Project Number 001907

DELIS
Dynamically Evolving, Large-scale Information Systems

Integrated Project
Member of the FET Proactive Initiative Complex Systems

Deliverable D3.3.1

Critical Evaluation of existing nonlinear and stochastic models on suitably selected problems
Start date of the project: January 2004
Duration: 48 months
Project Coordinator: Prof. Dr. math. Friedhelm Meyer auf der Heide
Heinz Nixdorf Institute, University of Paderborn, Germany
Due date of deliverable: June 2004
Actual submission date: October 2004
Dissemination level: PU – public
Work Package 3.3: Information Theory of Wireless Networks
Participants: Computer Technology Institute, Patras, Greece
University of Paderborn, Germany
TU Wroclaw, Poland
Authors of deliverable: Sotiris Nikoletseas (nikole@cti.gr)
Paul Spirakis (spirakis@cti.gr)
1 Interaction in Mobile Communications

Wireless Networks are composed of a large number of hosts that move in the network area and communicate in a wireless manner. In the particular case where no fixed infrastructure is available (either because it is expensive/impossible to build one, or because such an infrastructure has been damaged) the mobile hosts must form a temporary network where the hosts may act as intermediate relays for forwarding information. Such networks are called ad-hoc mobile networks. In other wireless networks a fixed topology underlying network exists, such as in the case of cellular networks. To better highlight the impact of mobility and interaction between hosts we focus on the ad-hoc mobile case.

One of the most essential characteristics of such wireless networks is their highly dynamic nature: mainly due to movement (but also due to failures, power saving modes etc) communication links may appear and disappear as hosts move in and out of the transmission range of other hosts. Thus, the underlying communication graph of the network is highly dynamic, changing according to the way hosts interact with each other as they move in the network area.

Another crucial feature of mobile wireless networks is the importance, necessity and impact of interaction between mobile hosts. Indeed, communication in such networks is impossible without having hosts to act as relays for information propagation (since the transmission range of hosts does not usually cover the network area and several “hops” are needed to forward a message to a target user). This interaction of hosts is closely related to their mobility, since hosts exchange information when, due to their movement, they come within transmission range of each other.

We note that this necessity of interaction applies to even the most basic versions of the communication problem, such as in pairwise communication i.e. when the sender host ($MH_S$) wishes to send information to a single receiver host ($MH_R$).

Other phenomena when interaction plays a key role include a) virus spread (i.e. a host may become infected when it contacts an infected one, or a virus “moving” in the network may be destroyed when reaching a host with special anti-virus software), b) intrusion propagation (the process of spreading of an attack to the security of hosts) and c) mobile agents.

In what follows we describe three models in the recent literature: a) a Markov Chains model for pairwise communication in ad-hoc mobile networks (introduced in [4]), b) a model of stochastic particle interaction for virus spread (introduced in [9]) and c) a random graph model that focuses on proximity-sensitive interaction (introduced in [15]).

2 A Markov Chains model for mobile communication

2.1 An explicitly model of motions

The set of previous research that follows the approach of slowly-changing communication graphs, models hosts motions only implicitly, i.e. via the pre-assumed upper bound on the rate of virtual link changes. In contrast, we present an explicit model of motions because it is apparent that the motions of the hosts are the cause of the fragility of the virtual links (see also [4]).

Thus we distinguish explicitly between (a) the fixed (for any algorithm) space of possible motions of the mobile hosts and (b) the kind of motions that the hosts perform inside this space. In the sequel we have model the space of motions only combinatorially, i.e. as a graph. We however believe that future research will complement this effort by introducing geometry details into the model.

In particular, we abstract the environment where the stations move (in three-dimensional space with possible obstacles) by a motion-graph. In particular, we first assume that each mobile host has a transmission range represented by a sphere $tr$ centered by itself. This means that any other host inside $tr$ can receive any message broadcast by this host. We approximate this sphere by a cube $tc$ with volume $V(tc)$, where $V(tc) < V(tr)$. The size of $tc$ can be chosen in such a way that its
volume $V(tc)$ is the maximum that preserves $V(tc) < V(tr)$, and if a mobile host inside $tc$ broadcasts a message, this message is received by any other host in $tc$. Given that the mobile hosts are moving in the space $S$, $S$ is divided into consecutive cubes of volume $V(tc)$.

