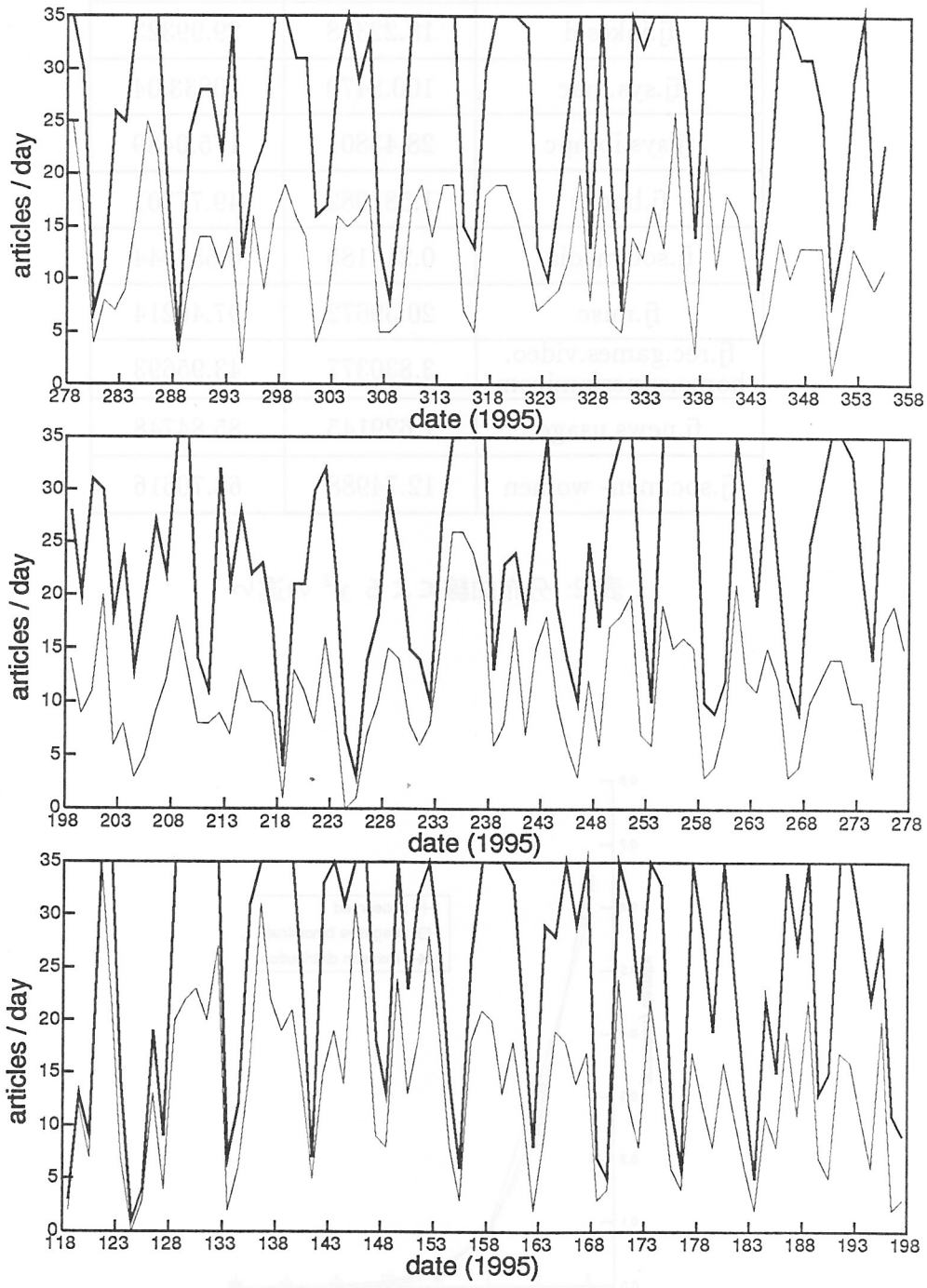

図 4: fj.jokes の投稿量の経時変化

News Group	負の二項分布	ポアソン分布
fj.jokes	71.45882	1741331.
fj.jokes.d	18.22308	29.99323
fj.sys.mac	100.8470	10833.04
fj.sys.ibmpc	28.43801	175.0489
fj.books	1.581982	49.77701
fj.soc.media	0.214185	0.551044
fj.misc	20.39672	97.40214
fj.rec.games.video. home.superfamicom	3.830377	43.95693
fj.news.usage	7.629145	85.84748
fj.soc.men- women	12.74988	68.70616

表 2: 分布曲線による χ^2 の違い

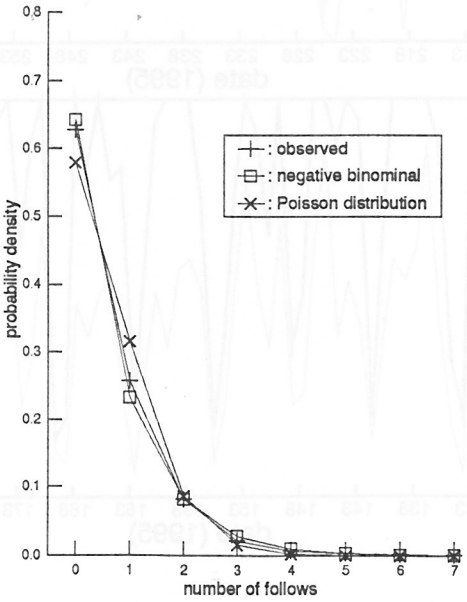

図 5: フォローの分布の比較 (fj.jokes)

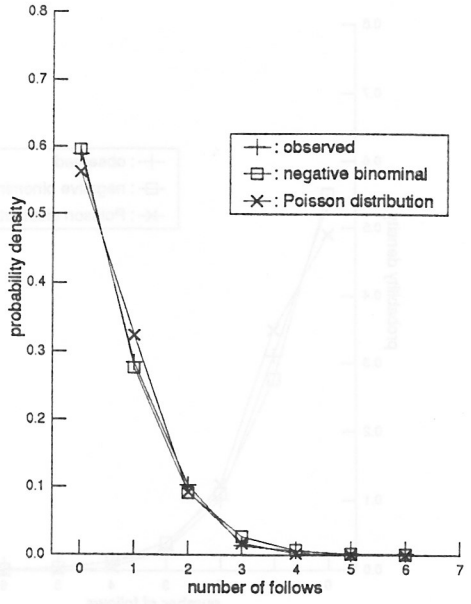

図 6: フォローの分布の比較 (fj.jokes.d)

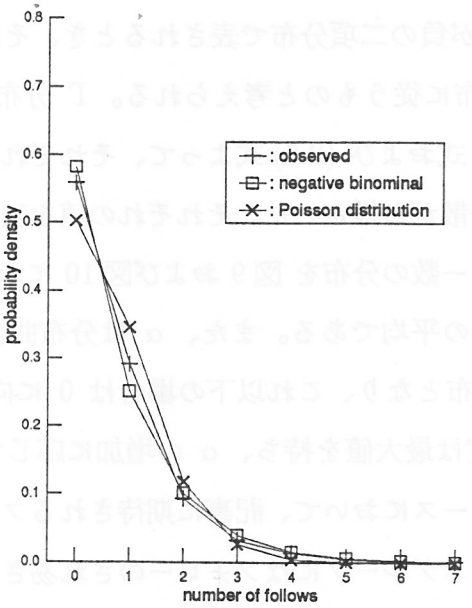

図 7: フォローの分布の比較 (fj.sys.mac)

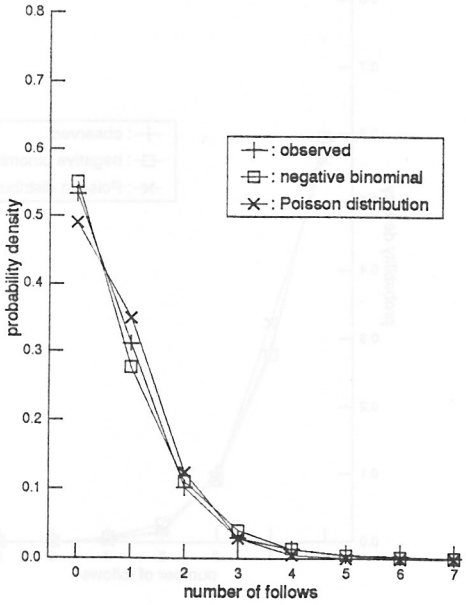

図 8: フォローの分布の比較 (fj.sys.ibmpc)

3.3. 記事の強さの分布

フォロー数の分布が負の二項分布で表されるとき、それぞれの記事のフォローの期待値は一様でなく、Γ 分布に従うものと考えられる。Γ 分布のパラメータ、即ち、(2.3) 式の α および θ を、(2.5) 式および (2.6) 式よって、それぞれのニュースグループにおけるフォローの数の平均値と分散から推定した。それぞれの値を表 3 に、これをから計算される、記事に期待されるフォロー数の分布を 図 9 および図 10 に示す。

α/θ はフォロー数の平均である。また、α は分布曲線の形状を決めるパラメータであり、$\alpha = 1$ で指数分布となり、これ以下の場合は 0 に向かって急速に増大する。α が 1 以上の場合は確率密度は最大値を持ち、α の増加に応じて平均値に近く、かつ急峻なピークを持つ。ネットニュースにおいて、記事に期待されるフォロー数の分布が急峻なピークを持つとき、そのニュースグループにはフォローのされ易さがほぼ等しい、類似した特性の記事が多いことを示す。逆に、α が 1 以下の場合は、フォローの期待されない記事が数多く存在する一方で、少数の記事が多数のフォローを集めることをことを意味する。これらのことから、α はニュースグループの記事の均質性を示すパラメータと考えることができる。

News Group	α	θ	α/θ
fj.jokes	1.067	1.948	0.548
fj.jokes.d	2.270	3.902	0.582
fj.sys.mac	1.153	1.672	0.690
fj.sys.ibmpc	1.670	2.330	0.716
fj.books	0.567	0.996	0.569
fj.soc.media	2.811	3.634	0.774
fj.misc	0.560	0.789	0.709
fj.rec.games.video. home.superfamicom	2.001	2.625	0.762
fj.news.usage	1.658	2.021	0.820
fj.soc.men- women	1.539	1.867	0.824

表 3: 記事の強さの分布パラメータ

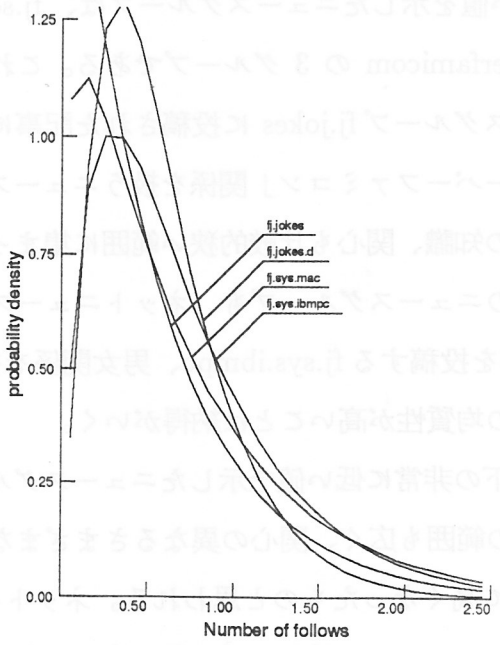

図 9: 記事の強さの分布 (その 1)

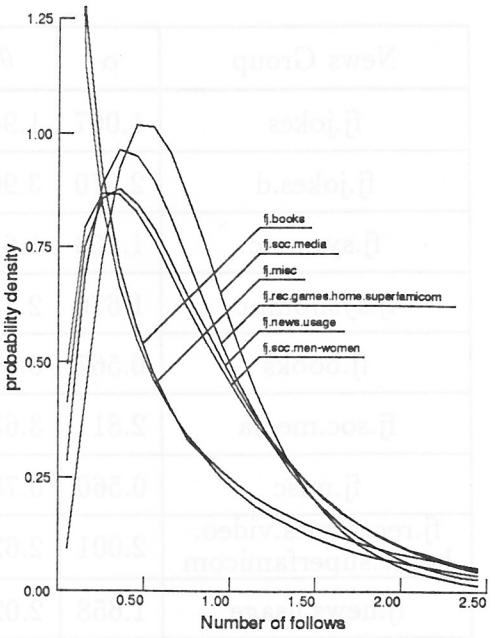

図 10: 記事の強さの分布 (その 2)

α が 2 以上の高い値を示したニュースグループは、fj.soc.media、fj.jokes.d、fj.rec.games.video.home.superfamicom の 3 グループである。これらはそれぞれ、マスコミニケーション関係、ニュースグループ fj.jokes に投稿された記事に関係する議論、任天堂製の家庭用ゲームマシン「スーパーファミコン」関係を扱うニュースグループであり、話題の領域が明確であり、参加者の知識、関心も比較的狭い範囲に集まっていると考えられる。またこれに次ぐ α が 1.5 以上のニュースグループも、ネットニュースの利用法を扱う fj.news.usage、IBM-PC 関係の記事を投稿する fj.sys.ibmpc、男女関係を扱う fj.soc.men-women の 3 グループであり、記事の均質性が高いことは納得がいく。

一方、α が 1 以下の非常に低い値を示したニュースグループは、fj.misc と fj.books である。これらは話題の範囲も広く、関心の異なるさまざまな参加者も参加することから、記事の不均質性が非常に高くなったものと思われる。ネットニュースは階層構造[6]で管理されているが、これら二つのニュースグループは fj の直下である点も興味深い。これらのニュースグループは fj 発足直後の古い時代に作られたニュースグループであり、その内部にまと

[6]階層構造は名前を「.」で区切ることで表現される。

まった話題が形成されると、これを扱う新たなニュースグループが作成され、関係する話題は新しいニュースグループに移される。そして、古くからあるニュースグループには、まとまりの薄い雑多な話題が残されることになる。この結果、ニュースグループの記事の均質性は低い値を示すものと考えられる。

ニュースグループ fj.jokes と fj.sys.mac の α の値は、これらの中間の、1 より多少高い値を示した。fj.jokes は冗談を投稿する場であり、完成度が高い作品であればフォローの必要はない。しかし、現実の投稿状況を観察した結果によれば、良くできたジョークにはフォローがなされているようであり、フォロー数の期待値が指数分布に近いという解析結果は、多数の受けない作品と少数の受ける作品が混在しているという現状を反映したものと思われる。fj.jokes は fj 直下の古いニュースグループであるにも関わらず、fj.misc や fj.books とは異なり、1 以上の高い α を示した。このような差が生じた原因として、joke という分野が一定のまとまりを持っていること、熱心な参加者 (常連) の存在などが考えられる。

解析期間	α	θ	α/θ
165 to 225	1.665	2.342	0.711
226 to 256	1.401	1.942	0.721
257 to 287	0.561	0.859	0.653
288 to 318	1.399	2.129	0.657
319 to 349	1.235	1.766	0.699

表 4: fj.sys.mac の記事の強さのパラメータの変化

fj.sys.mac は fj.sys.ibmpc とほぼ同等の性格を持つニュースグループであり、fj.sys.ibmpc と同様の均質性を持つものと考えられるが、分析の結果、α は 1 に近い低い値を示した。この原因を調べるため、解析期間を 5 区間に再分割して Γ 分布のパラメータを求めた。その結果、表 4 に示すように、α は、ほとんどの区間で 1.5 に近い値を示したものの、期間 257-287 (9/14 - 10/14) において 0.561 という極めて低い値を示した。この原因を探るためこの期間に投稿された記事の内容を調査したところ、9/23 に投稿された記事に始まる Mac と Windows95 の比較記事が大きなツリーを形成していることが見い出された。これとは別のツリーであるが同じ趣旨の記事も、その後いくつか投稿されている。このような結

果から、fj.sys.mac は通常は fj.sys.ibmpc と同等の均質性を示すが、Mac と Windows95 との比較という、ニュースグループ本来の話題とは趣を異にする記事が大量に投稿されたため、ニュースグループの記事の均質性が失われたものと考えられる。

ニュースグループにおける記事の均質性は、参加者にノイズの少ない情報を与え、参加者同士の親密性を増す効果があると考えられる。α の値を監視して、これを一定のレベルに維持することは、参加者の満足感の高い居心地の良いニュースグループをつくる上で有益であろう。一方で、均質性を重視するあまり、異質な記事の無視、排除を招くようではネットニュースの発展を阻害する結果ともなろう。fj.sys.mac で行なわれた Mac と Windows95 の比較記事の盛り上がりの結果、OS の比較を行なうための専用のニュースグループを設けようという議論が起こった。これは、記事の均質性をあげてニュースグループに流れる情報の質を高めようという試みとして評価できようが、過度に均質性を追求する余り異端を排除してしまうという危険もまた考慮すべきである。このような判断に際して、パラメータ α は一つの指標となろう[7]。今回解析を行なったニュースグループは、fj のニュースグループの中の一部のニュースグループであり、この結果を直ちに fj 全体の評価に結び付けることは危険があろうが、α 値の高い、良質の情報が交わされるニュースグループがある一方で、α 値の低い、種々雑多な話題を吸収する場があるという事実は、ネットニュースの現時点における効用と将来へ向けての発展という二つの要求に fj のニュースグループ群が応えた結果であるということもできよう。

3.4. フォローの遅れ

長期間にわたってデータ収集を行なった 4 つのニュースグループ、fj.jokes、fj.jokes.d、fj.sys.mac および fj.sys.ibmpc について、フォローの遅れの分布からチェック間隔の分布を求めた。結果を図 11、図 12、図 13 および図 14 に示す。チェック間隔の頻度は 0 日および 1 日で最も高く、投稿者の多くがネットニュースに毎日アクセスしていることを示している。これ以上のチェック間隔では、2 – 3 日まで頻度の山が裾を引いていること、7 日前後に山が見られるなどの特徴が読みとられる。

[7]今回の解析結果からは、α 値の目安として、2 以上は粒の揃った家族的雰囲気のニュースグループで、悪くいえば他所者の入り難い場、1.5 前後は暖かみには多少欠けるかも知れないが、問題解決のための議論が効率的に行なわれる場、0.5 近くまで下がると、話題の分散した、まとまりに欠ける場といえそうである。

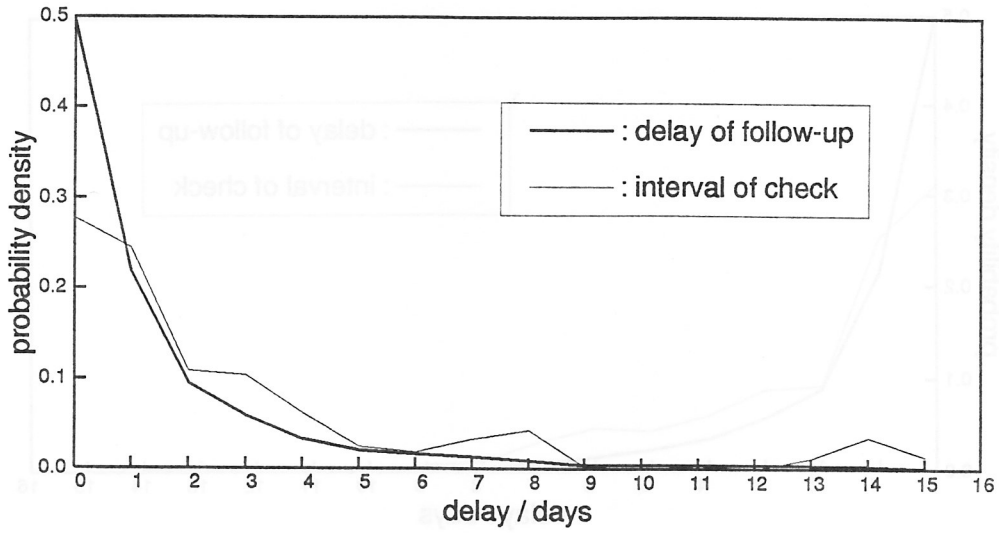

図 11: フォローの遅れとチェック間隔 (fj.jokes)

