

Recolha e Análise de Dados de Contactos Físicos e Sociais numa Rede Tolerante a Atrasos

João Antunes*, António Costa†, Joaquim Macedo^

Centro Algoritmi

Universidade do Minho

Braga, Portugal 4710-057

* Email: joao99antunes@gmail.com

† Email: costa@di.uminho.pt

^ Email: macedo@di.uminho.pt

Resumo—As redes tolerantes a atrasos surgiram com o propósito de abordar o problema de comunicação em redes onde a ligação é intermitente e feita através de contactos oportunistas. Um caso particular destas redes são aquelas em que os nós são dispositivos transportados por pessoas, as *Pocket Switch Networks*. A relação social entre os nós tem sido recentemente explorada na decisão de encaminhamento neste tipo de redes.

Neste trabalho, foi concebido um sistema de recolha de dados dos contactos físicos e sociais numa RTA com o objetivo de avaliar uma nova métrica social a ser usada na decisão de encaminhamento.

Palavras-chave— Redes Tolerantes a Atrasos, Sistema de Recolha, Datasets, *Pocket Switch Networks*, Redes Sociais.

I. INTRODUÇÃO

O enorme sucesso da Internet nas últimas três décadas deve-se ao uso de protocolos denominados TCP/IP, pois estes garantem a flexibilidade, eficiência e robustez, que lhe permitem suportar diversas aplicações em diferentes cenários. No entanto, em cenários com longos atrasos e conexões intermitentes, os protocolos TCP/IP não funcionam e é necessário a criação de novos protocolos. Redes com estas características específicas são denominadas Redes Tolerantes a Atrasos (RTAs), e um dos seus principais desafios é o encaminhamento, pois é necessário determinar rotas sem o conhecimento da existência de um caminho fim-a-fim[1].

Para contornar os problemas de atrasos longos e conexões intermitentes, as RTAs usam técnicas de troca de mensagens e armazenamento de dados. Quando uma mensagem precisa de ser enviada, ela é armazenada e encaminhada nó a nó desde a origem até o destino, ou seja, são redes do tipo *store-and-forward*: a mensagem é recebida integralmente e armazenada, para depois ser enviada ao próximo nó, que pode ou não ser o nó destino[2].

Um dos tipos de RTAs mais usuais são as chamadas *Pocket Switched Networks* (PSNs), que consistem em contactos oportunistas entre os nós da rede. Estes nós encontram-se em constante movimento e nem sempre estão conectados à rede,

visto serem controlados diretamente por seres humanos, através de smartphones, computadores portáteis, etc.[3].

Tendo em conta que os nós das redes dependem de fatores humanos, os protocolos de encaminhamento para este tipo de redes podem basear-se em dois campos: o campo físico, que consiste no histórico de contactos¹ entre os nós da rede; e o campo social que consiste na relação social entre os mesmos. Existem atualmente alguns estudos sobre a relação entre o grafo social e o grafo de contactos físicos[4],[5], no entanto, o propósito final deste trabalho científico é verificar a validade da inclusão na decisão de encaminhamento de uma métrica que relaciona o campo social e o histórico de contactos físicos.

Para obter a relação social entre os nós, podemos utilizar as redes sociais existentes na Internet, como o Facebook, o Twitter ou o MySpace. Estas redes são utilizadas regularmente por milhões de pessoas em todo o mundo e permitem obter dados da vida social de cada um.

Este trabalho propõe um sistema baseado no conceito Cliente-Servidor para criação de uma RTA através da qual se pretende obter os dados do histórico de contactos entre os nós e a relação social entre eles, para posterior construção e análise de um grafo que relacione estes dados. Para isso, o trabalho irá consistir em três partes distintas: uma aplicação em Android instalada em cada nó da RTA, responsável pela obtenção do histórico dos contactos físicos entre os nós e de um perfil social de cada nó enviando a informação obtida para o servidor; um servidor responsável por receber dados dos nós da rede e pela construção de um grafo que relacione o histórico de contactos entre os nós e a relação social entre eles; um sistema de análise do grafo construído no servidor que permita obter resultados acerca do relacionamento entre o histórico de contactos e os dados sociais entre os nós numa RTA.