**Definition 2.1** The motion graph $G(V, E)$, $(|V| = n, |E| = m)$, which corresponds to a quantization of $S$ is constructed in the following way: a vertex $u \in G$ represents a cube of volume $V(tc)$. An edge $(u, v) \in G$ if the corresponding cubes are adjacent.

In the general case, we assume that the motions of such hosts are decided by an oblivious adversary. The adversary determines motion patterns in any possible way but independently of the distributed algorithm for communication. In other words, we exclude the case where some of the hosts not in the support are deliberately trying to maliciously affect the protocol (e.g. to avoid the hosts in the support).

For purposes of studying efficiency of distributed algorithms for ad-hoc networks on the average, we examine the case where the motions of any host not affected by the algorithm are modeled by concurrent and independent random walks. In fact, the assumption that the mobile users move randomly, either according to uniformly distributed changes in their directions and velocities or according to the random waypoint mobility model by picking random destinations, has been used by other research.

### 2.2 A basic communication protocol

A **basic communication problem**, in ad-hoc mobile networks, is to send information from some sender user, $MH_S$, to another designated receiver user, $MH_R$.

One way to solve this problem is the protocol of notifying every user that the sender $MH_S$ meets (and providing all the information to it) hoping that some of them will eventually meet the receiver $MH_R$.

To avoid flooding, we propose the idea of forcing only a small subset of the deployed hosts to move as per the needs of the protocol, i.e. we propose to exploit the support idea.

In simple terms, the protocol works as follows: The nodes of the support move fast enough so that they visit (in sufficiently short time) the entire motion graph. Their motion is accomplished in a distributed way via a support motion subprotocol $P_1$. When some node of the support is within communication range of a sender, an underlying sensor subprotocol $P_2$ notifies the sender that it may send its message(s).

The messages are then stored “somewhere within the support structure”. When a receiver comes within communication range of a node of the support, the receiver is notified that a message is “waiting” for him and the message is then forwarded to the receiver.

The messages received by the support are propagated within the structure when two or more members of the support meet on the same site (or are within communication range). A synchronization subprotocol $P_3$ is used to dictate the way that the members of the support exchange information.

In a first alternative implementation of the support protocol (the “snake” protocol), the nodes of the support move in a coordinated way, always remaining pair wise adjacent (i.e., forming a chain of $k$ nodes), so that they sweep (given some time) the entire motion graph.

A different approach to implement $MS$ is to allow each member of $\Sigma$ not to move in a snake-like fashion, but to perform an independent random walk on the motion graph $G$, i.e., the members of $\Sigma$ can be viewed as “runners” running on $G$. In other words, instead of maintaining at all times pair wise adjacency between members of $\Sigma$, all hosts sweep the area by moving independently of each other. When two runners meet, they exchange any information given to them by senders encountered using a new synchronization subprotocol $P_3^R$. As in the snake case, the same underlying sensor sub-protocol $P_2$ is used to notify the sender that it may send its message(s) when being within communication range of a node of the support.
2.3 Modeling and Analysis Summary

We assume that the head of the snake does a continuous time random walk on $G(V,E)$, without loss of generality (if it is a discrete time random walk, all results transfer easily, see [1]). We define the random walk of a host on $G$ that induces a continuous time Markov chain $M_G$ as follows: The states of $M_G$ are the vertices of $G$ and they are finite. Let $s_t$ denote the state of $M_G$ at time $t$. Given that $s_t = u$, $u \in V$, the probability that $s_{t+dt} = v$, $v \in V$, is $p(u,v) \cdot dt$ where

$$p(u,v) = \begin{cases} \frac{1}{d(u)} & \text{if } (u,v) \in E \\ 0 & \text{otherwise} \end{cases}$$

and $d(u)$ is the degree of vertex $u$.

We assume that all random walks are concurrent and that there is a global time $t$, not necessarily known to the hosts.

**Note 2.1** Since the motion graph $G$ is finite and connected, the continuous Markov chain abstracting the random walk on it is automatically time-reversible.

**Definition 2.2** $P_i(E)$ is the probability that the random walk satisfies an event $E$ given it started at vertex $i$.

**Definition 2.3** For a vertex $j$, let $T_j = \min\{t \geq 0 : s_t = j\}$ be the first hitting time of the walk onto that vertex and let $E_iT_j$ be its expected value, given that the walk started at vertex $i$ of $G$.

**Definition 2.4** For the random walk of any particular host, let $\pi()$ be the stationary distribution of its position after a sufficiently long time.

