

**SOBRE O COMPORTAMENTO DE AGENTES
RACIONAIS
EM REDES COMPLEXAS**

PEDRO OLMO STANCIOLI VAZ DE MELO

**SOBRE O COMPORTAMENTO DE AGENTES
RACIONAIS
EM REDES COMPLEXAS**

Tese apresentada ao Programa de Pós-Graduação em Ciência da Computação do Instituto de Ciências Exatas da Universidade Federal de Minas Gerais como requisito parcial para a obtenção do grau de Doutor em Ciência da Computação.

ORIENTADOR: ANTONIO ALFREDO FERREIRA LOUREIRO
CO-ORIENTADOR: VIRGILIO AUGUSTO FERNANDES ALMEIDA,
CHRISTOS FALOUTSOS

Belo Horizonte
Setembro de 2011

PEDRO OLMO STANCIOLI VAZ DE MELO

**ON THE BEHAVIOR OF RATIONAL AGENTS
IN COMPLEX NETWORKS**

Thesis presented to the Graduate Program in
Computer Science of the Federal University
of Minas Gerais in partial fulfillment of the re-
quirements for the degree of Doctor in Com-
puter Science.

ADVISOR: ANTONIO ALFREDO FERREIRA LOUREIRO
CO-ADVISOR: VIRGILIO AUGUSTO FERNANDES ALMEIDA, CHRISTOS
FALOUTSOS

Belo Horizonte

September 2011

Melo, Pedro Olmo Stancioli Vaz de.

M528s Sobre o comportamento de agentes racionais em redes complexas / Pedro Olmo Stancioli Vaz de Melo. — Belo Horizonte, 2011.
xxix, 140 f. : il. ; 29cm

Minas Tese (doutorado) — Universidade Federal de Gerais - Departamento de Ciência da Computação.

Orientador: Antônio Alfredo Ferreira Loureiro.
Coorientador: Virgílio Augusto Fernandes Almeida.

1. Computação - Teses. 2. Redes de computadores - Teses. 3. Redes Complexas - Teses. I. Orientador. II. Orientador. III. Título.

519.6*22(043)



UNIVERSIDADE FEDERAL DE MINAS GERAIS
INSTITUTO DE CIÊNCIAS EXATAS
PROGRAMA DE PÓS-GRADUAÇÃO EM CIÊNCIA DA COMPUTAÇÃO

FOLHA DE APROVAÇÃO

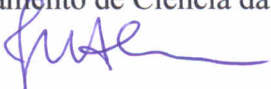
Sobre o comportamento de agentes racionais em redes complexas

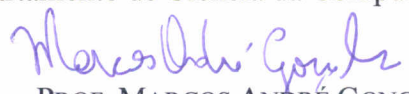
PEDRO OLMO STANCIOLI VAZ DE MELO

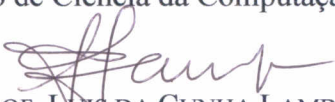
Tese defendida e aprovada pela banca examinadora constituída pelos Senhores:


PROF. ANTONIO ALFREDO FERREIRA LOUREIRO - Orientador
Departamento de Ciência da Computação - UFMG


PROF. VIRGÍLIO AUGUSTO FERNANDES ALMEIDA - Co-orientador
Departamento de Ciência da Computação - UFMG


PROFA. JUSSARA MARQUES DE ALMEIDA
Departamento de Ciência da Computação - UFMG


PROF. MARCOS ANDRÉ GONÇALVES
Departamento de Ciência da Computação - UFMG


PROF. LUIS DA CUNHA LAMB
Departamento de Informática Teórica - UFGRS


DR. CARLOS ALBERTO KAMIENSKI
Centro de Matemática, Computação e Cognição - UFABC

Belo Horizonte, 19 de setembro de 2011.

I dedicate this thesis to my mom and dad.

Acknowledgments

First, I thank my family and friends for all the support. Special thanks to my mom and dad, who were there for me all the time I needed. This won't be forgotten. Then, I thank everybody who contributed and helped me in writing this thesis. My advisor Antonio Loureiro, who was always with me, giving valuable advices and keeping my head together after the bad waves. My co-advisor Virgilio Almeida, who lead my work and my life to a completely different and better direction. My advisor at CMU, Christos Faloutsos, for all the unexpected time spent in my work and for being so bright, committed and humble, an example that I will take for my life. Raquel Mini, who believed in me and helped me to join the doctorate program. My colleagues at UFMG, CMU and INRIA for all the ideas and for making this journey much more fun. Special thanks to Fernando, Guidoni, Fernanda, Felipe, Heitor, Marcelo, Leandro, Dudu, Leman and Robson. I also thank all the professors at UFMG who thought me so much inside and outside the classroom.

*“Why is it that when one man builds a wall, the next man immediately needs to know what’s
on the other side?”*

(George R.R. Martin)

Resumo

O avanço constante dos sistemas de informação permite, a uma taxa de crescente, que mais dados sejam gerados e armazenados. A partir das mais rotineiras situações, tais como conversas telefônicas, a atuação de sistemas tecnológicos de alta complexidade, tais como redes de sensores sem fio (RSSF) para detectar eventos climáticos, dados são gerados e armazenados registrando cada ação e decisão tomada pelos agentes desses sistemas. É fascinante que, por trás desses registros, vemos o reflexo do ambiente em si, já que por trás de cada registro, há uma decisão tomada por alguma entidade. Portanto, o conhecimento de como processar esse valioso banco de dados em evolução pode levar, conseqüentemente, para uma melhor compreensão dos interesses e da dinâmica de cada entidade em um determinado ambiente ou na sociedade.

Nesta tese, nos concentramos em sistemas que são compostos de entidades (indivíduos, organizações e sistemas computacionais), capazes de interagir entre si de uma maneira racional, refletindo seus interesses e dinâmica de atividade. Nós chamamos esses sistemas de *Redes Complexas Baseadas em Decisão (RCBD)*, estes indivíduos e/ou organizações, nós ou agentes, e as interações entre nós, arestas. A principal característica de uma RCBD é que ela evolui de acordo com as motivações pessoais dos seus nós. Portanto, o objetivo principal desta tese é analisar o comportamento dos agentes de uma RCBD. Pretendemos observar cenários reais e hipotéticos, onde as decisões têm um papel importante na evolução da rede. Quando compreendemos plenamente as motivações por trás das ações dos agentes, poderemos modelar, prever e controlar o seu comportamento. Dividimos o objetivo principal desta tese em três objetivos específicos: (i) **Modelagem**, em que o nosso objetivo é representar fielmente o comportamento dos agentes em uma RCBD; (ii) **Predição**, em que o nosso objetivo é prever como evoluirá o sistema, e (iii) **Controle**, em que o nosso objetivo é criar mecanismos para fazer os agentes agirem de acordo com um objetivo determinado.

Em primeiro lugar, analisamos as redes de comunicação formada a partir de registros telefônicos de um operadora móvel privada de uma grande cidade. Propomos modelos para o comportamento individual dos usuários dessa rede e, a partir disso, propomos aplicações para redução de dados, detecção de anomalias e monitoramento de rede. Então, analisamos

redes competitivas formadas a partir de ligas esportivas, como a liga profissional de basquete americana (NBA) e a liga profissional de *baseball* (MLB). Propomos modelos de previsão que podem ser usados para identificar as equipes mais prováveis para ganhar a temporada seguinte. Finalmente, investigamos a tomada de decisões em redes de computadores, em um cenário onde um grupo de RSSFs são implantados na mesma região e elas podem interagir entre elas para solicitar ou compartilhar recursos computacionais. Nós modelamos o problema da cooperação entre duas ou mais RSSFs diferentes pelos conceitos da teoria dos jogos e com isso nós apresentamos, como uma aplicação de controle, um protocolo para permitir a cooperação entre eles.

Palavras-chave: Redes Sociais, Redes Complexas, Tomada de Decisão.

Abstract

The constant advancement of information systems allows, at a growing rate, more data to be generated and stored. From routine situations, such as phone conversations, to the actuation of highly complex technological systems, such as Wireless Sensor Networks (WSNs) to detect weather events, data are generated and stored registering every action and decision made by the agents of these systems. It is fascinating that, behind these records, we see the reflection of the environment itself, since behind every record, there is a decision made by some entity. Therefore, the knowledge of how to process this valuable and very large evolving database can lead, consequently, to a better understanding of the interests and dynamics of each entity in a determined environment or in the society.

In this thesis, we focus on systems that are made up of entities (individuals, organizations and computational systems) capable of interacting among themselves in a rational way, reflecting their interests and activity dynamics. We call these systems *Decision-based Complex Networks (DBCN)*, these individuals and/or organizations, nodes or agents, and the interactions between nodes, edges. The main characteristic of a DBCN is that it evolves according to the personal motivations of its nodes. Therefore, the main objective of this thesis is to analyze the behavior of agents of DBCN. We plan to observe real and hypothetical scenarios where decisions play an important role in the evolution of the network. When we fully understand the motivations behind the actions of the agents, we may be able to model, predict and control their behavior. We divide the main objective of this thesis in three specific goals: (i) **Modeling**, where we aim to accurately represent the behavior of the agents in a DBCN; (ii) **Predicting**, where we aim to predict how the system will evolve, and (iii) **Controlling**, where we aim to be able to make the agents to act according to a determined goal.

First, we show a compact analysis of the three aspects of DBCNs we tackle in this thesis, i.e., modeling, predicting and controlling, in three real-world datasets describing user mobility activity. Then, we focus solely on modeling communication networks. We proposed models for the individual behavior of the users of this network and, from this, we proposed applications for data summarization, anomaly detection and network monitoring.

Then, we focus on predicting in competitive networks formed from sports leagues such as the North American National Basketball Association (NBA) and the Major League Baseball (MLB). We proposed a prediction model that can be used to identify likely teams to win and to fail in a following season. Finally, we focus on controlling, investigating decision making in computer networks, in a scenario where a group of WSNs are deployed in the same region and they may interact with themselves to request or share computational resources. We modeled the problem of cooperation among two or more different WSNs by the concepts of game theory and from it we present, as a control application, a protocol to allow cooperation among them.

Palavras-chave: Social Networks, Complex Networks, Decision Making.

Resumo Estendido

Motivação

O avanço constante dos sistemas de informação permite, a uma taxa de crescente, que mais dados sejam gerados e armazenados. A Internet, por exemplo, consiste em milhões de dispositivos computacionais sendo que cada um deles gera, armazena e transmite uma quantidade incontável de dados. Claramente, a Internet está indo na direção do paradigma de computação ubíqua, que, tal como previsto no artigo clássico de Mark Weiser, permite que qualquer pessoa, em qualquer lugar, a qualquer momento, interaja com o ambiente. Assim, a partir das mais rotineiras situações, tais como conversas telefônicas, a atuação de sistemas tecnológicos de alta complexidade, tais como as redes de sensores sem fio (RSSF) para detectar eventos climáticos, dados são gerados e armazenados registrando cada ação e decisão tomada pelos agentes desses sistemas.

Atualmente, existem estudos sobre os dados de chamadas telefônicas, redes sociais online, ferrovias, sites de Internet, redes de citação, filmes e atores, ligas desportivas e muitos outros. A partir desses estudos agora sabemos como as pessoas vinculam sites em suas páginas e como comunidades de pessoas em uma rede social online evoluem ao longo do tempo. É fascinante que, por trás desses registros, vemos o reflexo do ambiente em si, já que por trás de cada registro, há uma decisão tomada por alguma entidade. Portanto, o conhecimento de como processar esse valioso banco de dados em evolução pode levar, conseqüentemente, para uma melhor compreensão dos interesses e da dinâmica de cada entidade em um determinado ambiente ou na sociedade.

Nesta tese, nos concentramos em sistemas que são compostos de entidades capazes de interagir entre si de uma forma autônoma, refletindo os seus interesses e dinâmicas de atividade. Nós chamamos esses sistemas *Redes Complexas Baseadas em Decisão* (RCBD) ou *Decision-based Complex Networks* (DBCN) e suas entidades de nós ou agentes. Uma RCBD é um tipo especial de rede complexa que contém nós capazes de tomar decisões autônomas, que são guiados principalmente por suas motivações pessoais. Por exemplo, as redes sociais formadas a partir de laços de amizade ou colaborações de trabalho são RCBDs,

uma vez que os nós dessas redes têm poder de decisão para criar arestas. No entanto, redes semânticas formado a partir de textos e redes de terremotos não são RCBDs, uma vez que a criação de arestas é guiada, nestes casos, por um processo central.

Assim como as redes complexas, RCBDs têm um grande número de vértices e arestas que seguem um ou vários padrões, tais como comunidades de nós ou vértices altamente conectados, chamados de *hubs*. Enquanto em uma rede simples com, no máximo, centenas de nós o olho humano é um instrumento de poder considerável, em uma complexa rede, esta abordagem é inútil. Assim, para estudar, analisar e caracterizar redes complexas, métodos estatísticos e algoritmos eficientes são necessários.

Objetivos

Esta tese tem como objetivo analisar o comportamento dos agentes de RCBDs. Observamos cenários reais e hipotéticos em que as decisões ambientais e sociais desempenham um papel importante na evolução da rede e sobre a forma como os agentes interagem. Quando somos capazes de entender completamente as motivações por trás das ações dos agentes, é possível desenvolver técnicas para modelar, prever e controlar o comportamento de diversas e grandes RCBDs e das suas entidades. Mas especificamente, os principais objetivos desta tese são:

1. **Modelagem.** Nosso objetivo é criar modelos que podem representar com precisão o comportamento dos agentes de uma RCBD, levando a um melhor entendimento do sistema;
2. **Predição.** A partir da compreensão das razões por trás das decisões dos agentes de uma RCBD, queremos entender como o sistema irá evoluir;
3. **Controle.** Uma vez que sabemos como o sistema se comportará, proporemos mecanismos de controle para fazer os agentes agirem de acordo com um objetivo determinado.

A partir das decisões locais feitas por agentes de RCBDs, investigamos a evolução global do sistema a fim de propor uma grande variedade de aplicações. Nesta tese, estudamos tipos muito diferentes de RCBDs e propomos diferentes aplicações para todas elas a partir do conhecimento que adquirimos do comportamento dos agentes. Todas as redes analisadas têm em comum o fato de que os agentes são capazes de: (i) tomar decisões autônomas, e (ii) interagir com outros agentes. Além disso, é importante ressaltar que a evolução da rede depende exclusivamente de como estas decisões são tomadas.

Contribuições

Primeiro, apresentamos uma análise compacta dos três aspectos de RCBDs que abordamos nesta tese (modelagem, predição e controle) usando três conjuntos de dados de mobilidade do mundo real (Capítulo 3). Em seguida, nos concentramos na modelagem de redes de comunicação (Capítulo 4). Propomos modelos para o comportamento individual dos usuários e, a partir disso, propomos aplicações de redução de dados, detecção de anomalias e monitoramento de rede. Em seguida, nos concentramos no problema de predição em redes competitivas, como a liga profissional de basquete (NBA) e de *baseball* (MLB) dos Estados Unidos da América (Capítulo 5). Propomos um modelo de predição para identificar o desempenho anual de equipes em ligas esportivas. Finalmente, nos concentramos no aspecto de controle em RCBDs, investigando a tomada de decisão em um cenário onde redes de sensores sem fio são depositadas na mesma região e interagem entre si para solicitar ou compartilhar recursos computacionais (capítulo 6). Modelamos este problema a partir da teoria dos jogos e apresentamos uma aplicação de controle que é um protocolo para permitir a cooperação entre essas redes.

Capítulo 3 – Redes de Mobilidade

A principal característica de RCBDs é que as interações entre as suas entidades são, geralmente, consequência de decisões semi-rationais. Escreve-se “normalmente” e decisões “semi-rationais” porque qualquer sistema está sujeito a eventos aleatórios e escolhas irracionais. No entanto, uma vez que a maioria das interações ainda decorrem de decisões conscientes feitas por suas entidades, a evolução de RCBDs é significativamente diferente da evolução de redes aleatórias como, por exemplo, redes de Erdős and Rényi. Assim, enquanto em uma RCBD as arestas são geradas a partir de decisões semi-rationais, que tendem a ser regulares e a se repetir, em uma rede aleatória as arestas são geradas independentemente dos atributos dos nós, ou seja, a probabilidade de dois nós se conectarem é constante.

Considere, por exemplo, uma RCBD constituída por pessoas e suas rotinas de mobilidade. Uma interação entre duas pessoas ocorre se elas se encontram. Se Silva e Moreira trabalham no mesmo escritório e suas horas de trabalho são de 8:00 às 18:00, pode-se prever facilmente que uma interação entre Silva e Moreira irá ocorrer em torno de 8:00 durante os dias da semana. Isso é baseado no fato de que acreditamos fortemente que tanto Silva quanto Moreira decidirão ir ao trabalho todos os dias pontualmente, já que este é, provavelmente, a decisão racional para se tomar. Entretanto, a maioria dos cenários está sujeita a eventos aleatórios que podem desviar o comportamento esperado dos agentes. Silva pode, por exemplo, viajar à negócios por alguns dias e encontrar pessoas totalmente diferentes. Ou então,

Moreira poderia ficar preso no trânsito e se atrasar, se encontrando com Silva somente às 10:00. O fato é que, apesar de decisões racionais serem regulares, as decisões aleatórias podem muitas vezes ocorrer também e se misturar em grande quantidade com as demais.

Assim, propomos o *Random rELationship CLASsifier sTrategy* (RECAST), um algoritmo classificador de relações sociais, capaz de detectar, entre outros, eventos aleatórios em RCBDs. Eventos aleatórios mascaram os padrões de comportamento comuns por introduzirem uma quantidade significativa de ruído, tornando assim o processo de descoberta de conhecimento em RCBDs uma tarefa ainda mais complexa. A capacidade de identificar com precisão os eventos aleatórios em grandes conjuntos de dados é essencial para a análise do comportamento social, bem como para aplicações que dependem de uma descrição precisa de rotinas humanas, tais como sistemas de recomendação, estratégias de roteamento *ad-hoc* e esquemas de divulgação de mensagens focalizando eficiência de cobertura com um número limitado de mensagens redundantes.

O RECAST nos permite observar muitas diferenças na evolução de RCBDs quando aplicada a três conjuntos de dados do mundo real descrevendo atividades de mobilidade. Foi verificado que essas diferenças são devido às características intrínsecas de cada cenário. Por exemplo, mostramos que o conjunto de dados que descreve o movimento de motoristas de táxi em San Francisco (EUA) tem, na sua maioria, propriedades não-sociais, o que torna a sua representação gráfica semelhante a uma rede aleatória. O mesmo não é verdade para pessoas se movendo em um campus, pois elas se encontram regularmente com um mesmo conjunto de pessoas, tais como seus colegas, professores etc. No entanto, verificamos que diferentes campi possuem dinâmicas de interação e de mobilidade diferentes.

Em resumo, neste capítulo modelamos conjuntos de dados de mobilidade em RCBDs de encontros e, a partir disso, propomos o RECAST. A clara classificação das relações dos usuários fornecida pelo RECAST alavanca imediatamente análises subsequentes. Por exemplo, mostramos que encontros entre duas pessoas que compartilham de um relacionamento social são mais fáceis de prever. Além disso, mostramos como é possível projetar uma aplicação de controle para disseminar dados na rede de forma eficiente usando apenas duas das quatro classes que o RECAST identifica.

Capítulo 4 – Redes de Comunicação

Neste capítulo mostramos que o problema da modelagem de agentes em redes de comunicação é extremamente desafiador. Padrões de comunicação humanos são propensos a mudar à medida que os aspectos tecnológicos e culturais da sociedade mudam. Por exemplo, a duração típica de uma chamada telefônica envolvendo dois telefones fixos é provavelmente diferente daquela que envolve dois telefones móveis. Por isso, usamos este cenário desafi-

ador no estudo da modelagem em RCBDs.

Analisamos a taxa na qual dois agentes sociais comunicam e a duração da sua comunicação. Primeiramente, analisamos o tamanho dos fluxos de comunicação (duração de chamadas de telefone), e mostramos como um bom esforço de modelagem pode conduzir a uma ampla variedade de aplicações. Em resumo, nós abordamos o seguinte problema: dada uma grande quantidade de registros telefônicos, qual é a melhor maneira de representar o comportamento da duração das chamadas de um usuário? Analisamos a duração de centenas de milhões de chamadas e propomos o modelo *Truncated Lazy Contractor* (TLAC) para descrever o tempo das durações telefônicas de um único usuário. Assim, o TLAC modela a distribuição da duração de chamadas (DDC) de um usuário e é parcimonioso, tendo apenas dois parâmetros, o *coeficiente de eficiência* ρ e o *coeficiente de fraqueza* β . Nós mostramos que o modelo TLAC foi a melhor alternativa para modelar a DDC de usuários de nosso conjunto de dados, principalmente porque produz uma distribuição que tem cauda e cabeça mais pesadas que as da distribuição log-normal, que é a distribuição mais comumente usada para modelar DDC.

Sugerimos também a utilização dos parâmetros do TLAC como a melhor e mais compacta forma de representar o comportamento da duração de chamadas de um usuário. Dessa maneira, propomos a *MetaDist* para modelar a população de usuários com um determinado comportamento de duração de chamadas. A *MetaDist* é a meta-distribuição dos parâmetros ρ_i e β_i da DDC de cada usuário i e, quando seus contornos são visualizados, a sua forma é surpreendentemente similar a uma distribuição gaussiana bivariada. Essa regularidade fascinante, observada em uma base de dados significativamente ruidosa, faz da *MetaDist* distribuição potencial a ser explorada no sentido de compreender melhor o comportamento da chamada de usuários móveis.

Na análise dos intervalos de tempo entre comunicações, foram considerados vários cenários. A literatura atual tem resultados aparentemente contraditórios para a distribuição marginal dos intervalos de tempo entre eventos. Alguns estudos afirmam bons ajustes com as leis de potência, outros com “processos de Poisson não homogêneos”. Fizemos uma modelagem elaborada que utiliza o conhecimento destes modelos, juntamente com novas observações em dados reais, para propor o *Self-feeding Process* (SFP), que representa muito bem as propriedades das marginais de vários conjuntos de dados de comunicação de grande porte, diversificados e do mundo real, tais como chamadas telefônicas, SMS, e-mails, e fóruns online. Mais importante, ele unifica as teorias existentes sobre a dinâmica da comunicação humana, gerando distribuições marginais com cauda pesada, caracterizada por rajadas de eventos e longos períodos de inatividade, além de possuir comportamento Poissoniano local. Além disso, o SFP é baseado em uma nova descoberta relatada nesta tese, de que há uma correlação positiva significativa entre intervalos de tempos consecutivos. Finalmente,

é importante ressaltar que o SFP é extremamente parcimonioso, usando no máximo dois parâmetros.

Capítulo 5 – Redes Competitivas

Em redes competitivas, os agentes do sistema competem entre si por uma recompensa ou recursos limitados. Como possíveis exemplos desse tipo de rede podemos citar redes de trabalhadores à procura de postos de trabalho, como visto no *LinkedIn*, e em ligas profissionais de esportes. Nesses casos, as organizações, empresas ou equipes querem contratar os melhores jogadores e técnicos, ao menor custo possível. Por sua vez, os jogadores e técnicos querem receber o maior salário possível e ainda trabalhar em uma boa equipe. Além disso, tanto as organizações quanto os jogadores e técnicos querem que as organizações cresçam, ou seja, querem expandir no mercado e conquistar títulos em suas ligas. Este cenário apresenta vários conflitos de interesses que podem revelar observações interessantes sobre os agentes sociais deste tipo de rede.

Nesta tese, analisamos a rede formada a partir dos times e jogadores de ligas esportivas. Começamos com a NBA em seus 63 primeiros anos de existência. Na etapa de modelagem, vemos a NBA como uma rede complexa em evolução. Em seguida, na fase de predição, propomos métricas que estão correlacionados com o comportamento dos times da NBA, levando em conta apenas a relação social e de trabalho entre os jogadores, treinadores e equipes. Então, com base nessas métricas, propomos modelos para prever o quão bem uma equipe irá jogar na temporada seguinte. Avaliamos também os modelos de predição sobre o conjunto de dados da *Major League Baseball*. Tanto o NetForY quanto o NetFor , que são os modelos propostos nesta tese, tem resultados surpreendentemente bem em ambos os conjuntos de dados.

Capítulo 6 – Redes de Sensores Sem Fio

Finalmente, analisamos RCBDs formada apenas por dispositivos computacionais. Neste tipo de rede, os agentes podem ser modelados como agentes sociais capazes de tomar decisões. O aspecto principal deste tipo de RCBD é que em redes de computadores agentes não tomam decisões irracionais, pois só podem agir de acordo com a forma que foram programados para agir. Isto significa que as decisões realizadas pelos agentes visam sempre maximizar a sua utilidade e nada mais, agindo de uma maneira puramente egoísta, e decisões irracionais causadas por instabilidades emocionais ou causas semelhantes não existem.

Nós consideramos um cenário em que diferentes redes de sensores sem fio (RSSF) com diferentes proprietários são depositadas no mesmo lugar. Neste cenário, existe a pos-

sibilidade de que um sensor de uma rede coopere com um sensor de outra rede. Quando duas RSSFs compartilham os seus nós sensores na execução de alguma atividade de forma inteligente, as duas redes podem melhorar as suas operabilidades, executando as suas atividades de forma mais eficiente. Apesar de ser óbvia e simples, esta idéia traz consigo muitas implicações que dificultam a cooperação entre as redes. Considerando que uma RSSF tem um caráter racional e egoísta, ela só irá cooperar com outra RSSF se esta oferecer serviços que justificam a cooperação.

Neste capítulo específico, mais uma vez abordamos os três aspectos de RCBDs que analisamos nesta tese: predição, modelagem e controle. Primeiro, temos o problema de modelagem da cooperação entre os diferentes RSSFs usando a teoria dos jogos, que é uma técnica interessante para modelar situações de conflito entre dois ou mais agentes racionais e egoístas. Em redes de computadores, decisões e métricas de utilidade (por exemplo, taxa de transferência e latência) são computacionalmente bem definidas, fazendo com que a teoria dos jogos seja uma ferramenta poderosa para modelar e principalmente prever o comportamento dos agentes. A partir deste conhecimento, pode-se projetar mecanismos de incentivo para controlar os seus comportamentos.

No problema da cooperação entre diferentes RSSFs depositadas no mesmo local, inicialmente nenhum nó sensor está cooperando com nós sensores de outras redes, só encaminhando mensagens pertencentes à sua própria rede. Uma rede N_i , que é um jogador racional e egoísta, deve, então, alterar a sua estratégia e fazer com que um ou mais dos seus nós sensores encaminhem pacotes de outras redes se, e somente se, isso for aumentar o seu *payoff* Π_i , que é um valor que representa o ganho que N_i terá com ou sem a cooperação. No entanto, se todas as redes mantiverem as suas estratégias e N_i alterar a sua estratégia inicial e fazer algum dos seus nós sensores cooperar e encaminhar mensagens de outras redes, o gasto de energia de N_i aumentará, fazendo que o *payoff* Π_i diminua. Assim, inicialmente, o jogo está em Equilíbrio de Nash, pois não é possível que uma rede aumente o seu *payoff* alterando a sua estratégia se todas as outras redes mantiverem as suas. Esse comportamento previsto para as redes só pode ser alterado caso haja intervenção externa como, por exemplo, a injeção de um protocolo distribuído de cooperação nos nós sensores.

Assim, como contribuição final deste capítulo, propomos protocolo *Virtual Cooperation Bond (VCB)*, que permite a cooperação entre diferentes RSSFs depositadas em um mesmo local. O protocolo proposto é totalmente distribuído, ou seja, não depende de nenhum controle central, estabelecendo a cooperação a partir de informações locais dos nós sensores. Além disso, o VCB garante que a cooperação só seja estabelecida quando for trazer benefícios para ambas as redes. Resultados de simulação mostram que o VCB economiza a energia das redes e prolonga o seus tempos de vida.

List of Figures

- 1.1 The summary of this thesis. From DBCNs scenarios, first we model the behavior of the agents and, from this, we are able to predict their behavior. Once we know how the agents will behave, we are able to design control mechanisms to make the agents to act according to a determined goal and, therefore, change the system. New modeling and predicting efforts should be made in the new system to better control its behavior. 3
- 2.1 Payoff matrix of “The Battle of the Sexes” game. 14
- 3.1 Snapshots of the networks G_t after two weeks. 19
- 3.2 The densification of the temporal graph $G_t(V_t, E_t)$ per each day t . (a) The number of new edges added to $G_t(V_t, E_t)$ per each day t . (b) The percentage of the graph that is covered by the new edges. 20
- 3.3 Evolution of the clustering coefficient of the three analyzed networks G_t and their random correspondents G_t^R 21
- 3.4 The complementary cumulative distribution function of the edge persistence (a,b,c) and topological overlap (d,e,f) for the three analyzed networks G_t and their random correspondents G_t^R after four weeks, $t = 28$ days. 23
- 3.5 The number of edges ((a) and (b)) and percentage of encounters ((c) and (d)) of a given class that appears in the first four weeks of data for a given value of p_{rnd} . 26
- 3.6 [Best viewed in color] Snapshots of the Dartmouth and USC networks after two weeks of interactions, considering only the social edges and only the random edges. *Friendship* edges are painted in blue, *Bridges* in red, *Acquaintanceship* in gray and *Random* in orange. 28
- 3.7 Evolution of the clustering coefficient of the Dartmouth and USC networks G_t when only *Random* edges are present compared to their random correspondents G_t^R 29

3.8	The distribution of encounters ((a) and (b)) and appearance ((c) and (d)) of each class in the one week <i>test set</i> as a function of p_{rnd}	29
3.9	The contamination of the network when edges of a selected class are removed. The vertical lines represent a 95% confidence interval.	32
3.10	The contamination of the networks using only edges of a selected class. The vertical lines represent a 95% confidence interval.	32
4.1	Comparison among the shapes of the log-normal, exponential and TLAC distributions.	42
4.2	Comparison of models for the distribution of the phone calls duration of a high talkative user, with 3091 calls. TLAC in red, log-normal in green and exponential in black. Visually, for the PDF both the TLAC and the log-normal distribution provide good fits to the CDD but, for the OR, the TLAC clearly provide the best fit.	44
4.3	Percentage of users' CDDs who were correctly fit vs. the user's number of calls c . The TLAC distribution is the one that provided better fittings for the whole population of customers with $c > 30$. It correctly fit more than 96% of the users, only significantly failing to fit users with $c > 10^3$, probably spammers, telemarketers or other non-normal behavior user.	45
4.4	Odds ratio of three talkative customers who were not correctly fit by the TLAC model.	46
4.5	Scatter plot of the parameters ρ_i and β_i of the CDD of each user i for the first month of our dataset. In (a) we can not see any particular pattern, but we can spot outliers. By plotting the isocontours (b), we can observe how well a bivariate Gaussian (c) fits the real distribution of the ρ_i and β_i of the CDDs ('meta-fitting')	47
4.6	Evolution of the <i>MetaDist</i> over the four months of our dataset. Note that the collective behavior of the customers is practically stable over time.	48
4.7	The TLAC lines of several customers plotted together. (a) We can observe that, given the negative correlation of the parameters ρ_i and β_i , that the lines tend to cross in one point. (b) We plot the isocontours of the lines together and approximately 50% of the customers have TLAC lines that pass on the high density point (duration=17s, OR=0.15).	49
4.8	Outlier whose CDD can not be modeled by the TLAC distribution.	50
4.9	Cumulative distributions for ρ and β . We can observe that ρ is lower bounded by 0.5 and β is lower bounded by 1. These values are coherent with the global intuition on human calling behavior.	51

4.10 Isocontours of the users' CDD efficiency coefficient ρ and their summarized attributes.	52
4.11 Isocontours of the users' CDD efficiency coefficient β and their summarized attributes.	53
4.12 The inter-event time distribution of the most talkative user of our four datasets, with 44785 SMS messages sent and received. We observe that both the power law fitting (PL fitting) with exponent 2 and the exponential fitting, generated by a PP, deviate from the real data. We also observe that the OR is very well fitted by a straight line with slope ≈ 1	55
4.13 IEDs of anonymous users of the PHONE dataset. Observe that for these three users, the Odds Ratio is well fitted by a power law. The MLE fitting gives the same values for the slope ρ	56
4.14 The goodness of fit of our proposed model. We show the Probability Density Function (PDF) of the R^2 s measured for every user in the four datasets. Observe that the R^2 value for the great majority of the users is close to 1.	57
4.15 The PDF of the slopes ρ_i measured for every user u_i of our four datasets. Observe that the typical ρ_i for the users of the phone, Enron and Digg datasets is approximately 1, while for the users of the SMS dataset is approximately 2.	58
4.16 The PDF of the medians μ_i measured for every user of our four datasets. Observe that the typical μ_i for the users of the phone, SMS and Enron datasets is approximately 1 hour, while for the users of the Digg dataset is approximately 2 minutes.	58
4.17 IEDs of anonymous users of the SMS dataset. Observe that for these three users, the Odds Ratio is well fitted by a power law. The MLE fitting gives the same values for the slope ρ	59
4.18 IEDs of three users of the Enron dataset. Observe that for these three users, the Odds Ratio is well fitted by a power law. The MLE fitting gives the same values for the slope ρ	60
4.19 IEDs of three stories of the DIGG dataset. Observe that for these three stories, the Odds Ratio is well fitted by a power law. The MLE fitting gives the same values for the slope ρ	61

4.20	I.I.D. fallacy: dependence between Δ_t and Δ_{t-1} . (a) The median of the inter-event times Δ_t s for their respective preceding Δ_{t-1} s for one talkative user of each of the four datasets and for a synthetic data generated by the SFP, all represented in logarithmic intervals. (b) The sample autocorrelation for the same individual of Figure 4.12 and for synthetic data generated by the SFP, PP and LLG-iid, with the same number of phone calls and median. Observe that both the real data and the SFP indicates Δ_t and Δ_{t-1} are positive correlated.	62
4.21	The empirical probability of an individual's inter-event times to be autocorrelated given his/her number of events. Note that as the number of events grows, the probability of having an autocorrelated series also grows.	63
4.22	Comparison of the inter-event times generated by the SFP0 model with the inter-event times of the user of Figure 4.12. Observe that both the PDF (a) and the OR (b) are almost identical.	65
4.23	Unification Power of SFP: non-Poisson/bursty in the long term (first two columns), but Poisson in the short term (last two columns). First row: Traffic of the user of Figure 4.12 - event-count per unit time ((a), (c)) and respective cumulative event-count ((b), (d)). Second row: synthetic traffic, by the SFP model (with matching μ, ρ and event-count). Observe that (1) both time series are visually similar; (2) both are bursty in the long run (spikes; inactivity) (3) both are Poisson-like in the short term (last two columns)	67
5.1	Average accuracy of the network-based models and of the Yesterday model, that is currently the state of the art. The accuracy is measured using the Spearman's rank correlation coefficient ρ . While the NetFor has similar performance, the NetForY consistently beats the "Yesterday", with up to 14% improvement. . .	72
5.2	The majority of players marginally contributes to their teams in terms of box-score statistics in comparison with a few players who have significant contributions. Observe the complementary cumulative distribution of the players efficiencies and averages in points, assists and rebounds per season.	76
5.3	The majority of players have scored in their careers marginal values of box-score statistics in comparison with a few players who have significant contributions. Observe the probability density function (PDF) of the number of points, rebounds and assists the players achieved in their careers. The distributions follow a power law with a cutoff in the tail.	76
5.4	The efficiency of players acquired by a team has, in general, little effect on its future performance.	78

5.5	NBA network changes: the number of new nodes and transactions (new edges) over the years. Note that after 1976 there is much more mobility (transactions), with the number of new nodes being around 70 while the number of transactions keep on growing.	81
5.6	The <i>Team Volatility</i> hurts the performance of the teams, i.e., the more a team hires and fires players and coaches, the worst. Observe the regularly negative Spearman’s rank correlation ρ between the <i>Team Volatility</i> Δd and the performance of the teams. Note that, differently than box-score statistics, our network-based features can capture such correlation.	83
5.7	Again, note that the <i>Roster Aggregate Volatility</i> hurts the performance of the teams, i.e., the more a team hires players and coaches that are constantly changing their teams, the worst. The Spearman’s rank correlation ρ between the <i>Roster Aggregate Volatility</i> $\Sigma\Delta d$ and the performance of the teams is consistently negative. This is another observation that cannot be drawn from usual box-score statistics.	84
5.8	There is no clear correlation between the <i>Team Inexperience</i> and the performance of the teams. However, a closer look indicates that the correlation fluctuates smoothly, what suggests that the temporal scenario plays an important role in this feature.	86
5.9	A high coherence and rapport among players and the coach is a good indicator that a team will succeed. Note that the Spearman’s rank correlation ρ between the <i>Roster Aggregate Coherence</i> \overline{cc}_t^y and the performance of the teams is regularly positive.	87
5.10	The larger the roster size, the worst for the team. The Spearman’s rank correlation ρ between the <i>Roster Size</i> s_t^y and the performance of the teams is consistently negative.	87
5.11	The proposed network-based models had good performances in predicting the whole rank of the teams. Observe the Spearman’s rank correlation coefficient ρ between Π_t^y and r_t^y . The average correlation $\bar{\rho}$ for the NetForY is 0.68 and for the NetFor is 0.59, significative high values.	92
5.12	The network-based models consistently indicate a high performance team as their top team. They correctly predicted the top performance team $\approx 35\%$ of the times. Note that the higher the count values at the right at the distribution, the better.	93
5.13	The average ρ between Π_t^y and r_t^y for three consecutive years. The global average correlation $\bar{\rho}$ for the NetForY is 0.59 and for the NetFor and “Yesterday” model is 0.52	97

5.14	The difference $\Delta\rho = \rho_{M1} - \rho_{M2}$ between the ρ coefficients achieved by the two models $M1$ and $M2$	97
5.15	Performance of the network-based models and of the “Yesterday” model in identifying the best performance team.	98
6.1	A scenario where different WSNs are deployed in the same location. We see three networks, N_1, N_2 and N_3 , and a possible cooperation edge $e_3(j, f)$, where node f accepts data of network N_3 coming from sensor node j	109
6.2	Payoff matrix of the “WSN-Cooperation” game.	109
6.3	The Virtual Cooperation Bond: a feasible model to execute the GTC strategy ($w(e) = d^2$).	109
6.4	The algorithm describing the establishment of the VCB.	112
6.5	Cooperation results when the number of nodes is varied.	112
6.6	Cooperation results when path loss exponent is varied.	112
6.7	Cooperation results when the number of networks is varied.	114
6.8	Cooperation results when the networks have different configurations.	114
A.1	Changing the value of C changes the location of the distribution. The median of the distribution μ varies linearly with C , $\mu = a \times C + b$, with $a = 2.6$ and $b = 3.8$. The 95% confidence interval for a is (2.715, 2.723) and for b is (-8.60, 16.3). Since the confidence interval for b contains 0, b is not significant.	121
A.2	Changing the value of a changes the slope ρ of the distribution in a way that $\rho = a^{-1}$	122
B.1	The weights w_i of the network-based prediction models over time.	123
B.2	Distributions of the null model results over 100000 simulations.	124
B.3	The proposed network-based models are not significantly sensitive to W_Y . Observe that we could even had used values of W_Y that would give slightly higher results than the ones we used in the previous sections.	126
B.4	Relationship between the clustering coefficient and the age of teams and players. In this case, the age is the number of years a team or player have being active in the league. Note that there is a clear non-linear correlation between the clustering coefficient and the age of the nodes.	127

List of Tables

- 3.1 Classes of relationship. 25
- 3.2 Datasets used in this thesis. 35

- 4.1 Table of symbols. 39
- 4.2 Evolution of the meta-parameters (rows 1-5) and the *Focal Points*(rows 6-7) during the four months of our dataset. 49
- 4.3 Correlations between summarized attributes and ρ and β 52

- 5.1 Table of symbols. 82
- 5.2 Comparison among the Network Model and other prediction models. In **bold** the best result and in *italic* the runner up. Note that our proposed NetForY always achieved the best result. Moreover, the NetFor had a better performance than the “*Yesterday*” in the early years and in selecting the best performance team. 95

- 6.1 Table of symbols. 102

Contents

Acknowledgments	xi
Resumo	xv
Abstract	xvii
Resumo Estendido	xix
Motivação	xix
Objetivos	xx
Contribuições	xxi
Capítulo 3 – Redes de Mobilidade	xxi
Capítulo 4 – Redes de Comunicação	xxii
Capítulo 5 – Redes Competitivas	xxiv
Capítulo 6 – Redes de Sensores Sem Fio	xxiv
List of Figures	xxvii
List of Tables	xxxiii
1 Introduction	1
1.1 Motivation	1
1.2 Objective	2
1.3 Contributions	4
1.4 Work Organization	6
2 Related Work	7
2.1 DBCN Modeling	7
2.2 Predicting and Controlling	10
2.3 Fundamentals of Game Theory	12

3	Decision-Based Complex Networks	17
3.1	Motivation	17
3.2	Modeling	19
3.3	Classification	22
3.3.1	DBCNs features	22
3.3.2	The RECAST algorithm	24
3.3.3	Classification results	26
3.4	Prediction	29
3.5	Controlling	30
3.6	DBCNs Analyzed	34
4	Modeling in Communication Networks	37
4.1	Introduction	37
4.2	Communication Flow Size	38
4.2.1	Preliminaries	38
4.2.2	Call Duration Distribution	40
4.2.3	TLAC Over Time	46
4.2.4	Applications	50
4.3	Communication Dynamics	53
4.3.1	Contradicting Models	53
4.3.2	Marginal Distributions	54
4.3.3	Temporal Correlations	61
4.3.4	Proposed Model: “SFP”	63
4.3.5	Discussion	65
4.4	Final Remarks	68
5	Predicting in Competitive Networks	71
5.1	Introduction	71
5.2	Related Work	73
5.3	The NBA Network	74
5.3.1	Problem Definition	74
5.3.2	Motivation	75
5.3.3	The Network Definition	78
5.4	Proposed Network Features	80
5.4.1	Preliminaries	80
5.4.2	Team Volatility	82
5.4.3	Roster Aggregate Volatility	83

5.4.4	Team Inexperience	84
5.4.5	Roster Aggregate Coherence	85
5.4.6	Roster Size	86
5.5	Network-based Prediction Models	88
5.5.1	Problem Definition	88
5.5.2	NetFor	88
5.5.3	NetForY	90
5.6	Results and Validation	91
5.6.1	Results	91
5.6.2	Comparison	93
5.7	Generality	96
5.8	Final Remarks	98
6	Controlling Computer Networks	101
6.1	Introduction	101
6.2	Cooperation Among Different Networks	103
6.3	Related Work	104
6.4	WSN Cooperation Game	105
6.5	A Solution to the Game	107
6.6	Establishing the VCB	110
6.7	Numerical Results	111
6.8	Final Remarks	113
7	Conclusions and Future Work	115
A	Modeling in Communication Networks	119
A.1	The Log-logistic Distribution	119
A.2	The need for C in SFP	120
A.3	The Two Parameters of SFP	120
B	Predicting in Competitive Networks	123
B.1	Metrics Weights	123
B.2	Random Model Comparison	124
B.3	Parameter Sensitivity Analysis	125
B.4	Network Features to Node Attributes	125
	Bibliography	129

Chapter 1

Introduction

1.1 Motivation

The constant advancement of information systems allows, at a growing rate, more data to be generated and stored. The Internet, for example, consists of millions of computing devices in which each one of them is responsible for generating, storing and transmitting countless data. Clearly, the Internet is going in the direction of the Ubiquitous Computing paradigm, which, as envisioned in Mark Weiser's classic paper *The computer for the 21st century* [Weiser, 1999], predicts the access to computing environments by any person, anywhere, at any time, so that computing devices are coupled to the most trivial objects, such as clothing labels, cups of coffee, pens or any personal object [de Araujo, 2003]. Moreover, there are Wireless Sensor Networks (WSNs) [Akyildiz et al., 2002], which are a special type of ad hoc network, designed to collect data from the environment they are inserted and providing such information to the final user.

