# Network and Location-aware Service Selection

Reliability Assessment of Location Estimation by means of Simulation

Networks and Distributed System

Group 14gr1002 - $10^{\text{th}}$ semester

Aalborg University
Department of Electronic Systems
Fredrik Bajers Vej 7B
DK-9220 Aalborg

**Department of Electronic Systems**
Fredrik Bajers Vej 7
DK-9220 Aalborg Ø
http://es.aau.dk

**AALBORG UNIVERSITY**
STUDENT REPORT

**Title:**
Network and Location-aware Service Selection
- *Reliability Assessment of Location Estimation by means of Simulation*

**Theme:**
Master's thesis

**Project Period:**
10th semester
Networks and Distributed System
February – June, 2014

**Project Group:**
14gr1002

**Participants:**
George Kadas

**Supervisors:**
Rasmus Løvenstein Olsen
Lars Møller Mikkelsen

**Copies:** 3

**Page Numbers:** 73

**Date of Completion:** June 4, 2014

**Abstract:**

This work contains a study to assess the impact of network-related parameters to the reliability of a user's localisation when the user's location can potentially change. User parameters such as speed and network parameters such as network delay, noise, localisation interface, etc. are tested in regard to the impact they have on the performance of service selection based on location estimation.
A generic context simulator, Siafu, was used to simulate the context generated by users based on a predefined behavioural pattern. Thereafter the functionalities of the aforementioned simulator were extrapolated to provide the basis as to enable operations on the provided information.
From the extracted results, for the first scenario a 2.6 times increase to the maximum mismatch probability is observed when the maximum delay is increased 9-fold. For the second examined scenario, it was seen that both low accuracy and high lock time for a localisation interface are triggers for high mismatch probability for a moving user. The GPS and cellular interfaces exhibited the highest mismatch probabilities (over 70 % in the ideal case). Since this number is thought to be high for a best case, the role of service density remains to be investigated.

# Preface

This work acts as the Master's thesis conducted on the 4$^{\text{th}}$ semester of Networks and Distributed Systems, at Aalborg University. The theme of this thesis is Context-Aware Networks which is applied in the analysis of the impact diverse context has on the reliability of the location estimation of the user and how it affects the performance of service selection tasks. This is done by utilising the generic context simulator Siafu and extrapolating its functions to emulate an application layer that interacts with the user and his context. All the cases that are examined, are developed through the simulation tool and evaluated over their impact on the performance of service selection.

Aalborg University, June 4, 2014

George Kadas
<gkadas12@student.aau.dk>

# Acknowledgements

As this is my final work, my master thesis, I would like to thank all the people that were there for me, helping me and supporting me in every imaginable way throughout the 2-year master's programme and this semester.
First and foremost I would like to thank my family. It was they who gave me the opportunity to study in Aalborg University and improve my skills in hopes for a brighter future. To them I owe all.
Secondly I would like to extend my thanks to my very good friends Kostantinos Paparas and Leonidas Simopoulos. It was they who picked me up when I was down, who showed me a different path when I thought there was none and they who helped me through this. I want to thank them for all our discussions, their useful feedback and all the times I heard them say or write "Yes, but have you tried...". This thesis wouldn't be today if it wasn't for them.
Last but not least, I would like to thank my supervisors, Rasmus Løvenstein Olsen and Lars Møller Mikkelsen, for taking up the difficult task of my supervision while I handled the rest. Their feedback on every aspect of this thesis and their diverse and unique insights are key to the completion of this thesis. The open mind they kept towards new ideas during our discussions helped me improve my skills and my methodology. They did teach me things that I will hold and value in the years to come.

# Contents

# 1 Introduction

*This introductory chapter serves for generic concepts about the project to be presented. A schema of the target environment of this project is drawn (Context-Aware Networks). The generic scope of this project is mentioned and some objectives are outlined.*
*Finally, a problem formulation is created based on the goals set in order to act as a guide during the progression of the project. At the end of the problem formulation, a question is asked and will act as a way to assess the project and the extent in which the set objectives were achieved.*

## 1.1 Motivation

Context-Aware Networks do not comprise a new field of research for the scientific community, however they make for a very interesting hybrid as they can combine "dumb" and "intelligent" network architectures in order to provide a system of "intelligent" network components that can be subjects to control and a global logic in general. Context-Awareness blurs the lines between those disparate architectures and assures that the operation of an application is expressed not only through its individual user but also from the collectivity of the network (Wikipedia, 2014a). The Semantic Web, Smart Grid as well as pervasive networks are all examples of context-aware networks. Below a depiction of five categories where the concept of Context-Awareness can be applied provide an easier understanding of the field at hand and also assists for the clarification of the individual sections.

