

A Fluid-Diffusive Approach for Modelling P2P Systems

G.Carofiglio¹, R.Gaeta², M.Garetto¹, P.Giaccone¹, E.Leonardi¹, M.Sereno²

1-Dipartimento di Elettronica - Politecnico di Torino, Italy.

2-Dipartimento di Informatica - Università di Torino, Italy.

Abstract— This paper presents an application of basic concepts of statistical physics to devise an approximate model describing the dynamics of large peer-to-peer networks, based on fluid-diffusive equations. The model we propose is quite general and highly modular, and allows to represent several effects related to resources distribution among peers, user behavior, resource localization algorithms and dynamic structure of the overlay topology. Since the complexity of the model is largely independent of the system size, it provides a viable alternative to Montecarlo approaches for the analysis of very large P2P systems.

I. INTRODUCTION AND PREVIOUS WORK

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 the whole Internet traffic [3].

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 [1], a P2P-based telephony application, attracts millions of users every day. Furthermore, the adoption of P2P paradigms has also been recently proposed for content delivery infrastructures (see for instance [7], [4]) to better face Internet 'flash crowds', i.e., unexpected rapid increases in the demand for particular resources.

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

What to model?

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 we describe the evolution of a P2P system trying to abstract from the particular implementation? 2) what are the fundamental performance indices 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 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 very fast rate and with relatively short expectation of life; hence, a model which focuses on one particular implementation may fail to provide answers whose validity is somehow general. For these reasons we have preferred to build a high-level model of a P2P system which abstracts from the specific implementation. We expect, in this way, to obtain less accurate but more general insights on the significant dynamics of P2P systems which could guide the design of future P2P systems.

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

Which modeling technique to use ?

Traditional Markovian modeling techniques relying on a microscopic descriptive approach, in which architectural elements are represented in great level of detail, do not permit to consider large systems like P2P. Recently, fluid models have been proposed as a viable approach to model P2P dynamics in both transient and steady state regime. Fluid models adopt an abstract, deterministic description of the "average" network dynamics through a set of ordinary differential equations, thus neglecting the stochastic short-term fluctuations in the system. The resulting set of differential equations is then solved either analytically (when possible) or numerically, obtaining estimates of the time-dependent system behavior. By so doing, however, many important phenomena related to the intrinsic stochasticity of the system dynamics are lost.

In this paper we propose a second-order "fluid-diffusive" approximation of system dynamics based on a set of partial differential equations which allows us to capture the impact of stochasticity on the system dynamics.

A. Related Work and Paper Contribution

In this section we briefly summarize previous work on analytical modeling of P2P systems that are related with our

proposal.

Paper [5] presents a simple approach for the performance evaluation of search strategies in large-scale decentralized unstructured P2P applications. The authors employ 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 [9], [10], [8], which explore the performance of BitTorrent-like P2P systems, are closer to our approach. The work in [9] considers users' and contents' diffusion dynamics which are modeled by age-dependent branching processes so as to analyze the transient dynamics of the system in terms of service capacity. This approach nicely captures the ability of this type of P2P systems to face flash crowds. Furthermore, the paper presents a separate Markovian model for the steady state regime. Paper [10] shares several assumptions with [9]. The differences between the two mostly reside in the modeling techniques and the selected performance metrics. In particular, [10] 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 on 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, upload bandwidth. In [8] 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.

In our work we adopt an approach similar to [9] and [10]. However our model is not tailored to a particular P2P system; instead, it describes the evolution of a generic unstructured P2P system trying to abstract from the particular P2P applications. In this manner the paper tries to provide answers whose validity are somehow general. Furthermore, we develop a methodology to study within a single framework both transient and stationary behavior 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 [9] and [10], our methodology allows to investigate the joint effect of several phenomena related to peer behavior, content localization algorithm, and the dynamic structure of the overlay topology.

Another remarkable difference concerns the modeling 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 modeling 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 of further investigations.

