Fluid-diffusive modelling for large P2P file-sharing systems

Fluid-diffusive modelling for large P2P file-sharing systems

G. Carofiglio[c], R. Gaeta[a]*, M. Garetto[a], P. Giaccone[b], E. Leonardi[b] and M. Sereno[a]

[a]*Dipartimento di Informatica, Università di Torino, Corso Svizzera 185, 10149, Italy;*
[b]*Dipartimento di Elettronica, Politecnico di Torino, Corso Duca degli Abruzzi 24, 10129, Italy;*
[c]*Bell Labs, Alcatel-Lucent, France*

This article presents an application of the basic concepts of statistical physics to devise an approximate model describing the dynamics of large peer-to-peer (P2P) file-sharing networks, based on fluid-diffusive equations. The model we propose is quite general and highly modular and allows to represent several effects related to resource distribution among peers, user behaviour, resource localization algorithms and dynamic structure of the overlay topology. As the complexity of the model is largely independent of the system size, it provides an effective method for the analysis of very large P2P systems.

Keywords: peer-to-peer networks; file-sharing systems; performance evaluation; fluid-diffusive modelling

1. Introduction

Peer-to-peer (P2P) applications have obtained an unexpected success in the Internet users' community; several statistics on IP traffic have recently put in evidence that traffic originated by P2P applications represents nowadays the dominant component of all Internet traffic [1,2]. Although the P2P paradigm has become very popular for file sharing (Napster, Gnutella, KaZaa, eDonkey, BitTorrent, to name a few), it can potentially play an important role in several other application contexts. As an example, Skype [3], a P2P-based telephony application, attracts millions of users every day. Furthermore, the adoption of P2P paradigms has also been recently proposed either for content delivery infrastructures (see for instance [4,5]) to better face Internet 'flash crowds', that is unexpected rapid increases in the demand for particular resources, or to broadcast video contents over Internet [6–8].

Although a significant effort was directed to the design of new protocols and architectures, only little has been done in the direction of modelling and understanding the fundamental dynamics of P2P networks. This is mainly due to the extreme complexity of P2P networks comprising hundreds of thousands (or often millions) of interacting users. However, in our opinion, modelling P2P systems is fundamental to better understanding the dynamics of such complex systems and to evaluate the impact that different components have on the behaviour of a P2P system as a whole.

Two fundamental questions naturally arise when trying to model a complex application such as P2P systems: (1) should we tailor the model to a particular P2P system or should

*Corresponding author. Email: rossano@di.unito.it

we describe the evolution of a P2P system trying to abstract from the particular implementation? and (2) what are the fundamental performance indexes of a P2P system we are interested in?

For what concerns the first question, many are the possible reasons to build a model tailored to a particular P2P implementation. By focusing on a particular implementation, it is possible to more accurately predict its performance and to identify the most critical aspects of the system under study. On the other hand, however, one should consider that the P2P arena is still highly magmatic, as different P2P implementations are proposed and deployed at a very fast rate and with a relatively short life expectation; hence, a model that focuses on one particular implementation may fail to provide answers whose validity is somehow general. For these reasons, we have devised a high-level model of an unstructured P2P file-sharing system (Gnutella-like) that abstracts from the specific implementation. We expect, in this way, to obtain less accurate but more general insights into the significant dynamics of P2P systems, which could guide the design of future P2P systems.

Concerning the second question, we have focused on user perspective, according to which the two most significant performance indexes are the probability to locate the desired content and the time needed to retrieve a copy of it.

Traditional Markovian modelling techniques relying on a microscopic descriptive approach, in which architectural elements are represented in a great level of detail, do not permit to consider large systems such as P2P. Recently, fluid models have been proposed as a viable approach to capture P2P long-term system dynamics. Fluid models represent the dynamics of 'average' network metrics (such as the average number of users storing a content, the average number of contents stored by users, etc.) through a set of ordinary differential equations. The resulting set of differential equations is then solved either analytically (when possible) or numerically. By so doing, however, the stochastic short-term fluctuations in the system are not represented. Thus, many important phenomena related to stochastic effects of the system dynamics are irremediably lost.

In this article, we propose a second-order 'fluid-diffusive' approximation of system dynamics based on a set of partial differential equations that allow us to capture the impact of stochastic effects on the system dynamics. The diffusion approximation, which consists in locally approximating the system dynamics with generalized (anisotropic) Brownian motions, matching the first two moments of the original process, has been proved to be effective in terms of both accuracy and computational efficiency in several application contexts [9,10–16].

1.1 Related work

In this section, we briefly summarize previous work on analytical modelling of P2P systems that are related to our approach.

The works in [17–19] present a simple technique for the performance evaluation of search strategies in large-scale decentralized unstructured P2P applications. The authors use the generalized random graph (GRG) methodology to model a snapshot of the P2P overlay topology. The analysis of GRG models is efficiently accomplished using the generating function of the GRG's degree distribution. The analytical framework allows to evaluate several different search strategies including flooding, probabilistic flooding, distance-dependent probabilistic flooding and any combination of these strategies. The works in [20–22], which explore the performance of BitTorrent-like P2P systems, are closer to our approach. The work in [20] considers users' and contents' diffusion dynamics, which are modelled by age-dependent branching processes so as to analyse the transient dynamics of the system in terms of service capacity. This approach exactly captures the ability of this type of P2P systems to face flash crowds. Furthermore, this article presents a separate Markovian model for the steady-state regime. The work in [21] shares several assumptions with [20]. The differences between the two mostly reside in the modelling techniques and the selected performance metrics. In particular, [21] presents a completely deterministic fluid model of BitTorrent aimed at studying the steady-state regime (i.e. the dynamical equilibrium point of the system), obtaining insights into the average number of downloaders (peers who have only a fraction of the resource), average number of seeds (peers who have the complete resource) and average download time as function of parameters such as peer arrival rate, downloaders leaving rate, seed leaving rate and upload bandwidth. In [22], the authors obtain asymptotic differential equations describing data replication in BitTorrent-like systems and provide interesting insights into the impact of different replication strategies on download time.

1.2 Article contribution

In our work, we adopt an approach similar to [20] and [21]. Although our model is not tailored to a particular P2P system, it describes the evolution of a generic unstructured P2P system trying to abstract from the particular P2P application. In this manner, the model tries to provide answers whose validity are somehow general. Furthermore, we develop a methodology to study within a single framework both transient and stationary behaviours of P2P systems, describing with a fairly good degree of accuracy many important dynamical effects that are lost in pure fluid models. In particular, in contrast to previous models presented in [20] and [21], our methodology allows to investigate the joint effect of several phenomena related to peer behaviour, content localization algorithm and dynamic structure of the overlay topology.

Another remarkable difference concerns the modelling technique used: we propose a second-order ‘fluid-diffusive’ approximation through partial differential equations that enables us to obtain fundamental distributions related to contents and users and, thus, account for stochastic effects of the system dynamics that cannot be captured by first-order modelling approaches. Moreover, our ‘fluid-diffusive’ approximation can relate the content diffusion to the content search and download processes, thus permitting to consider the impact of different elements of a P2P architecture in a modular fashion. For these reasons, we believe that the proposed approach is promising and worth further investigations.

Summarizing the contribution of the article is threefold:

(1) we have decomposed the dynamics of a generic unstructured P2P system into those of three sub-systems that can be analytically treated, identifying how these sub-systems interact;
(2) we have shown how each of the three sub-systems can be relatively easily modelled in both transient and stationary conditions using a second-order ‘fluid-diffusive’ approximation;
(3) we have used our model to show how parameters and design choices may affect the system performance.

2. Modelling unstructured P2P systems: an overview

Unstructured P2P systems exhibit very high dynamism resulting from complex interactions among users, contents and underlying overlay topology. User and content dynamics are

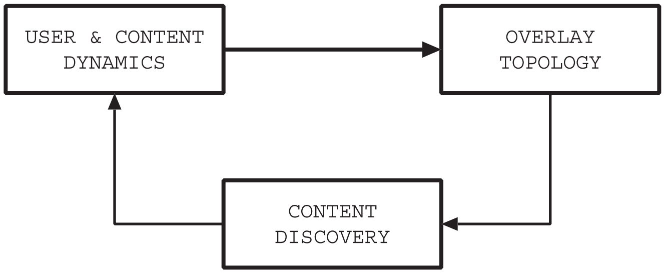

Figure 1. A schematic representation of our model.

closely tied, because users continuously retrieve new contents based on their availability and remove contents already in their possession. Moreover, users continuously join and leave the system, modifying the structure of the overlay topology. When connected to the system, they discover other peers, thus establishing new connections, which change the underlying network graph. On the other hand, the availability of contents within the overlay topology has an impact on the contents' spreading itself, because users can retrieve only those contents that are made available by the content search algorithm, which usually relies on a partial exploration of the overlay topology.

To find a reasonable trade-off between the complexity of the model and its ability to represent the dynamics under study, we propose the decomposition approach depicted in Figure 1. First, we describe the joint evolution of user and content dynamics assuming that the rate at which users retrieve new contents from the system is known. We separately model the impact of user and content dynamics on the overlay network structure through a system of equations, which allows to obtain the distribution of nodes' degree (i.e. the number of connections established with other peers). We can then relate the effectiveness of the content search algorithm to the nodal degree distribution of the overlay topology, obtaining the probability $p_{\text{hit}}(n)$ to localize a content in the network that is stored by n users. Finally, the impact of the content search algorithm is introduced back into the main equations of user and content dynamics using the above probability $p_{\text{hit}}(n)$. The three models jointly evolve over time, allowing to study the transient behaviour of a P2P system in an integrated way.