¹ Um contacto é o encontro entre dois nós da rede, ou seja, quando um nó se encontra ao alcance de outro nó de modo a que estes possam comunicar

II. ESTADO DA ARTE

Podemos dividir os protocolos em RTAs em dois tipos distintos: protocolos baseados em redes onde existem infraestruturas e protocolos baseados em redes puramente móveis. Este último tipo de redes pode também ser dividido em dois tipos: protocolos *dissemination-based* e *context-based*. A diferença entre estes dois tipos consiste na falta de informação de contexto no encaminhamento de mensagens entre os nós da rede nos protocolos *dissemination-based*, enquanto que os protocolos *context-based* baseiam a decisão de encaminhamento de dados numa informação de contexto que pode incluir dados relativos ao histórico de contactos entre os nós ou da relação social entre eles, ou mesmo uma junção entre estes dois campos.

Nos protocolos *dissemination-based* podemos incluir o Epidemic Routing[6]. Este protocolo consiste na distribuição das mensagens pela rede de uma forma epidémica, isto é, a cada novo contato entre dois nós da rede existe uma troca de informações entre os nós de modo a que cada nó receba as mensagens que ainda não tenha guardado. Desta forma, as mensagens são rapidamente distribuídas entre duas porções de rede que estejam próximas uma da outra contando para isso com a mobilidade dos nós. Este protocolo garante uma probabilidade elevada na entrega de mensagens, contudo, sobrecarrega a rede devido à falta de informação de contexto e consome demasiados recursos aos nós da rede. Para contrariar a sobrecarga da rede e dos nós, posteriores versões deste protocolo incluíram um número limite de saltos e de cópias das mensagens presentes na rede.

Dentro dos protocolos *context-based* incluem-se o PROPHET[7], o SimBet[8], o FairRouting[9], o BubbleRap[10] e o PeopleRank[11]. Cada um destes protocolos utiliza uma informação de contexto na decisão de encaminhamento de mensagens entre os nós da rede, sendo que a diferença entre eles está na natureza dessa informação. O PROPHET baseia-se unicamente no histórico de contactos entre os nós e na transitividade na decisão de encaminhamento, isto é, se dois nós se encontrarem regularmente, têm maior probabilidade de entregar uma mensagem um ao outro do que dois nós que raramente se encontrem. O SimBet inclui na sua informação de contexto a centralidade do nó da rede, a similaridade dos nós e a força das ligações entre eles. A informação de contexto no FairRouting inclui a força de interação e o grau de *assortativeness* (pessoas com os mesmos interesses tendem a encontrar-se mais vezes). Já o BubbleRap utiliza a divisão dos nós em comunidades e a centralidade de cada um deles na decisão de encaminhamento. Por fim, o PeopleRank utiliza a relação social entre os nós da rede como informação de contexto.

Em relação à comunicação entre os nós existe um consenso em torno de um protocolo experimental que é usado na troca de mensagens numa rede RTA, o Bundle Protocol RFC 5050[12] que pode ser usado como suporte em todos os protocolos de encaminhamento. Este protocolo foi elaborado pelo IRTF's Delay Tolerant Networking Research Group (DTNRG), e divide blocos de dados em pacotes, transmitindo-os, usando um mecanismo de store-and-forward.

III. SISTEMA DE RECOLHA E ANÁLISE DE DADOS

No sistema desenvolvido foi necessário definir algumas questões como a forma de comunicação entre os nós da rede, o tipo de base de dados utilizada pelo servidor para construir o grafo e a forma de criar o perfil social de cada nó.

A. Comunicação entre os nós da rede

No que concerne a comunicação entre os nós da rede, foram estudadas algumas hipóteses: Bluetooth, infravermelhos, redes ad-hoc Wi-Fi, redes Wi-Fi e redes celulares.

Destas cinco possíveis formas de comunicação, três delas foram automaticamente excluídas por razões distintas. Os infravermelhos foram rejeitados por serem uma tecnologia em decadência, visto já não existirem na maioria dos dispositivos tecnológicos. As redes celulares GSM ou UMTS por terem um custo associado na troca de mensagens não foram consideradas. E por fim, as redes Wi-Fi foram excluídas devido à necessidade de usarem pontos de acesso, o que não permite a possibilidade de comunicação a qualquer momento, um aspeto crucial na rede criada.

Restam duas tecnologias: o Bluetooth e as redes ad-hoc Wi-Fi. A escolha recaiu sobre o Bluetooth, visto que esta tecnologia requer um consumo de energia menor que a utilização do Wi-Fi[13] em dispositivos móveis. Por outro lado a tecnologia ad-hoc Wi-Fi só está acessível em versões mais recentes do sistema operativo Android.