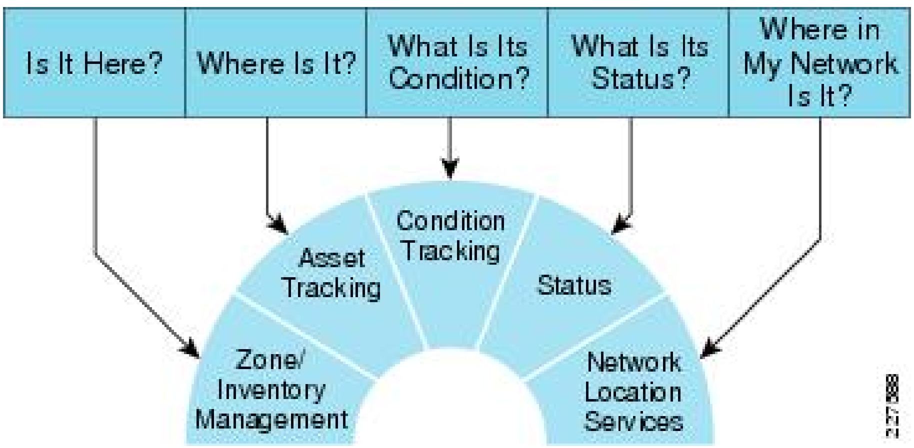


**Figure 1.1:** Context-awareness under different context

While context-awareness on a network basis sounds like an interesting task, there are certain challenges where requirements from other services restrict the network. (Wikipedia, 2014a).
Service selection is the process of selecting an available service based on certain information of the user or about the environment that the user is in. Nowadays, the number of mobile devices has increased many times over compared to a decade ago. This inherently means that the devices that can be part of a context-aware network have also increased and the concept of service selection is more meaningful. The latter is due to the rapid development of applications as well as hardware that are able to take advantage of this context. It is a choice of this thesis to examine the concept of service selection under the scope of Intelligent Transportation Systems (ITS). Intelligent Transportation Systems have been on the rise of the scientific interest since it is a major component of the Smart City concept which is a forthcoming development for the cities of the future.
Intelligent Transportation Systems comprise advanced applications and infrastructure that aim to provide innovative services relating to different modes of transport and traffic management always being oriented around the end-user as quoted in (Wikipedia, 2014c).
ITS has major correlation with context-aware networking in large scales. In a city environment the location context of the user affects in a great way the selection of services for the user. However the location context is not restricted to a GPS location; it can also be context that can be related to a specific location (e.g. Starbucks Wi-Fi signal coverage indicates a specific geographical region).

**Figure 1.2:** Feasibility of Context-Awareness in Outdoor places (Cisco Systems Inc., 2014)

Intelligent Transportation Systems in combination with Context-aware networking has some quite strict time constraints and someone should consider an equilibrium for those two disparate fields. The time constraints ensure both the quality of experience of the user but also the usefulness of the context-aware information. Especially the context information in most cases play an important role in the way the service selection will be performed.
However, the development and combinations of those two network aspects does not only bring new opportunities but also a few problems towards its implementation and service selection. The problem is intensified even more difficult with the recent multitude of information about the user and the network; the set of useful information is getting larger

and larger.

The motivation behind this project is the opportunity to identify how the performance of a network throughout the day affects the correctness of service selection decisions. Additionally, how is the location information of the user able to affect the performance of this task.

## 1.2 Statement of Problem

Below a few key points of the project are presented and summarize some of the main concepts of this project. Through these points what this thesis is trying to achieve is explained.

1. Investigate reliability aspects of service selection in a network environment in conjunction with context-awareness.
2. By making an analysis for specific network parameters, the aim is to understand how reliable a service selection based on location criteria can be.
3. By comparing the results of the different set-ups conclusions can be drawn on how each parameter affects the reliability of a position estimate.

