Data Collection in Future Mobile Networks

Marc Fouquet[1], Christian Hoene[2], Morten Schläger[3], and Georg Carle[1]

[1] Technical University of Munich
[2] University of Tübingen
[3] Nokia Siemens Networks, Berlin

Abstract. Future mobile networks will be increasingly heterogeneous. Already today, wireless LAN is used by many mobile network operators as an addition to traditional technologies like GSM and UMTS; WiMax and 3GPP Long Term Evolution (LTE) will be added. Having heterogeneous wireless networks, one challenging research question needs to be answered: Which user should be served by which access network and when to conduct a handover? For such decisions, information on the state of networks and terminals is required. In this publication, we simulate mobile networks in which a central entity called Network Resource Management (N-RM) gives handover recommendations to mobile terminals. Based on these recommendations and local knowledge on link qualities, the terminals choose the cell to switch to. The N-RM should have a global view on the networks to give best recommendations. We designed the Generic Metering Infrastructure (GMI), a publish/subscribe system to collect information about access networks and terminals efficiently.
We investigate the tradeoff between the signalling overhead caused by data collection and the quality of the handover decisions and show, how smart monitoring can reduce the amount of measurement data while ensuring the efficient use of heterogeneous networks. In addition, our simulation results show that combined local and central handover decisions significantly increase the capacity of the networks as compared to only local decisions.

1 Introduction

Future mobile networks will be heterogeneous, i.e., consisting of GSM, UMTS, WLAN, WiMax [1], LTE [2], and other radio access technologies. 3GPP is currently standardizing IP-based mobility solutions that will allow seamless handovers between such technologies [3].

Having a heterogeneous network, we need to decide, which user should be served by which access technology at which time. This problem is complex, as the decision can depend on many factors like signal strength, user location and movement, the load in different cells, capabilities of the mobile terminal, the currently running sessions, etc.

As different user and operator preferences must be considered, a policy-driven decision engine has to make the handover decisions [4]. It is probably impossible to develop one "perfect" algorithm that fits all needs.

Deciding handovers locally in the mobile terminal is sub-optimal, as the global situation in the network, i.e. the load in the different cells, can not be taken into account. It would be advantageous to have a "Wizard of Oz" view on the access networks and

make network-side handover decisions with perfect information, but, of course, it is impossible for physical reasons to collect all data and to make decisions on time. Thus, any central view on the network will be inaccurate and delayed.

Flat hierarchies in future mobile networks make the process of data collection even more difficult. In LTE networks there will no longer be a node like the UMTS RNC, which already has load- and radio data of hundreds of cells, but only evolved nodeBs located much closer to the antennas. With wireless LAN the situation is similar because all measurement data has to be collected at the access points.

Collecting management data is expensive as base stations are spread in the country-side, with costly rented or wireless "backhaul"-links connecting them to the operator's core network. Management and control traffic has to share these links with user data — and as the customer's data is what an operator is paid for, the share of measurement traffic has to be minimized.

To get an efficient view on the state of the network, we have designed the **Generic Metering Infrastructure (GMI)** [5], a publish/subscribe system for collecting and distributing measurement data in an operator's network. It uses various techniques to increase efficiency of data collection, i.e. by building ditribution trees, by caching data and by data compression. The GMI is intended for all kind of management applications, i.e. for fault management, security management, and resource management. In this publication we concentrate on the usage of GMI for making handover decisions.

In this work we do not aim at increasing the performance of the handover system because it could follow diverse goals and any arbitrary policies. Instead, we take a few algorithms as examples and study, how the accuracy of information influences the quality of central decisions. More precisely, we primarily focus on the effect of different strategies for data collection and the resulting quality of mobility decisions.

Our achitecture assumes that the mobile devices have a certain degree of freedom to choose networks based on local preferences. However this can be overridden by network-side decisions, which also relieves the mobile device from constantly having to scan for alternative radio networks. Our network-side decision engine called **Network Resource Management (N-RM)** resides in the core network (Figure 2). It implements algorithms, which give the terminals handover recommendations. The mobile terminals make the final handover decisions based on their local knowledge on the radio conditions and the N-RM recommendations. We base this exemplary system on research results by Fan et al. [6] achieved in the BmBF project "ScaleNet".

We describe the GMI in Section 2 and the setup for our experiments in Section 3. Results are presented in Section 4 and interpreted in terms of data volume in Section 5. We discuss related work in Section 6 and finally conclude the paper in Section 7.

2 The Generic Metering Infrastructure

In this section we describe the Generic Metering Infrastructure (GMI), our publish/-subscribe system for future networks. It allows for efficient data distribution by enabling clients to selectively subscribe for information they need, by distributing the data to interested parties in a multicast-like fashion, by caching and by compressing data.

Like all publish/subscribe systems, the GMI offers an event service (see Figure 1). Clients (called consumers in publish/subscribe terms) can subscribe for "types" of information they are interested in. Whenever new information is created at a producer, it is sent to the event service which then takes care of distributing the new data to all interested consumers.

