Aalborg Universitet

AALBORG UNIVERSITY
DENMARK

Location Assisted Handover Optimization for Heterogeneous Wireless Networks

Nielsen, Jimmy Jessen; Madsen, Tatiana Kozlova; Schwefel, Hans-Peter

Location Assisted Handover Optimization for Heterogeneous Wireless Networks

Jimmy Jessen Nielsen[†], Tatiana K. Madsen[†], Hans-Peter Schwefel[†‡]

[†] Networking and Security, Department of Electronic Systems, Aalborg University

[‡] Forschungszentrum Telekommunikation Wien - FTW, Vienna, Austria

jjn@es.aau.dk, tatiana@es.aau.dk, schwefel@ftw.at

Abstract—Mobile users typically experience better connectivity if their mobile device performs handover to an available WiFi network rather than using a cellular network. For a moving user the window of opportunity is limited and the timing of the handover is therefore crucial.

In this work we propose two location-based look-ahead handover prediction algorithms that are based on the assumption that a database of expected throughput for a given location of all networks is available. The first algorithm uses an analytical formulation of the handover problem to determine the optimal sequence of handovers within a time window, which is computationally feasible for up to 3-4 handovers within the window. The second algorithm is a heuristic algorithm, which is computationally feasible for any reasonable number of handovers within the window. We have used simulations to obtain the achieved throughput of these algorithms for a mobile user in an urban scenario with ubiquitous cellular coverage and 250 WiFi APs/km^2, and compared the results to a hysteresis-based greedy algorithm and the case of "always cellular-connected".

Our results show that the proposed look-ahead algorithms outperform the hysteresis-based and "always cellular-connected", but also show that the look-ahead algorithms are highly dependent on accurate movement tracking and movement prediction systems. The heuristic algorithm is also shown to achieve the highest throughput for large look-ahead windows.

I. INTRODUCTION

WiFi networks typically provide mobile users with better connectivity in terms of e.g. throughput than cellular networks, as shown in the two experimental performance comparisons of 3G and Wi-Fi in [1] and [2]. Here it is clear that if a terminal is able to use the network that offers the highest throughput at any given time, an overall performance increase is possible. To make use of the best network at any given time, the user terminal needs to handover between the cellular network and the available WiFi networks. A handover is the process of switching from one network to another, e.g., from the cellular network to a WiFi network or between two WiFi networks. Every such handover involves several steps such as access point (AP) association, DHCP look-up, and IP configuration, which require time and leads to a signaling overhead. It is therefore preferred that the amount of handovers is limited, which makes it important to choose the networks to use wisely.

Thus, an important functionality for mobile devices intended for use in heterogenous network scenarios is the handover (HO) algorithm, which should maximize the user's benefit of the available networks but also limit the number of handovers. Depending on the user's preferences and applications, the benefit could for instance be to achieve a higher throughput than what the cellular network provides.

In this paper we investigate how the knowledge of location information can help in the multisystem HO decision. The main problem we address is how location information can be used to guide a mobile device's selection between the ubiquitous cellular network and any locally available WiFi networks. A main assumption in this work is the availability of a database that contains the average throughput of all available networks at any position for the considered geographical area.

In the literature some HO decision algorithms based on location information already exist. In [3] the authors present a survey of vertical HO decision strategies. One example of exploiting location information is given in [4], where a Location Server Entity provides information such as coverage area, bandwidth and latency of nearby wireless networks, which is used by the mobile terminal for power management and HO selection. In [5] the authors propose a HO algorithm based on neural networks. The method is superior compared to simple RSS threshold and hysteresis-based schemes, however the algorithm requires substantial training beforehand in form of RSS traces and desired outputs. Both [4] and [5] are reactive schemes where the HO decision is based on instantaneous conditions and does not take into account the expected future conditions. The authors of [6] and [7] have both formulated the HO decision as Markov decision processes, taking into account different parameters such as: connection duration, QoS parameters, location information and predicted movements, network access cost, and the signalling load incurred on the network. The proposed algorithms are shown to work well compared to state of the art algorithms. However, in both [6] and [7] the authors assume that the achievable throughput of the available networks are constant within the coverage region, which is not the case in practice [1].

In the present contribution, we propose two HO decision algorithms that work proactively by using movement prediction to plan the HO ahead in time. Also, the algorithms use continous throughput functions to accurately determine the best point for a user terminal to handover along the expected

movement trajectory.

II. System Model

We consider a single multi RAT terminal with location $X(t) = [x(t), y(t)]^T$ at time t. This location is however not known, but it is estimated as $\hat{X}$ using a localization system, which can be GPS or network-based. The mobile terminal's past movement trajectory given by $\hat{X}(t \leq t_c)$, where t_c is the current time, is based on interpolation of previous location estimates. The mobile terminal's predicted future movement trajectory $\hat{X}^*(t_c \leq t \leq t_c + W)$, where W is a time window specifying the prediction horizon, is based on an extrapolation of the past trajectory, t_h seconds back in time. In the following we denote W as the *look-ahead window*.

By connecting to an access point or a base station with index a, whose coordinates are known a priori, the mobile terminal at location $\hat{X}$, at time t, achieves a throughput Ω:

$$\Omega_a(t) = S_a(X(t)) + V \tag{1}$$

where the random variable V accounts for variations in the actual throughput, caused by non-deterministic factors such as small and medium-scale fading. We assume that V is a zero-mean gaussian stochastic variable, characterized by the standard deviation σ_{TP}.