Thus, we can expect an increasing amount of data to be generated from the most diverse situations. Currently, for instance, there are research studies on data from phone call records [Du et al., 2009; Seshadri et al., 2008; Hidalgo and Rodriguez-Sickert, 2008], online social networks [Leskovec et al., 2007; Hill and Nagle, 2009; Kumar et al., 2006], railroads [Faloutsos et al., 1999], Internet websites [Faloutsos et al., 1999; Albert et al., 1999], citation networks [Guo et al., 2009], movies and actors [Jensen et al., 2008], sports leagues [Vaz de Melo et al., 2008a] and many others. From these studies we now know, for instance, how people link websites in their homepages and how communities of people evolve over time. It is fascinating that, behind the names and the numbers registered in all these data, we see the reflection of the environment itself, i.e., behind every record, there is a decision made by some entity. Therefore, the knowledge of how to process this invaluable evolving database can lead, consequently, to a better understanding of the interests and

dynamics of each entity in a determined system, community or in the society.

Crandall et al. [2008], for instance, analyzed a large dataset from Wikipedia and they verified that the communications between editors is related to the similarities they have, namely the probability of occurring a communication between two individuals increases as there are similarities between them. As another example, Backstrom et al. [2006a] analyzed a *LiveJournal* [LiveJournal, 2010] dataset, which is an online social network with more than 10 million users, and the DBLP [DBLP, 2010], which is a database of publications on some areas of computer science with over 400000 articles, in order to understand how user communities are formed and evolve over time. They concluded, among other things, that the tendency of an individual entering a community is influenced not only by the number of friends he/she has within the community, but mainly by how those friends are connected among them. The fact is that the data generated from social interactions has a tremendous potential for uncovering and modeling the human behavior.

In this thesis, we focus on systems that are made up of entities capable of interacting among themselves in an autonomous way, reflecting their interests and activity dynamics. We call these systems *Decision-based Complex Networks (DBCNs)*, these entities, nodes or agents, and the interactions between nodes, edges. A DBCN is a special type of complex network Newman [2003] that contains nodes capable of making autonomous decisions, which are guided mainly by their personal motivations. For instance, social networks [Backstrom et al., 2006b; Kumar et al., 2006; Leskovec et al., 2008; Mislove et al., 2007] formed from friendship ties or work collaborations are DBCNs, since the nodes of these networks have decision power to create edges. On the other hand, semantic networks formed from texts [Steyvers and Tenenbaum, 2005] and earthquake networks [Abe and Suzuki, 2004] are not DBCNs, since the edges creation is guided, in these cases, by a central process.

As general complex networks, DBCNs are characterized by having a large number of vertices and edges that exhibit a pattern, such as communities or highly connected vertices, called hubs [Albert et al., 1999]. While in a simple network with at most hundreds of nodes the human eye is a tool of considerable power, in a complex network, this approach is useless. Thus, to study, analyze and characterize complex networks, fast statistical methods and algorithms are necessary.

1.2 Objective

The main objective of this thesis is to analyze the behavior of agents of DBCNs. We observe real and hypothetical scenarios where environmental and social decisions play an important

role on the evolution of the network. When we fully understand the motivations behind the actions of the agents, we are able to devise techniques to model, predict and control the behavior of diverse and large DBCNs. As we show in Figure 1.1, we divide the main objective of this thesis in:

1. **Modeling.** Our goal is to design models that can accurately represent the behavior of the agents in a DBCN, significantly contributing to a better understanding of the system;
2. **Predicting.** From the understanding of the reasons behind the decisions of the agents, we aim to predict how the system will evolve;
3. **Controlling.** Once we know how the system will behave, we design control mechanisms to make the agents to act according to a determined goal.

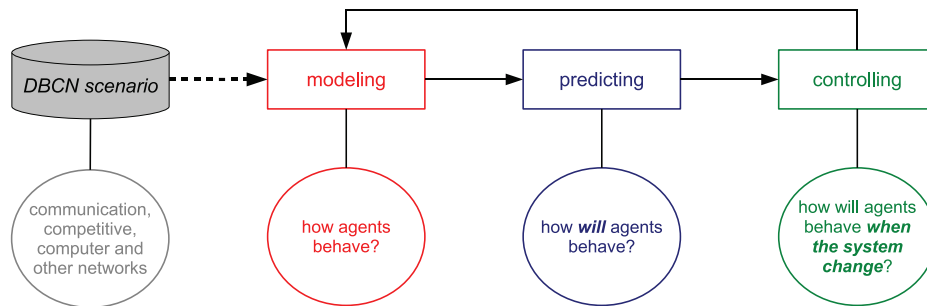


Figure 1.1. The summary of this thesis. From DBCNs scenarios, first we model the behavior of the agents and, from this, we are able to predict their behavior. Once we know how the agents will behave, we are able to design control mechanisms to make the agents to act according to a determined goal and, therefore, change the system. New modeling and predicting efforts should be made in the new system to better control its behavior.

In summary, from the local decisions made by the agents, we plan to understand the global evolution of the system. This immediately opens space for a vast and diverse number of applications that we also tackle in this work. In this thesis, we study three significantly different types of network systems and propose different applications for all of them, based on the knowledge of the behavior of their agents. Thus, all the analyzed networks have in common the fact that agents are capable of: (i) making autonomous decisions, and (ii) interacting with other agents, together with the fact that their evolution significantly depends on

how these decisions are made. The analyzed systems and the respective proposed applications for them are described below.

Communication Networks. First, we analyze communication networks, in which every agent is an individual capable of communicating freely to any other agent in the network. More specifically, we investigate more than 1 billion mobile phone records from a private mobile phone company of a large city, spanning ≈ 0.1 TB. We study and propose models for the individual behavior of the users of this network and, from this, we propose applications for data summarization, anomaly detection, user classification and network monitoring.

Competitive Networks. Second, we tackle competitive networks, in which the social agents of the system compete among themselves for a reward or limited resources. In this case, we analyze two sports leagues: the North American National Basketball Association (NBA) and the Major League Baseball (MLB). In these networks, the social agents are both the teams and the players, and they compete for titles and money, i.e., while the players want to have high salaries, the teams want to pay less for their players, at the same time that both types of agents want to win games and titles. For these networks, we propose a prediction model that can be used to identify likely teams to win and to fail in the following season.

Computer Networks. Finally, we consider a scenario in which the agents are computing devices with a limited set of decisions that aim to improve the operability of the system they are inserted in. More specifically, we investigate a scenario where a group of WSNs is deployed in the same region and they may interact among themselves to request or share computational resources. We model the problem of cooperation among two or more different WSNs by the concepts of game theory and, from it, we present, as a control application, a protocol to allow cooperation.

1.3 Contributions

In this thesis, we propose that the understanding of the motivations for the local decisions made by the agents of DBCNs lead to the understanding of how the network globally evolves. We analyzed three different types of networks from the perspective of agents and, in the following, we enumerate the contributions we propose from this analysis:

1. The proposal of the Truncated Lazy Contractor (TLAC) distribution to model the size of the communication flow between two autonomous agents. From the TLAC distribution, we introduce the *MetaDist*, which shows that the *collection* of TLAC parameters

follow a striking bivariate Gaussian; Finally, we show several applications that can be drawn from the TLAC and the *MetaDist*, such as anomaly detection and data summarization;

2. Proposal of the Self Feeding Process (SFP) model for the inter-event time between human activities, that matches real data well, and has a very simple, intuitive explanation: the time for the next communication event depends on the time it took for the previous event to occur. The SFP model generates a particular case of the TLAC distribution and unifies earlier communication traffic models;
3. We propose the use of network features to analyze the decisions made by the agents of competitive networks, more specifically teams, players and coaches in sports leagues. From this analysis, we construct parameter-free models to predict a team success in sports leagues. We show that complex network metrics provide good prediction information without using box score statistics;
4. Finally, we analyze the behavior of agents of computer networks when they are allowed to make local decisions. More specifically, we consider a scenario where different WSNs are deployed in the same region and we propose a protocol to allow cooperation among the networks.

The following publications are the current results for this thesis:

- In [Vaz de Melo et al., 2010], we analyze phone call durations of millions of users of a large mobile phone operator;
- In [Vaz de Melo et al., 2011a], we analyze the inter-event times of millions of users of a large mobile phone operator.
- In [Vaz de Melo et al., 2008a], we show that complex network metrics can be used to predict the behavior of basketball teams;
- In [Vaz de Melo et al., 2012], we propose a model to predict the behavior of sports teams;
- In [Vaz de Melo et al., 2008b] and [Vaz de Melo et al., 2008c], we analyze the problem of cooperation when different Wireless Sensor Networks are deployed in the same location;
- In [Vaz de Melo et al., 2009], we propose a distributed protocol for establishing the cooperation when different Wireless Sensor Networks are deployed in the same location;

- In [Vaz de Melo et al., 2011b], we show how game theory is being used in Wireless Sensor Networks;

1.4 Work Organization

This thesis is organized as follows.

First, in Chapter 2, we present the general related work according to three aspects we tackle in this thesis. A more specific related work for each aspect is presented in the corresponding chapter. Moreover, in Chapter 3, we introduce the Decision-based Complex Networks and show how simple efforts in modeling, predicting and controlling can lead to interesting results. Then, In Chapter 4, we analyze a large communication network and propose the TLAC distribution and the SFP generative model. In Chapter 5, we propose the use of network metrics to analyze the decisions made by agents of competitive networks. From this analysis, we construct a model to predict the behavior of the teams in sports leagues. In Chapter 6, we consider computer networks in which agents are allowed to make decisions. We propose a cooperation protocol for the scenario where different WSNs are deployed in the same location. Finally, in Chapter 7, we present the conclusion and future work.

Chapter 2

Related Work

In this thesis, our goal is to study the decisions made by the agents of DBCNs. We divide this study by analyzing separately three important aspects of these networks: modeling, predicting and controlling. Thus, in this chapter, we show how these three aspects are being tackled in the literature when focusing DBCNs. In Section 2.1, we show relevant models of real and modern DBCNs. In Section 2.2, we show the challenges and opportunities for predicting and controlling in DBCNs. Finally, in Section 2.3, we show the basics concepts of game theory, since it is a fundamental mathematical technique to study decisions of rational agents.

2.1 DBCN Modeling

In the literature, there are several generative models that construct networks from local decisions of the nodes [Mitzenmacher, 2004; Chakrabarti and Faloutsos, 2006; Zhou and Lipowsky, 2005]. These models were designed to reproduce real world networks that can not be generated by the traditional Erdős and Rényi [1960] random model. In nature, there are examples of complex networks that were modeled from the most diverse databases, such as phone call records [Du et al., 2009; Seshadri et al., 2008; Hidalgo and Rodriguez-Sickert, 2008], online social networks [Leskovec et al., 2007; Hill and Nagle, 2009; Kumar et al., 2006], railroads [Faloutsos et al., 1999], Internet websites [Faloutsos et al., 1999; Albert et al., 1999], citation networks [Guo et al., 2009], movies and actors [Jensen et al., 2008], sports leagues [Vaz de Melo et al., 2008b; Onody and de Castro, 2004] and many others.

Formally, each network $G(V, E)$ has a set of nodes (agents) V , a set of edges (connections) E , and a set of N observable characteristics c_1, c_2, \dots, c_N , that may or may not change over time. The goal of the generative models is, from a initial and simple network $G_0(V_0, E_0)$

and from a node local function f , to construct a network that has all the N observable characteristics c_1, c_2, \dots, c_N of the network. This local function f is designed to dictate which decisions each node should make in order to construct the network with the desired characteristics.

In random networks, such as the Erdős and Rényi networks, this f function is the most simple as possible: the probability of node i to connect to node j is $p \forall j \in V$. However, in real complex networks, this probability changes according to the local characteristics of the nodes i and j . For instance, it was verified that in online social networks, two nodes are more likely to connect as the number of friends they share increase [Kumar et al., 2006]. Moreover, it was also verified that as a website is more popular, i.e., it has many other websites pointing to it, it is also more likely to receive new links [Faloutsos et al., 1999]. Thus, in complex networks, the function f that governs the nodes' decisions is constructed by modeling their preferences, which can be the desire to connect to friends of their friends, or just to connect to the most popular nodes in the network.

One of the main observed characteristics of several complex networks that exist in nature is that they are scale-free networks (SFNs). SFNs [Newman, 2003; Li et al., 2005; Keller, 2005] were introduced by Barabasi and Albert [1999] to model network topologies such that the degree distribution follows a power law [Clauset et al., 2009]. The distribution of the random variable X defines that the degree distribution follows a power law if, given its cumulative distribution function $F(x) = P(X \leq x)$ and its complementary cumulative distribution function $\bar{F}(x) = P[X > x]$, $\bar{F}(x) = 1 - F(x) \approx cx^{-\alpha}$ for some constant $0 < c < \infty$ and tail index $\alpha > 0$ [Li et al., 2005]. For $1 < \alpha < 2$, F has infinite variance and finite mean but, for $0 < \alpha \leq 1$, both the variance and the mean are infinite. Besides this, one interesting property of this distribution is that $\log(P[X > x]) \approx \log(c) - \alpha \log(x)$, making the plot of \bar{F} in logarithmic scales to be a line with slope $-\alpha$ for high values of x . For more examples and details on how to identify power laws, please refer to [Clauset et al., 2009].

To the best of our knowledge, the first model that aimed to reproduce the scale-free characteristic of complex networks from the local decisions of the agents was the BA model [Barabasi and Albert, 1999]. This model, inspired in the *cumulative advantage* model proposed by Price [1976], incorporates two characteristics to explain the network formation: growth and preferential attachment. The first one stipulates that the number of nodes of the network grows with time, while the second one stipulates that the more connections a node has, the higher the chances for it to receive more edges. This phenomenon, also known as “the rich gets richer”, will result in a network with a few nodes with many connections, called *hubs*, and many nodes with a few.

The BA model considers that when an agent joins the network, it decides to connect to the nodes with more connections. In summary, the formation of the network according to

the BA model is given from a connected network with $n_0 > 2$ initial nodes v_1, \dots, v_{n_0} . Then, a new node v_{n_0+1} joins the network and connects to a node $v_j, 0 < j \leq n_0$ according to a probability that is proportional to the degree d_j of v_j . Formally, the probability $p_{i,j}$ of a node v_i to connect to an existing node v_j is:

$$p_{i,j} = \frac{d_j}{\sum_{u=1}^n d_u}.$$

Leskovec et al. [2008] extensively validated the BA model for four collections of real data: FLICKR, DELICIOUS, LinkedIn and YAHOO! ANSWERS. It was found that, even compared with more sophisticated variations, the BA model is the one that best explains the degree distribution of the databases analyzed. However, the BA model considers that the nodes have the global knowledge of the network, which is not realistic for most networks. To overcome this drawback, Leskovec et al. [2008] and Vázquez [2003] propose a local version of the BA model, where the process of creating the edges does not depend on the global knowledge of the network. These models stipulate that the nodes decide to connect to a node if the edge will close a triangle, considering, then, solely the local information of the node: its neighborhood.

While the scale-free characteristic could be successfully reproduced by the local function of generative models, there were still other relevant network characteristics that remained without any explanation. Therefore, other generative models were proposed in order to develop a unified theory on the local behavior of the nodes of the most popular complex networks.

Chen and Shi [2004] proposed two variations of the BA model considering relocation and deletion of edges of the network. Holme and Kim [2002] defined an additional step to the BA model, where each edge created by the BA model generates a second additional edge that closes a triangle. This model is interesting because it creates a network with characteristics of an SFN and a “small-world” network [Milgram, 1967; Albert et al., 1999; Amaral et al., 2000].

Moreover, Leskovec et al. [2007] proposed the *forest fire* model, which in addition to the characteristics previously described, is also capable of reproducing two new observations made from the temporal analysis of real networks: *the densification law* and *the shrinking diameter*. The first stipulates that the number of edges grows super-linearly with the number of nodes while the second one stipulates that the network diameter decreases with time. To generate these characteristics, the local node function states that, after connecting to a close node j in the network, the new node i should also connect to a set N_1 of neighbors j_1, j_2, \dots, j_n of j and recursively do this step connecting to a set N_k of neighbors of j_k .

Considering weighted networks, McGlohon et al. [2008] modified the *forest fire* model, proposing the *butterfly* model, that states that the nodes should also connect to nodes that are distant from its vicinity. This model is able to capture discovered characteristics of weighted graphs, such as the *fortification effect*, which states that the relationship between the number of edges of a graph and their total weight is superlinear, following a power law. Still in weighted graphs, Du et al. [2009] proposed the *Pay and Call* game to model the behavior of users in a network formed from mobile users of a telecom operator. It was considered that each user is a rational agent that will make calls if the benefit from a call is higher than its cost. With this model, the authors could reproduce some of the characteristics we described so far, such as scale-free property and also new characteristics, such as the “clique participation law” that states that the number of maximal cliques in the network follows a power law.

2.2 Predicting and Controlling

The theoretical models that were proposed for characterizing the evolution of social networks improved significantly our knowledge on these systems. Therefore, as our knowledge on the behavior of social networks grows, the problem of predicting which links will appear in the future becomes easier or, at least, less complicated. The fact is that the methods for forecasting the occurrence of future edges are based on the knowledge embedded in these models, as we see in this section.

Liben-Nowell and Kleinberg [2007] defined the link prediction problem as a basic computational problem underlying social network evolution as: Given a snapshot of a social network at time t , how can we accurately predict the edges that will be added to the network during the interval from time t to a given future time t' . Basically the link prediction problem asks to what extent can the evolution of a social network be modeled using features intrinsic to the network itself. As we see in this definition given by Liben-Nowell and Kleinberg [2007], the link prediction problem is directly related to problem of modeling in DBCNs.

Moreover, the problem of link prediction play an important role in understanding the evolution of DBCNs and is relevant to a large number of applications. For instance, a company can analyze the interactions within the informal social network among its members and then suggest promising interactions or collaborations that have not yet been utilized within the organization [Kautz et al., 1997; Raghavan, 2002]. Moreover, a social network analysis could be made monitoring terrorist networks in order to spot individuals who are working together even though their interaction has not been directly observed [Krebs, 2002]. Finally, the link prediction problem is also related to the problem of inferring miss-

ing links from an observed network, i.e., links that, while not directly visible, are likely to exist [Goldberg and Roth, 2003; Taskar et al., 2003].

There are several simple methods that are used in the literature to do link prediction. The methods assign a score $s(i, j)$ to pairs of nodes (i, j) and then produce a ranked list with the edges which are more likely to appear in the future. Perhaps the most basic approach to rank the probable future edges is by using the distance between the pairs of nodes (i, j) , following the notion that social networks are “small worlds” [Watts and Strogatz, 1998]. Another simple method consists in ranking the pairs of nodes according to the number of neighbors they have in common, i.e., the probability of an edge to appear between nodes (i, j) increases with the number of neighbors they share [Newman, 2001]. This method uses the concept that similar nodes are more likely to connect in the future. Thus, instead of using the number of common neighbors, one can use other similarity metrics to do link prediction. For instance, Crandall et al. [2008] showed that Wikipedia users who share common interests are more likely to connect in the future. Finally, non-similar nodes can also have a high probability of connecting, as stated by the preferential attachment method [Barabasi and Albert, 1999], in a way that highly connected nodes tend to attract links from new low connected nodes. For more methods please see [Liben-Nowell and Kleinberg, 2007].

We saw that the problem of link prediction aims to predict which connections will occur in the future based on the attributes of the nodes. Instead of waiting for the links to appear, a system administrator could, for instance, use this knowledge to directly suggest to the nodes that they could connect. This simple idea is essentially the basis for the design of all recommendation systems [Adomavicius and Tuzhilin, 2005; Burke, 2002], that is one of the main control mechanisms of social networks.

In 2006 and in the following years, a lot of attention was given to the design of efficient recommendation systems. This happened because Netflix, an online DVD-rental service, announced a 1 million dollar prize for those who achieve a 10% or more improvement on the performance of their own recommendation system [Bennett et al., 2007]. After 3 years, the prize was given to the BellKor’s Pragmatic Chaos team, solving the problem by merging several algorithms [Toscher et al., 2009].

Recommendation systems are usually classified into three categories, based on how recommendations are made [Balabanovic and Shoham, 1997]. In the Content-based recommendations (i), the user is recommended items similar to the ones the user preferred in the past. In the Collaborative recommendations (ii), the user is recommended items that similar users to him/her liked in the past. Finally, Hybrid approaches (iii) combine collaborative and content-based methods. Again, all these methods use the attributes of the network’s nodes and edges in order to offer a future connection to the nodes and, therefore, partially controlling the network’s evolution. For more details on recommendation systems the surveys

of Adomavicius and Tuzhilin [2005] and Burke [2002].

2.3 Fundamentals of Game Theory

One of the most used techniques to analyze the decisions of rational agents is game theory [Fudenberg and Tirole, 1991; Nisan et al., 2007; Luce and Raiffa, 1957]. Game theory is the formal study of conflict and cooperation among agents when their action is independent [Turocy and von Stengel, 2001]. The major advantage of using game theory is that it provides the methodology for analyzing and structuring strategic decision problems. The concepts of game theory are widely used in the decision analysis of agents such as companies vying for market share or nations in the war. In the last decades, game theory came to be used also to study the behavior of computational agents, or entities which cannot act differently than the way they were programmed to act. In this case, game theory deals mainly with the optimization of a global result from local decisions in distributed systems. This problem is the generalization of diverse problems found in computer networks and, amongst them, congestion control [Alpcan and Basar, 2002], load balancing [Suri et al., 2007], routing [Roughgarden and Tardos, 2002], peering [Corbo and Petermann, 2004] and P2P system design [Buragohain et al., 2003] have solutions in the literature modeled from the game theoretic concepts.

Modeling a problem from the game theory is done by defining a game, the players of this game and their allowed actions, reactions and preferences. One of the most classic games of game theory is “The Battle of the Sexes” [Fudenberg and Tirole, 1991]. This game has two players, one man and one woman. The man and woman want to go out together but each one wants to go to a different place. The man prefers to go to a football match, while the woman prefers to go to the movies. If they are together in the football match, the man will have a greater satisfaction than woman. On the other hand, if they go to the movies together, she will have a greater satisfaction. If each one goes alone to his favorite program, both will be dissatisfied. The action of each player is to select the place he/she will go. However, their satisfaction will depend on the action chosen by both.

Measuring the satisfaction of a player is done numerically by a real number that is generated from a function that takes into account the action of every other player participating in the game. This numerical value that each player gets to determine his/her satisfaction is called the *payoff* and the function that generates this number is called the *utility function*. Formally, a game has a finite set of players $G = g_1, g_2, \dots, g_n$ and each player $g_i \in G$ has a finite set of strategies $S_i = s_{i1}, s_{i2}, \dots, s_{im_i}$ called the *pure strategies* of player g_i ($m_i \geq 2$). The notation s_{-i} is used to represent all the pure strategy set of the players, except player g_i .

A vector $s = (s_{1j_1}, s_{2j_2}, \dots, s_{nj_n})$, where s_{ij_i} is a pure strategy of player $g_i \in G$ is called a pure strategy profile. The set of all pure strategy profiles generates the Cartesian product

$$\prod_{i=1}^n S_i = S_1 \times S_2 \times \dots \times S_n,$$

called the *pure strategy space* of the game. For each player $g_i \in G$, there is an utility function $u_i : S \rightarrow \mathfrak{R}$, $s \rightarrow u_i(s)$, which associates the payoff $u_i(s)$ of player g_i to each pure strategy profile $s \in S$.

From these definitions, it is possible to model the “The Battle of the Sexes” game in the following way:

$$\begin{aligned} G &= \{man, woman\}, \\ S_{man} &= \{football, movies\}, \\ S_{woman} &= \{football, movies\}, \\ S &= \{(football, football), (football, movies), \\ &\quad (movies, football), (movies, movies)\}. \end{aligned}$$

The two utility functions $u_{man} : S \rightarrow \mathfrak{R}$ e $u_{woman} : S \rightarrow \mathfrak{R}$ should return real numbers for each player and, in this case, we can define arbitrary values such as

$$\begin{aligned} u_{woman}(football, football) &= 5; \\ u_{woman}(football, movies) &= 0; \\ u_{woman}(movies, football) &= 0; \\ u_{woman}(movies, movies) &= 10; \\ u_{man}(football, football) &= 10; \\ u_{man}(football, movies) &= 0; \\ u_{man}(movies, football) &= 0; \\ u_{man}(movies, movies) &= 5. \end{aligned}$$

The results described above can be represented by a matrix named *payoff matrix*, which can be seen in Figure 2.1. Each quadrant of the matrix contains a pair of values that represents a possible outcome of the game. The first value is the payoff given to the “row player”, the

man. The second value is the payoff given to “column player”, the woman. The rows and columns of the table contains labels that represent the possible strategies of the players. The woman chooses the strategies defined by the columns of the table and the man chooses the strategies defined by the rows of the table.

	Football	Movies
Football	10, 5	0, 0
Movies	0, 0	5, 10

Figure 2.1. Payoff matrix of “The Battle of the Sexes” game.

The solution of a game is a prediction about its outcome [Brigida Sartini, 2004]. The most common way to find the solution of a game is to find its Nash equilibrium, if this exists. A Nash equilibrium is characterized when no player is able to increase its payoff changing its strategy when all your opponents keep theirs [Fudenberg and Tirole, 1991]. Thus, a strategy profile

$$s^* = (s_1^*, \dots, s_{i-1}^*, s_i^*, s_{i+1}^*, \dots, s_n^*) \in S$$

is a Nash equilibrium if

$$u_i(s_i^*, s_{-i}^*) \geq u_i(s_{ij_i}, s_{-i}^*)$$

for all $i = 1, \dots, n$ and all $j_i = 1, \dots, m_i$, where $m_i \geq 2$.

When players of a game reaches a Nash equilibrium, the game to which they belong has reached a completely stable state, i.e., no player will change their strategies. It is important to note that the steady state is ensured taking into account the selfish and rational character of the players. There are several ways to calculate strategies of players that will lead to a Nash equilibrium and they vary according to the characteristics of the game [Fudenberg and Tirole, 1991]. It is also important to note that there are games which have no Nash equilibrium for pure strategies as the game of “even or odd”, and others that have multiple Nash equilibria, as the “The Battle of the Sexes” game, with profiles (*football, soccer*) and (*movies, movies*) being solutions that are in Nash equilibrium.

An alternative to games that do not have Nash equilibrium strategies is to consider a probabilistic point of view. In the “even or odd”, for example, we can consider that the players have 50% chance of choosing even and 50% chance of choosing odd. A probability distribution on the set S_i of pure strategies of the player is called *mixed strategy profile* p_i for the player $g_i \in G$. Non-cooperative games always have a Nash equilibrium for mixed strategies [Nash, 1950, 1951]. For details and descriptions of algorithms to find a Nash

equilibrium, see the work of Porter et al. [2004].

Chapter 3

Decision-Based Complex Networks

3.1 Motivation

The main characteristic of DBCNs is that the interactions between their entities are, usually, a consequence of semi-rational decisions. We say “usually” and “semi-rational” decisions because any system is subject to random events and irrational choices due to, for instance, the “trembling hand” effect [Fudenberg and Tirole, 1991]. In summary, DBCNs can be characterized according to two main factors: (i) *the interactions among their entities are based on semi-rational decision that tend to be regular and to repeat themselves*; (ii) *communities are naturally formed in the network reflecting the social decisions of its entities*. Because of that, DBCNs pose as a very interesting and promising scenario for the three well connected aspects we approach in this thesis: (i) modeling, (ii) predicting and (iii) controlling.

Social vs. random interactions: Because most of the interactions in DBCNs still arise from conscious decisions made by their entities, the evolution of DBCNs is significantly different from the evolution of random networks, e.g., Erdős and Rényi networks Erdős and Rényi [1960]. Thus, while in DBCNs the edges are created from semi-rational decisions, which tend to be regular and to repeat themselves, in a *random network the edges are created independently of the attributes of the nodes, i.e., the probability of connecting two nodes is constant*.

For instance, considering a network composed of people, their routine in everyday life correlates to their interactions: if Smith and Johnson work at the same office, they are likely to meet at 9am during weekdays. This is because we strongly believe Smith and Johnson **decide** to go to work every day on time, since this is, probably, the rational decision to perform. However, their decision can be affected by random events, such as being stuck in traffic on a turnpike. Although rational decisions are regular, random decisions may often occur as well.

Formally, an agent may execute a *social decision*, or a *random decision*. Intuitively, if its probability of performing a social decision is greater than its probability of a random one, the network evolves to a well-structured social network. On the opposite, the network evolves as a random network, such as the Erdős and Rényi networks.

Differentiating social from random network: Among the main features observed in DBCNs, there is the presence of communities, which are groups of individuals who are strongly connected to each other because they share the same interests or activity dynamics Backstrom et al. [2006b]; Kumar et al. [2006]. On the other hand, in a random network, edges are created independently of the attributes of each node, i.e., a node i has the same probability p to connect to any other node j of the network. This fact is fundamental to differentiate a social network from a random network.

The network clustering coefficient Newman [2003] efficiently discriminates random from social networks. Given an undirected graph $G(V, E)$, the clustering coefficient cc_i measures the probability of two given neighbors of a node i to be directly connected, being calculated as $cc_i = \frac{2|(j,k)|}{d_i(d_i-1)} : j, k \in N_i, (j, k) \in E$, where N_i is the set of neighbors and d_i is the degree of node i . The clustering coefficient \bar{cc} of the network is the average $cc_i, \forall i \in V$.

By introducing the equivalent random network G^R as the random network constructed with the same number of nodes, edges and empirical degree distribution of its real world counterpart G , Watts and Strogatz Watts and Strogatz [1998] show that the clustering coefficient of a social network G is one order of magnitude higher than the clustering coefficient of G^R . Thus, when a given network G exhibits a clustering coefficient that is significantly higher (i.e., orders of magnitude higher) than that of its random equivalent G^R , then we can state that (part of) the decisions made by the agent of G are non-random.

Real-world DBCN datasets: In this chapter we look at three real-world datasets (or traces) that describe movements of entities in campus and city scenarios of DBCNs. The Dartmouth College dataset Henderson et al. [2004] is a mobility trace of more than 1,000 individuals in the university campus, recorded over eight weeks. The USC campus dataset Jen Hsu and Helmy [2005] is also a mobility trace in a campus scenario, with more than 4,000 individuals over eight weeks. Finally, the San Francisco dataset Rojas et al. [2005] contains records of the mobility of 551 taxis in San Francisco, over one month. In all cases, the contact events between two individuals are traced using start date of contact and its duration. For both Dartmouth and USC traces, two individuals are in contact if they are using the same access point to connect to the wireless network on campus. In the San Francisco trace, two individuals are in contact if their distance is lower than 250 meters, which is the maximum range of the IEEE 802.11 network Piorkowski et al. [2009]. More details of the datasets we use in this chapter and in the rest of this thesis are described in Table 3.6.

In the rest of this chapter we show how a simple effort on modeling, predicting and

controlling DBCNs can already generate very interesting and promising results. First, from the modelling effort we are able to better understand the behavior of the agents. Then, from the knowledge obtained in this stage, we predict their behavior. Finally, since we know how the agents will behave, we show how a control application can benefit from this knowledge.

3.2 Modeling

The previously introduced datasets list *encounter events* given by the time and duration of any two nodes meeting. We divide the whole duration of the dataset into discrete time steps of duration δ . In our study, we considered a duration of $\delta = 1$ day, since the data sets originate from human activities who mostly have a common routine repeated day by day (e.g., go to work/school or have lunch in the same place). All encounter events occurring between time steps $k - 1$ and k are represented in an *event graph* $\mathcal{G}_k(\mathcal{V}_k, \mathcal{E}_k)$. The set of vertices \mathcal{V}_k is composed of all nodes enlisted in the encounter events between time steps $k - 1$ and k . An edge between vertex i and j exists in \mathcal{E}_k if nodes i and j have met between time steps $k - 1$ and k .

From that, we can define a time varying representation of the DBCN using a temporal accumulative graph $G_t = (V_t, E_t)$. Formally, $G_t = \{\mathcal{G}_1 \cup \mathcal{G}_2 \cup \dots \cup \mathcal{G}_t\}$. As such, V_t (resp. E_t) is the set of all vertices (respectively edges) that appeared in the dataset between time 0 and time step t . G_t evolves over time and considers both the routine encounters and the random encounters between two individuals.

In Figure 3.1, we show the accumulation graph G_t calculated for $t = 2$ weeks of data, drawn using a force-directed layout algorithm (FDLA) Fruchterman and Reingold [1991]. Although the FDLA draws the graph so there are as few crossing edges as possible, notice how difficult it is to compare or extract any knowledge from the networks using only the preview.

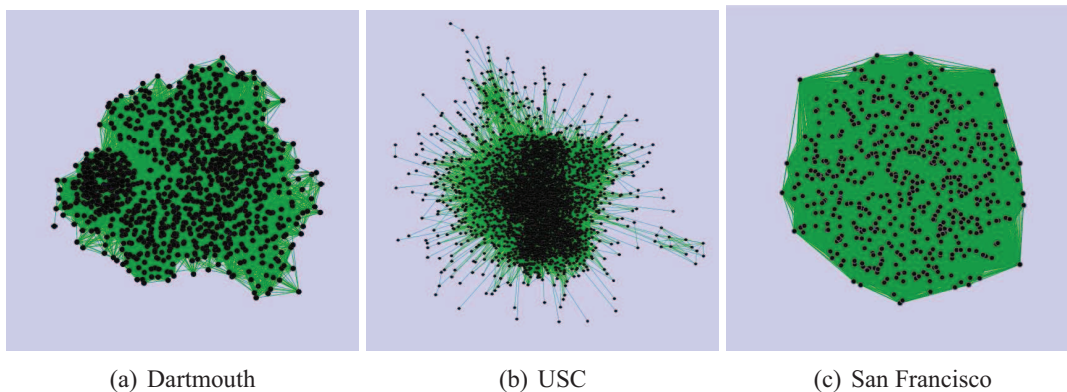


Figure 3.1. Snapshots of the networks G_t after two weeks.

In Figure 3.2, we show the densification of G_t . First, in Figure 3.2-a, we show the number of new edges added by \mathcal{G}_k to $\mathcal{G}_1 \cup \mathcal{G}_2 \cup \dots \cup \mathcal{G}_{k-1}$ per each day k . Observe the ups and downs in the curves, which are clearly a consequence of the seasonality, i.e., weekdays have more activity than weekends. Also, observe that the number of new edges added to the USC network is, in general, orders of magnitude higher than the other networks. However, when we normalize this number of new edges by the number of possible edges $\frac{|V_t| \times |V_t - 1|}{2}$, we see that the San Francisco dataset is the one with the higher densification, completing almost 80% of the graph in the first day of the analysis, see Figure 3.2-b. Note that the USC and the Dartmouth networks densify similarly.

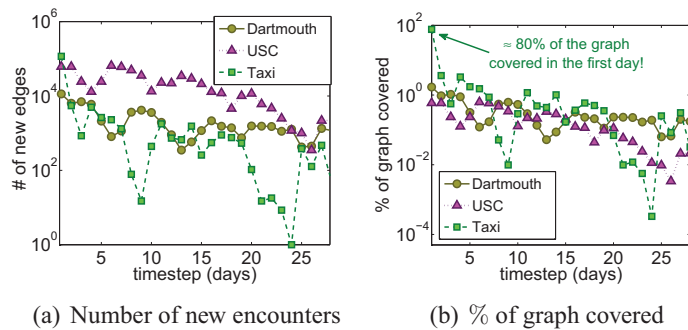


Figure 3.2. The densification of the temporal graph $G_t(V_t, E_t)$ per each day t . (a) The number of new edges added to $G_t(V_t, E_t)$ per each day t . (b) The percentage of the graph that is covered by the new edges.

The first step to analyze the mobility patterns of graph G_t is to build its random version G_t^R . This version should contain similar topological characteristics of the real graph, i.e., the same number of nodes, edges, and empirical degree distribution. Thus, the only difference between G_t and G_t^R is in the connections among nodes. While in G_t the nodes connect in a “semi-rational” way, in G_t^R the connections happen in a purely random fashion. This allows to accurately determine the extent of randomness in the mobility of individuals in temporal social networks.

In this thesis, we use two algorithms to compare real data with random graphs. The first algorithm, which we will call RND, is well known in the scientific community Chung and Lu [2002]. The algorithm $G^R = \text{RND}(G)$ receives a graph $G(V, E)$ as a parameter and returns a random graph $G^R(V, E^R)$ with the same topological characteristics of G . Given the degree distribution $D = (d_1, d_2, \dots, d_n)$ of G with n nodes, this algorithm assigns an edge between nodes i and j with probability $p_{i,j} = (d_i \times d_j) / \sum_{k=1}^{|V|} d_k$.

The second random generator algorithm used in this chapter, which we call T-RND, is an extension of RND and is able to generate random graphs from a temporal network G_t . The

temporal graph $G_t = \{\mathcal{G}_1 \cup \mathcal{G}_2 \cup \dots \cup \mathcal{G}_t\}$ is the union of event graphs \mathcal{G}_t . Thus, the algorithm $G_t^R = \text{T-RND}(\mathcal{G}_1, \mathcal{G}_2, \dots, \mathcal{G}_t)$ receives as parameters a set of consecutive event graphs \mathcal{G}_t and returns a random temporal graph G_t^R . It constructs G_t^R by executing RND in each event graph \mathcal{G}_t and then aggregating it in a way that $G_t^R = \{\text{RND}(\mathcal{G}_1) \cup \text{RND}(\mathcal{G}_2) \cup \dots \cup \text{RND}(\mathcal{G}_t)\}$.

We first demonstrate the analytical power of random comparison in Figure 3.3, where we show the behavior of the clustering coefficient for graphs G_t and G_t^R over time for the three analyzed networks. As we have previously mentioned, the clustering coefficient is a good metric to differentiate social networks from random networks. As we observe in Figure 3.3-a, for the Dartmouth dataset, in the first days, the clustering coefficient of G_t and G_t^R are different in orders of magnitude. However, as time goes by, their values get closer, as more random encounters are occurring. On the other hand, as we see in Figure 3.3-b, the clustering coefficients of G_t and G_t^R for the USC dataset are almost constant over time. However, the difference between them is not significantly high, since they have the same order of magnitude. We discuss these differences in detail later on.

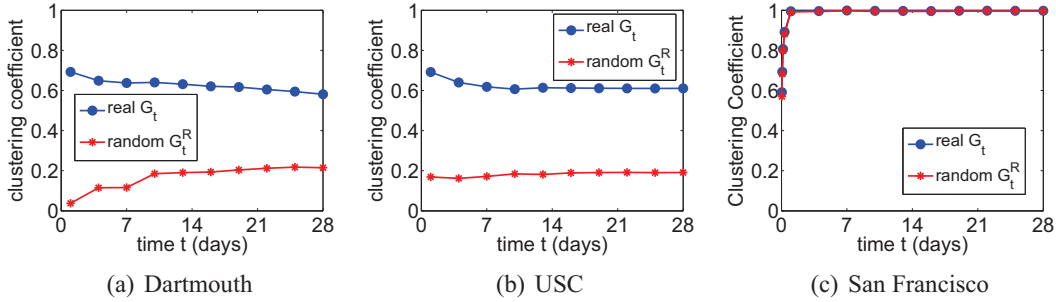


Figure 3.3. Evolution of the clustering coefficient of the three analyzed networks G_t and their random counterparts G_t^R .

Finally, as we observe in Figure 3.3-c, the clustering coefficients of the San Francisco networks are practically the same, being close to 1. In fact, after a few hours, the network becomes similar to a clique, indicating a global high mobility, allowing each individual taxi to encounter most of the other taxis at some point of the day. Formally, this indicates that G_t and G_t^R are very similar for the San Francisco dataset, i.e., the probability of random decisions is much more higher than the probability of social decisions, what makes the San Francisco network similar to a random mobile network. This makes sense since taxis' decisions depend on the decisions of their passengers they are collecting at random on the streets.

3.3 Classification

From the modeling, we propose the *Random rELationship CLASsifier sTrategy (RECAST)* to differentiate relationships among individuals in a social network. More precisely, the purpose of RECAST is to tell apart, in a DBCN, random interactions, which are hard to predict, from social-driven ones, which are easier. To that end, Section 3.3.1 presents the DBCNfeatures used by RECAST. Then, Section 3.3.2 introduces the RECAST algorithm and, finally, Section 3.3.3 discusses the results obtained by applying RECAST to the previously introduced datasets.

3.3.1 DBCNs features

In order to identify social relationships, we must point out which features distinguish a social relationship from a random one. Indeed, two characteristics are always present in social relationships:

1. **Regularity.** It is well known that social relationships are regular, in that they repeat over time Eagle et al. [2009]. If two individuals are friends, e.g., co-workers or daily commuters, they see each other regularly.
2. **Similarity.** It is expected that two individuals who share a social relationship have common acquaintances between them Onnela et al. [2007]. As an example, two individuals who share a large number of friends will most probably know each other as well.

Regularity and **Similarity** can be mapped into DBCNfeatures that, in turn, can be computed from a contact dataset so as to identify what kind of relationship two individuals share. In the following, we discuss such features.