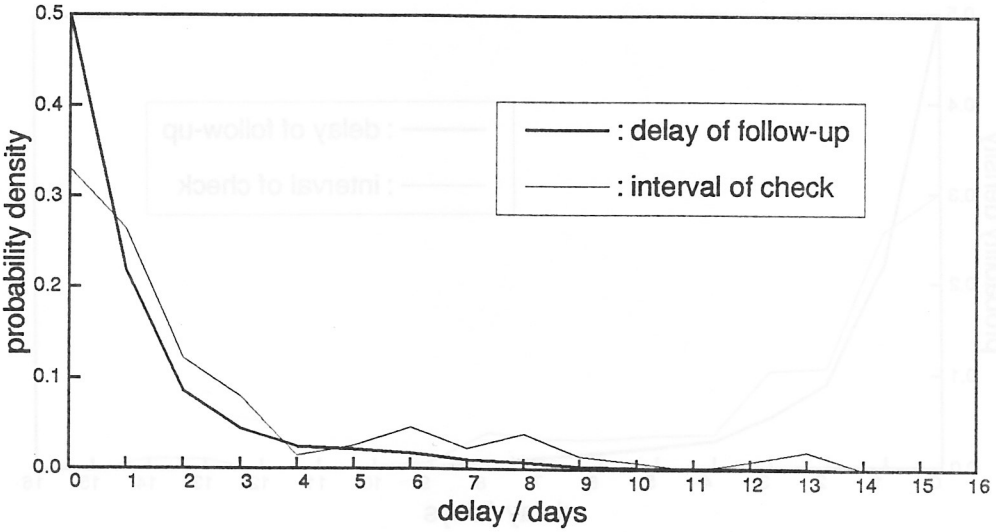

図 12: フォローの遅れとチェック間隔 (fj.jokes.d)

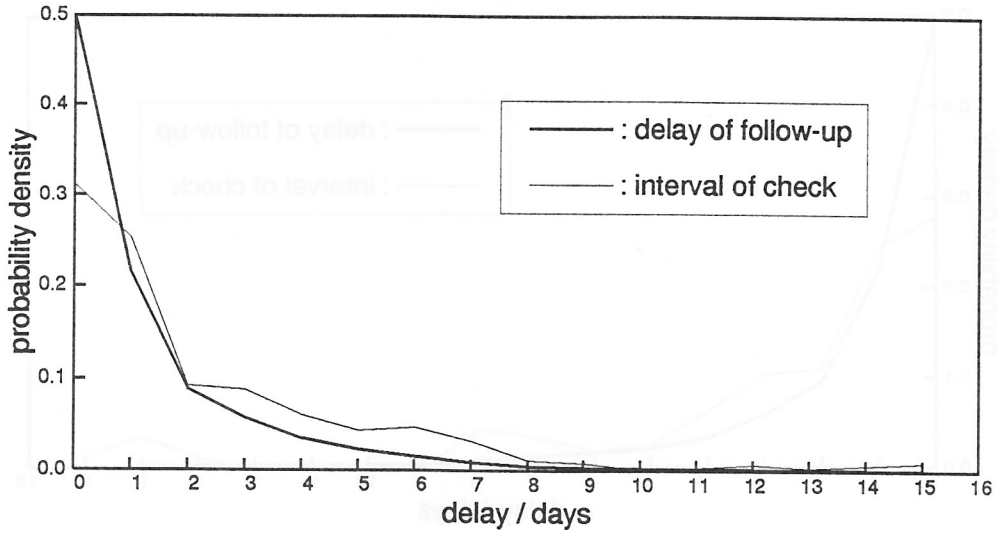

図 13: フォローの遅れとチェック間隔 (fj.sys.mac)

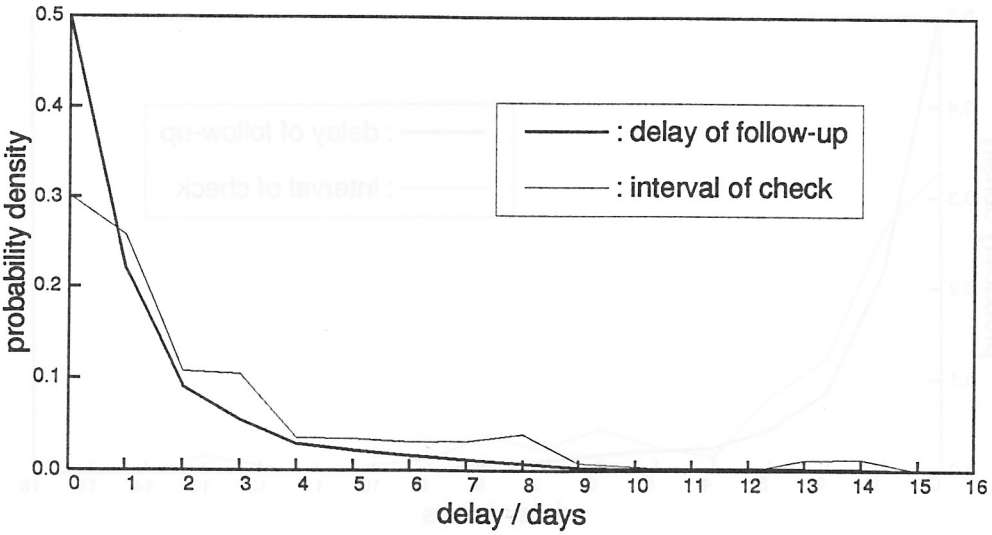

図 14: フォローの遅れとチェック間隔 (fj.sys.ibmpc)

3.5. 投稿量の平均と分散

実データの解析により各ニュースグループに対して得られたパラメータを表5 に、これを用いて投稿量の平均と分散を (2.11) 式および (2.17) 式で求めて実測値と比較した結果を表6 に示す。

計算値を実測値と比較すると、投稿量の平均値は比較的良く一致するものの、分散は実測値に比べてかなり小さい。この原因として、周期変動やトレンドなど、ランダムではない変動要素が現実の投稿量に含まれている点があげられる。また、それぞれの記事に対するフォローの数の期待値は、記事毎に独立であるとの前提の下に解析を行なっているが、ネットニュースにおいては、特別に議論を呼びやすい話題が提供されるとこれに対するフォローが更にフォローを呼ぶという現象が観察されている。これは、フォローの期待値は話題を同じくするフォロー記事に継承される可能性があることを示唆している。このような特性は投稿量の分散を増加させる方向に作用すると考えられるが、解析的に扱うことは困難であり、シミュレーションモデルによる解析が現実的であると思われる。

News Group	λ	α	θ	$\sum_{d=0}^{\infty} Q(d)^2$
fj.jokes	12.55	1.07	1.95	0.24
fj.jokes.d	3.25	2.27	3.90	0.27
fj.sys.mac	17.08	1.15	1.67	0.26
fj.sys.ibmpc	5.16	1.67	2.33	0.26
fj.books	3.97	0.57	1.00	0.19
fj.soc.media	0.78	2.81	3.63	0.22
fj.misc	2.32	0.56	0.79	0.26
fj.rec.games.video. home.superfamicom	9.82	2.00	2.63	0.25
fj.news.usage	4.83	1.66	2.02	0.27
fj.soc.men- women	2.13	1.55	1.87	0.23

表 5: 計算に用いたパラメータ

	実測値		(2.11) 式		(2.17) 式	
News Group	平均	分散	平均	分散	平均	分散
fj.jokes	27.46	196.27	27.76	50.83	27.76	31.88
fj.jokes.d	7.91	38.55	7.77	13.51	7.77	8.91
fj.sys.mac	54.05	834.61	55.01	148.11	55.01	69.40
fj.sys.ibmpc	17.73	172.01	18.20	48.91	18.20	22.63
fj.books	9.11	46.97	9.23	21.46	9.23	10.93
fj.soc.media	3.31	14.31	3.45	10.42	3.45	4.16
fj.misc	9.57	143.68	7.99	30.50	7.99	11.26
fj.rec.games.video. home.superfamicom	36.59	651.21	41.34	127.41	41.34	51.77
fj.news.usage	27.00	210.29	26.86	115.42	26.86	36.52
fj:soc.men- women	13.76	116.67	12.12	54.43	12.12	15.73

表 6: 投稿量の平均値と分散の比較

3.6. 週間変動

それぞれのニュースグループに投稿された記事を、投稿日の曜日によって区分して計数した結果を表 7 に示す。土曜日と日曜日の投稿量は、明らかに他の曜日に比べて低下する。週末における投稿量の低下の程度は、fj.rec.games.video.home.superfamicom で著しく、fj.news.usage で少ない。このような傾向も、ニュースグループの性質を反映しているのではないかと推測される。ビデオゲームの愛好者は週末はゲームに忙しくて投稿する時間的余裕がないこと、fj.news.usage は、議論が深刻な方向に発展する傾向があるため、週末といえどもネットニュースから離れられない、等がその背景として考えられる。

時系列解析を厳密に行なうには周期変動も考慮に入れるべきである。2.節で展開した確率過程の解析では、周期変動を無視した単純なモデルを採用した。周期変動を考慮した確率過程の解析的研究は今後の課題として残されている。

前節で、投稿量の予測を解析的に行なった結果と実測値が一致しない原因の一つに周期変動をあげた。実データに含まれる週間変動の影響を除去するためには、曜日毎に層別して計算を行なえば良い。そこで、曜日毎の投稿量の変動係数を求めて、全体に対する値と、計

算値との比較を行なった[8]。結果を表8に示す。曜日毎に層別化したデータの変動係数は、全体に対して計算される変動係数に比べて低下し、(2.11)式で計算された変動係数に近い値をとる。しかしながら、より現実に近いと思われる、フォローの遅れの分布を考慮した(2.17)式で計算される変動係数は、週間変動を除去した実測値と比較しても小さい値であった。計算値と実測値の差は、週間変動によって全てを説明することはできず、投稿量の分散を推定するに際しては、話題の継承などの、他の要因についても考慮する必要がある。

News Group	日	月	火	水	木	金	土
fj.jokes	5.62	16.54	16.38	18.17	19.08	16.59	7.61
fj.jokes.d	5.16	15.87	18.45	19.55	18.58	15.78	6.60
fj.sys.mac	6.05	17.10	17.27	17.12	17.59	16.94	7.92
fj.sys.ibmpc	4.85	18.81	19.35	19.06	17.59	14.46	5.87
fj.books	4.48	10.45	18.21	21.79	20.00	16.12	8.96
fj.soc.media	5.71	18.10	27.62	11.43	15.24	13.33	8.57
fj.misc	4.08	15.16	25.66	19.24	14.58	15.16	6.12
fj.rec.games.video. home.superfamicom	2.63	20.63	20.41	20.26	18.51	14.05	3.51
fj.news.usage	10.58	19.80	18.09	13.54	13.77	14.90	9.33
fj.soc.men- women	6.80	17.86	9.90	23.69	16.50	16.89	8.35

表 7: 曜日による投稿量の違い (曜日毎の百分率)

[8]分散は平均値の影響を受けるため、ここでは変動係数を用いて比較を行なった。

News Group	全体	(2.11) 式	(2.17) 式	日	月	火	水	木	金	土
fj.jokes	0.51	0.26	0.20	0.45	0.30	0.32	0.27	0.29	0.39	0.36
fj.jokes.d	0.79	0.47	0.38	0.72	0.56	0.69	0.68	0.57	0.51	0.83
fj.sys.mac	0.53	0.22	0.15	0.37	0.30	0.30	0.35	0.37	0.45	0.37
fj.sys.ibmpc	0.74	0.38	0.26	0.59	0.42	0.51	0.56	0.66	0.50	0.61
fj.books	0.75	0.50	0.36	0.70	0.66	0.61	0.32	0.44	0.09	0.35
fj.soc.media	1.14	0.94	0.59	0.75	0.81	0.66	1.59	1.19	0.77	0.58
fj.misc	1.25	0.69	0.42	0.61	0.45	0.99	0.80	0.79	0.75	0.87
fj.rec.games.video. home.superfamicom	0.70	0.27	0.17	0.42	0.19	0.19	0.19	0.38	0.28	0.48
fj.news.usage	0.54	0.40	0.22	0.50	0.37	0.28	0.26	0.51	0.39	0.34
fj.soc.men- women	0.79	0.61	0.33	0.52	0.68	0.43	0.50	0.58	0.27	0.77

表 8: 曜日毎に区分した場合の投稿量の変動係数

4. シミュレーションモデル

4.1. 目的

ネットニュースを運用するために必要な計算機資源を予測するためには、単位時間内に投稿される記事の平均と分散を知る必要がある。2.3.節および 2.5.節では、ニュースグループの特性パラメータから、これらを解析的に求める方法について述べた。この結果を実測値と比較すると、投稿量の平均値は良く一致するものの、分散は実測値に比べて小さい値となった。この原因の一つに、実際の投稿量には 7 日を周期とする週間変動があり、分散を増加させていることが考えられる。また、更に、現実のネットニュースにおいては、興味を引く一連の話題に関する記事は、他の話題に比べてフォロー数の期待値も大きくなることが予想され、フォローの投稿に対するフォロー数の期待値は、元記事に対するフォロー数の期待値と独立であるとした解析理論の前提とも異なる可能性が考えられる。これらの複雑な現象を定量的に扱うため、投稿行動を数値的にシミュレートするモデルを作成し、種々のパラメータに対する投稿量の推移を計算することを試みた。

4.2. 投稿行動モデル

図15に投稿行動のモデルを示す。投稿行動の仮定は次のステップで行なわれる。

1. それぞれの期間に、ポアソン分布に従う数の新規投稿が発生する。週間変動を考慮する場合は、ポアソン分布のパラメータである平均投稿量として、曜日毎の新規投稿の投稿量の平均値を用いる。
2. それぞれの新規投稿は、Γ分布に従うフォローの期待値を持つ。
3. それぞれの投稿に対するフォローの数は、フォローの期待値をパラメータとするポアソン分布によって決定される。
4. フォローの投稿は、フォローの遅れの分布に従う遅れをもって行なわれる。週間変動を考慮する場合は、フォローの投稿日の曜日を求め、その曜日におけるフォロー投稿量の頻度割合が区間 $[0,1]$ の一様乱数よりも小さい場合はフォローの遅れを再計算する。
5. フォロー投稿に対するフォローの期待値は、元記事のフォローの期待値と、Γ分布に従うフォローの期待値の加重平均によって与える。ここで、元記事のフォロー期待値に対する重みの割合を「継承率」とする。
6. 前記フォローの期待値をパラメータとするポアソン分布により、次のフォローの数を決定する。
7. 4 に戻る

4.3. プログラムの構成

図16 に投稿行動のシミュレーションプログラムのフローチャートを示す。内容は、前記モデルと同様であるが、新規とフォローの投稿を一つの手続きにまとめたこと、フォローをフォロー日でソートされたスタックに積むなどにより、プログラムの簡素化と処理の高速化を図っている。

メインルーチンの処理は、指定された日数のそれぞれの日について、まず、当日に行なわれる新規投稿数をポアソン乱数によって決定する。次に、新規投稿数だけ投稿ルーチンを呼び出す。この際Γ乱数 [16, 14, 15] によってその投稿に期待されるフォローの平均値を求めておく。

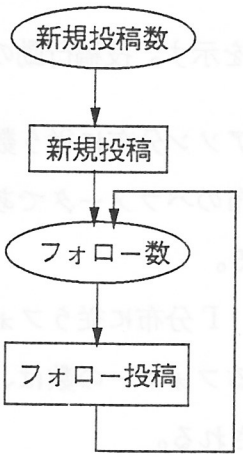

図 15: 投稿行動のモデル

Γ 分布

$$w(p) = \frac{\theta^{\alpha}}{\Gamma(\alpha)} p^{\alpha-1} e^{-\theta p}$$

に従う Γ 乱数は次のようにして発生させた。

まず、$\alpha < 1$ の場合は文献 [14] に示された次の手順で乱数を生成する。

1. 区間 $[0,1]$ の一様乱数 u_0 および u_1 を生成する。
2. $u_0 \leq e/(\alpha + e)$ の場合、
   (a) $p = (\frac{(\alpha+e)u_0}{e})^{-\alpha}$ を計算する。
   (b) $u_1 \leq e^{-p}$ なら p/θ を乱数値とする。
   (c) そうでなければ、一様乱数 u_0、u_1 の発生からやり直す。
3. そうでない場合、
   (a) $p = -\log(\frac{(\alpha+e)(1-u_0)}{\alpha e})$ を計算する。
   (b) $u_1 \leq p^{\alpha-1}$ なら p/θ を乱数値とする。
   (c) そうでなければ、一様乱数 u_0、u_1 の発生からやり直す。

また、$\alpha \geq 1$ の場合は文献 [15] に示された次の手順で乱数を生成する。

1. $c_1 = \alpha - 1$、$c_2 = \frac{\alpha - \frac{1}{6\alpha}}{c_1}$、$c_3 = 2/c_1$、$c_4 = c_3 + 2$、$c_5 = \frac{1}{\sqrt{\alpha}}$ とする。

2. 区間 $[0,1]$ の一様乱数 u_1 および u_2 を生成する。
3. α が 2.5 より大きい場合は $u_1 = u_2 + c_5(1 - 1.86u_1)$ とする。
4. u_1 が区間 $(0,1)$ に収まらない場合は、一様乱数 u_0、u_1 の発生からやり直す。
5. $w = \frac{c_2 u_2}{u_1}$ とする。
6. $\frac{c_3 u_1 + w + 1}{w} \leq c_4$ なら $\frac{c_1 w}{\theta}$ を乱数値とする。そうでなければ一様乱数の生成からやり直す。
7. $c_3 \log(u_1) - \log(w) + w < 1$ なら $\frac{c_1 w}{\theta}$ を乱数値とする。そうでなければ一様乱数の生成からやり直す。

次いで、その日にフォローすべき記事がスタック上にあれば、それらについて投稿ルーチンを呼び出す。投稿ルーチンは、与えられたフォローの期待値を用いてポアソン乱数を生成することによりフォロー数を決定する。ポアソン乱数は、フォローの期待値を用いてポアソン分布を計算して累積分布表を作成した後、一様乱数によりこれを逆引きすることによって求めた。

それぞれのフォローに対し、あらかじめフォローの遅れの累積分布を設定した数表を一様乱数で逆引きすることによりフォローの遅れを与え、投稿日を決定する。また、Γ 乱数によりそれぞれのフォロー記事に対するフォローの数の期待値を求める。この際、フォローに与えるフォローの数の期待値として、元記事のフォローの数の期待値を一部継承させることができるようにプログラムを作成した。これは、元記事のフォローの数の期待値を p_o、Γ 乱数で求めたフォローの期待値を p_Γ、継承率を r とするとき、次式によってフォローの期待値を与えることによって行なっている。

$$p = rp_o + (1 - r)p_\Gamma$$

フォローは後日なされるため、これらの値はフォロー投稿日でソートしてスタックに積む。

シミュレーションの結果である、毎日の新規投稿数とフォロー投稿数の経時変化は、実際のニュースグループから得られたデータと同じ形式でプロットする。また、それぞれの投稿の記事番号と投稿日、およびフォローの投稿である場合は元記事の記事番号と遅れを記録し、統計量の算出に用いる。

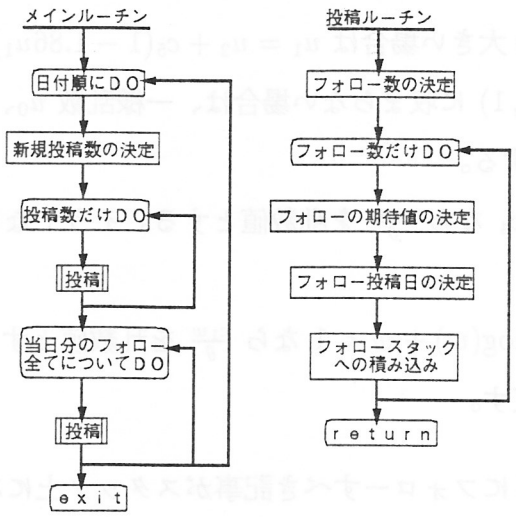

図 16: シミュレーションプログラムの流れ

4.4. 実行結果

実際のネットニュースから得られたパラメータを用いて、投稿行動のシミュレーションを行なった。計算された一日当たりの投稿量の内訳を、実測値と比較した結果を表 9 に、投稿量の推移を図 17、図 18、図 19 および 図 20 に示す。投稿量に関しては、シミュレーション計算の結果は、実測値とほぼ一致した。

ネットニュースの記事の参照関係によって形成される木構造の形状を表すパラメータである、子の数、木の大きさ、木の平均長さについて、実測値とシミュレーション計算値を比較した結果を表 10 に示す。パラメータの平均値は比較的良く一致したが、一部のニュースグループで、分散が大きく異なる結果が得られた。