The expected throughput for a network a is achieved from the database by the approximation:

$$\Omega_a^*(t) = S_a(\hat{X}^*(t)) \tag{2}$$

where $S_a(\hat{X}^*(t))$ is the expected throughput at the predicted location $\hat{X}^*(t)$ at time t, given the path-loss of the link between the terminal and the AP or BS. Notice that currently the available resources of APs or BSs is determined purely from the path-loss, and does not depend on the load caused by other users. This point will be discussed later in this paper.

The mobile terminal can choose to connect to a different AP/BS by performing a handover (HO). Within the time window $[t_c; t_c + W]$, the terminal may perform a sequence of HOs $\mathcal{H}$, defined as:

$$\begin{aligned} \mathcal{H}^K = &\{(a_i, t_i), \ i = 1 \ldots K\}, \tag{3} \\ &a_{i-1} \neq a_i, \\ &t_c < t_1 < t_2 < \ldots < t_K < t_c + W \end{aligned}$$

which describes a HO to network a_i at time t_i, where index K denotes the number of HOs in the sequence. Notice that the target network in a handover is never the same as the source network. Each performed HO in the sequence $\mathcal{H}$, may incur a cost due to lost connectivity while switching from one network to another, since this requires steps such as AP association, Dynamic Host Configuration Protocol (DHCP) look-up, and Internet Protocol (IP) address configuration. In this work we assume that this cost denoted C_a is a downtime or handover delay, where the throughput is zero that depends only on the target network. The actual handover cost will in practice also include a certain signaling overhead, which motivates to keep the number of handovers low. As the achieved throughput
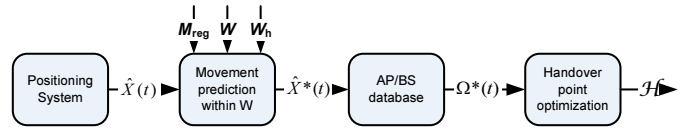

Fig. 1. Look-ahead prediction algorithm principle. M_{reg} is the order of regression, W is the look-ahead window length, t_h is the look-back window length, and M_{poly} is the polynomial approximation order.

is zero for the duration of each handover, optimizing for the highest throughput should also result in relatively few handovers for not too small values of C_a.

In order to experience the best performance in terms of throughput, the mobile terminal needs to determine when is the best time to perform HO(s) and which network(s) to connect to, taking into account the handover cost C_a.

III. Look-ahead Prediction Algorithms

In the following, we will describe the considered HO decision algorithms. The first two algorithms are so-called look-ahead prediction algorithms that determine a HO sequence $\mathcal{H}$, as depicted in Fig. 1. The main assumption for these algorithms is that a fingerprinting database that contains the average throughput of all available networks for the considered geographical area is available. The third algorithm, which is described in sec. V-B, is a hysteresis-based greedy algorithm.

A. Optimal K-Handover Look-ahead Algorithm

The optimal sequence of HOs that maximizes the throughput within the time window W, with exactly K HOs, may be defined as:

$$\mathcal{H}_{opt}^K = \arg\max_{\mathcal{H}^K}(f(\mathcal{H}^K)) \tag{4}$$

$$f(\mathcal{H}^K) = \int_{t_0}^{t_1} \Omega_{a_0}(t)dt + \sum_{i=1}^{K} \left(\int_{t_i+C_{a_i}}^{t_{i+1}} \Omega_{a_i}(t)dt \right) \tag{5}$$

where $t_{K+1} = t_0 + W$. The integration of $\Omega_a(t)$ over time corresponds to the throughput experienced when connected to network a along the predicted movement trajectory, so $f(\cdot)$ is the total throughput achieved within W for a given HO sequence. Notice that we determine the optimal $\mathcal{H}_{opt}^K$ for each of the cases $K = \{1, 2, \ldots K_{max}\}$ separately, and then select the best number of HOs:

$$K_{opt} = \arg\max_{K}(f(\mathcal{H}_{opt}^K)), \ K = 1 \ldots K_{\max}. \tag{6}$$

This is solved by iterating over all considered values of K and selecting K_{opt} as the K that leads to the highest throughput.

In order to determine the optimal HO sequence for a value of K, we consider all possible combinations of networks to handover from and to, as well as the candidate HO times, which will be defined subsequently.

The N possible network combinations are:

$$A_n = \{a_0^n \ a_1^n \cdots a_i^n \cdots a_K^n\}, \ n = 1 \ldots N, \tag{7}$$
$$a_{i-1}^n \neq a_i^n,$$
$$a_0^n = a_0^j, j = 1 \ldots N.$$

For every HO i between two consecutive networks in A_n, there is a set of M time instants that are candidates for optimal HO points between these two networks. This set is defined as:

$$T_n^i = \{t_i^{n,1} \ t_i^{n,2} \cdots t_i^{n,m} \cdots t_i^{n,M}\}. \tag{8}$$

Each unique combination of networks and time instants from A_n and $T_{n,m}$ constitute a unique sequence:

$$\mathcal{H}_{n,m}^K = \{(a_i^n, t_i^{n,m}), \ i = 1 \ldots K\}. \tag{9}$$

Now, given the n'th combination of networks A_n we can determine the set of candidates for optimal handover points T_n^i, as the t_i's that satisfy:

$$\frac{df(\mathcal{H}_n^K)}{dt_i} = 0 \tag{10}$$

since these points result in either maxima or minima for $f(\mathcal{H}_n^K)$, which expresses the total throughput in W. Such optimization by differentiation of course requires that the throughput functions $\Omega_a(t)$ are continuously differentiable.