Independentemente da tecnologia utilizada como forma de comunicação entre os nós deve ser possível retirar informação acerca de cada contacto entre dois nós. Todos os dados interessantes passíveis de serem retirados de um contato físico entre dois nós são: a identificação dos sujeitos (endereço MAC); a duração do contacto; o instante de início do contacto; a localização geográfica de onde ocorreu o contacto; e o nível de energia e de memória de cada nó;

Depois de analisados estes pontos, concluiu-se que alguns campos poderiam ser dispensados. Entre eles, a localização geográfica de onde ocorreu o contacto, que é desnecessário pois requer o uso de tecnologias que iriam consumir mais recursos aos nós da rede. Também, o nível de energia e de memória de cada nó considera-se dispensável, pois não acrescenta informação quanto à força da ligação entre os nós, sendo que estes valores só são importantes quando tratamos de RTA com nós egoístas.

B. Registo do Histórico de Interações

Para o desenvolvimento deste projeto, é necessária a construção de uma base de dados pelo servidor de forma a guardar os dados de contactos físicos entre os nós da rede e o perfil social desses mesmos nós. Para esse efeito, foi necessário escolher um tipo de base de dados, sendo que essa escolha poderia recorrer sobre uma base de dados relacional (SQL) ou uma base de dados não relacional (NoSQL).

Visto que, para este projeto, o que se pretende guardar são dados obtidos de uma rede, o tipo de base de dados mais indicado seria uma base de dados não relacional, ou seja, a base de dados NoSQL.

O termo NoSQL (Not only SQL) refere-se a uma classe de base de dados não relacional, acabando, desta forma, com o uso exclusivo de bases de dados relacionais. A principal razão do aparecimento deste tipo de base de dados deve-se à necessidade crescente de obter maior escalabilidade no armazenamento de dados. As bases de dados relacionais também promovem escalabilidade, no entanto, quanto maior for o tamanho dos dados, mais custosa se torna a escalabilidade, seja pelo custo de novas máquinas ou pela manutenção de especialistas no local onde se encontram as bases de dados. Por esta razão, foram criadas as bases de dados não relacionais NoSQL, que permitem uma escalabilidade mais barata e menos trabalhosa, visto não necessitarem de máquinas muito poderosas e a sua manutenção ser simples, permitindo assim reduzir o número de profissionais necessários.

Existem alguns tipos de bases de dados não relacionais: bases de dados chave/valor, bases de dados orientados a documentos e bases de dados de grafos. Enquanto o primeiro e segundo tipo não são indicados para usar com grafos, o último tipo de bases de dados é bastante compatível com o nosso caso já que estamos a tratar de redes que, por sua vez, podem ser representadas por grafos[14]. A ideia deste modelo é representar os dados como grafos dirigidos. Este tipo de modelo funciona melhor quando se pretende dar mais relevância à conectividade ou à topologia dos dados que aos dados propriamente ditos. É constituído por três elementos básicos: os nós (vértices do grafo), as relações (as arestas) e as propriedades dos nós e das relações. Algumas das bases de dados que utilizam o modelo de grafos são: Neo4j, Infinite Graph, InforGrid, HyperGraphDB, etc. No âmbito deste projeto, vai ser utilizada a base de dados Neo4j que está desenhada para ser usada na linguagem de programação Java. Esta base de dados foi criada pela Neo Technology e o seu uso é bastante fácil e intuitivo, proporcionando uma forma eficaz de representar bases de dados por grafos e dando a possibilidade de atribuir, de uma forma simples, atributos tanto aos nós do grafo, como às ligações entre eles.

C. Obtenção do Perfil Social

Uma premissa para a realização deste trabalho foi recolher o perfil social de um grupo de pessoas (os utilizadores da aplicação) para posteriormente construir um grafo social. De forma a recolher estes dados, decidiu-se utilizar as redes sociais, visto estarem muito em voga na atualidade. Entre elas, as mais conhecidas e as mais utilizadas são: o Facebook, o Twitter, o MySpace e o LinkedIn.

O Facebook é a rede social online mais utilizada no mundo inteiro, por ser gratuita, de interesse generalizado e simples de usar. Nesta rede social, cada utilizador tem um perfil com a sua informação pessoal, lista de interesses, fotos e uma lista de amigos. Os utilizadores podem trocar mensagens privadas e públicas entre si. Cada pessoa tem um mural onde os seus amigos podem colocar publicações (frases, fotografias, vídeos, etc.) e que pode ser acedido por todos os utilizadores da rede, ou não, dependendo da privacidade definida pelo próprio utilizador. Este tem, também, a possibilidade de criar ou aderir a grupos restritos, identificar amigos em publicações ou fotos, criar publicações no seu próprio mural e “gostar” de

publicações. A página principal do Facebook é o feed de notícias onde o utilizador pode ver a atividade recente dos seus amigos e notícias dos seus grupos e das páginas de que gosta.