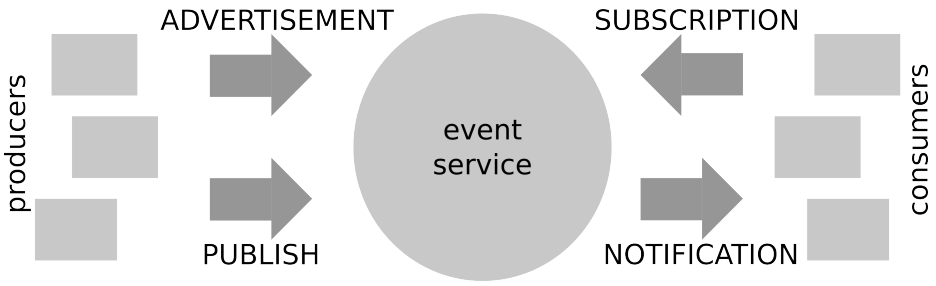

Fig. 1. Basic concept of an Event Service.

2.1 GMI Design

The GMI's event brokers are called **Metering Management and Collection Entities (MMCEs)**. These nodes manage the subscriptions and create distribution trees if multiple receivers are interested in the same data.

Our generic metering infrastructure is a subject-based publish/subscribe system. This means that the "types" of information as described above are organized in a tree, an individual leaf of the tree is identified by a DNS-like address, for example:

```
percentUsed.bandwidth.nodeB0123↩
.mmce.mnc123.mcc123.3gppnetwork.org
```

The general decoupling of senders and receivers which is generally typical for publish/subscribe systems is partially removed by this form of addressing, as the abstract "subject" can include information about the source node of the data (nodeB0123). This is necessary as we sometimes need to get specific information from certain nodes. However, we still have logical addressing as not the information source (called **meter** in our terms) is addressed directly, but the associated MMCE. The interface between MMCEs and meters will be discussed in Section 2.3.

In the GMI, subscriptions not only configure the actual event service but also the meters. This is why the clients send CREATE messages instead of SUBSCRIBE messages, a message from a client will create a new measurement job in a meter if it does not exist yet. Therefore, data that no one is interested in will never be transmitted over the network. Similar measurement jobs are merged and as the MMCEs form a distribution tree for each job, we have a multicast-like "late duplication" property — the same chunk of data is transmitted only once over each link.

The GMI supports three different types of measurement jobs:

Periodic measurements: A client can subscribe to periodic metering tasks. In this case, the subscriber specifies a desired report period and the subject it wants to stay informed about.

Triggers: It is also possible to set triggers for measurements. For such a subscription, the client specifies one or multiple thresholds for a metered value. If the value rises above or falls below the given threshold, the client is informed immediately. Usually our triggers are defined with hysteresis, so there is an upper threshold and a lower threshold. If, for example, the upper threshold has been crossed, the metered value has to fall below the lower threshold to cause another trigger message.

Request/Reply: The last type of reporting is an immediate response to a request of a metering client for a certain value of data. This notification is not an event in the classical sense of a p/s-system. In this case, the metering client simply sends a request for a subject (which is a message similar to a subscription) and receives a reply containing the value (which is handled like a notification). Such requests should be used sparingly in practice. As they do not benefit from the "late duplication" property of the multicast-like data distribution, they cause more load in the system than other notifications.

There are also some additional advanced GMI features that are not mentioned here, the interested reader might consult [5] for an in-depth description of the GMI.

2.2 Granularity Periods and Report Periods

Monitoring in 3GPP networks usually does not directly work on raw data from the network elements, but with derived "key performance indicators" (KPI). A network element collects raw data for an interval called **granularity period (GP)**. After the GP is finished, this data is used to calculate the KPIs [7]. With today's network elements like RNCs, the minimum GP is 5 minutes, while 30 minutes are a far more typical value. Of course monitoring with such a granularity barely helps when building a system to improve handover performance.

For the GMI we expect the meters to work the same way as they do today, but with shorter granularity periods. Each meter has a specific minimum GP. New data for the GMI's publish/subscribe system is only available when a GP has ended. This means that with *periodic measurements*, the minimum report interval is one GP. With *triggers*, the system can only evaluate at the end of each GP, whether or not the trigger has fired. *Request/reply-style queries* are answered after the current GP has ended, so they may get delayed for one GP. Alternative implementations would be to immediately send the last (possibly outdated) value or to mix between the two approaches — to wait for a new value if the last one is too old.

Granularity periods are not a property of the GMI, but of the meters. The GMI publish/subscribe system works event-based and forwards messages whenever they arrive — it is not bound to granularity periods. GMI can also cope with different meters having different GPs, i.e. if old UMTS hardware delivers a value only each 5 minutes while newer hardware might be more responsive.

For periodic measurement jobs, the client specifies the reporting interval in multiples of the granularity period. We call this property of measurement jobs the **report**

period (RP). So a report period of 10 GPs means that each 10th metered value is actually reported to the client.

2.3 GMI in future mobile networks

Figure 2 shows, how the GMI could be deployed in a 3G-beyond network. For UMTS, meters would be placed at the Radio Network Controller (RNC), a central node that controls hundreds of cells. However in other radio access networks, the necessary data has to be collected at much more distributed locations.

The links between the cells (that may be located somewhere in the countryside) and the packet core network are a scarce and expensive resource for mobile network operators. So any management task has to be careful to save bandwidth here. Therefore the MMCEs should generally be placed "above" bandwidth bottlenecks in the packet core network or on its border. The meters reside at the actual locations where data is produced. The interface between MMCEs and meters depends on the type of meter and the link between meter and MMCE. Legacy meters could use well-known protocols like SNMP here, while GMI-enabled meters would use protocols that are optimized to compress the data.