We denote $E_{\mu}[\cdot]$ the expectation for the chain started at time 0 from any vertex with distribution $\mu$ (e.g. the initial distribution of the Markov chain).

We know (see [1]) that for every vertex $\sigma$, $\pi(\sigma) = \frac{d(\sigma)}{2m}$ where $d(\sigma)$ is the degree of $\sigma$ in $G$ and $m = |E|$.

**Definition 2.5** Let $p_{j,k}$ be the transition probability of the random walk from vertex $j$ to vertex $k$. Let $p_{j,k}(t)$ be the probability that the random walk started at $j$ will be at $k \in V$ in time $t$.

**Definition 2.6** Let $X(t)$ be the position of the random walk at time $t$.

By using the continuous time Markov Chain above for the walk on the graph, we show that the expectation of the first meeting time of the support with either the sender or the receiver node is bounded by a function of the motion graph size only. To do this we construct strong stationary time sequences for the head’s walk, by considering a sequence of stopping times to control separation from stationarity. Thus we prove correctness:

**Theorem 2.1** The support $\Sigma$ and the management subprotocol $M_\Sigma$ guarantee reliable communication between any sender-receiver ($MH_S, MH_R$) pair in finite time, whose expected value is bounded only by a function of the relative motion space size $\rho$ and does not depend on the number of hosts, and is also independent of how $MH_S, MH_R$ move, provided that the mobile hosts not in the support do not deliberately try to avoid the support.

For the average case time efficiency of the communication protocol we study hitting times of the support’s walk to the sender, receiver nodes and provide the following upper bound that uses spectral characteristics of the motion graph.
Theorem 2.2 The expected communication time of our scheme is bounded above by the formula

\[ E(T_{\text{total}}) \leq \frac{2}{\lambda_2(G)} \Theta\left(\frac{n}{k}\right) + \Theta(k) \]

where \( \lambda_2 \) is the second eigenvalue of the adjacency matrix of the graph, \( n \) relates to the motion area size and \( k \) is the support size.

We also prove improved communication times under special network assumptions, as well as tight bounds for the worst case of motion and lower bounds.

3 Stochastic Interaction of Particles

3.1 Introduction and Model

Properties of interacting particles (moving on a finite graph) are of interest within a number of areas of Science, primarily Physics in the early days, but increasingly Biology, the Social Sciences and Computer Science today, since they model interesting phenomena like magnetism, spatial competition, tumor growth, spread of infection, economic systems and mobile communication. The book by Liggett ([13]) and Chapter 14 of the Aldous and Fill book ([1]) are good references.

We consider here a special case of interacting particles: The motion of each of the particles is a random walk on the graph. The walks are concurrent; for simplicity, the initial positions of the particles are those of the steady-state probability distribution of the walks.

Let \( k \) the number of particles. Let \( G(V,E) \) the graph of the walks. Let \( |V| = n \geq k, |E| = m \). Initially, one particle is red and all others are white. Let \( \pi() \) the steady-state distribution of each of the walks. The particles interact according to the following Infection Rule: When a red particle meets (at a graph node) one (or more) white particle(s), the white particle(s) turn red.

Initially, each vertex of \( G \) can have at most one particle (thus \( k \leq n \)). Let \( \phi \) be the initial distribution of particles.

Definition 3.1 Let \( T_k^\phi \) be the least time instant at which all particles are red.

Note that \( T_k^\phi \) is a random variable. We are interested here in the expected value of \( T_k^\phi, E_{\phi}T_k^\phi \).

Definition 3.2 We call Infection Time, w.r.t. an initial distribution \( \phi \), the value of \( E_{\phi}T_k^\phi \).

Definition 3.3 Let an initial probability distribution \( \phi \) be called pure when it assigns a single position (vertex) to each particle, with probability 1.

Definition 3.4 We call the Infection Time \( T_k \) the worst \( E_{\phi}T_k^\phi \) over all pure distributions \( \phi \).

Note that \( \pi() \) always exists when the random walk is of continuous time with transition rates \( q_{v,v} = \frac{1}{d_v} \) if \( \{v,v\} \) is an edge (\( d_v \) is the degree of vertex \( v \)), and \( q_{v,v} = 0 \) if not, because, then, the random walk is aperiodic and ergodic.

In fact, we start by considering continuous time walks. Note that the discrete-time walks defined by the transition probability \( p_{uv} = \frac{1}{d_u} \) if \( \{u,v\} \in E \), and \( p_{uv} = 0 \) else, have the same stationary distribution mean hitting times as in the continuous case ([1]).
3.2 Summary of Results and Related Work

We here present some first results achieved in [9].