Differentiation of $f(\mathcal{H}_n^K)$ with respect to t_i, reduces to:

$$\frac{df(\mathcal{H}_n^K)}{dt_i} = \Omega_{a_{i-1}}(t_i) + \Omega_{a_i}(t_i + C_{a_i}). \tag{11}$$

Setting this expression equal to zero and finding all solutions for every t_i gives the candidate handover points T_n for the n'th combination of networks.

In short, the complete algorithm can be described as shown in Algorithm 1, where the function `generate_combinations`(N_a, K, a_0) generates all possible combinations of networks as specified in eq. (7). N_a is the number of networks available within W, and K is the number of allowed HOs with W. Further, N is the number of network combinations in A.

The HO algorithm is run periodically, looking a time W ahead and looping over the $K_{\max}$ possible handovers to determine the optimal number of handovers as described in eq. (6). The HOs are done as planned, and after time W has passed since last run, the algorithm is run again. However, due to uncertainties caused by localization inaccuracy and unknown future movements, the predicted behaviour, which is used to calculate a HO sequence, is expected to become less trustworthy with increasing lengths of the look-ahead window.

Algorithm complexity: The determining factor for the complexity of the algorithm is the number of different network combinations that the algorithm is trying out.

Considering the constraint in eq. (7): that a HO is always to a different network than the current, the number of entries in A_n becomes:

$$N = (N_a - 1)^{K+1}. \tag{12}$$

---

for $K = 1 \ldots K_{max}$ **do**
  $A = $ `generate_combinations`(N_a, K, a_0)
  for $n = 1 \ldots N$ **do**
    $\mathcal{H}_n^K = A_n$
    for $i = 1 \ldots K$ **do**
      Solve $\frac{df(t, \mathcal{H}_n^K)}{dt_i} = 0 \ \rightarrow \ T_n^i$
    end
    for $m = 1 \ldots M$ **do**
      $\mathcal{H}_n^K = \{A_n, T_{n,m}\}$
      $s(K, n, m) = f(t, \mathcal{H}_{n,m}^K)$
    end
  end
end
$(K, n, m) = \max\{s(K, n, m)\}$
$\mathcal{H}_{\text{opt}} = \mathcal{H}_{n,m}^K$

Algorithm 1: Optimal K-HO algorithm.

From this it is clear, that for a large number of available networks, trying out all combinations can become infeasible as the number of HOs K increases. In order to keep the complexity low, we consider only the networks in W whose expected throughput exceeds that of the cellular network, hereby reducing N_a.

B. Heuristic Look-ahead Algorithm

In addition to the previous algorithm that determines the optimal K-HO solution within the window W, we also consider a less complex heuristic look-ahead algorithm.

This algorithm always tries to handover to the network with the highest expected troughput; however, only if the handover gain exceeds the cellular network throughput by more than a threshold ρ.

Assume that $\{t_1 \ t_2 \ \cdots \ t_j \ \cdots \ t_J\}$ where $t_j < t_{j+1}$ is a list of timestamps for when the network with the highest expected throughput, $a^{\max}$, changes. Then the preferred network for the j'th timespan $(t_j; t_{j+1}]$ is:

$$a_j^{\text{pref}} = \begin{cases} a_j^{\max} & \text{if } \int_{t_j}^{t_{j+1}} (\Omega_{a_j^{\max}}(t) - \Omega_1(t)) dt - \rho > 0 \\ 1 & \text{otherwise} \end{cases} \tag{13}$$

where $\Omega_1(t)$ is the throughput function of the cellular network and ρ is a threshold, used to filter out unhelpful HOs, which is set as defined in Table I.

Finally, the heuristic sequence is:

$$\mathcal{H}_{\text{heu}} = \{(a_j^{\text{pref}}, t_j), \ t = 1 \ldots J\} \tag{14}$$

Notice that for simplicity, this heuristic algorithm does not take the HO delay into account when calculating the timestamps in the HO sequence.

C. Movement prediction

For the look-ahead HO algorithms we use a linear movement prediction algorithm that uses historical location measurements within a time window W_h to predict the direction and speed of the mobile user, W seconds ahead.

First, the direction of movement is determined using a 1st order Total Least Squares (TLS) regression. TLS is used, since it minimizes the perpendicular distance to the regression line and not the vertical distance as ordinary Least Squares (LS) does. The TLS is realized using the Principal Component Analysis (PCA) method [8]. The PCA method gives a vector of unit length along which, the variance of the data is the highest. But the PCA does not tell if the movement direction is along or opposite the resulting vector, denoted $\hat{\mathbf{w}}$. Therefore we use the first and last historic data points to determine the sign of the direction vector as:

$$\hat{\mathbf{u}} = \begin{cases} \hat{\mathbf{w}} & \text{if } \text{sign}((\mathbf{a}^T \hat{\mathbf{w}})\hat{\mathbf{w}}) = \text{sign}(\hat{\mathbf{w}}) \\ -\hat{\mathbf{w}} & \text{otherwise} \end{cases} \quad (15)$$

where $\mathbf{a}$ is the last and first historic data points subtracted:

$$\mathbf{a} = \hat{X}(t_{\mathrm{c}}) - \hat{X}(t_{\mathrm{c}} - W_{\mathrm{h}}) \quad (16)$$

where $\hat{X}(t_{\mathrm{c}})$ is the estimated current location and $\hat{X}(t_{\mathrm{c}} - W_{\mathrm{h}})$ is the oldest location estimate in the look-back window.