Point 1 comprises the investigation phase, mostly through relevant literature, where aspects such as network load, selection decisions, network interface, etc. are examined in order to make a better delimitation later on. Context-aware networking concepts will be investigated to assure as good as possible coverage of the scientific field. This extensive study will result in point 2.

Point 2 summarises the implementation part of this thesis in relation to understanding the impact specific parameters have on the network architecture and how these can be tuned to provide an acceptable service in terms of correctness always in relation to the context of the network.

In point 3, a comparison of the extracted results from all the different scenarios will be done. As stated in the motivation section, through this procedure, we want to evaluate the performance of the network with a specific configuration and extrapolate it that to draw a general conclusion on what are the most important parameters that affect the service selection task assuming a varying environment.

This leads to the initial problem, which is stated in the form of a two questions:

**What is the impact of varying network performance ,in the course of a day, on the reliability of the user's position estimate and how does this in turns affects a location-based service selection?**
**Additionally, how this network performance, if perceived as context, affects the reliability of the user's location estimate and how is that reflected in the performance of service selection tasks (correct selection of service)**

## 1.3 Conclusion

This introductory chapter is the starting point of this thesis and the place where the motivation behind it, is described. An abstract scope of the project is also set only to be further refined in later chapters; alongside that, the objectives set by the group are presented and some of the possibilities over them are discussed. The objectives that are initially set are abstract enough but at the same time provide a set of milestones should be achieved in this master's thesis. Lastly, the end goal of this master's thesis that is presented here in the form of a question, will be answered in the best way possible by the time it reaches its end.

# 2 Pre-analysis

*In order to investigate how to answer the initial problem stated in section 1.2, some background information is needed. By analysing different topics related to the greater field of Context-Aware Networks and Service Selection, the main trends and leading approaches are identified and work as a guide to achieve the objectives of this master's thesis. In short, by performing the pre-analysis, knowledge on how to answer the initial problem is acquired.*

## 2.1 Service Selection

In this section a higher level analysis for the service selection task we done so the user can initially understand what is a service selection task without going into the specific goals and view of this thesis. Service selection can be used in many fields related to the Web but also to the Networking field of research.
We use service selection primarily as a means to provide better Quality of Service (QoS) and Quality of Experience (QoE) for the end user. This has become a necessity since nowadays devices and service are entering and leaving the network with increased frequency (Cuddy et al., 2005) The means through which we can achieve this always affect the performance of the network itself. In order to perform a service selection we have to have certain criteria on which we can base it. Usually criteria are based on information about the user. The information about the user or a network entity is usually termed as the context of that entity. A formal definition would be:

*Context can be defined as any kind of user related information which has an impact on the relevance of the discovered services.(Tanasescu et al., 2009).*

Generally, network context cannot be described easily as it involves many aspects that define it. The context of the network is affected and shaped by *1*) the user environment *2*) the network conditions *3*) and the content information all the above generate. The image below tries to illustrate the three parameters that affect the context of the network. The environment of the user, the conditions of the network and the generated content are intertwined and affect each other.

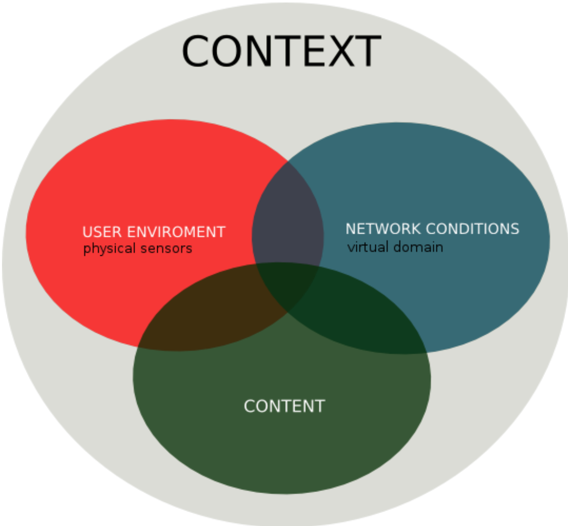


**Figure 2.1:** Relational Model between Network, User and their Content

The image above wants to visually explain the fact that a context-aware service selection takes into account both the context of the user (content mostly generated by physical sensors) as well as the existing network conditions (which can be accessed through the virtual domain) as they comprise the context of the network which a service selection decision will be based on. Moreover, each parameter affects each other as it is varying over time and inherently changing the context of the network. The last parameter, content, is there to declare that the information generated by the user and the network, essentially their content can change according to a predominant state. The size, type or their priority can change according to the dominating state at a given point in time.