II. MODELING UNSTRUCTURED P2P SYSTEMS: AN OVERVIEW

Unstructured P2P systems exhibit very high dynamicity resulting from complex interactions among users, contents and the underlying overlay topology. Users and contents dynamics are closely tied, since users continuously retrieve new contents based on their availability, and remove contents already in their possess. Moreover, users continuously join and leave the system, modifying the structure of the overlay topology. While 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, since users can retrieve only those contents which 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 dynamics 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 allow 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 $p_{hit}(n)$ to localize a content in the network which is stored by n users. Finally, the impact of the content search algorithm is introduced back into the main equations of users and contents dynamics using the above probability $p_{hit}(n)$. The three models jointly evolve over time, allowing to study the transient behavior of a P2P system in an integrated way.

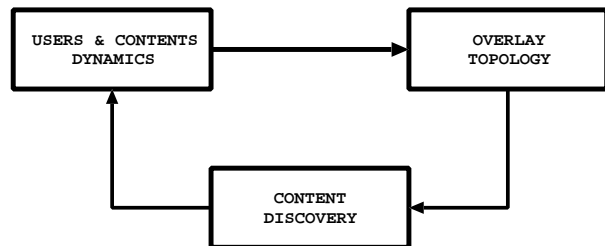


Fig. 1. A schematic representation of our model

III. THE MAIN FLUID-DIFFUSIVE MODEL

In this section we first introduce our basic model describing users' and contents' 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); (ii) instantaneous download (i.e., the bandwidth available in the physical network is infinite). Our model captures the macroscopic behavior of users and

Symbol	Meaning
$\lambda_u(t)$	rate at which new users subscribe the system
$1/\mu_u(t)$	average system subscription period
$1/\mu_{as}(t)$	average duration of an active period
$1/\mu_{sa}(t)$	average duration of a sleeping period
$h_{sa}(t)$	variation coefficient of sleeping period
$h_{as}(t)$	variation coefficient of active period
$\theta(t)$	request rate for a given available content
$h_r(t)$	variation coefficient of inter-request time
$1/\mu_h(t)$	average content holding time
$h_h(t)$	variation coefficient of holding time
$\lambda_c(t)$	rate at which new contents are made available by an active user

TABLE I
PHYSICAL PARAMETERS OF USER BEHAVIOR

contents in an ideal system; in Section V 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 III-A, and the point of view of contents, described in Section III-B.

A. User dynamics

Users dynamically *subscribe* and *un-subscribe* 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 they are storing. Sleeping users, instead, are not connected and they 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 initial 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, he/she makes new contents available to the community at rate $\lambda_c(t)$. The parameters used to describe the user behavior are summarized in Table I. We remark that we allow all of them to vary over time according to given functions of time. This is indeed one of the main strengths of our model, i.e., the ability to study the transient behavior of non-stationary P2P systems. In [17] it has been reported a measurement study of user dynamics 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 behavior (i.e., the same set of parameters as reported in Table I). Diversity in user behavior 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 density of users storing x contents, at time t . We wish to model how this distribution evolves over time. Each user is modeled as a moving particle whose instantaneous position $x(t)$ represents the number of contents he/she is currently holding. According to the diffusion approximation, the movement of each user-particle can be described by an independent, inhomogeneous Brownian motion with parameters $m(x, t)$ and $\sigma^2(x, t)$, which represents the instantaneous average speed and speed variance for the user-particle in position x at time t . When considering a large population of user-particles, each one moving according to an independent Brownian motion (as commonly assumed in statistical-physics), we can statistically describe the dynamics of the entire population using a Fokker-Planck equation:

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

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:¹:

$$\begin{aligned} \frac{\partial G_a(x, t)}{\partial t} = & -\frac{\partial m_a(x, t)G_a(x, t)}{\partial x} \\ & + \frac{1}{2} \frac{\partial^2 \sigma_a^2(x, t)G_a(x, t)}{\partial x^2} - [\mu_{as}(t) + \mu_u(t)]G_a(x, t) \\ & + \mu_{sa}(t)G_s(x, t) + \lambda_u(t)\delta(x) \end{aligned} \quad (2)$$