3. The main fluid-diffusive model

In this section, we first introduce our basic model describing user and content dynamics under two simplifying assumptions that will be removed later: (i) ideal search (i.e. a content is always found if there is at least one copy of it stored by an active user) and (ii) instantaneous download (i.e. the bandwidth available in the physical network is infinite). Thus, our model captures the macroscopic behaviour of users and contents in an ideal system; in Section 5, we will explain how the basic model can be extended to consider the limitations of a real system, by incorporating the effects of content search and download.

We look at the system from two different perspectives: the point of view of users, described in Section 3.1, and the point of view of contents, described in Section 3.2. For the sake of readability, in Tables 1–3 we report the notations corresponding to all parameters and state variables in the system.

In general, the evolution of the number of contents stored by a dynamic population of users can be described by a complex Markovian process over a general state space.

Table 1. Physical parameters of user behaviour.

Symbol	Meaning (referred at time t)
$\lambda_u(t)$	Rate at which new users subscribe the P2P system
$1/\mu_u(t)$	Average time for which a user subscribes the P2P system
$1/\mu_{as}(t)$	Average duration of users' active periods
$h_{as}(t)$	Variation coefficient of active periods
$1/\mu_{sa}(t)$	Average duration of users' sleeping periods (i.e. periods of user inactivity)
$h_{sa}(t)$	Variation coefficient of sleeping periods
$\theta(t)$	Rate at which a given content is requested by a user
$h_r(t)$	Variation coefficient of the content inter-request time
$1/\mu_h(t)$	Average time during which a user holds the content (during both active and sleeping periods)
$h_h(t)$	Variation coefficient of holding time
$\lambda_c(t)$	Rate at which new contents are made available (i.e. shared) by an active user

Table 2. Overview of the main state variables in the system.

Symbol	Meaning (referred at time t)
$G(x,t)$	Number of users storing x contents
$G_a(x,t)$	Number of active users storing x contents
$G_s(x,t)$	Number of sleeping users storing x contents
$F(x,t)$	Number of contents available in x copies
$E(x,t)$	Number of active users whose nodal degree is x in the overlay topology

Table 3. Overview of the main derived state variables in the system.

Symbol	Meaning (referred at time t)
$C_T(t)$	Number of contents available in the system
$U(t)$	Total number of users
$U_a(t)$	Number of active users
$U_s(t)$	Number of sleeping users
$O_a(t)$	Number of copies stored by active users
$O_s(t)$	Number of copies stored by sleeping users

When all of the random variables describing the user behaviour are exponentially distributed, the stochastic process describing the users' evolution degenerates into a standard Markov chain. Although more general distributions can be approximated by Coxian or phase-type distributions, the numerical solution of the resulting Markovian model may be computationally very expensive (especially when transient solutions are required). For these reasons, in this article we propose a diffusion approximation to model the dynamics of contents and users in large P2P networks. We remark that according to the adopted diffusion approximation, all the modelled quantities are relaxed to the continuous.

3.1 User dynamics

Users dynamically *subscribe* and *unsubscribe* the system: we assume that new users arrive according to a stochastic process with rate $\lambda_u(t)$ and leave the system after a period of time randomly distributed with mean $1/\mu_u(t)$. During the subscription period, users alternate phases in which they are either *active* or *sleeping*; the durations of active and sleeping periods are assumed to be i.i.d. random variables with means $1/\mu_{as}(t)$ and $1/\mu_{sa}(t)$, respectively. Active users are connected and retrieve new contents from other active users in the system; at the same time, they share the contents stored by them. Sleeping users, instead, are not connected and do not interact in any way with the other users of the community. We assume that, when active, users search for a given content available in the network at rate $\theta(t)$. Notice that, in this ideal model, this is also the rate at which a given content is successfully retrieved by a user. We allow the inter-request time to have a generic distribution with coefficient of variation $h_r(t)$. A user holds the copy of a content for a time period with mean $1/\mu_h(t)$ and coefficient of variation $h_h(t)$. Finally, the user makes new contents available to the community at rate $\lambda_c(t)$. The parameters used to describe the user behaviour are summarized in Table 1. We remark that we allow all of them to vary over time according to the given functions of time. This is indeed one of the main strengths of our model, that is, the ability to study the transient behaviour of non-stationary P2P systems. In [23], a measurement study of user dynamics has been reported that provides useful indications on how to set the above parameters.

In the following, we describe the dynamics of a population of users having the same behaviour (i.e. the same set of parameters as reported in Table 1). Diversity in user behaviour is taken into account in our model introducing different classes of users. We focus now on the dynamics of a single class of users. In particular, we first consider the case of a constant population of users always connected to the system. Let $G(x,t)$ be the distribution representing the number of users storing x contents, at time t. We wish to model how this distribution evolves over time. Each user is representing as a moving particle whose instantaneous position $x(t)$ represents the number of contents a user is currently holding.

The dynamics of each user-particle can be exactly represented by a Markovian process over a general space state, whose analysis, however, appears rather prohibitive. For the above reasons, we adopt a second-order diffusion-approximated description of the user-particles' movements, according to which user-particles' trajectories are described by independent, in homogeneous Brownian motions. We wish to emphasize that because trajectories of Brownian motion are continuous, our approximation requires to relax, to continuous, quantities that are discrete in nature, such as the amount of contents stored by the users.

We notice that our approximation is able to exactly capture the first two moments of the instantaneous virtual speed for user-particles that at time t are in position x, while possibly deviating from exact values when considering higher order statistics.

The evolution of a large population of user-particles, each one moving according to an independent Brownian motion (as commonly assumed in statistical-physics), can be described by the well-known Fokker–Planck partial differential equation:

$$\frac{\partial G(x,t)}{\partial t} = -\frac{\partial m(x,t)G(x,t)}{\partial x} + \frac{1}{2}\frac{\partial^2 \sigma^2(x,t)G(x,t)}{\partial x^2} \tag{1}$$

Note that when $\sigma^2(x,t)$ is set to zero, the Fokker–Planck equation degenerates in a first-order wave equation, in which $m(x,t)$ represents the instantaneous speed at time t of the particles in x. The presence of the speed variance $\sigma^2(x,t)$ induces a diffusion effect represented in (1) by the second-order term.

In our model, we need to consider both active and sleeping users, and the fact that users join and leave the system at arbitrary points in time, besides alternating between the active and sleeping phases. Taking these dynamics into account, we obtain the following set of coupled fluid-diffusive equations (a subscript a or s is used to distinguish between active and sleeping users):

$$\frac{\partial G_{\rm a}(x,t)}{\partial t} = -\frac{\partial m_{\rm a}(x,t)G_{\rm a}(x,t)}{\partial x} + \frac{1}{2}\frac{\partial^2 \sigma_{\rm a}^2(x,t)G_{\rm a}(x,t)}{\partial x^2} + \\ - [\mu_{\rm as}(t) + \mu_{\rm u}(t)]G_{\rm a}(x,t) + \mu_{\rm sa}(t)G_{\rm s}(x,t) + \lambda_{\rm u}(t)\delta(x) \tag{2}$$

$$\frac{\partial G_{\rm s}(x,t)}{\partial t} = -\frac{\partial m_{\rm s}(x,t)G_{\rm s}(x,t)}{\partial x} + \frac{1}{2}\frac{\partial^2 \sigma_{\rm s}^2(x,t)G_{\rm s}(x,t)}{\partial x^2} + \\ - [\mu_{\rm sa}(t) + \mu_{\rm u}(t)]G_{\rm s}(x,t) + \mu_{\rm as}(t)G_{\rm a}(x,t) \tag{3}$$

defined for $x \in [0, C_{\rm T}(t)]$, where $C_{\rm T}(t)$ is the number of contents available in the system at time t. Functions $m_{\rm a}(x,t)$ and $\sigma_{\rm a}^2(x,t)$ refer to active users, whereas $m_{\rm s}(x,t)$ and $\sigma_{\rm s}^2(x,t)$ refere to sleeping users; all of them are defined below. Besides the two terms that appear also in a classical Fokker–Planck equation, the last three terms on the right-hand of (2) represent the flow of users who either become inactive or unsubscribe the system, the flow of users who become active and the flow of newcomers entering the system for the first time. The last two terms on the right-hand of (3) represent the flow of users who either become active or unsubscribe the system and the flow of users who become inactive.

For simplicity, we assume that newcomers join the system in the active phase without storing any content. This explains the term $\lambda_{\rm u}(t)\delta(x)$ in (2). However, one could easily model the situation in which users who subscribe the system start sharing an initial number of contents arbitrarily distributed.

We need to specify parameters $m(x,t)$ and $\sigma^2(x,t)$ for both active and sleeping users.[1] As already observed, $m(x,t)$ represents the average speed at which a user-particle moves along the x axes, which can be expressed, in general, as the difference between the rate $r(x,t)$ at which a user acquires new contents and the rate $d(x,t)$ at which contents are removed by the user. Active users acquire new contents by retrieving them from other peers and by introducing new contents themselves. Thus, we have

$$r_{\rm a}(x,t) = \theta(t)\,[C_{\rm T}(t) - x] + \lambda_{\rm c}(t) \tag{4}$$

Notice that we assume that the search rate of a user is proportional to the number of available contents that the user does not currently hold; however, more complex dependencies of the user content retrieval rate on the number of stored contents can be considered well in our model.