We first show an upper bound on the expected infection propagation time $T_k$ for any $k > 2$ and for any undirected graph $G$, where we consider continuous time walks.

We demonstrate that this bound is tight on a Lollipop graph for certain values of $k$ and an initial position of particles.

Then, we turn into discrete time walks and derive much smaller bounds on $T_k$ for the clique and for expander graphs. Note that our model of continuous time walks is just the “continuization” of the corresponding discrete time chains. Thus the results hold in both cases.

Previous work and Comparison. Coppersmith et al ([5]) evaluated the expected meeting time of two random walks (i.e. $T_2$). They showed that in the worst case (and assuming discrete time), $T_2 = (4/27 + o(1))n^3$. Tetali and Winkler ([19]) gave some earlier bounds on $T_2$. Sunderam and Winkler ([18]) examined a related but different problem: each node of a clique is a processor having some information piece. They estimate techniques to minimize the time when all processors know all pieces of the information. There, processors do not move and messages are exchanged in discrete, synchronized rounds.

Works on other models of interacting particles (e.g. the anti-voter and voter models), can be found in Ch. 14 of [1] and in [13] (see also next subsection). Infection models were studied well in the past, under the direction of population biology. Their targets are similar but the main difference is the missing of the graph as space of motions. See [16] and references there.

So, our work extends the results of [5, 19] to $k > 2$ concurrent random walks on a finite undirected graph.

3.3 Other Models of Stochastic Particle Interaction

A well known model is the voter model which we describe below. Let $G$ be an $r$-regular graph with $n$ vertices. In the voter model we envisage a person at each vertex. Each of the $n$ persons has an opinion (say person $i$ has opinion $i$) which is initially different from the opinions of other persons. As time passes each person can change his opinion according to the following rule. For each person $i$ and each time interval $[t, t + dt]$, with probability $dt$ the person chooses uniformly at random a neighbour of his and changes (if necessary) his opinion to the current opinion of the neighbour he chose.

Another model which is closely related to the voter model is the untivoter model. This model differs from the voter model in two ways. First, we suppose that there are only two different opinions, which we denote by $\pm 1$. Second, the evolution rule is as follows: for each person $i$ and each time interval $[t, t + dt]$, with probability $dt$ the person chooses uniformly at random a neighbour of his and changes (if necessary) his opinion to the opposite opinion of that neighbour. We note that while in the voter model the number of different opinions in the graph is decreasing with time (and eventually reaches 1), this is not the case in the antivoter model since opinions don’t disappear.

3.4 Techniques and main results

We use the following basic tools for the analysis: exponential tails of hitting times and meeting times.

Definition 3.5 Let $A \subseteq V$ and $i \in V$. Define $E_iT_A$ to be the expected value of the first hitting time to $A$ (i.e. the first time to arrive at a node of $A$) of a random walk on $G$, starting from vertex $i$. Let $T_A$ be the corresponding random variable. Let $t^*_A = \max_{i \in V} E_iT_A$.

Definition 3.6 Let $E$ an event about a random walk on $G$. We denote by $Pr_\phi(E)$ the probability of $E$ when the starting position of the walk is according to the distribution $\phi()$. 

5
In continuous time, if we choose \( s = et_A^* \), we get

**Lemma 3.1 (SubExponentiality Lemma 1a) ([1])** For any \( t : 0 < t < \infty \) it is

\[
\sup_{\phi} \Pr\{T_A > t\} \leq \exp(-|t/et_A^*|)
\]

The machinery stated above can be used also for the meeting times:

**Definition 3.7** Let \( M_{i,j} \) be the first time that two independent copies of a random walk on \( G \) meet given that they start from \( i, j \). Let \( m^* = \max_{i,j} M_{i,j} \).

Working exactly as above, we then get:

**Lemma 3.2 (SubExponentiality Lemma 1b) ([1])**

\[
\Pr\{M_{i,j} > t\} \leq \exp(-|t/et_m^*|)
\]

Another interesting fact about the relation of hitting times and meeting times, for continuous time walks, is the following:

**Lemma 3.3 (Aldous and Fill [1], Chapter 14, Proposition 5)**

\[
\max_{i,j} E M_{i,j} \leq \max_{i,j} E T_{i,j}
\]

Our first result is a tight upper bound on infection time for general graphs. Consider, instead of our process (call it \( P_1 \)) of infection, another process of the same walks governed by the rule:

“When particles meet, they coalesce into clusters and the cluster thereafter sticks together and moves as a single random walk. If the cluster contains red particles then it is colored red”.