$$\begin{aligned} \frac{\partial G_s(x, t)}{\partial t} = & -\frac{\partial m_s(x, t)G_s(x, t)}{\partial x} \\ & + \frac{1}{2} \frac{\partial^2 \sigma_s^2(x, t)G_s(x, t)}{\partial x^2} - [\mu_{sa}(t) + \mu_u(t)]G_s(x, t) \\ & + \mu_{as}(t)G_a(x, t) \end{aligned} \quad (3)$$

defined for $x \in [0, C_T(t)]$, where $C_T(t)$ is the number of contents available in the system at time t . For simplicity we assume that newcomers join the system in the active phase without storing any content. This explains the term $\lambda_u(t)\delta(x)$ 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. 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

$$r_a(x, t) = \theta(t)[C_T(t) - x] + \lambda_c(t) \quad (4)$$

Notice that we assume that the search rate of a user is proportional to the number of available contents that he/she does not currently hold. Sleeping users do not acquire new

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

contents, thus $r_s(x, t) = 0$. The removal rate is the same for both active and sleeping user, and it is proportional to the number of contents currently hold: $d_a(x, t) = d_s(x, t) = \mu_h(t)x$. Finally, the impact of variation coefficients is taken into account using the approach proposed in [11], obtaining

$$\begin{aligned} m_a(x, t) &= \frac{2}{1 + h_h^2(t)} [r_a(x, t) - d_a(x, t)] \\ m_s(x, t) &= \frac{2}{1 + h_h^2(t)} [-d_s(x, t)] \\ \sigma_a^2(x, t) &= \frac{2}{1 + h_h^2(t)} \left[\theta(t) [C_T(t) - x] \frac{2h_r^2(t) + h_h^2(t) - 1}{h_h^2(t) + 1} \right. \\ &\quad \left. + \lambda_c(t) + d_a(x, t) \right] \\ \sigma_s^2(x, t) &= \frac{2}{1 + h_h^2(t)} [d_s(x, t)] \end{aligned}$$

Note that all quantities depend on the second moment of the copies holding time, through the variation coefficient $h_h(t)$. The variation coefficient of inter-request time $h_r(t)$, instead, affects only the speed variance of active users.

The number of active users at time t is $U_a(t) = \int G_a(x, t) dx$, whereas the number of sleeping users at time t is $U_s(t) = \int G_s(x, t) dx$. The total number of users is $U(t) = U_a(t) + U_s(t)$. We also introduce the total number of copies stored by active users $O_a(t) = \int x G_a(x, t) dx$ and the total number of copies stored by sleeping users $O_s(t) = \int x G_s(x, t) dx$.

Remarks.

To conclude the section, we would like to spend a few words on the fluid-diffusive approximation. 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. When all of the random variable describing the user behavior 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). 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 both in terms of accuracy and computational efficiency in several application contexts [11], [12], [13], [14], [15], [16]. For these reasons in this paper we have proposed a diffusion approximation to model the dynamics of contents and users in large P2P networks.

B. 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. Similarly to users' dynamics, we model contents' dynamics through a second order diffusion approximation: each content in the system is modeled 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:

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

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: (i) the rate at which new copies of the considered content are made available by active users retrieving it, and (ii) 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 (ii). Indeed, we have:

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

The effective removal rate $d(x, t)$ of a content is the sum of the rate at which copies of the considered content are canceled 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

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

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:

$$\nu_0(t) = \frac{1}{2} \frac{\partial \sigma^2(x, t)F(x, t)}{\partial x} \Big|_{x=0} \quad (7)$$

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

$$C_T(t) = \int_0^{U(t)} F(x, t) dx$$

IV. 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, i.e. 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, for eventually reappearing later when it becomes active again.

During the activity period, users maintain a list of neighboring nodes which is dynamically updated, removing address of nodes which are no longer active and acquiring addresses of new nodes. Several mechanisms (for example the ping-pong mechanism in Gnutella) have been defined to dynamically acquire the information on new neighbors during the activity period of a user. To simplify our model we assume that users that become inactive lose memory of their neighbor list. Let $E(x, t)$ be the number of active users whose nodal degree (i.e., their neighbor list length) at time t is x . We can write:

$$\begin{aligned} \frac{\partial E(x, t)}{\partial t} = & -\frac{\partial m_e(x, t)E(x, t)}{\partial x} + \frac{1}{2} \frac{\partial^2 \sigma_e^2(x, t)E(x, t)}{\partial x^2} + \\ & - [\mu_{as}(t) + \mu_u(t)]E(x, t) + [\mu_{sa}(t)U_s(t) + \lambda_u(t)]D_i(x) \end{aligned} \quad (8)$$

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 (assumed to be known).