But then again, one can argue on the definition of a service as well and what can be thought of as its context. A service can be a printer, with a physical location for the users to access it, a service can be a web page that is hosted in multiple servers in different locations, a service can be the payment to a specific third party for using a licensed software or premises. The definition of service is quite broad so the way we use the context of the user or any network entities to support a service selection depends mostly on the specific case that is examined.
However, the context is quite versatile on what it can include and it is mostly up to its respective entity to define its contents. It can also be generated by different sources; context can be generated by physical sensors but also include sensory data from any virtual domain depending on the generating entity. Nevertheless, the context of an entity might not be static. The location of the user can change over time and the load of a network can change depending on the time of day. These changes can affect the performance of the service selection and this active context can and should be taken into account for a service selection decision. A key component of the active user context is the user's location. By selecting services based on the user location the network load is distributed in a more logical way according to the position of the user. However this approach can be very easily affected by the density of users in a specific area or in a specific time-span. For

example, the response time of a service during daytime might differ a lot from the response time during night due to radical changes in the network traffic, ending up degrading the quality of experience for the user. The location of the user in correlation to the location of a specific service (the definition of service might vary as the way we set the location of it may lead to a correct service selection according to the user's context.
The most common ways to estimate the location of a user currently are cell tower triangulation and the utilisation of GPS. Both approaches have a better performance in outdoor environments. GPS location estimate can be inhibited by line of sight thus making it not fit for indoor localisation cases, whereas cell tower triangulation can be affected by the density of the cell towers. In urban areas a much more accurate estimation of a wireless device is possible due to high density of antennas in comparison to rural areas where they are sparse. Generally, it is common for cellular technologies to be used in fee-based location transactions such as payment for parking service.

Focusing on the location information of the user as context, which is also a partial focus of this thesis, one can easily deduce that a raw location estimate from the user himself is not possible. This is due to the kind of the sensed information provided by the user. Among the information that a user can sense are the direction, the velocity, the acceleration, etc. Moreover, and especially for the cell tower triangulation approach additional context may be sensed and utilised on the user side, such as nearest or neighbouring towers. None of the aforementioned pieces of information constitute on their own a location estimation of the user. In order for a location estimation to be produced, all or part of the sensory data must be processed into a position estimate for the user. The calculations for the position estimate can be done in two different ways:

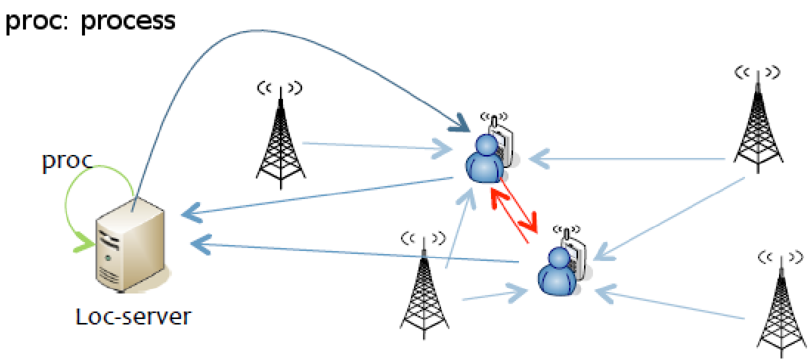


**Figure 2.2:** Location Estimation in a Centralised Fashion (Nielsen, 2013)

In the centralised case presented in the figure above, there is a central server that is tasked with the processing of the sensory data that the users upload and then sends the calculated position estimate (with the received sensory data) back to the user. This approach has the disadvantage of being easily affected by the state of the network and the mismatch probability tends to increase along with the network load.
The alternative and completely opposite to this centralised case is, as expected, the decentralised one.