3.3.1.1 Edge Persistence

A complex network metric mapping of the **Regularity** of a relationship is the edge persistence Hidalgo and Rodriguez-Sickert [2008]. Basically, considering the set of event graphs $\{\mathcal{G}_1, \dots, \mathcal{G}_t\}$, the edge persistence $per_t(i, j)$ measures the percentage of times the edge (i, j) occurred over the past discrete time steps $1, 2, \dots, t$. Formally, it is defined as $per_t(i, j) = \frac{1}{t} \sum_{k=1}^t \mathbb{1}_{[(i,j) \in \mathcal{E}_k]}$, where $\mathbb{1}_{[(i,j) \in \mathcal{E}_k]}$ is an indicator function that assumes value 1 if the edge (i, j) exists in \mathcal{E}_k at time k , and 0 otherwise. Note that here the persistence computation is performed over the set of event graphs \mathcal{G}_t and not over the aggregated temporal graphs G_t . For instance, assuming that each day of the week is a time step, if Smith

and Johnson met each other two times in a given week, their edge persistence is the number of times they encountered, i.e., 2, divided by the total number of time steps, i.e., 7, $per_{t=7}(\text{Smith}, \text{Johnson}) = 2/7$. Edge persistence allows thus to spot regular and routine relationships between two individuals. We again emphasize that in the aggregated graph G_t , only one edge exists between Smith and Johnson after this week.

We show in Figure 3.4 (first row) the edge persistence as measured in the three datasets. Considering the set of event graphs $\{\mathcal{G}_1, \dots, \mathcal{G}_t\}$ of all three real network and their T-RND-generated random counterparts $\{\text{RND}(\mathcal{G}_1), \dots, \text{RND}(\mathcal{G}_t)\}$, we portray the complementary cumulative distribution function (CCDF) $\bar{F}_{per(i,j)}(x) = P[per_t(i,j) > x]$ (second row). There, the time step is one day and each curve is obtained by analyzing four weeks of contacts, or, in other words, $t = 28$ corresponds to the length of the shorter considered dataset, i.e., the San Francisco dataset. From Figures 3.4-a and 3.4-b, both the Dartmouth and USC networks have edge persistence distributions that significantly differ from their random equivalent. More precisely, while the CCDFs of random networks show an exponential decay, in the real networks, individuals tend to see each other regularly, i.e., for reasons beyond pure randomness. On the other hand, as we observe in Figure 3.4-c, the encounters in the San Francisco dataset occur almost in a random fashion.

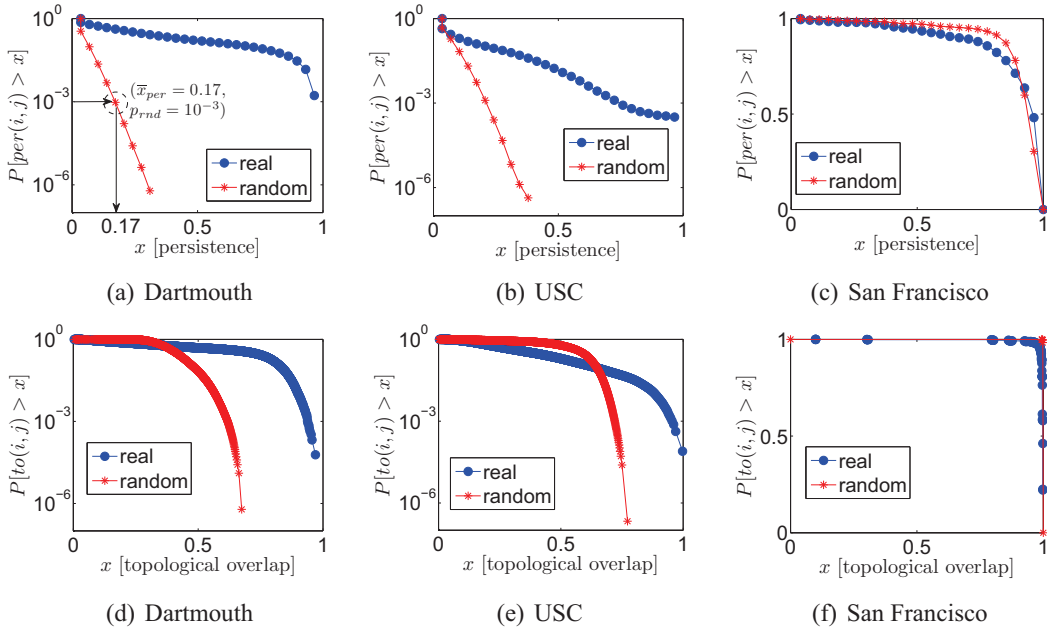


Figure 3.4. The complementary cumulative distribution function of the edge persistence (a,b,c) and topological overlap (d,e,f) for the three analyzed networks G_t and their random counterparts G_t^R after four weeks, $t = 28$ days.

3.3.1.2 Topological Overlap

The **Similarity** of contacts can be mapped to the topological overlap feature of a complex network Onnela et al. [2007]. This metric is extracted from the aggregated temporal graph G_t . The topological overlap $to_t(i, j)$ of a pair of nodes i and j is defined as the ratio of neighbors shared by two nodes, or, formally

$$to_t(i, j) = \frac{|\{k \mid (i, k), (j, k) \in E_t\}|}{|\{k \mid (i, k) \in E_t\} \cup \{k \mid (j, k) \in E_t\} \setminus \{k \mid (i, k), (j, k) \in E_t\}|}$$

In Figure 3.4 (second row), we show the CCDF $\bar{F}_{to(i,j)}(x) = P[to_t(i, j) > x]$ of the topological overlap of the edges of the real networks G_t and their respective random networks G_t^R , generated by the T-RND mechanism. Again, we pick one day as the time step and consider four weeks of contacts (i.e., $t = 28$ days). Similar to what occurred to the edge persistence, we note that for the Dartmouth and USC networks CCDFs significantly differ from their random counterparts, in Figures 3.4-d and 3.4-e. Indeed, pairs of individuals in these datasets share common neighbors in a way that could not happen randomly. Conversely, in Figure 3.4-f, the San Francisco network again behaves like a random contact network. Since all results indicate that the San Francisco network is random by nature, in the remainder of this chapter we will focus on the Dartmouth and the USC datasets.

3.3.2 The RECAST algorithm

We have seen that both the edge persistence and the topological overlap behave significantly differently in social networks and in plain random networks. Therefore, given a contact dataset, it is possible to exploit such a diversity to identify which edges are consequences of random or social events. In particular, we propose four classes of relationships or edges, which depend on the possibility that a determined edge feature (i.e. persistence or topological overlap) is generated randomly. The four classes of relationships are described in Table 3.1. A feature value is called “social” if there is an almost zero probability of this value be generated randomly. On the other hand, a feature value is called “random” if there is a significant probability of this value be generated randomly.

Table 3.1. Classes of relationship.

Class	Edge persistence	Topological overlap
<i>Friendship</i>	social	social
<i>Acquaintanceship</i>	random	social
<i>Bridges</i>	social	random
<i>Random</i>	random	random

Relationships classified as *Friendship* characterize pairs of individuals who meet each other regularly and also tend to know the same people. The *Acquaintanceship* class includes relationships among individuals sharing a lot of acquaintances, but not meeting often. Considering a friendship network, friends of friends who see each other once in a while, in occasions such as birthday parties, graduation ceremonies or weddings, would be classified as *Acquaintanceship*. The last social class is that of *Bridges* that characterizes pairs of individuals who see each other regularly, but do not share a large number of common friends. E.g., the so-called familiar strangers, people who meet every day but do not really know each other (i.e., because they just commute between common home and work areas) are very likely to be classified as *Bridges*. Finally, when an edge is neither persistent nor characterized by topological overlap, it is considered the result of a random contact, and we classify it as *Random*.

In order to distinguish “social” from “random” values of the DBCN’s features, we resort to the distributions we previously discussed. More precisely, we define a value p_{rnd} , the only parameter in RECAST, and we identify the feature value \bar{x} for which $\bar{F}(x) = p_{rnd}$ for the random network G_t^R . The value \bar{x} represents then a threshold, such that feature values higher than \bar{x} occur with a probability lower than p_{rnd} in a random network. If we set p_{rnd} to some small value, we can finally state that feature values higher than \bar{x} are very unlikely to occur in a random network, i.e., they are most probably due to actual social relationships. The parameter p_{rnd} can also be seen as the expected classification error percentage. For instance, in Figure 3.4-a consider $p_{rnd} = 10^{-3}$. This gives a threshold $\bar{x} = 0.17$, then all values higher than $\bar{x} = 0.17$ will be classified as social. However, there is a $p_{rnd} = 10^{-3}$ probability of randomly generating values that the RECAST classified as social edges (i.e., false positives). In other words, we expect that $10^{-3} = 0.01\%$ of the edges classified as social to be, in fact, random. The full RECAST mechanism is described in Algorithm 1, where the criteria used in each classification are detailed. In this algorithm, index t is omitted for *per* and *to* metrics for clarity purposes.

The complexity of RECAST is upper bounded by the construction of G_t^R using T-RND, which is $O(t \times (|\mathcal{V}_t| + |\mathcal{E}_t^R|))$, i.e., the minimum complexity for the generation of a degree

Algorithm 1: RECAST: classify edges of G_t

```

Require:  $p_{rnd} \geq 0$ 
return  $\text{class}(i, j) \quad \forall (i, j) \in \cup_t E_t$ 
Construct  $G_t^R$  and set  $\{\text{RND}(\mathcal{G}_1), \dots, \text{RND}(\mathcal{G}_t)\}$  using T-RND
Get  $\bar{F}_{to}(x)$  from  $G_t^R$  and  $\bar{F}_{per}(x)$  from  $\{\text{RND}(\mathcal{G}_1), \dots, \text{RND}(\mathcal{G}_t)\}$ 
Get  $\bar{x}_{to} \mid \bar{F}_{to}(\bar{x}_{to}) = p_{rnd}$  and  $\bar{x}_{per} \mid \bar{F}_{per}(\bar{x}_{per}) = p_{rnd}$ 
for all edges  $(i, j) \in E_t$  do
  if  $per(i, j) > \bar{x}_{per}$  and  $to(i, j) > \bar{x}_{to}$  then
     $\text{class}(i, j) \leftarrow \text{Friendship}$ 
  else if  $per(i, j) > \bar{x}_{per}$  and  $to(i, j) \leq \bar{x}_{to}$  then
     $\text{class}(i, j) \leftarrow \text{Bridges}$ 
  else if  $per(i, j) \leq \bar{x}_{per}$  and  $to(i, j) > \bar{x}_{to}$  then
     $\text{class}(i, j) \leftarrow \text{Acquaintanceship}$ 
  else
     $\text{class}(i, j) \leftarrow \text{Random}$ 
  end if
end for

```

sequence-based random graph available to date Chung and Lu [2002]. After the construction of G_t^R , the complexity of the classification mechanism is $O(|E_t^R| \times |V_t|)$, where $O(|V_t|)$ is the cost of computing the topological overlap of an edge.

3.3.3 Classification results

We apply RECAST to the Dartmouth and the USC networks. We are omitting the results for the San Francisco dataset, since, as previously stated, the random-like nature of taxi routes makes the analysis uninteresting, with all edges classified as *Random*. In Figure 3.5, we show the number of edges per class as a function of the p_{rnd} value. An initial and quite surprising observation is that, by varying p_{rnd} through four orders of magnitude, the number of edges per class stays in the same magnitude. This shows that RECAST is robust with respect to p_{rnd} , i.e., it does not need a fine calibration of the parameter to return a consistent edge classification.

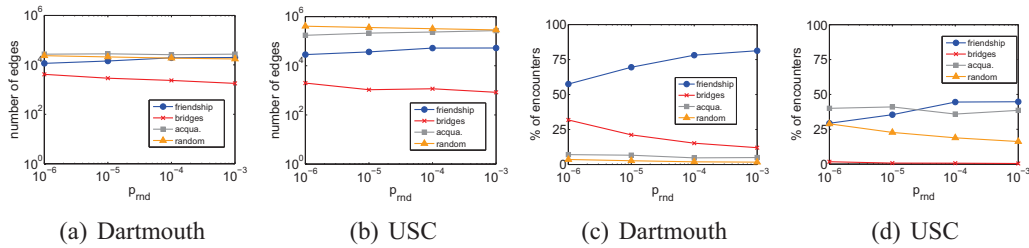


Figure 3.5. The number of edges ((a) and (b)) and percentage of encounters ((c) and (d)) of a given class that appears in the first four weeks of data for a given value of p_{rnd} .

Secondly, in both datasets, the number of *Bridges* is orders of magnitude lower than the other classes, a clear indication that in the analyzed social networks regular connections among different communities are rare. Also the number of *Friendship* edges is similar in the two networks, implying similar dynamics in tight relationships among individuals in the two campuses. This also agrees with the biological constraint on a social interaction that limits humans' social network size, i.e., the number of *Friendship* Dunbar [1998]. However, the two datasets differ when looking at the number of edges classified as *Acquaintanceship* and *Random*, which is one order of magnitude larger in USC than in Dartmouth. This is the result of the actual size of the two campuses, USC accounting for a population around ten times larger than that of Dartmouth. This aspect is also reflected by the size of the traces in Table 3.6 that clearly leads, in the USC network, to (i) many more *Random* contacts among individuals who do not actually know each other, but just happen to cross while strolling on campus, and (ii) an increased presence of strangers who happen to know the same people, leading to more *Acquaintanceship* edges.

As mentioned, the proportion of *Random* edges is similar but the number of individual encounters is significantly different between the Dartmouth and USC networks. In Figure 3.5, we show the percentage of encounters of a given class that appear in the first four weeks of data for a given value of p_{rnd} . Observe that the percentage of *Random* encounters in the Dartmouth network is close to zero, varying from 1.7% to 3.6% as p_{rnd} decreases. On the other hand, in the USC network, this percentage varies from 16% to 29%. In fact, the proportion of *Random* encounters provides a good estimate of the probability of random decisions in the analyzed networks. Thus, the USC network has a significantly higher tendency to evolve to a random topology than the Dartmouth network.

The analysis is confirmed by Figure 3.6¹, portraying the snapshots of the Dartmouth and USC networks after two weeks of interactions, when considering only social edges (i.e., *Friendship*, *Acquaintanceship* and *Bridges*) or *Random* edges. Edges of the former networks in Figures 3.6-a and 3.6-c are distinguished by colors, according to the same code used in Figure 3.5 (*Friendship* edges are painted in blue, *Bridges* in red, *Acquaintanceship* in gray, and *Random* in orange). The difference between these social-only networks and their complete counterpart (in Figure 3.1) or their random-only equivalents (in Figures 3.6-b and 3.6-d) is striking. Social networks are characterized by a complex structure of *Friendship* communities, linked to each other by *Bridges* and *Acquaintanceship*. More precisely, when comparing the Dartmouth and USC social networks, the former appears to be dominated by *Friendship* interactions, while the sheer number of *Acquaintanceship* in the latter drives its graph structure.

¹Please view Figure 3.6 in color.

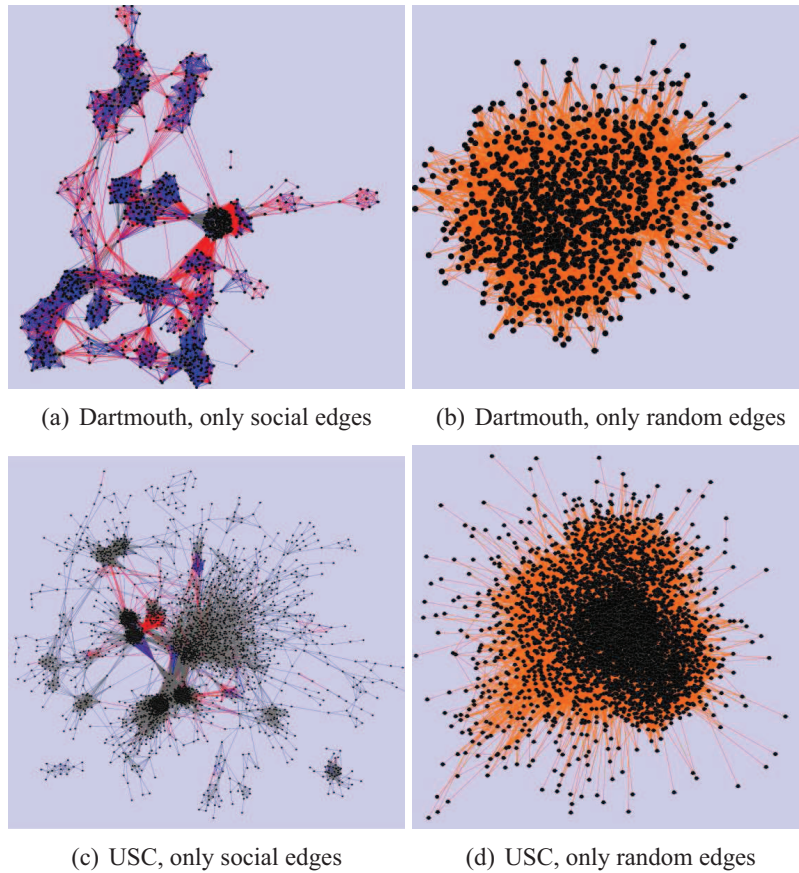


Figure 3.6. [Best viewed in color] Snapshots of the Dartmouth and USC networks after two weeks of interactions, considering only the social edges and only the random edges. *Friendship* edges are painted in blue, *Bridges* in red, *Acquaintanceship* in gray and *Random* in orange.

Conversely, networks containing only *Random* edges do not show any structure and look like random graphs. A rigorous way to verify the randomness of such networks, and thus validate the efficiency of the RECAST classification, is to perform a clustering coefficient analysis. Figure 3.7 compares the clustering coefficients of the Dartmouth and USC networks *when only Random edges are present*, i.e., comparison of graph G_t against the same metric computed in their random counterparts G_t^R . The clustering coefficient, commonly employed to determine the actual randomness of a network, has very similar evolutions in G_t and G_t^R , thus proving how RECAST is able to extract from a real-world contact dataset edges that correspond to purely random encounters.

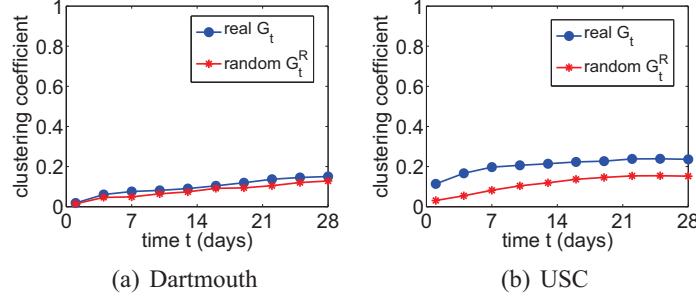


Figure 3.7. Evolution of the clustering coefficient of the Dartmouth and USC networks G_t when only *Random* edges are present compared to their random counterparts G_t^R .

3.4 Prediction

In this section, we show a simple prediction effort using the proposed RECAST. We classify the edges of G_t after four weeks of encounters (referred as *training set*, where encounters happened during 28 days are considered) and test the classification in the 5th week (referred as *test set*, where encounters happened during the 7 days of the 5th week are considered). Analysis of both sets is presented in Figure 3.8, as a function of p_{rnd} value.

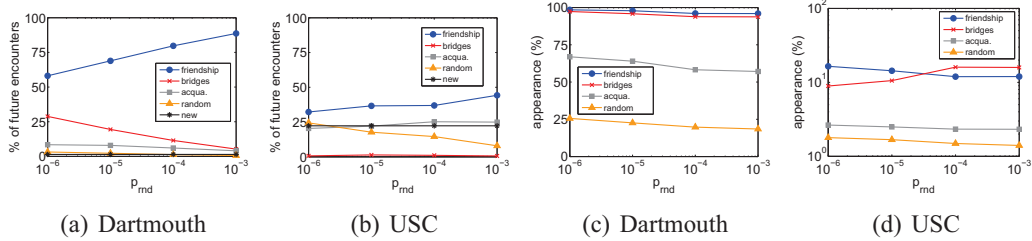


Figure 3.8. The distribution of encounters ((a) and (b)) and appearance ((c) and (d)) of each class in the one week *test set* as a function of p_{rnd} .

First, in Figure 3.8-(a) and -(b), we show the proportion of encounters of a determined class that appear in the 5th-week *test set* of each dataset. We also show the number of encounters involving edges that were not classified (referred as **new**), i.e., edges which appeared for the first time in the network in the *test set*. First, observe that the Dartmouth and USC networks behave differently. In the Dartmouth network, the almost-zero proportion of **new** encounters, allied to the fact that the proportion of encounters in the *test set* is similar to the proportions in the *training set* (see Figure 3.5-c), suggests that this network is stable. On the other hand, only the proportion of *Random* and *Bridges* encounters stays the same for the USC network compared with the *training set* (see Figure 3.5-d). The *Friendship* encounters slightly decrease, the *Acquaintanceship* encounters decrease $\approx 30\%$ and $\approx 25\%$ of the

encounters are **new**, suggesting that the USC network is not stable. These facts are, again, a consequence of the significant difference between the probabilities of random decisions of this network.

Second, in Figure 3.8-(c) and -(d), we show the percentage of edges of a given class that appeared in the *test set*. For instance, for $p_{rnd} = 10^{-6}$ in the Dartmouth network in Figure 3.8-(c), almost 100% of the edges classified in the *Friendship* class in *training set* are still friends in *test set*. Observe that edges which are put in a high persistence class (i.e., *Friendship* and *Bridges*) have more chance to appear than low persistence class (i.e., *Acquaintanceship* and *Random*), considering both networks. Moreover, considering the low persistence classes (i.e., *Acquaintanceship* and *Random*), an edge is more likely to appear if it has a high topological overlap (i.e., *Acquaintanceship*), being curious the fact that this is not the case for high persistence classes. The Dartmouth and USC networks present significantly different percentages because of the differences we explained in the last paragraph, i.e., while the Dartmouth network is stable (low probability of random encounters), the USC is not (high probability of random encounters).

3.5 Controlling

Given the knowledge obtained from the RECAST, we are able to analyze the impact of each class of relationship on data dissemination application using trace driven simulations. According to each dataset, we classify the edges after a *training set* of days using the RECAST with $p_{rnd} = 10^{-4}$ and verify the performance of the dissemination in a *test set* of days². For the Dartmouth network, we classify the edges after a *training set* of four weeks of encounters and verify the performance of the dissemination in a *test set* corresponding to the fifth subsequent week. For the USC network, since the number of encounters is lower than in the Dartmouth network, we extend the *test set* to three weeks. Moreover, since the number of new encounters is proportionally high, we extend the *training set* to six weeks.

The methodology we use in this chapter is the same as described in Zyba et al. [2011]. Messages are disseminated using flooding and, since the outcome depends on the start time of the simulation, we repeat the simulation by uniformly sampling many start times t_0 between the beginning of the selected week and the middle of that week. At the start of each simulation, only one individual *src* carries the message, which is the first node that appears in the trace after t_0 . Simulations last three and half days to ensure they all complete within the week-long trace.

²Other p_{rnd} values give similar results.

The message is transmitted via broadcast in a wireless fashion, i.e., everyone in contact with the sender receives the message. However, only *available* edges trigger transmissions. We consider this rule because we replay each trace multiple times making available different combinations of edge classes. For instance, if only *Friendship* edges are available, then only encounters of this class trigger transmissions, although everyone in the transmission range will receive the message, no matter which relationship they share with the sender. First, we analyze the impact on dissemination when all edges are available except the ones of a determined class. Then, we observe the behavior when only edges of a specific class are available. Finally, we show mixes of combining pairs of classes that give almost identical results as the scenario when all edges are available.

Moreover, as considered in Zyba et al. [2011], we also assume that message transfers are instantaneous. We also use contamination as the metric characterizing message dissemination. Contamination is the number of individuals who received a given message as a function of time. It reflects how effective a given population is at disseminating information in a given area. During the simulations, we consider that the population of individuals is solely the ones who appear in the trace within the simulation time. We also consider only encounters between pairs of individuals who have a class, i.e., those appearing in the training set. Thus, when all edges are considered and all individuals are reachable from *src* after t_0 , the contamination is 100%.

Figure 3.9 shows the contamination behavior when we remove from the traces all edges of a given class. For comparison, we also show the behavior of the contamination when all edges (except new encounters) are available. First, observe that for the USC network, only the removal of *Random* edges impacts significantly the performance of the contamination. On the other hand, for the Dartmouth network, the *Friendship* edge class is the one that impacts more significantly the contamination performance, but the removal of *Random* edges also has a significant impact. For instance, after five hours, the contamination is at approximately 58% when all edges are available. When we remove the *Random* edges, the contaminations drops to approximately 46%. However, the largest impact is caused by the removal of *Friendship* edges, dropping the contamination to approximately 36%.

Figures 3.10-a and 3.10-b show the contamination in the Dartmouth and the USC networks when only edges of a selected class are available. Observe that the individual role of each class is significantly different in the analyzed networks. The difference in the behavior of the contamination in the networks is mainly due to the proportion of random encounters in these networks. For instance, in the fifth week of encounters, the proportion of random encounters in the Dartmouth network is close to 0, while in the USC represents approximately

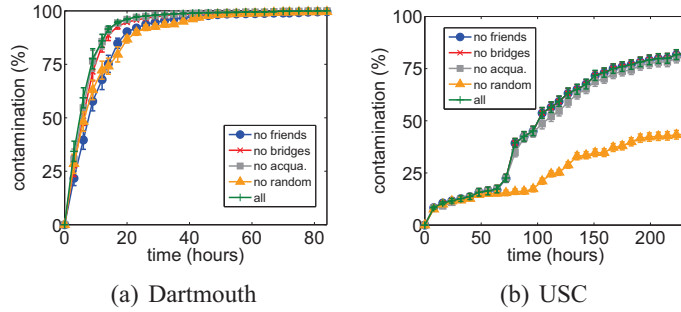


Figure 3.9. The contamination of the network when edges of a selected class are removed. The vertical lines represent a 95% confidence interval.

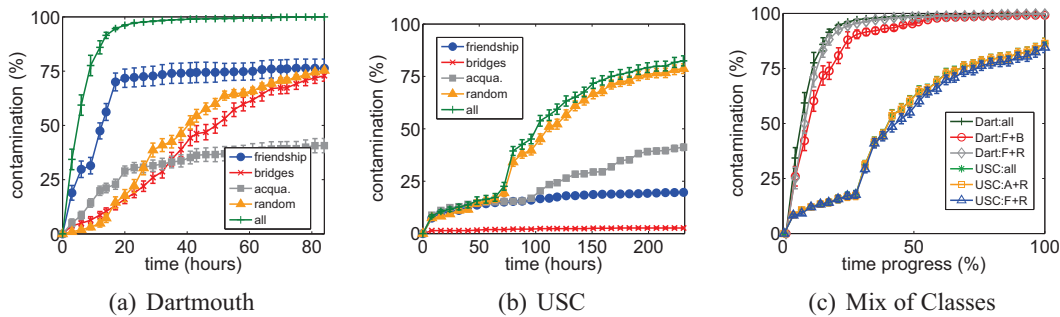


Figure 3.10. The contamination of the networks using only edges of a selected class. The vertical lines represent a 95% confidence interval.

20% of the total number of encounters (see Figure 3.8).

While in the Dartmouth network the *Friendship* edges are the ones that achieve the highest contamination, in the USC network the *Random* edges are the ones that perform better. In the Dartmouth network, the *Friendship* edges achieve a high contamination in the first hours, not spreading much more after that. This happens because *Friendship* characterize strong relationships inside communities, so the contamination spreads fast inside the community, but cannot find an easy way out of it to other groups of individuals. Since in the USC network the number of *Friendship* edges is significantly lower, the propagation does not spread at all through the community borders. Also note how the contamination using *Bridges* spreads slowly but always increasing in the Dartmouth network. This happens because the number of encounters involving *Bridge* edges is high, although the number of *Bridge* edges is low. Finally, it is interesting to observe that the contamination involving only *Random* has similar performance and the propagation curve in both networks is also similar for the given time windows, considering the fact that the total number of encounters.

It is also interesting the fact that for the Dartmouth network the shape of the curve when all edges are available is more similar to the shape of the curve when only *Friendship*

edges are available. On the other hand, for the USC network, the shape of the curve when all edges are available is more similar to the shape of the curve when only *Random* edges are available. This is an expected result since, as we have seen, in the Dartmouth network the individuals have a more social mobility behavior (low probability of random encounters), while in the USC network the individuals have a more random behavior (high probability of random encounters). Finally, it is worth noting that 100% of contamination is never reached in the USC network even when all edges are available. This is due to (i) the fact that we consider in the simulation time only encounters that appeared in the *training set*, (ii) the higher percentage of **new** edges present in the USC network (as seen in Figure 3.8, approximately 25% of the encounters in the fifth week of the test set are **new**) and (iii) 98% of the nodes present in the *test set* have **new** encounters.

In summary, after observing the contamination performance according to the use/removal of edges of a specific class, we can identify the most important classes of relationships in a given network. For instance, the *Acquaintanceship* class accounts for a large number of edges in both networks, but edges of this class seem to produce a low impact in data dissemination. On the other hand, we can use the edges classified as *Friendship* to disseminate information very fast inside a community and edges classified as *Random* to spread the information through different communities. This knowledge is fundamental in classes of applications that use, for instance, DTN routing schemes.

In Figure 3.10-c, we show how combinations of edge classes can give almost identical performance as if all edges were available for transmission. For the Dartmouth network, we observe the contamination when only *Friendship* and *Random* (**Dart:F+R**) are available and only *Friendship* and *Bridges* (**Dart:F+B**) are available. For the USC network, we observe the contamination when only *Acquaintanceship* and *Random* (**Dart:A+R**) are available. The main idea behind these combinations is that an efficient contamination through contact networks needs (i) edges that provide a high number of encounters inside communities, i.e. *Friendship* edges for Dartmouth and *Acquaintanceship* for USC, and (ii) edges that provide a high number of connections among individuals in different communities, i.e., *Random* and *Bridges* for Dartmouth and *Random* for USC. It also important to point out that the contamination when *Bridge* edges are added to *Friendship* edges in the Dartmouth network is similar to the case where *Random* edges are added to *Friendship*. However, the number of *Bridge* edges is only $\approx 12\%$ the number of *Random* edges. This shows the importance of detecting *Bridge* edges in the process of saving computational resources.

3.6 DBCNs Analyzed

In this chapter we showed a simple analysis on three mobility networks, contemplating the three main aspects we propose in this thesis, i.e., modeling, predicting and controlling, which are discussed separately. In Chapter 4, we focus on modeling large communication networks. In Chapter 5, we tackle the prediction task over competitive networks. Finally, in Chapter 6, we propose a protocol to control computer networks. Table 3.6 lists the real datasets we use to construct the DBCNs we analyze in this thesis.

Dataset	Reference	Description	Size	# of Agents	Edge
Dartmouth	[Henderson et al., 2004]	Mobility records in a campus	700MB	1156	Contact
USC	[Jen Hsu and Helmy, 2005]	Mobility records in a campus	160MB	4558	Contact
Taxi	[Rojas et al., 2005]	Mobility records in a city	161MB	551	Contact
Phone	[Vaz de Melo et al., 2010]	Phone call records	100GB	>3M	Call
SMS	[Vaz de Melo et al., 2011a]	SMS records	40GB	>3M	SMS
Enron	[Klimt and Yang, 2004]	E-mail records	1.3GB	158	E-mail
DIGG	[De Choudhury et al., 2009]	Phone call records	50MB	1,485	Comment
NBA	[databaseSports.com, 2007]	NBA temporal statistics	8 MB	4199	Labor relationship
MLB	[Lahman, 2008]	NBA temporal statistics	57 MB	17588	Labor relationship

Table 3.2. Datasets used in this thesis.

Chapter 4

Modeling in Communication Networks

4.1 Introduction

In this chapter, we show elaborate models for communication networks. In this network, every agent is an individual capable of communicating freely to any other agent in the network. Such networks can be used to model online social network records, e.g. as Orkut [Orkut, 2010], e-mail server data, phone calls, and Short Message Service (SMS) messages. In these networks, if a single agent i wants to communicate with another agent j , then a communication event is likely to happen between i and j , i.e., i calls or sends a message to j . Moreover, once a communication event occurs between i and j , a communication flow is established. The size $d_{i,j}$ of the communication flow grows while both of the agents still want to communicate. If one agent decides to stop the communication, the flow ends.

As we show in the extent of this chapter, modeling the agents of this type of network is extremely challenging. Human communication patterns are likely to change as the technological and cultural aspects of society change. For instance, the typical duration of a phone call involving two fixed phones is probably different from the one involving two mobile phones. Therefore, we use this challenging scenario in the study of modeling in DBCNs.

More specifically, in this chapter we analyze the rate in which two social agents communicate and also the duration of their communication flow. First, in Section 4.2, we analyze the size of the communication flows, i.e., the duration of phone calls, and we show how a good modeling effort can lead a wide variety of applications. Then, in Section 4.3, we analyze the inter-event time between communications in several communication scenarios. As we show in Section 4.3, there is an apparent contradiction in the existing literature about

which model should be used to represent inter-event times. Thus, in this part of this chapter we focus on proposing a intuitive and parsimonious model that unifies the previous theories. With the knowledge present in Section 4.2 and 4.3, we are able to describe the intrinsic motivations behind the social relationships among humans, which can be used for several applications. In Section 4.4, we present the conclusions and future directions from the work described in this chapter.

The main contributions of this chapter are:

- We propose the TLAC model for individual phone call durations and inter-event times;
- We describe applications based on TLAC model, such as the introduction of *MetaDist*, for monitoring the collective behavior of individuals;
- We propose the SFP model to generate TLAC distributions;
- We show that the communication events generated by the SFP model unifies existing theories on communication traffic.

Table 4.1 presents the symbols used in this chapter.

4.2 Communication Flow Size

To the best of our knowledge, this is the first work that analyzes the size of communication flows of individuals in such a large scale. The accurate knowledge on the size of the communication flow of individuals leads to several applications, such as user classification, network monitoring, infrastructure design and many others. We discuss this in details in Section 4.2.4.

4.2.1 Preliminaries

In the literature, Seshadri et al. [2008] proposed the Double Pareto Log-normal (DPLN) distribution to model the total time of talk minutes per customer, aggregating the phone calls of all users in the database. The DPLN distribution is a four parameter distribution that has two straight lines in log-log scales. Moreover, Guo et al. [2007] analyzed mobile phone calls that arrived in a mobile switch center in a GSM system of Qingdao, China, and they found that the duration of the phone calls is best modeled by a log-normal distribution. In these two studies, the call durations of different users are aggregated, what may generate noisy distributions with an undetermined observable pattern.

Symbol	Description
CDD	calls duration distribution
IED	inter-event distribution
n_i	number of phone calls made by user i
TLAC	Truncated Lazy Contractor model
OR	Odds Ratio
PDF	Probability Density Function
Δ_t	inter-event time between times t and $t - 1$
ρ	the OR slope
β	the OR intercept
μ	the median of the IED, $OR(\mu) = 1$
P, B, Σ	<i>MetaDist</i> parameters
C	location parameter of the <i>SFP0</i> and <i>SFP</i> models
a	<i>SFP</i> parameter that defines the value of ρ
$\bar{\rho}$	the average ρ of the users of a dataset
PP	Poisson Process
SFP0	Simplified Self-Feeding Process
SFP	Self-Feeding Process
SFP-N	Aggregated Self-Feeding Process
LRD	long range dependence
PL-P	Power Law – Poisson conundrum
LRD-P	Long Range Dependency – Poisson conundrum

Table 4.1. Table of symbols.

This was also observed by Willkomm et al. [2008], who studied the duration of mobile calls arriving at a base station during different periods and found that they are neither exponentially nor log-normally distributed, possessing significant deviations that make them hard to model. They verified that about 10% of calls have a duration of around 27 seconds, which correspond to calls not answered by mobile users and the calls were redirected to voicemail. This makes the call duration distribution to be significantly skewed towards smaller durations due to nontechnical failures, e.g., failure to answer. The authors showed that the distribution has a “semi-heavy” tail, with the variance being more than three times the mean, which is significantly higher than that of exponential distributions. Comparing to a log-normal distribution, though the tails agree better, but diverge at large values, what asks for a more heavy-tailed distribution.

4.2.2 Call Duration Distribution

4.2.2.1 Problem Definition

Given the preliminaries, we analyze mobile phone records of 3.1 million customers obtained from the network of a large mobile operator of a large city during four months. In this period, more than 1 billion phone calls were registered and, for each phone call, we have information about the duration of the phone call, the date and time it occurred and encrypted values that represent the source and the destination of the call, which may be mobile or not. When not stated otherwise, the results shown in this thesis refer to phone call records of the first month of our dataset. The results for the other three months are explicitly mentioned in Section 4.2.3. We used this methodology because we want to analyze the evolution of the user behavior over time, verifying how its behavior changes from month to month.

The Call Duration Distribution (CDD) is the distribution of the call duration per user in a period of time, which in our case, is one month. In the literature, there is no consensus about what well-known distribution should be used to model the CDD. There are researchers who claim that the CDD should be modeled by a log-normal distribution [Guo et al., 2007] and others that it should be modeled by the exponential distribution [Tejinder S. Randhawa, 2003]. Thus, in this section, we tackle the following problem:

Problem 1. *CDD FITTING.* Given d_1, d_2, \dots, d_{n_i} durations of n_i phone calls made by a user i in a month, find the most suitable distribution for them and report its parameters.

As we mentioned before, there is no consensus about what well known distribution should be used to model the CDD, i.e., for some cases the log-normal fits well [Guo et al., 2007] and for others, the exponential is the most appropriate distribution [Tejinder S. Randhawa, 2003]. Thus, finding another specific random distributions that could provide good fittings to a particular group of CDDs would just add another variable to Problem 1, without solving it. Therefore, we propose that the distribution that solves Problem 1 should necessarily obey the following requirements:

- **R1:** Intuitively explains the intrinsic reasons behind the call duration;
- **R2:** Provides good reliable fits for the great majority of the users.

In the following sections, we present a solution for Problem 1. In Section 4.2.2.2, we tackle Requirement *R1* by presenting the TLAC model, which is an intuitive model to represent CDDs. Then, in Section 4.2.2.3, we tackle Requirement *R2* by showing the goodness TLAC model fit for our dataset.

4.2.2.2 TLAC Model

Given these constraints, we start solving Problem 1 by explaining the evolution of the call duration by a survival analysis perspective. We consider that all calls c_1, c_2, \dots, c_C made by a user in a month are individuals who are alive while they are active. When a phone call c_j starts, its initial lifetime $l_j = 1$ and, as time goes by, l_j progressively increments until the call is over. It is obvious that the final lifetime of every c_j would be its duration d_j .

In the survival analysis literature, an interesting survival model that can intuitively explain the lifetime of entities such as human beings and companies in the market is the *log-logistic* distribution [Fisk, 1961; Bennett, 1983; Mahmood, 2000; Lawless and Lawless, 1982]. The log-logistic distribution models a modified version of the well-known “rich gets richer” phenomenon. First, for a variable to be “rich”, it has to face several risks of “dying” but, if it survives, it is more likely to get “richer” at every time. We propose that the same occurs for phone calls durations. After the initial risks of hanging up the call, e.g., wrong number calls, voice mail calls and short message calls such as “I am busy, talk to you later” or “I am here. Where are you?” type of calls, the call tends to get longer at every time. As an example, the lung cancer survival analysis case [Bennett, 1983] parallels our environment if we substitute endurance to disease with propensity to talk: a patient/customer who has stayed alive/talking so far, will remain as such, for more time, i.e., the longer is the duration of the call so far, the more the parties are enjoying the conversation and the more the call will survive. It is important to point out that the use of the log-logistic distribution is not limited to the survival analysis. There are also examples of its use in the distribution of wealth [Fisk, 1961], flood frequency analysis [M.I. Ahmad and Werritty, 1988] and software reliability [Gokhale and Trivedi, 1998].

Thus, to solve Problem 1, we propose the **Truncated Lazy Contractor (TLAC)** model, that is a truncated version of the log-logistic distribution, since it does not contain the interval $[0; 1)$. Firstly we show, in Figure 4.1-a, the Probability Density Function (PDF) of the TLAC, the log-normal and exponential distributions, in order to emphasize the main differences between these models. The parameters were chosen accordingly to a median call duration of two minutes for all distributions. The TLAC and log-normal distributions are very similar, but the TLAC is less concentrated in the median than the log-normal, i.e., it has power law increase ratios in its head and in its tail. We believe that this is another indication that the TLAC is suitable to model the users’ CDD, since as it was verified by [Willkomm et al., 2008], CDDs have “semi-heavy” tails. The basic formulas for the log-logistic distribution and, consequently, for the TLAC, are [Lawless and Lawless, 1982]:

$$PDF_{TLAC}(x) = \frac{\exp(z(1 + \sigma) - \mu)}{(\sigma(1 + e^z))^2}, \quad (4.1)$$

$$CDF_{TLAC}(x) = \frac{1}{1 + \exp(-\frac{(\ln(x) - \mu)}{\sigma})}, \quad (4.2)$$

$$z = (\ln(x) - \mu)/\sigma, \quad (4.3)$$

where μ is the location parameter and σ the shape parameter.

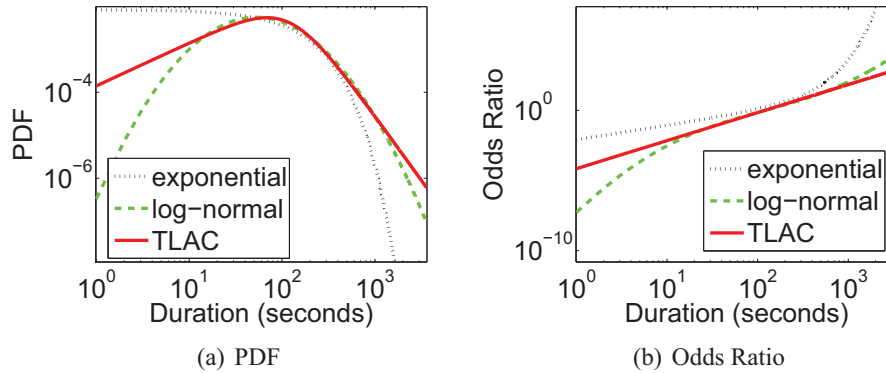


Figure 4.1. Comparison among the shapes of the log-normal, exponential and TLAC distributions.

