Simulation of 3G/WLAN Offload: First Steps

Nickolay Amelichev
St. Petersburg Electrotechnical University,
Open Source and Linux Lab (OSLL)
Russia, St. Petersburg, 5 Prof. Popova St.
namelichev@acm.org

Kirill Krinkin
St. Petersburg Electrotechnical University,
Open Source and Linux Lab (OSLL)
Russia, St. Petersburg, 5 Prof. Popova St.
kirill.krinkin@fruct.org

Abstract

Our current research concentrates on offloading 3G networks through WLAN — switching from one mobile network to another. The switch occurs if the new network is better than the old by some fitness criteria.

As a part of ongoing research, simulation of handover in pure 3G and WLAN networks is demonstrated, using a custom version of industry-standard network simulator ns-2. Results of basic handover simulation in both WLAN and 3G networks are promising: the delay incurred is reasonable.

Network fitness criterias are also briefly evaluated, and a simple fitness function f to estimate overall quality of wireless 3G and WLAN network is proposed.

I. Introduction

Modern mobile computers, such as Nokia N900, Apple iPhone, and HTC Magic, are equipped with at least two network interfaces: one for cellular network and one for WiFi. Both WiFi and cellular networks have their distinct advantages and disadvantages (table I). For instance, WiFi is faster and cheaper than typical cellular networks, but has poorer quality of service and availability than the cellular.

The main idea of 3G/WLAN offload is to get the best of both worlds: transparently switch to the network N currently having the maximum value of *network fitness function* $f[N]$. This raises three research problems, which should be solved in order:

1) Determine whether transparent switching is possible and reasonably efficient (in terms of handover delay d) for the WiFi↔WiFi, 3G↔3G and 3G↔WiFi switching scenarios.
2) Devise network fitness function f, which reflects network quality (both actual and perceived), user's preferences, and is easy to calculate.
3) Develop schedule for the calculation of network fitness, to cause minimal influence on power consumption of the mobile device.

TABLE I
A brief comparison between different WiFi and broadband cellular networks

Network	Max Downlink, Mbit/s	Typical Cost	QoS Control	Availability	Built-In Handover
Public WiFi (802.11b)	11	free	–	moderate	–
Public WiFi (802.11g)	54	free to low	basic	moderate	–
Public WiFi (802.11n)	600	low to moderate	full	low	–
Cellular 3G (WCDMA)	2	moderate	full	high	+
Cellular 3.5G (HSDPA)	14.4	moderate	full	moderate	+
Cellular 4G (LTE)	173	high	full	very low	+

In this paper we consider only problems 1 and 2. Firstly, the simulation of handover in pure 3G and WiFi networks is discussed. (Simulation of 3G↔WiFi handover is currently in the works.) Secondly, a simple approach to defining fitness function f is proposed.

II. Handover

Transparent switching from one mobile network to the other is called *handover*. Handover typically occurs when current network is out of reach, but it can also be triggered when a better-than-the-current network is detected. In this section handover in pure WiFi and pure 3G networks is briefly discussed. Simulation model demonstrating 3G and WiFi handover in ns-2, developed by us, is presented.

A. Theoretical Considerations

1) WiFi↔WiFi Handover: WiFi standards (802.11 family) do not inherently support handover. The main problem lies in the fact that Access Point (AP) assigns IP address to the client using DHCP. After leaving one AP and joining another there is no chance mobile device has the same IP address. So, the server cannot send data to the mobile device without re-connecting.

There are two solutions to this problem:

1) Use a so-called *mobility protocol*.
2) Use MIH (Media Independent Handover) [1], which is work in progress.

In our work, we considered using Proxy Mobile IPv6 (PMIPv6) mobility protocol [2], because (according to [3], [4]) it is light (almost no wasted bandwidth c_{PMIPv6}) and fast (small handover delays d_{PMIPv6}). It is also a preferred way of organizing 3G-WiFi handover in 3GPP specs [5].