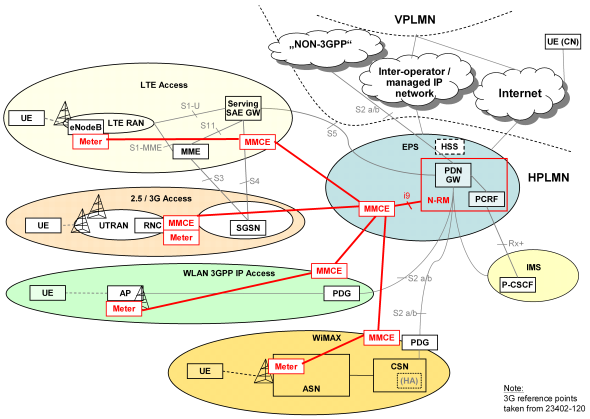

Fig. 2. Mapping of the GMI to the SAE network architecture.

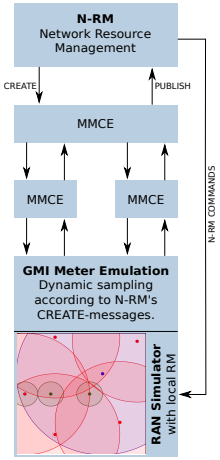

Fig. 3. Experiment setup including N-RM.

Our concept of the GMI is not only suitable for heterogeneous access management, but also for general management tasks or for security applications (i.e. distributed intrusion detection). Therefore we assume that we will have many consumers that use the GMI to collect data. However in our simulations we will only have a single consumer that cares for heterogeneous access management. It is clear that many of the GMI's advantages — like the possibility to build up distribution trees — do not come to play when only a single client is active.

3 Setup for N-RM Experiments

In this section we describe the setup of our experiments. All components of our simulations can be seen in Figure 3 and are described in the following sections. They were run on a single machine, but as separate applications using TCP/IP communications.

3.1 The User- and Radio Network Simulator

The simulator is an application that simulates the mobile networks and their users. Several cells belonging to different **radio access networks (RANs)** are placed on a map. The users move around on this map and start and stop sessions. We use an "accelerated real-time" simulation, events in the simulator happen 15 times faster than they would in reality. The simulator is able to accept GMI subscriptions for a large number of parameters regarding cells and users. The minimum granularity period of all meters is one second, which would be 15 seconds in the real system. In the following we will give the report period in multiples of the granularity period — and not in seconds — to avoid confusion between real time and simulated time.

Map and Radio Access Networks: The map which was used in our simulations is shown in Figure 4. There are three different radio access technologies, which differ in the capacity and range of their cells. The range is given in meters while the capacity is defined as the number of simultaneous sessions that a cells supports. Sessions require a fixed amount of bandwidth and there is only one type of sessions. Compared to the cell capacity, the sessions can be regarded as video calls. We also do not distinguish between upstream and downstream traffic.

RAN A is a network of high capacity and high range, it could be realized using real-world technologies like LTE or WiMAX. *RAN B* is a cellular network. The individual cells have a lower capacity, but *RAN B* has coverage everywhere on the map and in most areas there is even an overlap between the cells. One could imagine *RAN B* as being GSM or UMTS. *RAN C* consists of cells with relatively high bandwidth but a very small range — they can be considered WLAN hotspots. On our map there is a total of nine cells in an area of 1.44 km^2. We assume free space radio propagation, users will lose connectivity as soon as the distance to the base station is larger than the range given in Table 1.

There is a network-internal resource management build into the simulator, which allows handovers between cells of the same RAN, the handover logic for this case is given in Algorithm 1 which is executed for each user periodically.

Access Technology	Number of Cells	Capacity (Simultaneous Sessions)	Range (m)
RAN A	1	31	800
RAN B	5	12	600
RAN C	3	20	120

Table 1. Different radio access networks.

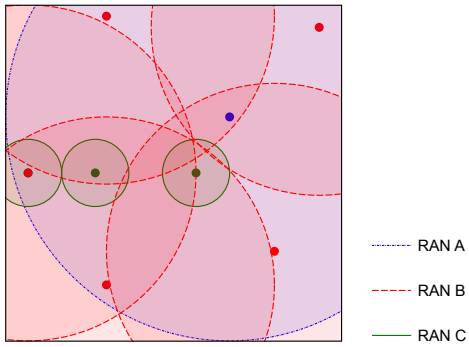

Fig. 4. The map for the simulations, with nine cells of three different radio access technologies.

Users: The movement model of the simulated users is a modified random waypoint pattern. The users select a new waypoint somewhere on the map with a probability of 40%. With 10% probability they will move to one of the three interesting locations that are covered by the *RAN C* hotspots. In 50% of the cases a user will stay at his current position for an exponentially distributed random time. On average, users move at a ”pedestrian” pace of 1 meter per second.

In reality users of mobile devices are inactive most of the time, but for our simulation it makes little sense to model inactive users. To keep our model simple, we have chosen users who start sessions with an exponentially distributed inter-arrival time. Session lengths are exponentially distributed with the same expectation. By this construction, each user has 0.5 active sessions on average.

The users produce actual load on the air interface by opening sessions, while the GMI produces signalling load which only occurs on the backhaul links between the base stations and the core network. Our assumption here is that there is no direct interaction between these two kinds of traffic, even though in reality they have to share the backhaul link. One primary goal of this paper is to estimate the load on the backhaul link which is caused by the measurements. We just monitor the bandwidth but do not limit it.