Moreover, in finite sparse data that spans for several orders of magnitude, which is the case of CDDs when they are measured in seconds, it is very difficult to visualize the PDFs, since the distribution is considerably noisy at its tail. A possible option is to smooth the data by reducing its magnitude by aggregating data into buckets, with the cost of losing information. Another option is to move away from the PDF and analyze the cumulative distributions, i.e., cumulative density function (CDF) and complementary cumulative density function (CCDF) [Clauset et al., 2009]. These distributions veil the sparsity of the data and also the possible irregularities that may occur for any particular reason. However, by using the CDF (CCDF) you end up losing the information in the tail (head) of the distribution. In order to escape from these drawbacks, we propose the use of the Odds Ratio (OR) function, that is a cumulative function where we can clearly see the distribution behavior either in the head or in the tail. This $OR(t)$ function is commonly used in the survival analysis and it measures the ratio between the number of individuals that have not survived by time t and the ones that survived. Its formula is given by:

$$OR(t) = \frac{CDF_{TLAC}(t)}{1 - CDF_{TLAC}(t)}. \quad (4.4)$$

Thus, in Figure 4.2-b, we plot the OR function for the TLAC, the log-normal and exponential distributions. The OR function of the exponential distribution is a power law until t reaches the median, and then it grows exponentially. On the other hand, the OR function of the log-normal grows slowly in the head and then rapidly in the tail. Finally, the OR function for the TLAC is the most interesting one. When plotted in log-log scales, is a straight line, i.e., it is a power law. Thus, as shown in [Bennett, 1983], the $OR(t)$ function can be summarized by the following linear regression model:

$$\ln(OR(t)) = \rho \ln(t) + \beta, \quad (4.5)$$

$$OR(t) = e^{\beta} t^{\rho}. \quad (4.6)$$

In our context, Equation 4.5 means that the ratio between the number of calls that will die by time t and the ones that will survive grows with a power of ρ . Moreover, given that the median \hat{t} of the CDD is given when $OR(t) = 1$ and $OR(t) < 1$ when $t < \hat{t}$, the probability of a call to end grows with t when $t < \hat{t}$ and then decreases forever. We call this phenomenon the “lazy contractor” effect, which represents the time a lazy contractor takes to complete a job. If the job is easier and does require less effort than the ordinary regular job, the contractor finishes it rapidly. However, for jobs that are harder and demand more work than the ordinary regular job, the contractor also gets lazier and takes even more time to complete it, i.e., the longer a job is taking to be completed, the longer it will take. The parameter ρ and β are the parameters of the TLAC model, with $\rho = 1/\sigma$.

We conclude this section and, therefore, the first part of the solution to Problem 1, by explaining the intuition behind the parameters of the TLAC model. The parameter ρ is the *efficiency* coefficient, which measures how efficient is the contractor. The higher the ρ , the more efficient is the contractor and the faster the job is completed. On the other hand, the location parameter β is the *weakness* coefficient, which gives the duration \hat{t} of the typical regular job a contractor with a determined efficiency coefficient ρ can take without being lazy, where $\hat{t} = \exp(-\beta/\rho)$. This means that the lower the β , the harder the jobs that the contractor is used to handle.

4.2.2.3 Goodness of Fit

In this section, we tackle the second requirement of Problem 1 by showing the goodness of fit of our TLAC model. First, we show in Figure 4.2-a, the PDF of the CDD for a high talkative user, with 3091 calls, and with the values put in buckets of 5 seconds to ease the visualization. We also show the best fittings using Maximum Likelihood Estimation (MLE) for the exponential and the log-normal distributions and also for our proposed TLAC model.

Visually, it is clear that the best fittings are the ones from the log-normal distribution and the TLAC distribution, with the exponential distribution not being able to explain neither the head and the tail of the CDD.

However, by examining the OR plot in Figure 4.2-b, we clearly see the TLAC model provides the best fitting for the real data. As verified for the exponential distribution in the PDF, in the OR case, the log-normal could also not explain either the head or the tail of the CDD. We also point out that we can see relevant differences between the TLAC model and the real data only for the first call durations, which happen because these regions represent only a very small fraction of the data. The results showed in Figure 4.2 validate again our proposal that the TLAC is a good model for CDDs.

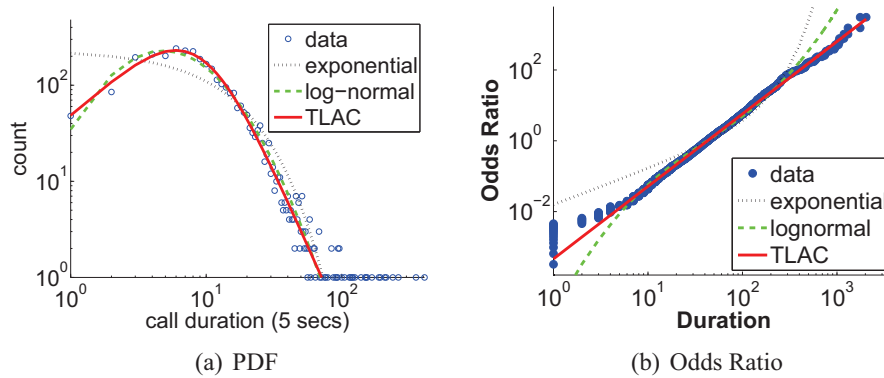


Figure 4.2. Comparison of models for the distribution of the phone calls duration of a high talkative user, with 3091 calls. TLAC in red, log-normal in green and exponential in black. Visually, for the PDF both the TLAC and the log-normal distribution provide good fits to the CDD but, for the OR, the TLAC clearly provide the best fit.

Given our initial analysis, we may state that the TLAC seems to be a good fit for the CDDs and also serve as an intuitively explanation for how the durations of the calls are generated. However, in order to conclude our answer for Problem 1, we must verify its generality power and also compare it to the log-normal and exponential generality power as well. Thus, we verify which one of the distributions can better fit the CDD for all the users of our dataset that have $n > 30$ phone calls. We calculated, for every user, the best fit according to the MLE for the TLAC, the log-normal and exponential distributions and we performed a Kolmogorov-Smirnov goodness of fit test [Massey, 1951], with 5% of significance level, to verify if the user's CDD is either one of these distributions. For now on, every time we mention that a distribution was correctly fit, we are implying that we successfully performed a Kolmogorov-Smirnov goodness of fit test.

In Figure 4.3, we show the percentage of CDDs that could be fit by a log-normal, a TLAC and an exponential distribution. As we can see, the TLAC distribution can explain the highest fraction of CDDs and the exponential distribution, the lowest. We observe that the TLAC distribution correctly fit almost 100% of the CDDs for users with $n < 1000$. From this point on, the quality of the fittings starts to decay, but significantly later than the log-normal distribution. We emphasize that the great majority of users have $n < 1000$, what indicates that some of these talkative users' CDD are probably driven by non natural activities, such as spams, telemarketing or other strong comercial-driven intents. This result, allied to the fact that the TLAC distribution could model more than 96% of the users, make it reasonable to answer Problem 1 claiming that the TLAC distribution is the standard model for CDDs in our dataset.

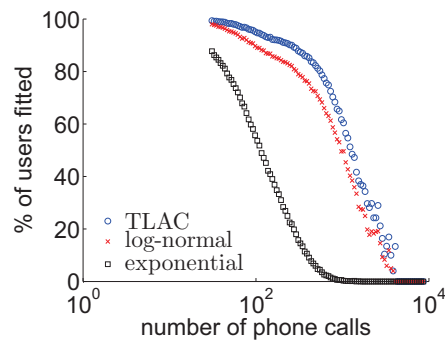


Figure 4.3. Percentage of users' CDDs who were correctly fit vs. the user's number of calls c . The TLAC distribution is the one that provided better fittings for the whole population of customers with $c > 30$. It correctly fit more than 96% of the users, only significantly failing to fit users with $c > 10^3$, probably spammers, telemarketers or other non-normal behavior user.

Finally, we further explore Problem 1 by looking at the OR of the talkative users who were not correctly fit by the TLAC model. These particular users have a significantly higher number of phone call durations with particular values, such as 1 minute, 2 minutes, 10 minutes, among others. This is probably happening due to a digitalization error, or policy, while recording their data, making some of the durations from 30 seconds to 1 minute being recorded as 1 minute durations, for instance. In Figure 4.4, we show the OR for three of these users. These digitalized values can be spotted on the small waves which are seen through the line. Despite of this, we observe that even these customers have a visually good fitting to the TLAC model. These results corroborate even more with the generality power of TLAC. Despite the fact that the irregularities of these customers' CDDs unable them to be correctly fit by the TLAC model, it is clear that the TLAC can represent their CDDs

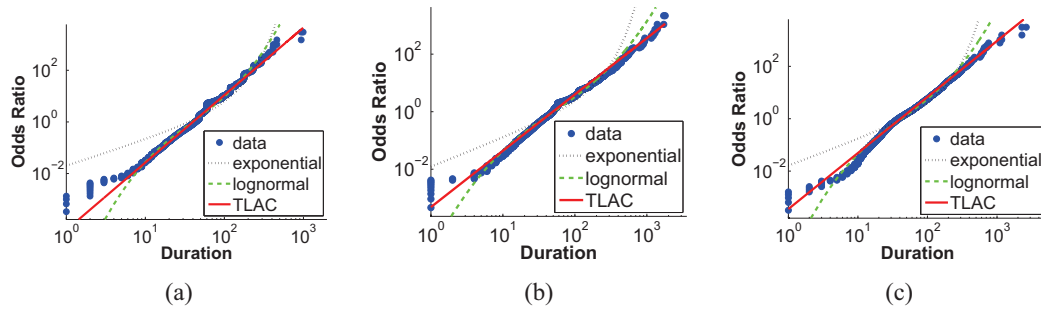


Figure 4.4. Odds ratio of three talkative customers who were not correctly fit by the TLAC model.

significantly well.

4.2.3 TLAC Over Time

We know it is trivial to visualize the distribution of users with a determined summarized attribute, such as number of phone calls per month or aggregate call duration. However, if we want to visualize the distribution and evolution of a temporal feature of the user such as his/her CDD, this starts to get more complicated. Thus, in this section, we tackle the following problem:

Problem 2. EVOLUTION. Given the ρ_i and β_i parameters of N customers ($i = 1, 2, \dots, N$), describe how they collectively evolve over time.

We propose two approaches to solve Problem 2. In Section 4.2.3.1 we describe the *MetaDist* solution and, in Section 4.2.3.2, we describe the *Focal Point* approach.

4.2.3.1 Group Behavior and Meta-Fitting

Since we know that the great majority of users' CDD can be modeled by the TLAC model, in order to solve Problem 2, we need to figure out how each user i is distributed according to their parameters ρ_i and β_i of the TLAC model. If the meta-distribution of the parameters ρ_i and β_i is well defined, then we can model the collective call behavior of the users and see its evolution over time. From now on, we will call the meta-distribution of the parameters ρ_i and β_i the *MetaDist* distribution.

In Figure 4.5-a, we show the scatter plot of the parameters ρ_i and β_i of the CDD of each user i for the first month of our dataset. We can not observe any latent pattern due to the overplotting but, however, we can spot outliers. Moreover, by plotting the ρ_i and β_i parameters using isocontours, as shown in Figure 4.5-b, we automatically smooth the visualization by disregarding lowly populated regions. While darker colors mean a higher

concentration of pairs ρ_i and β_i , white color mean that there are no users with CDDs with these values of ρ_i and β_i .

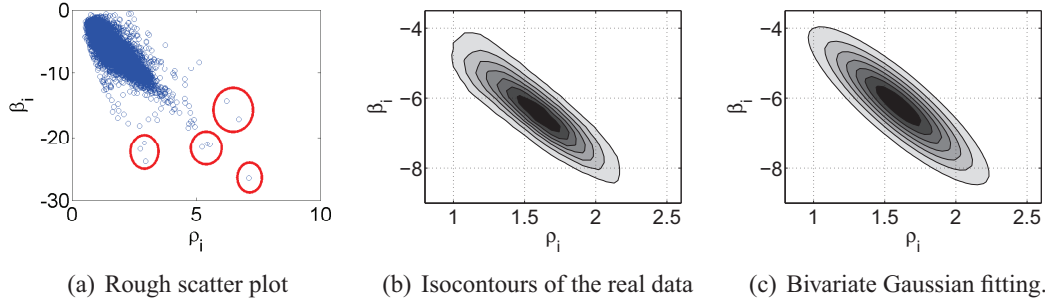


Figure 4.5. Scatter plot of the parameters ρ_i and β_i of the CDD of each user i for the first month of our dataset. In (a) we can not see any particular pattern, but we can spot outliers. By plotting the isocontours (b), we can observe how well a bivariate Gaussian (c) fits the real distribution of the ρ_i and β_i of the CDDs ('meta-fitting')

Surprisingly, we observe that the isocontours of Figure 4.5-b are very similar to the ones of a bivariate Gaussian. In order to verify this, we extracted from the *MetaDist* distribution the means P and B of the parameters ρ_i and β_i , respectively, and also the covariance matrix Σ . We use these values to generate the isocontours of a bivariate Gaussian distribution and we plotted it in Figure 4.5-c. We observe that the isocontours of the generated bivariate Gaussian distribution are similar to the ones from the *MetaDist* distribution, which indicates that both distributions are also similar. Thus, we conjecture that a bivariate Gaussian distribution fits the real distribution of ρ s and β s, making the *MetaDist* a good model to represent the population of users with a determined calls duration behavior.

Given that the *MetaDist* is a good model for the group behavior of the customers in our dataset, we can now visualize and measure how they evolve over time. In Figure 4.6 we show the evolution of the *MetaDist* over the four months of our dataset. The first observation we can make is that the bivariate Gaussian shape stands well during the whole analyzed period, what validates the robustness of the *MetaDist*. Moreover, a primary view indicates that the meta-parameters have also not changed significantly over the months. This can be confirmed by the first five rows of Table 4.2, which describe the value of the meta-parameters P , B and $\Sigma(\sigma_{\rho_i}^2, \sigma_{\beta_i}^2, cov(\rho_i, \beta_i))$ for the four analyzed months. This indicates that the phone company already reached a stable state towards its customers concerns, such as prices, plans and services. In fact, the only noticeable difference occurs between the first month and the others. We observe that the meta-parameters of the first month have a slightly higher variance than the others, what indicates that this is probably an atypical month for the residents of the country in which our phone records were collected. But in spite of that, in general, the meta-parameters do not change through time. Then, we can state the following observation:

Observation 1. TYPICAL BEHAVIOR. *The typical human behavior is to have an efficiency coefficient $\rho \approx 1.59$ and a weakness coefficient $\beta \approx -6.25$. Thus, the median duration for a typical mobile phone user is 51 seconds and the mode is 20 seconds.*

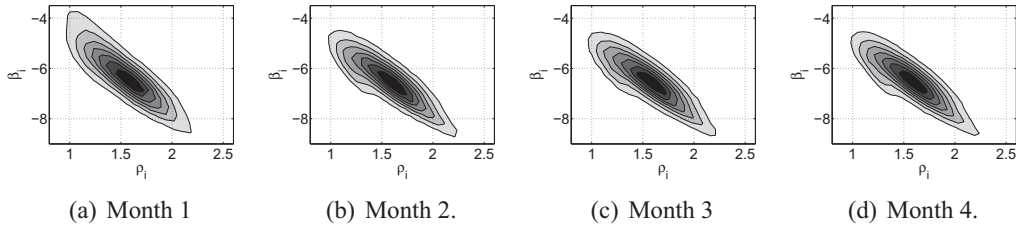


Figure 4.6. Evolution of the *MetaDist* over the four months of our dataset. Note that the collective behavior of the customers is practically stable over time.

4.2.3.2 Focal Point

An interesting observation we can derive from the *MetaDist* showed in Figure 4.5 is that there exists a significant negative correlation between the parameters ρ_i and β_i . This negative correlation, more precisely of -0.86 , lead us to the fact that the OR lines, i.e., the TLAC odds ratio plots of the customers of our dataset, when plotted together, should cross over a determined region. In order to verify this, we plotted in Figure 4.7-a the OR lines for some customers of our dataset. As we can observe, it appears that these lines are all crossing in the same region, when the call duration is approximately 20 seconds and the odds ratio is approximately 0.1. Then, in Figure 4.7-b, we plotted together the OR lines of 20,000 randomly picked customers and derived from them the isocontours to show the most populated areas. As we can observe, there is a highly populated point when the duration is 17 seconds and the OR is 0.15. By analyzing the whole month dataset, we verified that more than 50% of the users have OR lines that cross this point. From now on, we call this point the *Focal Point*.

Formally, the *Focal Point* is a point on the OR plot with two coordinates: a coordinate $FP_{duration}$ in the duration axis and a coordinate FP_{OR} in the OR axis. When a set of customers have their OR plots crossing at a *Focal Point* with coordinates $(FP_{duration}, FP_{OR})$, it means that for all these customers the $\frac{FP_{OR}}{1+FP_{OR}}$ -th percentile of their CDD is on $FP_{duration}$ seconds. Thus, in the two bottom lines of Table 4.2, we describe the *Focal Point* coordinates for the four months of our analysis and, surprisingly, the *Focal Point* is stationary. Thus, we can make the following observation:

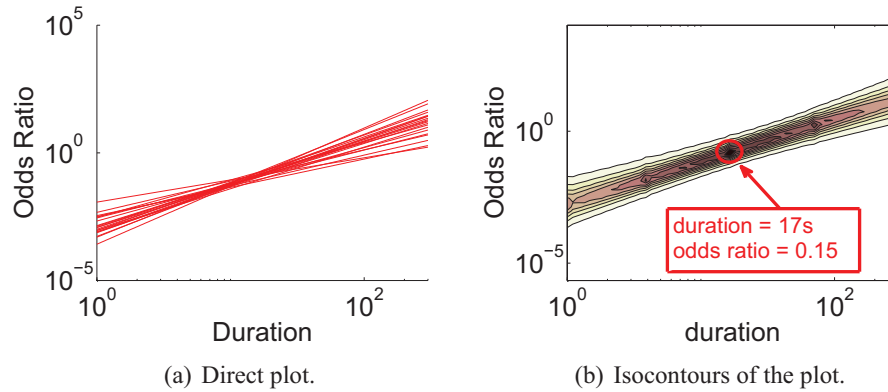


Figure 4.7. The TLAC lines of several customers plotted together. (a) We can observe that, given the negative correlation of the parameters ρ_i and β_i , that the lines tend to cross in one point. (b) We plot the isocontours of the lines together and approximately 50% of the customers have TLAC lines that pass on the high density point (duration=17s, OR=0.15).

Observation 2. UNIVERSAL PERCENTILE. *The vast majority of mobile phone users has the same 10th percentile, that is on 17 seconds.*

Observation 2 suggests that one of the risks for a call to end acts in the same way for everyone. We conjecture that, given the 17 s durations, this is the risk of a call to reach the voice mail of the destination's mobile phone, i.e., the callee could not answer the call. The duration of this call involves listening to the voice mail record and leaving a message, what is coherent with the 17 s mark. It would be interesting to empirically verify the percentage of phone calls that reaches the voice mail and compare with the *Focal Point* result.

-	1st month	2nd month	3rd month	4th month
P	1.59	1.58	1.59	1.59
B	-6.16	-6.28	-6.32	-6.30
$\sigma_{\rho_i}^2$	0.095	0.086	0.084	0.083
$\sigma_{\beta_i}^2$	1.24	0.98	0.95	0.94
$cov(\rho_i, \beta_i)$	-0.30	-0.24	-0.24	-0.23
$FP_{duration}(s)$	17	17	17	17
FP_{OR}	0.15	0.12	0.11	0.11

Table 4.2. Evolution of the meta-parameters (rows 1-5) and the *Focal Points*(rows 6-7) during the four months of our dataset.

4.2.4 Applications

4.2.4.1 Practical Use

In the previous section, we showed the collective behavior of millions of mobile phone users is stationary over time. We described two approaches to do that, one based on the *MetaDist* and the other based on the *Focal Point*. The initial conclusions of both approaches are the same. First, the collective behavior of our dataset is stable, i.e., it does not change significantly over time. Second, we could see a slight difference between the first month and the others, indicating that this month is an atypical month in the year. We believe that these two approaches can succinctly and accurately aid the mobile phone companies to monitor the collective behavior of their customers over time.

Moreover, since we could successfully model more than 96% of the CDDs as a TLAC, a natural application of our models would be for anomaly detection and user classification. A mobile phone user who does not have a CDD that can be explained by the TLAC distribution is a potential user to be observed, since he has a distinct call behavior from the majority of the other users. To illustrate this, we show in Figure 4.8 a talkative node with a CDD that can not be modeled by a TLAC distribution. We observe that this node, indeed, has an atypical behavior, with his/her CDD having a noisy behavior from 10 to 100 seconds and also an impressive number of phone calls with duration around 1 hour (or 5×700 seconds). Moreover, another way to spot outliers is to check which users have a significant distance from the main cluster of the *MetaDist*. As we showed in Figure 4.5-c, this can be easily done even visually.

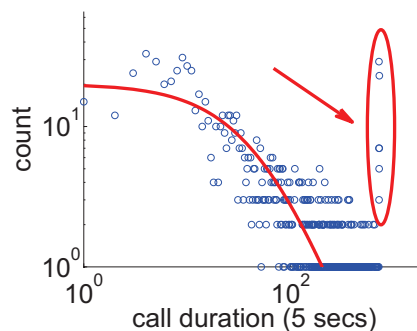


Figure 4.8. Outlier whose CDD can not be modeled by the TLAC distribution.

Another application that emerges naturally for our models is the summarization of data. By modeling the users' CDD into TLAC distributions, we are able to summarize, for each user i , hundreds or thousands of phone calls into just two values, the parameters ρ_i and β_i of the TLAC model. In our specific case, we could summarize over 0.1 TB of phone calls

data into less than 80 MB of data. In this way, it is completely feasible to analyze several months, or even years of temporal phone calls data and verify how the behavior of the users is evolving over time. Also, all the models proposed in this thesis can be directly applied to the design of generators that produce synthetic data, allowing researchers who do not have access to real data to generate their own.

4.2.4.2 Generality of TLAC

As we mentioned earlier, one of the major strengths of the TLAC model is its generality power. We showed that even for distributions that oscillate between log-normal and log-logistic, or that have irregular spikes that unable them to be correctly fit by TLAC, TLAC can represent them significantly well. Besides this, the simplicity of the TLAC model allows us to directly understand its form when its parameters are changed and verify its boundaries. For instance, in the case of the CDD, e^β gives the odds ratio when the duration is 1 s. Thus, when $e^\beta > 1$, most of the calls have a lower duration than 1 s, which makes the CDD to converge to a power law, i.e., the initial spike is truncated. Moreover, as $\rho \rightarrow 0$, the odds ratio tends to be the constant e^β , what causes the variance to be infinity. By observing Figure 4.9 and concerning human calling behavior, we conjecture that β is upper bounded by 1 and ρ is lower bounded by 0.5. These values are coherent with the global intuition on human calling behavior.

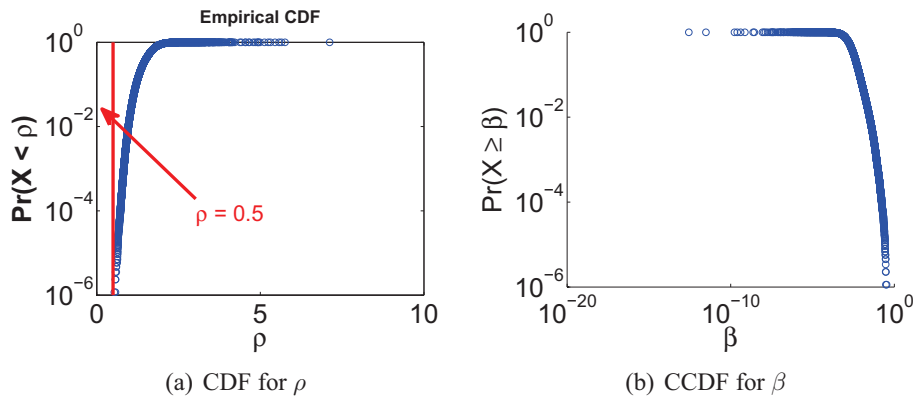


Figure 4.9. Cumulative distributions for ρ and β . We can observe that ρ is lower bounded by 0.5 and β is lower bounded by 1. These values are coherent with the global intuition on human calling behavior.

4.2.4.3 Additional Correlations

Given that the vast majority of users' CDDs can be represented by the TLAC model, it would be interesting if we could predict their parameters ρ_i and β_i based on one of their summarized

attributes. One could imagine that a user that makes a large number of phone calls per month might have a distinct CDD than a user that makes only a few. Moreover, we could also think that a user that has many friends and talk to them by the phone regularly may also have a distinct CDD from a user that only talks to his/her family on the phone. Thus, in this section we analyze the attributes of users who were correctly fitted by the TLAC model, i.e., approximately 96% of the users.

In Figures 4.10 and 4.11, we show, respectively, the isocontours of the behavior of the ρ_i and β_i parameters for users with different values of number of phone calls n_i , aggregate duration w_i and number of partners p_i , i.e., the distinct number of people that the user called in a month. With the exception made for the ρ_i against w_i , we observe that the variance decreases as the value of the summarized attribute increases. This suggests that the CDD of high or long talkative users, as well as users with many partners, is easier to predict. Moreover, as we can observe in the figures and also in Table 4.3, there is no significant correlation between the TLAC parameters and the summarized attributes of the users. Thus, we make the following observation:

Observation 3. INVARIANT BEHAVIOR. *The ρ_i and β_i parameters of user i are almost invariant with respect to (a) number of phone calls n_i , (b) aggregate duration w_i and (c) number of partners p_i .*

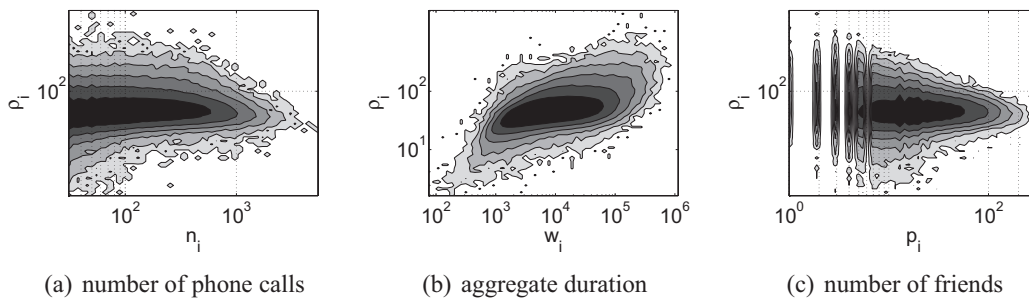


Figure 4.10. Isocontours of the users' CDD efficiency coefficient ρ and their summarized attributes.

Attribute	Correlation with ρ	Correlation with β
number of phone calls	0.14	-0.18
aggregate duration	-0.21	0.01
number of partners	0.18	-0.18

Table 4.3. Correlations between summarized attributes and ρ and β .

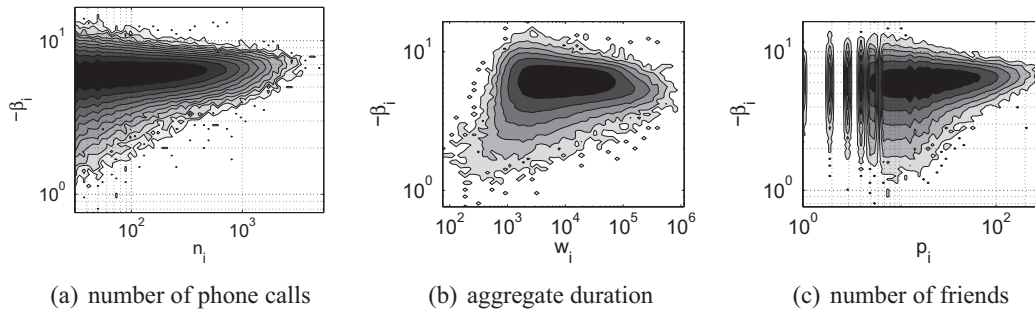


Figure 4.11. Isocontours of the users’ CDD efficiency coefficient β and their summarized attributes.

Finally, since there is no significant correlation between the users’ CDD parameters ρ_i and β_i with their summarized attributes, we emphasize that these parameters should be considered when characterizing user behavior in phone call networks. Moreover, besides characterizing individual customers, the TLAC model can also be directly applied to the relationship between users, analyzing how two persons call each other. One could use, for instance, the ρ parameter as the weight of the edges of the social network generated from phone call records.

4.3 Communication Dynamics

How long will it take until your next phone call, given your past history of phone call timestamps? Does phone call behavior differ than the behavior for text messaging, e-mails or postings on blogs? These communication activities are all “point processes”, and the simplest way to model them is by the Poisson Process (PP) [Haight, 1967]. Unfortunately, this simple and elegant model has proved unsuitable [Oliveira and Barabasi, 2005; Eckmann et al., 2004; Garriss et al., 2006; Vázquez et al., 2006], since analysis of real data have shown that humans have very long periods of inactivity and bursts of intense activity [Barabási, 2005; Jiang and Dovrolis, 2005], in contrast to the Poisson Process, where activities occur at a fairly constant rate.

4.3.1 Contradicting Models

Although researches agree that the Poisson Process (PP) is not suitable, there is no consensus about the right model between two major schools of thought. The main goal of this section is to propose a model which reconcile these differences.

The first viewpoint tries to fit power-laws. Barabási [2005] reported that the time intervals between human activities can be modeled by the so called *Universality Class Model*,

which states that a power law is an appropriate fit for the Probability Density Function (PDF) of the *inter-event times distribution* (IED). Barabási proposed that bursts and heavy-tails in human activities are a consequence of a decision-based queuing process, when tasks are executed according to some perceived priority. In this way, most of the tasks are rapidly executed while some of them may take a very long time. The queuing models generate power law Faloutsos et al. [1999] distributions $p(X = x) \approx x^{-\alpha}$ with slopes $\alpha \approx 1$ or $\alpha \approx 1.5$.

The second viewpoint is that the IED is well explained by variations of the Poisson Process (PP), such as the Interrupted Poisson [Kuczura, 1973] (IPP), Non-Homogeneous Poisson Process [Malmgren et al., 2008, 2009b], Kleinberg’s burst model [Kleinberg, 2002] and others. What they all have in common is that they suggest a piece-wise Poisson process: for the first interval, have rate λ ; for the next, change the rate (say, to zero, for the IPP, or to double-or-half for Kleinberg’s model), and continue. Malmgren et al. [2008] proposed a non-homogeneous Poisson process, where the rate $\lambda(t)$ varies with time, in a periodic fashion (e.g. people answer emails in the morning; then go to lunch; then answer more e-mails, etc). Although this model explains the data, it is not parsimonious, requiring several parameters and careful data analysis, being impractical for synthetic data generators, for instance. Later, the authors adapted this model to a more parsimonious version [Malmgren et al., 2009a], but it still has 9 parameters.

Given this two approaches, we ask: which is the correct one? Can we reconcile them all? A unifying model which reconcile these theories should, then: (a) generate a power-law-tail distribution for the inter-event time marginal, like [Barabási, 2005]; (b) behave as Poisson in the short term like [Malmgren et al., 2008] and (c) be also *parsimonious*, requiring a few an intuitive parameters.

Moreover, unlike all the articles we mentioned, we also analyze the temporal correlations, illustrating the “I.I.D. fallacy”, that has been routinely ignored. We show that, unlike the PP, that generates independent and identically distributed inter-event times (I.I.D.), individual sequences of communications tend to show a high dependence between consecutives inter-arrival times.

4.3.2 Marginal Distributions

In order to model human communication dynamics, we analyze four datasets containing social interactions and their timestamps. Each dataset presents a different type of interaction that is common and routine in most human lives. From this, we expect to accurately capture a large part of human dynamics behavior and expand our knowledge on it. We interchangeably call *time interval* and *inter-event time* Δ_t the time between the occurrence of two consecutive communication events. We call the *inter-event distribution* (IED) the distribution of the set of

the time intervals $\Delta_1, \Delta_2, \dots, \Delta_T$ that occurred between the communication events of a single type, e.g., phone call, SMS etc, for a given user. In our four datasets, since the granularity of t is in seconds, all values of Δ_t extracted from them is also in seconds.

4.3.2.1 The Most Active User

In Figure 4.12, we show the distribution of the time intervals Δ_t for the most active user in our four datasets, with 44785 SMS messages sent or received. The histogram is showed in Figure 4.12-a and, as we can observe, this user had a significantly high number of events separated by small periods of time and also long periods of inactivity. Moreover, either the power law fitting, which in the best fit has an exponent of -2 , or the exponential fitting, which is generated by a PP, deviates from the real data. The method we use to fit the power law is based on the Maximum likelihood estimation (MLE) and is described by Clauset et al. [2009].

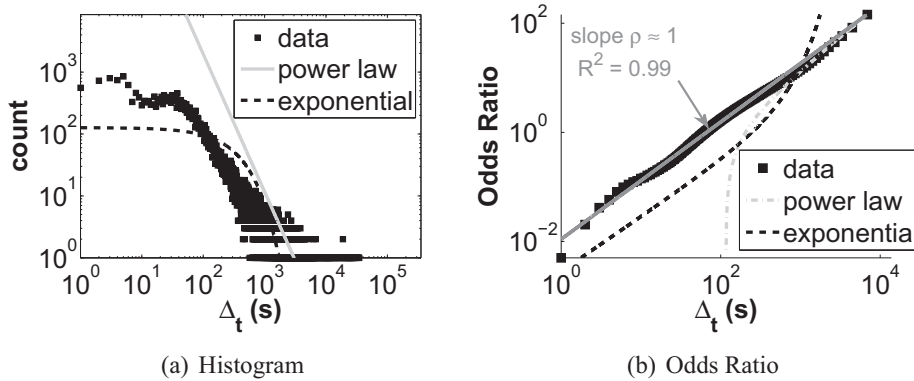


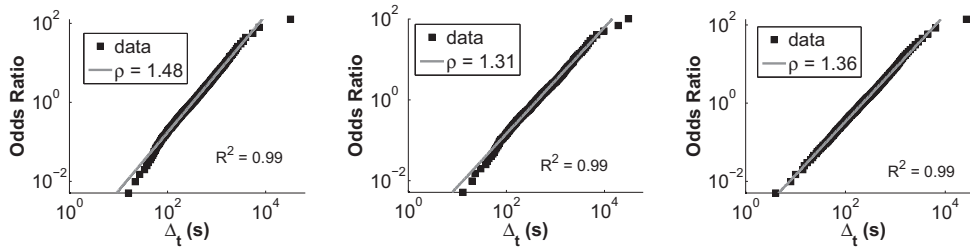
Figure 4.12. The inter-event time distribution of the most talkative user of our four datasets, with 44785 SMS messages sent and received. We observe that both the power law fitting (PL fitting) with exponent -2 and the exponential fitting, generated by a PP, deviate from the real data. We also observe that the OR is very well fitted by a straight line with slope ≈ 1 .

As we did in Section 4.2, in Figure 4.12-b, we also plot the OR for the selected user. From the OR plot, we can clearly see the cumulative behavior in the head and in the tail of the distribution. Also, observe again that both the exponential and the power law significantly deviate from the real data. Moreover, we can also observe that the OR of the inter-event times seems to entirely follow a linear behavior in logarithmic scales exactly as in the TLAC distribution, having, in this case, a power law behavior with OR slope $\rho = 1$. This is a surprising and fascinating result that we explore in the following sections.

4.3.2.2 Phone Data

In this section, we analyze the same dataset we analyzed in Section 4.2, which contains more than 3.1 million customers obtained from the network of a large mobile operator of a large city, with more than 263.6 million phone call records registered during *one month*. Again, we define that a communication event occurred for a user when he/she made or received a phone call. The IED of a user is computed by calculating the time intervals between two consecutive communication events of this user.

In Figure 4.13, we show the Odds Ratio of the IEDs of three anonymous and typical users of our phone dataset. We observe that the OR of all IEDs is well fitted by a power law, as the user of Figure 4.12. The only small deviations occurred for the inter-event times (i) below 10 seconds, and (ii) around 10 hours. The first deviation is probably due to errors in registering a phone call, since the time between two consecutive calls is usually lower bounded by a setup time Δ_t^0 that involves dialing the numbers, waiting for the signal, and waiting for the other part to answer the call. The second deviation is explained by the regular sleep intervals between two communication events. Since everyone has to go to sleep and it is not usual to call during the sleeping hours, it is common to have a time interval of approximately 10 hours at every 24 hours.



(a) User “Jones”, 2570 events (b) User “Jackson”, 1751 events (c) User “Johnson”, 3390 events

Figure 4.13. IEDs of anonymous users of the PHONE dataset. Observe that for these three users, the Odds Ratio is well fitted by a power law. The MLE fitting gives the same values for the slope ρ .

In order to verify the generality of this result, we check whether the OR of the IEDs of all users of our phone dataset can be explained by a power law. We perform a linear regression using least squares fitting on the OR of the IEDs of all users who have more than 30 communication events. Since the OR is a cumulative distribution, the linear regression is accurate. We performed a Kolmogorov-Smirnov goodness of fit test, but because of the setup time and the sleep intervals deviations, this test presented biased results. In the linear regression, we do not consider inter-event times lower than 10 seconds and also the $N_{days} - 1$ highest ones, where N_{days} is the number of days the analyzed month has.

In Figure 4.14, we show the Probability Density Function (PDF) of the coefficient of determination R^2 of the performed linear regressions¹. We observe that for the vast majority of the users of the phone dataset, the R^2 is close to 1.0. More specifically, the R^2 average is 0.99, which allows us to state that for the vast majority of the users, the OR of their IEDs is well fitted by a power law.

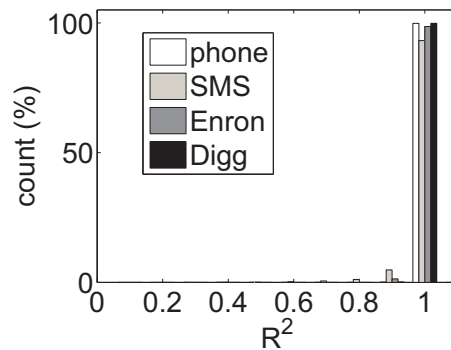


Figure 4.14. The goodness of fit of our proposed model. We show the Probability Density Function (PDF) of the R^2 s measured for every user in the four datasets. Observe that the R^2 value for the great majority of the users is close to 1.

In Figure 4.15, we show the PDF of the slope ρ of the linear regression. We observe that in the phone dataset, the vast majority of the users have a slope $\rho \approx 1$, the same one of the user in Figure 4.12-b. More specifically, the average slope $\bar{\rho} = 1.17$. Also related to the properties of the IEDs of the phone dataset, we show in Figure 4.16 the PDF of the median μ measured in seconds. Observe that the typical μ for the users of the phone dataset is approximately 1 hour.

4.3.2.3 SMS Data

From the same mobile operator of Section 4.3.2.2, we also have a SMS dataset of 300,000 users spanning 6 months of data, for a total of 8,784,101 records. For this dataset, we also have the information about the date and time it occurred and encrypted values that represent the source and the destination of the message, together with the time delay it took for a message to leave the source's mobile phone and arrive at its destination.

In Figure 4.17, we plot the OR of three IEDs of anonymous users of the SMS dataset. We observe that the OR of these three users is also well explained by a power law. Moreover,

¹The R^2 coefficient of determination is a statistical measure of how well the regression line approximates the real data points. An R^2 of 1.0 indicates that the regression line perfectly fits the data.

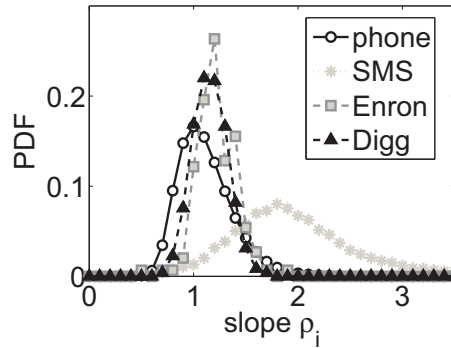


Figure 4.15. The PDF of the slopes ρ_i measured for every user u_i of our four datasets. Observe that the typical ρ_i for the users of the phone, Enron and Digg datasets is approximately 1, while for the users of the SMS dataset is approximately 2.

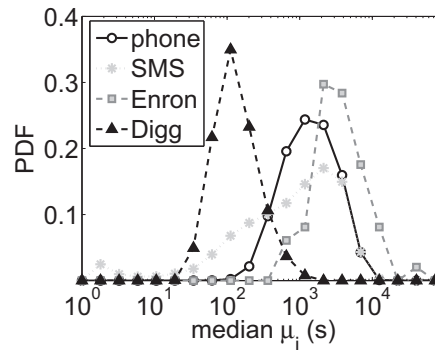


Figure 4.16. The PDF of the medians μ_i measured for every user of our four datasets. Observe that the typical μ_i for the users of the phone, SMS and Enron datasets is approximately 1 hour, while for the users of the Digg dataset is approximately 2 minutes.

this time we see several inter-event times with values lower than 10 seconds. This can be explained by the fact that when a user wants to communicate to multiple individuals simultaneously, he/she can do it by using a SMS message, while he/she can not do it by using a phone call. However, we again observe small deviations at the end of the OR function as we observed in the phone dataset, probably also due to the sleep intervals, which are around the 10-hour mark.

We also verify the generality of the fittings by performing a linear regression using least squares fitting on the OR of the IEDs of all users who have more than $6 \times 30 = 180$ communication events. Again, we do not consider the $N_{days} - 1$ highest ones, where $N_{days} - 1$ is the number of days the six months of our dataset have. Thus, in Figure 4.14, we also show the histogram of the coefficient of determination R^2 of the performed linear regressions in the SMS dataset. We observe that for the vast majority of the users, the R^2 is again close

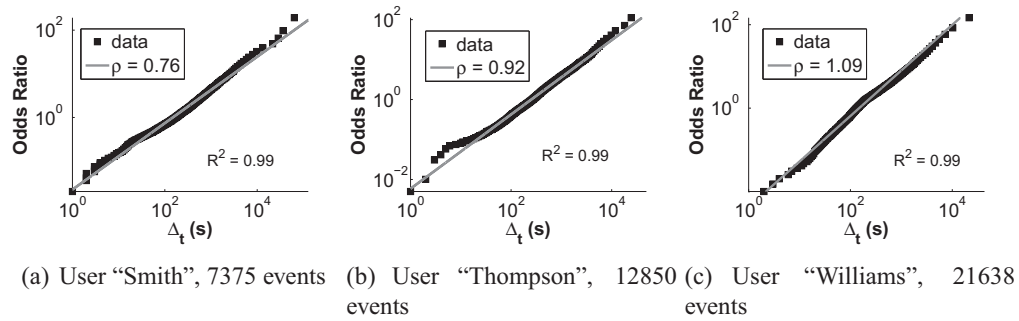


Figure 4.17. IEDs of anonymous users of the SMS dataset. Observe that for these three users, the Odds Ratio is well fitted by a power law. The MLE fitting gives the same values for the slope ρ .

to 1.0. More specifically, the R^2 average is 0.96, which allows us to state that for the vast majority of the users, the OR of their IEDs is well fitted by a power law. However, this result is worse than the one we obtained in the phone dataset, mainly because it is jeopardized by the existence of a significant transmission delay for a large fraction of the messages. This transmission delay is the time between a message leaves a mobile phone and arrives at the destination. It may happen, for instance, due to infrastructure or personal issues, e.g., a customer left his/her mobile phone unattended or the battery died, delaying all the incoming SMS messages for when the mobile phone is recharged again. These delays overestimate the inter-event times between SMS messages, making some of the IEDs significantly noisy. However, for highly talkative users, like the ones in Figure 4.17, the delays are less frequent and, therefore, do not play a significant role, making their IEDs to be accurately represented by the power law.