Como já foi referido antes, retirou-se o perfil social dos nós da rede através do Facebook, visto ser a rede social online com mais utilizadores [15]. Como a aplicação está feita em programação Android, utilizou-se a ferramenta Facebook SDK integrada no eclipse. Esta ferramenta permite aceder de uma forma simples ao Graph API do Facebook através do qual podemos ler e escrever dados. Todos os dados relevantes passíveis de serem retirados do Facebook são: identificação do utilizador (nome e id); lista de amigos; lista de interesses; lista de familiares; localização; número de identificações em fotos/posts de amigos; número de feeds/mensagens trocadas com amigos.

Tendo em conta que o Facebook SDK tem algumas limitações na recolha de dados, e visto que se pretende preservar a máxima privacidade possível do utilizador, decidiu-se prescindir de obter os dois últimos campos, ou seja, o número de identificações em fotos/posts de amigos e o número de feeds/mensagens trocadas com amigos.

IV. ARQUITETURA

O sistema construído para a recolha de dados e construção de grafos sociais e de contactos físicos numa rede RTA, divide-se em dois componentes: os nós da rede e o servidor. Os nós da rede serão responsáveis por recolher os dados de contactos físicos entre eles e de obter o seu perfil social, enviando a informação obtida para o servidor. O servidor é responsável por receber os dados dos nós e construir um grafo que relacione o histórico de contactos entre os nós e o seu perfil social.

Os nós da rede serão simulados por utilizadores que terão instalado nos seus dispositivos móveis uma aplicação com as seguintes responsabilidades: definir um protocolo de comunicação para verificar a existência de contactos entre utilizadores; guardar em cache o histórico de contactos até obter acesso ao servidor; enviar o histórico de contactos ao servidor; obter o perfil social do utilizador; enviar o perfil social do utilizador para o servidor; e oferecer segurança na troca de dados.

O servidor é uma aplicação desenvolvida em Java que deve estar registada na internet para poder comunicar com os clientes. O servidor tem a responsabilidade de: permitir a ligação dos nós da rede através da internet; receber o perfil social e o histórico de contactos dos nós da rede; permitir adicionar e remover novos nós; construir um grafo que relacione o histórico de contactos entre os nós e a sua relação social. Na figura 1, podemos ver a arquitetura global do sistema. Na figura 2 podemos observar os módulos da aplicação e as interações entre estes.

A. Nó da rede

Para a realização deste projeto é necessário obter autorização de um grupo de pessoas para ter acesso ao seu perfil social e ao histórico dos contactos físicos existentes entre eles. Para conseguir essa autorização, decidiu-se criar um incentivo para que as pessoas aceitassem entregar os seus

dados - o acesso gratuito a uma aplicação denominada de SocialConnector. A aplicação está disponível para telemóveis com o sistema operativo Android e funcionará como um nó da rede RTA.

Depois de um estudo que visou a escolha de uma aplicação que fizesse com que várias pessoas instalassem a aplicação, decidiu-se que a SocialConnector iria consistir no seguinte: uma aplicação que nos mostra os interesses das pessoas que estão ao nosso lado, e que tenham a aplicação instalada, em qualquer local onde nos encontremos. A aplicação funciona à base de avisos, ou seja, quando descobrir alguém na mesma zona que tenha a aplicação instalada, notifica o utilizador da descoberta dessa pessoa, bem como do seu perfil social. Os interesses dos utilizadores são obtidos através do Facebook e são atualizados regularmente. A comunicação entre os utilizadores será feita por Bluetooth. Estando ligada a aplicação pode funcionar em três modos diferentes: modo normal, modo egoísta ou modo social. No modo normal, a aplicação tanto está à procura de novos dispositivos para receber os seus perfis, como torna o seu dispositivo visível disponibilizando-se para entregar o seu perfil ao nó encontrado. No modo egoísta, a aplicação só procura novos utilizadores para obtenção de novos perfis não se tornando visível para os outros nós e não permitindo, assim, enviar o seu perfil a novos utilizadores. No modo social, a aplicação torna-se visível para os outros nós da rede, enviando para estes o seu perfil, mas não procura novos dispositivos para receber perfis. A atratividade desta aplicação encontra-se no facto de ajudar as pessoas a socializar, pois permite aos utilizadores poderem descobrir os interesses de uma pessoa desconhecida, facilitando, assim, o início de uma conversa casual e, possivelmente, uma nova amizade.