3.2 The MMCEs

As explained in Chapter 2, the MMCEs are the message brokers of our publish/subscribe system. In a real network they would collect data for the N-RM and other applications from various different locations in the network. In our current setup, all data comes from the simulator. Details about the MMCEs can be found in our previous publication [5].

3.3 Network Resource Management (N-RM)

The N-RM is our global application for handover management. It gets data from the simulator via GMI after subscribing by sending CREATE messages. Its decisions are

Algorithm 1: Mobile terminal- and RAN-internal RM decisions

input : users, cells
constants: $\delta > 0$
foreach *u* ∈ *users* **do**
  `// mobile terminal-side logic. If N-RM recommends cells, the mobile terminal tries to connect there.`
  foreach *c* ∈ *cells_recommended_by_global_rm(u)* **do**
    `// check signal quality`
    if *sq(c, u)* $> th_{acceptable\_sq}$ **then**
      $u.handoverTo(c)$;
      $continue\_with\_next\_user$;
  `// This is the "local RM" logic. It only makes handovers within cells of the same RAN.`
  foreach *c* ∈ *nearby_cells_of_current_ran* **do**
    `// check minimum requirements for load and signal`
    if *sq(c, u)* $> th_{critical\_sq}$ **and** *load(c)* $< th_{critical\_load}$ **then**
      `// find best available cell`
      $score(c) \leftarrow calculate\_score(load(c), sq(c, u))$;
      if $score(c) > score(former\_best\_cell) + \delta$ **then**
        $new\_best\_cell \leftarrow c$;
  if *new_best_cell found* **then**
    $u.handoverTo(new\_best\_cell)$;

Algorithm 2: N-RM decision logic on bad signal quality

precondition: u ∈ users, sq(u) $< th_{critical\_sq}$
input : users, cells
$cells_{recommended} \leftarrow \emptyset$;
foreach *n* ∈ *neighbour_cells(cell(u))* **do**
  if $load(n) < th_{acceptable\_load}$ **then**
    $cells_{recommended} \leftarrow cells_{recommended} + \{n\}$;
sort_by_load($cells_{recommended}$);
$send\_recommendation(u, cells_{recommended})$;

Algorithm 3: N-RM decision logic on cell overload

precondition: c ∈ cells, load(c) $> th_{critical\_load}$
input : users, cells, user_list(c)
$cells_{recommended} \leftarrow \emptyset$;
foreach *n* ∈ *neighbour_cells(c)* **do**
  if $load(n) < th_{acceptable\_load}$ **then**
    $cells_{recommended} \leftarrow cells_{recommended} + \{n\}$;
sort_by_load($cells_{recommended}$);
$receivers = random\_subset(users(c))$;
foreach *r* ∈ *receivers* **do**
  $send\_recommendation(r, cells_{recommended})$;

passed back to the simulator and influence the mobile terminal's cell selection (see Figure 3).

We assume that N-RM does not know the exact location of the user — it only knows the user's location on a "per cell" granularity. Therefore it can not tell exactly which cells are available at the user's current position; it only knows which cells overlap.

The mobile terminal initially does not know about cells of other RANs. In real life it would have to scan different frequencies to find alternative cells — and this constant scanning would waste battery power. So in our case, N-RM will give the mobile terminal hints, which cells to search for. In reality these hints would also contain radio parameters. The mobile terminal will only scan for cells of different RANs after receiving such a hint.

We are testing five different algorithms:

- *Experiments without N-RM*
  In these runs, the users will stay in their *RAN* as long as they have connectivity. When the user leaves the coverage-area and losses connectivity, he will start scanning for other radio access networks and connect to the strongest cell he sees.
- *Experiments with a purely trigger-based N-RM*
  The N-RM subscribes for the load in each cell and the signal quality of each user. The subscriptions are trigger-based, which means that N-RM is notified whenever a threshold-value is crossed.
  In case of a user with bad signal quality, N-RM will request load information from surrounding cells in a request/reply fashion. Based on the results it will recommend cells to the user (see Algorithm 2). This recommendation will have an impact on the next run of the local RM (Algorithm 1) inside the simulator.
  In case of an overloaded cell, N-RM will again be informed by a trigger and then request load information from the neighbour cells. It will choose a subset of the current users in the overloaded cell and send its recommendations to this subset (see Algorithm 3).
  With this basic triggered N-RM, the resource management will only be informed when a critical situation occurs, there is no feedback whether the situation persists despite the countermeasures. As our triggers are defined with hysteresis, N-RM will only take action again when i.e. the load in a cell crosses the upper threshold again after it has crossed the lower threshold.
- *Experiments with an N-RM getting on periodic reports*
  N-RM subscribes to load information of each cell and signal quality information of each user on a periodic basis. This means that there is a constant flow of reports which does not change during the simulation. When receiving information, N-RM will check if a parameter is critical. With this measurement strategy N-RM will never request current information when it needs specific data, but it uses the last reported value.
  The basic decision algorithms are Algorithm 2 and 3. These are run periodically whenever new data has arrived.
- *Combination of triggers and periodic reports*
  This variant works with triggers again, but after a trigger has fired and N-RM has taken action, it enters a success control loop. In this state the trigger is turned off and

replaced by a periodic measurement job which continuously monitors the critical value. Then, with each arrival of a current value, N-RM will check if the system still is in a critical state. If this is the case, it will again request additional data which is needed for the decision and take action accordingly. There is a different (much lower) threshold for cancelling the periodic measurement job and returning to triggered measurements when the situation has been resolved.