Now turning our attention to $m_e(x, t)$ and $\sigma_e^2(x, t)$, it results:

$$\begin{aligned} m_e(x, t) &= \frac{2}{1 + h_{as}^2(t)} [\lambda(x, t) - x\mu_{as}(t) - x\mu_u(t)] \\ \sigma_e^2(x, t) &= \frac{2}{1 + h_{as}^2(t)} [\lambda(x, t) + x\mu_{as}(t) + x\mu_u(t)] \end{aligned}$$

where $\lambda(x, t)$ the rate at which users already having x neighbors acquire new neighbors. The expression of $\lambda(x, t)$ depends on the specific mechanism for acquiring new neighbors. 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:

$$\lambda(x, t) = p_{\text{reach}}(x) [\mu_{sa}(t)U_s(t)\bar{D}_i(t) + \lambda_{\text{ping}}(t)U_a(t)]$$

where $\bar{D}_i(t)$ 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 connection is selected by a peer activating a new connection. The latter probability can be evaluated using random graphs

techniques. We observe that $p_{\text{reach}}(x)$ exhibits a dependency on x because users having more neighbors are more likely selected by other peers.

V. SEARCH AND DOWNLOAD PHASES

The basic model introduced in Section III can be extended to capture the limitations 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 developed a statistical physics model based on the generating function method for random graphs.
- the probability of a successful download, $p_{\text{down}}(t)$, which depends on several factors, like 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:

$$\begin{aligned} r(x, t) = & \theta(t)[U_a(t) - x] p_{\text{hit}}(x, t) p_{\text{down}}(t) \\ & + \mu_{sa}(t) \frac{O_s(t)}{C_T(t)} \end{aligned} \quad (9)$$

In the following Sections V-A and V-B we will present the models to capture the search and the download phases, respectively.

A. 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 employed in the system.

In this paper 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 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 walks, probabilistic flooding, or hybrid flooding-random walk explorations of the overlay topology.

The modeling methodology employed in this paper to compute $p_{\text{hit}}(x, t)$ is based on the exploitation of *Generalized Random Graphs* (GRG) to describe the overlay topology, introduced in [5]. Here we just provide some high-level considerations of how the methodology works and refer the reader to [5] 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, i.e., the probability that a randomly chosen node has a certain number of edges emanating from it. This distribution can be related to user behavior and application-specific details using the model described in Section IV.

In [5] it is shown how to obtain the generating function of the number of peers that 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 $p = x/U_a$ 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 .

B. Download phase

When modeling the content download process we consider an ideal transport network, i.e. 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) at the peers. An accurate estimate of the download process should take into account the dynamics of the available bandwidth of both peers which are taking part to the content transfer, as done in [6].

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 modeled 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 both on the the distribution of contents stored at peers and on the server selection policy. In this paper we assume that a random load-balancing policy is adopted, i.e., among all peers sharing the desired content which are 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 hold by x users, then $\theta(t)p_{hit}(t, x)[U_a(t) - x]$ represents its download request rate. Since 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 $\frac{\theta(t)p_{hit}(t, x)[U_a(t) - x]}{x}$.

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

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 with download requests are addressed toward users storing y contents is then expressed

by the following pdf:

$$\frac{yG_a(y, t)}{E[y]U_a(t)}$$

where $E[y]U_a(t) = \int yG_a(y, t)dy$.

Let $T_{down}(\lambda)$ 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

$$E[T_{down}](t) = \int_y T_{down}(y\lambda_{down}) \frac{yG_a(y, t)}{E[y]U_a(t)} dy$$

Similarly, given the average arrival rate λ of requests at the server, the probability that a download fails because the the server disconnects from the P2P system before the end of 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.