Sleeping users do not acquire new contents, thus $r_{\rm s}(x,t) = 0$. The removal rate is the same for both active and sleeping users, and it is proportional to the number of contents currently held: $d_{\rm a}(x,t) = d_{\rm s}(x,t) = \mu_{\rm h}(t)x$. Finally, the impact of variation coefficients is taken into account using the approach proposed in [9], obtaining

$$m_{\mathrm{a}}(x,t) = \frac{2}{1+h_{\mathrm{h}}^2(t)}[r_{\mathrm{a}}(x,t) - d_{\mathrm{a}}(x,t)]$$

$$m_{\mathrm{s}}(x,t) = \frac{2}{1+h_{\mathrm{h}}^2(t)}[-d_{\mathrm{s}}(x,t)]$$

$$\sigma_{\mathrm{a}}^2(x,t) = \frac{2}{1+h_{\mathrm{h}}^2(t)}\left[\theta(t)\,[C_{\mathrm{T}}(t) - x]\,\frac{2h_{\mathrm{r}}^2(t) + h_{\mathrm{h}}^2(t) - 1}{h_{\mathrm{h}}^2(t) + 1} + \lambda_{\mathrm{c}}(t) + d_{\mathrm{a}}(x,t)\right]$$

$$\sigma_{\mathrm{s}}^2(x,t) = \frac{2}{1+h_{\mathrm{h}}^2(t)}[d_{\mathrm{s}}(x,t)]$$

Note that all quantities depend on the second moment of the holding time of the copies, through the variation coefficient $h_{\mathrm{h}}(t)$. The variation coefficient of inter-request time $h_{\mathrm{r}}(t)$, instead, affects only the speed variance of active users.

The number of active users at time t is $U_{\mathrm{a}}(t) = \int G_{\mathrm{a}}(x,t)dx$, whereas the number of sleeping users at time t is $U_{\mathrm{s}}(t) = \int G_{\mathrm{s}}(x,t)dx$. The total number of users is $U(t) = U_{\mathrm{a}}(t) + U_{\mathrm{s}}(t)$. We also introduce the total number of copies stored by active users $O_{\mathrm{a}}(t) = \int xG_{\mathrm{a}}(x,t)dx$ and the total number of copies stored by sleeping users $O_{\mathrm{s}}(t) = \int xG_{\mathrm{s}}(x,t)dx$.

3.1.1 Special case

To better understand the dynamics expressed by the fluid-diffusive equations above, we consider a special case corresponding to the dynamics of a classical queueing system. Indeed, we can assume that all the parameters in Table 1 do not depend on t and that the number of contents is fixed: $C_{\mathrm{T}}(t) = C_{\mathrm{T}}$ and $\lambda_{\mathrm{u}} = 0$. Users remain active forever. Now the content acquisition rate (4) becomes $r_{\mathrm{a}}(x) = (C_{\mathrm{T}} - x)\theta$ and the content removal rate becomes $\mu_{\mathrm{h}}x$. Furthermore, content request events form a Poisson process and content holding times are exponentially distributed. Hence, the system is modelled as a Continous Time Markov Chain (CTMC), represented in Figure 2, in which state x represents the number of distinct contents available in each active user in the P2P system. Such CTMC is the same as an $M/M/C_{\mathrm{T}}$ queue with a finite population of C_{T} users arriving at rate θ. Our approach for this special case consists of describing the CTMC evolution using a second-order approximation, following the same approach as described in [24].

3.2 Content dynamics

From the point of view of contents in the network, we are interested in the number of copies of each content that are stored by active users at a given point in time. Similar to users' dynamics, we model contents' dynamics through a second-order diffusion approximation:

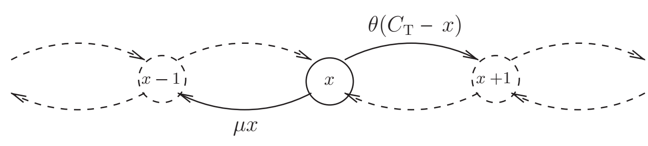

Figure 2. Transition diagram of the CTMC corresponding to the special case in which the system dynamics degenerate into a $M/M/C_{\mathrm{T}}$ queue; state x represents the number of contents available in each active user.

each content in the system is modelled as a moving particle whose instantaneous position $x(t)$ represents the number of available copies at time t (i.e. the number of active users who are storing a copy of the considered content). Hence, we can describe the evolution of the number $F(x, t)$ of contents available in x copies in the system by a Fokker–Planck equation:

$$\frac{\partial F(x,t)}{\partial t} = -\frac{\partial m(x,t)F(x,t)}{\partial x} + \frac{1}{2}\frac{\partial^2 \sigma^2(x,t)F(x,t)}{\partial x^2} + U_a(t)\lambda_c(t)\delta(x-1) \tag{5}$$

defined for $x \in [0, U_a(t)]$. The term $U_a(t)\lambda_c(t)\delta(x-1)$ represents contents newly introduced by users (thus initially available in just one copy).

The mean and variance of the content-particles' speed along x can be expressed as the difference between the effective rate $r_c(x, t)$ at which new copies are made available and the effective rate $d_c(x, t)$ at which available copies are removed from the system.

The effective rate $r_c(x, t)$ is given by the sum of two contributions: (1) the rate at which new copies of the considered content are made available by active users retrieving it and (2) the rate at which new copies of the considered content become available when sleeping users storing it transit to the active phase making again available to the community their contents. We do not model the content diffusion among the sleeping users and we resort to a simple approximation to compute contribution (2). Indeed, we have

$$r_c(x,t) = \theta(t)[U_a(t) - x] + \mu_{sa}(t)\frac{O_s(t)}{C_T(t)} \tag{6}$$

The effective removal rate $d_c(x, t)$ of a content is the sum of the rate at which copies of the considered content are cancelled by active users and the rate at which active users storing it transit to the sleep state or abandon the system:

$$d_c(x,t) = x[\mu_h(t) + \mu_{as}(t) + \mu_u(t)]$$

We obtain

$$m_c(x,t) = \frac{2}{1+h_h^2(t)}(r_c(x,t) - d_c(x,t))$$

$$\sigma_c^2(x,t) = \frac{2}{1+h_h^2(t)}(r_c(x,t) + d_c(x,t))$$

Note that the request process of a content is given by the superposition of many independent request processes of individual users; hence, the resulting process tends to Poisson, and the impact of the variation coefficient of inter-request time for individual users can be neglected.

The rate $\nu_0(t)$ at which contents disappear from the network is given by

$$\nu_0(t) = \frac{1}{2}\left.\frac{\partial \sigma^2(x,t)F(x,t)}{\partial x}\right|_{x=0} \tag{7}$$

The evolution of the total number of available contents $C_T(t)$ is given by

$$\frac{dC_T(t)}{dt} = U_a(t)\lambda_c(t) - \nu_0(t)$$

which can formally be obtained integrating (5) with respect to x. The total number of available contents in the system is

$$C_{\mathrm{T}}(t) = \int_0^{U(t)} F(x,t)dx$$

4. Topology dynamics

Here, we show how the evolution of the overlay topology can also be described through a fluid-diffusive equation. The goal is to obtain the distribution of nodes' degree, that is, the distribution of the number of logical connections maintained by a peer.

A node in the overlay topology corresponds to an *active* P2P user. As soon as a user becomes inactive it disappears from the overlay topology, eventually reappearing later when it becomes active again.

During the activity period, users maintain a list of neighbouring nodes, which is dynamically updated, removing the address of nodes that are no longer active and acquiring addresses of new nodes. Several mechanisms (e.g. the ping-pong mechanism in Gnutella) have been defined to dynamically acquire the information on new neighbours during the activity period of a user. To simplify our model, we assume that users who become inactive loose memory of their neighbour list. This assumption can be justified because even if a peer stores the list of his or her previous neighbours when the peer becomes again active his or her neighbours might not be active anymore. Let $E(x,t)$ be the number of active users whose nodal degree (i.e. their neighbour list length) at time t is x. We can write

$$\begin{aligned}\frac{\partial E(x,t)}{\partial t} = &- \frac{\partial m_{\mathrm{e}}(x,t)E(x,t)}{\partial x} + \frac{1}{2}\frac{\partial^2 \sigma_{\mathrm{e}}^2(x,t)E(x,t)}{\partial x^2} + \\ &- [\mu_{\mathrm{as}}(t) + \mu_{\mathrm{u}}(t)]E(x,t) + [\mu_{\mathrm{sa}}(t)U_{\mathrm{s}}(t) + \lambda_{\mathrm{u}}(t)]D_{\mathrm{i}}(x)\end{aligned} \tag{8}$$

where $D_i(x)$ is the distribution of the number of addresses initially given to a node either joining the network for the first time or becoming active after a sleeping period (this distribution is assumed to be known). Now turning our attention to $m_{\mathrm{e}}(x,t)$ and $\sigma_{\mathrm{e}}^2(x,t)$, it results:

$$m_{\mathrm{e}}(x,t) = \frac{2}{1 + h_{\mathrm{as}}^2(t)}[\lambda(x,t) - x\mu_{\mathrm{as}}(t) - x\mu_{\mathrm{u}}(t)]$$

$$\sigma_{\mathrm{e}}^2(x,t) = \frac{2}{1 + h_{\mathrm{as}}^2(t)}[\lambda(x,t) + x\mu_{\mathrm{as}}(t) + x\mu_{\mathrm{u}}(t)]$$

where $\lambda(x,t)$ is the rate at which users already having x neighbours acquire new neighbours. The expression of $\lambda(x,t)$ depends on the specific mechanism for acquiring new neighbours. In this sense, our model is quite general; it permits, for example, to account for the *preferential attachment* phenomenon, by making $\lambda(x,t)$ proportional to x. For simplicity, we consider only the case of a flat, unstructured P2P system like the original Gnutella network, for which $\lambda(x,t)$ can be expressed as

$$\lambda(x,t) = p_{\mathrm{reach}}(x)[\mu_{\mathrm{sa}}(t)U_{\mathrm{s}}(t)\bar{D}_i(t) + \lambda_{\mathrm{ping}}(t)U_{\mathrm{a}}(t)]$$