– *N-RM with full information*
For comparison all simulations have also been run using a N-RM which subscribes for periodic reports of all values in the simulator with a report period of 1 GP. This can be seen as the theoretical maximum amount of data that N-RM could possibly subscribe to. The decision algorithms are still the same.

Parameter	Threshold	Value
Load	Hysteresis higher threshold ($th_{critical\_load}$)	0.94
	Hysteresis lower threshold ($th_{acceptable\_load}$)	0.89
	Control loop stop threshold	0.70
Signal Quality	Control loop stop threshold	0.15
	Hysteresis higher threshold ($th_{acceptable\_sq}$)	0.07
	Hysteresis lower threshold ($th_{critical\_sq}$)	0.06

Table 2. Threshold values for N-RM.

Triggers and threshold values: Table 2 shows the different threshold values that were used in the simulations. The hysteresis critical threshold (0.94 for the cell load, 0.06 for the signal quality) is crosses (upwards and downwards respectively), the N-RM starts to take. For example, when the cell load of *RAN B* is equal or higher than 0.94, the N-RM will send recommendations to switch to a cell with lower capacity.

To limit the amount of trigger messages, after reaching once the hysteresis critical threshold, the value must reach the hysteresis acceptable threshold, before the high load or low signal quality events can be triggered again. So to say, the hysteresis acceptable threshold (0.89 for the cell load and 0.07 for the signal quality) re-activates the load-trigger after it has fired once. This avoids a message storm if the value oscillates around the actual critical threshold.

The third threshold (0.70 for the cell load) is only active when triggers and periodic reports are combined. Crossing this threshold from above causes the reporting to switch back from periodic to triggered reporting.

3.4 Simulation Flow

On startup, users will connect to the strongest cell in range. *RAN A* and *RAN C* have no overlapping cells, so there are no horizontal (intra-RAN) handovers. For users of the cellular *RAN B*, it is assumed that there is a local resource management instance in the RAN (i.e. an UMTS RNC), which knows the load of all cells and the signal quality for

all users. For each user, this controller calculates a score based on the signal quality of the radio link between the user and each cell and the load of the cell. If the score of a different cell is better by a certain delta than the user's current cell's score, the user will conduct a handover to the new cell (see Algorithm 1). There are no handovers between different RANs as long as global resource management application (N-RM) is not running. When users leave the coverage area of the current RAN they will lose their sessions and scan for a new *RAN* as soon as they notice that the old one is no longer reachable.

When a user decides to create a session, it has to pass an admission control. The admission control checks, if there is enough capacity in the cell to allow the session, otherwise the session is denied.

In case that an N-RM is active, it monitors load and signal quality of the user's devices. Assume that a user approaches the border of his current cell. There is no other cell of the same RAN in range, so the local resource management has no possibility to improve the situation. Depending on the measurement strategy, N-RM might now be informed and decide to take action. N-RM does not know the exact location of the user and signal qualities between the users and other cells, it only knows which cells overlap with the user's current cell. Further, it aims for sending the user into a cell with low load. Therefore, it has to request load information from the overlapping cells if there is no sufficiently current data in the local cache.

With this information, N-RM will sort the cells by load and send the sorted list of cells to the user's device. The UE will then try the alternative cells in the given order and make a handover if the signal quality is sufficient and if admission control for the user's possibly active session succeeds.

The other situation which is handled by N-RM is an overloaded cell. Again N-RM might be informed about this in time depending on the chosen measurement strategy. If N-RM decides to take action, it fetches the list of users of the cell and the load of all other cells that overlap with it. Then it creates a list of recommended cells, ordered by load, and sends this list to a subset of the users in the cell (i.e., to 10% of the users). Again the users will change to a recommended cell if possible.

4 Results of N-RM Experiments

In this section the results of our simulations are described. The values have been normalized to 1 hour of simulated time and 1 km^2 wherever possible. Figure 5 shows the amount of traffic which is produced by GMI reports. On the horizontal axis we have increased the user density and therefore the offered load. One can see that the curve for periodic reports with a report period of 10 GP is linear in the number of users as expected. The curve for a report period of 1 GP is also linear but comparably high.

When the system load is low, triggers produce only very few messages as almost no critical events happen. However, when the load increases, the number of trigger-messages explodes. As one can see, the number of messages decreases again when the load is extremely high, as the parameters always stay above their thresholds.

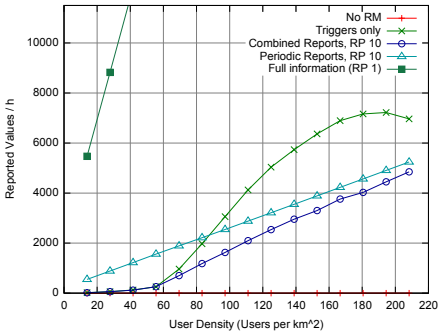

Fig. 5. Number of transmitted values.

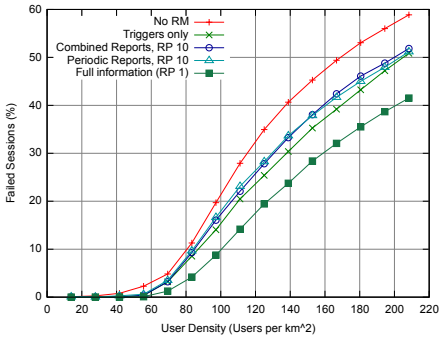

Fig. 6. Failed sessions.