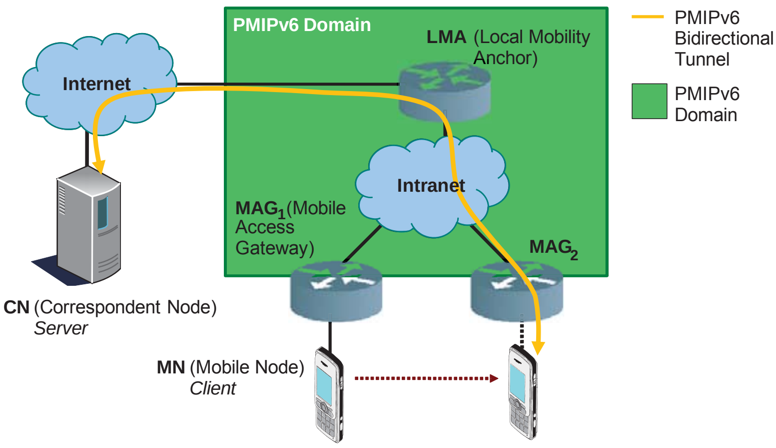

Fig. 1. PMIPv6 Protocol Usage Example

In general, mobility protocols solve the problem with changing IP address by assigning a constant *Home Address* to the receiving side (*Mobile Node*, MN), and introducing a routing entity called *Home Agent*, which tunnels all packets destined at the Home Address to the correct subnetwork.

Proxy Mobile IPv6 is a so-called *local* mobility protocol. It creates a domain (PMIPv6 domain) inside which all data destined or sourced at MNs home address go through the bidirectional tunnel (PMIP tunnel). Entity called LMA (*Local Mobility Anchor*) is *Home Agent* in the PMIPv6 domain. It records information about current MN position and access gateway used, and routes packets going to the *Home Address* to the appropriate MNs *Foreign Address* (Real current address). LMA must be situated just behind the gateway router of the LAN where the PMIPv6 is being deployed (fig. 1).

As soon as MN disconnects from one MAG and connects to the other, that other MAG transmits information (*Proxy Binding Update*) about MNs real address and position to the LMA. (If all goes well, LMA answers by *Proxy Binding Acknowledgement*).

This scheme avoids signaling by the mobile node (unlike many other mobility protocols, notably MIPv6 [6]) — this conserves power, and, more importantly, doesnt require the mobile device to have some special software to support PMIPv6.

2) 3G↔3G Handover: 3G Handover is built-in for the UMTS network and specified in 3GPP 25.308 [7] and 25.331 [8]. We assume that the only type of handover that happens is the change of cell the mobile device is in, and the cells belong to different Node Bs (base stations) connected to a common radio network controller (RNC). (This is so-called *soft handover*). The handover model that we use is based on Serving HS-DSCH Cell Change procedure from 3GPP 25.308, implementation of which in ns-2 simulator is outlined in [9, section 3.3].

3) Estimation of Handover Delay: *Handover delay* d is essentially the time during which no network connection is active. To estimate it, we sample mobile device throughput $\mathrm{BW}[T]$ at regular time intervals T_k, with period $\Delta T = T_{k+1} - T_k = 0.1$ s. Let M be mean throughput. If $\mathrm{BW}[T_k] \ll M$ for $k = \overline{i,j}$ and $j - i > 1$, the time period $\Delta T(j - i + 1)$ is marked as handover period.

Mean handover period, standard error and 95% confidence intervals (assuming a Student's t distribution) are then calculated.

B. Simulation

To demonstrate WiFi↔WiFi handover using PMIPv6 and 3G↔3G handover, we decided to use network simulation, because it was easier than to build a testbed in hardware and properly configure it.

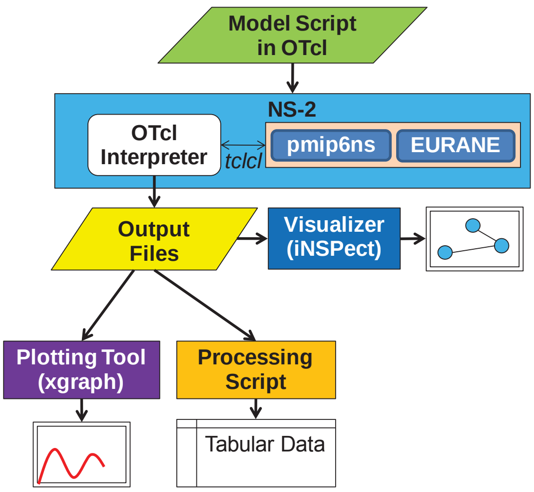

Fig. 2. Model flowchart

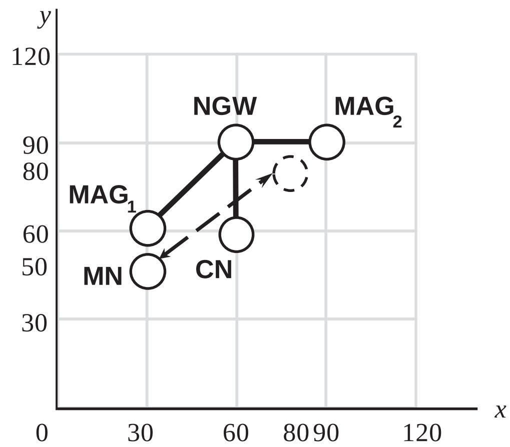

Fig. 3. Model topography and movement

1) Modeling Environment: Simulation was done using a custom version of ns-2 network simulator [10], version 2.33. We integrated three independent packages into standard distribution of ns-2.