Being $p_d(\lambda)$ ² the probability that a download request directed toward a peer who is experiencing a request rate λ is successful, by averaging with respect to the request arrival rate λ , the average probability that a download in the network is successful can be obtained as:

$$p_{down}(t) = \int_y p_d(y\lambda_{down}) \frac{yG_a(y, t)}{E[y]U_a(t)} dy$$

VI. 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 [2]. The simulator accounts for the detailed behavior 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 analyze 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 $p_{hit}(n)$ of finding a content in the network available in a given number n of copies. The simulator code and additional details on it can be found at [2].

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

² $p_d(\lambda)$ can be analytically related to the distribution of $T_{down}(\lambda)$ and to the distribution of the peer active period

A. Users' 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 5 contents/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 equal 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 starts 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, though the variation coefficients h_r and h_h .

Figure 2 reports the steady-state distribution $G_a(x, t)$ of the number of active users storing a given number of contents, for various combinations of the above variation coefficients. Continuous lines represent model results while discrete points stand for simulation results. The 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

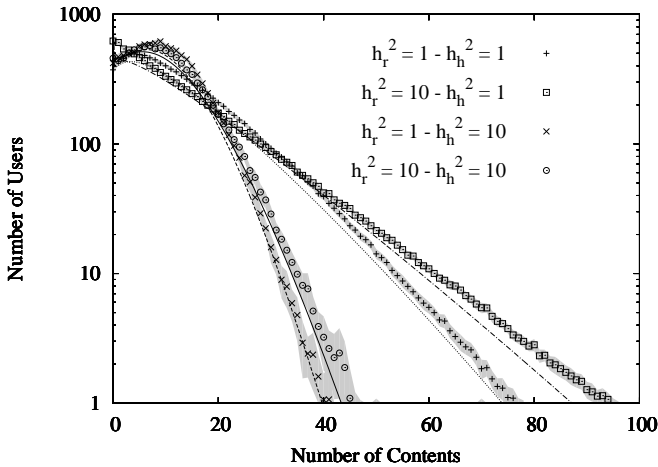


Fig. 2. Distribution of the total number of users storing a given number of contents (dashed lines for $h_r^2 = 10$)

are in good agreement; this is not surprising since model and simulator share the same values of parameters describing the user behavior. From the curves it can be observed that distribution $G_a(x, t)$ is strongly influenced by coefficients of variations. Not only the tails of the distributions are different, but also the average number of contents stored by a user can vary. In particular, it is strongly affected by the coefficient of variation of the holding time, whereas it does not depend on the variation coefficient of inter-request time, as reported in Table II and confirmed by simulation. An accurate prediction

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

TABLE II
AVERAGE NUMBER OF CONTENTS STORED BY AN ACTIVE USER FOR
VARIOUS COMBINATIONS OF VARIATION COEFFICIENTS

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 average values of content holding time and inter-request cannot fully characterize the resulting user behavior, thus more accurate models (like ours) capturing the impact of higher moments of these quantities are needed.

B. Content survivability

To validate our diffusion approximation of contents dynamics, we consider the case of a single content which 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 , i.e. its survivability function. Actually, two basic scenarios can happen: i) the content prematurely dies out after being removed by all users storing it; ii) 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 its copy before it is successfully downloaded by another user, the content already disappears from the system. The transient phase of the diffusion of a new content has been studied in [9] 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 VI-A. 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 3 obtained for $h_h^2 = 1$ and $h_h^2 = 10$. 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 $h_h^2 = 10$, 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 $h_h^2 = 1$ 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), since 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 behavior quite well.