The term in square brackets represents the aggregate rate at which new connections are established in the system: $\bar{D}_i(t)$ is the average degree of sleeping users when they become active again, $\lambda_{\text{ping}}(t)$ is the rate of new connections established by ping messages sent by an active user and $p_{\text{reach}}(x)$ is the probability that a user already having x connections is selected by a peer activating a new connection. The latter probability can be evaluated using random graphs techniques (see [17]). We observe that $p_{\text{reach}}(x)$ exhibits a dependency on x because users having more neighbours are more likely selected by other peers.

With a slightly different approach, we can evaluate $\hat{E}(x,t)$, the nodal distribution given the time t spent in the active phase.

The equation driving $\hat{E}(x,t)$ is

$$\frac{\partial \hat{E}(x,t)}{\partial t} = -\frac{\partial m_{\text{e}}(x,t)\hat{E}(x,t)}{\partial x} + \frac{1}{2}\frac{\partial^2 \sigma_{\text{e}}^2(x,t)\hat{E}(x,t)}{\partial x^2} \tag{9}$$

under the condition $\hat{E}(x,0) = D_i(x)$.

From $\hat{E}(x,t)$, the steady-state degree distribution of a peer $E_{\text{s}}(x)$ can be accurately related to its distribution of active phases.

Let $f_{\text{a}}(\tau)$ be the distribution of the duration of active period for a generic peer and $f_{\text{l}}(\tau)$ the stationary distribution of the time spent in active period by peers. From standard renewal theory, we know that

$$f_{\text{l}}(\tau) = \frac{\int_{\tau}^{+\infty} f_{\text{a}}(t)dt}{\int_0^{+\infty} t f_{\text{a}}(t)dt} \tag{10}$$

and we can compute the steady-state degree distribution of an active peer as

$$E_{\text{s}}(x) = \int_0^{+\infty} f_{\text{l}}(\tau)\hat{E}(x,\tau)d\tau \tag{11}$$

5. Search and download phases

The basic model introduced in Section 3 can be extended to capture other typical characteristics of a real P2P system by properly setting $m(x,t)$ and $\sigma^2(x,t)$. Here, we consider the effects of the search phase and of the download process. The main observation is that the actual rate at which contents are successfully transferred among users is affected by

- the probability of a successful search, $p_{\text{hit}}(x,t)$, which depends on the content diffusion among the active users; to compute this probability, we have adopted the statistical physics model in [17], which is based on the generating function method for random graphs.
- the probability of a successful download, $p_{\text{down}}(t)$, which depends on several factors, such as the network congestion (number of concurrent downloads), the user impatience and the probability that the download is interrupted because the server switches to the sleeping state or leaves the system. We have adopted a processor-sharing model at the server side to estimate the average download delay, which is then used to compute $p_{\text{down}}(t)$.

Both effects can be incorporated in (6) (and similarly in (4)) with the following modification:

$$r(x,t) = \theta(t)[U_{\mathrm{a}}(t) - x]p_{\mathrm{hit}}(x,t)p_{\mathrm{down}}(t) + \mu_{\mathrm{sa}}(t)\frac{O_{\mathrm{s}}(t)}{C_{\mathrm{T}}(t)} \qquad (12)$$

In the following Sections 5.1 and 5.2, we will present the models to capture the search and the download phases, respectively.

5.1 Content search algorithm

In an unstructured P2P system, a content is accessible only if the search mechanism is able to localize at least one copy of it inside the network. Of course, the probability of hitting a content (i.e. localize at least one copy of it) depends on the particular content search mechanism used in the system.

In this article, we consider a simple flood-based mechanism over a flat overlay topology; we emphasize, however, that our approach can be extended to deal either with hierarchical overlay topologies [19] in which two types of users with different functionalities coexist (e.g. simple peers and super-nodes), or with different searching mechanisms based on random walk, probabilistic flooding or hybrid flooding, explorations of the overlay topology.

The modelling methodology used in this article to compute $p_{\mathrm{hit}}(x,t)$ is based on the exploitation of *Generalized Random Graphs* (GRG) to describe the overlay topology, introduced in [17]. Here we only provide some high-level considerations of how the methodology works and refer the reader to [17] for further details.

GRGs are stochastic models for collections of homogeneous large graphs. A GRG with N nodes is completely specified by the degree distribution of nodes, that is, the probability that a randomly chosen node has a certain number of edges emanating from it. This distribution can be related to user behaviour and application-specific details using the model described in Section 4.

In [17], it is shown how to obtain the generating function of the number of peers who receive a query message for very general search mechanisms. Furthermore, assuming that each node in the network stores a particular file with probability p (in our context it results $p = x/U_{\mathrm{a}}$ being x the number of copies of the file available in the network), it is possible to obtain an expression for the probability that at least one node storing the considered file is hit by a flooding search mechanism with assigned TTL.

At last, we remark that it is possible to adopt other simple analytical approaches to evaluate $p_{\mathrm{hit}}(x,t)$ when the search strategy is based on controlled random walks [25].

5.2 Download phase

When modelling the content download process, we consider an ideal transport network, that is, no congestion arises in the IP backbone, but only at network edges. Thus, the download time in our model depends only on the limited capacity (bandwidth) of the peers. An accurate estimate of the download process should take into account the dynamics of the available bandwidth of both peers who are taking part in the content transfer, as done in [26].

We neglect the effect on the content transfer time of the limited bandwidth available at the client side, considering only the impact of limited bandwidth available at the server side. The dynamics of uploads at each peer of the network can be modelled by an M/G/1 processor-sharing queue, reasonably assuming that the request process incoming to each peer is Poisson.

In a P2P system, however, the download request rate is not uniformly distributed among peers but depends on both the distribution of contents stored at peers and the server selection policy. In this article, we assume that a random load-balancing policy is adopted, that is among all peers sharing the desired content that is hit by the search mechanism, one is picked at random and selected as server for the file transfer.

Under random load-balancing policy, we expect that all the peers storing the same amount of contents experience the same average request rate. Moreover, the incoming request rate is proportional to the number of contents stored by a peer. Consider a content held by x users, then $\theta(t)p_{\text{hit}}(t,x)[U_{\text{a}}(t)-x]$ represents its download request rate. As this rate is evenly distributed among the peers storing the content, the incoming rate of requests for a given content at a peer holding it can be expressed as

$$\frac{\theta(t)p_{\text{hit}}(t,x)[U_{\text{a}}(t)-x]}{x}$$

Then, averaging with respect to the number of available copies x of contents in the network, which is distributed according to $F(x,t)$, we obtain the aggregate request rate arriving at a user for each stored content

$$\lambda_{\text{down}}(t)=\int_x\left[\frac{\theta(t)p_{\text{hit}}(t,x)[U_{\text{a}}(t)-x]}{x}\right]F(x,t)dx$$

Finally, for a user who is storing y contents, the aggregate incoming request rate is proportional to y. However, users storing more contents are more likely to be addressed by download requests; the frequency at which download requests are addressed towards users storing y contents is then expressed by the following pdf:

$$\frac{yG_{\text{a}}(y,t)}{E[y]U_{\text{a}}(t)}$$

where $E[y]U_{\text{a}}(t)=\int yG_{\text{a}}(y,t)dy$.

Let $T_{\text{down}}(\lambda)$ be the average download time for a M/G/1 processor-sharing queue expressed as a function of λ, the arrival rate at the queue. The average download time for a content in the network is then

$$E[T_{\text{down}}](t)=\int_y T_{\text{down}}(y\lambda_{\text{down}})\frac{yG_{\text{a}}(y,t)}{E[y]U_{\text{a}}(t)}dy$$

Similarly, given the average arrival rate λ of requests at the server, the probability that a download fails because the server disconnects from the P2P system before the end of the download, can be expressed analytically as a function of the server average residual time in the active state, and of the distribution of job completion time resulting from the M/G/1 processor-sharing queue.