The "combined reports" curve shows the result of an N-RM algorithm which switches to periodic reports after receiving a trigger for a subject. As one can see this curve always produces less messages than the triggered or periodic curves.

The quality of the decisions is evaluated using Figure 6, which shows the percentage of failed sessions. Here one can see that the three "realistic" systems are in the middle between the resource management with perfect information and the complete lack of resource management.

One astonishing fact is that the trigger curve is better than periodic reports and combined reports. This was not expected as the used trigger algorithm is quite primitive, it only acts when it gets an ascending trigger and does not control the success of its actions. However the variance in the measured parameters is high, so the resource management is still triggered very often — which causes N-RM to send more recommendations to the users.

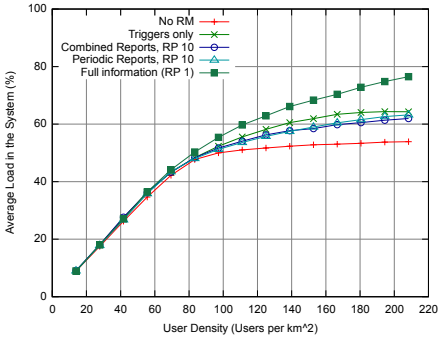

Fig. 7. Average load.

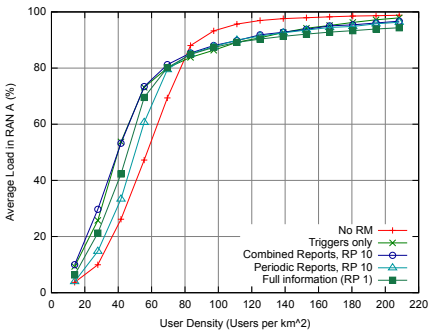

Fig. 8. Load in *RAN A*.

Figure 7 shows the average load in the system. Again the user density can be seen as a measure of the offered load, while the vertical axis is the actual load which could

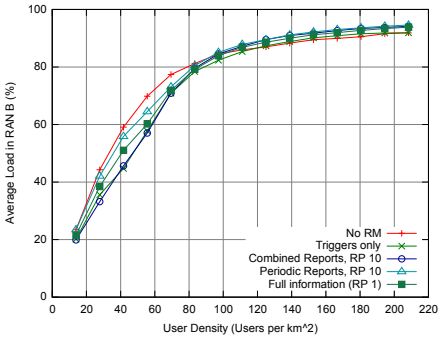

Fig. 9. Load in *RAN B*.

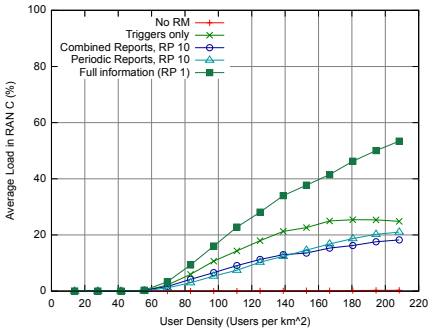

Fig. 10. Load in *RAN C*.

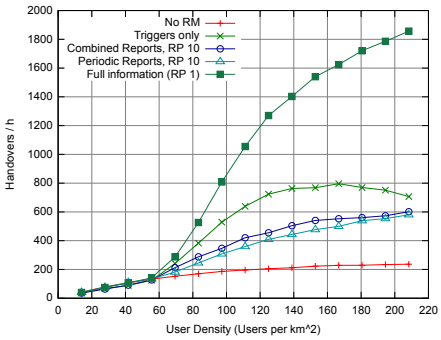

Fig. 11. Number of Handovers.

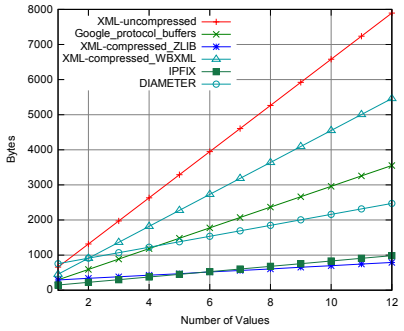

Fig. 12. Data volume when transporting measurement data using different protocols.

be handled by the RANs. All five curves are very similar as long as the number of users is low. At around 80 users per km^2, the curves split. Without a resource management, the system is almost saturated at this point. With 210 users per km^2, the system runs at 54% load. The actual realistic systems are able to run the system at 62% to 64% load, again the triggered resource management achieves the best results. Using the resource management with perfect information, the system runs at 76% load with 210 users per km^2.

RAN A (Figure 8) which consists of one large cell with high capacity is always well-utilized for geographical reasons. The curve for "No RM" crosses the other curves. When the overall load is low, N-RM moves users from *RAN B* into *RAN A* when they experience bad signal quality in the center of the map. With a high user density, the load in *RAN A* becomes very high, so N-RM moves users away from *RAN A* — mostly to *RAN B*. As expected, *RAN B* which has a higher overall capacity shows the opposite effect, but on a smaller scale (Figure 9).

The most interesting curve is the load in *RAN C* (Figure 10) which is equivalent to WLAN. Without resource management, the users will never switch to one of the

Resource Management	Users per km^2	Failed Sessions	Data Volume per Cell
None	56	2.28%	0.00 kBit/s
Combined Reports (RP 10)	56	0.40%	0.01 kBit/s
Maximum Information (RP 1)	56	0.09%	0.64 kBit/s
None	112	28.01%	0.00 kBit/s
Combined Reports (RP 10)	112	22.09%	0.09 kBit/s
Maximum Information (RP 1)	112	14.16%	1.18 kBit/s

Table 3. Data volume vs. success

very small *RAN C* cells but stay in *RAN A* or *RAN B* even when passing by a *RAN C* cell. However, with N-RM, some users can be moved to *RAN C*, therefore the available bandwidth of the whole network can be used better. This can be seen in Figure 10.