- pmip6ns [11] by HyonYoung Choi (to simulate PMIPv6 over WiFi);

- EURANE [12] patched by Abdulmohsen M. Mutairi [9] (to simulate UMTS with multiple cells and handover procedures);
- Q2S [13] by Laurent Paquereau (to use multi-interface nodes in ns-2).

Packages have been substantially altered to work on ns-2 version 2.33 and to have no conflicts with each other. In addition, we have developed several test cases for all of them. Our patch is freely available at [1], with test cases at [2].

Simulation and post-processing of results was done according to the flowchart shown in fig. 2. Simulation scripts were run, resulting network traces were visualized in iNSpect visualizer [14]. Mobile node throughput was graphed, and for handover delays mean value, variance and confidence intervals were calculated.

2) Nodes and Movement: We used a 120×120 m rectangular grid to lay out the nodes of the model (fig. 3). Model had the following nodes:

- *Mobile Access Gateways* MAG_1 and MAG_2. In case of WiFi, two access points were used. In case of 3G, there was a *Radio Network Controller* (RNC) node controlling four base stations (Node Bs) laid out uniformly on the grid. Only two of 3G BSs were active during the simulation.
- *Mobile Node* MN (the client node), moving diagonally from MAG_1 to MAG_2 and back with the velocity $v = 5$ m/s.
- *Correspondent Node* CN (the server node) running the network application connecting to the MN.
- *Network Gateway* NGW, which corresponded to PMIPv6 LMA (Local Mobility Anchor) entity for WiFi, and to GGSN and SSGN nodes for 3G.

3) Traffic Generation: To generate traffic, a simulated network application was run on the correspondent node (CN), and periodically sent packets to the mobile node (MN).

Two application types were studied.

- TCP application: FTP server sending data at the rate of 1 Mbit/s.
- UDP application: constant bitrate server sending data at the rate of 1 Mbit/s.

4) Expected Handovers: We decided that the simulation should last 100 s. During this time, MN moves diagonally with velocity $v = 5$ m/s. Movement phase takes 12 s. After the movement phase, MN stands still for 12 s, and then resumes movement. We supposed that this would lead to 4 approximately same-sized handover intervals d, reflected in mobile node throughput measurements (fig. 4).

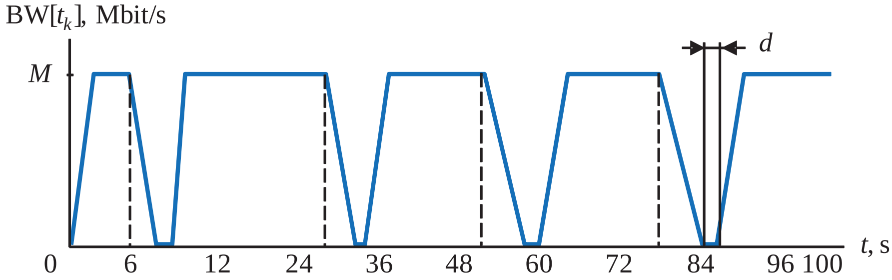

Fig. 4. Expected mobile node throughput graph

[1] http://osll.spb.ru/repositories/browse/bsc-amelichev/patch
[2] http://osll.spb.ru/repositories/browse/bsc-amelichev/model/test

C. Simulation Results

Simulation results for mobile node throughput $\mathrm{BW}[T]$ displayed good agreement with our expectations (4 distinct handovers lasting approximately the same time).

The delays d calculated from throughput measurements have relatively low accuracy, because we focused primarily on demonstration of handover and not on measurement of its numerical characteristics. The delays we calculated, though imprecise, suggest that handover procedures studied are quite reasonable to use. This confirms qualitative results of other authors studying handover (for instance, [4]).