The average speed $\bar{\mathbf{v}}$ is determined from the average distance between the projections of the historical data points in matrix $\mathbf{B}$ onto the direction vector:

$$\mathbf{b} = \mathbf{B}\hat{\mathbf{w}} \quad (17)$$

$$\bar{\mathbf{v}} = \frac{\sum_{i=2}^{N_{\mathrm{h}}} \mathbf{b}(i) - \mathbf{b}(i-1)}{N_{\mathrm{h}} - 1} \quad (18)$$

where, in $\mathbf{B}$, each row contains an x, y coordinate pair, and N_{h} is the number of historical data points.

D. Implementation Considerations for Look-ahead Algorithms

The nature of the K-HO optimal and heuristic look-ahead prediction algorithms do not dictate that they need to be implemented in the mobile device or in the network. However, the algorithms have some dependencies that make the network-based approach most attractive. First and foremost we have noticed in our simulation prototype that the processing power required to determine the best HO sequence with the optimal algorithm is quite substantial for 4 or more HOs within W. A battery-driven device may therefore experience a significant reduction in battery life-time if these calculations are performed locally. However, the heuristic algorithm requires less processing power and could therefore be implemented in the mobile device. Secondly, the algorithms need to look up the average throughput of networks along the expected trajectory. Doing these regular database look-ups over a wireless link, could incur a significant increase in the signaling overhead. Also, in practical networks it may be necessary to account for the number of collocated users and their instantaneous load on the different APs and BSs, which advocates for a network-based approach. Finally, the algorithms rely on a prediction of the mobile device's future movement trajectory. In cases where this is based on only GPS location estimates or a distributed localization algorithm the device-based solution could be attractive. However, if a centralized network-based approach where fusion of measurements from e.g. GPS and cellular and WiFi networks is considered, as in [9], then the network-based approach would benefit from a lower communication latency.

IV. EVALUATION SCENARIO

This section describes the scenario that is considered for evaluation of the proposed look-ahead HO prediction algorithm. We have used matlab-based simulations for evaluation. Two different scenarios have been considered as shown in Table I. The *ideal* scenario assumes perfect location estimation and a constant speed linear mobility model. This scenario is used to see how well the algorithm itself performs. Additionally, we consider a *realistic* scenario with localization errors and a random mobility model.

Parameter name	Default Values	
Scenario	*Ideal*	*Realistic*
Simulation time	2000 s	
Independent simulation runs (seeds)	32	
Scenario size	1 km x 1 km	
No. of APs	250	
Hand-over cost (C)	2 s	
Max. number of HOs for opt. alg. ($K_{\max}$)	3	
Prediction window size (W)	20 s	
Historical window size (W_{h})	10 s	
Hysteresis threshold (β_{hyst})	1 Mbit/s	
Throughput variation std. dev. (σ_{TP})	2 Mbit/s	
Localization error std. dev. (σ_{pos})	0 m	2 m
Max. angular acceleration $\alpha_{\max}$	0 rad/s	0.1725 rad/s
Max. acceleration ($a_{\max}$)	0 m/s^2	0.9 m/s^2
Min. speed ($v_{\min}$)	2 m/s	0.5 m/s
Max. speed ($v_{\max}$)	2 m/s	3.5 m/s
Cellular throughput function (Ω_1)	3.6 Mbit/s	
Heuristic handover threshold (ρ)	$2 \cdot C \cdot \Omega_1 = 14.4$ Mbit	

TABLE I
EVALUATION PARAMETERS.

A. Throughput functions

Based on the known locations of the BS and APs, the mobile device can create discrete throughput function for each available network.

The throughput functions for the WiFi networks are constructed using the throughput model described in [10]. Using this model, we have calculated the maximum throughput for all IEEE 802.11a modulation schemes for different distances, and use this curve, as depicted in Fig. 2 to characterize the achievable WiFi throughput. Table II shows the model parameters used in this work.

Parameter name	Values
Bit rates	$6, 9, 12, 18, 24, 36, 48, 56$ Mbit/s
Max. retransmissions ($R_{\max}$)	7
Payload size (B_{MSDU})	1024
Transmit power (P_{tx})	100 mW
Ricean K	15
Path loss exponent	2.9

TABLE II
WIFI THROUGHPUT MODEL PARAMETERS. DETAILS IN [10].

For evaluating the proposed algorithm, we have used polynomial approximations for the $\hat{\Omega}(t)$ function, as this enables
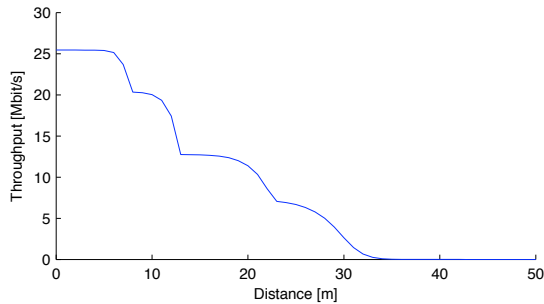
Fig. 2. Distance-throughput relationship for WiFi networks, generated using the model in [10] and the parameters in Table II.

us to differentiate and find roots for the throughput functions. We have used the matlab function `polyfit` for the approximations. Since the goodness of the polynomial approximation depends on the amount of source data, which depends on W in our case, we define the polynomial order as:

$$M_{\text{poly}} = \min(8 + 2 \cdot \left\lfloor \frac{W}{10} \right\rfloor, 18) \tag{19}$$

which has been determined empirically through experiments.