Finally, Figure 11 shows the number of handovers that occur in the system. Here one can see that a perfect resource management results in a very large number of handovers. This is mostly due to the event-triggered nature of the N-RM, which takes action more often if it gets more input data. This is another type of overhead which can be caused by N-RM besides the signalling data on the backhaul links. Again there is a trade-off between the benefits like better load distribution and these extra handovers. There might be ways to improve this by using a more sophisticated resource management, however this is out of scope for this publication.

5 Data Volume

So far, we have only been talking about the number of values that were sent through the GMI. In this section we will discuss the volume of management data which has to be sent over the backhaul links for heterogeneous resource management.

In our implementation, the GMI uses an uncompressed XML data format. However in a production environment it would be favourable to use more bandwidth-efficient protocols. Basically any protocol which transmits key-value-pairs is suitable for transporting the GMI messages.

Figure 12 shows the performance of six different candidate protocols when transmitting a representatively chosen GMI dataset. This section estimates the practically required data volume. With each of the presented algorithm there are almost unlimited possibilities to encode the data differently which of course affects the volume. It should also be noted that only application layer data volume is considered, there are no IP-headers and also no headers of the layer 4 protocols that might be used in combination with our six candidates (TCP, UDP, SCTP).

- Uncompressed XML has advantages in terms of transparency as the data is transmitted in plain text. However it wastes bandwidth on the expensive backhaul links.
- WBXML [8] is a standard by the Open Mobile Alliance, which replaces XML tags by shorter binary strings, but leaves the actual content of the tags unchanged. In absence of a DTD, it builds a string table from the tag names and uses references

to this table afterwards, which was the case here. In our example the data has been compressed to 69% of the size of the original XML.

- Google Protocol Buffers [9] is another representation which preserves the hierarchical structure of XML. In the example the data was compressed to 45% of the original size. The direct comparison with WBXML may be a bit unfair, as the Protocol Buffers encoder was able to use meta-information on the structure of the document. With a DTD, we would expect WBXML to perform roughly equivalent to Google Protocol Buffers.
- Diameter [10] is shown in this comparison as it is a common accounting protocol in 3GPP networks. It is the first protocol in the comparison, which has to map the document structure to flat key-value-pairs. However using meta-knowledge about GMI messages, the whole information can be reconstructed at the receiver. With our example data, the data volume used by Diameter was 25% of the original XML.
- IPFIX [11] is based on Cisco's Netflow v9 protocol. It is basically meant to export traffic flow data, but it is flexible enough for our purpose. IPFIX also works on flat attribute-value-pairs, but on the link it separates the attributes from the values. The attributes are sent once in so-called template records in the beginning of the transmission, while the values are sent separately in data records. It is also possible to use options templates for redundancy reduction, so values like long strings that appear in the data quite often only have to be sent once. Therefore IPFIX can save bandwidth compared to Diameter. Our data was compressed to 12% of the original size.
- The last candidate protocol is a simple LZ77-zipped [12] version of the original XML data. The messages have not been compressed one by one, but the state on both sides is held during the transmission of multiple messages. This method of transmission is very efficient. In the long term, after 50 messages, the data volume could be reduced to 7% of the original XML size. However this advantage comes with increased costs in terms of memory and CPU-usage at sender and receiver.

For Table 3 the results from Section 4 have been combined with knowledge about message sizes. Here we assume data transport by IPFIX, which was the second-best solution in the comparison above, as we want to avoid the computational effort of compressing the data using LZ77. We also assume that each value is sent in a separate packet — which is a worst-case scenario — and add TCP and IP headers. As mentioned in Section 3.1, our granularity period is 15 seconds.

As we can see, heterogeneous access management gives a high benefit while using very little bandwidth. With 56 users per km^2 in the system the result with our combined periodic and triggered reporting is almost as good as the result obtained with the maximum available information.

With 112 users per km^2 the radio network is already overloaded as we can see from the success rate of the sessions. However the number of messages we need for N-RM still remains negligible compared to the amount of user data transferred through HSDPA or LTE cells. Here one could even consider to send all available data (GP 1) to the N-RM, which produces 14 times the data volume of the combined reports method, but still stays around 1 kBit/s per cell.

6 Related Work

In recent years, there has been a lot of research on handovers in heterogeneous networks. The approaches in [13], [4] and [14] leave the decision which network to choose to the mobile terminal. This is a reasonable design concept since the information about signal quality of the surrounding base stations is available there, while with network-centric decision engines this information must be transported to the core network. Transporting this data also introduces undesired delay that leads to potentially imprecise values.

On the other hand a management facility inside the network is able to take global information, i.e. on the load situation into account and therefore is able to make better decisions. Additionally, it can help the mobile terminal to find adjacent networks without forcing it to scan for available access points which would deplete its battery power.

The authors of [15] use a network-centric approach which basically integrates WLAN into an UMTS UTRAN network. Our approach attempts to be more general by abstracting the handover-logic from details of the RANs. The authors of [16] adjust load triggers on the network-side to optimize handover performance. This work is about the actual handover decision process, i.e. in a combined GERAN/UTRAN network, while we primarily focus on data collection and intentionally keep the decision process as simple as possible. In our approach ping-pong effects are mitigated by the score calculation of the local resource management (Algorithm 1).