1) WiFi↔WiFi: WiFi↔WiFi handover using PMIPv6 mobility protocol had been demonstrated in simulation (fig. 5, 6). Our findings show that it has a delay of $d_{\mathrm{WiFi-TCP}} = (500 \pm 300)$ ms for TCP traffic and $d_{\mathrm{WiFi-UDP}} = (400 \pm 200)$ ms for UDP traffic. Note that variation of handover delay is almost certainly caused by insufficient number of WiFi handovers studied.

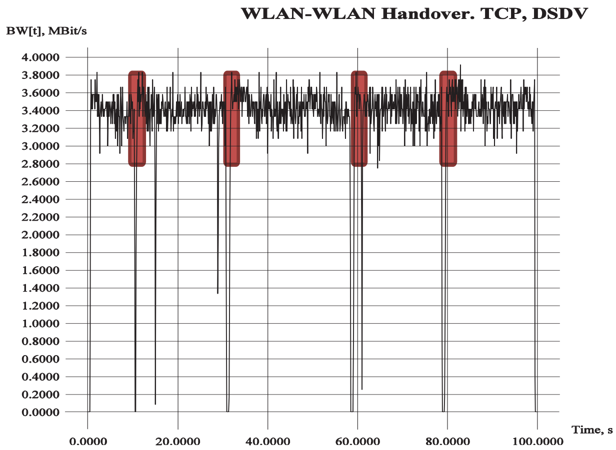

Fig. 5. MN Throughput in a pure WiFi network for TCP traffic. Handovers highlighted

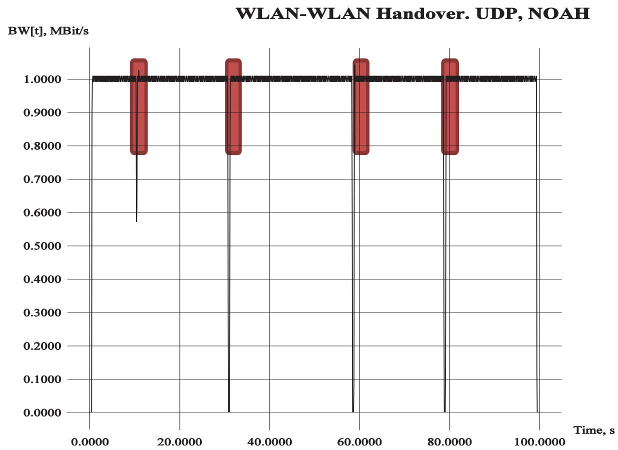

Fig. 6. MN Throughput in a pure WiFi network for UDP traffic. Handovers highlighted

2) 3G↔3G: 3G↔3G soft handover had been demonstrated in simulation (fig. 7, 8). Our findings show that it has a delay of $d_{\mathrm{3G-TCP}} = (103 \pm 5)$ ms for TCP traffic (Acknowledged Mode) and $d_{\mathrm{3G-UDP}} = (110 \pm 30)$ ms for UDP traffic (Unacknowledged Mode). Low variation of handover delay might be caused by simplified handover procedure adopted.

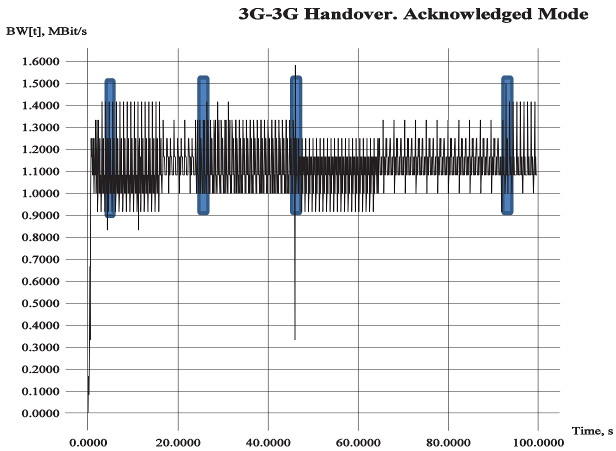

Fig. 7. MN Throughput in a pure 3G network in Acknowledged Mode. Handovers highlighted