Observe again in Figure 4.15 the PDF of the slope ρ of the linear regressions. It is interesting that in the case of the SMS dataset, we observe that the vast majority of the users have a slope $\rho \approx 2$, different from the typical behavior we observed in the phone dataset. The average $\bar{\rho}$ is, in this case, 2.04. On the other hand, the typical median μ is almost the same as the one observed in the phone dataset, as shown in Figure 4.16, being approximately 1 hour. However, observe that the variance is higher and also a second mode on 1 second, which represents *multicast* SMS messages, i.e., messages that were sent to multiple recipients. These properties make the SMS traffic harder to predict in comparison to phone traffic.

4.3.2.4 Enron Data

The third dataset we look at is the public Enron e-mail dataset, consisting of 200,399 messages belonging to 158 users with an average of 757 messages per user [Klimt and Yang, 2004]. We consider that a communication event occurred for a user when he/she received or

sent an e-mail. We consider all the e-mails of the user, except the ones in the *delete* folder, that might be *SPAM* or other undesired e-mails.

In Figure 4.18, we plot the OR of three IEDs of highly active users of the Enron dataset. Again, we observe that the OR of these three users can be well explained by a power law. In this case, the main deviation we observe occurs in the 60-second mark. This occurs because some e-mails have their timestamps rounded up to the minute base when they are registered. For example, an e-mail that was sent at 8:00:35 is registered as it was sent at 8:01:00, making some inter-event times that were, in reality, lower than 60 seconds to be registered as if they were of 60 seconds. We also see some irregularities after the 10-hour mark, some of them due to the sleep intervals, others due to the weekend/Sunday interval.

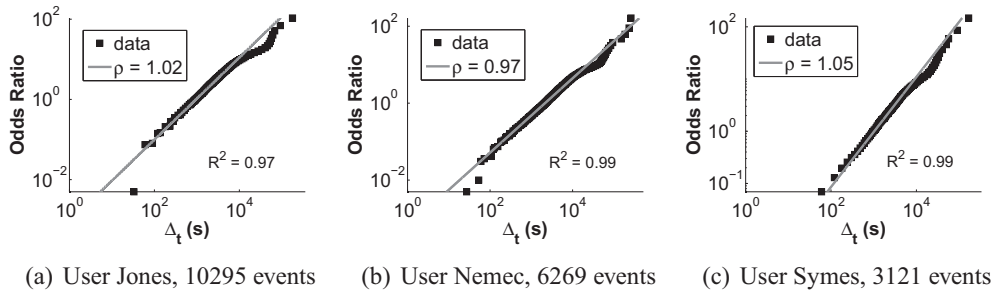


Figure 4.18. IEDs of three users of the Enron dataset. Observe that for these three users, the Odds Ratio is well fitted by a power law. The MLE fitting gives the same values for the slope ρ .

Figure 4.14 shows the PDF of the coefficient of determination R^2 of the performed linear regressions. Again, we observe that most of the R^2 values are close to 1. The average R^2 is 0.97, which is significantly high, considering the irregularities in the points below 60 seconds. Considering the slope ρ of the linear regressions, observe again in Figure 4.15 that the typical ρ is similar to the one verified in the phone dataset. This time, the average value is approximately $\bar{\rho} = 1.25$. The same happens for the typical median μ , that is close to the ones verified so far, being approximately 1 hour, as we observe in Figure 4.16.

4.3.2.5 DIGG Data

Finally, we analyze a public online news dataset, containing a set of stories and comments over each story. More specifically, the data is from the popular social media site Digg and has 1,485 stories and over 7 million comments [De Choudhury et al., 2009]. In this dataset, we consider that each individual is a story and a communication event is a comment published in a story.

In Figure 4.19, we show the OR of the IEDs of three highly active stories. Again, we observe that a power law is a good fit for the three stories. In these examples, we see deviations only when the inter-event times are above 10 hours, probably due to the sleep intervals. The PDF of the coefficient of determination R^2 of the performed linear regressions are again close to 1, as shown in Figure 4.14, with an average $R^2 = 0.98$. The average $\bar{\rho}$ is again similar to the average value verified in the phone and in the Enron dataset, being approximately $\bar{\rho} = 1.21$. On the other hand, the typical μ value is significantly small, being close to 2 minutes. This happens because the lifetime of a story published on Digg is also small, usually not generating interest after the first few days it was published.

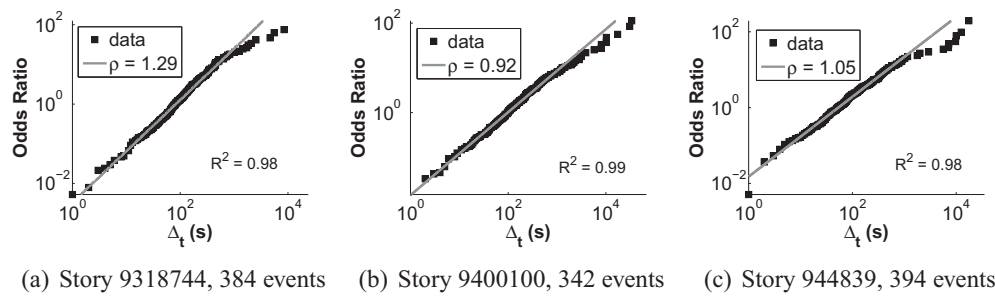


Figure 4.19. IEDs of three stories of the DIGG dataset. Observe that for these three stories, the Odds Ratio is well fitted by a power law. The MLE fitting gives the same values for the slope ρ .

Conclusions In this section we analyzed four different real datasets. We found a pattern for the IED of individuals: the Odds Ratio function of the inter-event times of individuals follows a power law with slope ρ . Thus, a realistic model for the IED should be able to generate this OR behavior.

4.3.3 Temporal Correlations

Most previous analysis focus on the marginal PDF. A subtle point is the *correlation* between successive inter-event times (Δ_{t-1} and Δ_t). Statistics textbooks routinely make the I.I.D. assumption: Independent, Identically Distributed. What we illustrate here is that the independence part is contradicted, by all our four real datasets.

In Figure 4.20-a, we plot, for one typical user of each dataset we are investigating, the median of the Δ_t s for their respective Δ_{t-1} s, all in logarithmic scales. The same was done for inter-event times generated from a Poisson process with $\beta = 3000$ seconds. Note that the Poisson process generates a flat line with slope 0, as expected. On the other hand, for the

four typical users of our dataset, the median of Δ_t grows with Δ_{t-1} : if I called you 5 years ago, my next phone call will be in about 5 years later. In short, there is a strong, positive dependency between the current inter-event time (Δ_t) and the previous one (Δ_{t-1}), clearly contradicting the 'independent' part of the I.I.D. assumption.

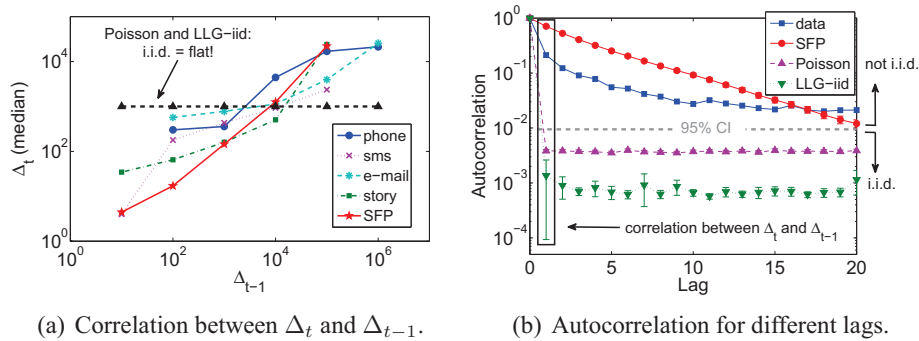


Figure 4.20. I.I.D. fallacy: dependence between Δ_t and Δ_{t-1} . (a) The median of the inter-event times Δ_t s for their respective preceding Δ_{t-1} s for one talkative user of each of the four datasets and for a synthetic data generated by the SFP, all represented in logarithmic intervals. (b) The sample autocorrelation for the same individual of Figure 4.12 and for synthetic data generated by the SFP, PP and LLG-iid, with the same number of phone calls and median. Observe that both the real data and the SFP indicates Δ_t and Δ_{t-1} are positive correlated.

We investigate this by analyzing the autocorrelation Box et al. [1994] of all the time series involving the inter-event times (Δ_t , $t = 1, \dots$) of the individuals of our datasets. The autocorrelation AC_k at lag k is defined as in the textbooks: it is the correlation of a time series with itself, after we lag it by k . A positive autocorrelation, which is suggested by Figure 4.20-a, indicates “persistence”, i.e., a tendency for a system to remain in the same state from one observation to the next.

Thus, we test if all the Δ_t time series of every individual of our datasets are random or autocorrelated. For this, we define the “independence” hypothesis H_0 , that a series $S = \{\Delta_0, \Delta_1, \dots, \Delta_n\}$ of inter-event times has values that are independent of each other. In that case, the autocorrelation coefficient AC_k should be near-zero ($AC_k \approx 0$) for all lags except zero ($k \neq 0$). More formally, if AC_k is between the 95% confidence interval for independence, then we accept the null hypothesis H_0 . As we show in Figure 4.20-b, we reject the null hypothesis H_0 that the inter-event times of the same individual of Figure 4.12 is independent, since all AC_k , $1 < k \leq 10$ are outside the confidence interval.

Since we are interested only in the case where the lag $k = 1$, we propose an alternative hypothesis test H_1 that the first-order autocorrelation coefficient AC_1 is greater than 0. If AC_1 is greater than the confidence interval for independence, then we accept H_1 , i.e.,

that there is a dependence between Δ_t and Δ_{t-1} . As shown in Figure 4.21, the empirical probability $P(H_1)$ of accepting H_1 for a given user with a given number of events increases rapidly as the number of communication events grows. This strongly suggests that the textbook I.I.D. hypothesis is violated by real data, that is, there is a dependence between Δ_t and Δ_{t-1} . Thus, we state that

$$\Delta_t = f(\Delta_{t-1}), \quad (4.7)$$

where f is a function that describes the dependency between Δ_t and Δ_{t-1} . This is the beginning of our proposed model.

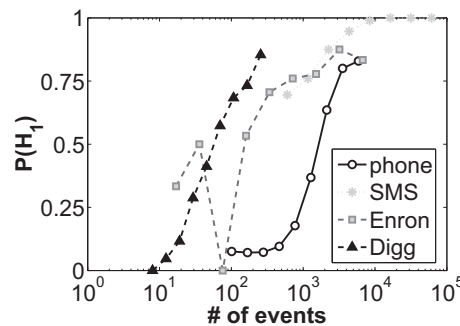


Figure 4.21. The empirical probability of an individual’s inter-event times to be auto-correlated given his/her number of events. Note that as the number of events grows, the probability of having an autocorrelated series also grows.

4.3.4 Proposed Model: “SFP”

Given all the above evidence (OR power law; I.I.D. fallacy) and all the previous evidence (power-law tails by Barabási; short-term Poisson behavior), can we possibly design a generator which will match all these properties? Our requirements for the ideal generator are the following:

R1: Realism – marginals The model should generate log-logistic marginal IED (thus obeying both the top-concavity, as well as the power-law for the odds-ratio);

R2: Realism – locally-Poisson: The model should behave as a Poisson Process, within a short window of time;

R3: Avoid the I.I.D. fallacy Two consecutive inter-event times should be correlated;

R4: Parsimony It should need only few parameters, and ideally, just one or two.

4.3.4.1 Self-Feeding Process

The textbook Poisson Process (PP), violates two of our four requirements above (4.3.4, 4.3.4). The two prevailing (but conflicting) models, also fall short: The model by Barabási [2005] does not show short-term Poisson behavior (requirement R2) nor addresses the I.I.D. fallacy (requirement R3). The non-homogeneous Poisson model proposed by Malmgren et al. [2008] violates the parsimony requirement (requirement R4), requiring an unspecified number of parameters, which are not easy to estimate from the data; neither it addresses the I.I.D. fallacy (requirement R3).

At the high level, our proposal is that the next inter-arrival time will be an exponential random variable, with rate that *depends on the previous* inter-arrival time. It is subtle, but in this way our generator behaves like Poisson in the short term, gives power-law tails in the long term (actually, we conjecture it is log-logistic), and is extremely parsimonious: just one parameter, the median μ of the IED. We call this model the *Self-Feeding Process* (SFP0).

We propose the generator as follows

Model 1. *Self-Feeding Process SFP0*(μ).

// μ is the desired median of the marginal PDF

$$\begin{array}{l} \Delta_1 \leftarrow \mu \\ \Delta_t \leftarrow \text{Exponential}(\text{mean } \beta = \Delta_{t-1} + \mu/e) \end{array}$$

where $\mu > 0$ is the only parameter of the model, being the median of the IED. This location parameter has to be higher than 0 to avoid Δ_t to converge to 0 and has to be divided by the Euler's number e to guarantee that the median of the generated IED is, in fact, μ – see more details in the Appendix. Although we selected μ to be the value of Δ_1 , we point out that Δ_1 may be an arbitrary value, since it does not affect the generated IED when the number of events generated is large.

In Figures 4.22-a and 4.22-b we compare, respectively, the PDF and the OR of the inter-event times generated by the SFP0model, all values rounded up, with the inter-event times of the user of Figure 4.12. Notice that the distributions are very similar, with only exception for the sleep intervals, which are not considered in the SFP0model (but could be easily added - we omit the details for clarity of exposition). Moreover, we see that both distributions are well fitted by a log-logistic distribution. As mentioned, the log-logistic distribution looks like a hyperbola, thus addressing both the power-law tail, as well as the top-concavity issue.

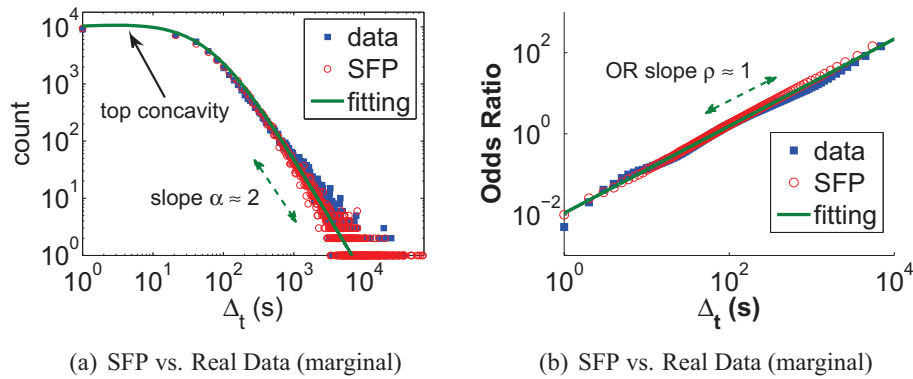


Figure 4.22. Comparison of the inter-event times generated by the SFP0 model with the inter-event times of the user of Figure 4.12. Observe that both the PDF (a) and the OR (b) are almost identical.

4.3.4.2 General Model

The current SFP0 model requires only one parameter, the median μ of the IED, giving power-law tails of slope $\alpha = -2$ and OR slope $\rho = 1$.

We show in Figure 4.15 the slopes ρ of the OR fitting for the IEDs of all users of our datasets. It is fascinating that the typical ρ_i for the users of the Phone, Enron and Digg datasets is approximately 1, the same slope generated by the SFP0 model. However, several users, mainly from the SMS dataset, have a much higher value of ρ , close to $\rho \approx 2$. To accommodate for that, we introduce our Generalized SFP model, which needs just one more parameter, ρ . Thus, we have

Model 2. *Generalized Self-Feeding Process* $SFP(\mu, \rho)$.

$$\begin{array}{l}
 \delta_1 \leftarrow \mu \\
 \delta_t \leftarrow \text{Exponential}(\text{rate: } \beta = \delta_{t-1} + \mu^\rho / e) \\
 \Delta_t \leftarrow \delta_t^{1/\rho}.
 \end{array}$$

Note the auxiliary variable δ_t , which stores the inter-event times without the influence of ρ .

4.3.5 Discussion

Earlier, we listed four requirements for the ideal model. The last one, parsimony (requirement R4), is clear: SFP usually needs just *one* parameter (and at most two, in general). For the remaining 3 requirements, we discuss next how SFP meets them all.

4.3.5.1 R1: SFP matches the marginals

As we have shown, the SFP model mimics the behavior of the IED of the vast majority of the users of our four datasets. To the best of our knowledge, this is the first paper that studies the IED of human communications using such a varied, modern and large collection of data. This collection of data represents four types of communication present in the routine of most human lives. Despite the fact these means of communications are intrinsically different, having their own idiosyncrasies, we have observed that the IED of individuals of these systems have the same characteristics, i.e., they follow an odds ratio power law behavior. This power law behavior is also generated by the SFP model, which naturally generates an odds ratio power law with slope $\rho = 1$, which is, by its turn, the slope that characterizes the majority of the users of our datasets, as we see in Figure 4.15.

Moreover, notice that when the OR slope $\rho = 1$, the power law exponent of the PDF is $\alpha = -2$. This is the same IED slope α reported by Hidalgo [2006] and Vázquez et al. [2006] as a result of fluctuations in the execution rate and in particular periodic changes. It has been argued that seasonality can only robustly give rise to heavy-tailed IEDs with exponent $\alpha = 2$. However, we again point out that the proposed (Generalized) SFP model can generate IEDs with power-law slopes α varying from $(-1, -\infty)$, agreeing with all the empirical reports we have knowledge.

4.3.5.2 R2 - Unifying power of the SFP

Requirement R4.3.4 states that, in the short term, real data behave as if Poisson. Our model captures that, since successive inter-event times are exponentially distributed, with similar (but not identical) rates. Thus, one of the major contributions of this work is the unification of the two seemingly-conflicting viewpoints, that we mention in the introduction.

The first viewpoint, expressed in [Barabási, 2005], [Oliveira and Barabasi, 2005] and later in [Barabasi et al., 2005], proposes a power-law model for the inter-event times distribution. The second viewpoint, [Stouffer et al., 2005; Malmgren et al., 2008, 2009b] argues that a power law is not suitable, and that, instead, a log-normal or a “non-homogeneous” Poisson process is better.

The proposed SFP model unifies both theories: like the second viewpoint [Malmgren et al., 2008, 2009b], it generates Poisson-like traffic in the short term, with slowly varying rate; like the first viewpoint [Barabási, 2005], it generates a power-law tail distribution (see the Appendix), even matching the top-concavity that power laws can not match.

The SFP also generates burstiness and long periods of inactivity, which are also characteristics of human generated communication traffic. Bursts are automatically triggered by

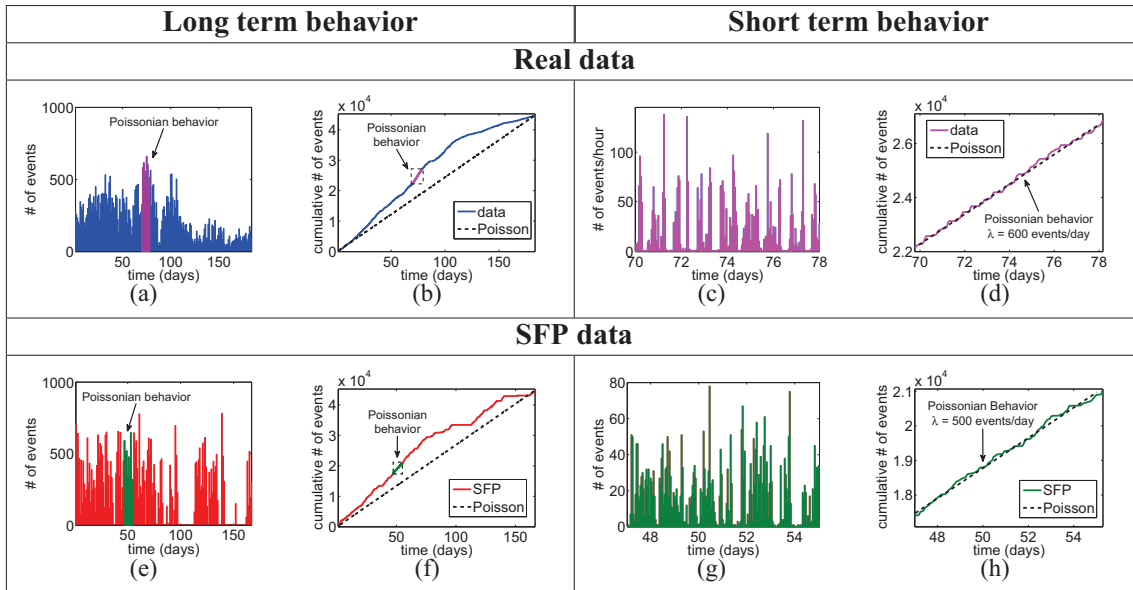


Figure 4.23. Unification Power of SFP: non-Poisson/bursty in the long term (first two columns), but Poisson in the short term (last two columns). First row: Traffic of the user of Figure 4.12 - event-count per unit time ((a), (c)) and respective cumulative event-count ((b), (d)). Second row: synthetic traffic, by the SFP model (with matching μ , ρ and event-count). Observe that (1) both time series are visually similar; (2) both are bursty in the long run (spikes; inactivity) (3) both are Poisson-like in the short term (last two columns)

the SFP when a small inter-event time Δ_t is generated. When this occurs, the tendency is that a random set of small inter-event times is subsequently generated, resulting in a burst of activity. A long period of inactivity is generated analogously when a large inter-event time Δ_t is generated.

In order to verify whether the SFP is able to mimic human communication behavior over time, we generate synthetic data from the SFP model using the same odds ratio slope ρ , median μ and number of events of the user of Figure 4.12. In Figure 4.23, we compare this synthetic data with the real data from this user. Notice the bursts of activity and also the long periods of inactivity, in the first two columns of Figure 4.23. Also notice that both synthetic and real traffic significantly deviate from Poisson (sloping lines in (b) and (f)) but are similar between themselves. However, in the short term (Figure 4.23(c-d), (g-h)), both real and synthetic data behave like Poisson, being practically on top of the black dashed lines of (d) and (h). This agrees with earlier reports of traffic characterization [Karagiannis et al., 2004].

Finally, we would like to point out that a lot of past activity tried to modify the Poisson process to match reality, such as the Interrupted Poisson [Kuczura, 1973] (IPP), Kleinberg's

burst model [Kleinberg, 2002], and Non-Homogeneous Poisson Process [Malmgren et al., 2008, 2009b]. What they all have in common is that they suggest a piece-wise Poisson process: for the first interval, have rate λ ; for the next, change the rate (say, to zero, for the IPP, or to double-or-half for Kleinberg's), and continue. They typically have too many parameters, and/or they do not exhibit a power-law behavior. We reconcile all these ideas with our SFP model: it behaves like a piece-wise Poisson process, with the subtle but important difference that it is *self-feeding*: the next arrival rate depends on the immediate past.

4.3.5.3 R3: SFP avoids the I.I.D. fallacy

Although the marginal IED can also be accurately matched by a plain log-logistic distribution, we point out that it cannot match the temporal dependence between consecutive inter-event times as the SFP does. We have shown that, for an individual, the time Δ_t it will take for the next communication event to arrive depends on the time Δ_{t-1} the previous one took to arrive, in a way that $\Delta_t = f(\Delta_{t-1})$, where $f(x)$ is the function defined by the classic PP. Consider, for instance, an event generator similar to a PP, that we call *LLG-iid*, which instead of generating exponentially distributed inter-event times, it generates log-logistically distributed inter-event times. Despite the fact LLG-iid provide good fits to the marginal IED, we see in Figure 4.20 that it shows no correlation between Δ_t and Δ_{t-1} .

4.4 Final Remarks

In this chapter, we explored the behavior of social agents in a large communication network. First, we analyzed the duration of the communication flow between two agents and then we studied the inter-event time interval between communication events. From these analysis, we could understand the motivations behind the communication activities between humans. We verified that the TLAC model is an appropriate model for either the size of the communication flow and for the inter-event intervals. The main contributions of the work described in this chapter are:

- Proposal of the SFP model, which matches real data well, and has a very simple, intuitive explanation: the time for the next communication event depends on the time it took for the previous event to occur;
- The SFP model unifies existing theories on human communication dynamics, i.e., it exhibits power law tail behavior, burstiness, as well as locally-Poisson behavior;
- The proposal of the TLAC model, which fits very well the vast majority of individual phone call durations, *much better* than log-normal and exponential;

- Proposal of applications based on the TLAC model, such as the introduction of *MetaDist*, which shows that the *collection* of TLAC parameters, and specifically the ρ and β ones, follow a striking bivariate Gaussian, with mean (P, B) . We show it can spot anomalies (see Figure 4.8) and it can succinctly describe spot correlations (or lack thereof) between total phone call duration, number of calls, and number of distinct patterns, for a given user.

Chapter 5

Predicting in Competitive Networks

5.1 Introduction

In competitive networks, the agents of the system compete among themselves for a reward or limited resources, i.e., there is no reward to every agent and the resources are divided unevenly. Examples of this kind of network can be found, for example, in networks of workers looking for job positions, e.g. LinkedIn [LinkedIn, 2010], and in professional sports leagues. In these cases, the organizations, companies or teams want to sign up the best agents, workers or players, for the lowest possible cost and, by their turn, these agents want to receive the maximum possible salary. Moreover, either the organizations and agents want the organizations to grow, that is, expand in the market or win titles in their leagues. This scenario presents several conflicts of interests that may unveil interesting observations over the social agents of this type of network.

Thus, in this chapter, we tackle two aspects of DBCNs: modelling and predicting. We analyze the network formed from the teams and players of sports leagues. We start our analysis with the National Basketball Association (NBA) through the 63 first years of the existence of the league. In the modeling stage, we view the NBA as a complex network in evolution. Then, in the prediction stage, we develop metrics that are correlated with the behavior of NBA teams, taking into account only the social and work relationship among players, coaches and teams. Then, based on these metrics, we propose models to predict how well a team will perform in the following season. We also evaluate the prediction models on the Major League Baseball (MLB) dataset and, as we can observe in Figure 5.1, the NetForY, one of our proposed models, performs surprisingly well on both datasets. In summary, the main contributions of this chapter are:

- we show that in the NBA the regular box-score statistics distributions are highly

skewed, with only a few players having high values. Moreover, acquiring players who presented high values of box-score statistics in previous seasons does not guarantee a team performance improvement;

- we propose general network features that are good indicators of the teams performance in sports leagues. As an example, we show that a high team volatility is not good for the team performance; and
- we propose network-based models to predict the behavior of teams in sports leagues, which present surprisingly good results when compared to other approaches.

We emphasize that the proposed prediction models are generic and may be applied to any team sports league, i.e., they do not rely on any particular box score statistics, relying only on the feedbacks generated by the social motivations of the agents.

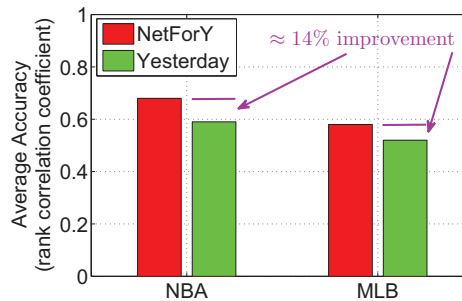


Figure 5.1. Average accuracy of the network-based models and of the Yesterday model, that is currently the state of the art. The accuracy is measured using the Spearman’s rank correlation coefficient ρ . While the NetFor has similar performance, the NetForY consistently beats the “Yesterday”, with up to 14% improvement.

This chapter is organized as follows. First, in Section 5.2, we present the related work. In Section 5.3, we show our method for modeling a sports league into a network and, in Section 5.4, we show the network implicit feedback features that give valuable information on the teams performance. From these features we propose the network-based prediction models in Section 5.5, which are evaluated in Section 5.6. In Section 5.7, we directly apply the proposed models to predict the behavior of the teams in the MLB dataset, showing that they are general models. Finally, in Section 5.8, we present some comments.

5.2 Related Work

The use of networks to model collective sports is particularly interesting because of their dynamics. Teams constantly change players, players may exchange techniques among them, synergy among players may influence the performance of their teams, and so on. However, little attention has been given in the literature to such dynamical systems in terms of networks.

Girvan and Newman [2002] modeled the United States college football league as a network, where teams are the vertices and edges represent regular-season games between the two teams they connect. They use the college football league network to check whether their algorithm detects the communities of teams, i.e., the league conferences. Later, Park and Newman [2005] proposed a ranking system for the United States college football league based on the network properties derived from this league. Moreover, Fast and Jensen [2006] modeled the National Football League (NFL) as a network to analyze the influence that coaching mentors have on their protégés. They identified notable coaches and characterized championship coaches, using this information and the social relationships to predict which teams will make the playoffs in a given year. Finally, Onody and de Castro [2004] analyzed the statistics of the Brazilian National Soccer Championship and they concluded that the players' connectivity has increased over the years while the clustering coefficient declined. They suggested that the possible semantic reasons for this phenomenon are the exodus of players going abroad, the increasing number of players traded among national teams and, finally, the increase in the player's career time.

Another interesting study relevant to our proposal is the work of Ben-Naim et al. [2007], which presents a statistical analysis to quantify the predictability of all sports competitions in five major sports leagues in the United States and England. To characterize the predictability of games, the authors measure the "upset frequency" (i.e., the fraction of times the underdog wins). While basketball and American football leagues have the lowest upset frequencies, soccer and baseball leagues have the highest ones, meaning that the former ones are easier to predict than the later ones. Given that, we start our analysis based on a basketball database and then we move to a baseball database, which is supposedly harder to predict and, therefore, a better test for our proposed network-based models.

Finally, in terms of forecasting in sports leagues, many different types of models have been constructed to predict the outcomes of sporting events, but unfortunately, many of these models have never been used in forecasting beyond the period of fit [Stekler et al., 2010]. We believe that the main reason for that is the difficulty of competing with one of the simplest but most powerful models, the "Yesterday" model. In a 2006 article for the feature section of ESPN.com *Page 2*, Easterbrook [2006], a NFL specialist, proposed a prediction for the

upcoming season: “I predict that every NFL team will end the 2006 season with the same record as it did in 2005”. What he proposed is the exact idea of the “*Yesterday*” model. He commented that this “obviously won’t be right, but it will be closer than the countless pseudo-scientific forecasts floating around”. In advance, we point out that he is partially right. The “*Yesterday*” is, to the best of our knowledge, the current state of the art for forecasting in generic team sports leagues and, as we will show in this chapter, present consistently good results. If there is one model to be beaten, this is the “*Yesterday*” model.

5.3 The NBA Network

5.3.1 Problem Definition

The main goal of this chapter is to show the high potential of network effects on predicting the behavior of complex social systems. Thus, we define the following general problem:

Problem 1. *NETWORK-BASED FORECASTING.* Given a network G of players, coaches and teams with edges signifying how long they played together, how can we use G to predict the future behavior of this system?

Despite the fact that in this chapter we are analyzing team sports leagues, we strongly believe that the relevance of Problem 1 goes beyond these systems. While the main sports leagues have uncountable explicit data available, real world systems frequently have missing or erroneous data. For example, in online auctions websites, e.g., *eBay*, some users may provide erroneous information to improve their credibility. In this case, network effects can be used to spot anomalies and fraudulent users [Pandit et al., 2007]. In the same direction, network effects can be used to detect frauds and other violations among brokers [Neville et al., 2005]. Moreover, as another example, the social connections a person has may tell more about him/her than his/her personal attributes explicitly described in his/her *curriculum vitae* [Shetty and Adibi, 2005]. These examples show the generality power of the network effects of social systems, being completely independent of any kind of individual attribute of their entities.

Thus, we begin our analysis by modeling the historical NBA database as a network in evolution. The NBA data we used in this chapter are publicly available at the site databaseSports.com [2007]. This site provides all the NBA statistical data in text files, from the year of 1946 to the year of 2008 and, among these data, it provides information on 3863 players, 265 coaches and 71 teams, season by season or by career. Our main goal is to move beyond the ordinary individual box-score statistics presented in that database and use network theory to discover new knowledge in the simple recorded numbers.

5.3.2 Motivation

The United States National Basketball Association (NBA) was founded in 1946 and since then is well known for its efficient organization and for its high level athletes. After each game played, a large amount of individual statistical data, such as Points Per Game (PPG), Assists Per Game (APG) and Rebounds Per Game (RPG), are generated describing the performance of each player in the match. These statistics, called box-score statistics, are used to characterize the performance of each player over time, dictating their salaries and the duration of their contracts. They are also used by many services to provide more reliable predictions on the outcome of upcoming games. But one question that naturally arises is: are the box-score statistics the only available information capable of aiding in the prediction of a team behavior? Again, we point out that PPG, RPG and APG only measure the actions of a player within a second or two when someone shoots the ball. The rest of the time, points, rebounds and assists measure nothing [Abbot, 2007].

In Figure 5.2, we show, by plotting the complementary cumulative distribution (CCD), the probability of a player to have points, assists and rebounds averages in a season greater than a determined value. We also show the CDD for the general efficiency of the players for every season analyzed, which is computed as

$$\text{Efficiency} = \frac{\text{points} + \text{rebounds} + \text{assists}}{\text{games played}}. \quad (5.1)$$

We observe that the CCDs for these box-score statistics have almost the same shape, characterized by a significant drop after the 90th percentile, i.e., $Pr(X > x) = 10^{-1}$. We see that for all seasons analyzed, 90% of the players have marginal box-score statistics averages, lower than 11 PPG, 4 APG, 8 RPG and efficiency of 12. This means that the majority of the players contribute to their teams in ordinary ways, if we look only at box-score statistics. As an example, considering that an average team scores 90 points per game, the probability of having a player in a season scoring more than one third of the team's points per game is less than 0.4%.

Moreover, we show in Figure 5.3 the probability density function (PDF) of the number of points, rebounds and assists the players achieved in their careers in the NBA. We observe that the distributions follow a power law with an exponential cutoff in the tail, which is approximately in the 90th percentile for the three distributions. This indicates that, in general, 90% of the players have not scored more points, rebounds or assists than the cutoff value that, respectively, is 22%, 14% and 11% of the maximum value in points, rebounds and assists. Once again, we conclude that the majority of players have ordinary careers in terms of

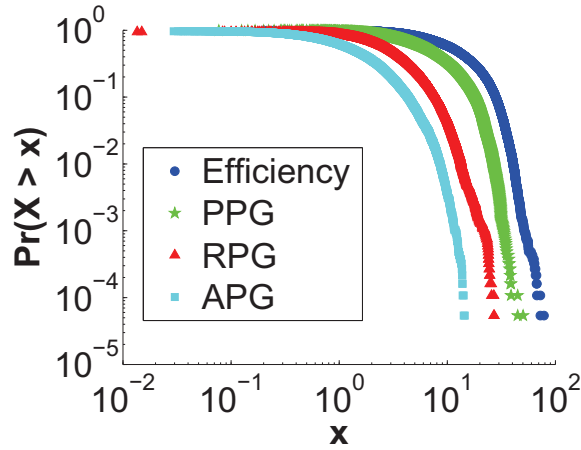


Figure 5.2. The majority of players marginally contributes to their teams in terms of box-score statistics in comparison with a few players who have significant contributions. Observe the complementary cumulative distribution of the players efficiencies and averages in points, assists and rebounds per season.

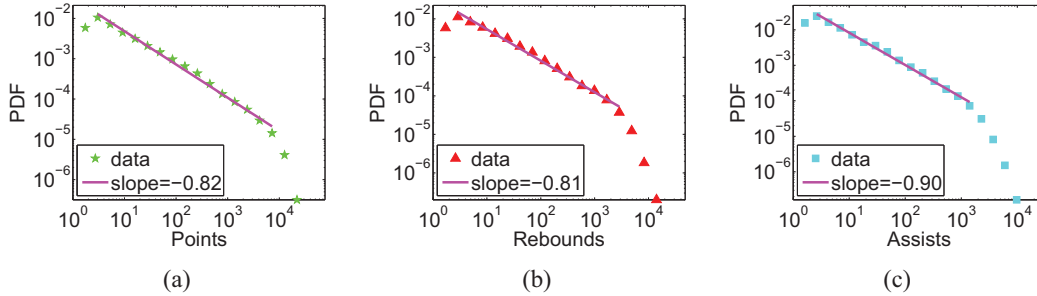


Figure 5.3. The majority of players have scored in their careers marginal values of box-score statistics in comparison with a few players who have significant contributions. Observe the probability density function (PDF) of the number of points, rebounds and assists the players achieved in their careers. The distributions follow a power law with a cutoff in the tail.

box-score statistics in comparison with a few players who have significant contributions.

Figures 5.2 and 5.3 lead us to conclude that only a few players significantly contribute to a team in terms of box-score statistics. Thus, if we consider that the only way to predict a team success is to analyze box-score statistics then we are restricted to the analysis of a small fraction of players.

Now we analyze the impact of acquiring or losing players with a determined efficiency value on the performance of the teams. But before that, we define the performance metric $f(t, y)$ of team t in year y as

$$f(t, y) = \text{number of victories of team } t \text{ in regular season } y. \tag{5.2}$$

We chose this performance metric because it characterizes the teams performance with fairness, since it judges the teams based on their performances over the entire season, with every team playing the same number of games.

Thus, given the performance metric $f(t, y)$, we define the performance r_{τ}^y of team τ in year y as

$$r_{\tau}^y = \text{the percentage of teams } t, \text{ such that } f(t, y) < f(\tau, y), \quad (5.3)$$

or simply the percentage that indicates the amount of teams that had a lower number of victories than team τ in regular season y . The best performance team has $r_t^y = 100\%$ and the worst performance team has $r_t^y = 0\%$.

In Figure 5.4 we show the average performance gain, with its standard deviation, a player produces when he is transferred from a team t_{out} to another team t_{in} . The performance gain g_m indicates how much the team the player left t_{out} lost and how much the team the player joined t_{in} won with the transaction¹ m , being defined as:

$$g_m = (r_{t_{in}}^y - r_{t_{in}}^{y-1}) + (r_{t_{out}}^{y-1} - r_{t_{out}}^y), \quad (5.4)$$

where the term $(r_{t_{in}}^y - r_{t_{in}}^{y-1})$ refers to how much t_{in} won with the transaction and the term $(r_{t_{out}}^{y-1} - r_{t_{out}}^y)$ refers to how much t_{out} lost. High values of the performance gain indicate that the team the player left decayed its performance with his departure and the team he joined improved its performance. If the performance gain is zero, no significant change occurred.

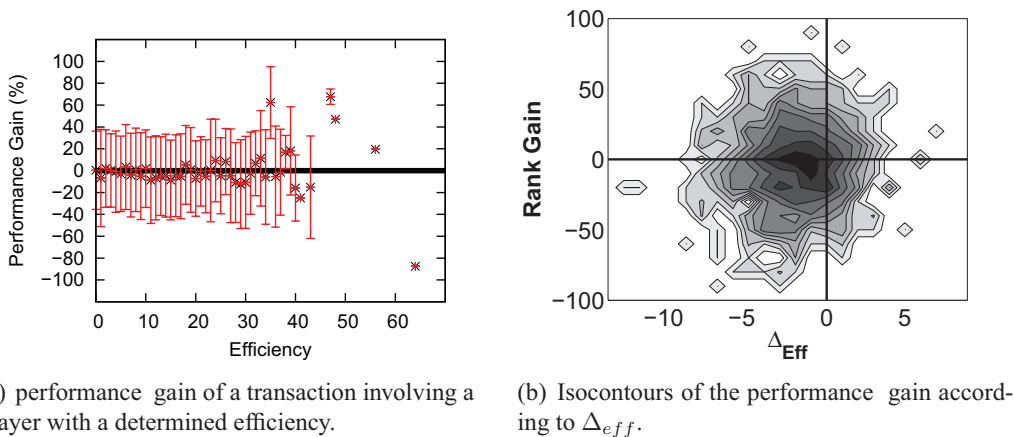
We observe in Figure 5.4-a that there is no rule for the performance gain based on the player efficiency, that is, the average performance gain is zero for all efficiency values below 40. For the efficiency values greater than 40, we cannot state anything, since the number of transactions involving players who have efficiency values higher than 40 is not significant, only 20 in the history of the NBA. This also corroborates the fact that individual box-score statistics of players are rarely relevant for predicting the performance of a team.

Finally, we also investigate whether a team will improve or decrease its performance when it changes its roster. In order to do this, we look at the efficiency values of the players this team had in the previous season and currently has. Thus, we define the ΔEff_t^y for a team t in year y as

$$\Delta Eff_t^y = avg\left(\sum_{\forall p_t^y} eff(p_t^y)\right) - avg\left(\sum_{\forall p_t^{y-1}} eff(p_t^{y-1})\right), \quad (5.5)$$

¹In this chapter, the term transaction refers to a exchange of teams by a player.

which is the difference between the averages of the efficiency values of the players this team had in year y and in year $y - 1$. A high value of ΔEff_t^y means that team t had higher efficiency valued players in year y compared to year $y - 1$. In Figure 5.4-b we show the isocontours of the performance gain according to ΔEff . Most of the transactions are in the dark red region where ΔEff is between -5 and 0 and the performance gain is approximately 0 . Moreover, it is clear that there is no correlation between ΔEff and the performance gain, since the performance gain values are almost symmetric to the $\Delta Eff = 0$ line. This also implies that simply acquiring players with high box-score statistics is not a guarantee for a team success. Besides this, it is interesting to note that most of the times the teams have negative ΔEff values from year to year, what suggests that changing the team roster is usually not a good strategy. This will become clearer when we describe the network-based features in Section 5.4.



(a) performance gain of a transaction involving a player with a determined efficiency.

(b) Isocontours of the performance gain according to Δ_{eff} .

Figure 5.4. The efficiency of players acquired by a team has, in general, little effect on its future performance.