**Definition 3.8** Let \( C_{k,n} \) be a random variable, which is the time at which all particles coalesce into one single cluster.

Clearly, \( \forall t > 0 \), it is \( \Pr\{C_{k,n} > t\} \geq \Pr\{T_k > t\} \), and thus

\[
E_{\phi} C_{k,n} \geq E_{\phi} T_k
\]

for all initial position distributions \( \phi \). But \( EC_{k,n} \) can be bounded as follows:

**Lemma 3.4** \( EC_{k,n} \leq e m^*(2 + \log k) \),

where \( e \) is the basis of the natural logarithms.

we get

**Theorem 3.1** \( T_k \leq e m^*(2 + \log k) \)

Although this bound is crude for a lot of graphs, there are graphs and positions of particles achieving it. Consider a special case of the Lollipop graph, with a clique of size \( n - \sqrt{n} \) and a path of length \( l = \sqrt{n} \) extending out of it. Initially, there is one red particle in the clique. There are \( k \) white particles, all being at the furthest \( k \) vertices of the path. Here \( k \ll \sqrt{n} \). In the sequel, take \( k \) so that \( \sqrt{n} - k = \Theta(\sqrt{n}) \).

We can show that:
Lemma 3.5 \( E(S) = \Theta(s^* \log k) \)
But \( s^* = \Theta(m^*) \) for the \( \sqrt{n} \) line as we said. So,

Lemma 3.6 \( E(S) = \Theta(m^* \log k) \)
Thus, we get:

Theorem 3.2 In the Lollipop graph and for \( k \) such that \( \sqrt{n} - k = \Theta(\sqrt{n}) \), there is a set of initial positions of particles so that \( T_{k+1} = \Theta(m^* \log k) \). Thus, the bound of Theorem 1 is tight.

Note however than even in the lollipop graph, for large \( k \), the infection time is much smaller than this upper bound. This is demonstrated in the experiments section and we believe that this can also be analytically shown.

The bound stated in Theorems 3.1 and 3.2 is not tight for “close to regular graphs”. We derive much better bounds for such graphs.

Theorem 3.3 Consider \( k \leq n \) particles, one red, \( k - 1 \) white, sitting on the nodes of a \( K_n \). Then the expected time when all particles become red after randomly walking on the \( K_n \) is bounded by

\[
\frac{4n}{k-1}(\ln k + O(1)).
\]

We also derive time bounds for graphs with good expansion (regular graphs, \( G_{n,p} \) random graphs).

4 A Proximity-Sensitive Model of Interactions

Random graphs, introduced by P. Erdős and A. Rényi in their celebrated work [10] in 1959, still continue to attract a huge amount of research and interest in the communities of Theoretical Computer Science, Graph Theory and Discrete Mathematics. This growing interest is due to (at least) the following reasons: a) the combinatorial interest of these structures themselves, as well as their mathematical beauty and conceptual challenges posed by their investigation, b) their motivation by real world aspects and in particular the issue of “reliable network computing”, in the sense that the random elements in these graphs may capture faults or unavailabilities in corresponding networks, c) the random graph is a “somewhat” typical instance and is thus heavily used in the average case analysis of graph algorithms.

There exist various models of random graphs. The most famous is the \( G_{n,p} \) random graph, a sample space whose points are graphs produced by randomly sampling the edges of a graph on \( n \) vertices independently, with the same probability \( p \). Because of the (technically convenient) independence properties of \( G_{n,p} \) graphs, this model has been extensively studied, both from the combinatorial as well as from the algorithmic point of view. Other models have also been quite a lot investigated: \( G_{n,r} \) (the “random regular graphs”, produced by randomly and equiprobably sampling a graph from all regular graphs of \( n \) vertices and vertex degree \( r \)), \( G_{n,M} \) (produced by randomly and equiprobably selecting an element of the class of graphs on \( n \) vertices having \( M \) edges). For an excellent survey of these models, see [3, 2].