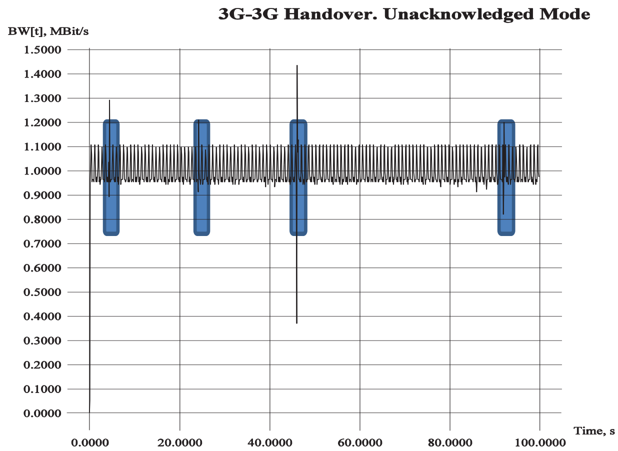

Fig. 8. MN Throughput in a pure 3G network in Unacknowledged Mode. Handovers highlighted

III. Network Fitness Function

We propose the following polynomial as the network fitness function:

$$f = \sum_{i}^{n} w_{1i}\mathrm{FIP}_i^2 + \sum_{j}^{m} w_{2i}\mathrm{SIP}_j - \mathrm{PENALTY}, \tag{1}$$

where $\mathrm{FIP}_i \in [1,5]$ $(i = \overline{1,n})$ are *first importance parameters*, $\mathrm{SIP}_j \in [1,5]$ $(j = \overline{1,m})$ are *secondary importance parameters*, $w_{kl} \in [0,1]$ are *parameter weights* and $\mathrm{PENALTY} \in [0,(25n+5m)/2]$ is a *static penalty* applied to networks known to incur high costs on the user.

There are a few important moments to note.

1) As (1) is a polynomial, it can be evaluated quickly and efficiently.
2) For situations with low mobile device power, we can easily skip evaluation of any parameter by simply setting its weight w_{kl} to zero. This will yield an approximation to network fitness score.
3) Weights can be easily changed for (1) to reflect personal preferences of the mobile device user, or current state of the device (e.g. velocity).

Selection of network parameters included in (1), assignment of weights and penalty are discussed in the following sections.

A. Parameters

For evaluation of fitness function (1) to be fast and efficient, we selected only the network parameters, which are directly available to the network driver or can be easily calculated:

- signal power, dBm;
- signal to noise ratio (SNR), dB;
- maximum downlink speed, Mbit/s;
- time since the network was detected (time of existence), min;
- packet loss, %;
- packet delay (latency), ms;
- jitter (packet delay variation), ms.

Each of these parameters is assigned a rank from 1 to 5 (from ”unsatisfactory” to ”excellent”). The ranks can be tuned by the device manufacturer. Suggested default ranks, borrowed from [15], [16], [17], are given in sections III-A2 and III-A3.

1) Parameter Caching: In concrete implementation, for maximum efficiency in evaluation of (1), caching of parameters should be employed. Also, changes should be propagated only when needed. For instance, for a WiFi network updating of signal power and maximum downlink speed should occur right after receiving 802.11 BEACON and/or PROBE RESPONSE frames.

Evidently, packet loss, packet delay and jitter can only be calculated only while connected to the network being estimated. Moreover, they should be calculated and cached and only if user has explicitly set the ”Favor VoIP-friendly networks” (or some similar) option.

2) First Importance Parameters: First Importance Parameters reflect the performance and persistence of the network being evaluated. All of them can be easily obtained or calculated one from the other.

TABLE II
SUGGESTED RATINGS FOR DIFFERENT VALUES OF SIGNAL POWER.

Signal Power, dBm	Rating	Comment
> -30	5	Max received power for 802.11
$[-30, -45]$	4	
$(-45, -60]$	3	
$(-60, -80]$	2	Typical received power for 802.11
< -80	1	Background noise

TABLE III
SUGGESTED RATINGS FOR DIFFERENT VALUES OF SNR.

SNR, dB	Rating	Comment
> 40	5	Excellent signal, very fast
$[25, 40]$	4	Healthy. Very good signal, fast
$[15, 25)$	3	Low signal, but usually fast
$[10, 15)$	2	Very low signal, slow. Connecting is difficult
< 10	1	No signal

TABLE IV
SUGGESTED RATINGS FOR MAXIMUM DOWNLINK SPEED.

Max Downlink Speed, MBit/s	Rating	Comment
> 300	5	802.11n (in theory)
$[100, 300]$	4	LTE, HSDPA, 802.11n
$[40, 100)$	3	802.11g, UMTS
$[1, 40)$	2	802.11b, 802.11g
< 1	1	GPRS, EDGE

TABLE V
SUGGESTED RATINGS FOR TIME OF NETWORK EXISTENCE.