5.3.3 The Network Definition

In the previous section, we showed that the usual box-score statistics are not good predictors of the future performance of NBA teams over the years. Moreover, we showed that box-score statistics alone cannot predict if a transaction will be good or bad for a team. Thus, in order to explain and better understand the consequences and causes instigating the transactions dynamics, we propose modeling the NBA as a network (or graph) in evolution. We believe that metrics that capture network changes over time may provide implicit feedback signals that might characterize the temporal situation of the teams and players in the NBA. We

emphasize that the following model can be extended to any team sports league that involves teams and players and, in fact, it will be used in Section 5.7 for modeling the MLB database.

In the network we construct from the NBA, the set of vertices V contains three types of vertices: the set of players vertices P , the set of coaches vertices C and the set of team vertices T . Since in the NBA the players, coaches and teams change over time, we construct one network $G^y(V^y, E^y)$ per year, each one containing a set of player vertices $P^y \subset V^y$, a set of coach vertices $C^y \subset V^y$ and a set of team vertices $T^y \subset V^y$. But before this, we initially construct the Yearly NBA Networks (YNN) $G'^y(V'^y, E'^y)$ for every year y , where V'^y is the set of players P'^y , coaches C'^y and teams T'^y that are active in season y . The set of edges E^y are defined according to the labor relationships that exist between players, coaches and teams in V^y . We link a player p or a coach c to a team t in E^y if and only if p or c played or coached for t in y . Moreover, we link a player p to other player q in G'^y if and only if p played together with q for a common team in y . Moreover, we link a player p to a coach c in G'^y if and only if p was coached by c in a common team in y . Clearly the data is temporal and the characteristics of players and teams change each year and, thus, we use t^y , c^y and p^y to denote, respectively, the nodes of team t , coach c and player p in year y .

After constructing all YNNs, we are now able to construct the NBA Networks that contain the information about the historical relationships among players and teams. In this way, we recursively define the NBA Network $G^y(V^y, E^y)$ as the graph that contains the information in G'^y and also the information in G^{y-1} . The use of historical information enables us, for instance, to search for players who have played for a high number of teams, or to search for teams that are frequently changing their rosters significantly over the years. There are several ways to propagate the information contained in G^y to G^{y+1} and we describe some possible propagation models below:

1. **Historical:** $G^{y+1} = G'^{y+1} \cup G^y$. This model propagates all the vertices and edges through the years, never removing them from the network. We use this model in this chapter.
2. **Yearly:** $G^{y+1} = G'^{y+1} \cup G^y(V'^{y+1}, E^y)$. This model removes from G^{y+1} all vertices that left the NBA in $y + 1$, i.e., it considers only players and teams that are active in y .
3. **Delayed:** $G^{y+1} = G'^{y+1} \cup (G^y(V^y, E'^{y+1} \cup (E^y - E^{y+1-\Delta_y}))$. This model removes from the G^{y+1} the edges that are Δ_y years old. When Δ_y is equal or higher than the total number of years in the dataset, the *Delayed* model propagates in the same way as the *Historical* model. On the other hand, when $\Delta_y = 1$, it propagates as the *Yearly* model does.

Thus, our goal is to move beyond the usual box-score statistics and discover new knowledge in the network properties of the NBA, which are defined by the labor and social relationships among players, coaches and teams. We are interested in knowing, for instance, if a team that had always in its roster players who have already played with each other in previous seasons is good for the teams performance or, as another example, if a player who has played already with a large amount of other players in previous seasons is a player who will help his team to improve its performance.

The labor and social relationships among players, coaches and teams are defined by network changes, which basically occur in two situations: (i) when a new player or team joins the NBA or (ii) when a transaction occurs, that is, an exchange of teams by a player. Semantically, Dilger [2002] showed that a player may leave a team when he is not performing well or when his salary is high enough to force his team to free some space in its payroll budget due to the salary cap². Besides these reasons, we point out that a player may also leave a team when he becomes a free agent and wishes to join another team which, according to his judgment, has more chances to win the championship [ESPN.com, 2009].

In this direction, we show in Figure 5.5 the evolution of the two situations that cause network changes, i.e., the number of transactions (new edges) and new players, coaches and teams (new nodes). First, we observe a high correlation between the number of new nodes and new edges for the years before the late seventies. Then, the number of new nodes stays practically the same while the number of new edges grows practically linearly. This suggests that the semantic reasons and implicit feedbacks associated with network changes for these two periods might be different due to, for instance, changes in regulations and player demands. The results presented in Section 5.6 reflect this observation.

5.4 Proposed Network Features

5.4.1 Preliminaries

Once defined the NBA Network and how it evolves over time, we are able to define and tackle the following problem:

Problem 2. FEATURE EXTRACTION. *Given a system in which the only information available is its historical network $G^y(V^y, E^y)$ for year y , which features can we extract from G^y that provide relevant information on its future behavior?*

²The salary cap is the maximum dollar amount teams can spend on player contracts.

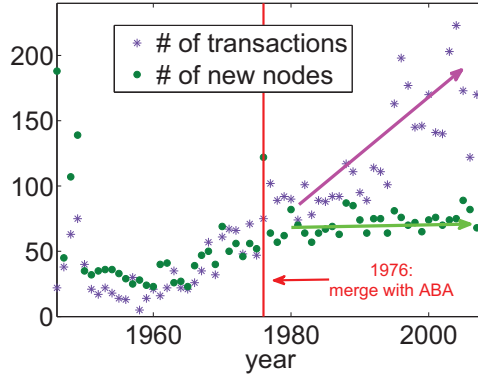


Figure 5.5. NBA network changes: the number of new nodes and transactions (new edges) over the years. Note that after 1976 there is much more mobility (transactions), with the number of new nodes being around 70 while the number of transactions keep on growing.

The features that can be extracted from G^y are exclusively network features that obviously do not contain any sort of explicit information about their nodes. Therefore, in order to solve Problem 2 considering the NBA dataset, we must rely on the implicit feedbacks that arise from network effects, which indicate if a team will have a good or bad performance in a year. Formally, each implicit feedback is a function $f_i(G^y, t^y) \Rightarrow \mathfrak{R}$ that receives as parameters the NBA network G^y for year y and the team node t^y in which we want to know its future performance in year $y + 1$. Moreover, each implicit feedback f_i is associated with a specific observable network effect that can be used as an implicit measure of interest, in our case, verifying if a team is likely to have a good or bad performance in the following season. As an example, a team that significantly increased its degree from G^{y-1} to G^y is a team that has significantly changed its roster and, as a consequence, might not have a good performance in the season y . On the other hand, if the degree between seasons is similar, the team is probably satisfied with its roster and might have a good performance.

In the following sections, we show our proposed network features. These features are based on the degree d_v^y and on the clustering coefficient [Newman, 2003]³ cc_v^y of the vertices v of the NBA Network $G^y(V^y, E^y)$, being the same metrics that were reported by Vaz de Melo et al. [2008a] as metrics that correlate with the performance of NBA teams. In Table 5.1, we show the summarization of these features and other relevant symbols.

³the clustering coefficient is the probability of two given neighbors of a certain node to be connected.

Table 5.1. Table of symbols.

Description	Symbol
node of team t in year y	t^y
node of player p in year y	p^y
node of coach c in year y	c^y
performance of team t in year y	r_t^y
first year of node v	$y0_v$
age of node v in year y	$a_v^y = y - y0_v$
roster of team t in year y	R_t^y
degree of node v in year y	d_v^y
clustering coefficient of node v in year y	cc_v^y
<i>Team Volatility</i>	$\Delta d_t^y = d_t^y - d_t^{y-\epsilon}$
<i>Roster Aggregate Volatility</i>	$\Sigma \Delta d_t^y = \sum_{\forall v \in R_t^y} \frac{d_v^y}{y - y0_v}$
<i>Team Inexperience</i>	$ixp_t^y = cc_t^y$
<i>Roster Aggregate Coherence</i>	$cc_t^y = avg(cc_v^y \times (y - y0_v)), \forall v \in R_t^y$
<i>Roster Size</i>	$s_t^y = \sum \forall p \in R_t^y$

5.4.2 Team Volatility

Our first feature is based on the well known quote “never change a winning team” by Sir Alf Ramsey, manager of the English national football team from 1963 to 1974. As we mentioned in Section 5.3.3, a player may leave a team when (i) he is not performing well, (ii) his salary is too high for the team or (iii) when he becomes a free agent and wishes to join another team which, according to his judgment, has more chances to win the championship [ESPN.com, 2009]. Since the reasons (ii) and (iii) are more related to high performance players and, as we showed in Section 5.3.2, we conjecture that most players have average box-score performance due to reason (i).

Thus, when the degree d_t of a team t is increasing constantly and at a high rate over the years, it probably means that t has not found a good roster yet. Therefore, a high degree increasing rate for a team node t before season y starts is an implicit feedback that this team will not perform well in y . Then, we define the feature *Team Volatility* Δd_t for team t in year y as

$$\Delta d_t^y = d_t^y - d_t^{y-\epsilon}, \quad (5.6)$$

where ϵ is a time window parameter that should be small, in order to capture the most recent behavior.

In Figure 5.6, we show the Spearman’s rank correlation ρ [Kendall and Gibbons, 1990]

between the *Team Volatility* Δd and the performance of the teams, with $\epsilon = 2$.⁴ The coefficient ρ measures how well the relationship between two variables can be described using a monotonic function, being appropriate to compare two ranks. The values of ρ vary from -1 , when the two ranks are completely opposite, to 1 , when the two ranks are the same. When $\rho \approx 0$, there is no relationship between the ranks. As we expected, we can observe that, exception made for the year of 1969, the correlation is always negative, with an average $\bar{\rho} = -0.52$. This indicates that the *Team Volatility* hurts the performance of the teams, i.e., the more a team hire and fire players and coaches, the worst for its future performance. Thus, it is reasonable to state that the *Team Volatility* feature is a good indicator of the team performance.

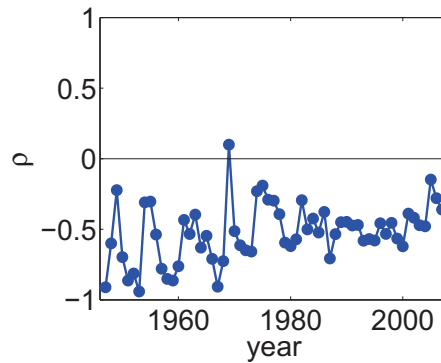


Figure 5.6. The *Team Volatility* hurts the performance of the teams, i.e., the more a team hires and fires players and coaches, the worst. Observe the regularly negative Spearman's rank correlation ρ between the *Team Volatility* Δd and the performance of the teams. Note that, differently than box-score statistics, our network-based features can capture such correlation.

5.4.3 Roster Aggregate Volatility

At the same time a team that is increasingly changing its roster is probably a team that is not achieving satisfactory results, a player who is constantly changing his team is probably a player who is not adapting. Thus, this kind of player is probably a player who will not provide any significant aid or performance improvement for his current and future teams. In the same way, this is also valid for coaches. Therefore, a team with a large number of members who have played for several teams is an implicit feedback that this team will not perform well in the season. Thus, in order to spot such teams we define the *Roster Aggregate*

⁴Other small values of ϵ were tested and the results are similar.

Volatility feature as

$$\Sigma\Delta d_t^y = \sum_{\forall v \in R_t^y} \frac{d_v^y}{y - y_{0_v}}, \quad (5.7)$$

where y_{0_v} is the year of the first season played by the team member v who can be a player or a coach. In summary, $\Sigma\Delta d_t^y$ is the sum of the degree increasing rates for all the team members in the roster R_t^y of team t in year y , counting from the first season of each member. Again, the use of the *Historical* propagation model is justified because we want to know the history of players and coaches who played with every team member $v^y \in t^y$.

In Figure 5.7, we show the Spearman's rank correlation ρ between the *Roster Aggregate Volatility* $\Sigma\Delta d$ and the performance of the teams. As we observe, the correlation is negative for the vast majority of the years, with an average $\bar{\rho} = -0.52$. Although this feature is similar to the *Team Volatility*, it has subtle but significant differences. For instance, a team that is changing its roster at a slow rate, according to the *Team Volatility* metric, is a likely team to have a good performance. However, if this team is hiring players that are changing their teams at a high rate, according to the *Roster Aggregate Volatility* metric, this team is likely to have a bad performance.

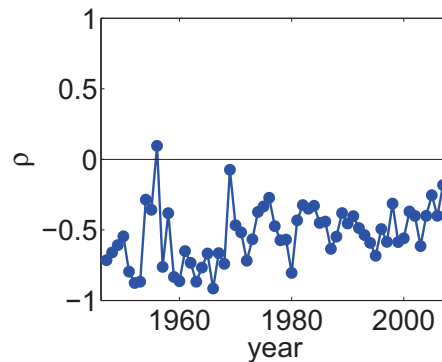


Figure 5.7. Again, note that the *Roster Aggregate Volatility* hurts the performance of the teams, i.e., the more a team hires players and coaches that are constantly changing their teams, the worse. The Spearman's rank correlation ρ between the *Roster Aggregate Volatility* $\Sigma\Delta d$ and the performance of the teams is consistently negative. This is another observation that cannot be drawn from usual box-score statistics.

5.4.4 Team Inexperience

It is well known that when a new team arrives at a determined sport league, the general expectation is that this team does not perform well [Reheuser, 2010]. One of the reasons for that is that the team managers may not have the proper number of connections and notoriety

to raise a significant number of funds from sponsors to hire well known good performance players. Then, we expect that the experienced teams have better performance than the newly arrived inexperienced teams.

The usual and most natural way to describe a team t experience in year y is by its age a_t^y , i.e., the number of years a team has played in the NBA. However, the amount of experience a team acquires during the years is not a linear function, i.e., the amount of experience a team gets after playing its first season is probably significantly higher than the amount it will get by playing its 20th season. Thus, as a measure of experience we propose the clustering coefficient of a node that has a non-linear relationship with the age of the team (for more details, see the Appendix). Moreover, the clustering coefficient also indicates whether a team has not experimented a wide variety of players in its roster, which in most cases may suggest that this team is not a good team. Thus, we define the *Team Inexperience* feature as:

$$exp_t^y = cc_t^y. \quad (5.8)$$

Thus, when a team t joins the NBA in year y , $exp_t^y = cc_t^y = 1$ and, as t makes transactions and become more experienced, cc_t^y decreases. Despite being a natural intuitive network feature, we can see in Figure 5.8 that the rank correlation ρ between the *Team Inexperience* feature and the performance of the teams is not regular, averaging only -0.14 . However, a closer look indicates that the correlation fluctuates smoothly, what suggests that the temporal scenario plays an important role in this feature, with some subsequent years the correlation being significant negative, and some subsequent years being close to zero. Moreover, the median performance of inexperienced teams, i.e., teams that have played a season with a team experience $exp_t^y > 0.95$, is 21%. This indicates that, in general, their performances are significantly below average, that is 50%. However, as we have shown in Figure B.4 (see the Appendix) and in [Vaz de Melo et al., 2008a], the clustering coefficient carries valuable information about the teams and players who can be jointly used with other information to predict the future behavior of teams and players. This is shown in the description of our next feature, which uses the clustering coefficient and is the one that shows the highest correlation with the performance of the teams.

5.4.5 Roster Aggregate Coherence

It is common sense among sports analysts that a good chemistry among players is essential for the success of a team. The so called chemistry involves a good relationship among the players and also a good knowledge on the future moves and plays of each teammate. Moreover, it is also important that the coach knows his players well, being able to request from

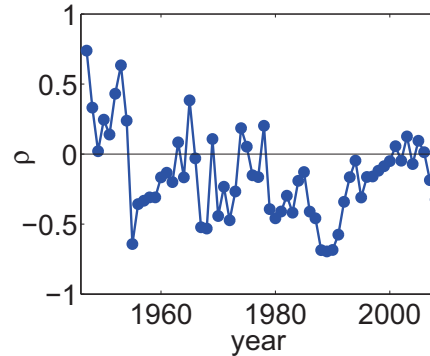


Figure 5.8. There is no clear correlation between the *Team Inexperience* and the performance of the teams. However, a closer look indicates that the correlation fluctuates smoothly, what suggests that the temporal scenario plays an important role in this feature.

them what he knows to be their best techniques. Thus, since the best way to build chemistry among players and the coach is through time, we define the metric *Roster Aggregate Coherence* \overline{cc}_t^y for team t in year y as

$$\overline{cc}_t^y = \text{avg}(cc_v^y \times (y - y0_v)), \forall v \in R_t^y, \quad (5.9)$$

that is the average of the clustering coefficients cc_v^y for every team member v_t^y who is in team t in year y multiplied by the current member's age $a_v^y = y - y0_v$. In summary, a high value of \overline{cc}_t^y means that the team roster is playing together for a substantial amount of time and few changes occurred. On the other hand, a low value of \overline{cc}_t^y means that the roster faced a high number of changes in recent time.

In Figure 5.9, we show the Spearman's rank correlation ρ between the *Roster Aggregate Coherence* \overline{cc}_t^y and the performance of the teams. As we expected, we observe that the correlation is positive for every year, with an average $\bar{\rho} = 0.55$, the highest among all the proposed features. This indicates that a high coherence and rapport among the players and the coach is a good indicator that a team will succeed. Again, this is a result that cannot be extracted from the usual box-score statistics, showing the usefulness of network effects.

5.4.6 Roster Size

Finally, we present the last feature we extract from the NBA Network, that is the *Roster Size*. The size of the roster of a team t in year y is the number of players playing (or have contracts) for t in y . Given that the salary cap⁵ is present in the NBA regulation since the

⁵The salary cap is the total amount of money the teams can spend on players' salaries.

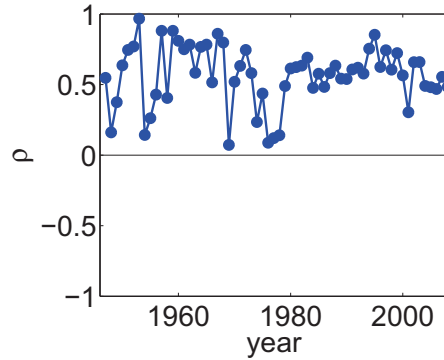


Figure 5.9. A high coherence and rapport among players and the coach is a good indicator that a team will succeed. Note that the Spearman's rank correlation ρ between the *Roster Aggregate Coherence* cc_t^y and the performance of the teams is regularly positive.

first season [nba.com, 2008], teams with a high number of players in their rosters tend, on average, to pay less to their players than teams with fewer players. Thus, it is expected that teams with less players have more valuable players, i.e., stars, since these players usually demand higher salaries. From this, we define the *Roster Size* feature s_t^y for team t in year y as

$$s_t^y = \sum \forall p^y \in R_t^y. \quad (5.10)$$

In Figure 5.10, we show the Spearman's rank correlation ρ between the *Roster Size* s_t^y and the performance of the teams. We observe that the correlation is negative for every year except the years of 1956 and 1958, with an average $\bar{\rho} = -0.41$. Thus, the larger the roster size, the worse for the team. This also enables us to consider the *Roster Size* feature as a good indicator of the teams performance.

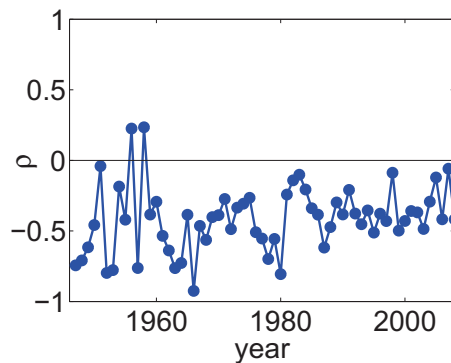


Figure 5.10. The larger the roster size, the worse for the team. The Spearman's rank correlation ρ between the *Roster Size* s_t^y and the performance of the teams is consistently negative.

For an analysis on the independence of the features presented in this section, please see the Appendix.

5.5 Network-based Prediction Models

5.5.1 Problem Definition

Once defined the features to be extracted from the NBA Network, we are able to tackle the main problem of this chapter:

Problem 3. RANK PREDICTION. *Given a NBA Network $G^y(V^y, E^y)$ for year y , predict the performance r_t^{y+1} of every team t in year $y + 1$?*

In Section 5.5.2, we present the *Network-based Forecaster (NetFor)*, which is our solution for Problem 3 and the main contribution of this chapter. The NetFor is a prediction model that uses only network features to predict the behavior of the teams. Also, in Section 5.5.3, we propose the *Network-based Plus Yesterday Forecaster (NetForY)*, which is a prediction model that uses both the network features and the information on the previous performances of the teams.

5.5.2 NetFor

We begin to solve Problem 3 by defining the basic operation of a prediction model M . First, we determine that a prediction model M_A has a function $F_A(t_y)$ to calculate a prediction score $\Pi_t^y = F_A(t_y)$ for each team t in every year y , where Π_t^y measures the likelihood of team t to perform well in year $y + 1$. Thus, the descending order of the Π_t^y values for every team t in year y gives the predicted rank of the teams for year $y + 1$, where the team t^* with the highest prediction score $\Pi_{t^*}^y$ is the most likely team to win the championship in year $y + 1$.

Thus, we propose as a solution to Problem 3 the *Network-based Forecaster (NetFor)* M_{NetFor} , which is based solely on network features. The function F_{NetFor} that calculates the Π_t^y value for each team t in year y is given by the features showed in the previous section. We showed that the *Team Volatility* Δd_t^y , the *Roster Aggregate Volatility* $\Sigma \Delta d_t^y$, the *Team Inexperience* ixp_t^y , the *Roster Aggregate Coherence* \overline{cc}_t^y and the *Team Volatility* s_t^y are good indicators of how well a team t will perform in a season y . Thus, we simply gather these five features in the function F_{NetFor} to compute the prediction score Π_t^y of each team t in year

y , in a way that

$$\begin{aligned}
\Pi_t^y &= F_{NetFor}(t^y, S) \\
&= \Delta d_t^{y-w1} \times \Sigma \Delta d_t^{y-w2} \times i x p_t^{y w3} \times \overline{c c_t^y}^{w4} \times s_t^{y-w5} \\
&= -w1 \times \log(\Delta d_t^y) - w2 \times \log(\Sigma \Delta d_t^y) - w3 \\
&\quad \times \log(i x p_t^y) + w4 \times \log(\overline{c c_t^y}) - w5 \times \log(s_t^y),
\end{aligned} \tag{5.11}$$

where the set of parameters $S = \{w1, w2, w3, w4, w5\}$ are real values that weigh the features in which they are associated with. The sign of the parameter is negative if the correlation is negative and positive if the correlation is positive. We are not using a plain linear model because the scale of the feature values may differ in orders of magnitude, what may make the size of the parameter space impractical. A plain multiplicative model removes from the parameters any dependency with the scale of the features they represent. In this case, the parameter values only measure, quantitatively, how relevant a feature is compared to another one. A linear alternative to the multiplicative model is the liner model using the logarithm of the feature values, which can be derived directly from the multiplicative model, as described in Equation 5.11.

In order to assign values to the feature parameters $w1, w2, w3, w4$ and $w5$, we propose a simple method. Since we want to predict the performance of the teams for year y , we can use all the information from the previous years $y - 1, y - 2, \dots, y - W_Y$ as the training set used to determine a good set of values $S_i^y = \{w1_i^y, w2_i^y, w3_i^y, w4_i^y, w5_i^y\}$ for the feature parameters. Thus, for a prediction for year y , our training set is a window of the W_Y previous seasons $y - 1, y - 2, \dots, y - W_Y$ and we use it to find a set S_*^y that gives the best results for these W_Y years.

Before explaining what is a good result and how we find the best ones, we list two classes of people that could directly benefit from a sports prediction model:

1. *Sports Analyst: the whole rank is important.* He/she wants to know how all teams will perform, i.e., he/she wants to know all the future team performance s;
2. *Gambler: only the top predicted team t^* is important.* He/she has only one bet for which team will be the champion, i.e., he/she only wants that the top team predicted by the model is also the champion.

Given these classes, one could use as the evaluation metric to find S_*^y , for instance, the Spearman's rank correlation coefficient to satisfy the *Sports Analyst* class or the number of times the prediction model selects the best performance team to satisfy the *Gambler* class. However, for simplicity, from now on we will use the Weighted Spearman's rank correlation ρ_W [da Costa and Soares, 2004] as our main evaluation metric for determining how good a

parameter set S is for a training set. The ρ_W weighs the distance between two ranks using a linear function of those ranks, giving more importance to higher ranks than lower ones, satisfying the two classes of people described above. Moreover, according to da Costa and Soares [2004], the main application for the proposed rank correlation coefficient is in the evaluation of rank prediction methods, which are mostly applied to recommendation systems, information retrieval, stock trading support and, in our case, sports outcome prediction.

Thus, we use the Weighted Spearman's rank correlation ρ_W to determine how good is a set of feature parameter values S . In order to find the quasi-optimal set of values S_*^y for the parameters, we search the parameter space for a set of values $S_*^y = \{w1_*^y, w2_*^y, w3_*^y, w4_*^y, w5_*^y\}$ that maximizes the value of ρ_W for the years $y - 1, y - 2, \dots, y - W_Y$ when we want to predict the teams' performance in year y , in a way that

$$S_*^y = \arg \max_{S^y} \left(\sum_{i=1}^{W_Y} \rho_W(r_t^{y-i}, f_{NetFor}(t^{y-i}, S^y)) \right). \quad (5.12)$$

In our case, since the parameter values only measure the relative temporal importance from one feature to another, we can search for parameter values in a reduced parameter space. Thus, we execute a linear local maxima search in the parameter space using the coordinate ascent optimization algorithm [Fessler and Hero, 1994] to find the quasi-optimal set S_*^y for every year $y - 1, y - 2, \dots, y - W_Y$ of our dataset. Each parameter w_i starts with the value 0.1 and linearly grows by 0.1 until the local maxima point is found. We emphasize that we tried different parameter spaces and different optimization algorithms and the results are similar.

5.5.3 NetForY

One advantage of the NetFor is that it is entirely based on implicit measures, which are generally thought to be less accurate than explicit measures [Nichols, 1998], but as large quantities of implicit data can be gathered at no extra cost to the user, they are attractive alternatives. Moreover, implicit measures can be combined with explicit data to obtain a more accurate representation of the analyzed system [Kelly and Teevan, 2003]. Thus, in this direction, we propose the *Network-based Plus Yesterday Forecaster (NetForY)*.

The NetForY is a simple extension of the NetFor, which together with the network features described in Section 5.4, also uses the information on the previous performance of the teams. Considering that the network features inform the amount of changes a team made in its roster in a non-trivial way, they can also indicate, using the previous performance information, whether a team will maintain its performance. For instance, if a team had a good performance in year y and the network features indicate that the network changes made in its roster are not significant, then it is very likely that this team will continue to have a good

performance in year $y + 1$. Thus, we simply gather the five network features used in the NetFor together with the previous performance of the teams r_t^{y-1} in the function $F_{NetForY}$ to compute the prediction score Π_t^y of each team t in year y , in a way that

$$\begin{aligned}\Pi_t^y &= F_{NetForY}(t^y) \\ &= -w1 \times \log(\Delta d_t^y) - w2 \times \log(\Sigma \Delta d_t^y) \\ &\quad + w3 \times \log(ixp_t^y) + w4 \times \log(\overline{cc}_t^y) \\ &\quad - w5 \times \log(s_t^y) + w6 \times \log(1 + r_t^{y-1}),\end{aligned}\tag{5.13}$$

where $F_{NetForY}(t^y)$ is identical to $F_{NetFor}(t^y)$ with the addition of the extra term $w6 \times \log(1 + r_t^{y-1})$. Moreover, the method for assigning values to the parameters $w1, w2, w3, w4, w5$ and now $w6$ is the same as the one described in the previous section. To see the parameter values and how they change over time for both network-based prediction models, see the Appendix.

5.6 Results and Validation

In this section, we describe the results of our proposed network-based models. In Section 5.6.1, we show the individual results of our models for metrics that are based on the two classes of people who were described in Section 5.5.2. Moreover, in Section 5.6.2, we compare the models with other models based on explicit descriptive statistics. For the W_Y parameter, we used a fixed sliding window of 10 years, i.e., $W_Y = 10$. We tried other values for W_Y and also the exponential weighted window, which gives more weight for more recent years, and the results are similar. For a W_Y sensitivity analysis, see the Appendix.

5.6.1 Results

As we mentioned earlier in this section, the basic evaluation metrics are targeted to the two classes of people described in Section 5.5.2 that can directly benefit from a NBA prediction model. First, we address the *Sports Analyst* class by showing, in Figure 5.11, the Spearman's rank correlation ρ^y between the performance r_t^y and the prediction score Π_t^y of every team t in year y . We observe a significant positive rank correlation for either the NetFor and the NetForY, with the NetForY performing better, on average. The average correlation $\bar{\rho}$ for the NetForY is 0.68 and for the NetFor is 0.59. We also performed, for both models, a linear regression for every pair (y, ρ^y) and we verified that the slope for both regressions is 0. This indicates that there is no particular trend for the performance of the models over time.

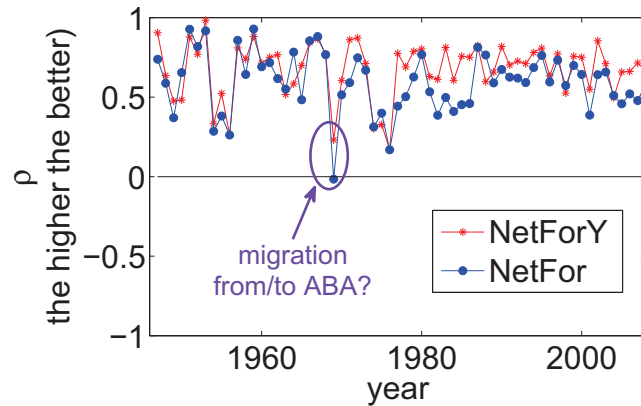


Figure 5.11. The proposed network-based models had good performances in predicting the whole rank of the teams. Observe the Spearman's rank correlation coefficient ρ between Π_t^y and r_t^y . The average correlation $\bar{\rho}$ for the NetForY is 0.68 and for the NetFor is 0.59, significative high values.

Second, we analyze how the network-based models perform according to the interests of the *Gambler* class. For this, we verify the performance $r_{t_1}^y$ of the team t_1^y that got the highest prediction score $\Pi_{t_1}^y$ in year y . A perfect prediction model gives, for every year y , the highest prediction score to the team t_*^y that was the team with the best performance in year y , i.e., $\Pi_{t_1}^y = \Pi_{t_*}^y$. Thus, we verify the percentage $hit_\%$ of times t_1^y was the actual best performance team t_*^y and also the average performance \bar{r} of the teams t_1^y selected by the model.

In Figure 5.12, we show the histogram of the performance of the teams t_1^y selected by the model over the 62 years of our dataset $1947 \leq y \leq 2008$. As an example, in the histogram, the number of teams with performance 80% that were selected by the NetFor is 7, meaning that in 7 years the model selected as a t_1^y a team with performance greater or equal to 80% and lower than 90%. Thus, we observe that the vast majority of the teams selected by the models are concentrated in the right side of the graph, indicating that the models generally select good performance teams as the most likely team to perform well in the following season. Moreover, for the NetFor, $hit_\% = 35\%$ of the teams selected by the NetFor model are the actual best performance teams and $\approx 68\%$ had, at least, a performance higher than 80%. On the other hand, the NetForY also selected $hit_\% = 35\%$ of the times the actual best performance team but, however, the model is more stable, selecting only once a team with a performance lower than 50% and with 90% of the teams t_1^y having a performance higher than 80%. The average performance \bar{r} of the teams selected by the NetFor is $\bar{r} \approx 83\%$ and by the NetForY is $\bar{r} \approx 89\%$.

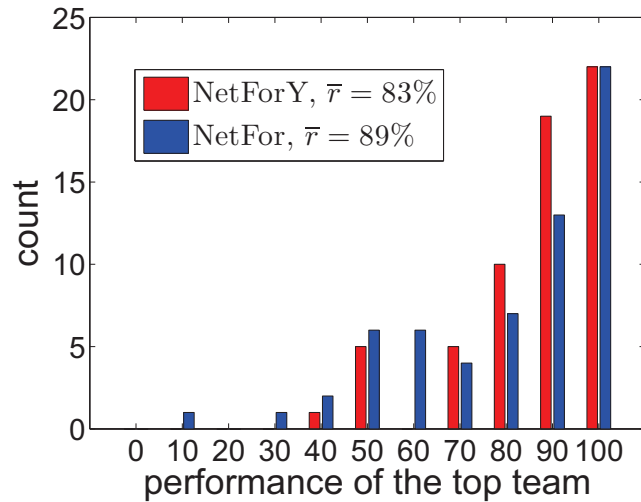


Figure 5.12. The network-based models consistently indicate a high performance team as their top team. They correctly predicted the top performance team $\approx 35\%$ of the times. Note that the higher the count values at the right at the distribution, the better.

5.6.2 Comparison

In this section we compare our network-based models with other prediction models, using the two metrics described in the previous section, the average Spearman’s rank correlation coefficient $\bar{\rho}$ and the average performance \bar{r} . Before presenting the results, we describe the models we use in the comparison.

“Yesterday”. This is currently the state of the art for forecasting team sports leagues in general, as we mentioned in Section 5.2. The “Yesterday” states that a teams performance in year y will be the same as its performance in year $y - 1$. Thus, the prediction score Π_t^y of team t in year y is simply the performance r_t^{y-1} of team t in year $y - 1$.

Aggregated Box-Score (ABS) Model. This model uses the aggregated value of the three main box-score statistics of team t in year $y-1$: points, rebounds and assists. Therefore, the prediction score Π_t^y of team t in year y is $\text{points}(t^{y-1}) + \text{rebounds}(t^{y-1}) + \text{assists}(t^{y-1})$. It is important to point out that when we use more sophisticated formulas, such as the APBRmetrics team efficiency difference rating [APBRmetrics, 2011], the prediction factor becomes significantly correlated to the prediction factor of the “Yesterday” model, giving similar results to it.

Efficiency-based models. We introduce two efficiency-based models, the Eff-1 and Eff-5 model. These models use solely the box-score efficiency of the players in the year $y - 1$ to calculate the prediction score of the teams in the year y . The only difference between these models is related to the number of players each one uses to compute the prediction score Π_t^y

of team t in year y . In the Eff-1 model, the prediction score Π_t^y of team t in year y is the highest efficiency of a player $p \in t^y$, and in the Eff-5 model is the average of the five highest efficiencies of players $p_j \in t^y$. Moreover, when $y < 1973$, we compute the efficiency of a player as the sum of his points, assists and rebounds achieved in a period divided by the total number of games he played in this period. When $y \geq 1973$, we compute the efficiency by using the NBA Efficiency Per Minute (EPM) formula [Paulsen, 2006], which can only be computed using more detailed box-score statistics, which were not recorded before 1973.

Adapted Page’s model. Page et al. [2007] used the 1996-1997 NBA season data to propose the use of Bayesian hierarchical modeling and box-score statistics to determine how each player position needs to perform in order to his team to be successful. He showed the most important box-score statistics for each player position in order to improve the team performance. Thus, we adapt his model, which predicts the winner game by game, to a season prediction model. We use the player’s data from year $y - 1$ in the equation of the model described in [Page et al., 2007], using the same parameters, to compute the prediction score Π_t^y of team t in year y . In this way, we expect that a team in year y that has more players that had high values of a box-score statistics in year $y - 1$ that is favorable to their respective positions, according to Page’s model, will be more likely to be successful in year y .

As shown in Figure 5.5, the network evolutionary behavior differs before and after the year of 1976. Thus, we divided the results into three blocks: (i) before the year of 1976, (ii) after the year of 1976 and (iii) the overall results, between 1947 and 2008. This division is also useful for other two reasons. First, after the year of 1973, more box-score statistics were recorded after each game, such as blocks, steals and offensive and defensive rebounds, which may collaborate to measure the efficiency of a player more accurately, by using the NBA EPM metric, for instance. Thus, this probably helps the efficiency-based models. More importantly, the Page’s model uses these box-score statistics to evaluate which team has the highest probability to win a match, and because of this, we can only evaluate the Page’s model after the year of 1973. Second, the labor relationships between players and teams – this is exactly what the edges of our networks measure – were significantly different [Bradley, 2009] before the year of 1976 from nowadays. For example, before the year 1976, players contracts had in it the option clause [Looney, 1976] that practically bounded a player to his team eternally, with the player being released to play for another team only when his team let him do so. This means that the implicit feedbacks of a network change before and after 1976 may be significantly different.

In Table 5.2 we compare the models described in this section with our proposed network-based models. We observe that the network-based models are the ones with the best results. The NetForY has the best results for all metrics in all periods, sometimes tying

with the NetFor . In comparison with our the best competitor, the “*Yesterday*” , the NetFor performs better in the first period and worse in the second. This could be explained because, as we saw in this chapter, the implicit feedbacks behind network changes before the year of 1976 were much more clear and simple. In this period, a player would move from a team to another only if he was not being efficient to the team.

In general, the NetForY is always the best model and both the NetFor and the “*Yesterday*” perform similarly, what is a fascinating result, since the NetFor considers only network features. More specifically, the network-based models are the ones that can achieve the best results, 35% of the times, being the best choices for the *Gambler* class. Considering the *Sports Analyst* class, while the NetFor has a similar performance to the “*Yesterday*” , the NetForY achieved $\approx 14\%$ improvement, what is an impressive result in terms of prediction power. This shows the potential of combining implicit network features with explicit features.

Concerning the models that behold to box-score statistics to calculate their prediction scores, they all presented worse results than the network-based models. The adapted Page’s model, as expected, presented better results than the efficiency models and the ABS model, since it is significantly more sophisticated. However, the fact that the efficiency models improved their results after the year of 1973 is quite interesting. There are two possible reasons for that. First, more high box-score statistics players are arising in the NBA along the years and they are being more determinant to their team success. Second, the new box-score statistics that are being recorded through the years are improving the characterization of the performance of the players in a game and, consequently, their importance to their team success.

Model	year \leq 1976		year $>$ 1976		Overall		
	$\bar{\rho}$	\bar{r}	$\bar{\rho}$	\bar{r}	$\bar{\rho}$	\bar{r}	hit%
NetForY	0.66	88	0.69	90	0.68	89	35%
NetFor	<i>0.62</i>	88	0.57	79	<i>0.59</i>	83	35%
“Yesterday”	0.57	85	62	88	<i>0.59</i>	87	26%
ABS	0.44	78	0.43	74	0.43	76	18%
Page’s	-	-	0.52	80	0.52	80	21%
Eff-1	0.37	74	0.41	79	0.39	77	24%
Eff-5	0.40	71	0.42	74	0.41	72	18%

Table 5.2. Comparison among the Network Model and other prediction models. In **bold** the best result and in *italic* the runner up. Note that our proposed NetForY always achieved the best result. Moreover, the NetFor had a better performance than the “*Yesterday*” in the early years and in selecting the best performance team.

5.7 Generality

In the previous section, we showed that the proposed network-based prediction models have significant good results when compared to other approaches. Moreover, since the models are parsimonious, i.e., they require a single parameter⁶, and do not rely on any particular box-score statistics, they are ready to be used in any system that share similar characteristics with the NBA. Thus, in this section, we verify the performance of the network-based models in the network formed from the North American Major League Baseball (MLB) dataset using the same value for the W_Y parameter, i.e., $W_Y = 10$.

We use the MLB dataset that is publicly available in Sean Lahman’s Baseball Archive site [Lahman, 2008]. It contains a huge amount of information about all the teams and players from 1871 to 2009, a total of 139 years of data! In our case, the only information we need is the roster and the ranking of each team per year. With this, we are able to construct the MLB networks and also verify the performance of our proposed network-based prediction models, as we did for the NBA dataset.

In Figure 5.13, we show the average ρ between Π_t^y and r_t^y for three consecutive years⁷ for the NetForY, the NetFor and the “Yesterday” models (the latter was our best competitor in the NBA dataset). In a first look, we observe that the behavior of the three models is very similar, with constant positive correlation values for the 138 years of the analysis. However, a closer look reveals that the NetForY model is robust to the oscillations of both the NetFor and the “Yesterday” models, having a more regular behavior. This can be verified by computing the global average correlation $\bar{\rho}$ that is ≈ 0.59 for the NetForY whereas is 0.52 for the NetFor and for the “Yesterday” models.

⁶the w_i parameters are calculated deterministically using this single parameter W_Y

⁷This was done to ease visualization.

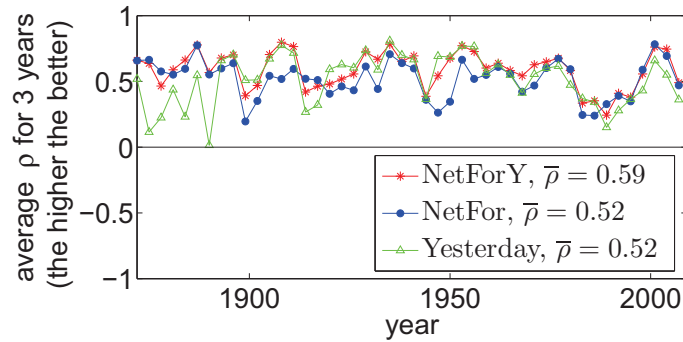


Figure 5.13. The average ρ between Π_t^y and r_t^y for three consecutive years. The global average correlation $\bar{\rho}$ for the NetForY is 0.59 and for the NetFor and “Yesterday” model is 0.52

The better performance of the NetForY can also be verified from Figure 5.14 that shows the difference $\Delta\rho = \rho_{M1} - \rho_{M2}$ between the ρ coefficients achieved by two models $M1$ and $M2$. We can observe that, while the NetFor and the “Yesterday” had similar performances, the NetForY had almost three times higher rank correlation coefficients ρ than the “Yesterday”. Finally, as verified when analyzing the NBA dataset, in the MLB dataset, while the “Yesterday” model presented an irregular behavior, the NetFor is the best model for the first years of the league, having constant high correlation values. This may indicate that for leagues where the rules and dynamics are not yet established, the NetFor may be the preferred choice for predicting the behavior of the teams.

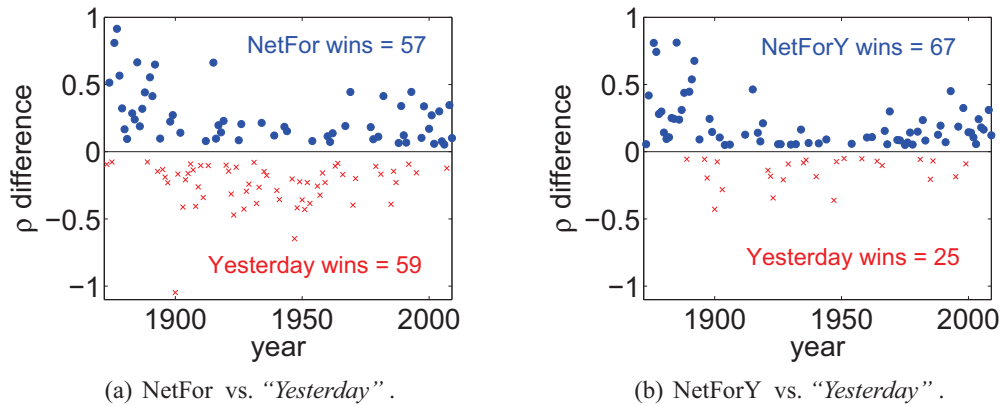


Figure 5.14. The difference $\Delta\rho = \rho_{M1} - \rho_{M2}$ between the ρ coefficients achieved by the two models $M1$ and $M2$.