実データの解析においても、シミュレーション計算においても、記事の性質は独立であり、フォローの期待値は他の記事の影響を受けずにランダムに決定されるという前提をおいて計算を行なった。しかしながら、現実には、フォロー記事に期待されるフォロー数は、その元記事に期待されるフォロー数と独立ではないと考えられる。即ち、ネットニュースの参加者にとって興味を引く話題に対しては、フォローの数は増加するものと考えられ、木の大きさ、あるいは長さの分散は、記事の性質を独立と仮定した場合に比べて大きな値をとるだろう。しかしながら、このような結果を示したのは、fj.jokes.d のみであり、木構造の形状パラメータの分散の決定には、他の要因も関連しているものと考えられる。木の大きさの分

上段: 平均、下段: 標準偏差

News Group	実測値			シミュレーション結果		
	新規話題投稿数	フォロー投稿数	全投稿数	新規話題投稿数	フォロー投稿数	全投稿数
fj.jokes	12.55	14.92	27.46	12.85	15.52	28.38
	6.84	8.62	14.01	5.73	8.87	13.24
fj.jokes.d	3.25	4.66	7.91	3.32	4.57	7.88
	2.60	4.74	6.21	2.18	3.85	5.23
fj.sys.mac	17.08	36.97	54.05	17.08	36.66	53.73
	9.39	20.95	28.89	7.08	19.05	24.27
fj.sys.ibmpc	5.16	12.57	17.73	5.48	13.33	18.81
	3.58	10.62	13.12	3.21	9.46	11.45
fj.books	3.97	5.13	9.11	3.92	5.14	9.05
	4.32	4.58	6.85	2.60	4.83	6.52
fj.soc.media	0.78	2.53	3.31	0.73	1.94	2.67
	1.46	3.28	3.78	0.93	2.97	3.31
fj.misc	2.32	7.24	9.57	2.42	5.19	7.61
	4.41	8.93	11.99	1.77	5.30	6.43
fj.rec.games.video.	9.82	26.77	36.59	9.50	29.32	38.82
home.superfamicom	7.73	19.56	25.52	5.47	19.69	23.97
fj.news.usage	4.83	22.17	27.00	4.65	22.74	27.39
	5.13	12.79	14.50	2.30	13.38	14.30
fj.soc.men-women	2.13	11.63	13.76	2.38	11.22	13.60
	2.12	9.72	10.80	1.71	9.00	9.67

表 9: 実測値とシミュレーションの比較

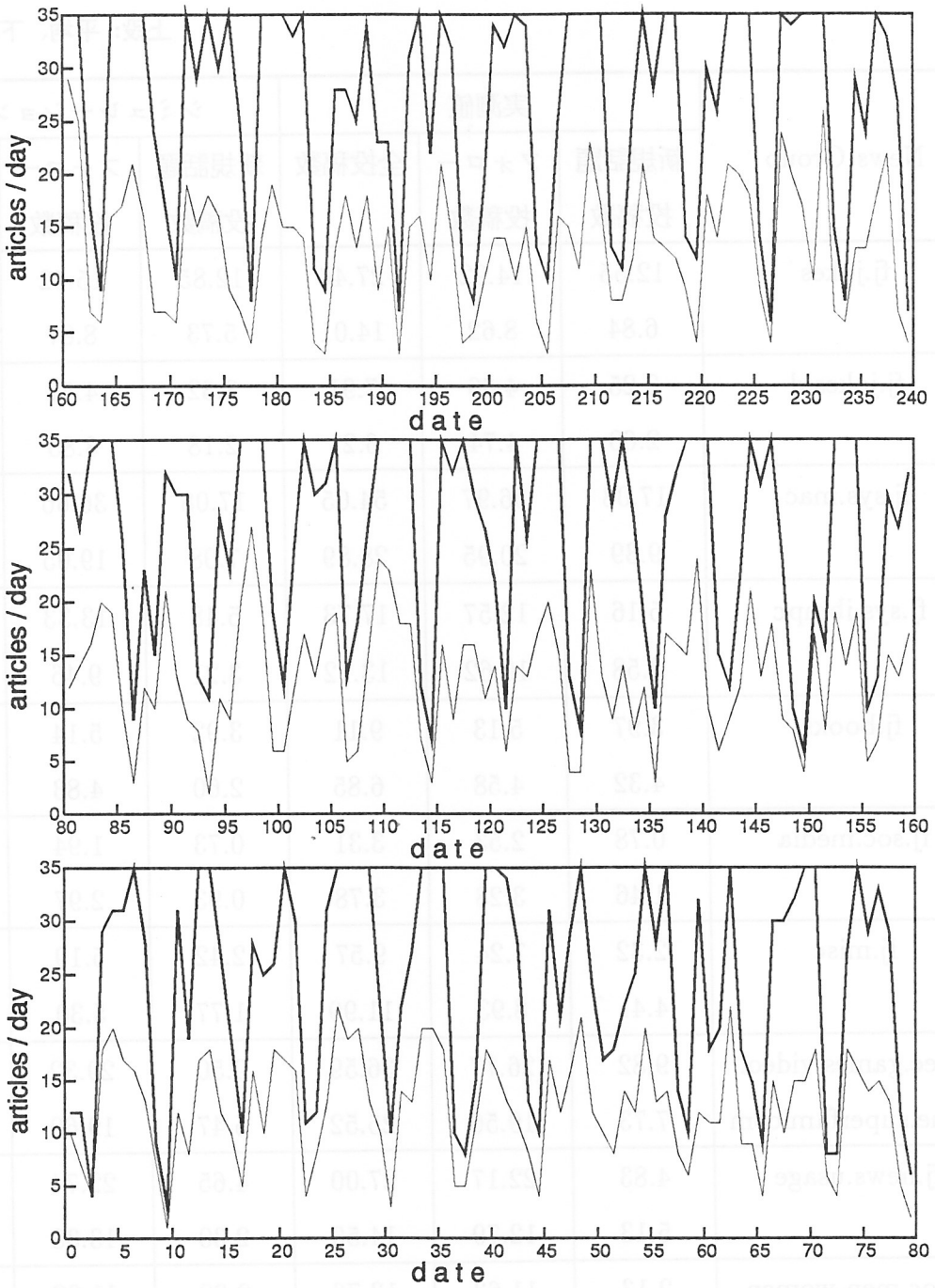

図 17: シミュレーション結果 (fj.jokes)

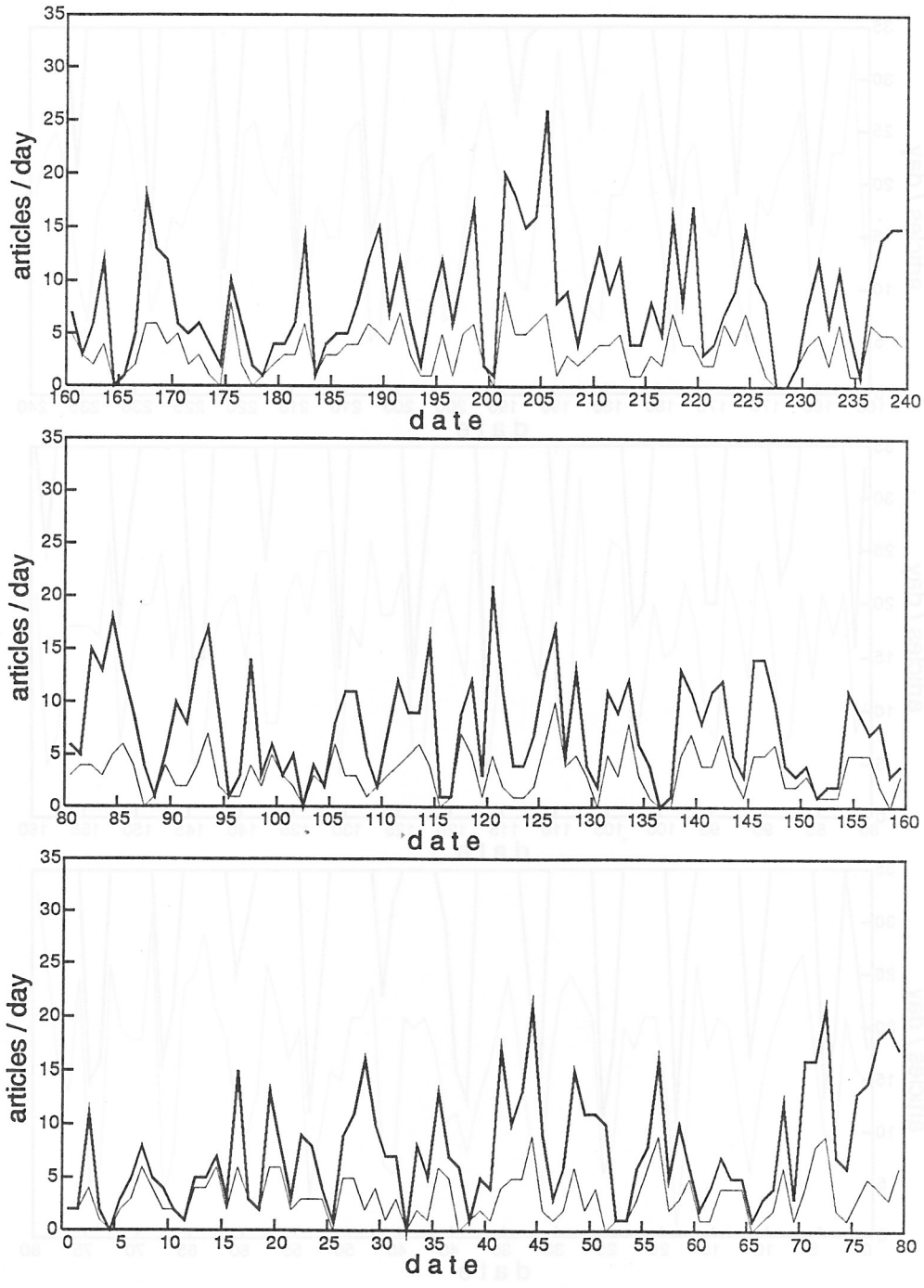

図 18: シミュレーション結果 (fj.jokes.d)

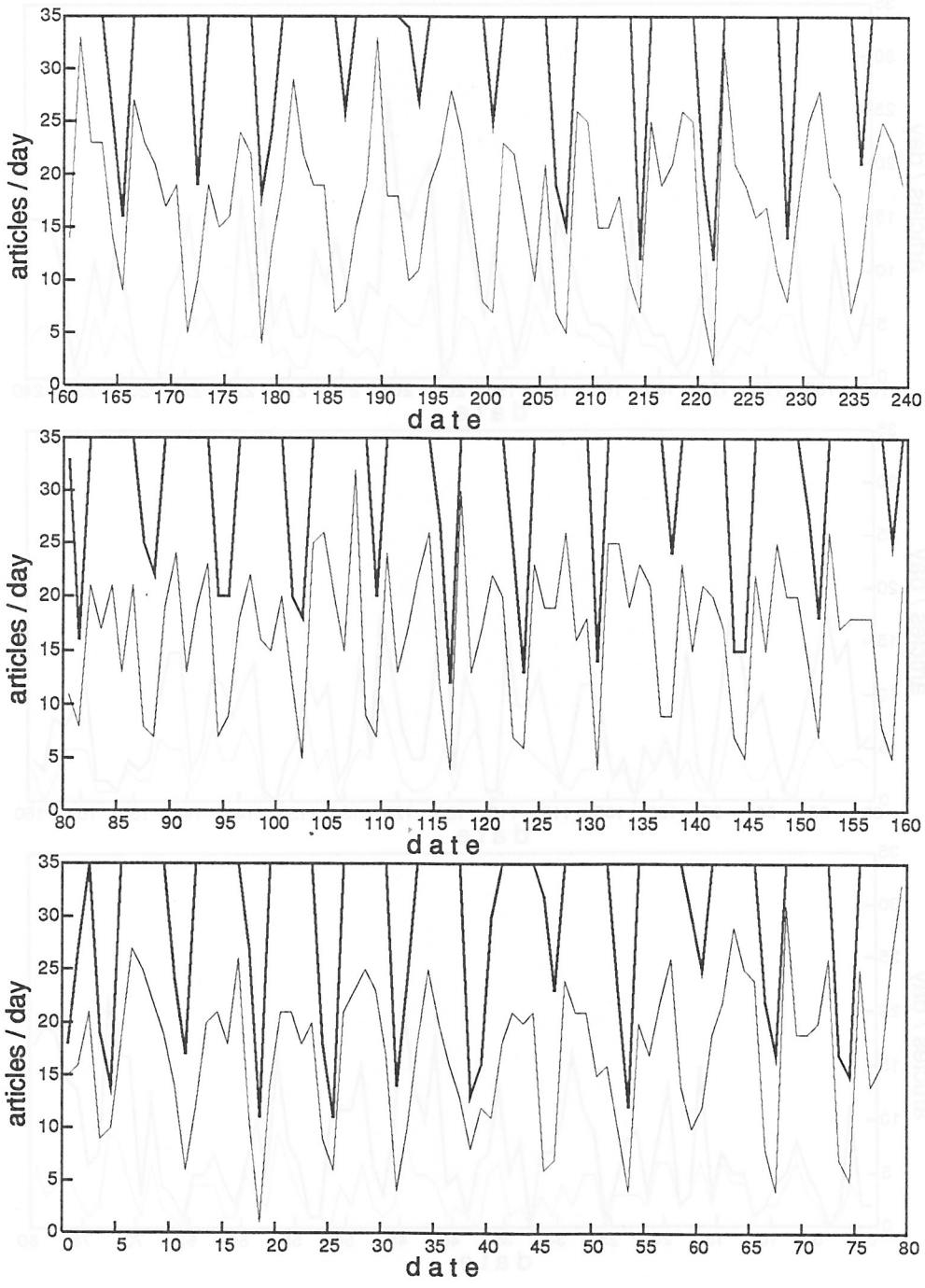

図 19: シミュレーション結果 (fj.sys.mac)

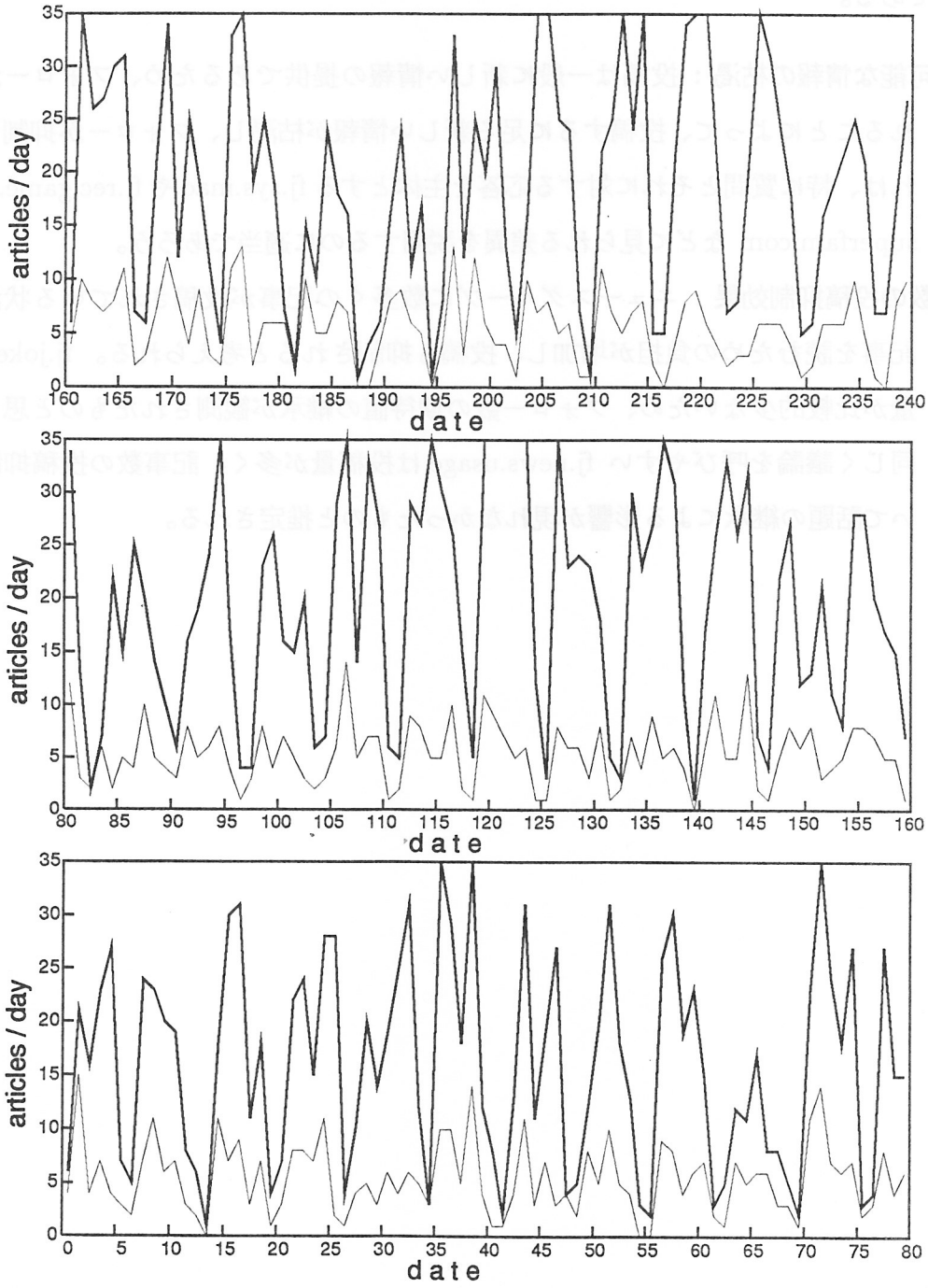

図 20: シミュレーション結果 (fj.sys.ibmpc)

散を低下させる機構として以下の二点を考えられる。これらの効果の定量的な評価は今後の課題である。

提供可能な情報の枯渇： 投稿は一般に新しい情報の提供であるため、フォローが繰り返されることによって、投稿するに足る新しい情報が枯渇し、フォローが抑制される。これは、特に質問とそれに対する応答を主体とする fj.sys.mac や fj.rec.game.video.home.-superfamicom などに見られる差異を説明するのに適当であろう。

記事数の投稿抑制効果： ニュースグループに数多くの記事が投稿されている状況下では、記事を読むための負担が増加し、投稿も抑制されると考えられる。fj.jokes.d は投稿量が比較的少ないため、フォロー数の期待値の継承が観測されたものと思われるが、同じく議論を呼びやすい fj.news.usage は投稿量が多く、記事数の投稿抑制効果が働いて話題の継承による影響が現れなかったものと推定される。

上段: 平均、下段: 分散

News Group	実測値			シミュレーション結果		
	子の数	大きさ	平均長	子の数	大きさ	平均長
fj.jokes	0.548	2.045	1.872	0.549	2.180	2.159
	0.829	5.819	3.483	0.838	7.873	5.878
fj.jokes.d	0.582	2.977	3.393	0.584	2.380	2.357
	0.731	41.31	52.04	0.754	8.494	6.444
fj.sys.mac	0.690	2.389	1.990	0.684	3.105	3.022
	1.102	9.163	3.273	1.079	31.12	18.89
fj.sys.ibmpc	0.716	2.847	2.544	0.707	3.398	3.441
	1.024	18.68	10.01	0.999	39.87	34.33
fj.books	0.569	2.006	1.978	0.571	2.327	2.312
	1.141	4.006	3.118	1.068	11.99	8.165
fj.soc.media	0.774	2.981	2.140	0.732	4.234	4.727
	0.986	19.49	2.613	0.932	53.05	53.21
fj.misc	0.709	3.294	3.469	0.685	3.194	3.256
	1.607	27.39	20.04	1.482	53.83	45.29
fj.rec.games.video.	0.762	2.940	2.622	0.760	4.236	4.353
home.superfamicom	1.053	15.57	9.349	1.032	70.57	60.14
fj.news.usage	0.820	5.416	5.647	0.832	6.437	6.738
	1.226	102.0	65.69	1.194	299.0	253.9
fj.soc.men-women	0.824	5.229	5.171	0.828	4.701	4.250
	1.265	103.6	79.86	1.270	91.02	49.03

表 10: 木構造の形状パラメータの比較

5. まとめ

コンピュータネットワーク上の社会の挙動に関する知見を得る試みの一つとして、ネットニュースにおける投稿行動の解析を行なった。主としてインターネットを介して流通しているネットニュースから、"fj" 以下のいくつかのニュースグループを選び、ここに投稿された記事を長期間にわたって収集、解析した。