Let $p_d(\lambda)$ be the probability that a download request directed towards a peer who is experiencing a request rate λ is successful; it can be analytically related to the distribution of $T_{\text{down}}(\lambda)$ and to the distribution of the peer active period. By averaging over the request arrival rate λ, the average probability that a download in the network is successful can be obtained as

$$p_{\text{down}}(t) = \int_y p_{\text{d}}(y\lambda_{\text{down}}) \frac{yG_{\text{a}}(y,t)}{E[y]U_{\text{a}}(t)} dy$$

6. Model validation

In this section, we validate our model comparing analytical predictions with simulation results obtained from an ad hoc event-driven simulator built at the application layer [27]. The simulator accounts for the detailed behaviour of all users and contents, keeping track of each individual user and of each individual copy in the network (there are no fluid/diffusion approximations in the simulation) and explicitly describing the dynamics of each server as a processor-sharing queue. Because of that, we have been able to consider only limited-size systems (up to a few thousands of users) while comparing analytical and simulation results. We remark that the model, instead, does not suffer from scalability problems and could be used to analyse much larger P2P systems as those found in the real world (having millions of users).

The simulator does not represent explicitly the underlying physical network topology, hence it shares with the model the same assumptions about the search phase and the successfully probability $p_{\text{hit}}(n)$ of finding a content in the network available in a given n number of copies. The simulator code and additional details on it can be found in [27].

To show the potentialities of our approach, we selected a set of performance measures that cannot be obtained by analytical models of P2P systems that previously appeared in the literature.

6.1 User dynamics

We start considering the dynamics of a large population of users characterized by the following parameters: the average duration of the active period is equal to 2 hours, whereas the average duration of the sleeping period is 48 hours. During the active periods users request new contents at an aggregate rate of five contents per hour. We assume, for now, that all requests are successful and that users eventually download the requested files. The average content-holding time is set to 3 days. New users subscribe the system at a rate of 14/hour, and the average stay into the system is assumed to be equal to 1 month. Users start holding zero contents and do not add new contents themselves.

Our fluid-diffusive equations allow to estimate the number of active or sleeping users storing a given number of contents. In particular, it permits to account for the effect of different distributions of the inter-arrival time between successive content requests and of the content-holding time, through the variation coefficients h_{r} and h_{h}.

Figure 3 reports the steady-state distribution $G_{\text{a}}(x,t)$ of the number of active users storing a given number of contents, for various combinations of the above variation coefficients. Markers refer to simulation results, whereas the lines closest to these refer to the corresponding analytical results. Shaded areas represent the 95%-level confidence interval for the distributions obtained by simulation (i.e. each shaded area is the ensemble of the confidence intervals obtained for each point of the distribution). First, we notice that model and simulation results are in good agreement; this is not surprising because model and simulator share the same values of parameters describing the user behaviour. From

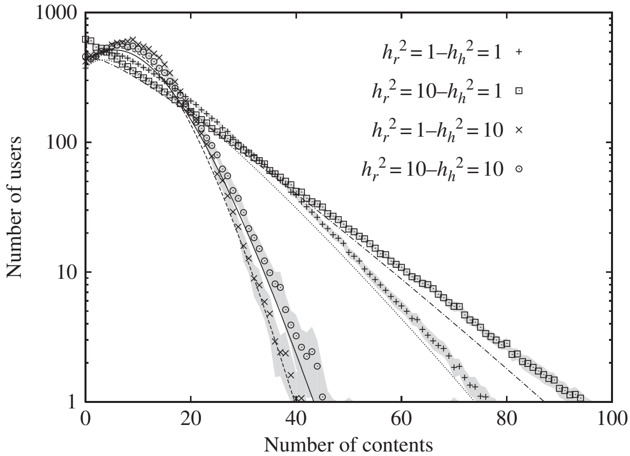

Figure 3. Distribution of the total number of users storing a given number of contents. Markers correspond to simulation results obtained for four different combinations of variation coefficients, whereas the lines closest to these refer to the corresponding analytical results. Shaded area represent 95%-level confidence interval for simulation results.

Table 4. Average number of contents stored by an active user for various combinations of variation coefficients.

h_r^2	h_h^2	Model	Simulation
1	1	22.05	21.98
10	1	23.99	22.86
1	10	10.72	10.98
10	10	11.12	11.29

the curves, it can be observed that distribution $G_a(x, t)$ is strongly influenced by the coefficients of variations. Not only are the tails of the distributions different but also the average number of contents stored by a user is different. In particular, it is strongly affected by the coefficient of variation of the holding time, whereas it does not depend on the of variation coefficient of the inter-request time, as reported in Table 4 and confirmed by simulation. An accurate prediction of the distribution of the number of contents available at a user is important because it directly affects its load and thus the download time of other users requesting contents from it. We conclude that models based only on the average values of content-holding time and inter-request time cannot fully characterize the resulting user behaviour; thus, more accurate models (like ours) capturing the impact of higher moments of these quantities are needed.

6.2 Content survivability

To validate our diffusion approximation of contents dynamics, we consider the case of a single content that is introduced in the P2P system at time $t = 0$ in only one copy (by an active user). We are interested in studying the probability that the content is still present in the system at a generic time t, that is, its survivability function. Actually, two basic scenarios can happen: (1) the content prematurely dies out after being removed by all users storing it or (2) the content is able to propagate in the network, being replicated in a sufficiently large number of copies to make the probability of disappearing from the system negligible. It turns out that the most critical period that affects the survivability of the content in the system are the early stages of its propagation. In particular, if the single user storing the content at time zero decides to cancel his or her copy before it is successfully downloaded by another user, the content immediately disappears from the system. The transient phase of the diffusion of a new content was studied in [20] using the theory of branching processes. There, the authors neglect the probability that the content disappears from the system and focus on the rate at which it can spread in the network. Here, instead, we study the probability that the content is present in at least one copy after a given period of time, accounting for the removal of copies by the users. We consider a large population of users characterized by the same parameters used in the scenario of Section 6.1. The aggregate request rate of active users for the considered content is set equal to $1/(8\,\text{hours})$. We expect the variation coefficient h_{h} of copy-holding time to play a significant role also in this scenario. This is indeed the case, as suggested by the results in Figure 4 obtained for $h_{\text{h}}^2 = 1$ and $h_{\text{h}}^2 = 10$. The plot reports the cumulative probability that the content disappears from the system by time t, as a function of time. The shaded areas represent the 95%-level confidence interval for the distributions obtained by simulation. We observe that when $h_{\text{h}}^2 = 10$, the content immediately disappears from the system at the very beginning with high probability (about 0.8). This is due to the fact that the first copy is deleted by the owner before other peers can download it. When $h_{\text{h}}^2 = 1$, the copy is much more likely to survive the initial critical phase of content spreading. After that, the CDF of content lifetime grows very slowly (tends to 1 as time goes to infinity), because there is always a non-null probability to go to the state in which there are zero copies in the system, given by (7). The model is able to predict this behaviour quite well.

6.3 Content search and download

In this section, we consider the joint effect of content search and download on the system performance. Recall that these effects are accounted for in the basic model by probabilities $p_{\text{hit}}(x,t)$ and $p_{\text{down}}(t)$ in (12).

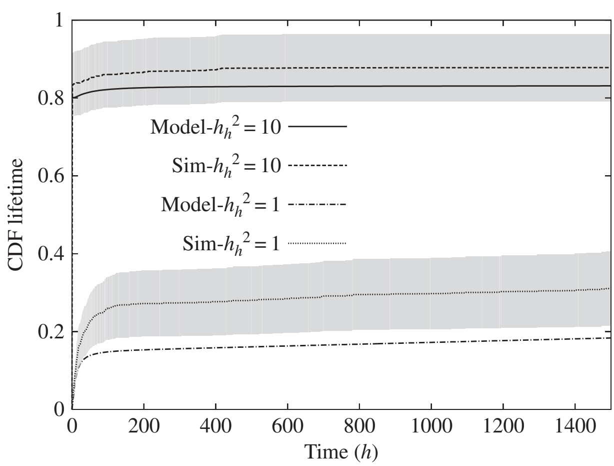

Figure 4. CDF of the content lifetime.

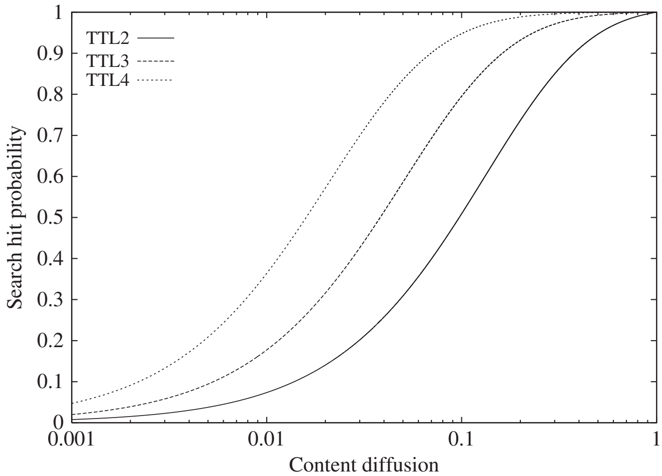

Figure 5. Values of p_{hit} as a function of the content diffusion, used to analyse the impact of imperfect searches.

We measure the system performance by looking at the number of copies of a specific content that are stored by all users. This number is directly related to the probability that users can retrieve the considered content from the network, thus it can be used as an indication of the users' satisfaction in using the P2P application. We consider a system in which the number of users is constant, equal to 1000. To evaluate the impact of imperfect searches in this very small population, we have used the p_{hit} values reported in Figure 5, which have been computed using the GRG methodology. The average duration of active and sleeping phases are both equal to 6 hours. The average copy holding time is 2 hours. The bandwidth available at each user is 1 MB/s. We vary the content request rate θ and the file length.