A aplicação SocialConnector está dividida em 6 módulos distintos, cada um com a sua responsabilidade:

- **Ligação ao Facebook:** responsável por aceder à rede social e retirar os dados desejados, criando o perfil social.
- **Gestor de perfil:** responsável por gerir o perfil do utilizador, ou seja, definir o que o utilizador pretende partilhar com os outros. Ao recolher os dados do Facebook, o utilizador deverá criar um perfil social onde irá constar a sua informação retirada do Facebook para partilhar com os outros utilizadores. Devido a questões de privacidade, na aplicação deverá ser permitido ao utilizador escolher que dados do seu perfil pretende partilhar com os outros dispositivos, sendo possível esconder alguns dos seus interesses ou dos seus amigos. Os dados que constam no perfil social são: o nome, a idade, a lista de interesses, a lista de amigos e a localização.
- **Comunicação entre nós:** responsável por definir e estabelecer o protocolo de comunicação, aquando do encontro entre dois nós. A aplicação deve permitir procurar novos dispositivos, ao mesmo tempo que permite ao nó ficar visível para outros. A comunicação funciona da seguinte forma: se um nó A encontrar um nó B, ou vice-versa, depois de verificar se o nó B tem a aplicação instalada e a funcionar no modo normal ou social, o nó A pede uma parte do perfil do nó B, com o seu nome, idade e localização. Depois de receber esses primeiros dados, o nó B deve decidir se pretende pedir o resto do perfil social, e caso o pretenda deve pedi-lo ao nó B. O nó B deve dar

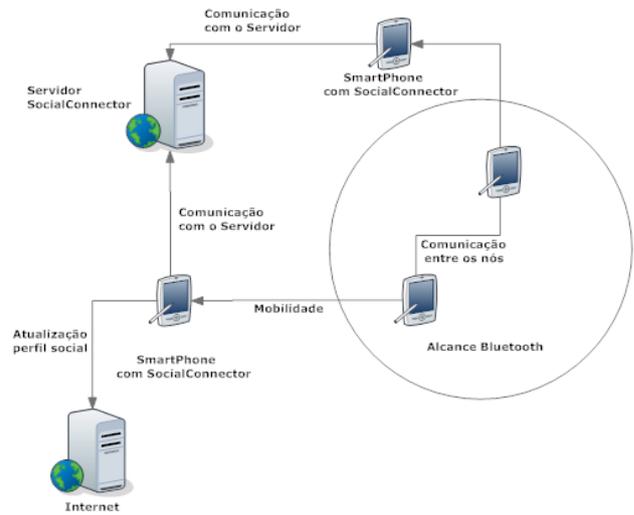


Figura 1 - Arquitetura do sistema

permissão para enviar o resto do perfil social. Caso o utilizador B permita o envio do perfil, é enviado para o nó A a totalidade do seu perfil: a sua foto, a sua lista de interesses e a lista de amigos (Figura 3). Os utilizadores devem guardar todos os perfis acedidos recentemente até ter acesso ao servidor. Aquando de um novo encontro, e depois da partilha de perfis entre si, os nós devem comunicar e, caso pretendam, devem enviar a lista de perfis encontrada recentemente, desta forma, contrariar o limite de alcance do Bluetooth, utilizando a transitividade na partilha de perfis.

- **Armazenamento de dados:** responsável por guardar temporariamente o histórico de contactos do nó e permanentemente o perfil social do utilizador.
- **Segurança e privacidade:** responsável por garantir a segurança na partilha de informação entre utilizadores e a privacidade dos mesmos. Para isso, é usada uma ligação RFCOMM segura com registo do serviço Bluetooth, sendo todos os identificadores sumariados com uma função de hash na comunicação entre nós e na ligação com o servidor.
- **Ligação ao servidor:** responsável por estabelecer a ligação ao servidor para o envio de dados. Cada vez que um nó acede à aplicação, verifica se tem acesso à internet, e, se tal for o caso, envia o seu histórico de contactos para o servidor, apagando-os da memória local. Na primeira ligação e quando houver uma atualização do perfil, o nó também envia uma mensagem para o servidor com o seu perfil social.