記事の流通量を規定する時系列モデルとして、ニュースグループに投稿される記事が、ランダムに投稿される新規の話題の投稿と、これらに対して投稿されるフォローの投稿とで異なった挙動を示すことに着目してモデルを構築した。このモデルでは、投稿行動は、新規記事の平均投稿量、記事に対するフォロー数の分布、フォローの遅れの3つのパラメータによって表される。

記事に対するフォロー数の分布の近似関数として、ポアソン分布に比べて、負の二項分布がより良い結果を与えた。これは、それぞれの記事が異なるフォロー数の期待値を持っており、その分布が Γ 分布で近似されることを意味する。フォロー数の期待値の分布は、ニュースグループ毎に異り、それぞれのニュースグループの特性を反映しているものと考えられる。特に、Γ 分布の形状パラメータ α は、投稿記事に対するフォロー数の分布形状を決定するパラメータであり、ニュースグループにおける記事の均質性を表す。現実のニュースグループの投稿内容を観察した結果によれば、α の高さはニュースグループのまとまりの良さと対応しており、これを利用してニュースグループの状態を自動的に検出することも可能と思われる。

ネットニュースに対する投稿量の推移を分枝過程で記述することにより、投稿行動を規定する上記パラメータから、定常過程における投稿量の平均と分散を計算する手法を与えた。また、より複雑なパラメータを含むモデルを解析するための数値シミュレーションプログラムを作成した。

フォローの遅れが読者がネットニュースをチェックする間隔によって規定されるという仮説を立て、チェック間隔の分布を求めた。チェック間隔の頻度は0－1日に集中しているが、6－8日にも小さなピークがあり、一定の曜日にニュースを読む読者の存在を示すものと思われる。

投稿量の経時変化には7日を周期とする変動があり、土曜日と日曜日の投稿量が明らかな減少を示す。この減少の程度も、ニュースグループによって異なり、ニュースグループ毎

の参加者の行動様式を反映しているものと推察される。

分枝過程に基づく投稿量の計算結果は、実測値と比較して、平均値は良く一致したものの分散は小さい値を示した。これは現実の投稿量の変動に周期変動が含まれるためであると考えられる。周期変動が除外される曜日毎の実測値は計算値に近付くが、充分ではない。

周期変動を考慮したシミュレーション計算結果は、投稿量の平均、分散とも実測値と良く一致した。しかしながら、木構造の大きさの分散に関しては、計算値が小さい値を示す場合計算と、値が大きい値を示す場合とがあった。前者は、フォローを通して話題が継承されるため、フォロー数の期待値も継承されるという機構が、後者に関してはフォローが繰り返されることによる投稿すべき情報量の枯渇、記事数の増加による投稿の抑制などの機構が考えられる。これらのパラメータの効果に関する検討は今後の課題として残されている。

参考文献

[1] 堺屋太一著「組織の盛衰」 PHP 研究所 (1993)

[2] Sproull,L., Kiesler,S., "CONNECTIONS: new ways of working in the networked organization", MIT Press (1992)

[3] Paxson,V.,Floyd, S.,"Wide Area Traffic: The Failure of Poisson Modeling", *IEEE/ACM Trans. on Networking*, **3**, No.3 (June), pp.226-244 (1995)

[4] 松井啓之「阪神・淡路大震災におけるインターネットの利用」、平成 7 年度 科学研究費補助金: 総合研究 (A) 研究成果報告書「情報ネットワーク技術の動向とその社会的インパクト」 pp.23-32, (1995)

[5] 川上善郎、川浦康至、池田謙一、古川良治「電子ネットワーキングの社会心理 — コンピュータ・コミュニケーションへのパスポート」誠信書房 (1993)

[6] Bass,F.M., "A New Product Grouth for Model Consumer Durables", *Management Science*, **15**, No.5(Jan.) pp.215-227 (1969)

[7] 力武健次著「インターネットコミュニティ」オーム社開発局 (1994)

[8] 村井純、吉村伸監修「bit 別冊 インターネット参加の手引 1994 年版」共立出版 (1944)

[9] Ed Krol 著、村井純監訳「インターネットユーザーズガイド」オーム社 (1994)

[10] Mark Moraes, *news.announce.newusers*, `news:D3wtAt.3yF@deshaw.com`, "What is Usenet?" (1995)

[11] Bass, F.M. *Management Science*, **15**, No.5 (Jan.) pp.215-227 (1969)

[12] fj-committee@etl.go.jp, *fj.news.group*, `news:fjan61@cow.nara.sharp.co.jp`, "Active Newsgroups List of fj" (1996)

[13] C.Malamud 著、後藤、村上、野島訳「インターネット縦横無尽」共立出版 (1994)

[14] Ahrens, J.H., Dieter U., "Computer Methods for Sampling from Gamma, Beta, Poisson and Bonomial Distributions", *Computing*, **12**, pp.223-246 (1974)

[15] Cheng, R.C.H., Feast, G.M., "Some Simple Gamma Variate Generators", *Appl. Statist.*, **28**, pp.290-295 (1979)

[16] Ripley, B.D., *Stochastic Simulation*, Wiley, New York (1987)

A　投稿量の推移

図 21、図 23、図 24、図 25、図 26、図 27、図 28、図 29 および図 30 は、ニュースグループ fj.jokes.d、fj.sys.mac、fj.sys.ibmpc、fj.books、fj.soc.media、fj.misc、fj.rec.games.video.home.superfamicom、fj.news.usage、fj.soc.men-women の各々に投稿された記事の一日あたりの投稿量の経時変化である。それぞれの図において、上側の折れ線は全投稿量の推移を、下側の折れ線は新規話題の投稿量の推移を示す。フォローの投稿量の推移は、これら二つの折れ線の間の幅で読み取ることができる。

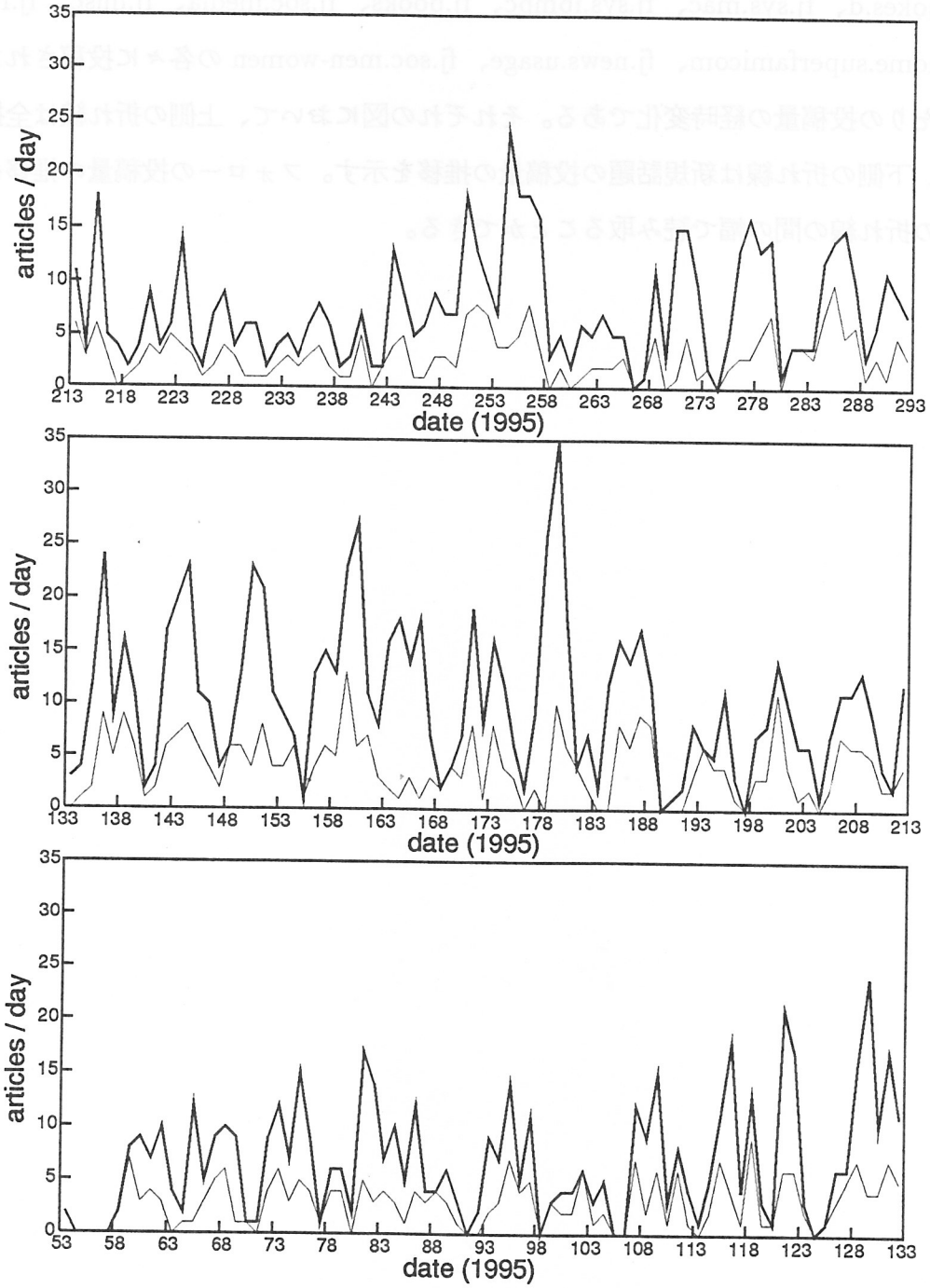

図 21: fj.jokes.d の投稿量の経時変化 (その 1)

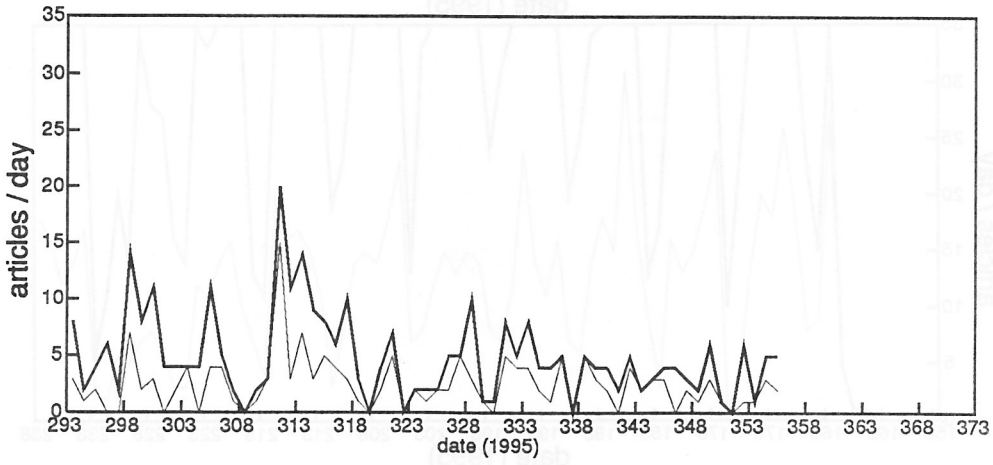

図 22: fj.jokes.d の投稿量の経時変化 (その 2)