First, we consider the case in which the search is perfect, that is, users are always able to find a peer having the requested content, if it exists: $p_{\text{hit}}(x, t) = 1, \forall x > 0$. Figure 6

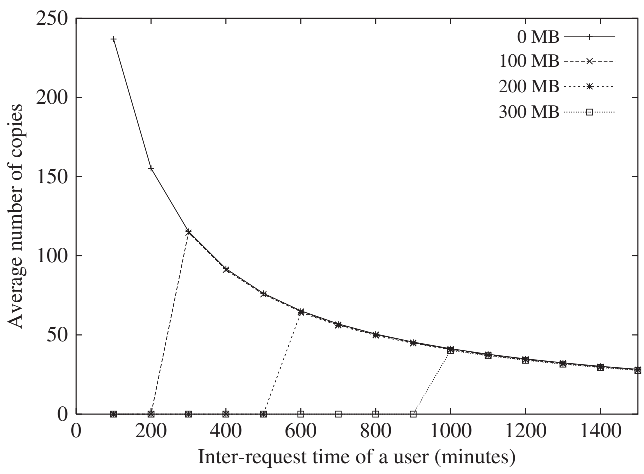

Figure 6. Average number of copies in the case of perfect search, according to the model.

reports the average number of copies in the system as a function of the inter-arrival time of requests from a single user, for different file lengths, according to the model. As expected the average number of copies decreases for increasing values of the inter-request time, as well as for increasing file lengths. However, we observe an interesting phenomenon, that is, files are not able to propagate in the network if the inter-request time is below a given threshold, which depends on the file size. This is due to a congestion collapse at the peers holding a copy of the content, which is exacerbated by the perfect search mechanism. Simulation results in Figure 7 confirm qualitatively this behaviour, although the threshold effect occurs for larger file sizes. We are still investigating the origin of this discrepancy. The 95%-level confidence intervals reported in Figure 7 suggest that accurate simulation results are difficult to obtain close to the transition point, due to the bi-stable behaviour of the system.

Interestingly, the congestion collapse is mitigated when the search mechanism is not perfect. Indeed, in this case the arrival rate of requests at peers holding a copy of the content is naturally ‘shaped’ by the hit probability, which reduces the load at the users increasing the probability that concurrent downloads end successfully. Figures 8 and 9 report the average number of copies in the system in the case of a flooding algorithm with TTL $= 2$, according to model and simulation, respectively. Confidence intervals have not been reported in Figure 9 to avoid cluttering of the plot. We observe that, in the case of imperfect search, the congestion collapse occurs only for small values of inter-request time and for much larger file sizes (of about 1000 MB), with good agreement between model and simulation.

An effective solution to the congestion collapse problem, which is actually implemented in many modern P2P applications, consists in limiting the number of concurrent uploads at a user. To demonstrate the benefit of this system design choice, we fix the inter-request time to 100 minutes and consider the average number of copies in the system as a function of the file length. We still assume a flood-base search with TTL $= 2$ to further mitigate congestion. Figure 10 compares the model and simulation results for two different values of the maximum number of concurrent uploads $U_{\max}$, equal to either 3 or 30. In the case of $U_{\max} = 30$, we observe the threshold effect occurring for file lengths of about 1400 MB (model) or 2000 MB (simulation). For all considered values of file size, the congestion collapse is eliminated by reducing the maximum number of concurrent uploads to 3.

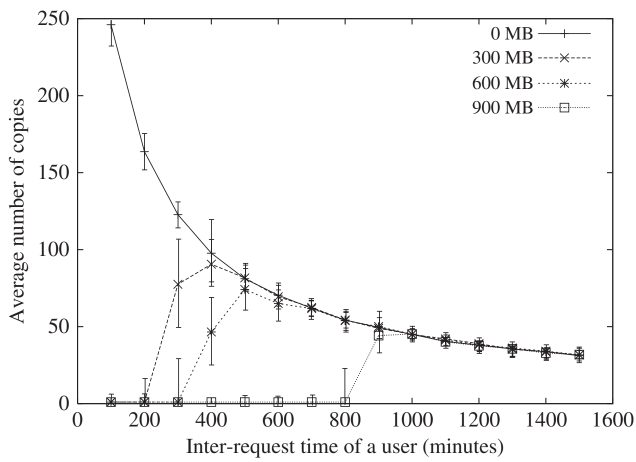

Figure 7. Average number of copies in the case of perfect search, according to simulation.

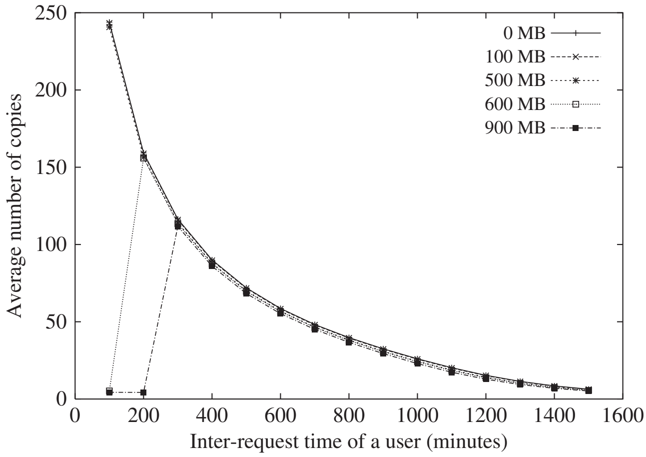

Figure 8. Average number of copies in the case of imperfect search (flooding), according to the model.

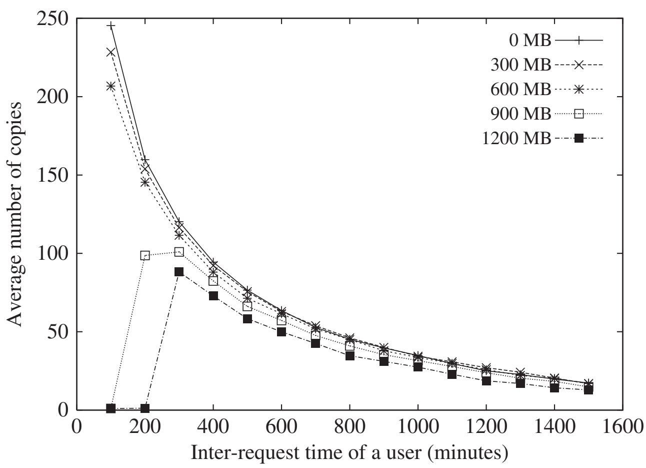

Figure 9. Average number of copies in the case of imperfect search (flooding), according to simulation.

6.4 TTL effects

In this scenario, we show the impact of the content search algorithm on the diffusion of contents in the network. The purpose of this investigation is to highlight the existing trade-off between the effectiveness of the content search algorithm in terms of $p_{\text{hit}}(x, t)$, which increases when TTL increases, and the overhead of the search mechanism in terms of bandwidth required to forward the queries, which increases exponentially with TTL. To this purpose, we evaluated the amount of bandwidth consumed at a user by the queries, assuming that the size of a query is equal to 2 KB, and subtracted this value from the bandwidth available to content download.

We consider a scenario comprising a finite population of 500,000 users and 10^6 contents, and we analyse the evolution over time of the total number of copies available in the network. To obtain the GRG topology required to compute the p_{hit} values as described in Section 4 we consider $\lambda_{\text{ping}}(t) = 0$ and D_i as a uniform distribution between 1 and 30.

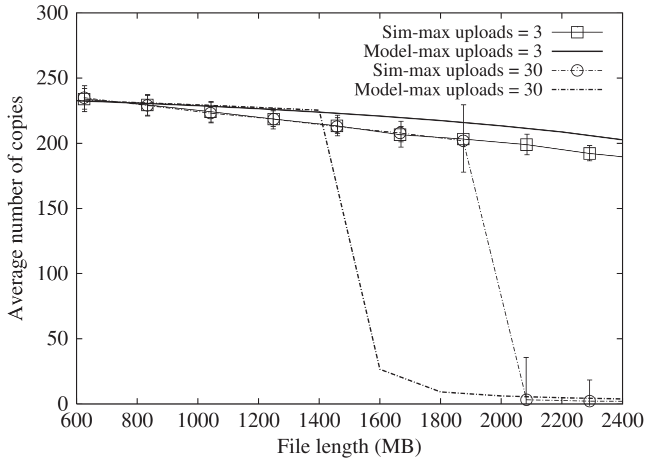

Figure 10. Average number of copies in the case of imperfect search (flooding), as a function of file length, limiting the number of concurrent uploads.

Table 5. Download and search performance in steady-state for different TTL value.

TTL	Download delay (min)	Prob. download OK	p_{hit}
2	16.0	0.789	0.80
3	16.9	0.780	0.99
4	28.5	0.678	0.99

The capacity at each user is limited to 100 KB/s, while the file size is set to 6 MB (i.e. a typical value for an MP3 file). Table 5 reports the download delay, the probability that the download is successful and the hit probability, whereas Figure 11 reports the steady-state distribution of the number of available copies in the network. We observe that, increasing

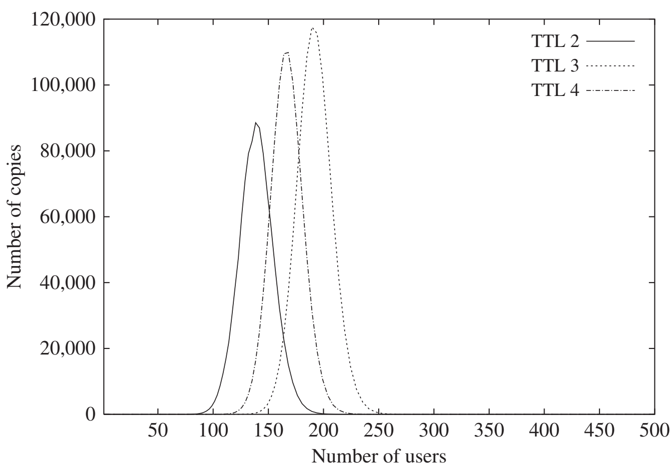

Figure 11. Steady-state distribution of copies for different TTL values.