Moreover, in Figure 5.15, we show the behavior of the network-based models and of the “Yesterday” model in identifying the best performance team. In Figure 5.15, we show

the histogram of the performance of the teams t_1^y selected by the model over the 138 years of our dataset $1872 \leq y \leq 2009$. Again, we observe that the NetForY is the one that presents the best results, with an average performance $\bar{r} = 0.81$, while the NetFor and the “Yesterday” had again a similar performance, with $\bar{r} = 0.79$. Moreover, the NetForY could identify the best performance team, on average, once at every three years, with $hit\% = 33.3\%$, while the NetFor identified 26% of the times and the “Yesterday”, 29% of the times.

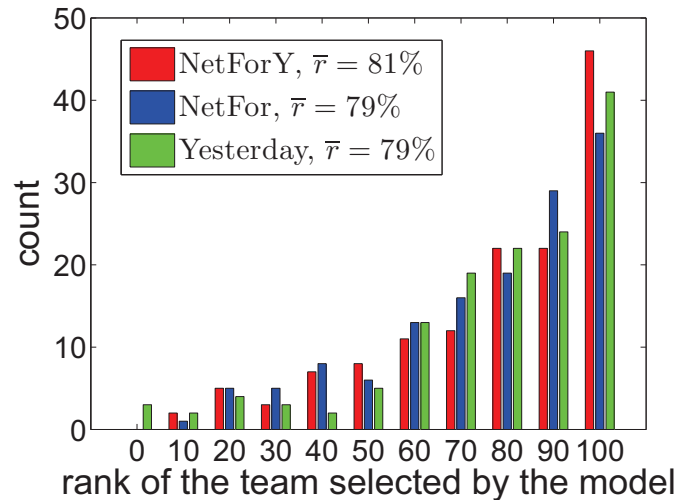


Figure 5.15. Performance of the network-based models and of the “Yesterday” model in identifying the best performance team.

In summary, we could observe that the network-based models presented good results even when applied to the MLB dataset that comprises 139 years of data and is related to a sport that was proved to be one of the most difficult to predict [Ben-Naim et al., 2007]. We believe that this test validates our proposed models and put the NetForY model as the state of the art for automated predictive models for general team sports leagues, since it consistently showed better results when compared to the “Yesterday” model that was to the best of our knowledge the best so far. More important is the finding that network effects have an enormous potential to describe the evolution of complex and dynamic social systems.

5.8 Final Remarks

In this chapter, we proposed the use of network implicit feedbacks to aid in the prediction of teams behavior in sports leagues. We proposed five temporal network features and, from them, we described two network-based models: the NetFor, which is entirely based on these

features, and the NetForY , which also considers the information on the previous performance of the teams. We analyzed the proposed models in two of the most popular professional sports leagues in North America, the National Basketball Association (NBA) and the Major League Baseball (MLB). In both leagues, the network-based models presented consistently good results, with the NetForY being the best model in all the considered metrics for both of the datasets when compared to other models in the literature.

In summary, the main contributions of this chapter are:

- We show that only a small fraction of players have significantly high box-score statistics values and, moreover, that acquiring these players does not guarantee a team performance improvement;
- We propose five general network features that are good indicators of the teams performance in sports leagues. For instance, we show that a high team volatility and roster size are not good for the team performance;
- We propose the NetFor and the NetForY network-based models to predict the behavior of teams in sports leagues. The models present surprisingly good results when compared to other approaches. The NetForY model presented better results than the current best model, with a $\approx 14\%$ accuracy improvement in predicting the next year rank;
- The network-based models are generic and may be applied to any team sports league, i.e., they do not rely on any particular box score statistics.

Chapter 6

Controlling Computer Networks

6.1 Introduction

Finally, we analyze DBCNs formed exclusively by computer devices. As we showed in [Vaz de Melo et al., 2011b], even in this type of network, agents can be modeled as social nodes with decision capabilities. The main difference between this type of networks and the previously analyzed ones is that in computer networks the agents do not make irrational decisions due to the “trembling hand” effect [Fudenberg and Tirole, 1991]. This means that all decisions performed by the agents have the goal to maximize their utility and nothing else, acting in a pure selfish way, and irrational decisions made by jitter or similar causes do not exist.

We consider a scenario where distinct Wireless Sensor Networks (WSNs) [Akyildiz et al., 2002] with different owners are deployed at the same place. In this scenario, there is the possibility that a sensor node of a network cooperates with a sensor network of another network. When two WSNs, installed at the same place, share their sensor nodes in the execution of one or more activities in a intelligent way, both networks may improve their operabilities and perform their activities in a more efficient way. Despite being obvious and simple, this idea brings with it many implications that hinder cooperation between the networks. Whereas a WSN has a rational and selfish character, it will only cooperate with another WSN if it provides services that justify the cooperation.

In this chapter, we tackle the three aspects of DBCNs we analyze in this thesis: modelling, predicting and controlling. First, we model the problem of cooperation among different WSNs using game theory. Game theory is an interesting technique to model conflict situations among two or more rational and selfish agents [Fudenberg and Tirole, 1991; Nisan et al., 2007; Luce and Raiffa, 1957]. In computer networks, since the decisions and the metrics of utility, e.g., throughput and latency, are computationally well defined, game theory

Symbol	Description
WSN	Wireless Sensor Network
VCB	Our proposed Virtual Cooperation Bond
GTC	Our proposed Give to Conquer strategy
N_i	Network i
n_i^j	sensor node j of network i
$G = (N, E)$	graph representing the sensor nodes of the set of networks N and their links E
$e_i(n^j, n^k)$	an edge between nodes n^j and n^k carrying data of network N_i
$w(e_i(n^j, n^k))$	energy cost of the edge between nodes n^j and n^k
f_{rx}	energy cost for receiving a message
f_{tx}	energy cost for transmitting a message at 1 unit of distance
$d(n^j, n^k)$	distance between nodes n^j and n^k
α	path loss exponent
c_i^j	cost that each node n_i^j incurs for its network N_i
Π_i	payoff of network N_i
ϕ	abstract value of the data that is transmitted

Table 6.1. Table of symbols.

can be a powerful tool to model the behavior of the social agents. After the game-theoretic model is well defined, the behavior of the agents is well predicted. From this knowledge, we can design incentive mechanisms to control their behaviors.

The main contributions of this chapter are:

- We present a game-theoretic formulation for the problem of cooperation among different WSNs deployed in the same location;
- We show that the only solution for the cooperation is a protocol that enables a joint strategy change by the nodes;
- We propose the *Virtual Cooperation Bond* (VCB) protocol, that is able to make the networks cooperate if and only if the cooperation is beneficial to them;

The rest of this chapter is organized as follows. In Section 6.2, we explain our motivations by solving the problem of cooperation among different networks and, later, in Section 6.3, we describe the previous approaches to solve this problem. In Section 6.4, we show our game theoretic formulation of this problem and, in Section 6.5, we show the solution of this game by proposing the *Virtual Cooperation Bond* (VCB). In Section 6.6, we show the protocol to establish the VCB and in Section 6.7 are described the simulation results. Finally, in Section 6.8, we present some comments. Table 6.1 presents the symbols used in this chapter.

6.2 Cooperation Among Different Networks

A Wireless Sensor Network (WSN) consists of spatially distributed autonomous devices that cooperatively monitor physical or environmental conditions, such as temperature, sound, luminosity, vibration, pressure, motion, and pollutants. The potential for observation and control of the real world allows WSNs to present themselves as a solution to a variety of monitoring and control applications, such as environment monitoring, biotechnology, industrial monitoring and control, public safety, transportation, and medical control. A WSN is comprised of compact and autonomous devices called sensor nodes, composed of processor, memory, battery, sensor devices and transceiver. A sensor network is the result of the fast convergence of three key technologies: microelectronics, wireless communication, and MEMS (Micro Electro-Mechanical Systems).

In a near future, we can expect to have WSNs deployed on the most different and varied places, like forests, volcanoes, seas, cities and deserts. Furthermore, a possibly forthcoming scenario will be distinct WSNs owned by different authorities working at the same place, serving as a supporting tool for a wide variety of applications. In the Amazon forest, for instance, we could have a WSN, owned by the government, deployed for detecting fires and another WSN, owned by a non-governmental organization, deployed for detecting the movement of specific species of animals.

Because of the environment in which a WSN is inserted, both size and production cost of a sensor node should be as small as possible and its battery replacement will be rarely possible. As a consequence, a sensor node has severe hardware constraints in terms of processing, memory, communication and energy capacity. In particular, depending on the sensing application, data communication can consume a significant amount of energy Pottie and Kaiser [2000]. Thus, in order to assure an efficient operability of the WSN, sensor nodes should cooperate with each other to improve the quality of the data and to reduce the energy consumption in the data communication process.

In this direction, when two WSNs, installed at the same place, share their sensor nodes in the execution of one or more activities in a profitable way, both networks may improve their capabilities and perform their activities in a more efficient way. Despite being obvious and simple, this idea brings several implications that hinder cooperation among the networks. A WSN has a rational and selfish character and will only cooperate with another sensor network if this association provides services that justify the cooperation. In Figure 6.1, we illustrate this scenario.

An interesting technique to model conflict situations among two or more rational and selfish agents is through the concepts of game theory Fudenberg and Tirole [1991];

Nisan et al. [2007]. Generally speaking, game theory is based on models that express the interaction among players by modeling them as rational and selfish agents in such a way that they act to maximize their own utility. This allows the analysis of existing algorithms and protocols for WSNs, as well as the design of equilibrium-inducing mechanisms that provide incentives for individual nodes to behave in socially-constructive ways. Game theory defines mechanisms for players to choose the best available action, but at the same time it provides a scenario where other players' utilities can also be maximized. By modeling the problem of cooperation among different WSNs as games, the networks' behaviors and actions can be analyzed in a formalized game structure, by which the theoretical achievements in game theory can be fully utilized. Each authority that governs its network wishes to both maximize its lifetime and its quality of service, and will only cooperate with another network if this cooperation brings benefits.

6.3 Related Work

In the literature, there are a few but significant proposals that studied this problem, and all of them present significant results that serve as base for our work. They usually use cooperation between different networks to reduce the distance that sensor nodes communicate, since the energy consumption varies exponentially with the distance at which the data is transmitted [Wieselthier et al., 2001], depending on the path loss exponent of the environment. Nevertheless, cooperation may be also used to improve other network functions, such as quality of the data by data aggregation.

The work of Felegyhazi et al. [2005] is, to our knowledge, the first one that addressed the problem of cooperation among different WSNs. In that work, the strategies of the owners of the networks are set if their sensor nodes forward messages coming from other networks and if they ask other networks to forward their messages. It is assumed that sensor nodes send messages periodically and synchronously to their respective sink nodes and these, in turn, send to their sensor nodes a bit telling if data collection was satisfactory. From this, sensor nodes control their strategies in order to minimize their energy consumption and maximize the data collection rate. In this model, the networks converge essentially to two equilibria, a non-cooperative in that no node provides and asks for services to another network, and a cooperative, which all nodes provide and ask for services of another network.

We showed in [Vaz de Melo et al., 2008c] that the problem of cooperation among different WSNs involves several parameters that can significantly influence the establishment of cooperation and its benefits. A solution that enables cooperation only when all of the sensor nodes cooperate is not practical, since it is very unlikely that two different WSNs

have the same configuration homogeneously through out the area they are deployed. Thus, an efficient practical solution should allow parts of the networks to cooperate where others do not. Then, in order to accomplish that, it is very important to have a distributed solution.

Moreover, in order to establish cooperation among sensor nodes of different WSNs, it is important that they know the costs and the benefits that will incur to them by cooperating. Min-You Wu [2005] showed that if the sensor nodes declare their cost to route messages, the networks are naturally encouraged to cooperate and to share their sensor nodes. Moreover, the authors showed that if the agent reveals its real cost for routing a packet, its benefits will be maximized.

Besides routing, Miller et al. [2005] consider the possibility of different WSNs to exchange different favors, such as routing, sensing, processing and data storage. Because of this, a stable solution and beneficial to both systems is only feasible if the owners of the networks sign a financial contract before the network deployment, which may be unpractical, since networks are deployed regularly and a control over it may not be possible.

Finally, Crosby and Pissinou [2007] demonstrate that cooperation is not evolutionary stable when the networks have mobile sensor nodes and are playing the iterated N-player prisoner's dilemma. However, they showed that in the case of stationary sensor nodes, there is some possibility for cooperation to emerge without any incentive when the same game is played. Our work goes in this direction, and in the next section, we model the problem of cooperation among different WSNs as the iterated N-player prisoner's dilemma and we propose a strategy to establish cooperation among the networks.

6.4 WSN Cooperation Game

In this section, we propose a new formulation for the problem of cooperation among different WSNs deployed at the same location. It is similar to the ones discussed earlier, but with this formulation we aim to guarantee cooperation in a distributed and ad-hoc fashion, for more realistic and dynamic scenarios. Let N be a set of m networks, N_1, \dots, N_m , and let n_0 be the sink node, which is shared by all networks. Each N_i network, $1 \leq i \leq m$, has a set of unique sensor nodes $N_i = \{n_i^1, n_i^2, \dots, n_i^{|N_i|}\}$ and the sink node n_0 is shared by all networks. We model the cooperation among the networks as a multigraph $G = (N, E)$, where N is the set of all networks and E is the set of edges between an arbitrary pair of sensor nodes. For the sake of simplicity, we model the traffic originating at each network in a unique way. Therefore, there will be a distinct edge for the collected data packets coming from each network and sent towards the sink. All edges that carry traffic from network N_i are represented by $e_i(n_j, n_k)$, where j and k are not necessarily different and $1 \leq j, k \leq m$. The

set of edges e_i represents the routing infrastructure responsible for carrying the traffic from network N_i . From a practical point of view, a communication link between a pair of nodes may represent a set of distinct edges.

For simplicity, we consider that the initial data collecting routing structure of each network is a routing spanning tree [Tanenbaum, 2002], but this is not a necessary condition, i.e., any routing structure can be used by the networks. We consider that the routing spanning tree generated is, for each network, the best tree that the network is able to generate following a defined QoS parameter, being optimal or not. Thus, each node n_i^u has at most a single outgoing edge $e_i(n_i^u, n_j^v)$ leaving n_i^u for each network N_j , i.e., n_i^u can forward messages from network N_j to at most one node. The set of edges that leaves n_i^u is denoted by E_i^u , $|E_i^u| \leq m$ and the weight $w(e_i(n_i^u, n_j^v)) > 0$ of edge $e_i(n_i^u, n_j^v)$ is independent of the network N_i , being given by the energy consumption model $w(e_i(n_i^u, n_j^v)) = f_{rx} + f_{tx} \times d(n_i^u, n_j^v)^\alpha$, where f_{rx} is the cost to receive a message, f_{tx} is the cost to send a message to a unit of distance d , $d(n_i^u, n_j^v)$ is the distance between n_i^u and n_j^v and α is the path loss exponent. We emphasize that the energy model w may be changed without invalidating this formulation.

The players in our game-theoretic formulation are the m networks and the strategy of network N_i is to set $S_i = \{E_i^u \forall n_i^u \in N_i\}$, i.e., each network defines to which sensor nodes its own sensor nodes will relay the networks' data collection messages. The cost in energy c_i^u that each node n_i^u incurs for its network is given by $c_i^u = \sum_{i=1}^m w(E_i^u)$, that is the sum of the weights of the edges that leaves node n_i^u . The utility function that maps the payoff Π_i of each network N_i , given its topology, is $\Pi_i = \sum_{\forall n_i^u \in N_i} (\phi - c_i^u)$, where ϕ is an abstract value of the data that is transmitted, where $\phi \gg c_i^u \forall n_i^u \in N_i$. As we mentioned earlier, it is natural that the value ϕ of the data be significantly higher than the cost to transmit it, since a WSN only exists to collect data, thus, without the data there is no network. The payoff Π_i captures, then, the data collection rate and the energy consumption in routing of network N_i , being determinant to define if the network cooperates or not with another network.

Initially, no sensor node n_i^u is cooperating with other networks besides its own network N_i . A network N_i , which is a rational and selfish player, should, then, change its strategy and make one or more of its sensor nodes to forward messages coming from other networks if and only if, it increases its payoff Π_i . However, if all the networks maintain their strategies and a network N_i changes its original strategy and make a sensor node n_i^u to cooperate and to forward messages from other networks, its cost c_i^u will increase, making the payoff Π_i to decrease. Thus, initially, the game is in Nash equilibrium because it is not possible for a network to increase its payoff changing its strategy if all other networks maintain theirs. This makes a network N_i to only be able to increase its payoff if it coordinates with another network N_j a joint change of strategies, $S_i \rightarrow S'_i, S_j \rightarrow S'_j$, so that S'_i benefits the network N_j and S'_j benefits the network N_i . In game theory, a game that allows two or more players

to coordinate their actions is called a cooperative game Fudenberg and Tirole [1991].

Thus, we can model this scenario as a cooperative, non-zero sum, two players game, where the game is played repeatedly during the networks N_i and N_j lifetimes. If both networks cooperate, their payoffs will be a “reward” R_i for network N_i and R_j for network N_j , that are greater than their non cooperative payoffs. Then, this game, that we call WSN-Cooperation game, is similar to the Iterated Prisoner’s Dilemma showed in Figure 6.2, having, as we mentioned, the “reward” payoff R , the “punishment” payoff P , the “temptation” payoff T and the “sucker’s payoff” S Helbing and Yu [2009]. In the ordered pairs we see the outcomes of the game given the strategies of each player, the networks N_i and N_j , with the first payoff given to the row player, N_i , and the second given to the column player, N_j . Given the inequality $T > R > P > S$, not to cooperate is always the best strategy, resulting in the worst outcome when both networks play it. However, as it happens in the Iterated Prisoner’s Dilemma game, if we manage to achieve the inequalities $R_i + R_j > T_i + S_j$ and $R_i + R_j > T_j + S_i$, then mutual cooperation returns the highest collective payoff and we can coordinate the networks to jointly cooperate and get better payoffs. Moreover, if we guarantee that if one defects, it will be punished by the other by breaking the cooperation agreement, the WSN-Cooperation game has a cooperation stable solution by Folk’s Theorem Fudenberg and Tirole [1991].

6.5 A Solution to the Game

A possible joint strategy change by two networks N_i and N_j is each one to create a cooperation edge that harms itself but that also benefits the other network, in a way that the benefits are greater than the losses. Formally, a sensor node n_i^u from N_i creates an outgoing edge $e_j(n_i^u, n_j^v)$ to a sensor node n_j^v from the other network N_j , implying that now it can relay messages from network N_j destined to sensor node n_j^q using edge $e_j(n_i^u, n_j^v)$. Simultaneously, the same process occurs in network N_j with a sensor node n_j^p creating an outgoing edge $e_i(n_j^p, n_i^q)$ to node n_i^q . If we guarantee that both networks N_i and N_j will use these cooperation edges $e_j(n_i^u, n_j^v)$ and $e_i(n_j^p, n_i^q)$ fairly, in a way that the benefits of using the external cooperation edge is greater, for both networks, than the costs of providing the internal cooperation edge, then cooperation may be established and be stable.

The cooperation strategy that the two edges $e_j(n_i^u, n_j^v)$ and $e_i(n_j^p, n_i^q)$ are created satisfying these restrictions will be called, from now on, the Give to Conquer (GTC) strategy. As the GTC strategy is repeated, creating different pairs of edges, the payoff of the networks involved in the process increases, since the process GTC guarantees that the benefits with the creation of the edges are greater than the losses. So, after all pairs of edges are created, the

payoff of the networks is the maximum possible that can be achieved with the cooperation using the GTC strategy, being, then, a Pareto Optimal solution Fudenberg and Tirole [1991], because no network is capable to increase its payoff without reducing the payoff of another.

The best way, according to the criteria of robustness and scalability, to make different WSNs to cooperate is making them interact with each other in a completely autonomous way. One must consider that they may change their topologies constantly and even that new networks may be deployed around them and also be part of the cooperation. Therefore, one should delegate the responsibility of executing the GTC strategy to the sensor nodes of the networks, which may locally control which sensors nodes are asking them for favors and to which ones they are asking favors. If a sensor node that is cooperating observes in its local environment that its network is losing more than it is benefiting with the cooperation, via a *watch dog* or other control policy, it can act immediately to stop cooperation in its neighborhood.

Figure 6.3 illustrates the scenario in which the GTC strategy may be executed in this way. Figure 6.3-a depicts two sensor nodes of a network N_1 , n_1^1 and n_1^2 , and two sensor nodes of another network N_2 , n_2^1 and n_2^2 , as well as the distances among them. Figure 6.3-b depicts the situation where there is no cooperation, with node n_1^1 sending a message to n_1^2 and node n_2^1 sending a message to n_2^2 . There are also illustrated the transmission costs C_{tx} , whereas $C_{tx} = d^\alpha$ and $\alpha = 2$. The total transmission cost of network N_1 is 81, and the total transmission cost of network N_2 is 64. In this scenario, $E_1^1 = \{e_1 n_1^1, n_1^2\}$, $E_1^2 = \emptyset$, $E_2^1 = \{e_2 n_2^1, n_2^2\}$ and $E_2^2 = \emptyset$. The networks payoffs are $\Pi_1 = \phi - (81f_{tx} + f_{rx})$ and $\Pi_2 = \phi - (64f_{tx} + f_{rx})$.

In Figure 6.3-c, a GTC strategy is executed and networks N_1 and N_2 start to cooperate. Initially, the GTC strategy creates two edges, the edge $e_1 n_1^1, n_1^2 \in E_1^1$ and edge $e_2 n_2^1, n_2^2 \in E_2^1$, indicating that the node $n_2^1 \in N_2$ cooperates with network N_1 and node $n_1^2 \in N_1$ cooperates with network N_2 . Then, the node n_1^1 , aware of the cooperation, removes the edge $e_1 n_1^1, n_1^2$ and adds $e_1 n_1^1, n_2^1$ to E_1^1 , because node n_2^1 is cooperating and will forward its messages. The same happens with node n_2^1 , that removes edge $e_2 n_2^1, n_2^2$ and creates edge $e_2 n_2^1, n_1^2$. Established the cooperation, the payoffs of the networks are $\Pi_1 = \phi - (61f_{tx} + 2f_{rx})$ and $\Pi_2 = \phi - (32f_{tx} + 2f_{rx})$, which are greater than when they were not cooperating, whereas in this work $f_{tx} \gg f_{rx}$.

The cooperation scenario shown in Figure 6.3-c is called Virtual Cooperation Bond (VCB). In a VCB, sensor nodes of a network act as team players and work jointly to reduce the energy consumption of their networks. One can see that the nodes n_1^2 increased its cost to help its network. In the VCB strategy, nodes involved are neighbors and can observe their respective actions. They can observe the amount of favors their network requests and

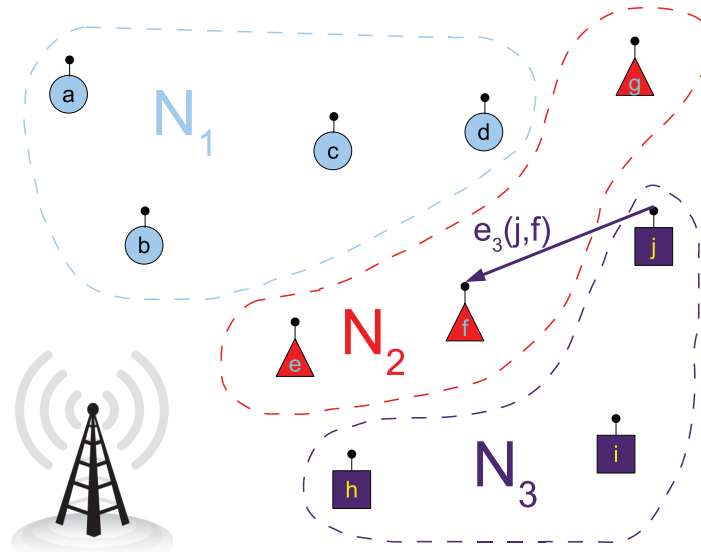


Figure 6.1. A scenario where different WSNs are deployed in the same location. We see three networks, N_1 , N_2 and N_3 , and a possible cooperation edge $e_3(j, f)$, where node f accepts data of network N_3 coming from sensor node j .

	Cooperate	Not Cooperate
Cooperate	P_i, P_j	S_i, T_j
Not Cooperate	T_i, S_j	R_i, R_j

Figure 6.2. Payoff matrix of the “WSN-Cooperation” game.

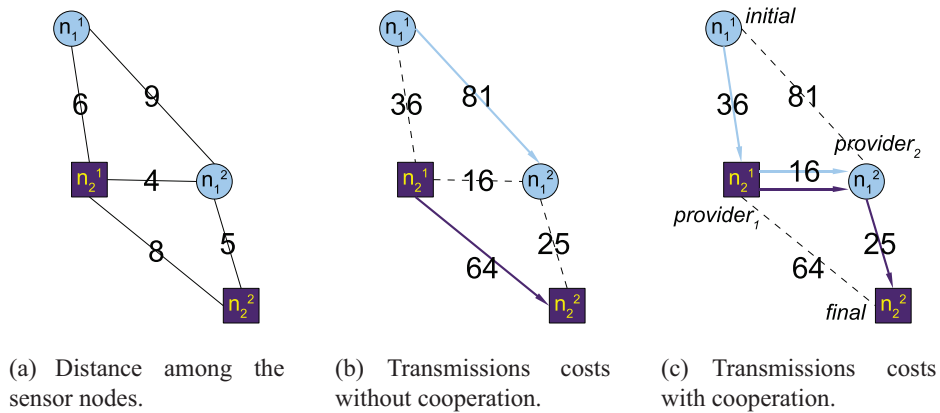


Figure 6.3. The Virtual Cooperation Bond: a feasible model to execute the GTC strategy ($w(e) = d^2$).

supplies, and that number is known by each node in the bond. Thus, the involved nodes control only the credit or debit that they have. This ensures the fairness in cooperation, i.e., no network provides favors at a higher rates that it receives, and vice versa, using only a upper limit Δ for the highest difference between favors provided and received that is allowed.

6.6 Establishing the VCB

Before explaining the method that describes how the cooperation is set, we make some considerations. First of all, we consider that every single node has a unique identifier and knows its coordinates. Second, we consider that there is a pre-established routing structure over the networks, in which every node has a target node that it sends or forwards its messages. Third, we consider that a sensor node is able to calculate the cost of a transmission, given the transmission distance.

Explained these considerations, we define roles to the nodes in the VCB, that are described in the topology of Figure 6.3-c. The role *initial* is played by the relaying node n_1^1 of Figure 6.3-c, which is the node in the VCB that provides no favors, but benefits from the cooperation by relaying its messages to a node from another network to forward. The node that receives the message from the node *initial* is the one that has the role of *provider₁*, which is the node n_2^1 in Figure 6.3-c, that besides providing favors to the node *initial*, also benefits from the cooperation by relaying its messages to the node that has the role of *provider₂*, which is from the other network and is represented by node n_1^2 in Figure 6.3-c. Node *provider₂* only provides favors, and forwards the messages from node *provider₁* to node of role *final* that, in the VCB, is only responsible for receiving these messages, and in Figure 6.3-c, it is represented by node n_2^2 .

In Figure 6.4, we describe the process for obtaining a VCB. The first step is to broadcast through the networks that wish to cooperate. Each node that wants to cooperate sends a broadcast message of type `WantToCooperate` to all its neighbors, i.e., nodes at its surroundings, informing its wish to cooperate and the value of its current transmission cost. Thus, each node that wants to cooperate receives, in each message of type `WantToCooperate`, the information about the costs of the routing edges of foreign networks, and can calculate locally if a VCB can be formed with its neighbors, guaranteeing that both networks will benefit with the cooperation. If so, it sends a `AskCooperation` message to the nodes in the VCB, informing what roles they should play in the VCB that could be formed. If all nodes agree with the assigned roles and they do not belong to other VCB, they send a `AcceptCooperation` message and cooperation between them is established. The node that sends the `AskCooperation` message is the node with the role

*provider*₁, since its location in the VCB is the only one that have direct access to all the other nodes in the VCB.

6.7 Numerical Results

In this section, we present the simulation results of the proposed VCB protocol. We consider a scenario where all sensors are deployed in an area of $100 \times 100 d^2$ with a flat topology. The communication range of each sensor starts with $30 d$ and may be reduced depending on the routing structure. During the 1000 seconds of simulation, events arrive in the network according to a Poisson Process with $\lambda_{Poisson}$ events per second. The default value of the path loss exponent is 4, and the number of nodes of each network is 50. All simulation results correspond to the arithmetic mean of r simulations, in which r provides a good confidence to the results. We use the *Network Simulator 3.21* to make these simulations.

The first result shows the behavior of VCB protocol when two networks cooperate and the network density varies from 50 to 200 nodes. In these results, the configuration of both networks is the same. Figure 6.5 shows the result comparing the energy spent on transmissions and receptions, as well as the percentage of cooperating nodes. We observe that the energy savings in transmissions, which is, as we mentioned, the highest energy consumer, is always between 20% and 30%. Moreover, as the networks' densities increase, the amount of cooperating nodes slightly reduces, together with the energy saving achieved in transmissions and receptions. This suggests that the VCB protocol is even more appropriate for scenarios which involve sparse networks.

Other factor that can influence the cooperation is the path loss exponent. Thus, the greater this exponent, the higher the costs for transmissions. Figure 6.6 shows the behavior of the VCB protocol when the path loss exponent varies. As expected, as the exponent grows, the higher is the energy economy provided by the VCB protocol, reaching 30% when $\alpha = 5$. This result suggests that the VCB protocol is highly recommended for noisy scenarios.

The VCB protocol is also able to establish cooperation when more than two networks are deployed in the same location, since its control is done locally. In Figure 6.7, we show the behavior of the VCB protocol when we increase the number of distinct networks deployed, all of them having the same configuration. We observe that the increase in the number of networks do not influence in the three analyzed metrics. The network has approximately 30% of nodes cooperating, and with this cooperation is possible to save more than 20% of energy in transmission, with approximately 35% increase in receptions. These results show

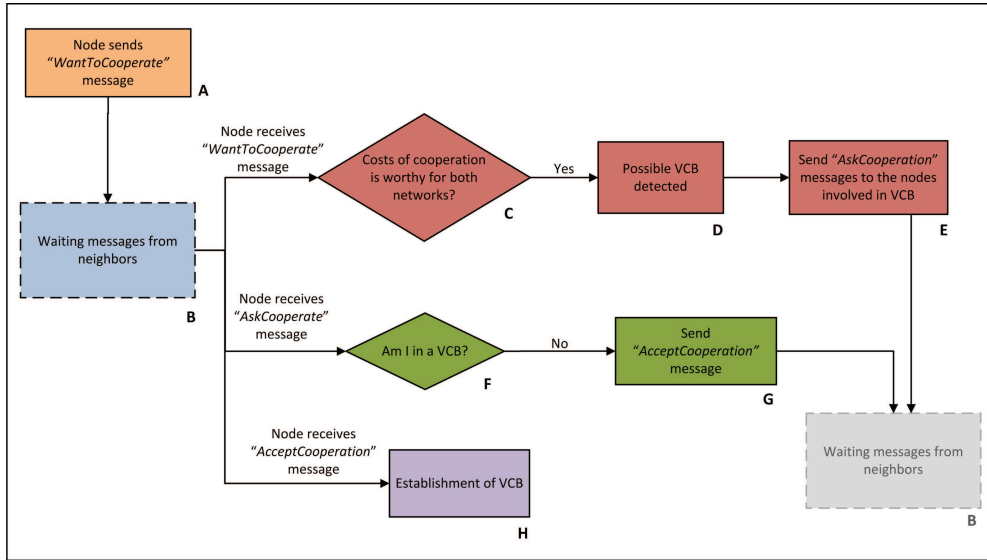


Figure 6.4. The algorithm describing the establishment of the VCB.

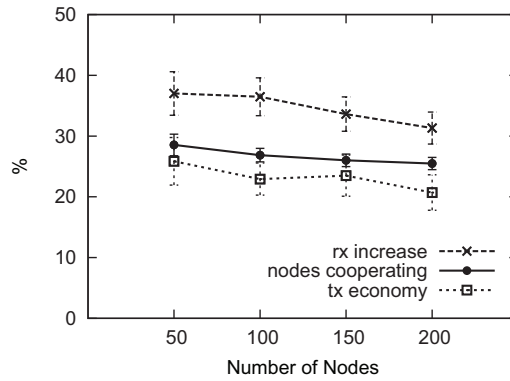


Figure 6.5. Cooperation results when the number of nodes is varied.

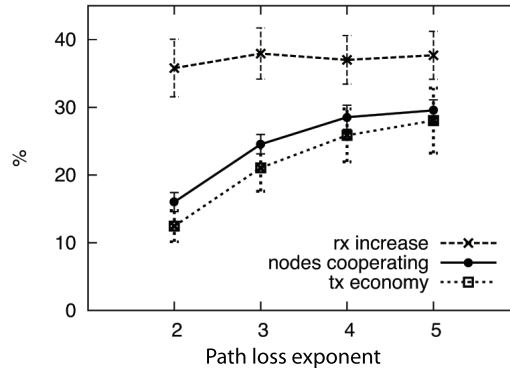


Figure 6.6. Cooperation results when path loss exponent is varied.

the proposed VCB protocol is scalable with the number of networks deployed.

The next scenario considers the case where the configurations of both networks are different. Thus, in the next results, the configuration of one network, N_1 , is kept constant whereas the configuration of the other network, N_2 , is varied. We analyze the performance of the VCB protocol in the network N_1 , which the configuration is kept constant. We compare the VCB protocol with a naive solution, which lets all the sensor nodes to cooperate indiscriminately.

First, we vary the density of N_2 from 10% to 50% of nodes of N_1 . We call the rate between the networks' densities δd . In Figure 6.8-a, we verify that the VCB protocol is able to save energy of N_1 even when $\delta d = 10\%$. On the other hand, the naive solution harms N_1 , making it consumes up to 40% more energy when cooperating. Second, we vary the number of events of N_2 according the parameter $\lambda_{Poisson}$, from 1 to 7. The $\delta\lambda$ is the difference among the number of events that are detected by the N_1 and N_2 . Thus, when the $\delta\lambda$ is 3, the network N_1 detected three times more events that the network N_2 . We see in the Figure 6.8-b that the VCB protocol keeps a constant behavior, saving the energy of N_1 even when the event rate of the network N_2 increased. In contrast, when the all the nodes cooperate and the $\delta\lambda$ is 7, the energy consumption of N_1 is 60% higher.

6.8 Final Remarks

In this chapter, we used game theory to model the behavior of the agents in the network formed when different WSNs are deployed in the same region. The viability of cooperation is not trivial and should be achieved and maintained in an autonomous way by the networks, given the scalability problem of WSNs. Thus, we proposed the Virtual Cooperation Bond (VCB) protocol as a distributed protocol that autonomously enables cooperation among different WSNs. First, we propose the *Give to Conquer* (GTC) strategy as a strategy of cooperation between two WSNs in a way that cooperation always brings benefits to both of the networks. Then, we propose the VCB as a model that implements the GTC strategy locally, in a way that cooperation may be autonomously achieved and maintained. To the best of our knowledge, our solution is the first one that is completely distributed and that addresses several practical issues of the problem of cooperation among different WSNs, such as cheating and networks with different configurations.

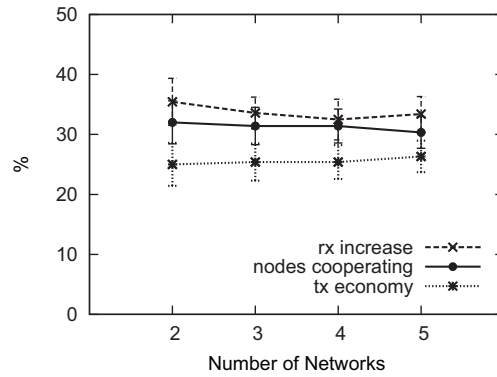
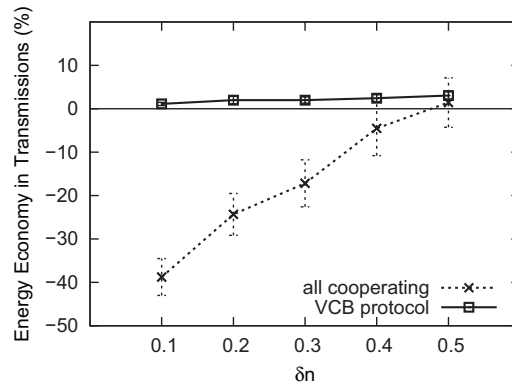
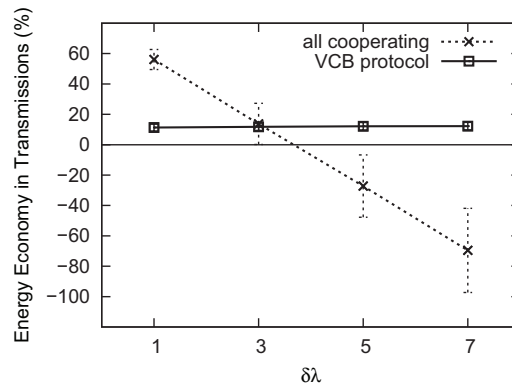


Figure 6.7. Cooperation results when the number of networks is varied.



(a) Networks have different densities.



(b) Networks have different data collection rates.

Figure 6.8. Cooperation results when the networks have different configurations.

Chapter 7

Conclusions and Future Work

In this thesis, we analyzed the behavior of rational agents in different Decision-based Complex Networks (DBCNs), which are described in Table 3.6. We showed that the decisions made by the agents of DBCNs have a crucial role in the evolution of the network. We also presented techniques to model, predict and control the evolution of the networks. First, we modeled the human behavior in communication networks formed from mobile phone records, e-mails and forum comments. Then, we analyzed how organizations and individuals interact with themselves in competitive networks, modeled from the North American National Basketball Association (NBA) and the Major League Baseball (MLB) data. We proposed a model to predict the behavior of the teams. Finally, we studied the scenario in which authorities of Wireless Sensor Networks have to decide if their networks should cooperate or not with others that are located in the same region. For this scenario, we proposed a cooperation control protocol. The contributions that were accomplished from these analysis are:

1. The TLAC distribution to explain the size of communication flows [Vaz de Melo et al., 2010];
2. The *MetaDist* as a summarization scheme for the collective behavior of users in communication networks [Vaz de Melo et al., 2010];
3. The discovery that the *MetaDist* remains the same over time, with very small fluctuations [Vaz de Melo et al., 2010];
4. Proposal of SFP model for the inter-event time between human activities [Vaz de Melo et al., 2011a];
5. Applications based on the TLAC and SFP models [Vaz de Melo et al., 2011a];

6. The network-approach for modeling competitive networks [Vaz de Melo et al., 2008a, 2012];
7. Network-based models for predicting the behavior of NBA teams [Vaz de Melo et al., 2008a, 2012];
8. A survey on the use of game theoretic solutions in WSNs [Vaz de Melo et al., 2011b];
9. Experimental projects that evaluate the impact of different parameters on the establishment of cooperation among different WSNs deployed in the same region [Vaz de Melo et al., 2008c];
10. The VCB protocol, that allows cooperation when different WSNs are deployed in the same region. [Vaz de Melo et al., 2009].

As future work for the modeling part (Chapter 4), we plan to mathematically prove that the SFP model generates a TLAC distribution. Moreover, we could focus on network effects for the TLAC model, that is, if two people talk to each other, what is the relationship between their TLAC parameters? A second promising direction is to check whether TLAC also fits well other modes of human (or computer) communications, like length of SMS messages and length of postings on FaceBook “walls”.

As future work for the predicting part (Chapter 5), we plan to apply network implicit feedback features to analyze other systems besides sports leagues, such as recommendation systems or business-oriented social networks, such as the *LinkedIn* network. We believe that the concepts presented in that chapter may be directly applied to other competitive systems and may lead to expressive results in different kinds of applications.

As future work for the network studied in the controlling part (Chapter 6), we intend to develop a detailed analytical model for the proposed protocol which predicts the implications of cooperation given the characteristics of the involved networks. Furthermore, we plan to analyze the performance of the proposed protocol for other metrics besides energy consumption. The proposed protocol naturally can save energy in transmissions, but it may serve as well in noisy scenarios, i.e., where the percentage of packet loss in the medium is high and a reliable data delivery has the highest priority. Finally, we aim to develop new strategies that can be aggregated toward the proposed protocol to facilitate the cooperation among different wireless sensor networks.

An immediate future direction is to interchange the aspects we studied in this thesis, i.e., modeling, predicting and controlling, among the networks we analyzed. For instance, it would be interesting to design a model for the evolution of competitive networks. Moreover, we proposed link prediction techniques for the communication networks. The point is that all

the aspects we studied in this thesis are not restricted to the type of networks we applied them. In fact, another future direction is to evaluate other types of DBCNs, such as collaboration networks or citation networks.

Appendix A

Modeling in Communication Networks

A.1 The Log-logistic Distribution

The log-logistic distribution was first proposed by Fisk [Fisk, 1961] to model income distribution, after observing that the OR plot of real data in log-log scales follows a power law $OR(x) = cx^\rho$. In summary, a random variable is log-logistically distributed if the logarithm of the random variable is logistically distributed. The logistic distribution is very similar to the normal distribution, but it has heavier tails. In the literature, there are examples of the use of the log-logistic distribution in survival analysis [Bennett, 1983; Mahmood, 2000], distribution of wealth [Fisk, 1961], flood frequency analysis [M.I. Ahmad and Werritty, 1988], software reliability [Gokhale and Trivedi, 1998] and phone calls duration [Vaz de Melo et al., 2010]. A commonly used log-logistic parametrization is [Lawless and Lawless, 1982]:

$$\begin{aligned} PDF_{TLAC}(x) &= \frac{\exp(z(1+\sigma)-\ln(\mu))}{(\sigma(1+e^z))^2}, \\ CDF_{TLAC}(x) &= \frac{1}{1+\exp(-\frac{\ln(x)-\ln(\mu)}{\sigma})}, \\ z &= (\ln(x) - \ln(\mu))/\sigma, \end{aligned} \tag{A.1}$$

where $\sigma = 1/\rho$, the slope of our SFP model, and μ is the same. Moreover, when $\sigma = 1$, it is the same distribution as the Generalized Pareto distribution [Lorenz, 1905] with shape parameter $\kappa = 1$, scale parameter μ and threshold parameter $\theta = 0$.