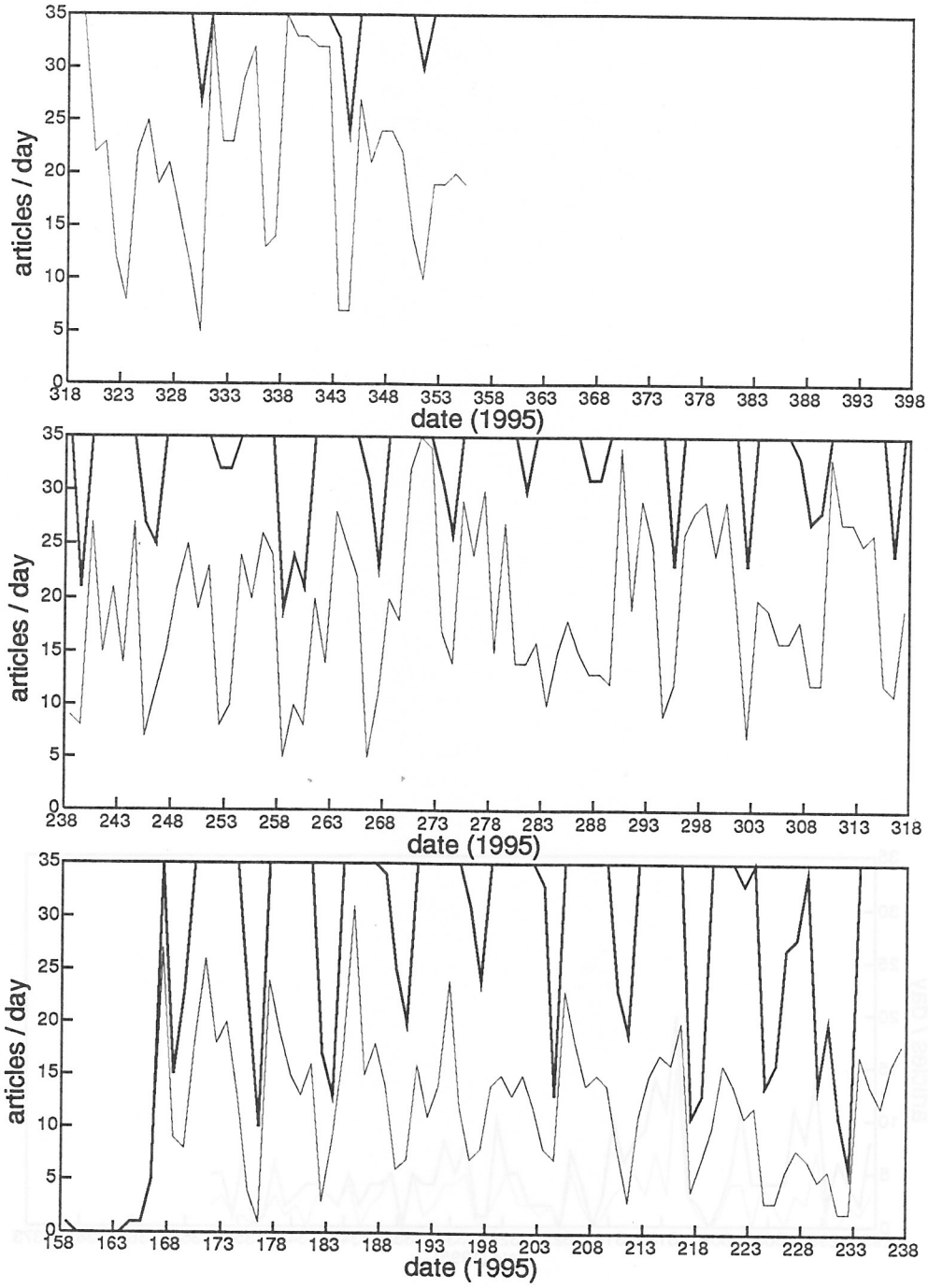

図 23: fj.sys.mac の投稿量の経時変化

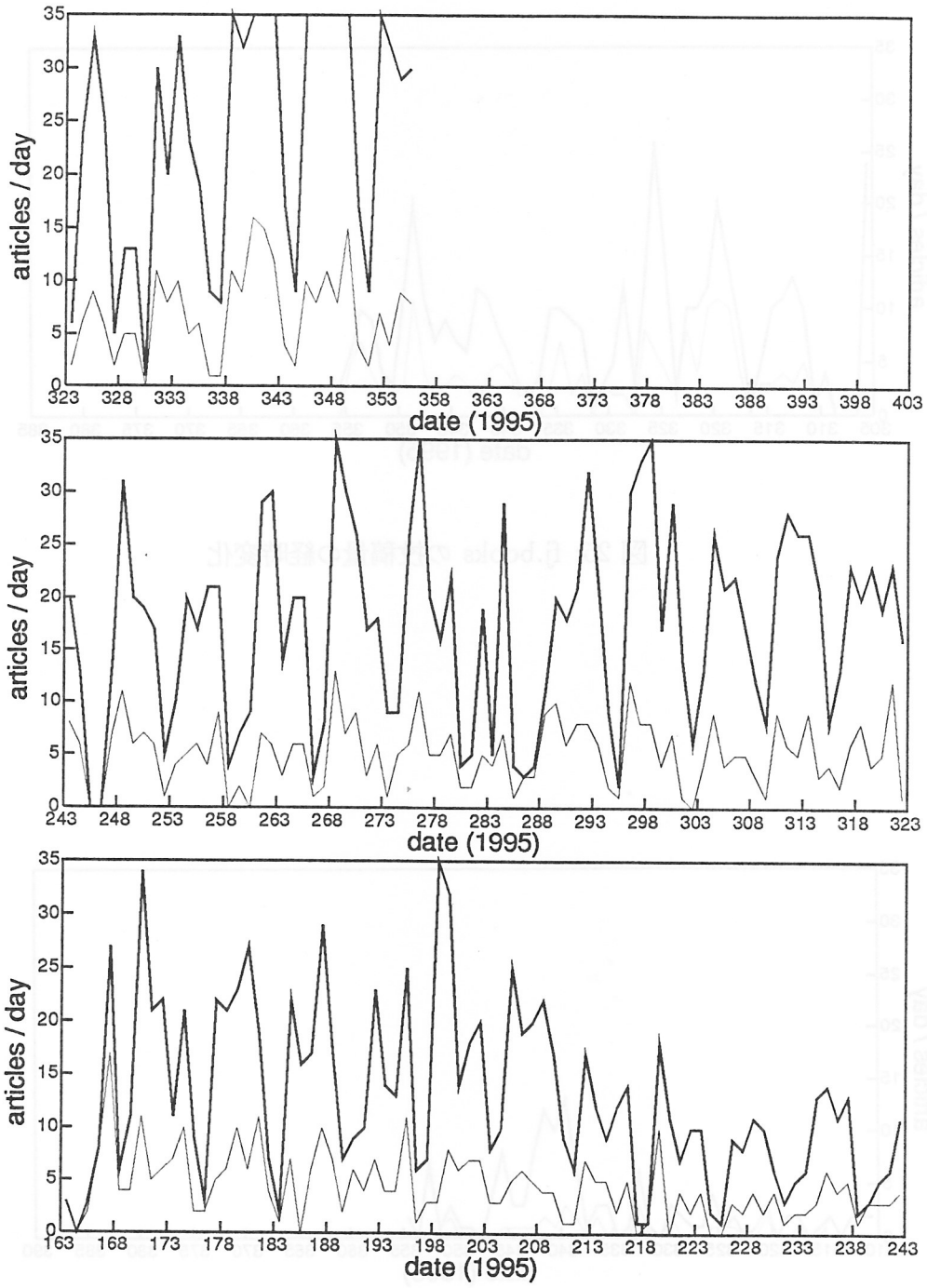

図 24: fj.sys.ibmpc の投稿量の経時変化

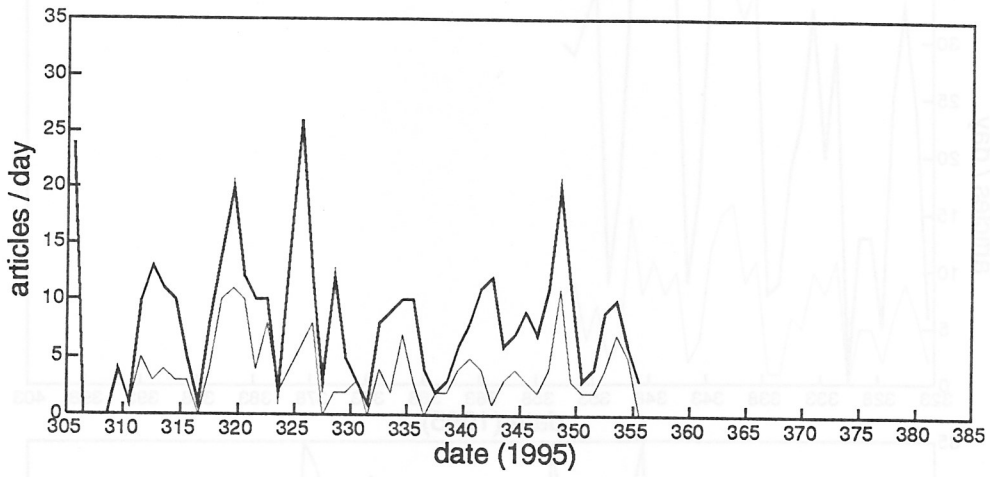

図 25: fj.books の投稿量の経時変化

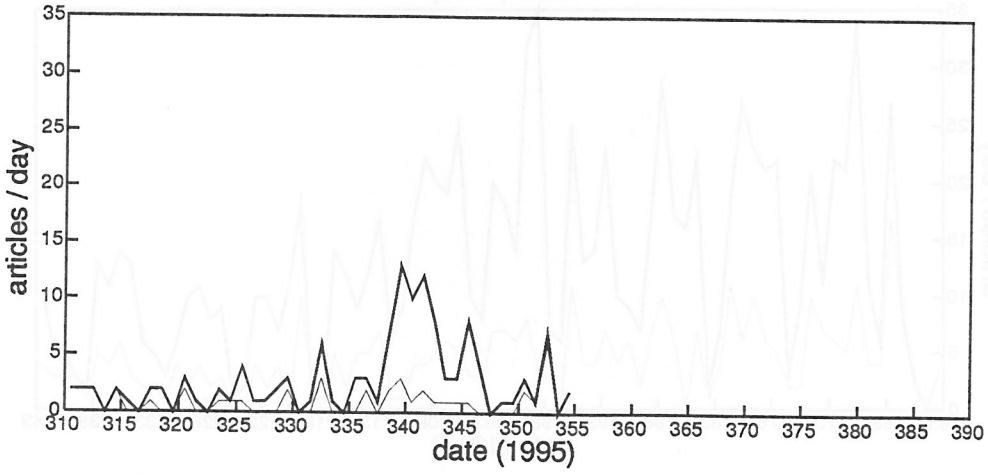

図 26: fj.soc.media の投稿量の経時変化

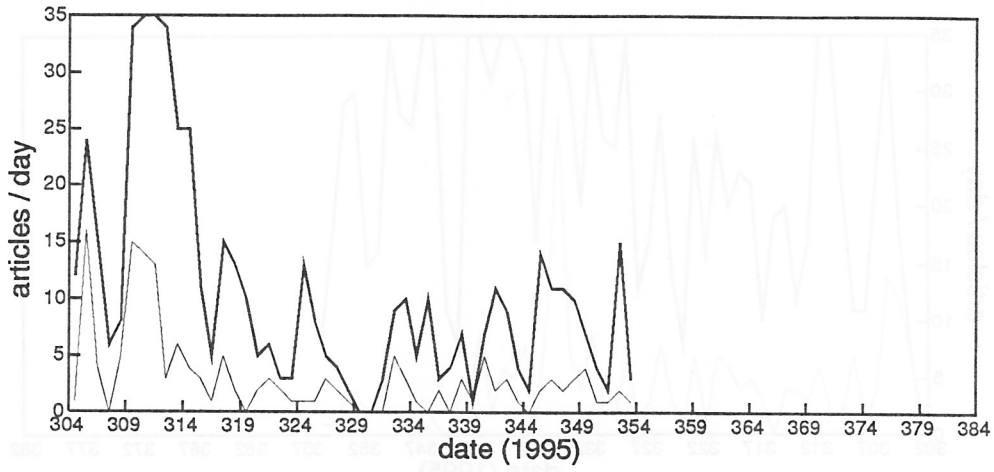

図 27: fj.misc の投稿量の経時変化

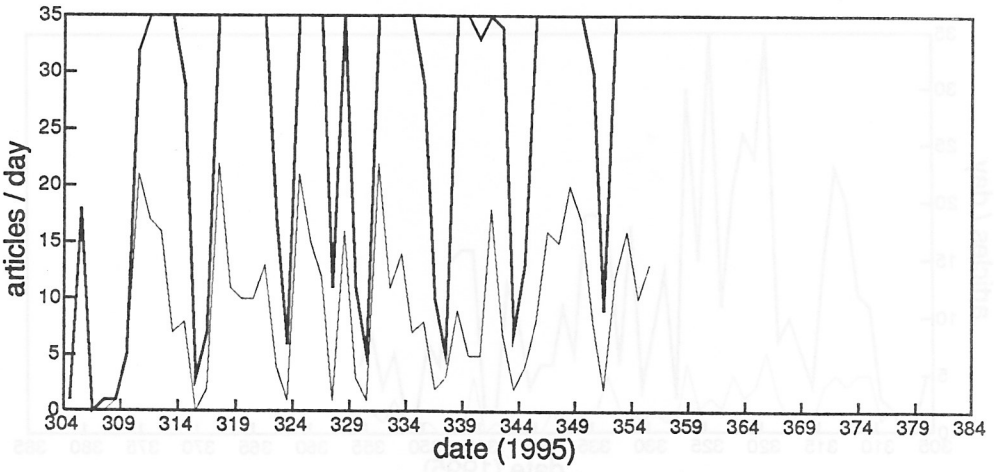

図 28: fj.rec.games.video.home.superfamicom の投稿量の経時変化

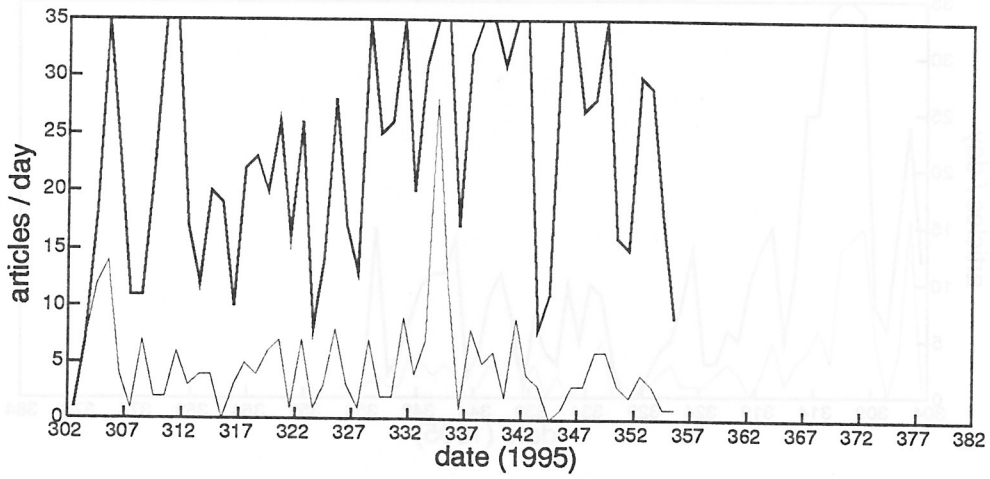

図 29: fj.news.usage の投稿量の経時変化

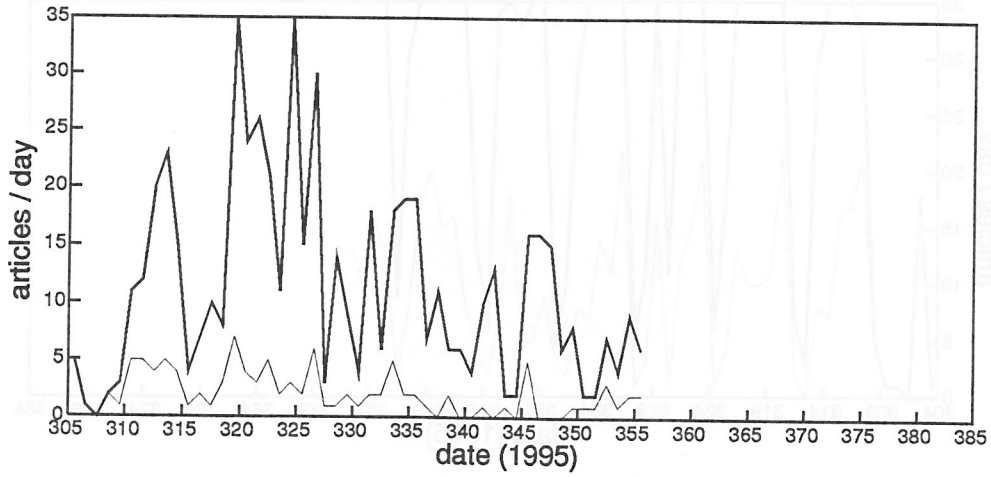

図 30: fj.soc.men-women の投稿量の経時変化

B　プログラムリスト

B1.　ヘッダー情報の抽出

ニュースリーダは記事をそれぞれ個別のファイルに格納する。ヘッダー情報抽出プログラムは、単一の記事を収めたファイルを標準入力から受けとり、ヘッダー情報を標準出力に書き出す。以下のシェルスクリプトで、特定のニュースグループの記事を収めたディレクトリのファイルを次々と処理して、全ての記事のヘッダー情報を一つのファイルに書き出す。

```
if [ -e $1.hed ]
        then rm $1.hed
        fi
for f in $1/*
        do  ./header $f >> $1.hed
        done
```

以下はヘッダー情報抽出プログラムである。ネットニュースの記事から Subject、From、Message-ID、References、Date、Lines を抽出する。

```
#include <stdio.h>
#define NH 8
#define Ref 5
#define MID 4
#define Cat 3
#define Sit 2
#define Frm 1
#define LL 256
unsigned char *headers[NH] ={
  "Subject: ",
  "From: ",
  "Site: ",
  "Category: ",
  "Message-ID: ",
  "References: ",
  "Date: ",
  "Lines: "};
unsigned char line[LL];
unsigned char data[NH][LL];
unsigned long hash(unsigned char *s){
  union {
    unsigned long lword;
    unsigned int word[2];
    unsigned char byte[4];} cell;
  unsigned long sum;
  int i, j;
  unsigned char c;
  for(sum = i = 0; s[i]; i++){
    j = (i * 9) % 32;
    cell.lword = 0;
    cell.byte[0] = s[i];
    if(j < 24) cell.lword <<= j;
    else{
      cell.lword <<= (j - 8);
      c = cell.byte[3];
      cell.byte[3] = cell.byte[2];
      cell.byte[2] = cell.byte[1];
```

```
      cell.byte[1] = cell.byte[0];
      cell.byte[0] = c;}
    sum += cell.lword;}
  return(sum);}
void refcont(void){
  unsigned char *pline;
  for(pline = line; pline; pline++) if(*pline != ' ') break;
  strcpy(data[Ref], pline);}
void refchk(void){
  unsigned char buf[LL], *dline, *pline = buf;
  strncpy(buf, data[Ref], LL);
  for(dline = data[Ref]; *pline; pline++){
    switch(*pline){
    case ' ':
    case ',':
    case '>': break;
    case '<': dline = data[Ref]; break;
    default: *(dline++) = *pline;}}
  *dline = '\0';
  strncpy(buf, data[MID], LL);
  for(dline = data[MID], pline = buf; *pline; pline++){
    switch(*pline){
    case ' ':
    case ',':
    case '>': break;
    case '<': dline = data[MID]; break;
    default: *(dline++) = *pline;}}
  *dline = '\0';}
fromcut2(){
  char *p, token[10][LL];
  int it, i;
  for(p = data[Frm], i = 0; (*p != '@') && *p; p++){
    if((*p == ' ') || (*p == ')') || (*p == '>') || (*p == '<'))
      i = 0;
    else token[0][i++] = tolower(*p);}
  token[0][i] = '\0';
  if(*p) p++;
  for(i = 0, it = 1; *p; p++){
    if(*p == '.'){
      token[it++][i] = '\0'; i = 0;}
    else if ((*p == ' ') ||(*p == '(') || (*p == '<') || (*p == '>'))
      break;
    else token[it][i++] = tolower(*p);}
  if(i) token[it][i] = '\0';
  else it--;
  if(!strcmp(token[it], "jp")){
    if(it >2) strncpy(data[Sit], token[it-2], LL-1);
    else data[Sit][0] = '\0';
    if(it > 1) strncpy(data[Cat], token[it-1], LL-1);
    else data[Cat][0] = '\0';}
  else{
    if(it > 1) strncpy(data[Sit], token[it-1], LL-1);
    else data[Sit][0] = '\0';
    if(it > 0) strncpy(data[Cat], token[it], LL-1);
    else data[Cat][0] = '\0';
    fprintf(stderr, "check! From: %s\n", data[Frm]);
    fprintf(stderr, "user = %s, site = %s, cat = %s\n",
            token[0], data[Sit], data[Cat]);}
  strncpy(data[Frm], token[0], LL-1);}
fromcut(){
  char *p, token[10][LL];
```

```
    int it, i;
    for(p = data[Frm], it = i = 0;
        (*p != '=') && (*p != ' ') && *p && (i < LL) &&
        (*p != '>') && (*p != '('); p++){
      if((*p == '.') || (*p == '@')){
        if(it++ < 10){
          token[it-1][i] = '\0';
          i = 0;}
        else{
          fprintf(stderr, "From error. From : %s\n", data[Frm]);
          exit(2);}}
      else token[it][i++] = tolower(*p);}
    token[it][i] = '\0';
    if(!token[it][0]) it--;
    if(it < 3){
      for(p = data[Frm], it = i = 0; i < LL; p++){
        if((*p == ' ') || (*p == '<')) it = i = 0;
        else if((*p == '@') || (*p == '.')){
          token[it++][i] = '\0'; i = 0;}
        else if((*p == '>') || (*p == '\0') || (*p == '(')){
          token[it][i] = '\0'; break;}
        else token[it][i++] = tolower(*p);}}
    if(!strcmp(token[it], "jp")){
      strncpy(data[Sit], token[it-2], LL-1);
      strncpy(data[Cat], token[it-1], LL-1);}
    else{
      strncpy(data[Sit], token[it-1], LL-1);
      strncpy(data[Cat], token[it], LL-1);
      fprintf(stderr, "check! From: %s\n", data[Frm]);
      fprintf(stderr, "user = %s, site = %s, cat = %s\n",
              token[0], data[Sit], data[Cat]);}
    strncpy(data[Frm], token[0], LL-1);}
main(int argc, char *argv[]){
    int i, id;
    FILE *fp, *fopen();
    if(!(fp = fopen(argv[1], "r"))){
      fprintf(stderr, "file %s cannot open\n", argv[1]);
      exit(3);}
    while(fgets(line, LL-1, fp)){
      *(line + strlen(line)-1) = '\0';
      if(line[strlen(line) -1] == '\r') line[strlen(line) -1] = '\0';
      if(strlen(line) < 2) break;
      for(i = 0; i < strlen(line); i++){
        if(line[i] == '\t') line[i] = ' ';}
      for(i = 0; i < NH; i++){
        if((i == Sit) || (i == Cat)) continue;
        if(!strncmp(line, headers[i], strlen(headers[i]))){
          id = i;
          strcpy(data[i], line + strlen(headers[i]));
          break;}}
      if((i >= NH) && (id == Ref) && (line[0] == ' ')) refcont();
      else if ((i >= NH) && (id == Frm) && (line[0] == ' ')){
        strcpy(data[Frm], line);}}
    refchk();
    fromcut2();
    sprintf(data[Ref], "%08lx", hash(data[Ref]));
    sprintf(data[MID], "%08lx", hash(data[MID]));
    printf("0 %s\n", argv[1]);
    for(i = 0; i < NH; i++) printf("%1d %s\n", i + 1, data[i]);}
```

B2. 参照関係のセット

前節のプログラムで生成されたヘッダー情報から、記事の参照関係を求め、子の数、部分木の大きさを求める。また、日付を 1995 年 1 月 1 日からの経過日数に換算し、フォローの遅れ日数を求める。

```
#include <stdio.h>
#include <string.h>
#define NF 20000
#define LL 256
#define FNO 0
#define SBJ 1
#define POS 2
#define SIT 3
#define CAT 4
#define MID 5
#define REF 6
#define DAT 7
#define LIN 8
char line[LL], *lp, poster[NF][9], site[NF][13], cat[NF][4];
unsigned long mid[NF], ref[NF];
int lines[NF], reffile[NF], refcnt[NF], allcnt[NF],
  dat[NF], tim[NF], delay[NF], fno;
FILE *fp1, *fp2, *fopen();
char monnam[12][4]={"Jan", "Feb", "Mar", "Apr",
                    "May", "Jun", "Jul", "Aug",
                    "Sep", "Oct", "Nov", "Dec"};
int monday[12]={   0,  31,  59,  90, 120, 151,
                 181, 212, 243, 273, 304, 334};
main(int argc, char *argv[]){
  int i, lineID, day, year, hour, min, sec;
  char month[4];
  if(argc < 2){
    fprintf(stderr, "Usage : \% refset <infile> <outfile>\n");
    exit(3);}
  if(!(fp1 = fopen(argv[1], "r"))){
    fprintf(stderr, "file %s cannot open\n", argv[1]);
    exit(2);}
  if(!(fp2 = fopen(argv[2], "w"))){
    fprintf(stderr, "file %s cannot open\n", argv[2]);
    exit(2);}
  fno = 0;
  while(fgets(line, LL - 1, fp1)){
    line[strlen(line) - 1] = '\0';
    if(line[strlen(line) - 1] == '\r')
      line[strlen(line) - 1] = '\0';
    sscanf(line, "%1d", &lineID);
    /*  printf("lineID : %1d, line : %s\n", lineID, line); */
    switch(lineID){
    case FNO: fno++;/*sscanf(line + 2, "/%d", &fno);*/
      break;
    case MID: sscanf(line + 2, "%lx", &mid[fno]);
      break;
    case REF: sscanf(line + 2, "%lx", &ref[fno]);
      break;
    case POS: strncpy(poster[fno], line + 2, 8);
      poster[fno][8] = '\0';
      break;
    case SIT: strncpy(site[fno], line + 2, 12);
      site[fno][12] = '\0';
      break;
    case CAT: strncpy(cat[fno], line + 2, 3);
```

```
      cat[fno][3] = '\0';
      break;
    case DAT:
      printf("[%d] : %s\n", fno, line);
      for(lp = line + 2; ; lp++){
        if((*lp >= '0') && (*lp <= '9')) break;}
      printf("[%d] : %s\n", fno, lp);
      sscanf(lp, "%d %3s %d %d:%d:%d",
             &day, month, &year, &hour, &min, &sec);
      printf("day : %d, month : %s, year : %d, ", day, month, year);
      printf("hour : %d, min : %d, sec : %d\n", hour, min, sec);
      for(i = 0; i <12; i++){
        if(strcmp(month, monnam[i]) == 0){
          dat[fno] = day + monday[i];
          break;}}
      if(i >= 12){
        fprintf(stderr, "error %s is not month name!\n",
                month);
        fprintf(stderr, "line : %s\n", line);
        fprintf(stderr, "  lp : %s\n", lp);}
      tim[fno] = 10 * (60 * hour + min) + sec/6;
      break;
    case LIN: sscanf(line + 2, "%d", &lines[fno]);
      break;}}
  for(fno = 0; fno <NF; fno++){
    if(ref[fno]) for(i = 0; i < NF; i++){
      if(ref[fno] == mid[i]){
        reffile[fno] = i;
        refcnt[i]++;
        delay[fno] = dat[fno] - dat[i];
        break;}}}
  for(i = 0; i < NF; i++){
    for(fno = i; reffile[fno]; fno = reffile[fno]){
      allcnt[reffile[fno]]++;}}
  for(i = 0; i < NF; i++) if(poster[i][0]){
    fprintf(fp2, "%5d %5d %5d %5d %5d %5d %7d %5d %s %s %s\n",
            i, refcnt[i], allcnt[i], reffile[i], delay[i],
            dat[i], tim[i], lines[i], poster[i], site[i],cat[i]);}}
```