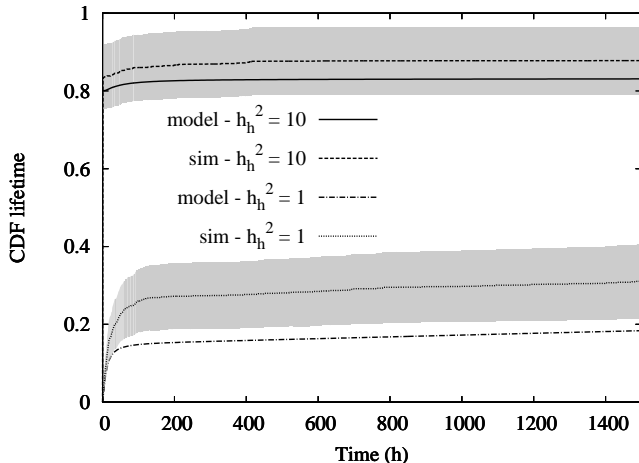


Fig. 3. Cdf of the content lifetime

C. 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 $p_{hit}(x, t)$ and $p_{down}(t)$ in (9).

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. The GRG topology we compute as described in Section IV is such that the p_{hit} value we obtain when there is only one copy of the resource is very high (≈ 0.5) even for very shallow flooding, i.e., $TTL=2$. Therefore, to evaluate the impact of imperfect searches in this very small population we consider p_{hit} values that are characterized by the graph depicted in Figure 4. The average duration on active and sleeping phases are both equal to 6 hours. The average copy holding time is 2 hours. The bandwidth available at each user for uploading the considered content 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, i.e. users are always able to find a peer having the requested content, if it exists: $p_{hit}(x, t) = 1, \forall x > 0$. Figure 5 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

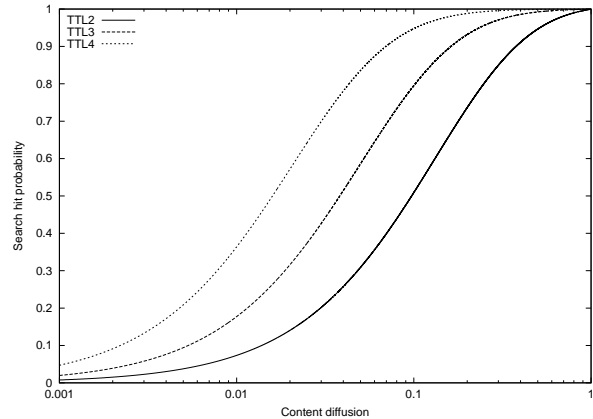


Fig. 4. Values of p_{hit} as a function of the content diffusion used to analyze the impact of imperfect searches

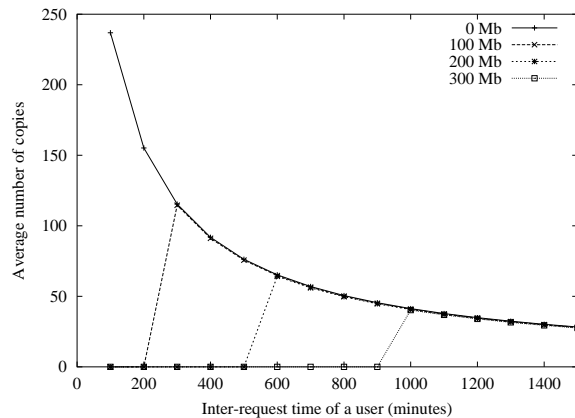


Fig. 5. Average number of copies in case of perfect search, according to the model

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 6 confirm qualitatively this behavior, 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 6 suggests that accurate simulation results are difficult to obtain close to the transition point, due to the bi-stable behavior 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 7 and 8 report the average number of copies in the system in case of a flooding algorithm with $TTL=2$, according to model and simulation, respectively. Confidence intervals have not been reported in Figure 8 to avoid cluttering 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,

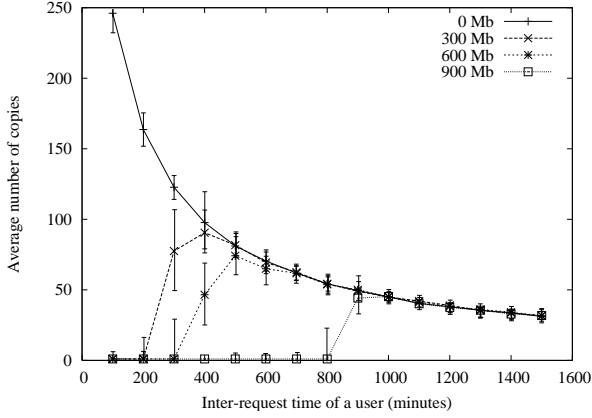


Fig. 6. Average number of copies in case of perfect search, according to simulation