We here present both combinatorially and algorithmically, a new model of random graphs. We nontrivially extend the \( G_{n,m,p} \) model (“random intersection graphs”) introduced by M. Karoński, E.R. Sheinerman and K.B. Singer-Cohen [12] and K.B. Singer-Cohen [17]. In their model, to each of the \( n \) vertices of the graph, a random subset of a universal set of \( m \) elements is assigned, by independently choosing elements with the same probability \( p \). Two vertices \( u, v \) are then adjacent in the \( G_{n,m,p} \) graph if and only if their assigned sets of elements have at least one element in common. We extend this model (which we call hereafter “uniform”, because of the same probability of selecting elements) by proposing two new models which we define below.
Definition 4.1 (General random intersection graph) Let us consider a universe $M = \{1,2,\ldots,m\}$ of elements and a set of vertices $V = \{v_1,v_2,\ldots,v_n\}$. If we assign independently to each vertex $v_j$, $j = 1,2,\ldots,n$, a subset $S_{v_j}$ of $M$ by choosing each element $i \in M$ independently with probability $p_i$, $i = 1, 2, \ldots, m$, and put an edge between two vertices $v_{j_1}, v_{j_2}$ if and only if $S_{v_{j_1}} \cap S_{v_{j_2}} \neq \emptyset$, then the resulting graph is an instance of the general random intersection graph $G_{n,m,\vec{p}}$, where $\vec{p} = [p_1,p_2,\ldots,p_m]$.

Definition 4.2 (Regular random intersection graph) Let us consider a universe $M = \{1,2,\ldots,m\}$ of elements and a set of vertices $V = \{v_1,v_2,\ldots,v_n\}$. If we assign independently to each vertex $v_j$, $j = 1,2,\ldots,n$, a subset $S_{v_j}$ consisting of $\lambda$ different elements of $M$, randomly and uniformly chosen, and draw an edge between two vertices $v_{j_1}, v_{j_2}$ if and only if $S_{v_{j_1}} \cap S_{v_{j_2}} \neq \emptyset$, then the resulting graph is an instance of the regular random intersection graph $G_{n,m,\lambda}$.

The latter model may abstract $\lambda$-SAT random formulae. We note the following:

Note 1: When $p_1 = p_2 = \cdots = p_m = p$ the general random intersection graph $G_{n,m,\vec{p}}$ reduces to the $G_{n,m,p}$ as in [12] and we call it the uniform random intersection graph.

Note 2: When in the uniform case $mp \geq \alpha \log n$ for some constant $\alpha > 1$ then the model $G_{n,m,p}$ and the model $G_{n,m,\lambda}$ for $\lambda \in (1 \pm \epsilon)mp$, $\epsilon \in (0,1)$, are essentially equivalent, i.e. they assign almost the same probability to edge monotone graph events. This follows from degree concentration via Chernoff bounds. Thus, all our results proved here for $G_{n,m,p}$ translate to $G_{n,m,\lambda}$.

Let us now emphasize the importance of random intersection graphs. First of all, we note that (as proved in [14]) any graph is a random intersection graph. Thus, the $G_{n,m,p}$ model is very general. Furthermore, for some ranges of the parameters $m,p$ ($m = n^\alpha, \alpha > 6$) the spaces $G_{n,m,p}$ and $G_{n,p}$ are equivalent (as proved by Fill, Sheinerman and Singer-Cohen [11], showing that in this range the total variation distance between the graph random variables has limit 0).

Second, random intersection graphs (and in particular our new, non-uniform model) may model real-life applications more accurately (compared to the $G_{n,p}$ case). This is because in many cases the independence of edges is not well-justified. In fact, objects that are closer (like moving hosts in mobile networks or sensors in smart dust networks) are more probable to interact with each other. Even epidemiological phenomena (like spread of disease) tend to be more accurately captured by this “proximity-sensitive” random intersection graphs model. Other applications may include oblivious resource sharing in a distributed setting, interactions of mobile agents traversing the WEB etc.

The model of uniform random intersection graphs was introduced by M. Karoński, E.R. Sheinerman and K.B. Singer-Cohen [12]. In the same work the authors study the evolution of such graphs with respect to the existence of small induced subgraphs. Other important properties, including connectivity, cliques and independent sets are investigated in [17]. The question of how close $G_{n,m,p}$ and $G_{n,p}$ are for various values of $m,p$ has been studied by Fill, Sheinerman and Singer-Cohen in [11].

Also, geometric proximity between randomly placed objects is nicely captured by the model of random geometric graphs (see e.g. [6, 8]) and important variations (like random scaled sector graphs, [7]).

References