For the cellular network, we assume an urban scenario where the mobile device is able to connect using High-Speed Downlink Packet Access (HSDPA) with a bit rate of $\Omega_1 = 3.6$ Mbit/s everywhere. Of course, other cellular technologies, such as higher-rate 3rd Generation Partnership Project (3GPP) High-Speed Packet Access (HSPA) or 3GPP Long-Term Evolution (LTE) could be used with this HO prediction scheme, as well as faster WiFi technologies such as IEEE 802.11n. The proposed algorithm is not tied to specific technologies, it simply tries to exploit the situations where a local area wireless network offers higher throughput than the cellular network.

B. Mobility model

The mobility model applied in this work is based on the model presented in [11], which is referenced in the mobility model survey in [12]. This model wraps around, meaning that there are no borders, as shown in [12]. This is an advantage because directionality changes imposed by the scenario, that are not taken into account by the movement prediction, can be avoided. The implemented mobility model corresponds to [11], however we use the following slightly different formula for calculating the shortest distance between two entities located at (x_1, y_1) and (x_2, y_2):

$$D_x = \min[|x_1 - x_2|, \min(|x_{\text{m}} - x_1 + x_2|, |x_{\text{m}} + x_1 - x_2|)]$$
$$D_y = \min[|y_1 - y_2|, \min(|y_{\text{m}} - y_1 + y_2|, |y_{\text{m}} + y_1 - y_2|)]$$
$$D = \sqrt{D_x^2 + D_y^2} \tag{20}$$

where x_{m} and y_{m} are the horizontal and vertical lengths of the considered area, respectively. Table I lists the parameters that have been used in this model.

V. RESULTS AND DISCUSSION

For evaluating the proposed schemes we consider an ideal scenario where we investigate the performance of the HO schemes under the assumption of perfect movement prediction and no localization error, while varying different scenario parameters. Secondly, we consider how different error terms affect performance. Finally, we consider a realistic scenario where we have introduced several error terms at the same time.

For comparison, we include two other algorithms that are described in the following.

A. Maximum Throughput Algorithm (C=0)

This algorithm outputs the maximum instantaneous throughput of all available network. This corresponds to always performing a handover to the network with the highest throughput, in the case where the handover delay $C = 0$. In other cases, the result of this algorithm is therefore not practically achievable, but serves as an upper bound on performance. Notice that for $C > 0$, the bound is not tight.

B. Hysteresis based HO Algorithm

This algorithm triggers a HO to another network, if the instantaneous throughput of another network exceeds the instantaneous throughput of the currently connected network by more than a threshold β_{hyst}. In a practical system the instantaneous throughput would be calculated from the instantaneous SNR, or the threshold would be given as an SNR-threshold. If more than one other network exceeds the threshold, the network offering the maximum throughput is chosen. That is, a HO is initiated if the set of candidate networks A_{hyst} is not empty. A_{hyst} consists of the networks a that fulfill:

$$\Omega_a(t) > \Omega_{a_0}(t) + \beta_{\text{hyst}} \tag{21}$$

where a_0 is the currently connected network.

The network to handover to is selected as:

$$a_{\text{max}} = \arg\max_{a \in A_{\text{hyst}}}(\Omega_a(t)). \tag{22}$$

Contrary to the look-ahead prediction algorithms, this algorithm is a greedy algorithm that does not plan ahead, but decides when to handover based on the instantaneous throughput.

C. Ideal scenario

In this scenario we use a linear constant speed mobility model and have set the localization error std. dev. to zero.

The plot in Fig. 3 shows an example of a prediction window and where HOs are triggered. Further, Fig. 4 shows an example of the achieved throughput during a simulation run. Notice how the throughput drops to zero during the duration of each HO and how the throughput bursts when in WiFi coverage.

In the following, aggregated results for multiple independent simulation runs are presented. In the plots we show the mean including the 95% confidence interval. The first result in Fig. 5 shows the impact on throughput of varying the window size. For small and medium length windows the optimal K-HO algorithm is best, but for the long look-ahead windows,
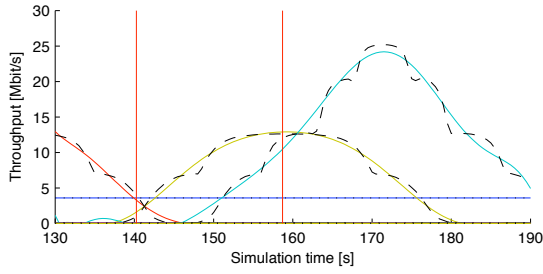
Fig. 3. Example of 60 s prediction window for optimal algorithm. Solid curves are the approximated throughput functions of the different networks, dotted lines are actual throughput curves, and vertical lines show where HOs are performed. This view maps to the $130 - 190$ s span in Fig. 4.
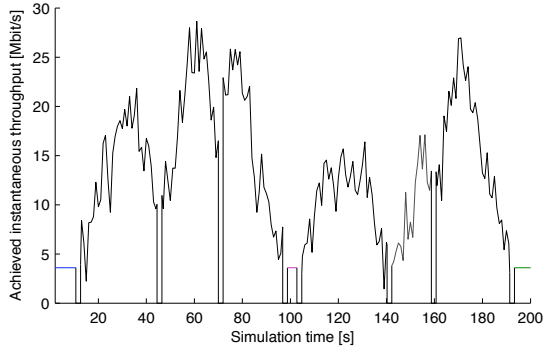