The authors of [6] use a network-assisted policy-based approach. They're using two decision engines, one of which is located in the core network while the other resides on the mobile terminal. This scenario is also the base of our work, the GMI could be used here to provide the network- and RAN-related information which is required by the decision engine on the network side.

A related standard is IEEE 802.21 [17], which specifies an information service, an event service and a command service to support heterogeneous access decisions and therefore consists of several building blocks that were similarly realized for our simulations. However IEEE 802.21 leaves the actual question of data transport open.

7 Conclusions

Today, users of mobile networks increasingly demand data services and voice-calls for flat prices. This makes business difficult for network operators, there is hard competition on the market, and revenues are shrinking. Operators have to cut costs — one way to do so is increasing network efficiency. This requires heterogeneous networks, as different access technologies have different strengths and weaknesses. It also leads to a need for new methods of data collection, as smart management is needed to take advantage of the different networks.

We have shown that the flexibility provided by our GMI has advantages when making heterogeneous handover decisions. It is possible to take good decisions with fewer data by switching between periodic reporting and setting triggers. Our simulation results show that customer experience can be enhanced significantly, while the cost in terms of produced overhead are comparably small.

Our concept of the GMI is not only suitable for heterogeneous access management, but also for general management or security tasks (e.g. distributed intrusion detection). If the GMI was be used for those purposes as well, the compression gains could be even larger because the multicast distribution and caching of measurement parameters.

8 Acknowledgements

The authors would like to thank Andreas Monger and Mark Schmidt for their work on the GMI concept and implementation, Vladimir Maldybaev for his help regarding the evaluation and Frank-Uwe Andersen for the basis of Figure 2.

References

1. IEEE802.16, "Air Interface for Fixed Broadband Wireless Access Systems," IEEE Standard for Local and Metropolitan Area Networks, Oct. 2004.
2. 3GPP, "TS 23.401 v8.4.1: General Packet Radio Service (GPRS) enhancements for Evolved Universal Terrestrial Radio Access Network (E-UTRAN) access," 2008.
3. ——, "TS 23.402 v8.4.1: Architecture enhancements for non-3GPP accesses," 2008.
4. H. J. Wang, R. H. Katz, and J. Giese, "Policy-Enabled Handoffs Across Heterogeneous Wireless Networks," in *WMCSA '99: Proceedings of the Second IEEE Workshop on Mobile Computer Systems and Applications*. Washington, DC, USA: IEEE Computer Society, 1999, p. 51.
5. A. Monger, M. Fouquet, C. Hoene, G. Carle, and M. Schläger, "A metering infrastructure for heterogeneous mobile networks," in *First International Conference on COMmunication Systems and NETworkS (COMSNETS)*, Bangalore, India, Jan. 2009.
6. C. Fan, M. Schläger, A. Udugama, V. Pangboonyanon, A. C. Toker, and G. Coskun, "Managing Heterogeneous Access Networks. Coordinated policy based decision engines for mobility management," in *LCN '07: Proceedings of the 32nd IEEE Conference on Local Computer Networks*. Washington, DC, USA: IEEE Computer Society, 2007, pp. 651–660.
7. 3GPP, "TS 32.401 V8.0.0: Telecommunication management; Performance Management (PM); Concept and requirements (Release 8)," 2008.
8. B. Martin and B. Jano, "WAP Binary XML Content Format," http://www.w3.org/TR/wbxml/, 1999.
9. Google Inc., "Protocol Buffers," http://code.google.com/apis/protocolbuffers/, 2008.
10. P. Calhoun, J. Loughney, E. Guttman, G. Zorn, and J. Arkko, "Diameter Base Protocol," RFC 3588 (Proposed Standard), Sept. 2003. [Online]. Available: http://www.ietf.org/rfc/rfc3588.txt
11. B. Claise, "Specification of the IP Flow Information Export (IPFIX) Protocol for the Exchange of IP Traffic Flow Information," RFC 5101 (Proposed Standard), Jan. 2008. [Online]. Available: http://www.ietf.org/rfc/rfc5101.txt
12. J. Ziv and A. Lempel, "A universal algorithm for sequential data compression," *IEEE Transactions on Information Theory*, vol. 23, pp. 337–343, 1977.
13. V. Gazis, N. Alonistioti, and L. Merakos, "Toward a generic "always best connected" capability in integrated WLAN/UMTS cellular mobile networks (and beyond)," *Wireless Communications, IEEE*, vol. 12, pp. 20–29, 2005.
14. E. Stevens-Navarro and V. Wong, "Comparison between vertical handoff decision algorithms for heterogeneous wireless networks," *Vehicular Technology Conference, 2006. VTC 2006-Spring. IEEE 63rd*, vol. 2, pp. 947–951, May 2006.

15. R. Pries, A. Mäder, and D. Staehle, “A Network Architecture for a Policy-Based Handover Across Heterogeneous Networks,” in *OPNETWORK 2006*, Washington D.C., USA, Aug 2006.
16. A. Tölli and P. Hakalin, “Adaptive load balancing between multiple cell layers,” *Vehicular Technology Conference, 2002. Proceedings. VTC 2002-Fall. 2002 IEEE 56th*, vol. 3, pp. 1691–1695 vol.3, 2002.
17. IEEE802.21, “Media Independent Handover Services,” http://www.ieee802.org/21/, 2007.