B3. 作図用関数

以下は、各種グラフを ps 形式で作図するための関数群とヘッダーファイルである。

ヘッダーファイル "psplot.h"

```
FILE * plots(char * file, int left, int bottom, int right, int top);
/* initialize psplot for file and return file pointer (fpp) */
/*  plotting area determind by left, bottom, right and top  */
void width(FILE * fpp, double w);
/* set line width to w points */
void plot(FILE * fpp, double x, double y, int iplot);
/* line(iplot=1) or move(iplot=0) to (x, y) by unit of cm */
void psputs(FILE * fpp, double x, double y, double deg, int pt, char * s);
/* put string s from (x, y) to deg direction by point pt */
void psputi(FILE * fpp, double x, double y, double deg, int pt,
            int c, int i);
/* put integer i by psputs, adjusting center(c='c') or right(c='r') */
```

```
void psputd(FILE * fpp, double x, double y, double deg, int pt,
            int c, char *fmt, double d);
/* put double d by format fmt, other parameters are same as psputi */
void plote(FILE * fpp);
/* end psplot and close file */
```

ps 作図関数 psplot.c

```
#include <stdio.h>
FILE * plots(char * file, int left, int bottom, int right, int top){
  FILE * fpp;
  if((fpp = fopen(file, "w")) == NULL) return NULL;
  printf("\nplot %s\n", file);
  fprintf(fpp, "%%!PS-Adobe-2.0 EPSF-1.2\n");
  fprintf(fpp, "%%%%Pages: 1\n");
  fprintf(fpp, "%%%%BoundingBox: %d %d %d %d\n",
          left, bottom, right, top);
  fprintf(fpp, "%%%%DocumentFonts: Helvetica\n");
  fprintf(fpp, "%%%%EndComments\n");
  fprintf(fpp, "/hv10 /Helvetica findfont 10 scalefont def\n");
  fprintf(fpp, "/hv12 /Helvetica findfont 12 scalefont def\n");
  fprintf(fpp, "/hv14 /Helvetica findfont 14 scalefont def\n");
  fprintf(fpp, "/hv16 /Helvetica findfont 16 scalefont def\n");
  fprintf(fpp, "/hv18 /Helvetica findfont 18 scalefont def\n");
  fprintf(fpp, "/hv20 /Helvetica findfont 20 scalefont def\n");
  fprintf(fpp, "/hv22 /Helvetica findfont 22 scalefont def\n");
  fprintf(fpp, "/hv24 /Helvetica findfont 24 scalefont def\n");
  fprintf(fpp, "newpath\n");
  fprintf(fpp, "0 setlinewidth\n");
  return fpp;}
void width(FILE * fpp, double w){
  fprintf(fpp, "closepath\n stroke\nnewpath\n%f setlinewidth\n", w);}
void plot(FILE * fpp, double x, double y, int iplot){
  /* plot line by unit of cm */
  static double point = 72 / 2.54;
  /*printf("plot %f %f %d\n", x, y, iplot);*/
  if(iplot) fprintf(fpp, "%f %f lineto\n", x * point, y * point);
  else      fprintf(fpp, "%f %f moveto\n", x * point, y * point);}
void psputs(FILE * fpp, double x, double y, double deg, int pt, char * s){
  static double point = 72.0 / 2.54;
  static char hvfonts[8][5] = {"hv10", "hv12", "hv14", "hv16",
                               "hv18", "hv20", "hv22", "hv24"};
  if(pt < 10) pt = 10; else if(pt > 24) pt = 24;
  pt = (pt - 10) / 2;
  fprintf(fpp, "%s setfont %f %f moveto %f rotate (%s) show %f rotate\n",
          hvfonts[pt], x * point, y * point, deg, s, -deg);}
void psputi(FILE * fpp, double x, double y, double deg, int pt,
            int c, int i){
  static double point = 72 / 2.54;
  char s[256];
  sprintf(s, "%d", i);
  switch (c){
  case 'c': x -= (5 / point) * strlen(s); break;
  case 'r': x -= (10 / point) * strlen(s); break;}
  psputs(fpp, x, y, deg, pt, s);}
void psputd(FILE * fpp, double x, double y, double deg, int pt,
            int c, char *fmt, double d){
```

```
  static double point = 72 / 2.54;
  char s[256];
  sprintf(s, fmt, d);
  switch (c){
  case 'c': x -= (5 / point) * strlen(s); break;
  case 'r': x -= (10 / point) * strlen(s); break;}
  psputs(fpp, x, y, deg, pt, s);}
void plote(FILE * fpp){
  fprintf(fpp, "closepath\n");
  fprintf(fpp, "stroke\n");
  fprintf(fpp, "showpage\n\n");
  fprintf(fpp, "%%%%Trailer\n\n");
  fprintf(fpp, "%%%%EOF\n");
  fclose(fpp);}
```