Lemma 1. *The log-logistic distribution has a power law tail, i.e., it converges to a power law when $x \rightarrow \infty$.*

Proof. Considering the Probability Density Function of the log-logistic distribution, if we set the location parameter $\mu = 1$ for simplicity, $e^z = x^{1/\sigma}$. Moreover, solving $\lim_{x \rightarrow \infty} PDF_{TLAC}(x)$ when $\mu = 1$, we find that $\lim_{x \rightarrow \infty} PDF_{TLAC}(x) = x^{-(1+1/\sigma)}$. Thus, when $x \rightarrow \infty$, the IED generated by the SFP model is a power law with slope

$$\alpha = -(1 + 1/\sigma) = -(1 + \rho). \quad (\text{A.2})$$

□

A.2 The need for C in SFP

Lemma 2. *The constant $C = \mu/e > 0$ of Model 2 is needed to assure that the inter-event times generated by the SFP model will not converge to zero.*

Proof. If we remove the constant C from Model 2, $\Delta_t = (\Delta_{t-1}) \times (-\ln(U(0,1)))$, or Δ_t will be equal to Δ_{t-1} multiplied by a random number X extracted from the exponential distribution with parameter $\beta = \lambda = 1$. If $(X = \frac{1}{k} \mid k > 1)$, then Δ_t will be equal to Δ_{t-1} divided by k . The probability of X to be $\frac{1}{k}$ is $P(X = \frac{1}{k}) = e^{-\frac{1}{k}} = \frac{1}{k^e}$. On the other hand, the probability of multiplying Δ_t by k and, therefore, return Δ_{t+1} to Δ_{t-1} value is $P(X = k) = e^{-k} = \frac{1}{e^k}$. Given these probabilities, observe that $P(X = \frac{1}{k}) = \frac{1}{k^e} > P(X = k) = \frac{1}{e^k}, \forall k > 1$. From this, we conclude that the expected value of Δ_t when $t \rightarrow \infty$ is 0. With C in the equation, even when $\Delta_{t-1} = 0$, $\Delta_t = -C \times \ln(U(0,1))$, that is a classic Poisson process with $\beta = C$, and, obviously, does not converge to 0. □

A.3 The Two Parameters of SFP

An easy and direct way to define the relationships between the model parameters and the distribution properties is through simulations. Thus, the first point we consider is the median μ of the inter-event times generated by the SFP0 model. As we mentioned, when $OR(x) = 1$, x is the median μ of the distribution. We see in Figure 4.22-b that μ is close but different than the value of $C = 1$. Thus, in Figure A.1-a, we plot the OR for different values of C . As we can observe, changing the value of C changes μ and, consequently, the location of the distribution, but maintains its slope.

In order to investigate the relationship between C and μ , we run simulations of the model for all integer values of C between $[1,10000]$. As we observe in Figure A.1-b, the median of the distribution μ varies linearly with C according to a slope of 2.72, that can be approximated by Euler's number e , in a way that $\mu \propto e \times C$. This allows us to generate

inter-event times with a determined μ when the slope $\rho = 1$. We ignore the constant factor 3.8 because its 95% confidence interval is $(-8.596, 16.3)$, which contains zero.

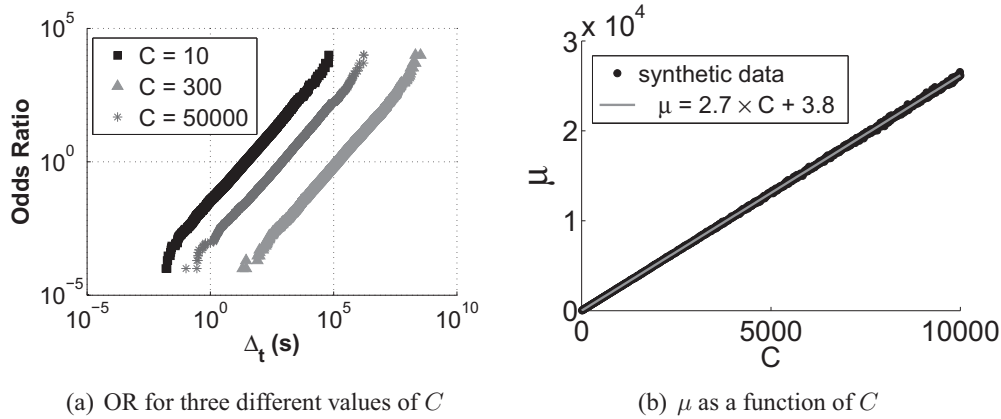


Figure A.1. Changing the value of C changes the location of the distribution. The median of the distribution μ varies linearly with C , $\mu = a \times C + b$, with $a = 2.6$ and $b = 3.8$. The 95% confidence interval for a is $(2.715, 2.723)$ and for b is $(-8.60, 16.3)$. Since the confidence interval for b contains 0, b is not significant.

Now we know how to generate inter-event times with different medians μ using the parameter C of SFP0 model. The next step is to analyze the parameter a of the SFP model to make it able to generate IEDs with a desired slope $\rho \neq 1$. Considering that up to this point the SFP model generates a set of inter-event times I_1 with a slope 1, the idea is to use a to transform I_1 into I_ρ , which is an IED with a different slope ρ . When we elevate each $\Delta_t \in I_1$ to the power of $a \neq 1$, the resulting slope ρ becomes different from 1. However, as we observe in Figure A.2-a, there is an inverse relationship between a and ρ , i.e., $\rho = a^{-1}$. Moreover, since the median of the distribution is also elevated to the power of a , we have to elevate the constant C to the power of $a^{-1} = \rho$ to preserve the median. Examples of IEDs with different slopes are shown in Figure A.2-b.

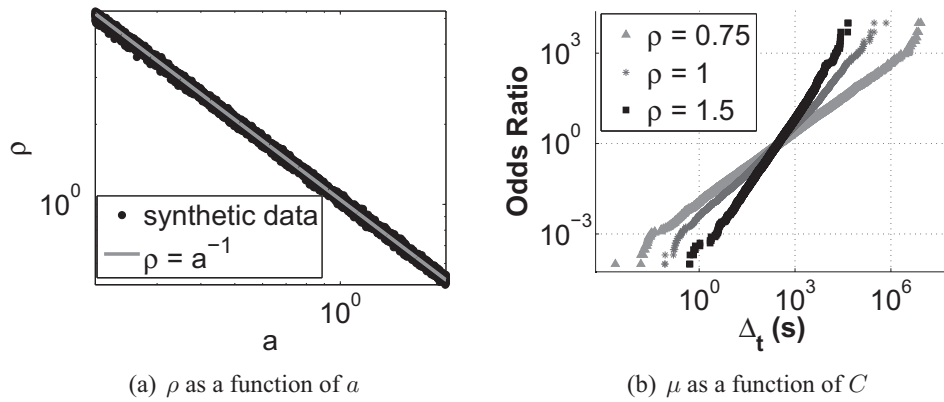


Figure A.2. Changing the value of a changes the slope ρ of the distribution in a way that $\rho = a^{-1}$.

Appendix B

Predicting in Competitive Networks

B.1 Metrics Weights

In Figure B.1 we show the model weight parameters w_i for the NetFor and NetForY models. First, we observe how they change over time, reflecting the temporal trends. Moreover, it is interesting to see how the weight parameters reflect the importance of each metric over time. For instance, looking at the NetForY weight parameters, excluding the year of 1949, there is always a network metric weight (w_1 to w_5) that has a high value than the yesterday metric weight (w_6). Again, this show how significant the network effects are in determining the performance of sports teams.

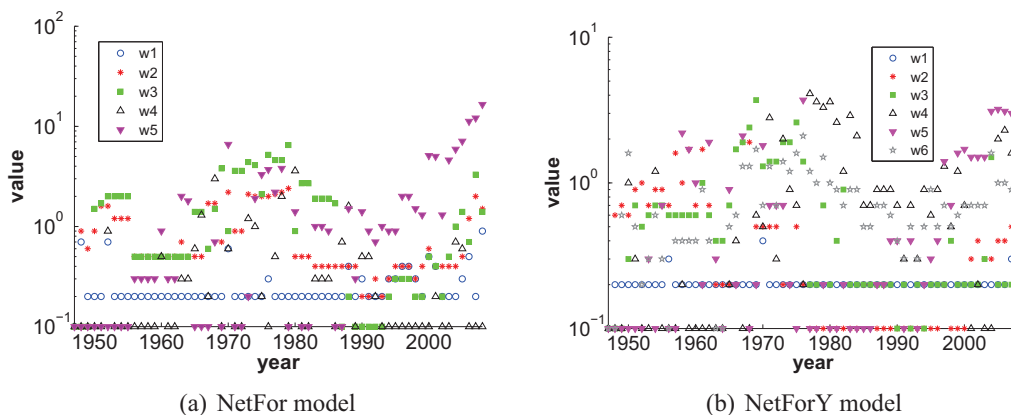


Figure B.1. The weights w_i of the network-based prediction models over time.

B.2 Random Model Comparison

Here we validate the network-based models by comparing them with a null model. The objective of this is to verify whether it is possible that the results of our models came from randomness, i.e., overfitting. We define the null model M_0 as a model that sets a uniformly random distributed number to the prediction score Π_t^y of team t in year y , in a way that:

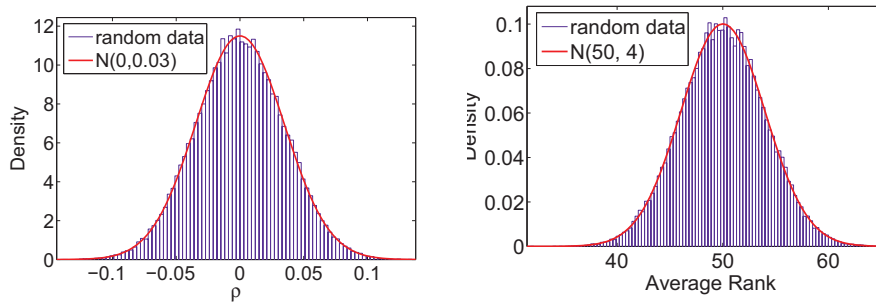
$$\Pi_t^y = f_0(t_y) = U(0, 1)$$

For this model, we ran 100,000 simulations in a way that one simulation is the prediction of every season between 1947 and 2008 using the null model. In order to analyze the behavior of the null model, we define two random variables, X_ρ and $X_{\bar{r}}$, in a way that:

[X_ρ =] the average Spearman rank correlation coefficient $\bar{\rho}$ of a simulation;

[$X_{\bar{r}}$ =] the average selected performance \bar{r} of a simulation;

In Figure B.2, we show the distribution of the results of the null model for 100000 simulations. In Figure B.2-a we show the density function of the random variable X_ρ and in Figure B.2-b we show the density function of the random variable $X_{\bar{r}}$. In red lines, we show the maximum-likelihood fitting of these density functions, where the random variables X_ρ and $X_{\bar{r}}$ were fitted, respectively, to the normal distributions $N_\rho(0, 0.03)$ and $N_{\bar{r}}(50, 4)$.



(a) Spearman $\bar{\rho}$ distribution. The distribution was fitted to a normal distribution $N_\rho(0, 0.03)$. (b) Average performance \bar{r} distribution. The distribution was fitted to a normal distribution $N_{\bar{r}}(50, 4)$.

Figure B.2. Distributions of the null model results over 100000 simulations.

Now we validate the network model by verifying if the results of the network-based models could be generated from the null model. Since the average results of the NetForY are always better than the results of the NetFor , we will only compare the null model with

the NetFor . Thus, we define the null hypothesis H_0 that states that the NetFor is an instance of the null model, in a way that:

$$H_0 : M_{NetFor} \subset M_0.$$

In order to reject H_0 , we verify the probability of making a simulation with the null model that results in an average Spearman rank correlation coefficient $\bar{\rho} > 0.59$ and an average selected performance $\bar{r} > 0.83$, that were the results obtained by the NetFor . Thus, based on the fitted distributions N_ρ and N_r we calculate the following probabilities:

$$P(X_\rho > 0.59) = 0$$

$$P(X_{\bar{r}} > 0.83) = 0$$

By these probabilities we can conclude that the network model results can not be a result that came from randomness, or from the null model, and, therefore, we reject the null hypothesis H_0 . We did this same test for the MLB dataset and the results are the same.

B.3 Parameter Sensitivity Analysis

We also investigate the sensitivity of the models with their only parameter W_Y , since there is the possibility that the results shown so far had come from a lucky W_Y . In Figure B.3, we plot the $\bar{\rho}$ of the NetForY and NetFor when the parameter W_Y is varied. We observe that these models are not significantly sensitive to W_Y . Observe that we could even had used values of W_Y that would give slightly higher results than the ones we used in the previous sections. This shows that the task of selecting a good value for W_Y is quite simple. One should only avoid significantly small values, that could give high importance to anomalous years, or significantly high values, that may fail to capture the temporal effects. However, as we show in Figure B.3, even extreme values of W_Y give results that are, in general, better than the results achieved from our competitors. We emphasize that we did this same analysis for the MLB dataset and the results are similar.

B.4 Network Features to Node Attributes

Another interesting application for the network features is to do a reverse engineering from network effects to node individual attributes. This reverse engineering can aid in the analysis of systems that has missing or erroneous explicit data, such as online social networks or

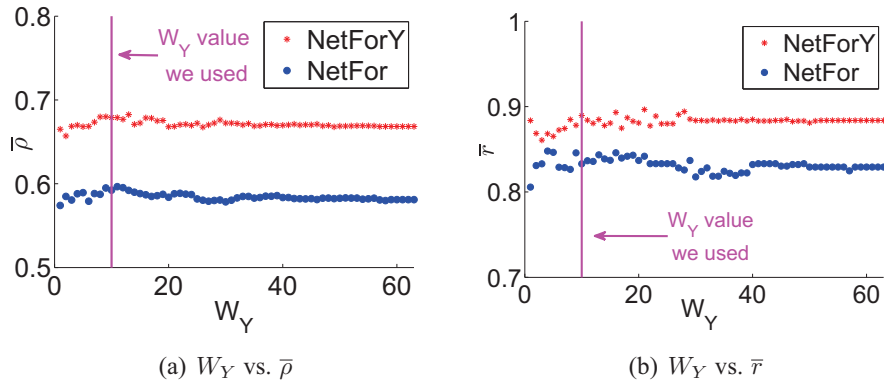


Figure B.3. The proposed network-based models are not significantly sensitive to W_Y . Observe that we could even had used values of W_Y that would give slightly higher results than the ones we used in the previous sections.

auction networks. In our case, we found out a significant correlation between the age of a node and its clustering coefficient when using the *Historical* propagation model, as we can observe in Figure B.4. In fact, this non-linear correlation is one of the main reasons for the *Team Inexperience* feature to aid in the prediction of teams behavior.

Although the usual and most natural way to describe a team t experience in year y is by its age a_t^y , i.e., the number of years a team have played in the NBA, the amount of experience a team acquire during the years is not a linear function, that is, the amount of experience a team get after playing its first season is probably significantly higher than the amount it will get by playing its 20th season. When we consider the clustering coefficient as a measure of experience, the experience difference between team t_1 that has $a_{t_1}^y = 0$ and another team t_2 that has $a_{t_2}^y = 2$ is higher than the experience difference between a team t_3 that has $a_{t_3}^y = 20$ and another team t_4 that has $a_{t_4}^y = 22$.

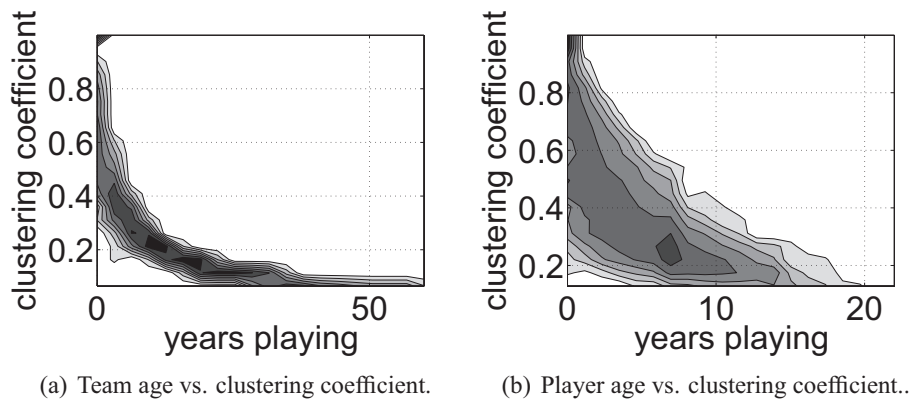


Figure B.4. Relationship between the clustering coefficient and the age of teams and players. In this case, the age is the number of years a team or player have being active in the league. Note that there is a clear non-linear correlation between the clustering coefficient and the age of the nodes.

Bibliography

- Abbot, H. (2007). Bad use of statistics is killing anderson varejão. *True Hoop*.
- Abe, S. and Suzuki, N. (2004). Scale-free network of earthquakes. *EPL (Europhysics Letters)*, 65(4):581.
- Adomavicius, G. and Tuzhilin, A. (2005). Toward the next generation of recommender systems: A survey of the state-of-the-art and possible extensions. *IEEE Transactions on Knowledge and Data Engineering*, 17:734--749.
- Akyildiz, I. F., Su, W., Sankarasubramaniam, Y., and Cayirci, E. (2002). Wireless sensor networks: a survey. *Computer Networks*, 38(4):393--422.
- Albert, R., Jeong, H., and Barabási, A.-L. (1999). Diameter of the World Wide Web. *Nature*, 401:130--131.
- Alpcan, T. and Basar, T. (2002). A game-theoretic framework for congestion control in general topology networks. *41st IEEE Conference on Decision and Control*.
- Amaral, L., Scala, A., Barthélémy, M., and Stanley, H. (2000). Classes of small-world networks. *Proceedings of the National Academy of Sciences*, 97(21):11149--11152.
- APBRmetrics (2011). www.apbrmetrics.com.
- Backstrom, L., Huttenlocher, D., Kleinberg, J., and Lan, X. (2006a). Group formation in large social networks: membership, growth, and evolution. In *KDD '06: Proceedings of the 12th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 44--54, New York, NY, USA.
- Backstrom, L., Huttenlocher, D., Kleinberg, J., and Lan, X. (2006b). Group formation in large social networks: membership, growth, and evolution. In *KDD '06: Proceedings of the 12th ACM SIGKDD*, pages 44--54, New York, NY, USA.

- Balabanovic, M. and Shoham, Y. (1997). Fab: Content-based, collaborative recommendation. *Communications of the ACM*, 40:66--72.
- Barabási, A. (2005). The origin of bursts and heavy tails in human dynamics. *Nature*, 435:207--211.
- Barabasi, A. and Albert, R. (1999). Emergence of Scaling in Random Networks. *Science*, 286(5439):509.
- Barabasi, A. L., Goh, K. I., and Vazquez, A. (2005). Reply to comment on "the origin of bursts and heavy tails in human dynamics".
- Ben-Naim, E., Vazquez, F., and Redner, S. (2007). Parity and predictability of competitions. *Journal of Quantitative Analysis in Sports*, 2(4):1.
- Bennett, J., Lanning, S., and Netflix, N. (2007). The netflix prize. In *In KDD Cup and Workshop in conjunction with KDD*.
- Bennett, S. (1983). Log-logistic regression models for survival data. *Journal of the Royal Statistical Society. Series C (Applied Statistics)*, 32(2):165--171.
- Box, G. E. P., Jenkins, G. M., and Reinsel, G. C. (1994). *Time Series Analysis, Forecasting, and Control*. Prentice-Hall, Englewood Cliffs, New Jersey, third edition.
- Bradley, R. (2009). Labor pains nothing new to the nba. *APBR.org*.
- Brigida Sartini, Gilmar Garbugio, H. B. e. P. S. e. L. B. (2004). Uma introducao a teoria dos jogos. *II Bienal da SBM*.
- Buragohain, C., Agrawal, D., and Suri, S. (2003). A game theoretic framework for incentives in p2p systems. *CoRR*, cs.GT/0310039.
- Burke, R. (2002). Hybrid recommender systems: Survey and experiments. *User Modeling and User-Adapted Interaction*, 12:331--370.
- Chakrabarti, D. and Faloutsos, C. (2006). Graph mining: Laws, generators, and algorithms. *ACM Comput. Surv.*, 38(1):2.
- Chen, Q. and Shi, D. (2004). The modeling of scale-free networks. *Physica A: Statistical Mechanics and its Applications*, 335(1-2):240 -- 248.
- Chung, F. and Lu, L. (2002). Connected Components in Random Graphs with Given Expected Degree Sequences. *Annals of Combinatorics*, 6(2):125--145.

- Clauset, A., Shalizi, C. R., and Newman, M. E. J. (2009). Power-law distributions in empirical data. *SIAM Review*, 51(4):661+.
- Corbo, J. and Petermann, T. (2004). Selfish peering and routing in the internet. In *Santa Fe Inst. Complex Systems Summer School*, Santa Fe, NM.
- Crandall, D., Cosley, D., Huttenlocher, D., Kleinberg, J., and Suri, S. (2008). Feedback effects between similarity and social influence in online communities. In *Proceedings of 14th ACM SIGKDD*, pages 160--168, New York, NY, USA.
- Crosby, G. V. and Pissinou, N. (2007). Evolution of cooperation in multi-class wireless sensor networks. *Local Computer Networks, Annual IEEE Conference on*, 0:489–495.
- da Costa, P. J. and Soares, C. (2004). A weighted rank measure of correlation. *Australian and New Zealand Journal of Statistics*, 47(4):515–529.
- databaseSports.com (2007). Database basketball. www.databasebasketball.com.
- DBLP (2010). The dblp computer science bibliography.
- de Araujo, R. B. (2003). Computação ubíqua: Princípios, tecnologias e desafios. In *XXI Simpósio Brasileiro de Redes de Computadores (SBRC)*, Natal, RN, Brasil.
- De Choudhury, M., Sundaram, H., John, A., and Seligmann, D. D. (2009). Social synchrony: Predicting mimicry of user actions in online social media. In *Proceedings of the 2009 International Conference on Computational Science and Engineering - Volume 04*, pages 151--158, Washington, DC, USA.
- Dilger, A. (2002). Never Change a Winning Team: An Analysis of Hazard Rates in the NBA. *SSRN eLibrary*.
- Du, N., Faloutsos, C., Wang, B., and Akoglu, L. (2009). Large human communication networks: patterns and a utility-driven generator. In *KDD '09: Proceedings of the 15th ACM SIGKDD*, pages 269--278, New York, NY, USA.
- Dunbar, R. I. M. (1998). The social brain hypothesis. *Evol. Anthropol.*, 6(5):178--190.
- Eagle, N., Pentland, A., and Lazer, D. (2009). From the Cover: Inferring friendship network structure by using mobile phone data. *Proceedings of the National Academy of Sciences*, 106(36):15274--15278.
- Easterbrook, G. (2006). The five-month nfl forecast. *ESPN.com*.

- Eckmann, J.-P., Moses, E., and Sergi, D. (2004). Entropy of dialogues creates coherent structures in e-mail traffic. *Proceedings of the National Academy of Sciences*, 101(40):14333-14337.
- Erdős, P. and Rényi, A. (1960). On the evolution of random graphs. pages 17--61.
- ESPN.com (2009). <http://sports.espn.go.com/nba/playoffs/2009/news/story?id=4135263>.
- Faloutsos, M., Faloutsos, P., and Faloutsos, C. (1999). On power-law relationships of the internet topology. In *SIGCOMM*, pages 251--262.
- Fast, A. and Jensen, D. (2006). The nfl coaching network: Analysis of the social network among professional football coaches. In *Proceedings of AAAI Fall Symposium on Capturing and Using Patterns for Evidence Detection*.
- Felegyhazi, M., Buttyan, L., and Hubaux, J. P. (2005). Cooperative packet forwarding in multi-domain sensor networks. In *Proceedings of IEEE PerSeNS 2005*, Hawaii, USA.
- Fessler, J. A. and Hero, A. O. (1994). Space-alternating generalized expectation-maximization algorithm. *IEEE Transactions on Signal Processing*, 42(10):2664--2677.
- Fisk, P. R. (1961). The graduation of income distributions. *Econometrica*, 29(2):171--185.
- Fruchterman, T. M. J. and Reingold, E. M. (1991). Graph drawing by force-directed placement. *Software: Practice and Experience*, 21(11):1129--1164.
- Fudenberg, D. and Tirole, J. (1991). *Game Theory*. MIT Press.
- Garriss, S., Kaminsky, M., Freedman, M. J., Karp, B., Mazières, D., and Yu, H. (2006). Re: Reliable email. In *Proceedings of the Third USENIX/ACM Symposium on Networked System Design and Implementation (NSDI'06)*, pages 297--310.
- Girvan, M. and Newman, M. E. (2002). Community structure in social and biological networks. *Proceedings of the National Academy of Sciences*, 99(12):7821--7826.
- Gokhale, S. S. and Trivedi, K. S. (1998). Log-logistic software reliability growth model. In *HASE '98: The 3rd IEEE International Symposium on High-Assurance Systems Engineering*, pages 34--41, Washington, DC, USA.
- Goldberg, D. S. and Roth, F. P. (2003). Assessing experimentally derived interactions in a small world. *Proceedings of the National Academy of Sciences*, 100(8):4372--4376.

- Guo, J., Liu, F., and Zhu, Z. (2007). Estimate the call duration distribution parameters in gsm system based on k-l divergence method. In *Wireless Communications, Networking and Mobile Computing, 2007. WiCom 2007. International Conference on*, pages 2988–2991.
- Guo, Z., Zhang, Z., Zhu, S., Chi, Y., and Gong, Y. (2009). Knowledge discovery from citation networks. In *2009 Ninth IEEE International Conference on Data Mining*, pages 800–805.
- Haight, F. A. (1967). *Handbook of the Poisson distribution [by] Frank A. Haight*. Wiley New York,.
- Helbing, D. and Yu, W. (2009). The outbreak of cooperation among success-driven individuals under noisy conditions. *Proceedings of the National Academy of Sciences*, 106:3680.
- Henderson, T., Kotz, D., and Abyzov, I. (2004). The changing usage of a mature campus-wide wireless network. In *Proceedings of the 10th annual international conference on Mobile computing and networking, MobiCom '04*, pages 187–201, New York, NY, USA.
- Hidalgo, C. A. (2006). Scaling in the inter-event time of random and seasonal systems. *Physica A: Statistical Mechanics and its Applications*, 369:877.
- Hidalgo, C. A. and Rodriguez-Sickert, C. (2008). The dynamics of a mobile phone network. *Physica A: Statistical Mechanics and its Applications*, 387(12):3017 – 3024.
- Hill, S. and Nagle, A. (2009). Social network signatures: A framework for re-identification in networked data and experimental results. In *CASON '09: Proceedings of the 2009 International Conference on Computational Aspects of Social Networks*, pages 88–97, Washington, DC, USA.
- Holme, P. and Kim, B. (2002). Growing scale-free networks with tunable clustering. *Physical Review E*, 65(026107).
- jen Hsu, W. and Helmy, A. (2005). Impact: Investigation of mobile-user patterns across university campuses using wlan trace analysis. *CoRR*, abs/cs/0508009.
- Jensen, D. D., Fast, A. S., Taylor, B. J., and Maier, M. E. (2008). Automatic identification of quasi-experimental designs for discovering causal knowledge. In *KDD '08: Proceeding of the 14th ACM SIGKDD*, pages 372--380, New York, NY, USA.
- Jiang, H. and Dovrolis, C. (2005). Why is the internet traffic bursty in short time scales? In *Proceedings of the 2005 ACM SIGMETRICS International Conference on Measurement and Modeling of Computer Systems (SIGMETRICS'05)*, pages 241--252.

- Karagiannis, T., Molle, M., Faloutsos, M., and Broido, A. (2004). A nonstationary Poisson view of Internet traffic. In *INFOCOM 2004. Twenty-third Annual Joint Conference of the IEEE Computer and Communications Societies*, volume 3, pages 1558--1569 vol.3.
- Kautz, H., Selman, B., and Shah, M. (1997). Referral Web: combining social networks and collaborative filtering. *Communications of the ACM*, 40(3):63--65.
- Keller, E. F. (2005). Revisiting scale-free networks. *BioEssays*, 27(10):1060--1068.
- Kelly, D. and Teevan, J. (2003). Implicit feedback for inferring user preference: a bibliography. *SIGIR Forum*, 37(2):18--28.
- Kendall, M. G. and Gibbons, J. D. (1990). *Rank Correlation Methods*. Oxford University Press, New York, 5th edition.
- Kleinberg, J. (2002). Bursty and hierarchical structure in streams. In *Proceedings of the eighth ACM SIGKDD*, KDD '02, pages 91--101, New York, NY, USA.
- Klimt, B. and Yang, Y. (2004). Introducing the enron corpus. In *CEAS'04: The First Conference on Email and Anti-Spam*.
- Krebs, V. (2002). Mapping networks of terrorist cells.
- Kuczura, A. (1973). The interrupted poisson process as an overflow process. *The Bell System Technical Journal*, 52:437--448.
- Kumar, R., Novak, J., and Tomkins, A. (2006). Structure and evolution of online social networks. In *KDD '06: Proceedings of the 12th ACM SIGKDD*, pages 611--617, New York, NY, USA.
- Lahman, S. (2008). The lahman baseball database. baseball1.com.
- Lawless, J. F. and Lawless, J. F. (1982). *Statistical Models and Methods for Lifetime Data (Wiley Series in Probability & Mathematical Statistics)*. John Wiley & Sons.
- Leskovec, J., Backstrom, L., Kumar, R., and Tomkins, A. (2008). Microscopic evolution of social networks. In *KDD '08: Proceeding of the 14th ACM SIGKDD*, pages 462--470, New York, NY, USA.
- Leskovec, J., Kleinberg, J., and Faloutsos, C. (2007). Graph evolution: Densification and shrinking diameters. *ACM Transactions on Knowledge Discovery from Data*, 1(1).
- Li, L., Alderson, D., Doyle, J., and Willinger, W. (2005). Towards a theory of scale-free graphs: Definition, properties, and implications. *Internet Mathematics*, 2(4):431--523.

- Liben-Nowell, D. and Kleinberg, J. (2007). The link-prediction problem for social networks. *Journal of the American Society for Information Science and Technology*, 58:1019--1031.
- LinkedIn (2010). LinkedIn.
- LiveJournal (2010). Livejournal.
- Looney, D. S. (1976). The start of a chain reaction? *Sports Illustrated*.
- Lorenz, M. O. (1905). Methods of measuring the concentration of wealth. *Publications of the American Statistical Association*, 9:209--219.
- Luce, R. D. and Raiffa, H. (1957). *Games and decisions : introduction and critical survey / R. Duncan Luce and Howard Raiffa*. Wiley, New York.
- Mahmood, T. (2000). Survival of newly founded businesses: A log-logistic model approach. *Journal Small Business Economics*, 14(3):223--237.
- Malmgren, R. D., Hofman, J. M., Amaral, L. A., and Watts, D. J. (2009a). Characterizing individual communication patterns. In *Proceedings of the 15th ACM SIGKDD international conference on Knowledge discovery and data mining, KDD '09*, pages 607--616, New York, NY, USA.
- Malmgren, R. D., Stouffer, D. B., Campanharo, A. S. L. O., and Amaral, L. A. N. (2009b). On universality in human correspondence activity. *SCIENCE*, 325:1696.
- Malmgren, R. D., Stouffer, D. B., Motter, A. E., and Amaral, L. A. N. (2008). A poissonian explanation for heavy tails in e-mail communication. *Proceedings of the National Academy of Sciences*, 105(47):18153--18158.
- Massey, F. J. (1951). The kolmogorov-smirnov test for goodness of fit. *Journal of the American Statistical Association*, 46(253):68--78.
- McGlohon, M., Akoglu, L., and Faloutsos, C. (2008). Weighted graphs and disconnected components: patterns and a generator. In *KDD '08: Proceeding of the 14th ACM SIGKDD*, pages 524--532, New York, NY, USA.
- M.I. Ahmad, C. S. and Werritty, A. (1988). Log-logistic flood frequency analysis. *Journal of Hydrology*, 98:205--224.
- Milgram, S. (1967). The small world problem. *Psychology Today*, 1:60--67.
- Miller, D., Tilak, S., and Fountain, T. (2005). "token" equilibria in sensor networks with multiple sponsors. In *CollaborateCom*. IEEE.

- Min-You Wu, W. S. (2005). Intersensornet: strategic routing and aggregation. In *IEEE Global Telecommunications Conference - GLOBECOM '05*.
- Mislove, A., Marcon, M., Gummadi, K. P., Druschel, P., and Bhattacharjee, B. (2007). Measurement and analysis of online social networks. In *IMC '07: Proceedings of the 7th ACM SIGCOMM conference on Internet measurement*, pages 29--42, New York, NY, USA.
- Mitzenmacher, M. (2004). A brief history of generative models for power law and lognormal distributions. *Internet Mathematics*, 1(2):226--251.
- Nash, J. F. (1950). Equilibrium points in n-person games. In *Proceedings of the National Academy of Sciences of the United States of America*.
- Nash, J. F. (1951). Non-cooperative games. *The Annals of Mathematics*, 54(2):286--295.
- nba.com (2008). www.nba.com.
- Neville, J., Şimşek, O., Jensen, D., Komoroske, J., Palmer, K., and Goldberg, H. (2005). Using relational knowledge discovery to prevent securities fraud. In *KDD '05: Proceedings of the eleventh ACM SIGKDD international conference on Knowledge discovery in data mining*, pages 449--458, New York, NY, USA.
- Newman, M. (2003). The structure and function of complex networks.
- Newman, M. E. (2001). Clustering and preferential attachment in growing networks. *Physical Review E*, 64(2):025102+.
- Nichols, D. M. (1998). Implicit rating and filtering. In *In Proceedings of the Fifth DELOS Workshop on Filtering and Collaborative Filtering*, pages 31--36.
- Nisan, N., Roughgarden, T., Tardos, E., and Vazirani, V. V. (2007). *Algorithmic Game Theory*. Cambridge University Press, New York, NY, USA.
- Oliveira, J. G. and Barabasi, A.-L. (2005). Human dynamics: Darwin and Einstein correspondence patterns. *Nature*, 437(7063):1251.
- Onnela, J. P., Saramäki, J., Hyvönen, J., Szabó, G., Lazer, D., Kaski, K., Kertész, J., and Barabási, A. L. (2007). Structure and tie strengths in mobile communication networks. *Proceedings of the National Academy of Sciences*, 104(18):7332--7336.
- Onody, R. N. and de Castro, P. A. (2004). Complex network study of brazilian soccer players. *Physical Review E*, 70:037103.
- Orkut (2010). Orkut.

- Page, G., Fellingham, G., and Reese, C. (2007). Using box-scores to determine a position's contribution to winning basketball games. *Journal of Quantitative Analysis in Sports*, 3(4):1.
- Pandit, S., Chau, D. H., Wang, S., and Faloutsos, C. (2007). Netprobe: a fast and scalable system for fraud detection in online auction networks. In *WWW '07: Proceedings of the 16th international conference on World Wide Web*, pages 201--210, New York, NY, USA.
- Park, J. and Newman, M. E. J. (2005). A network-based ranking system for us college football. *Journal of Statistical Mechanics: Theory and Experiment*, 2005(10):P10014.
- Paulsen, J. (2006). Efficiency per minute. *The Scores Report*.
- Piorowski, M., Djukic, N. S., and Grossglauser, M. (2009). A Parsimonious Model of Mobile Partitioned Networks with Clustering. In *The First International Conference on COMMunication Systems and NETWORKS (COMSNETS)*, Bangalore, India.
- Porter, R., Nudelman, E., and Shoham, Y. (2004). Simple search methods for finding a nash equilibrium. In *AAAI*, pages 664--669.
- Pottie, G. and Kaiser, W. (2000). Embedding the internet wireless integrated network sensors. In *Communications of the ACM*, volume 43, pages 51--58.
- Price, D. D. S. (1976). A general theory of bibliometric and other cumulative advantage processes. *Journal of the American Society for Information Science*, pages 292--306.
- Raghavan, P. (2002). Social networks: From the web to the enterprise. *IEEE Internet Computing*, 6:91--94.
- Reheuser, R. (2010). Bucking the trend. *NBA Encyclopedia*.
- Rojas, A., Branch, P., and Armitage, G. (2005). Experimental validation of the random waypoint mobility model through a real world mobility trace for large geographical areas. In *Proceedings of the 8th ACM international symposium on Modeling, analysis and simulation of wireless and mobile systems, MSWiM '05*, pages 174--177, New York, NY, USA.
- Roughgarden, T. and Tardos, E. (2002). How bad is selfish routing? *Journal of the ACM*, 49(2):236--259.
- Seshadri, M., Machiraju, S., Sridharan, A., Bolot, J., Faloutsos, C., and Leskove, J. (2008). Mobile call graphs: beyond power-law and lognormal distributions. In *KDD '08: Proceeding of the 14th ACM SIGKDD*, pages 596--604, New York, NY, USA.

- Shetty, J. and Adibi, J. (2005). Discovering important nodes through graph entropy the case of enron email database. In *LinkKDD '05: Proceedings of the 3rd international workshop on Link discovery*, pages 74--81, New York, NY, USA.
- Stekler, H., Sendor, D., and Verlander, R. (2010). Issues in sports forecasting. *International Journal of Forecasting*, 26(3):606 – 621. Sports Forecasting.
- Steyvers, M. and Tenenbaum, J. B. (2005). The Large-Scale Structure of Semantic Networks: Statistical Analyses and a Model of Semantic Growth. *Cognitive Science: A Multidisciplinary Journal*, 29(1):41--78.
- Stouffer, D. B., Malmgren, R. D., and Amaral, L. A. N. (2005). Comment on barabasi, nature 435, 207 (2005).
- Suri, S., Toth, C. D., and Zhou, Y. (2007). Selfish load balancing and atomic congestion games. *Algorithmica*, 47:79--96.
- Tanenbaum, A. (2002). *Computer Networks*. Prentice Hall Professional Tech. Reference.
- Taskar, B., Wong, M. F., Abbeel, P., and Koller, D. (2003). Link Prediction in Relational Data. In *in Neural Information Processing Systems*.
- Tejinder S. Randhawa, S. H. (2003). *Network Management in Wired and Wireless Networks*. Springer-Verlag New York, LLC.
- Toscher, A., Jahrer, M., and Bell, R. M. (2009). The bigchaos solution to the netflix grand prize.
- Turocy, T. L. and von Stengel, B. (2001). Game theory. Technical report, Texas A&M University and London School of Economics.
- Vaz de Melo, P. O. S., Akoglu, L., Faloutsos, C., and Loureiro, A. A. F. (2010). Surprising patterns for the call duration distribution of mobile phone users. In *ECML/PKDD (3)*, pages 354–369.
- Vaz de Melo, P. O. S., Almeida, V. A. F., Loureiro, A. A., and Faloutsos, C. (2012). Forecasting in the nba and other team sports: Network effects in action. *ACM Transactions on Knowledge Discovery from Data*.
- Vaz de Melo, P. O. S., Almeida, V. A. F., and Loureiro, A. A. F. (2008a). Can complex network metrics predict the behavior of nba teams? In *KDD '08: Proceeding of the 14th ACM SIGKDD international conference on Knowledge discovery and data mining*, pages 695--703, New York, NY, USA.

- Vaz de Melo, P. O. S., Cunha, F. D., Almeida, J. M., Mini, R. A., and Loureiro, A. A. F. (2008b). O problema da cooperação entre redes de sensores sem fio. In *XXVI Simpósio Brasileiro de Redes de Computadores (SBRC)*, Rio de Janeiro, RJ, Brasil.
- Vaz de Melo, P. O. S., da Cunha, F. D., Almeida, J. M., Loureiro, A. A. F., and Mini, R. A. (2008c). The problem of cooperation among different wireless sensor networks. In *MSWiM '08: Proceedings of the 11th international symposium on Modeling, analysis and simulation of wireless and mobile systems*, pages 86--91, New York, NY, USA.
- Vaz de Melo, P. O. S., Faloutsos, C., and Loureiro, A. A. F. (2011a). Human dynamics in large communication networks. In *SDM*, pages 968--879. SIAM / Omnipress.
- Vaz de Melo, P. O. S., Fernandes, C., Mini, R. A. F., Loureiro, A. A. F., and Almeida, V. A. F. (2011b). *Game Theory for Wireless Communications and Networking*, chapter Game Theory in Wireless Sensor Networks. Routledge, Taylor & Francis Group.
- Vaz de Melo, P. O. S., Loureiro, A. A. F., do V. Machado, M., and Mini, R. A. F. (2009). Um protocolo baseado em teoria dos jogos para a cooperação entre diferentes redes de sensores sem fio. In *XXVII Simpósio Brasileiro de Redes de Computadores e Sistemas Distribuídos*.
- Vázquez, A. (2003). Growing network with local rules: Preferential attachment, clustering hierarchy, and degree correlations. *Physical Review E*, 67(5):056104+.
- Vázquez, A., Oliveira, J. G., Dezsö, Z., Goh, K. I., Kondor, I., and Barabási, A. L. (2006). Modeling bursts and heavy tails in human dynamics. *Physical Review E*, 73(3):036127+.
- Watts, D. J. and Strogatz, S. H. (1998). Collective dynamics of "small-world" networks. *Nature*, 393:440--442.
- Weiser, M. (1999). The computer for the 21st century. *SIGMOBILE Commun. Rev.*, 3(3):3-11.
- Wieselthier, J. E., Nguyen, G. D., and Ephremides, A. (2001). Algorithms for energy-efficient multicasting in static ad hoc wireless networks. *Mobile Network Applications*, 6(3):251--263.
- Willkomm, D., Machiraju, S., Bolot, J., and Wolisz, A. (2008). Primary users in cellular networks: A large-scale measurement study. In *New Frontiers in Dynamic Spectrum Access Networks, 2008. DySPAN 2008. 3rd IEEE Symposium on*, pages 1--11.

- Zhou, H. and Lipowsky, R. (2005). Dynamic pattern evolution on scale-free networks. *Proceedings of the National Academy of Sciences*, 102(29):10052--10057.
- Zyba, G., Voelker, G. M., Ioannidis, S., and Diot, C. (2011). Dissemination in opportunistic mobile ad-hoc networks: The power of the crowd. In *Proceedings of IEEE INFOCOM 2011*, pages 1179--1187.