Fig. 4. Example of the achieved throughput for a simulation run for the optimal algorithm.

the heuristic algorithm performs best. The reason for this is indicated in Fig. 6, which shows that as the window W becomes longer, the more HOs are required within W by the optimal algorithm for the best sequence. In these simulations we have limited the maximum allowed no. of HOs to 3 to make the simulations computationally feasible. However, since 3 HOs are highly preferred for window lengths of 120s and 180s we expect that 4 or more HOs would actually yield better results in these cases, hence the limit of maximum 3
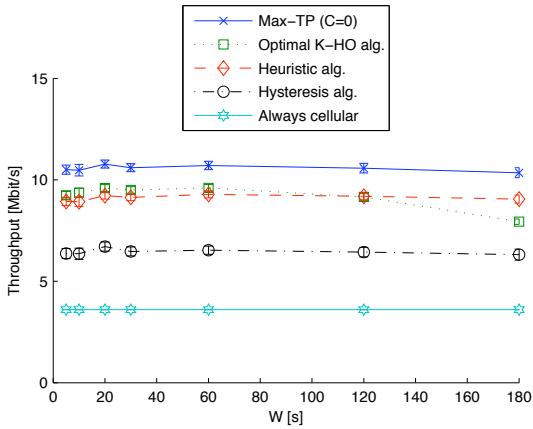

Fig. 5. Performance impact of increasing window size (W). The legend presented in this figure applies to the following figures in this section.

HOs within W causes the drop in Fig. 5. This is supported by Fig. 7, which shows the average number of HOs made by the heuristic algorithm. Here it is clearly shown that the avg. number of required HOs grows linearly with the window length.
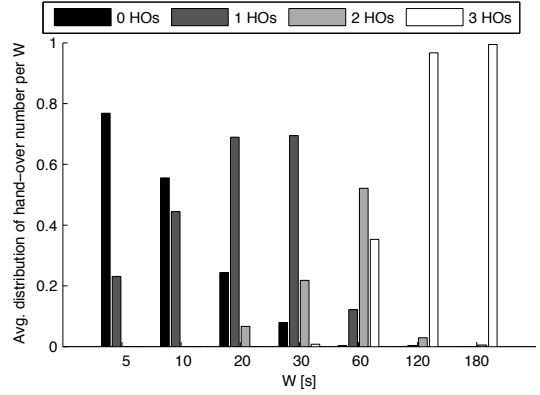

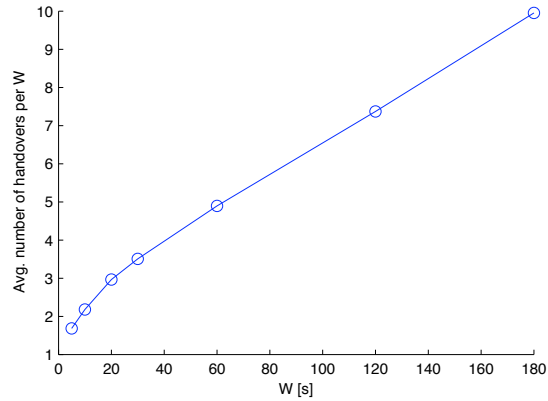Fig. 6. Distribution of no. of HOs for optimal algorithm for different window sizes.

Fig. 7. Avg. no. of HOs for heuristic algorithm for different window sizes.

In Fig. 8 we show how performance is improved when the density of WiFi networks increases. Notice how the optimal algorithm gains more Mbit/s than both the heuristic algorithm and the hysteresis-based algorithm when increasing the number of access points from 50 to 500. The optimal algorithm is clearly better at choosing the networks to handover to, when many options are available, even though it is significantly below the maximum throughput algorithm.

Fig. 9 shows the effect of increasing cost of a HO, expressed as the HO delay C. As expected the increase of C leads to a decrease in throughput. Contrary to the look-ahead algorithms, the greedy hysteresis-based algorithm suffers greatly for even small values ($0.5 - 1$ s) of C. The heuristic algorithm is gradually becoming worse than the optimal algorithm as the HO cost is increasing, due to the simpler algorithm that does not take into account the handover delay C, nor does it consider other networks than those with the highest throughput. The maximum throughput algorithm is of course not affected by
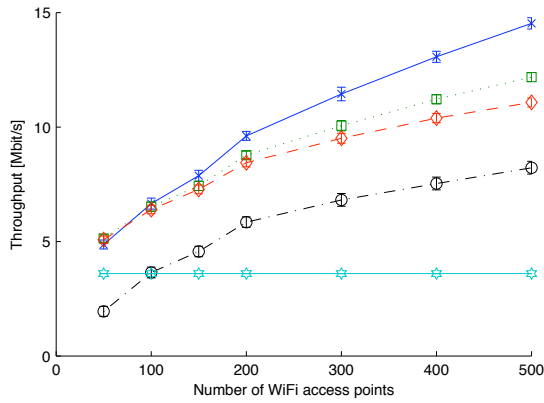
Fig. 8. Performance impact of increasing the number of access points.