B4. 統計処理プログラム

このプログラムは、前節のプログラムで生成されたファイルを処理して、ニュースグループの各種統計的パラメータを求める。また、各種グラフを ps 形式で作成すると共に、次節のシミュレーションプログラムのためのデータファイルを作成する。

```
#include <stdio.h>
#include <math.h>
#include "psplot.h"
#define MAX 100
#define LINE 80
#define DAY 365
#define PMAX 100

int nref[MAX], nref2[MAX], ndat, kmax, startdat, enddat;
double pobs[MAX], pest[MAX], pest2[MAX], nest[MAX], nest2[MAX], q2, lambda;
symbol(FILE * fpp, double s, int is){
  s *= 72 / 2.54;
  switch (is % 3 + 1){
  case 1:{ /* + */
    fprintf(fpp, "0 %f rlineto\n", s);
    fprintf(fpp, "0 %f rlineto\n", -2.0 * s);
    fprintf(fpp, "0 %f rlineto\n", s);
    fprintf(fpp, "%f 0 rlineto\n", s);
    fprintf(fpp, "%f 0 rlineto\n", -2.0 * s);
    fprintf(fpp, "%f 0 rlineto\n", s);
    break;}
  case 2:{ /* X */
    fprintf(fpp, "%f %f rlineto\n", 0.7 * s, 0.7 * s);
    fprintf(fpp, "%f %f rlineto\n", -1.4 * s, -1.4 * s);
    fprintf(fpp, "%f %f rlineto\n", 0.7 * s, 0.7 * s);
    fprintf(fpp, "%f %f rlineto\n", 0.7 * s, -0.7 * s);
    fprintf(fpp, "%f %f rlineto\n", -1.4 * s, 1.4 * s);
    fprintf(fpp, "%f %f rlineto\n", 0.7 * s, -0.7 * s);
    break;}
  case 3:{ /* [] */
    fprintf(fpp, "%f %f rmoveto\n", 0.7 * s, 0.7 * s);
    fprintf(fpp, "%f 0 rlineto\n", -1.4 * s);
    fprintf(fpp, "0 %f rlineto\n", -1.4 * s);
    fprintf(fpp, "%f 0 rlineto\n", 1.4 * s);
    fprintf(fpp, "0 %f rlineto\n", 1.4 * s);
    fprintf(fpp, "%f %f rmoveto\n", -0.7 * s, -0.7 * s);
    break;}}}
```

```
double power(double x, int i){
  double r;
  if(i == 0) return 1.0L;
  if(i > 0) for(r = x; --i; ) r *= x;
  else for(r = 1 / x; ++i; ) r /= x;
  return r;}
void week(int n, int* x, double ave[]){
  int whist[7], i, total, wn[7];
  double sd[7];
  for(i = 0; i < 7; i++){
    ave[i] = sd[i] = wn[i] = whist[i] = 0;}
  for(i = total = 0; i < n; i++){
    whist[i % 7] += x[i];
    total += x[i];
    wn[i % 7]++;
    ave[i % 7] += x[i];
    sd[i % 7] += x[i] * x[i];}
  for(i = 0; i < 7; i++){
    ave[i] /= wn[i];
    sd[i] = sd[i] / wn[i] - ave[i] * ave[i];}
  printf("day +-distribution-+  +------- post/day -------+\n");
  printf("       total      %%    ave    var   s.d. sd/ave\n");
  for(i = 0; i < 7; i++){
    printf("%3d  %8d %6.2f %6.2f %6.2f %6.2f %6.3f\n",
           i, whist[i], (100.0 * whist[i]) / total, ave[i], sd[i],
           sqrt(sd[i]), sqrt(sd[i]) / ave[i]);}}
void estpoi(double * nest2, int * nref2, int kmax){
  int i, n;
  double t1, t2, p, factk;
  for(t1 = t2 = i = 0; i < kmax; i++){
    t1 += nref2[i];
    t2 += nref2[i] * i;}
  p = t2 / t1;
  for(i = 0; i < kmax; i++){
    if(!i) factk = 1; else factk *= i;
    pest2[i] = exp(-p) * power(p, i) / factk;
    nest2[i] = pest2[i] * t1;}}
void kgraph(FILE * fpp, int mind, int maxd, int dd, int n,
            int dn, double x0, double y0, double wx, double wy,
            int d0, int d1){
  double x, y;
  int i, d;
  width(fpp, 1.0);
  plot(fpp, x0, y0, 0);
  plot(fpp, x0 + wx, y0, 1);
  plot(fpp, x0 + wx, y0 + wy, 1);
  plot(fpp, x0, y0 + wy, 1);
  plot(fpp, x0, y0, 1);
  plot(fpp, x0, y0, 0);
  for(i = mind; i < maxd; i += dd){
    y = y0 + wy * (i - (double)mind) / (maxd - (double)mind);
    plot(fpp, x0, y, 0);
    plot(fpp, x0 + 0.2, y, 1);
    plot(fpp, x0 + 0.2, y, 0);}
  for(i = mind; i < maxd + dd / 2; i += dd){
    y = y0 + wy * (i - (double)mind) / (maxd - (double)mind);
    psputi(fpp, x0 - 0.1, y - 0.1, 0., 10, 'c', i);}
  for(i = 0; i <= n; i += dn){
    x = x0 + wx * i / (double)n;
    plot(fpp, x, y0, 0);
```

```
    plot(fpp, x, y0 + 0.2, 1);
    plot(fpp, x, y0 + 0.2, 0);}
  for(i = 0; i <= n; i += dn){
    x = x0 + wx * i / (double)n;
    d = d0 + i * (d1 - d0) / n;
    psputi(fpp, x + 0.2, y0 - 0.35, 0., 10, 'c', d);}}
void kgraph2(FILE * fpp, double mind, double maxd, double dd, int n,
             int dn, double x0, double y0, double wx, double wy,
             int d0, int d1){
  double x, y, i, d;
  width(fpp, 1.0);
  plot(fpp, x0, y0, 0);
  plot(fpp, x0 + wx, y0, 1);
  plot(fpp, x0 + wx, y0 + wy, 1);
  plot(fpp, x0, y0 + wy, 1);
  plot(fpp, x0, y0, 1);
  plot(fpp, x0, y0, 0);
  for(i = mind; i < maxd; i += dd){
    y = y0 + wy * (i - mind) / (maxd - mind);
    plot(fpp, x0, y, 0);
    plot(fpp, x0 + 0.2, y, 1);
    plot(fpp, x0 + 0.2, y, 0);}
  for(i = mind; i < maxd + dd / 2; i += dd){
    y = y0 + wy * (i - mind) / (maxd - mind);
    psputd(fpp, x0 - 0.1, y - 0.1, 0., 10, 'c', "%3.1f", i);}
  for(i = 0; i <= n; i += dn){
    x = x0 + wx * i / (double)n;
    plot(fpp, x, y0, 0);
    plot(fpp, x, y0 + 0.2, 1);
    plot(fpp, x, y0 + 0.2, 0);}
  for(i = 0; i <= n; i += dn){
    x = x0 + wx * i / (double)n;
    d = d0 + i * (d1 - d0) / n;
    psputi(fpp, x + 0.2, y0 - 0.35, 0., 10, 'c', d);}}
void igraph(FILE * fpp, int * dat, int * dend, int mind, int maxd, double dy,
            int n, double x0, double y0, double wx, double wy, double wl){
  double x, y;
  int i;
  width(fpp, wl);
  for(i = 0; (i < n) && (dat < dend); i++, dat++){
    y = y0 + wy * (*dat - (double)mind) / (maxd - (double)mind);
    if(y < y0) y = y0; else if(y > y0 + wy) y = y0 + wy;
    x = x0 + wx * (i + 0.5) / n;
    plot(fpp, x, y, i);}
  plot(fpp, x, y, 0);}
void igraph2(FILE * fpp, int * dat, int * dend, int mind, int maxd, double dy,
             int n, double x0, double y0, double wx, double wy, double wl){
  double x, y;
  int i;
  width(fpp, wl);
  for(i = 0; (i <= n) && (dat < dend); i++, dat++){
    y = y0 + wy * (*dat - (double)mind) / (maxd - (double)mind);
    if(y < y0) y = y0; else if(y > y0 + wy) y = y0 + wy;
    x = x0 + wx * i / n;
    plot(fpp, x, y, i);}
  plot(fpp, x, y, 0);}
void igraph3(FILE * fpp, double * dat, double * dend, double mind,
             double maxd, double dy, int n, double x0, double y0,
             double wx, double wy, double wl){
```

```
  double x, y;
  int i;
  width(fpp, wl);
  for(i = 0; (i <= n) && (dat < dend); i++, dat++){
    y = y0 + wy * (*dat - mind) / (maxd - mind);
    if(y < y0) y = y0; else if(y > y0 + wy) y = y0 + wy;
    x = x0 + wx * i / n;
    plot(fpp, x, y, i);}
  plot(fpp, x, y, 0);}
double gamma_ratio(double alpha, int k){
  /* return Gamma(alpha + k) / Gamma(alpha)
     = (alpha + k - 1) * (alpha + k -2) * ... * (alpha) */
  int i;
  double r;
  if(k < 0) printf("gamma_ratio called by k < 0, result set to 1\n");
  if(k > 0) for(i = 1, r = alpha; i < k; i++) r *= alpha + i;
  else r = 1.0;
  return r;}
void delayx(int * ndelay, int dmax, double pweek[]){
  int i, j, k, n, dlmt;
  double work[18];
  double pdelay[MAX], pterm[MAX], qterm[MAX], pdelays[MAX], qdelay[MAX];
  double q, ave, var, alpha, theta, fact_k, x, y, x0, y0;
  FILE *fp, *fpp;
  for(i = n = 0; i <= dmax; i++) n += ndelay[i];
  for(q2 = i = 0; i <= dmax; i++){
    q = (double)ndelay[i] / n;
    q2 += q * q;}
  printf("Sigma Q^2 = %f\n", q2);
  fp = fopen("simvars.dat", "w");
  for(i = 0; i < 7; i++) fprintf(fp, "%f\n", pweek[i]);
  for(i = 0; i <= dmax; i++){
    q = (double)ndelay[i] / n;
    fprintf(fp, "%f\n", q);}
  fclose(fp);
  for(i = n = 0; i <= dmax; i++){
    ave += ndelay[i] * i;
    n += ndelay[i];}
  ave /= n;
  for(i = var = 0; i < MAX; i++) var += ndelay[i] * (i - ave) * (i - ave);
  var /= n;
  for(i = 1; i <= dmax; i++) ndelay[i - 1] += (ndelay[i] /= 2);
  ndelay[dmax -1] += ndelay[dmax];
  for(i = 0; i < dmax; i++){
    pdelay[i] = qdelay[i] = (double)ndelay[i] / n;}
  for(i = dmax - 1; i >= 0; i--){
    qterm[i] = pdelay[i] * (i + 1);
    for(j = 0; j < i; j++) pdelay[j] -= pdelay[i];}
  printf("\ndelay analysis :  ave = %lf, var = %lf\n", ave, var);
  theta = ave / (var - ave);
  alpha = ave * theta;
  printf("estimated : alpha = %lf, theta = %lf\n\n", alpha, theta);
  printf("estimated distribution\n\n    k : neg.bin., observed\n");
  for(k = 0; k < dmax; k++){
    for(i = fact_k = 1; i <= k; i++) fact_k *= i;
    pdelay[k] = pdelays[k] = (gamma_ratio(alpha, k) / fact_k) *
      pow((1 + theta)/theta, -alpha - k) / power(theta, k);
    printf("%5d : %6.2lf, %6.2lf\n",
```

```
            k, pdelay[k] * 100.0, 100.0 * (double)ndelay[k] / n);}
    for(i = dmax - 1; i >= 0; i--){
      pterm[i] = pdelay[i] * (i + 1);
      for(j = 0; j < i; j++) pdelay[j] -= pdelay[i];}
    printf("\n check interval distribution \n");
    printf("day delay1 check1 delay2 check2\n");
    printf("          %c       %c       %c       %c \n", '%', '%', '%', '%');
    for(i = 0; i < dmax; i++){
      printf("%3d  %6.3f %6.3f %6.3f %6.3f\n",
            i, pdelays[i] * 100, pterm[i] * 100,
            qdelay[i] * 100, qterm[i] * 100);}
    dlmt = (dmax < 16) ? dmax : 16;
    fpp = plots("delay.obj", -20, -20, 520, 350);
    x0 = 1.5, y0 = 1.5;
    kgraph2(fpp, 0.0, 0.5, 0.1, 16, 1, x0, y0, 16.0, 8.0, 0, 16);
    psputs(fpp, x0 + 6.5, y0 - 0.9, 0., 14, "delay / days");
    psputs(fpp, x0 - 0.8, y0 + 2.0, 90., 14, "probability density");
    igraph3(fpp, &qterm[0], &qterm[dlmt], 0.0, 0.5, 0.2,
           16, x0, y0, 16.0, 8.0, 0.0);
    igraph3(fpp, &qdelay[0], &qdelay[dlmt], 0.0, 0.5, 0.2,
           16, x0, y0, 16.0, 8.0, 1.0);
    width(fpp, 1.1);
    plot(fpp, x0 + 8.0, y0 + 6.0, 0);
    plot(fpp, x0 + 9.5, y0 + 6.0, 1);
    psputs(fpp, x0 + 9.6, y0 + 5.85, 0., 14, ": delay of follow-up");
    width(fpp, 0.0);
    plot(fpp, x0 + 8.0, y0 + 5.0, 0);
    plot(fpp, x0 + 9.5, y0 + 5.0, 1);
    psputs(fpp, x0 + 9.6, y0 + 4.85, 0., 14, ": interval of check");
    plot(fpp, x0 + 7.5, y0 + 4.5, 0);
    plot(fpp, x0 + 7.5, y0 + 6.5, 1);
    plot(fpp, x0 + 14.0, y0 + 6.5, 1);
    plot(fpp, x0 + 14.0, y0 + 4.5, 1);
    plot(fpp, x0 + 7.5, y0 + 4.5, 1);
    plote(fpp);}
void setstat(int argc, char *argv[]){
    char buf[LINE];
    int i, j, k, n, nd, dmax;
    int nall[MAX], ndelay[MAX], nnew[DAY], nfol[DAY], nboth[DAY];
    int maxdat, mindat, fdat, tdat, nlength;
    int nhday[7][PMAX], fhday[7][PMAX], bhday[7][PMAX], nhmax, fhmax, bhmax;
    int number, refcnt, allcnt, reffile, delay, dat, tim, lines;
    int work[11];
    double pref, pall, pdelay, ave, var, length, avea, vara, avel, varl;
    double avenew, sdnew, avefol, sdfol, avebot, sdbot, pweek[7];
    double x, y, x0, y0;
    FILE *fp, *fpp;
    for(i = 0; i < MAX; i++) nref[i] = nall[i] = ndelay[i] = 0;
    for(i = 0; i < DAY; i++) nnew[i] = nfol[i] = nboth[i] = 0;
    for(i = 0; i < PMAX; i++) for(j = 0; j < 7; j++)
      nhday[j][i] = fhday[j][i] = 0;
    if(argc < 2){
      fprintf(stderr, "error in %s\n", argv[0]);
      fprintf(stderr, "Usage : %s FILENAME\n", argv[0]);
      exit(3);}
    if(!(fp = fopen(argv[1], "r"))){
      fprintf(stderr, "file %s cannot open.\n", argv[1]);
      exit(3);}
    if(argc >=3) sscanf(argv[2], "%d", &fdat);
```

```
else fdat = 0;
if(argc >=4) sscanf(argv[3], "%d", &tdat);
else tdat = 0;
for(mindat = DAY, maxdat = 0; fgets(buf, LINE, fp);){
  sscanf(buf, "%5d %5d %5d %5d %5d %5d %7d %5d",
         &number, &refcnt, &allcnt, &reffile, &delay, &dat, &tim, &lines);
  if((dat >= 0) && (dat < DAY)){
    nboth[dat]++;
    if(reffile) nfol[dat]++; else nnew[dat]++;
    if(dat < mindat) mindat = dat; else if(dat > maxdat) maxdat = dat;}}
startdat = mindat + 7; enddat = maxdat - 7;
printf("File = %s, date = %d to %d\n",argv[1], startdat, enddat);
if(fdat && (fdat >= startdat)){
  printf("startdat limited >= %d\n", startdat = fdat);}
if(tdat && (tdat <= enddat)){
  printf("enddat limited <= %d\n", enddat = tdat);}
printf("\ndairy new postings(-) and follows(+)\n");
for(dat = mindat, nhmax = fhmax = bhmax = 0; dat <= maxdat; dat++){
  if((dat >= startdat) && (dat <= enddat)) printf("%5d : ", dat);
  else printf("(%4d): ", dat);
  for(i = 0; i < nnew[dat]; i++) putchar('-');
  for(i = 0; i < nfol[dat]; i++) putchar('+');
  printf("\n");
  if((dat >= startdat) && (dat <= enddat)){
    i = dat % 7;
    nhday[i][nnew[dat]]++; fhday[i][nfol[dat]]++; bhday[i][nboth[dat]]++;
    if(nnew[dat] > nhmax) nhmax = nnew[dat];
    if(nfol[dat] > fhmax) fhmax = nfol[dat];
    if(nboth[dat] > bhmax) bhmax = nboth[dat];}}
for(dat = startdat, avenew = avefol = avebot = 0; dat <= enddat; dat++){
  avenew += nnew[dat]; avefol += nfol[dat]; avebot += nboth[dat];}
lambda = avenew /= (enddat - startdat + 1);
avefol /= (enddat - startdat + 1);
avebot /= (enddat - startdat + 1);
for(dat = mindat, sdnew = sdfol = sdbot = 0; dat <= maxdat; dat++){
  sdnew += (nnew[dat] - avenew) * (nnew[dat] - avenew);
  sdfol += (nfol[dat] - avefol) * (nfol[dat] - avefol);
  sdbot += (nboth[dat] - avebot) * (nboth[dat] - avebot);}
sdnew /= (enddat - startdat + 1);
sdfol /= (enddat - startdat + 1);
sdbot /= (enddat - startdat + 1);
printf("\n daily statistics : start = %4d, end = %4d", startdat, enddat);
printf("\n new post / day : ave = %6.2f, var = %6.2f", avenew, sdnew);
printf(" sd = %6.2f, ratio = %6.3f", sqrt(sdnew), sqrt(sdnew)/avenew);
printf("\n   follow / day : ave = %6.2f, var = %6.2f", avefol, sdfol);
printf(", sd = %6.2f, ratio = %6.3f", sqrt(sdfol), sqrt(sdfol)/avefol);
printf("\n    total / day : ave = %6.2f, var = %6.2f", avebot, sdbot);
printf(", sd = %6.2f, ratio = %6.3f", sqrt(sdbot), sqrt(sdbot)/avebot);
fpp = plots("daily.obj", -20, -20, 520, 700);
x0 = 1.5, y0 = 1.5;
for(dat = mindat, y0 = 0.5; (dat <= maxdat) && (y0 <18.0);
    dat += 80, y0 += 8.0){
  kgraph(fpp, 0, 35, 5, 80, 5, x0, y0, 16.0, 7.0, dat, dat + 80);
  psputs(fpp, x0 + 6.5, y0 - 0.8, 0., 14, "date (1995)");
  psputs(fpp, x0 - 0.7, y0 + 2.0, 90., 14, "articles / day");
  igraph(fpp, &nnew[dat], &nnew[maxdat], 0, 35, 0.1,
         80, x0, y0, 16.0, 7.0, 0.0);
  igraph(fpp, &nboth[dat], &nboth[maxdat], 0, 35, 0.1,
         80, x0, y0, 16.0, 7.0, 1.0);}
plote(fpp);
```

```
 if(dat <= maxdat){
   fpp = plots("daily2.obj", -20, -20, 520, 700);
   x0 = 1.5, y0 = 1.5;
   for(y0 = 0.5; (dat <= maxdat) && (y0 <18.0);
       dat += 80, y0 += 8.0){
   kgraph(fpp, 0, 35, 5, 80, 5, x0, y0, 16.0, 7.0, dat, dat + 80);
   psputs(fpp, x0 + 6.5, y0 - 0.7, 0., 10, "date (1995)");
   psputs(fpp, x0 - 0.7, y0 + 2.0, 90., 14, "articles / day");
   igraph(fpp, &nnew[dat], &nnew[maxdat], 0, 35, 0.1,
          80, x0, y0, 16.0, 7.0, 0.0);
   igraph(fpp, &nboth[dat], &nboth[maxdat], 0, 35, 0.1,
          80, x0, y0, 16.0, 7.0, 1.0);}
   plote(fpp);}
 printf("\n\nweek day histgram of new postings\n");
 week(enddat - startdat, &nnew[startdat], pweek);
 printf("\n\nweek day histgram of follow postings\n");
 week(enddat - startdat, &nfol[startdat], pweek);
 printf("\n\nweek day histgram of new and follow postings\n");
 week(enddat - startdat, &nboth[startdat], pweek);
 rewind(fp);
 for(ave = avea = avel = var = vara = varl = n = nd = dmax = 0;
     fgets(buf, LINE, fp);){
   sscanf(buf, "%5d %5d %5d %5d %5d %5d %7d %5d", &number, &refcnt,
          &allcnt, &reffile, &delay, &dat, &tim, &lines);
   if((dat >= startdat) && (dat <= enddat)){
     ave += refcnt; var += (double)refcnt * refcnt;
     avea += allcnt; vara += allcnt * allcnt;
     if(refcnt){
       length = allcnt / (double)refcnt;
       nlength++; avel += length; varl += length * length;}
     n++; ndat++;
     if(refcnt >= MAX) refcnt = MAX - 1;
     nref[refcnt]++;
     if(allcnt >= MAX) allcnt = MAX - 1;
     nall[allcnt]++;
     if(delay >= MAX) delay = MAX - 1;
     if(reffile) {ndelay[delay]++; nd++;}}}
 ave /= n; var = var / n - ave * ave;
 avea /= n; vara = vara / n - avea * avea;
 avel /= nlength; varl = varl / nlength - avel * avel;
 printf("\ntree form : tree count = %d", n);
 printf("\nnumber of child : ave = %6.3f, var = %6.3f", ave, var);
 printf("\nsize of tree(ts): ave = %6.3f, var = %6.3f", avea, vara);
 printf("\nlength for ts>0 : ave = %6.3f, var = %6.3f", avel, varl);
 printf("\n\nhistogram : total Data Count = %d\n", n);
 printf("\n N     refcnt     allcnt     delay\n");
 printf(" -     -      %c     -      %c     -      %c\n", '%', '%', '%');
 for(i = 0; i < MAX; i++) if(nref[i]) kmax = i;
 for(i = 0; i < MAX; i++) if(ndelay[i]) dmax = i;
 for(i = 0; i < MAX; i++){
   if((i > kmax) && (i > dmax)) break;
   pref = 100.0 * nref[i] / n;
   pall = 100.0 * nall[i] / n;
   pdelay = 100.0 * ndelay[i] / nd;
   printf("%3d%5d%6.2lf%5d%6.2lf%5d%6.2lf\n",
          i, nref[i], pref, nall[i], pall, ndelay[i], pdelay);
   pobs[i] = pref / 100;
   nref2[i] = nref[i];}
 delayx(ndelay, dmax, pweek);
```

```
  estpoi(nest2, nref2, kmax);}
void nbdcalc(){ /* estimate as negative binomial distribution */
  FILE *fpp, *fp;
  int i, n, k;
  double ave, var, alpha, theta, fact_k, x, y, x0, y0, d;
  double ave1, var1, ave2, var2;
  for(i = ave = n = 0; i < MAX; i++){
    ave += nref[i] * i;
    n += nref[i];}
  ave /= n;
  for(i = var = 0; i < MAX; i++) var += nref[i] * (i - ave) * (i - ave);
  var /= n;
  printf("\nrefcnt analysis :  ave = %lf, var = %lf\n", ave, var);
  theta = ave / (var - ave);
  alpha = ave * theta;
  if(fp = fopen("simpara.dat","w")){
    fprintf(fp, "%f\n", lambda);
    fprintf(fp, "%f\n", alpha);
    fprintf(fp, "%f\n", theta);
    fclose(fp);}
  else printf("\n*** error *** : file \"simpara.dat\" cannot open");
  ave1 = ave2 = lambda * theta /( theta - alpha);
  var1 = lambda * theta * (theta * theta + alpha) /
    ((theta - alpha) * (theta - alpha) * (theta + alpha));
  var2 = lambda * theta * (alpha * q2 + theta * theta) /
    ((theta * theta - alpha * alpha * q2) * (theta - alpha));
  printf("estimated : alpha = %f, theta = %f\n", alpha, theta);
  printf("            lambda = %f, Q^2 = %f\n", lambda, q2);
  printf("estimated : ave1 = %f, var1 = %f, sd = %f, sd/ave = %f\n",
         ave1, var1, sqrt(var1), sqrt(var1) / ave1);
  printf("estimated : ave2 = %f, var2 = %f, sd = %f, sd/ave = %f\n",
         ave2, var2, sqrt(var2), sqrt(var2) / ave2);
  printf("\nestimated distribution\n\n    k : neg.bin., observed, poison\n");
  for(k = 0; k < kmax; k++){
    for(i = fact_k = 1; i <= k; i++) fact_k *= i;
    pest[k] = (gamma_ratio(alpha, k) / fact_k) *
      pow((1 + theta)/theta, -alpha - k) / power(theta, k);
    nest[k] = pest[k] * ndat;
    printf("%5d : %6.2lf, %6.2lf, %6.2f\n",
           k, pest[k] * 100., pobs[k] * 100., pest2[k] * 100.);}
  fpp = plots("follow.obj", -20, -20, 520, 700);
  x0 = 1.5; y0 = 1.5;
  width(fpp, 1.0);
  for(y = y0 + 20.0, d = 0.8; y >= y0 - 0.1; y -= 2.5, d -= 0.1){
    psputd(fpp, x0 + 0.3, y-0.15, 0., 14, 'r', "%4.1f", d);}
  plot(fpp, x0 + 0.5, y0 + 20.0, 0);
  for(y = y0 + 20.0; y >= y0 + 2.4; y -= 2.5){
    plot(fpp, x0, y, 1);
    plot(fpp, x0, y - 2.5, 1);
    plot(fpp, x0+ 0.5, y - 2.5, 1);}
  plot(fpp, x0+ 0.5, y, 0);
  plot(fpp, x0, y0, 0);
  for(x = x0 + 1.0; x <= 17.1; x += 2.0){
    plot(fpp, x, y0, 1);
    plot(fpp, x, y0 + 0.5, 1);
    plot(fpp, x, y0, 1);}
  for(x = x0 + 1.0, i = 0; x <= 17.1; x += 2.0, i++){
    psputi(fpp, x+0.05, y0-0.7, 0., 14, 'c', i);}
```

```
psputs(fpp, x0 + 5.0, y0 - 1.4, 0., 18, "number of follows");
psputs(fpp, x0 - 1.2, y0 + 8.0, 90., 18, "probability density");
width(fpp, 0.0);
for(i = 0; i < ((kmax <= 8) ? kmax : 8); i++){
  x = x0 + (i + 0.5) * 2.0;
  y = y0 + pobs[i] * 25.0;
  plot(fpp, x, y, i);}
plot(fpp, x, y, 0);
width(fpp, 1.0);
for(i = 0; i < ((kmax <= 8) ? kmax : 8); i++){
  x = x0 + (i + 0.5) * 2.0;
  y = y0 + pobs[i] * 25.0;
  plot(fpp, x, y, 0);
  symbol(fpp, 0.3, 3);}
width(fpp, 0.0);
for(i = 0; i < ((kmax <= 8) ? kmax : 8); i++){
  x = x0 + (i + 0.5) * 2.0;
  y = y0 + pest[i] * 25.0;
  plot(fpp, x, y, i);}
plot(fpp, x, y, 0);
width(fpp, 1.0);
for(i = 0; i < ((kmax <= 8) ? kmax : 8); i++){
  x = x0 + (i + 0.5) * 2.0;
  y = y0 + pest[i] * 25.0;
  plot(fpp, x, y, 0);
  symbol(fpp, 0.3, 2);}
width(fpp, 0.0);
for(i = 0; i < ((kmax <= 8) ? kmax : 8); i++){
  x = x0 + (i + 0.5) * 2.0;
  y = y0 + pest2[i] * 25.0;
  plot(fpp, x, y, i);}
plot(fpp, x, y, 0);
width(fpp, 1.0);
for(i = 0; i < ((kmax <= 8) ? kmax : 8); i++){
  x = x0 + (i + 0.5) * 2.0;
  y = y0 + pest2[i] * 25.0;
  plot(fpp, x, y, 0);
  symbol(fpp, 0.3, 1);}
x0 = 8.0; y0 = 14.0;
width(fpp, 1.0);
plot(fpp, x0, y0, 0);
plot(fpp, x0 + 8.0, y0, 1);
plot(fpp, x0 + 8.0, y0 + 3.0, 1);
plot(fpp, x0, y0 + 3.0, 1);
plot(fpp, x0, y0, 1);
width(fpp, 0.0);
plot(fpp, x0 + 0.5, y0 + 2.5, 0);
plot(fpp, x0 + 1.5, y0 + 2.5, 1);
width(fpp, 1.0);
plot(fpp, x0 + 1.0, y0 + 2.5, 0);
symbol(fpp, 0.3, 3);
psputs(fpp, x0 + 1.6, y0 + 2.3, 0., 18, ": observed");
width(fpp, 0.0);
plot(fpp, x0 + 0.5, y0 + 1.5, 0);
plot(fpp, x0 + 1.5, y0 + 1.5, 1);
width(fpp, 1.0);
plot(fpp, x0 + 1.0, y0 + 1.5, 0);
symbol(fpp, 0.3, 2);
psputs(fpp, x0 + 1.6, y0 + 1.3, 0., 18, ": negative binominal");
width(fpp, 0.0);
```

```
  plot(fpp, x0 + 0.5, y0 + 0.5, 0);
  plot(fpp, x0 + 1.5, y0 + 0.5, 1);
  width(fpp, 1.0);
  plot(fpp, x0 + 1.0, y0 + 0.5, 0);
  symbol(fpp, 0.3, 1);
  psputs(fpp, x0 + 1.6, y0 + 0.3, 0., 18, ": Poisson distribution");
  plote(fpp);}
double ksstat(int n, int kmax, int obs[], double est[]){
  double D20, D10, D05, D01;
  double f[MAX], s[MAX], sumobs, sumest, d, dd, sqn;
  int i;
  if(kmax < 1){
    printf("error ksstat: kmax = %d < 1\n", kmax);
    exit(3);}
  sqn = sqrt((double) kmax);
  D20 = 1.07 / sqn;
  D10 = 1.22 / sqn;
  D05 = 1.36 / sqn;
  D01 = 1.63 / sqn;
  printf("\nKolmogonov-Smirnov statistic test\n");
  printf("kmax = %6d\n", kmax);
  printf("D20 = %6.2lf, D10 = %6.2lf, D05 = %6.2lf, D01 = %6.2lf\n",
         D20, D10, D05, D01);
  for(i = 1; i < kmax; i++){
    obs[i] += obs[i - 1];
    est[i] += est[i - 1];}
  sumobs = obs[kmax - 1]; sumest = est[kmax - 1];
  for(d = i = 0; i < kmax; i++){
    s[i] = obs[i] / sumobs;
    f[i] = est[i] / sumest;
    dd = f[i] - s[i]; if(dd < 0) dd = -dd;
    if(dd > d) d = dd;}
  printf("\n ksstat : d = %lf\n", d);
  if(d > D01) return 0.01L;
  if(d > D05) return 0.05L;
  if(d > D10) return 0.10L;
  if(d > D20) return 0.20L;
  return 1.00L;}
double chisqtst(int n, int k, int obs[], double est[]){
  double chisq0_99[30] = {0.000157, 0.0201, 0.115,  0.297,  0.554,
                          0.872,  1.239,  1.646,  2.088,  2.558,
                          3.053,  3.571,  4.107,  4.660,  5.229,
                          5.812,  6.408,  7.015,  7.633,  8.260,
                          8.897,  9.542, 10.196, 10.856, 11.524,
                         12.198, 12.879, 16.565, 14.256, 14.953};
  double chisq0_95[30] = {0.00393,  0.103,  0.352,  0.711,  1.145,
                          1.635,  2.167,  2.733,  3.325,  3.940,
                          4.575,  5.226,  5.892,  6.571,  7.261,
                          7.962,  8.672,  9.390, 10.117, 10.851,
                         11.591, 12.338, 13.091, 13.848, 14.611,
                         15.397, 16.151, 16.928, 17.708, 18.493};
  double chisq0_90[30] = {0.0158, 0.211, 0.584, 1.064, 1.610,
                          2.204, 2.833, 3.490, 4.168, 4.865,
                          5.578, 6.304, 7.042, 7.790, 8.547,
                          9.312, 10.085, 10.865, 11.651, 12.443,
                          13.240, 14.042, 14.848, 15.659, 16.473,
                          17.292, 18.114, 18.939, 19.768, 20.599};
  double chisq0_5[30] = {3.841, 5.991, 7.815, 9.488, 11.070,
                         12.592, 14.067, 15.507, 16.919, 18.307,
                         19.675, 21.026, 22.362, 23.685, 24.996,
                         26.296, 27.587, 28.869, 30.144, 31.410,
                         32.671, 33.924, 35.172, 36.415, 37.652,
                         38.885, 40.113, 41.337, 42.557, 43.773};
```

```
  double chisq0_1[30] = {6.635, 9.210, 11.341, 13.277, 15.086,
                         16.812, 18.475, 20.090, 21.666, 23.209,
                         24.725, 26.217, 27.688, 29.141, 30.578,
                         32.000, 33.409, 34.805, 36.191, 37.566,
                         38.932, 40.289, 41.638, 42.980, 44.314,
                         45.642, 46.963, 48.278, 49.588, 50.892};
  double chisq, chisq99, chisq95, chisq5, chisq1;
  int i, j;
  if(n < 30){
    chisq99 = chisq0_99[n];
    chisq95 = chisq0_95[n];
    chisq5 = chisq0_5[n];
    chisq1 = chisq0_1[n];}
  else{
    chisq99 = 0.5 * pow(sqrt((double)(2 * n - 2)) - 2.32635, 2);
    chisq95 = 0.5 * pow(sqrt((double)(2 * n - 2)) - 1.64485, 2);
    chisq5 = 0.5 * pow(sqrt((double)(2 * n - 2)) + 1.64485, 2);
    chisq1 = 0.5 * pow(sqrt((double)(2 * n - 2)) + 2.32635, 2);}
  printf("\nchi square test : n = %d, k = %d\n", n, k);
  printf("\n     k : nobs   nest\n");
  for(i = 0; i < k; i++){
    printf("%6d  %5d %7.1lf\n", i, obs[i], est[i]);}
  if(k >= 30) printf("err chisqtst, k >=30");
  for(i = chisq = 0; i < k; i++){
    chisq += power(obs[i] - est[i], 2) / est[i];}
  printf("\n     chi square = %lf\n", chisq);
  if(chisq < chisq0_99[k -2]) return 0.99;
  else if(chisq < chisq0_95[k - 2]) return 0.95;
  else if(chisq < chisq0_90[k - 2]) return 0.90;
  else if(chisq < chisq0_5[k - 2]) return 0.05;
  else if(chisq < chisq0_1[k - 2]) return 0.01;
  else return 0.0;}
void main(int argc, char *argv[]){
  setstat(argc, argv);
  nbdcalc();
  printf("\ndistribution of article strength assumed\n");
  chisqtst(ndat, kmax - 2, nref, nest);
  printf("\nconstant strength of article assumed\n");
  chisqtst(ndat, kmax - 2, nref2, nest2);
  ksstat(ndat, kmax, nref, nest);}
```