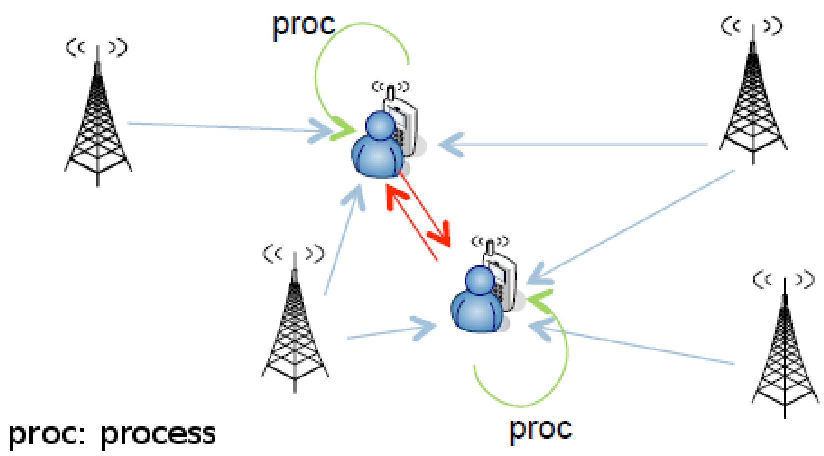


**Figure 2.3:** Location Estimation in a Decentralised Fashion (Nielsen, 2013)

In the decentralised case, the users themselves with their respective devices (from a smartphone to a vehicle) are responsible for the processing of the sensed information and the calculation of the position estimate. This approach gain in terms of speed regarding the calculation of the position estimate. But it might be limited by the capabilities of each device or the need of additional sensory data. Moreover the user might experience lower battery life depending on the frequency of the calculations and the amount of sensory data used to produce them.
Generally, since the position of the user often has major impact on the service selection decision it is logical that a more accurate and fast acquisition of the position estimate can positively affect the performance of service selection tasks.
In this thesis we will examine the former, centralised case, where the existence of a remote server that takes care of providing a location estimate for the user that upload information to it. Since the calculation of the position estimate is remote, delays can be applied to both downstream and upstream for the user. Looking into the centralised approach from a higher point of view, it is simpler since there is no need for advanced hardware specifications for the user as he himself does not perform any kind of calculation but only sends out information. However, this approach is easily affected by network conditions as every user is sending to one location (the server) rather than having decentralised, locally-computed position estimates. It also has the disadvantage that in the unfortunate case of server malfunction the whole process can fail , leading to a potentially less reliable system.

## 2.2 Constraints

Context-Aware Networks (CAN) are a special category of networks and facilitate a specific set of services. However, in order to be able to compare the different scenarios with each other one should be able to understand their environment. As environment we define the number and type of components and entities participating in the network as well as topologies and mechanisms introduced to the network capable of affecting its performance. Most of the constraints were extracted through self-study and discussions with the supervisors, Rasmus L. Olsen and Lars M. Mikkelsen. Not all the constraints are part of the focus of this project and this fact will be discussed further on in the delimitations section 4.

### 2.2.1 Generic Constraints

The provision of correct service selection can be affected by many characteristics that belong to the user, network and context domains. An enumeration of such follows, always in relation to the location context of the user:

1. the relation between estimation time and estimation radius.
2. time constraints on the sensory data.

Regarding the relation between the estimation time and the estimation radius. This lies heavily upon the kind of accuracy that is needed. It surely affects the service selection decision depending on the user context (direction, velocity, etc.). The direction variable is constraint by the roads a user can follow and depending on the environment can be assumed to be "resistant" to change. The velocity variable is directly correlated with the environment that the user is in. In urban environments the speed can be thought of static due to the slow changes in it.
The impact, direction and velocity have on the correctness of the service selection decision is highly correlated with the size of the service area. It is logical that it is easier to calculate a position estimate with a larger marginal error, meaning that in the trade-off between position accuracy and calculation time the latter is favoured. However, even when a smaller calculation time is preferred, when the size of the service area is large enough to allow it, it can still be used to support a correct service selection. The size of the area affects the magnitude of this error.