Time of Existence, min.	Rating	Comment
> 30	5	Persistent Network
$[15, 30]$	4	
$[5, 15)$	3	
$[1, 5)$	2	
< 1	1	Unpromising

a) Signal Power: This parameter estimates packet reception power. In ns-2 simulator, it is calculated using one of the various propagation models (Shadowing, TwoRayGround, etc.) and retrieved from the txInfo_→RxPr_ field of the received packet. It is more convenient to rank signal power in decibels above milliwatt (dBm), so RxPr_ needs to be converted from W to dBm using the formula

$$\mathrm{SPdBm} = 10\log_{10}[\mathrm{RxPr\_}] + 30. \tag{2}$$

Table II represents ratings for signal power.

b) SNR: Signal-to-noise ratio is one of the main characteristics of reception, showing whether signal power is enough to discern meaningful information from the noise. It is calculated as

$$\mathrm{SNR} = 10\log_{10}\left[\frac{\mathrm{RxPr\_}}{\mathrm{NoisePr}}\right] = \mathrm{SPdBm} - \mathrm{NPdBm}, \tag{3}$$

where SPdBm is packet reception power in dBm, $\mathrm{NoisePr} = 10^{-12}$ W, $\mathrm{NPdBm} = 90$ dBm is typical noise power [16].

Table III represents ratings for SNR taken from [16].

c) Maximum Downlink Speed: Maximum downlink speed as advertised by the network. Ratings (table IV) are based on maximum theoretical and practical values for different WiFi and cellular networks [18], [19], [20], [21]. Maximum supported speed can be determined from 802.11 BEACON and 802.11 PROBE RESPONSE frames for WiFi, and from radio network controller (RNC) information in 3G networks.

d) Time of existence: On the first moment of new network detection, the existence time $T[N]$ for it is set to 0. On the next moments, if the network is still present, update period T_U is added to $T[N]$; otherwise, all records about network N, including $T[N]$, are deleted. It is recommended that weight of $T[N]$ in (1) be enlarged in case of fast movement (device velocity $v > 15$ m/s). Table V represents ratings for $T[N]$.

3) Secondary Importance Parameters: Secondary Importance Parameters deal with suitability of network to high quality-demanding applications, such as VoIP, streaming music and video, and interactive network games. All of them can be calculated *only after connecting to the network* at either the network driver level, or at the socket level, depending on the implementation.

TABLE VI
SUGGESTED RATINGS FOR PACKET LOSS.

Packet Loss, %	Rating
< 0.05	5
$[0.05, 0.3]$	4
$[0.3, 0.5]$	3
$[0.5, 1]$	2
> 1	1

TABLE VII
SUGGESTED RATINGS FOR ONE-WAY PACKET DELAY.

Packet Delay, ms	Rating
< 20	5
$[20, 50]$	4
$(50, 100]$	3
$(100, 150]$	2
> 150	1

TABLE VIII
SUGGESTED RATINGS FOR JITTER.

Jitter, ms	Rating
< 0.1	5
$[0.1, 0.6]$	4
$(0.6, 0.9]$	3
$(0.9, 2]$	2
> 2	1

a) Packet Loss: Expresses, in %, a ratio of network packets sent from the device, but not received at their destination, to the total number of packets sent from the device. In ns-2 network simulator, it can be calculated from drop (number of dropped packets) and tx (number of packets sent from the device):

$$\mathrm{PLR} = \begin{cases} \frac{\mathrm{drop}}{\mathrm{tx}} 100\%, & \mathrm{tx} \neq 0 \\ 0\%, & \mathrm{tx} = 0 \end{cases} . \tag{4}$$

Packet loss is typically updated after sending and receiving a certain large number of packets. In ns-2 simulator, packet loss can be re-calculated at each packet's drop and send events.

Ratings for packet loss in table VI are laid out according to Cisco recommendations for VoIP networks, cited in [17].

b) Packet Delay: This parameter represents mean time that is needed for a ready-to-send network packet to arrive at its destination. The delay for a successfully transmitted packet is defined in [22] as the time interval from the moment T_{head} the packet is at the head of the queue, until the moment T_{ack} acknowledgement for this packet is received:

$$\mathrm{PDRT} = T_{\text{ack}} - T_{\text{head}}. \tag{5}$$

Time delay for dropped packets is not included in the calculation of mean packet delay.