TTL from 2 to 3 has beneficial effects on the hit probability, hence on the growth of the number of copies present in the network. However, a further increase of TTL from 3 to 4 entails a reduction in the contents' diffusion. In this case, the beneficial effects on $p_{\text{hit}}(x, t)$ are marginal, because TTL = 3 already provides good chances to find the contents, while the traffic overhead due to queries increases considerably, leading to a significant degradation of download performance. Unfortunately, due to the size of the considered system, we were unable to validate our model predictions with simulation results.

6.5 *Effects of preferential attachment and peers behaviour on topology degree*

In this section, we investigate the topology degree distribution, obtained by (11), comparing topologies created by preferential attachment (PA) schemes with the ones without preferential attachment, denoted as flat (FA) attachment schemes.

For simplicity, we consider the rate $\lambda(x, t)$ at which new peers attach to peers with x contents, $x \geq 1$, to be $x\lambda$ for PA schemes and constant λ for FA schemes. Let $\mu = \mu_{\text{as}} + \mu_{\text{u}}$ the rate at which peers interrupt their activity period or leave the system. Hence, (9) becomes:

$$m_{\text{e}} = \frac{2}{1 + h_{\text{as}}^2(t)} \times \begin{cases} (x\lambda - x\mu), & \text{for PA schemes} \\ (\lambda - x\mu), & \text{for FA schemes} \end{cases}$$

A similar formula can be found for (9). If m is the maximum number of neighbours allowed in the P2P network, FA schemes correspond to the $M/M/m/m$ queueing model, being μ the service rate and λ the arrival rate, to describe the degree distribution of the peers. We assume the distribution $f_{\text{a}}(\tau)$ of the duration of the active period for a peer ($\tau > 0$) to be either exponential (Exp) or Pareto (Par) distributed, both with mean $1/\mu$:

$$f_{\text{a}}(\tau) = \begin{cases} \mu e^{-\mu\tau}, & \text{for exponential activity} \\ \frac{\mu\alpha(\alpha-1)^{\alpha}}{(\mu\tau+\alpha-1)^{\alpha+1}}, & \text{for Pareto activity} \end{cases}$$

being $\alpha > 2$ the shape parameter of the Pareto distribution. Thanks to (10):

$$f_{l}(\tau) = \begin{cases} \mu e^{-\mu\tau}, & \text{for exponential activity} \\ \frac{\mu(\alpha-1)^{\alpha}}{(\mu\tau+\alpha-1)^{\alpha}}, & \text{for Pareto activity} \end{cases}$$

By solving (11), we show the steady-state degree distribution $E_{\text{s}}(x)$ in Figure 12, under the following choice of parameters, $\mu = 0.5$ peers/hour, whereas the values of λ, shown in Table 6, have been chosen to obtain the same final average degree equal to 12 neighbours for all scenarios.

FA-Exp combination provides a degree distribution with exponentially decreasing tail, whereas PA-Exp shows a distribution exhibiting a polynomially decaying tail for moderate values of the degree, due to the PA scheme: this is coherent with the power-law distributions found in random graphs built adopting preferential attachment schemes (see [28]). Observe, however, that this effect does not trivially descend from a classical PA model, because users stay active in our system only for limited periods, and this mitigates in part the effect of PA.

More interesting, the power-law effect can be obtained when no PA scheme is adopted but the activity duration of each peer is Pareto distributed; this is especially true for α approaching to 2, corresponding to a tail of the distribution that is more and more heavy,

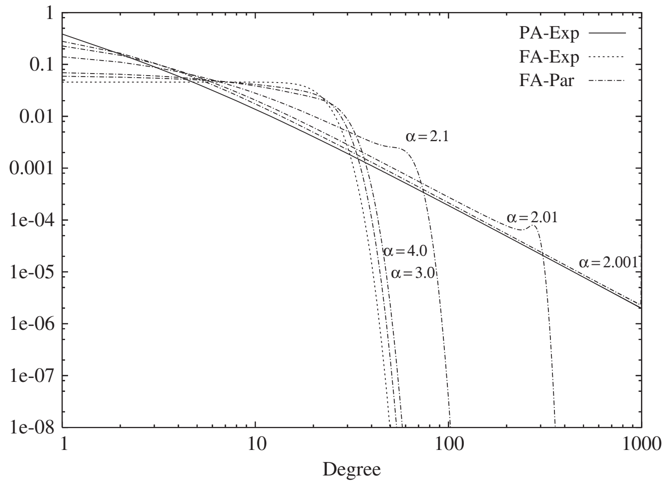

Figure 12. Steady-state degree distribution combining preferential attachment (PA) or flat (FA) schemes with exponential (Exp) and Pareto (Par, with shape parameter α) distributed activity period.

Table 6. Values of λ to guarantee average neighbour degree equal to 12 for the distributions of Figure 12.

Scenario	λ (peers/hour)
PA/Exp	1.0
FA/Exp	10.9
FA/Par ($\alpha = 4$)	12.2
FA/Par ($\alpha = 3$)	13.5
FA/Par ($\alpha = 2.1$)	30.6
FA/Par ($\alpha = 2.01$)	143
FA/Par ($\alpha = 2.001$)	1100

even if the first two moments of the activity period remain finite. In conclusion, P2P topologies can show heavy tail distributions in the degree (provided that the maximum degree allowed by the protocol is large enough) due not only to preferential attachment schemes present in the protocol but also to the activity behaviour of the peers. We believe these effects can have an important impact on the system performance.

6.6 Effects of topology degree

In this section, we show the effect of imperfect search in peer topologies created by different topology dynamics. We consider the topologies obtained in the previous section and evaluate the effect of heavy tail distributions on the degree of the topologies in the P2P system.

From the steady-state degree distribution $E_s(x)$, we were able to compute the probability $p_{hit}(x)$ of successful search for a content present in x copies, in the case of flooding search and for different TTL values, following the approach of Section 4. The simulation settings were as follows: 10 copies of a content initially available, 20,000 users, average content-holding time equal to 8 minutes, each user requests the content every 4 days. These parameters were chosen to model a free-riding scenario, in which users share the content for few minutes after the download.

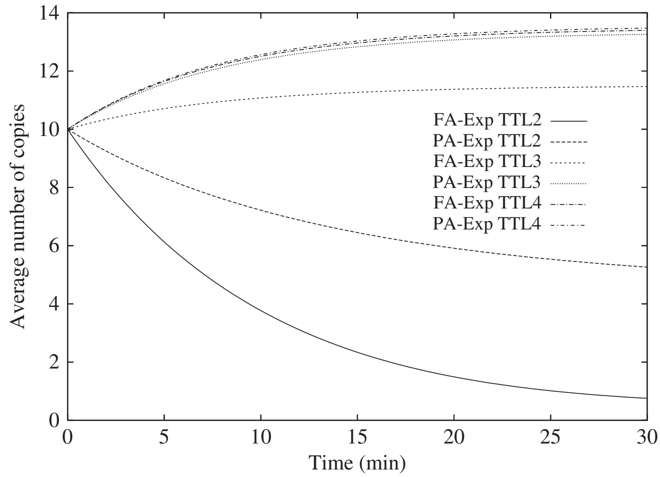

Figure 13. Instantaneous average number of copies for preferential attachment (PA) and flat (FA) schemes, for different values of TTL in the flooding mechanism.

Running the complete model, we obtained the instantaneous number of copies of a content in Figure 13. We plotted only the curves corresponding to PA-Exp and FA-Exp because they represent the extreme cases in Figure 12. The number of copies is affected by the search probability, and as seen in Section 6.3, higher TTL allows to increase the content diffusion. At the same time, topologies with heavy tail degree distributions are more efficient from the search point of view, as already discussed in Section 6.5, favouring the diffusion of a larger number of copies and increasing the content survivability.

7. Conclusions

In this article, we have modelled a large, unstructured P2P file-sharing system exploiting the basic concepts of statistical physics. We have described high-level system dynamics through a set of second-order fluid-diffusive equations and represented the overlay topology using Generalized Random Graphs. Our methodology allows to study within a single framework both transient and stationary behaviours of the P2P system, incorporating with a fairly good degree of accuracy many important dynamical effects related to resource distribution among peers, peer behaviour, content search mechanisms, as well as the dynamic nature of the overlay topology. Moreover, as the complexity of our model is largely independent of the system size (i.e. number of users and contents), it represents a scalable alternative to Monte-Carlo simulations for the performance analysis of very large systems.

Acknowledgement

This work was supported by project Amalfi ORTO119W8J funded by Università degli Studi di Torino and Compagnia di San Paolo.