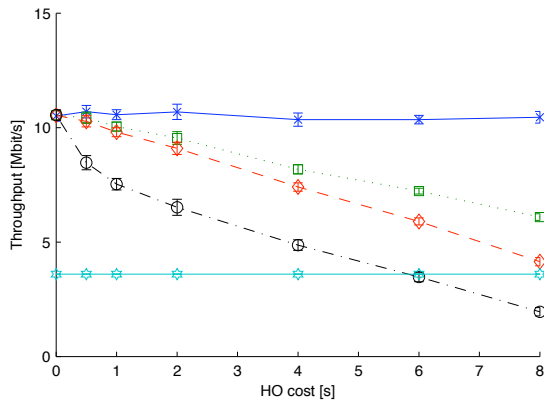the increasing C, since it assumes that C is always zero.

Fig. 9. Performance impact of increasing handover delay C.

D. Varying error terms

The effect of increasing the localization error is shown in Fig. 10. The plot clearly shows, that performance deteriorates with higher localization inaccuracy, since it leads to erroneous movement prediction and in turn bad handovers.
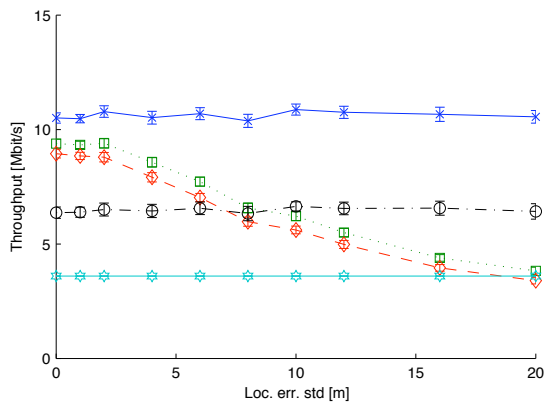

Fig. 10. Performance impact of increasing localization error.

Fig. 11 shows how the hysteresis-based algorithm performs worse the more the instantaneous throughput varies. Since

the look-ahead algorithm uses a priori knowledge of average throughput levels and the actual throughput varies around the mean, it is not affected similarly.
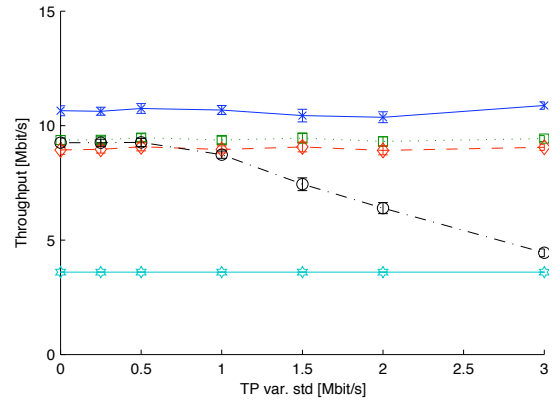

Fig. 11. Performance impact of increasing throughput variation.

Finally, Fig. 12 shows the effect of varying the maximum angular acceleration α_{max}. Larger values of α_{max} reduces the accuracy of the movement prediction, thus the achieved throughput drops for the prediction algorithms.
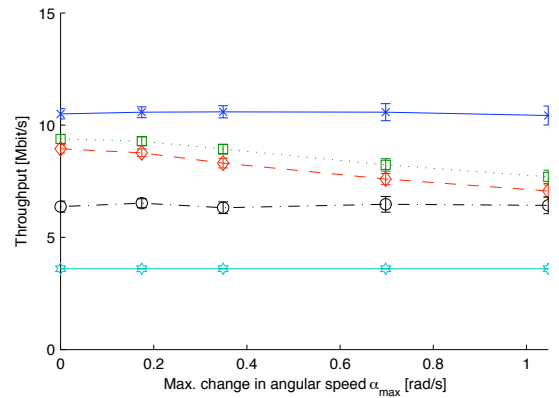

Fig. 12. Performance impact of increasing α_{max}.

E. Realistic scenario

Considering now the realistic scenario, we show in Fig. 13 how the algorithms are affected by different look-ahead window sizes under realistic circumstances. Now, as the window size increases, the localization errors and the random mobility model that are considered in the realistic scenario result in a more rapid decrease in throughput. This means, that while a long look-ahead window where several HOs are planned may look attractive in an ideal system, then in practical systems with imperfect movement prediction shorter prediction windows are necessary. Only in cases where the movement prediction is good, longer look-ahead windows can be useful.