From this moment on, we would deal with *one-way packet delay*

$$\mathrm{PD} = \frac{\mathrm{PDRT}}{2}. \tag{6}$$

The reason for this is that one-way packet delay is given in standards, such as ITU-T G.114 [23], and provider service-level agreements (for example, mentioned in [17]).

Note. Link delay, if it is relatively constant, can be included in calculation of PD to improve its accuracy.

Ratings for packet delay are given in table VII. The acceptable values for packet delay given by service providers are borrowed from [17]. 150 ms delay is chosen for ”weak” (2/5) rating of performance. This value is maximum acceptable delay for the *entire* voice path as per ITU-T G.114 [23].

c) Jitter: We would use the term *jitter* to describe variation of packet delay:

$$\mathrm{J} = \max \mathrm{PD} - \min \mathrm{PD}. \tag{7}$$

Jitter causes unpleasant effects in the audio and video live streams. To compensate for varying packet delay, most VoIP endpoint devices have *jitter buffers*, but these have a finite length and are effective with delay variations of less than 100 ms (Cisco quoted in [17]). Backbone providers usually guarantee jitter not to exceed 2 ms ([17]), but ISPs and local network could both incur additional jitter.

Ratings for jitter are given in table VIII. The acceptable values for jitter given by service providers are borrowed from [17].

B. Determining Weights

In the initial stages of development, weights w_{kl} in (1) could be set to $w_{1i} = 1/n$, $w_{2i} = 1/m$ for the sake of simplicity.

In real-world scenarios, however, weights should be adjusted. This can be either done through *expert analysis* or through *user survey*.

In case of expert analysis, each expert would rank the relative importance of each parameter, and then statistical analysis of the answers would be made, determining weight of each parameters according to the experts' answers.

In case of user survey, a sufficiently large number of users (around $N_{\text{users}} = 50$) will be asked to rank a number of networks (around $N_{\text{net}} = 5$) given values of its first and secondary importance parameters. Then the least squares fit will be performed to find the values of the weights w_{kl} giving minimal deviation of formula (1) from the survey data. Recommended non-technical names of the parameters to be used in the survey are given in table IX.

TABLE IX
RECOMMENDED NAMES FOR NETWORK PARAMETERS IN A USER SURVEY

Parameter	Nontechnical Name
signal power	distance to base station/access point
signal to noise ratio	signal strength
maximum downlink speed	download speed
time of network existence	network availability
packet loss	network response times
packet delay	persistent delays in games, VoIP, streaming video
jitter	random short-lived distortion of video and audio streams

C. Penalty

We apply a penalty if connecting to and using the network is potentially costly to the user. These are (including, but not limited to):

- fixed penalty for 3G network that is not the home network (obviously, roaming is costly);
- fixed penalty if the WiFi network supports only WPA2 with certificate authentication. Such networks do not permit authentication if the user certificate is nonexistent. Even if a valid certificate for the user exists, digital certificate verification is rather power-consuming. So it might be better to act conservatively and avoid such networks altogether (at least in presence of free WiFi networks).

Applying a penalty is optional. The range of each penalty is to be selected by the device vendor, but the sum of penalties applied should not exceed $(25n + 5m)/2$ (half of maximum network fitness score).

IV. Further work and investigation areas

This paper considers reasonable handover parameters between 3G and WLAN nets. Splitting control and workload traffic is an interesting problem as well. Mobile node can forward all control traffic like ICMP or Routing traffic through 3G channel, as it provides better QoS. One more research direction could be investigation of ability to provide seamless work through the mesh (IEEE802.11s) networks. In this particular case we could investigate possible procedures for conducting MAC and IP level routing. Those topics will be discovered in future works.

V. Conclusion

We have completed the two most significant tasks in simulation of offloading 3G networks through WiFi.

Firstly, we have demonstrated handover in pure WiFi and 3G networks. We remark that our estimates of handover delays d_{PMIPv6} and d_{3G} are reasonable for most network applications, except the apps sensitive to delays (such as VoIP or streaming video). It should also be noted that handover delays for TCP and UDP traffic don't differ very much, which is understandable given that handover works at and underneath the MAC layer.

Secondly, we have proposed the equation (1) for the network fitness function, which is easy to evaluate and easy to change to different network parameter sets and user personal preferences.