B5. シミュレーションプログラム

以下は投稿行動のシミュレーションプログラムである。

```
#include <stdio.h>
#include <math.h>
#include <stdlib.h>
#include <malloc.h>
#include "psplot.h"

#define KMAX 1000
#define DMAX 240
#define AMAX 32000

int days = DMAX;
int nnew[DMAX], nboth[DMAX], pday = 240, ndelay;
int rcnt[AMAX], acnt[AMAX], part[AMAX], aday[AMAX], adly[AMAX], ano;
int astart = DMAX, aend;
double pdelay[DMAX], pweek[7];
```

```
double potnew = 3, alpha = 1, theta = 1.7, rih = 0;
struct stack {int day; double pot; int pairent;
              int delay; struct stack * next;};
struct stack * top = NULL;
int iout = 0;
FILE *fout;
void kgraph(FILE * fpp, int mind, int maxd, int dd, int n,
            int dn, double x0, double y0, double wx, double wy,
            int d0, int d1){
  double x, y;
  int i, d;
  width(fpp, 1.0);
  plot(fpp, x0, y0, 0);
  plot(fpp, x0 + wx, y0, 1);
  plot(fpp, x0 + wx, y0 + wy, 1);
  plot(fpp, x0, y0 + wy, 1);
  plot(fpp, x0, y0, 1);
  plot(fpp, x0, y0, 0);
  for(i = mind; i < maxd; i += dd){
    y = y0 + wy * (i - (double)mind) / (maxd - (double)mind);
    plot(fpp, x0, y, 0);
    plot(fpp, x0 + 0.2, y, 1);
    plot(fpp, x0 + 0.2, y, 0);}
  for(i = mind; i < maxd + dd / 2; i += dd){
    y = y0 + wy * (i - (double)mind) / (maxd - (double)mind);
    psputi(fpp, x0 - 0.1, y - 0.1, 0., 10, 'c', i);}
  for(i = 0; i <= n; i += dn){
    x = x0 + wx * i / (double)n;
    plot(fpp, x, y0, 0);
    plot(fpp, x, y0 + 0.2, 1);
    plot(fpp, x, y0 + 0.2, 0);}
  for(i = 0; i <= n; i += dn){
    x = x0 + wx * i / (double)n;
    d = d0 + i * (d1 - d0) / n;
    psputi(fpp, x + 0.2, y0 - 0.35, 0., 10, 'c', d);}}
void igraph(FILE * fpp,int * dat, int * dend, int mind, int maxd, double dy,
            int n, double x0, double y0, double wx, double wy, double wl){
  double x, y;
  int i;
  width(fpp, wl);
  for(i = 0; (i < n) && (dat < dend); i++, dat++){
    y = y0 + wy * (*dat - (double)mind) / (maxd - (double)mind);
    if(y < y0) y = y0; else if(y > y0 + wy) y = y0 + wy;
    x = x0 + wx * (i + 0.5) / n;
    plot(fpp, x, y, i);}
  plot(fpp, x, y, 0);}
void push(int day, double pot, int pairent, int delay){
  struct stack *add, *cur;
  add = (struct stack *)malloc(sizeof(struct stack));
  add->day = day; add->pot = pot;
  add->pairent = pairent; add->delay = delay;
  if(!top){top = add; add->next = NULL;}
  else{
    if(top->day > day){
      add->next = top; top = add;}
    else{
      for(cur = top;
          (cur->next) && (cur->next->day <= day);
          cur = cur->next);
      add->next = cur->next; cur->next = add;}}}
```

```
int pop(double *ppot, int *pairent, int *delay){
  /* pop pot and return day or -1 (queue null) */
  struct stack *cur;
  if(top == NULL) return -1;
  cur = top;
  top = cur->next;
  *ppot = cur->pot; *pairent = cur->pairent; *delay = cur->delay;
  return cur->day;}
int poirnd(double p){/* calculate poison's rundum number */
  double sum, x, u;
  int k;
  u = rand() / (double)RAND_MAX;
  for(sum = k = 0, x = 1.0 / exp(p); k < KMAX; k++){
    if(k) x *= p / k;
    if((sum += x) > u) break;}
  return k;}
double grand1(double alpha, double theta){
  /* generate Gamma randum for a < 1 by Ahrens & Dieter 1974 */
  double e, u0, u1, p;
  e = exp((double)1);
  for(;;){
  u0 = (double)rand() / (double)RAND_MAX;
  u1 = (double)rand() / (double)RAND_MAX;
  if(u0 <= (e / (alpha + e))){
    p = pow((alpha + e) * u0 / e, 1 / alpha);
    if(u1 <= exp(-p)) return p / theta;}
  else{
    p = -log((alpha + e) * (1 - u0) / (alpha * e));
    if(u1 <= pow(p, alpha - 1)) return p / theta;}}}
double grand2(double alpha, double theta){
  /* generate Gamma randum for alpha > 1 by Cheng & Feaaat 1979 */
  double c1, c2, c3, c4, c5, u1, u2, w;
  c1 = alpha -1;
  c2 = (alpha - 1 / (6 * alpha)) / c1;
  c3 = 2 / c1;
  c4 = c3 + 2;
  c5 = 1 / sqrt(alpha);
  for(;;){
    do{
      u1 = (double)rand() / (double)RAND_MAX;
      u2 = (double)rand() / (double)RAND_MAX;
      if(alpha > 2.5) u1 = u2 + c5 * (1 - 1.86 * u1);
    } while ((u1 <= 0) || (u1 >= 1));
    w = c2 * u2 / u1;
    if((c3 * u1 + w + 1 / w) <= c4) return c1 * w / theta;
    if((c3 * log(u1) - log(w) + w) < 1) return c1 * w / theta;}}
double newpot(){
  double r;
  int i, a;
  if(alpha <= 1) return grand1(alpha, theta);
  else return grand2(alpha, theta);}
double potset(double pot){/* calculate potential */
  return(rih * pot + (1 - rih) * newpot());}
#define LINE 256
void varinit(char *file){
  FILE *fp;
  int i;
  char buf[LINE];
  double sum;
  if((fp = fopen(file, "r")) == NULL){
    fprintf(stderr, "error: file %s cannot open.", file);
```

```
    exit(3);}
  for(i = 0; (fgets(buf, LINE, fp)) && (i < 7); i++){
    sscanf(buf, "%lf", &pweek[i]);}
  if(i < 7) for(i = 0; i < 7; i++) pweek[i] = 1.0/7.0;
  for(sum = i = 0; i < 7; i++) sum += pweek[i];
  if(sum <= 0.0) for(i = 0; i < 7; i++) pweek[i] = 1.0/7.0;
  else for(i = 0; i < 7; i++) pweek[i] /= sum;
  for(i = 0; i < 7; i++) printf("iw[%d] = %f\n", i, pweek[i]);
  for(ndelay = 0; (fgets(buf, LINE, fp)) && (ndelay < DMAX); ndelay++){
    sscanf(buf, "%lf", &pdelay[ndelay]);}
  for(i = 1; i < ndelay; i++) pdelay[i] += pdelay[i - 1];
  for(i = 0; i < ndelay; i++) pdelay[i] /= pdelay[ndelay - 1];}
int dlyset(int day){/* calculate delay */
  double u;
  int i, j;
  do{
    u = rand() / (double)RAND_MAX;
    for(i = 0; (u > pdelay[i]) && (i < ndelay); i++);
  } while(pweek[(day + i) % 7] < rand() / (double)RAND_MAX);
  return i;}
void post(int day, double pot){
  int i, n, delay;
  n = poirnd(pot);
  for(i = 0; i < n; i++){
    if((delay = dlyset(day)) < 0) delay = 0;
    push(day + delay, potset(pot), ano, delay);}}
void postnew(int day){
  int i, n;
  double pot;
  pot = newpot();
  putchar('-');
  fprintf(fout, "%d %d %d %d\n", ano, day, 0, 0);
  post(day, pot);
  if(ano < AMAX){
    rcnt[ano] = acnt[ano] = adly[ano] = 0;
    aday[ano] = day, part[ano] = -1;
    if(day < astart) astart = day;
    if(day > aend) aend = day;}
  ano++;
  if(day < pday){
    nnew[day]++; nboth[day]++;}}
void postfol(int day){
  double pot;
  int dday, pairent, delay, pno;
  while((dday = pop(&pot, &pairent, &delay)) >= 0){
    if(dday > day){
      push(dday, pot, pairent, delay);
      return;}
    putchar('+');
    post(day, pot);
    fprintf(fout, "%d %d %d %d\n", ano, day, pairent, delay);
    if(ano < AMAX){
      rcnt[ano] = acnt[ano] = 0;
      part[ano] = pairent; adly[ano] = delay;
      aday[ano] = day;
      if(day < astart) astart = day;
      if(day > aend) aend = day;}
    if((pairent >= 0) && (pairent < AMAX)){
      rcnt[pairent]++;
```

```
      for(pno = pairent; pno >= 0; pno = part[pno]) acnt[pno]++;}
    ano++;
    if(day < pday) nboth[day]++;}}
void stat(){
  double aven, varn, avef, varf, aveb, varb, avec, varc;
  double avea, vara, avel, varl;
  double avenw[7], varnw[7], avefw[7], varfw[7], avebw[7], varbw[7], length;
  int day, a, n, nfol, nw[7], iw, nlength;
  printf("\nstat start");
  for(iw = 0; iw < 7; iw++){
    avenw[iw] = varnw[iw] = avefw[iw] = varfw[iw] = avebw[iw] = varbw[iw] = 0;
    nw[iw] = 0;}
  aven = varn = avef = varf = aveb = varb = n = 0;
  for(day = astart + 7; day < aend - 7; day++){
    iw = day % 7; nw[iw]++; n++; nfol = nboth[day] - nnew[day];
    aven += nnew[day]; varn += (double)nnew[day] * nnew[day];
    avef += nfol; varf += (double)nfol * nfol;
    aveb += nboth[day]; varb += (double)nboth[day] * nboth[day];
    avenw[iw] += nnew[day]; varnw[iw] += (double)nnew[day] * nnew[day];
    avefw[iw] += nfol; varfw[iw] += (double)nfol * nfol;
    avebw[iw] += nboth[day]; varbw[iw] += (double)nboth[day] * nboth[day];}
  if(n >0){
    aven /= n; avef /= n; aveb /= n;
    varn = varn / n - aven * aven;
    varf = varf / n - avef * avef;
    varb = varb / n - aveb * aveb;}
  else printf("\nerror n = 0!\n");
  for(iw = 0; iw < 7; iw++){
    if(nw[iw] > 0){
      avenw[iw] /= nw[iw]; avefw[iw] /= nw[iw]; avebw[iw] /= nw[iw];
      varnw[iw] = varnw[iw] / nw[iw] - avenw[iw] * avenw[iw];
      varfw[iw] = varfw[iw] / nw[iw] - avefw[iw] * avefw[iw];
      varbw[iw] = varbw[iw] / nw[iw] - avebw[iw] * avebw[iw];}
    else("\nerror nw[%d] = 0!\n", iw);}
  printf("\n daily statistics : start = %4d, end = %4d", astart + 8, aend - 8);
  printf("\n new post / day : ave = %6.2f, var = %6.2f,", aven, varn);
  printf(" sd = %6.2f, ratio = %6.3f", sqrt(varn), sqrt(varn)/aven);
  printf("\n   follow / day : ave = %6.2f, var = %6.2f,", avef, varf);
  printf(" sd = %6.2f, ratio = %6.3f", sqrt(varf), sqrt(varf)/avef);
  printf("\n    total / day : ave = %6.2f, var = %6.2f,", aveb, varb);
  printf(" sd = %6.2f, ratio = %6.3f\n\n", sqrt(varb), sqrt(varb)/aveb);
  printf("day +----- new post/day ------+");
  printf("+----- fol post/day ------+");
  printf("+----- all post/day ------+\n");
  printf("       ave    var   s.d. sd/ave");
  printf("    ave    var   s.d. sd/ave");
  printf("    ave    var   s.d. sd/ave\n");
  for(iw = 0; iw < 7; iw++){
    if(avebw[iw]<= 0) continue;
    printf("%3d %6.2f %6.2f %6.2f %6.3f", iw, avenw[iw], varnw[iw],
           sqrt(varnw[iw]), sqrt(varnw[iw]) / avenw[iw]);
    printf(" %6.2f %6.2f %6.2f %6.3f", avefw[iw], varfw[iw],
           sqrt(varfw[iw]), sqrt(varfw[iw]) / avefw[iw]);
    printf(" %6.2f %6.2f %6.2f %6.3f\n", avebw[iw], varbw[iw],
           sqrt(varbw[iw]), sqrt(varbw[iw]) / avebw[iw]);}
  for(avec = varc = avea = vara = avel = varl = a = n = nlength = 0;
      (a < ano) && (a < AMAX); a++){
    if((aday[a] > astart + 7) &&(aday[a] < aend - 7)){
      n++;
      avec += rcnt[a]; varc += (double)rcnt[a] * rcnt[a];
      avea += acnt[a]; vara += (double)acnt[a] * acnt[a];
```

```
      if(rcnt[a]){
        nlength++; length = (double)acnt[a] / (double) rcnt[a];
        avel += length; varl += length * length;}}}
  avec /= n; varc = varc / n - avec * avec;
  avea /= n; vara = vara / n - avea * avea;
  avel /= nlength; varl = varl / nlength - avel * avel;
  printf("\ntree form : tree count = %d", n);
  printf("\nnumber of child : ave = %6.3f, var = %6.3f", avec, varc);
  printf("\nsize of tree(ts): ave = %6.3f, var = %6.3f", avea, vara);
  printf("\nlength for ts>0 : ave = %6.3f, var = %6.3f\n", avel, varl);}
main(int argc, char * argv[]){
  int day, nnewp, i;
  double dummy;
  double x0, y0;
  FILE *fpp, *fp;
  static char file[256] = "simvars.dat";
  if(fp = fopen("simpara.dat","r")){
    fscanf(fp, "%lf\n", &potnew);
    fscanf(fp, "%lf\n", &alpha);
    fscanf(fp, "%lf\n", &theta);
    fclose(fp);}
  if(argc <= 1){
    printf("usage : sim days potnew alpha theta rih\n");
    printf("        no arguments, default value used.\n\n");}
  if(argc > 1) sscanf(argv[1], "%d", &days);
  if(argc > 2) sscanf(argv[2], "%lf", &potnew);
  if(argc > 3) sscanf(argv[3], "%lf", &alpha);
  if(argc > 4) sscanf(argv[4], "%lf", &theta);
  if(argc > 5) sscanf(argv[5], "%lf", &rih);
  if(argc > 6) sscanf(argv[6], "%s", file);
  printf("days = %d, potnew = %f, alpha = %f, theta = %f, rih = %f\n",
         days, potnew, alpha, theta, rih);
  if(days < DMAX) pday = days; else pday = DMAX;
  varinit(file);
  fout = fopen("sim.out", "w");
  for(day = 0; day < days; day++){
    printf("\n day = %4d : ", day);
    nnewp = poirnd(potnew * 7.0 * pweek[day % 7]);
    for(i = 0; i < nnewp; i++) postnew(day);
    postfol(day);}
  printf("\n");
  stat();
  fpp = plots("simres.obj", -20, -20, 520, 700);
  x0 = 1.5;
  for(day = 0, y0 = 1.5; (day < pday) && (y0 < 18.0);
      day += 80, y0 += 8.0){
    kgraph(fpp, 0, 35, 5, 80, 5, x0, y0, 16.0, 7.0, day, day + 80);
    psputs(fpp, x0 + 6.5, y0 - 0.8, 0, 14, "d a t e");
    psputs(fpp, x0 - 0.7, y0 + 2.0, 90., 14, "articles / day");
    igraph(fpp, &nnew[day], &nnew[pday], 0, 35, 0.1,
           80, x0, y0, 16.0, 7.0, 0.0);
    igraph(fpp, &nboth[day], &nboth[pday], 0, 35, 0.1,
           80, x0, y0, 16.0, 7.0, 1.0);}
  plote(fpp);}
```

次のシェルスクリプトにより、統計解析とシミュレーションを連続的に行なう。ディレクトリ /home/seo/news はニュース記事を格納するディレクトリの親ディレクトリであり、ヘッダー情報抽出ファイル等はここに置かれている。

```
stat /home/seo/news/$1.ref >$1.sta
mv daily.obj day$1.obj
mv delay.obj dly$1.obj
mv follow.obj fol$1.obj
sim >$1.lst
mv simres.obj sim$1.obj
mv simvars.dat var$1.dat
mv simpara.dat par$1.dat
```

B6. Γ 分布のプロット

以下のプログラムはΓ分布をプロットするプログラムである。乱数発生ルーチンのテストを兼ねて、Γ乱数を発生して、この頻度分布を求めることにより計算している。

```
#include <stdio.h>
#include <stdlib.h>
#include <math.h>
#include "psplot.h"

double grand1(double alpha, double theta){
  /* generate Gamma randum for a < 1 by Ahrens & Dieter 1974 */
  double e, u0, u1, p;
  e = exp((double)1);
  for(;;){
  u0 = (double)rand() / (double)RAND_MAX;
  u1 = (double)rand() / (double)RAND_MAX;
  if(u0 <= (e / (alpha + e))){
    p = pow((alpha + e) * u0 / e, 1 / alpha);
    if(u1 <= exp(-p)) return p / theta;}
  else{
    p = -log((alpha + e) * (1 - u0) / (alpha * e));
    if(u1 <= pow(p, alpha - 1)) return p / theta;}}}

double grand2(double alpha, double theta){
  /* generate Gamma randum for alpha > 1 by Cheng & Feaaat 1979 */
  double c1, c2, c3, c4, c5, u1, u2, w;
  c1 = alpha -1;
  c2 = (alpha - 1 / (6 * alpha)) / c1;
  c3 = 2 / c1;
  c4 = c3 + 2;
  c5 = 1 / sqrt(alpha);
  for(;;){
    do{
      u1 = (double)rand() / (double)RAND_MAX;
      u2 = (double)rand() / (double)RAND_MAX;
      if(alpha > 2.5) u1 = u2 + c5 * (1 - 1.86 * u1);
    } while ((u1 <= 0) || (u1 >= 1));
    w = c2 * u2 / u1;
    if((c3 * u1 + w + 1 / w) <= c4) return c1 * w / theta;
    if((c3 * log(u1) - log(w) + w) < 1) return c1 * w / theta;}}

#define N 1000000
#define NH 25
#define MAXP 2.5
#define HMAX (2 * N * MAXP / NH)

main(int argc, char *argv[]){
  char name[256];
  int ic, up;
  long i, k, hist[NH], hmax;
  double p, alpha, theta;
  double x, y, x0, y0, xp, yp, pd[NH], haba;
  double xl, yl, xx, yy, xxl, yyl, xc, yc, xcl, ycl;
  FILE *fp;
  if(argc < 3){
```

```
    fprintf(stderr, "usage : gamma alpha theta ...\n");
    exit(3);}
  fp = plots("pot.obj", -20, -20, 500, 700);
  x0 = 1.5; y0 = 2.5;
  plot(fp, 0.5+x0, 20.0+y0, 0);
  for(i = 20; i > 0; i -= 4){
    plot(fp, x0, i + y0, 1);
    plot(fp, x0, i - 4.0 + y0, 1);
    plot(fp, x0+0.5,  i - 4.0 + y0, 1);}
  for(i = 3; i <= 15; i += 3){
    plot(fp, x0 + i, y0, 1);
    plot(fp, x0 + i, y0 + 0.5, 1);
    plot(fp, x0 + i, y0, 1);}
  for(i = 0; i <= 20; i += 4){
     psputd(fp, x0 + 0.2, y0 + i - 0.2, 0.0, 14, 'r', "%.2f",
            (double)i / 16.0);}
  for(i = 3; i <= 15; i += 3){
    psputd(fp, x0 + i - 0.25, y0 - 0.5, 0., 14, 'c', "%.2f",
           (double)i * 2.5 / 15.0);}
  psputs(fp, x0 + 5.0, y0 - 1.3, 0., 18, "Number of follows");
  psputs(fp, x0 - 1.5, y0 + 7.0, 90., 18, "probability density");
  plot(fp, x0 + 15, y0, 0);
  xp = 15.0 / NH;
  yp = 16.0;
  for(ic = 1; ic <argc; ic += 3){
    sscanf(argv[ic], "%lf", &alpha);
    sscanf(argv[ic + 1], "%lf", &theta);
    sscanf(argv[ic + 2], "%256s", name);
    printf("\nalpha = %f, theta = %f\n", alpha, theta);
    for(i = 0; i < NH; i++) hist[i] = 0;
    for(i = 0; i < N; i++){
      if(alpha <= 1) p = grand1(alpha, theta);
      else p = grand2(alpha, theta);
      k = p * NH / MAXP;
      if((k >= 0) && (k < NH)) hist[k]++;}
    for(i = 0; i < NH; i++) pd[i] = (double)hist[i] * NH / (MAXP * N);
    for(xl = yl = up = i = 0; i < NH; i++){
      x = (i + 0.5) * xp + x0;
      y = pd[i] * yp + y0;
      plot(fp, x, y, i?1:0);
      if(!up){
        yy = x + y0 - x0 + 6.0 - 0.3 * ic;
        if((yl > yyl) && (y <= yy)) up = i;
        else{
          yl = y; yyl = yy;}}}
    plot(fp, x, y, 0);
    x = (up + 0.5) * xp + x0;
    xl = (up - 0.5) * xp + x0;
    y = pd[up] * yp + y0;
    yl = pd[up - 1] * yp + y0;
    xc = (x - xl) * (yyl - yl) / (y - yl - yy + yyl) + xl;
    yc = xc + y0 - x0 + 6.0 - 0.3 * ic;
    xcl = x0 + 7.0;
    ycl = xcl + y0 - x0 + 6.0 - 0.3 * ic;
    plot(fp, xc, yc, 0);
    plot(fp, xcl, ycl, 1);
    plot(fp, xcl + 0.23 * strlen(name), ycl, 1);
    psputs(fp, xcl + 0.1, ycl + 0.2, 0., 12, name);}
  plote(fp);}
```