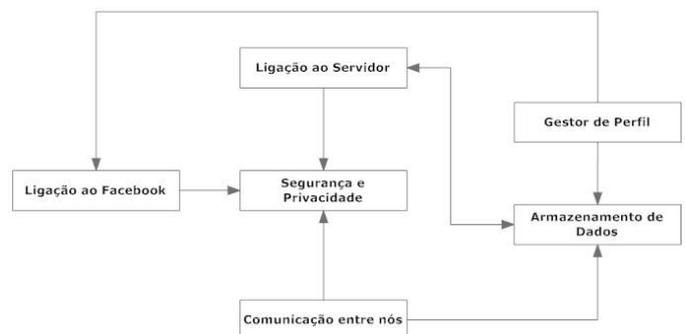


Figura 2 - Esquema SocialConnector

TABELA 1. DATASET SIGCOMM09

Número de participantes	76
Duração da experiência	5 Dias
Total de interesses	711
Encontros entre os nós	285879
Média contactos por nó	≈ 3762
Média interesses por nó	≈ 12
Média amizades por nó	≈ 4

definir um mecanismo de segurança que permita garantir a privacidade dos utilizadores na apresentação dos dados, isto é, não deve mostrar o nome do utilizador nem o seu perfil social, mas sim identificar esses dados por identificadores já sumariados nos clientes.

V. DISCUSSÃO E AVALIAÇÃO DO SISTEMA

O objetivo inicial era recolher efetivamente os dados, instalando a aplicação SocialConnector num conjunto de dispositivos que seriam utilizados por um grupo de pessoas previamente escolhidas. Nesta fase de desenvolvimento do trabalho, não foi ainda possível coletar dados reais utilizando o SocialConnector, tendo-se recorrido a datasets já existentes para avaliar o sistema.

Utilizando o crawdad, foi encontrado um dataset que permite avaliar o nosso sistema, o dataset sigcomm2009. Este contém dados coletados de uma aplicação desenvolvida para telemóveis com o sistema operativo Windows Phone: a MobiClique[4]. A aplicação foi usada por 76 pessoas durante os dias da conferência SIGCOMM, em 2009, em Barcelona. O dataset inclui os dados de proximidade obtidos através de Bluetooth, criação e disseminação de mensagens oportunistas e os perfis sociais, incluindo a lista de amigos e de interesses dos participantes. Foi pedido aos participantes que utilizassem um smartphone que lhes era fornecido com a aplicação MobiClique, e que o utilizassem durante os dois dias da experiência. Os dados sobre os contactos físicos e sociais foram guardados na memória dos próprios dispositivos. Na tabela 1, podemos verificar os valores globais acerca dos encontros e dos perfis sociais obtidos nesta experiência e que servirão para ser colocados no servidor desenvolvido de forma

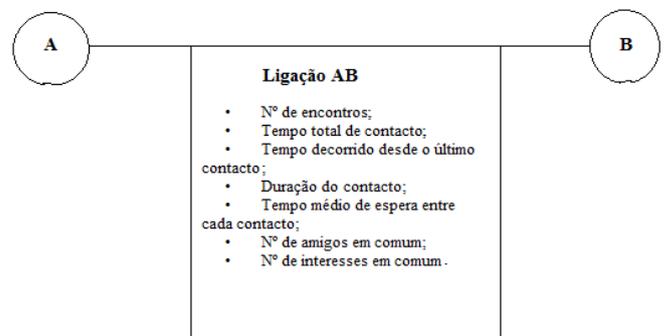


Figura 4 - Exemplo de uma ligação do grafo

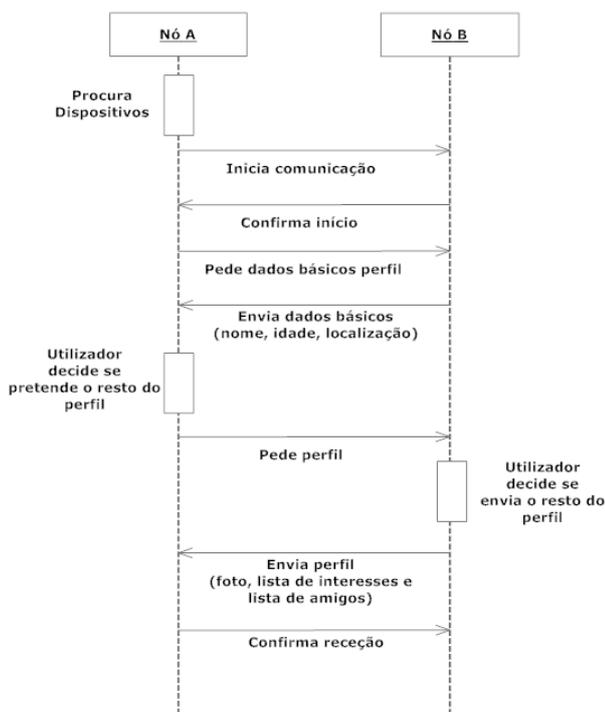


Figura 3 - Comunicação entre dois nós

B. Servidor

O servidor irá ser desenvolvido em Java e está dividido em cinco módulos:

- Registo na internet: responsável por colocar o servidor disponível na internet, de modo a que esteja acessível aos nós da rede.
- Comunicação com os nós: responsável por definir um protocolo de comunicação com os nós da rede, de modo a receber periodicamente o histórico de contactos e o perfil social de cada nó. O servidor deve receber do nó uma mensagem por cada contacto que este tenha, e que deve ter os seguintes campos: identificação do sujeito encontrado (endereço MAC), duração do contacto e instante de início do contacto.
- Processamento dos dados obtidos: responsável pelo processamento das mensagens recebidas dos nós da rede, de modo a armazenar a informação num grafo NoSQL.
- Criação do grafo: responsável por guardar permanentemente os dados obtidos num grafo numa base de dados NoSQL. Com os valores obtidos do histórico de contactos e dos perfis sociais, o servidor deve construir um grafo. Esse grafo, deve representar os nós e as ligações entre estes, relacionando os contactos físicos com o perfil social de cada utilizador. Uma ligação entre dois nós do grafo deve parecer-se com o da Figura 4. O servidor deverá, também, guardar para cada nó a seguinte informação: número total de contactos com todos os outros nós, duração total dos contactos com todos os outros nós, lista de amigos e lista de interesses.
- Segurança e privacidade: responsável por garantir a segurança na partilha de informação com os nós da rede e por

a serem analisados através do nosso programa de análise de dados.

O objetivo final deste trabalho é obter a relação entre o histórico de contactos físicos e a relação social dos nós numa rede RTA.

Com esse propósito, foram definidas duas métricas: o coeficiente social entre o nó A e o nó B $C_{S_{AB}} \in [0,1]$; e o coeficiente dos contactos físicos entre o nó A e o nó B $C_{F_{AB}} \in [0,1]$.

O coeficiente social é calculado da seguinte forma:

$$C_{S_{AB}} = \sigma \times C_{A_{AB}} + \beta \times C_{I_{AB}} \quad (1)$$

Nesta métrica estão incluídos o coeficiente de amizade $C_{A_{AB}} \in [0,1]$ e o coeficiente de interesses $C_{I_{AB}} \in [0,1]$. O primeiro indica a razão entre o número de amigos em comum entre A e B, sobre o número total de amigos que os dois têm em conjunto. O segundo refere-se à razão entre o número de interesses em comum entre A e B, sobre o número total de interesses que os dois têm em conjunto. σ e β são valores que, somados, têm que dar 1, e que se referem à importância que se pretende dar a cada coeficiente.

O coeficiente de contactos físicos é calculado da seguinte forma:

$$C_{F_{AB}} = [\delta \times C_{C_{AB}} + \alpha \times C_{D_{AB}}] \times \rho^{(tempo\ médio\ espera)_{AB}} \quad (2)$$

Em relação ao coeficiente dos contatos físicos, este consiste, igualmente, em dois coeficientes: o coeficiente do número de contactos físicos $C_{C_{AB}} \in [0,1]$; e o coeficiente da duração desses mesmos contactos físicos $C_{D_{AB}} \in [0,1]$. O primeiro alude à razão entre o número de contactos entre os nós A e B na rede, sobre o número total de contactos com todos os nós da rede que os dois têm em conjunto. O segundo aponta a razão entre a duração total dos contactos entre A e B, sobre a duração total dos contactos com todos os nós da rede que os dois têm em conjunto. δ e α são valores que, somados, têm que dar 1 e que se referem à importância que se pretende dar a cada coeficiente. $\rho \in [0,1]$ é a constante de envelhecimento; e $(tempo\ médio\ espera)_{AB}$ é a média de tempo decorrido entre os contactos dos nós A e B.

Por fim, para relacionar os dois coeficientes fazemos a razão entre eles e retiramos as devidas ilações.

$$Amizade_{AB} = \frac{C_{F_{AB}}}{C_{S_{AB}}} \in [0,1] \quad (3)$$

$$\begin{cases} Amizade_{AB} \approx 1 \rightarrow \text{relação social} \approx \text{relação física} \\ Amizade_{AB} \ll 1 \rightarrow \text{+relação social} \\ Amizade_{AB} \gg 1 \rightarrow \text{+relação física} \end{cases} \quad (4)$$

Já foram realizados os primeiros testes experimentais com estas métricas, contudo, visto tratar-se de um trabalho ainda em desenvolvimento, estes não foram devidamente validados, o que inviabiliza a apresentação de resultados.