However, much remains to be done. We would concentrate on weighting network parameters (perhaps using the user survey method), extensive testing of the network fitness function on network models and real wireless networks, and development of the optimal fitness function calculation schedule. Also, in the nearest future 3G$\leftrightarrow$WiFi handover will be demonstrated, and a number of network offloading scenarios involving a massive number of wireless networks (both WiFi and 3G) will be tried out in simulation.

Acknowledgement

The authors would like to thank Finnish-Russian University Cooperation in Telecommunication Program (FRUCT, fruct.org) for provided support and Nokia Siemens Networks experts for their consultacy work. This work has been developed in Open Source and Linux Lab (osll.fruct.org).

References

[1] IEEE 802.21. http://www.ieee802.org/21/, 2010. [Online; accessed 26-September-2010].

[2] S. Gundavelli, K. Leung, V. Devarapalli, K. Chowdhury, and B. Patil. *RFC5213. Proxy Mobile IPv6*, 2008.

[3] X. Fu and J. Lei. Evaluating the Benefits of Introducing PMIPv6 for Localized Mobility Management, Technical Report No. IFI-TB-2007-02. Technical report, Institute of Computer Science, University of Goettingen, Germany, 2007.

[4] A. Udugama. Evaluation of a Network based Mobility Management Protocol : PMIPv6. In *Proceedings of Vehicular Technology Conference*, 2009.

[5] *3GPP 23.402 "Technical Specifcation Group Services and System Aspects, Architecture enhancements for non-3GPP Accesses" (Release 9)*, 2009.

[6] D. Johnson, C. Perkins, and J. Arkko. *RFC3775. Mobility Support in IPv6*, 2004.

[7] *3GPP 25.308 High Speed Downlink Packet Access; Overall Description; Stage 2 (Release 7), v7.3.0*, 2007.

[8] *3GPP 25.331 "RRC Protocol Specification" (Release 9)*, 2009.

[9] A. Mutairi and U. Baroudi. Ns-2 enhancements for detailed hsdpa simulations. In *Proceedings of the 3rd International Conference on Performance Evaluation Methodologies and Tools*, 2008.

[10] ns-2. the network simulator, version 2.33. http://www.isi.edu/nsnam/ns/, 2010. [Online; accessed 26-September-2010].

[11] H. Choi. Proxy Mobile IPv6 in ns-2.29. http://commani.net/pmip6ns, 2009. [Online; accessed 26-September-2010].
[12] Enhanced UMTS Radio Access Network Extensions. http://eurane.ti-wmc.nl/eurane/, 2007. [Online; accessed 26-September-2010].
[13] Laurent Paquereau. Extensions to ns-2. http://www.q2s.ntnu.no/~paquerea/ns.php, 2009. [Online; accessed 26-September-2010].
[14] S. Kurkowski, T. Camp, M. Collagrosso, and N. Mushell. A Visualization and Analysis tool for ns-2 Wireless Simulations : iNSpect. In *MASCOTS'05: Proceedings of the 13th IEEE International Symposium on Modeling, Analysis, and Simulation of Computer and Telecommunication Systems*, pages 503–506, 2005.
[15] dBm. http://en.wikipedia.org/wiki/DBm, 2010. [Online; accessed 26-September-2010].
[16] Jim Geier. SNR Cutoff Recommendations. http://www.wi-fiplanet.com/tutorials/article.php/3468771/SNR-Cutoff-Recommendations.htm, 2005. [Online; accessed 26-September-2010].
[17] QoS - voip-info.org. http://www.voip-info.org/wiki/view/QoS, 2010. [Online; accessed 26-September-2010].
[18] 802.11b. http://en.wikipedia.org/wiki/802.11b, 2010. [Online; accessed 26-September-2010].
[19] 802.11n. http://en.wikipedia.org/wiki/802.11n, 2010. [Online; accessed 26-September-2010].
[20] UMTS. http://ru.wikipedia.org/wiki/UMTS, 2010. [Online; accessed 26-September-2010].
[21] 3GPP Long Term Evolution. http://en.wikipedia.org/wiki/3GPPLongTermEvolution, 2010. [Online; accessed 26-September-2010].
[22] P. Raptis, V. Vitsas, K. Paparrizos, P. Chatzimisios, A. C. Boucouvalas, and P. Adamidis. Packet delay modeling of ieee 802.11 wireless lans. In *Proceedings of CITSA*, 2005. http://pelopas.uop.gr/~acb/PacketDelayModelingofIEEE802.11WirelessLANs.pdf.
[23] *ITU-T G.114 Recommendation. One-way Transmission Time*, 2003.