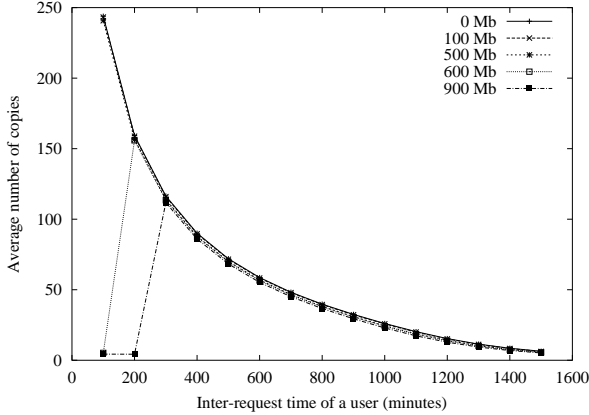


Fig. 7. Average number of copies in case of imperfect search (flooding), according to the model

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 9 compares model and simulations results for two different values of the maximum number of concurrent uploads U_{max} , equal to either 3 or 30. In 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.

D. 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 tradeoff between the effectiveness of the content search algorithm in terms of $p_{hit}(x, t)$, 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 have evaluated the amount of bandwidth consumed at a user by the queries, assuming that

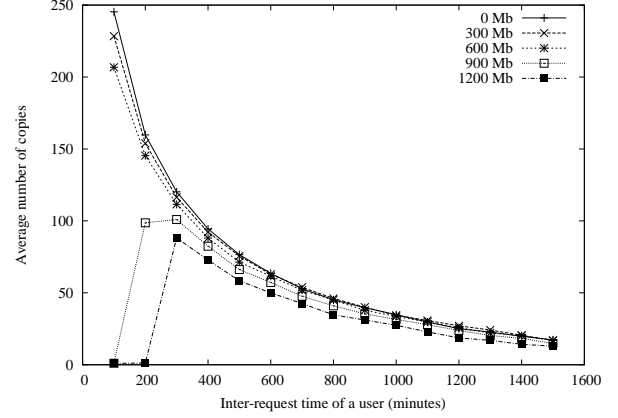


Fig. 8. Average number of copies in case of imperfect search (flooding), according to simulation

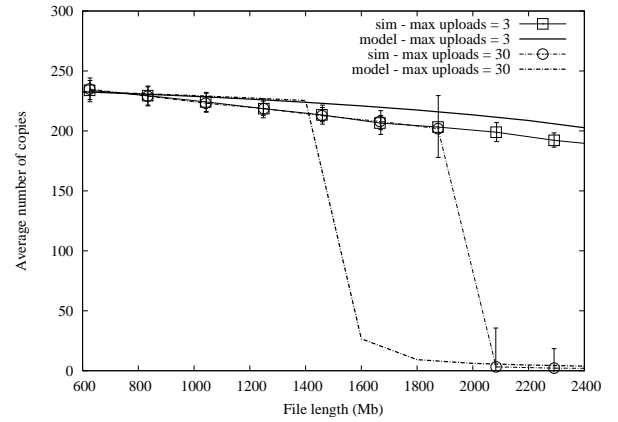


Fig. 9. Average number of copies in case of imperfect search (flooding), as a function of file length, limiting the number of concurrent uploads

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 analyze 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 IV we consider $\lambda_{ping}(t) = 0$ and D_i as a uniform distribution between 1 and 30. The capacity at each user is limited to 100 kbit/s, while the file size is set to 6MB, a typical value for an MP3 file. Table III reports the download delay, the probability that the download is successful and the hit probability, whereas Figure 10 reports the steady state distribution of the number of available copies in the network. We observe that, increasing 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 $p_{hit}(x, t)$ are marginal, since $TTL = 3$ 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.