VI. CONCLUSION AND OUTLOOK

In this work we have considered the problem of determining when to handover and which network(s) to handover to, within
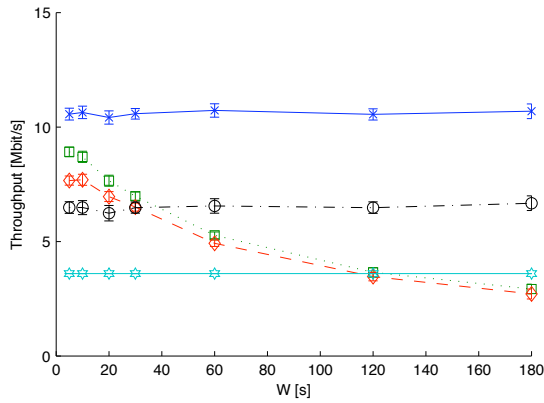
Fig. 13. Performance impact of increasing window size (W), in realistic scenario.

a fixed look-ahead window for a multi-network scenario, in order to maximise the achieved throughput of a mobile multi-radio terminal. Based on an analytical formulation of the handover problem, we have proposed an optimal and a heuristic algorithm for this and compared them to a simple hysteresis-based algorithm and the case where the cellular network is always used. The optimal algorithm finds the optimal handover sequence for up to K handovers within the look-ahead window. In this work we have found that the optimal algorithm is computationally feasible for up to 3-4 handovers, whereas the proposed heuristic algorithm works with any practical number of handovers. The algorithms have been implemented and evaluated using simulations in matlab for a scenario with ubiquitous cellular coverage and randomly scattered high-speed WiFi hotspots.

Our results for the ideal scenario where the movement prediction is assumed to be perfect, have shown that the optimal algorithm achieves the highest throughput for cases where less than 4 handovers are required. For longer look-ahead windows where more handovers are needed, the heuristic algorithm achieves the highest throughput.

The two look-ahead algorithms are equally affected by localization errors, where errors of up to appr. 4 m std. dev. result in only a minor drop in performance.

A general prerequisite for the look-ahead prediction algorithms is an accurate movement prediction. Our results for a realistic scenario shows that inaccurate movement prediction strongly limits the look-ahead window length. However, in cases where the movement of the user is constrained physically by e.g. roads, sidewalks or walls, this can be exploited for improved movement prediction.

An obvious future work item is to reconsider how the look-ahead predictions are used. In this work, the handover decisions made within one prediction window is not re-evaluated as time passes. However, a better handover sequence may be achievable if for example a new prediction window is made after each performed handover. In this way the impact of inaccuracies in movement prediction is kept low.

In this work the handover decision has not taken into account the available resources at APs and BSs. However, in actual networks, the available resources are typically shared between users and it would therefore make sense to take this aspect into account. For the case of a single mobile user, where the load imposed by other users is relatively constant, the proposed selection algorithms can be used as long as the expected throughput along the predicted movement trajectory can be retrieved from the database. In the case where multiple users are performing handovers - potentially to and from the same networks, it is necessary to extend the optimization problem to consider multiple users jointly, since the decision taken by one user will affect the decisions of the remaining users. Given the computational effort required to find the optimal solution for the single user case, it does not seem practically feasible to consider the optimal solution for the multi-user case.

REFERENCES

[1] R. Gass and C. Diot, "An experimental performance comparison of 3g and wi-fi," in *Passive and Active Measurement*, ser. Lecture Notes in Computer Science. Springer, 2010, vol. 6032, pp. 71–80.

[2] P. Deshpande, X. Hou, and S. Das, "Performance Comparison of 3G and Metro-Scale WiFi for Vehicular Network Access," *10th ACM Internet Measurement Conference (IMC 2010), November 1-3*, 2010.

[3] M. Kassar, B. Kervella, and G. Pujolle, "An overview of vertical handover decision strategies in heterogeneous wireless networks," *Computer Communications*, vol. 31, no. 10, pp. 2607–2620, 2008.

[4] W. Chen and Y. Shu, "Active application oriented vertical handoff in next-generation wireless networks," in *2005 IEEE Wireless Communications and Networking Conference*, 2005, pp. 1383–1388.

[5] K. Pahlavan, P. Krishnamurthy, A. Hatami, M. Ylianttila, J. Makela, R. Pichna, and J. Vallstron, "Handoff in hybrid mobile data networks," *IEEE Personal Communications*, vol. 7, no. 2, pp. 34–47, 2000.

[6] C. Sun, E. Stevens-Navarro, and V. Wong, "A Constrained MDP-based Vertical Handoff Decision Algorithm for 4G Wireless Networks," in *Conference on Communications*. Citeseer, 2008, pp. 2169–2174.

[7] J. Zhang, H. Chan, and V. Leung, "A location-based vertical handoff decision algorithm for heterogeneous mobile networks," *Proc. of IEEE GLOBECOM*, 2006.

[8] I. Jolliffe, *Principal component analysis*, ser. Springer series in statistics. Springer, 2002.

[9] C. Mensing, S. Sand, J. Nielsen, B. Denis, M. Maman, J. Rodriguez, S. Hadzic, J. Bastos, Z. He, Y. Ma *et al.*, "Performance Assessment of Cooperative Positioning Techniques," *ICT Future Network and Mobile Summit, Florence, Italy*, 2010.

[10] J. J. Nielsen, T. K. Madsen, and H.-P. Schwefel, "Location-based relay selection and power adaptation enabling simultaneous transmissions," in *Proc. of the GLOBECOM Workshops*, 2010.

[11] Z. Haas, "A new routing protocol for the reconfigurable wireless networks," in *1997 IEEE 6th International Conference on Universal Personal Communications Record, 1997*, vol. 2, 1997.

[12] T. Camp, J. Boleng, and V. Davies, "A survey of mobility models for ad hoc network research," *Wireless communications and mobile computing*, vol. 2, no. 5, pp. 483–502, 2002.