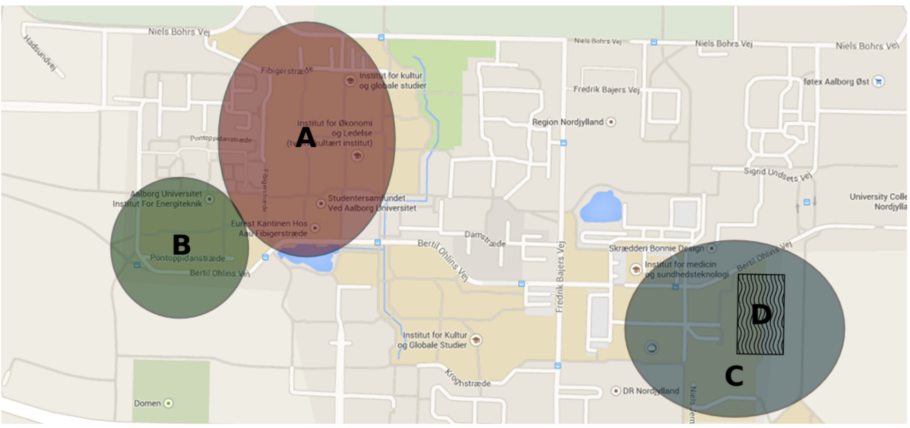


**Figure 2.4:** Possible configuration of multiple service areas - why position reliability is important

The above figure illustrates the points that were previously described. It is easily seen that the location accuracy needed is directly correlated to the specific service area that a user is in. Moreover, we see that there exists the possibility of the layout of the service areas themselves to increase the mismatch probability if the location accuracy remains unchanged.
Even when we are using the location context of the user to support a service selection, this doesn't mean that all errors are eliminated. This is simply because the position estimate might be wrong. Depending on the user's context the time for estimation or the accuracy needed might change. The sensory data for the user have a period for which they are useful as context information. During that period it is imperative that a position estimate is received by the remote server. If that fails to happen due to network delays then the position estimate no longer represents the user's location to a specific accuracy and can lead to location mismatch and potentially to a wrong service selection decision.

### 2.2.2 The Network

In the real world, the way certain networks are laid out on a geographical area is in most cases limited to areas that include building structures (residential, working and entertainment facilities, etc.). An additional parameter is the fact that the movement of the user is constraint by physical routes, vehicle roads, pedestrian paths, etc.
By knowing this information it is easier to predict where the user is heading towards and base certain network decisions based on this spatial information. In that sense, this information about the user becomes his context, thus enabling an adaptive way of handling the communication load.

In this thesis, the impact of contextual information on service selection will be investigated. On the side of the user this mainly involves location information (e.g. pinpoint, destination, direction, velocity, acceleration, etc.). On the side of the network, context is related to its conditions such as available bandwidth, delays, packet loss, etc. It is important to also take into account the network as it heavily impacts the usefulness of the context information. Regarding the context of the user, all the collected information have an "expiry date" after which they are useless in making a decision about the user. If certain time requirements must be kept then the amount of information uploaded onto the network must always be calculated in relation to the network's conditions and the said time constraints.
In real-life practice, the amount and kind of information created by a single user have different "weights" and "expiry dates" and can be used coherently.

While there are many aspects that this master's thesis can focus on, it has become clear early on the project that among the parameters that heavily affect the performance of the network and the quality of experience for the user are *a*) network conditions *b*) user behaviour *c*) and contextual information as more will be explained on a later chapter. However those 3 conditions can be used as initial constraints of the project. The user behaviour as well as the network conditions can be predefined. As for the context information, their collective size can be thought of as a static packet depending on each case.

**Network Conditions**
As network conditions are described all the relevant, to this master's thesis, network performance metrics such as available bandwidth, throughput, end-to-end delay, packet loss, etc. The network conditions are part of the contextual information, or maybe more accurately they are meta-context. This piece of information does not explicitly belong to a specific user rather than to the network that the user is in, but he helps shape this information through his usage of the network. However, based on this information a decision on the actual context of the user can be made.

**User Behaviour**
In relation to the service selection decision, the user behaviour is critical. What, where and when the user does affects his context. If we extrapolate this notion to location information, it is easily seen that the accuracy of the this information is affected by network performance. Inductively this greatly affects the correctness of the service selection decision.

**Contextual Information**
Context is a very broad term. It describes the context of the user as well as the context of the network. By user context, in the scope of this thesis, we define the location information of the user and the accuracy it provides. The network context describes its performance throughout the day. The performance of the network is highly correlated with the behaviour of the users but it can be anticipated based on that correlation. The location information of the user in a network with low performance can easily become outdated. This leads to incorrect service selection. The context of the user can further be modified by taking into account the performance of the network. Through this the aim is to increase the probability of correct service decision even when experiencing low network performance. Obviously when the network performance is high for a user there are less reasons to lead in an incorrect decision.