Note

1. A subscript *a* or *s* is used to distinguish among active and sleeping users attributes.

References

[1] Internet Study 2008/2009. Available at www.ipoque.com/resources/internet-studies/internet-study-2008_2009 (Accessed February 2010).

[2] K. Cho, K. Fukuda, H. Esaki, and A. Kato, *The impact and implications of the growth in residential user-to-user traffic*, ACM SIGCOMM 2006, Pisa, Italy, September 2006.

[3] Available at www.skype.com/ (Accessed February 2010).

[4] J. Li, P.A. Chou, and C. Zhang, *MutualCast: An efficient mechanism for one-to-many content distribution*, ACM SIGCOMM ASIA Workshop, Beijing, China, 2005.

[5] B. Cohen, *BitTorrent protocol specification*, First Workshop on Economics of Peer-to-Peer Systems (P2P '03), Berkeley, CA, 2003.

[6] PPStream. Available at: http://www.PPStream.com (Accessed February 2010).

[7] PPLive. Available at: http://www.pptv.com (Accessed February 2010).

[8] SOPCast. Available at: http://www.sopcast.com (Accessed February 2010).

[9] L. Kleinrock, *Queueing Systems, Vol. 2: Computer Applications*, Wiley, New York, 1976.

[10] H. Kobayashi, *Application of the diffusion approximation to queueing networks I: Equilibrium queue distributions*, J. ACM 21 (2) (1974), pp. 316–328.

[11] H. Kobayashi, *Application of the diffusion approximation to queueing networks II: Nonequilibrium distributions and applications to computer modeling*, J. ACM 21 (3) (1974), pp. 459–469.

[12] D.P. Gaver, *Diffusion approximations and models for certain congestion problems*, J. Appl. Prob. 5 (1968), pp. 607–523.

[13] D. Chen, Y. Hong, and K.S. Trivedi, *Second-order stochastic fluid models with fluid-dependent flow rates*, Perform. Eval. 49 (1–4) (2002), pp. 341–358.

[14] D. Mitra, *Stochastic theory of a fluid model of producers and consumers coupled by a buffer*, Adv. Appl. Probab. 20 (3) (1988), pp. 646–676.

[15] N. Bisnik and A. Abouzeid, *Modeling and analysis of random walk search algorithms in P2P networks*, Second International Workshop on Hot Topics in Peer-to-Peer Systems, San Diego, CA, July 2005.

[16] R. Gaeta, G. Balbo, S. Bruell, M. Gribaudo, and M. Sereno, *A simple analytical framework to analyze search strategies in large-scale peer-to-peer networks*, Perform. Eval. 62 (1–4) (2005), pp. 1–16.

[17] S.N. Ethier and T. Nagylaki, *Diffusion approximations of Markov chains with two time scales and applications to population genetics*, Adv. Appl. Probab. 12 (1) (1980), pp. 14–49.

[18] R. Gaeta and M. Sereno, *Generalized probabilistic flooding in unstructured peer-to-peer networks*, IEEE Trans. Parallel Distrib. Syst. 22 (12) (2011), pp. 2055–2062.

[19] R. Gaeta and M. Sereno, *Random graphs as models of hierarchical peer-to-peer networks*, Perform. Eval. 64 (9–12) (2007), pp. 838–855.

[20] X. Yang and G. de Veciana, *Service capacity of peer to peer networks*, IEEE INFOCOM 2004, Hong Kong, March 2004.

[21] D. Qiu and R. Srikant, *Modeling and performance analysis of bit torrent-like peer-to-peer networks*, ACM SIGCOMM 2004, Portland, OR, September 2004.

[22] L. Massoulie and M. Vojnovic, *Coupon replication systems*, ACM SIGMETRICS 2005, Banff, AB, June 2005.

[23] R. Gaeta, M. Gribaudo, D. Manini, and M. Sereno, *Analysis of resource transfers in peer-to-peer file sharing applications using fluid models*, Perform. Eval. 63 (3) (2006), pp. 149–174.

[24] W. Whitt, *A diffusion approximation for the $G/GI/n/m$ queue*, Oper Res. 52 (6) (2004), pp. 922–941.

[25] H.J. Kushner and P. Dupuis, *Numerical Methods for Stochastic Control Problems in Continuous Time*, Springer, Berlin, 2001.

[26] D. Stutzbach and R. Rejaie, *Characterizing churn in peer-to-peer networks*, IMC '06: ACM SIGCOMM Conference on Internet Measurement, Rio de Janeiro, Brazil, 2006.

[27] A P2P Application Layer Simulator. Available at http://www.telematica.polito.it/~garetto/piera/ (Accessed February 2010).

[28] R. Pastor-Satorras and A. Vespignani, *Evolution and Structure of the Internet: A Statistical Physics Approach*, Cambridge University Press, Cambridge, 2004.

Appendix A. Numerical evaluation of fluid-diffusive equations

The numerical technique used to solve the fluid-diffusive equations of our model requires careful selection of the discretization step so as to assure stability and convergence of the numerical solution.

We will discuss now the numerical solution of the basic Fokker–Plank Equation (1), which is the core of the equations governing the users, the contents and the topology dynamics. The algorithm is based on the finite difference method and has been carefully designed to deal with the non-linear nature of the equations.

Consider a generic density function $f(x,t)$ satisfying, similar to (1):

$$\frac{\partial f(x,t)}{\partial t} = -\frac{\partial m(x,t) f(x,t)}{\partial x} + \frac{\partial^2 \upsilon(x,t) f(x,t)}{\partial x^2} \tag{A1}$$

which is a non-linear, second-order partial differential equation, for which $\upsilon(x,t) > 0$. Now we can use a first-order, consistent approximation and obtain from (A1), for any $\tau > 0$ small enough:

$$\begin{aligned} f(x,t+\tau) &= f(x,t) + \frac{\partial f(x,t)}{\partial t}\tau = \\ & f(x,t) + [-\frac{\partial m(x) f(x,t)}{\partial x} + \frac{\partial^2 v(x) f(x,t)}{\partial x^2}]\tau \end{aligned} \tag{A2}$$

If we now sample $f(x,t)$ at time t_k with time-step τ and at space x_n with space-step δ, we can define the following vector[1]: $\mathbf{f}^k = [f(x_1,t_k) f(x_2,t_k) \ldots f(x_N,t_k)]'$ and rewrite (A2) as follows:

$$\mathbf{f}^{k+1} = (\mathbf{I} + \mathbf{D}^k \tau)\mathbf{f}^k \tag{A3}$$

being $\mathbf{I}$ the identity matrix and $\mathbf{D}^k$ an appropriate $N \times N$ matrix defining the differential operator of the numerical scheme to be used during the k-th time-step.

The main challenge is to design $\mathbf{D}^k$, possibly different for each time-step, such that the following conditions for stability should be satisfied, for any $n = 1, \ldots, N$: mass conservation, i.e. $\sum_{m=1}^{N} \mathbf{D}^k_{nm} = 0$; positive mass, i.e. $\mathbf{I} + \mathbf{D}^k \tau \geq 0$. Note that these conditions imply that $\mathbf{D}^k_{nn} \leq 0$ and $\mathbf{D}^k_{mn} \geq 0$ for any $m \neq n$. Let $m^k_n = m(x_n, t_k)$ and $v^k_n = v(x_n, t_k)$. The above conditions imply that $\| \mathbf{I} + \mathbf{D}^k \tau \|_1 = 1$ and its spectral radius is bounded by 1; hence, the numerical scheme is stable for small perturbations of the initial conditions.

Depending on the sign of m^k_n, we combined the forward and backward finite difference approximation for the first derivative, and the central approximation for the second derivative to guarantee mass conservation. Finally, $\mathbf{D}^k$ became a consistent differential operator, described by a tridiagonal matrix, not symmetric, whose central elements of n-th row are the following:

- if $m^k_n < 0$:

$$\ldots, \frac{2v^k_{n-1}}{2\delta^2}, \frac{m^k_n}{\delta} - \frac{2v^k_n}{\delta^2}, -\frac{m^k_{n+1}}{\delta} + \frac{2v^k_{n+1}}{2\delta^2}, \ldots$$

- if $m^k_n > 0$:

$$\ldots, \frac{m^k_{n-1}}{\delta} + \frac{2v^k_{n-1}}{2\delta^2}, -\frac{m^k_n}{\delta} - \frac{2v^k_n}{\delta^2}, \frac{2v^k_{n+1}}{2\delta^2}, \ldots$$

To guarantee positive mass, the central diagonals of $\mathbf{D}^k$ should satisfy, for all n:

$$1 + \mathbf{D}^k_{nn}\tau \geq 0 \Rightarrow 1 - \tau \quad \frac{|m^k_n|}{\delta} + \frac{2v^k_n}{\delta^2} \quad \geq 0$$

from which the time-step should be chosen such that

$$\tau \leq \min_{n=1,\dots,N} \left\{ \frac{\delta^2}{|m_n^k|\delta + 2v_n^k} \right\}$$

Note that this time-step is, in general, time-dependent (even if its variations were quite small in the considered scenarios) and requires to implement a variable step method in (A3).

Furthermore, to prevent other mass losses, particular care should be devoted to manage the barrier in $x_1 = 0$ and the differential operator when $m_n^k > 0$ and $m_{n+1}^k < 0$. For brevity, we do not report the details.

Note

1. All vectors are column vectors and denoted by non-capital letters; all matrices are denoted by capital letters.