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

TABLE III
DOWNLOAD AND SEARCH PERFORMANCE IN STEADY-STATE FOR
DIFFERENT TTL VALUE

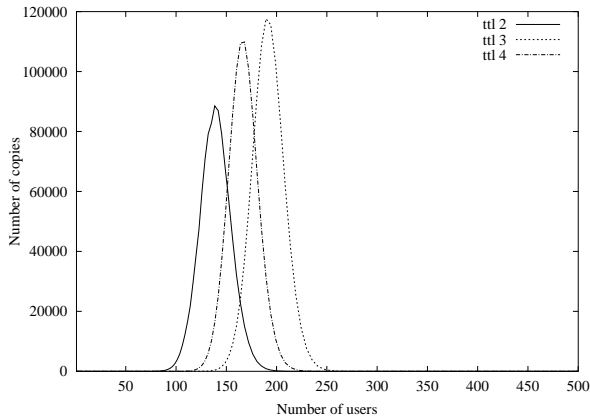


Fig. 10. Steady-state distribution of copies for different TTL values

VII. CONCLUSIONS

In this paper we have modeled large P2P systems exploiting 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 behavior of P2P systems, incorporating with a fairly good degree of accuracy many important dynamical effects related to resources distribution among peers, peer behavior, content search mechanisms, as well as the dynamic nature of the overlay topology. Moreover, since 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 Montecarlo simulations for the performance analysis of very large systems.

REFERENCES

- [1] Global Index (GI). Technical report. http://www.skype.com/skype_p2pexplained.htm.
- [2] A P2P Application Layer Simulator. <http://www.telematica.polito.it/~garetto/piera/>.
- [3] The true picture of peer-to-peer file sharing. Technical report, Cachelogic Research, July 2004. <http://www.cachelogic.com/research/>.
- [4] B. Cohen. BitTorrent protocol specification. In *First Workshop on Economics of Peer-to-Peer Systems (P2P '03)*.
- [5] 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. *Performance Evaluation*, 62(1-4):1-16, 2005.
- [6] R. Gaeta, M. Gribaudo, D. Manini, and M. Sereno. Analysis of Resource Transfers in Peer-to-Peer File Sharing Applications using Fluid Models. *Performance Evaluation*, 62(1-4):1-16, 2006.
- [7] J. Li, P. A. Chou, and C. Zhang. Mutualcast: An Efficient Mechanism for Content Distribution in a Peer-to-Peer (P2P) Network. Tech. rep., Microsoft Research, MSR-TR-2004-100, 2004.
- [8] L. Massoulié and M. Vojnovic. Coupon Replication Systems. In *Proc. of ACM SIGMETRICS 2005*.
- [9] X. Yang and G. de Veciana. Service Capacity of Peer to Peer Networks. In *Proceedings of INFOCOM 2004*.

- [10] D. Qiu and R. Srikant. Modeling and Performance Analysis of Bit Torrent-Like Peer-to-Peer Networks. In *Proc. of ACM SIGCOMM 2004*.
- [11] W. Whitt. A Diffusion Approximation for the G/GI/n/m Queue. *Operations Research*, 52(6):922-941, 2004.
- [12] H. Kobayashi, "Application of the Diffusion Approximation to Queuing Networks I: Equilibrium Queue Distributions", *Journal of the ACM*, 21(2), pp. 316-328, 1974.
- [13] H. Kobayashi, "Application of the Diffusion Approximation to Queuing Networks II: Nonequilibrium Distributions and Applications to Computer Modeling", *Journal of the ACM*, 21(3), pp. 459-469, 1974.
- [14] D. P. Gaver, "Diffusion Approximations and Models for certain Congestion Problems", *J. Appl. Prob.*, Vol 5, pp. 607-523, 1968.
- [15] D. Chen, Y. Hong, K.S. Trivedi, "Second-order stochastic fluid models with fluid-dependent flow rates", *Performance Evaluation*, Vol. 49, n. 1-4, pp. 341-358, 2002.
- [16] D. Mitra, "Stochastic Theory of a Fluid Model of Producers and Consumers Coupled by a Buffer", *Advances in Applied Probability*, Vol. 20, n. 3, pp. 646-676, 1988.
- [17] D. Stutzbach, R. Rejaie, "Characterizing Churn in Peer-to-Peer Networks", Technical Report CIS-TR-05-03, University of Oregon, June 2005 <http://www.cs.uoregon.edu/~reza/PUB/tr05-03.pdf>