VI. CONCLUSÃO

O principal objetivo deste trabalho é verificar a existência de uma relação entre o histórico de contactos físicos e a relação social entre os nós de uma rede RTA, mais concretamente de

uma Pocket Switched Network. Para obter essa relação, foi construído um sistema de recolha e análise de dados de contactos físicos e sociais numa rede tolerante a atrasos que consiste num esquema Cliente-Servidor, onde o cliente é uma aplicação desenvolvida em Android, responsável por recolher os dados relativos aos contactos físicos e ao seu perfil social e enviá-los para o servidor. O servidor é uma aplicação em Java que recebe os dados dos clientes e organiza-os num grafo onde relaciona o campo social com o campo do histórico de contactos. Depois de obtidos os dados, procede-se à análise dos mesmos, através de uma métrica que relaciona os dois campos já referidos: social e físico. Sendo que este trabalho ainda se encontra em desenvolvimento, não foram obtidos os testes para verificar a validade desta métrica. No entanto, no final, pretende-se que o teste seja feito, utilizando para isso um dataset já existente: o dataset SIGCOMM09. Caso seja validada no teste, a métrica calculada pode ser usada futuramente como uma métrica numa decisão de encaminhamento para um protocolo que use informação de contexto social numa RTA.

REFERÊNCIAS

- [1] K. Fall, "A delay-tolerant network architecture for challenged internets," *Proc. 2003 Conf. Appl. Technol. Archit. Protoc. Comput. Commun. - SIGCOMM '03*, p. 27, 2003.
- [2] L. Fratta, M. Gerla, and L. Kleinrock, "The flow deviation method: An approach to store-and-forward communication network design," *Networks*, 1973.
- [3] P. Hui, A. Chaintreau, J. Scott, R. Gass, J. Crowcroft, and C. Diot, "Pocket switched networks and human mobility in conference environments," *Proceeding 2005 ACM SIGCOMM Work. Delay-tolerant Netw. - WDTN '05*, pp. 244-251, 2005.
- [4] A. Pietiläinen, E. Oliver, and J. LeBrun, "MobiClique: middleware for mobile social networking," *2nd ACM Work. Online Soc. networks*, pp. 49-54, 2009.
- [5] A. Pietiläinen and C. Diot, "Dissemination in opportunistic social networks: the role of temporal communities," *MobiHoc '12 Proc. Thirteen. ACM Int. Symp. Mob. Ad Hoc Netw. Comput.*, 2012.
- [6] A. Vahdat and D. Becker, "Epidemic routing for partially connected ad hoc networks," 2000.
- [7] A. Lindgren, A. Doria, and O. Schelén, "Probabilistic routing in intermittently connected networks," *ACM SIGMOBILE Mob. Comput. Commun. Rev.*, vol. 7, no. 3, p. 19, Jul. 2003.
- [8] E. M. Daly and M. Haahr, "Social Network Analysis for Information Flow in Disconnected Delay-Tolerant MANETs," *IEEE Trans. Mob. Comput.*, vol. 8, no. 5, pp. 606-621, May 2009.
- [9] J. M. Pujol, A. Lopez Toledo, and P. Rodriguez, "Fair Routing in Delay Tolerant Networks," *IEEE INFOCOM 2009 - 28th Conf. Comput. Commun.*, pp. 837-845, Apr. 2009.
- [10] P. Hui, J. Crowcroft, and E. Yoneki, "BUBBLE Rap: Social-Based Forwarding in Delay-Tolerant Networks," no. November, pp. 1576-1589, 2011.
- [11] A. Mtibaa and M. May, "Peoplerank: Social opportunistic forwarding," *INFOCOM 2010 Proc. IEEE*, 2010.
- [12] K. Scott, S. Burleigh, and The MITRE Corporation, "Bundle Protocol Specification," pp. 1-51, 2007.
- [13] R. Balani, "Energy Consumption Analysis for Bluetooth, WiFi and Cellular Networks."
- [14] C. Vicknair, M. Macias, Z. Zhao, X. Nan, Y. Chen, and D. Wilkins, "A comparison of a graph database and a relational database," in *Proceedings of the 48th Annual Southeast Regional Conference on ACM SE 10*, 2010, p. 1.
- [15] "Redes Sociais." [Online]. Available: <http://www.ebizmba.com/articles/social-networking-websites>. [Accessed: 13-Sep-2013].