## 2.3 Conclusion

By performing the pre-analysis, a more precise definition of the focus of the project is achieved. The constraints describe specifications that will be taken into account during the design of the network. Through these an application and an approximate network architecture can be designed. The choices that are presented in the Network section 2.2.2 provide inspiration for the design of the end system.

# 3 Analysis

*In this chapter an analysis of the studied field will be presented for the reader and the need for context-awareness in networking environments will be approached through a realistic example. In that example certain consequences from the lack of context-awareness are outlined. Moreover, the methods and tools used to approach this problem will be presented however in a higher fashion, while maintaining specific details that help with the comprehension of the tool itself. Lastly the performance metrics used to evaluate this scheme are presented and an analysis of their meaning and the calculation method is done.*

A realistic case about where context-awareness can be used will be presented so the reader can associate the abstract concepts described in the pre-analysis to the specific use case that is built for this thesis.
In dense environments like urban ones the need and search for parking places are an every day phenomenon. Nowadays this is followed by many private companies taking up the role of the provider for the service. Assume a city environment; the vehicles that move around looking for parking are numerous, all with different destinations. Each car has to communicate with a specific service provider to complete a potentially necessary transaction, to pay for the parking service. This procedure may be assisted by location estimates, mainly implemented through GPS in vehicles. There are many ways to utilise the GPS to assist this procedure. This thesis will only distinguish between two:

- initiate and complete a service selection task based on the destination set by the GPS and the nearby service providers.
- when there is no destination set, approximate the user's future position through sensor measurements and base the service selection task on this approximation.

Both approaches have a common disadvantage; Both base the service selection to a location estimate. However a question rises: Can the location estimate be trusted to be accurate and correct?
In relation to the previous question one may argue against the severity of a wrong estimate. Inherently this leads to the next issue: What are the consequences of a wrong service selection?
The consequences of a wrong service selection can be both

- financial
- network-related

Regarding the financial repercussions of point 3, there is the case when the user of a service is not corrected billed for it, due to a wrong service selection. In that case we not only

have an amount of money deposited to a wrong account, that being the initial loss, but also the amount of money that have to be paid for the ticket the user probably will receive due to lack of payment for the service at his actual location. An additional cost can also be assumed in order to pay for the correct service provider. A wrong service selection can result in a great loss of money from at least 3 sources, so minimising the probability of a wrong service selection is always in benefit of the user's pocket.
As far as point 3 goes, assuming the case of an incorrect service selection, even if the user complies with the financial repercussions, a more accurate estimate of his position will be required so a new payment to the correct provider can be issued, according to 3. This position estimation with the increased accuracy will result in increased network traffic in order to be calculated. It is logical that while the amount of incorrect service selection increases, the amount of traffic in the network also increases (in order to get the more accurate estimate) thus degrading the performance of the network, especially in dense environments.

The accuracy of the position estimate can be affected by various parameters, especially in an urban environment. As it is common with location information, there is an "expiry date" to them. The problem is intensified in vehicular environments where the position of the user/vehicle changes rapidly. With all the previous in mind and the assumption (or expected situation) that an urban network will have low performance due to high loads, it stands to reason to claim that there is a high chance that the calculated estimate will probably not match the actual position of the user. This leads to an incorrect selection of service which can further degrade the performance of the network and ultimately the user experience as explained in 3
Since there is ubiquitous connectivity in urban environments, it can be easily exploited to increase the probability of correct service selection. We can relate the level of accuracy we need

- to the size of the service area
- to the network performance.

Those two parameters may vary, the first being affected by the scale of the area that is examined, the number of the service areas inside it and their respective sizes. The variation of the second parameter firstly, lies upon specific network characteristics that can affect its performance. the characteristics, such as equipment, interfaces, etc. that affect the performance of a network are out of scope for this project and will be assumed to provide an arbitrarily designed performance over an area of interest. Secondly, the network performance greatly depends on the behaviour of the users and how it evolves over time.

## 3.1 Simulation: Practical Value

From a practical point of view, simulations are more often used to predict a system's behaviour and can be used for conditions that are difficult or even impossible through test-beds due to scalability constraints. When it comes to large networks that can cover

a large, infeasible for real-life implementation, area the most efficient approach of measuring their performance is to create a realistic simulation that approximates the system's expected behaviour. The simulation approach helps us obtain a better and directly comparable view of the system's performance.
Moreover, it is generally less costly than experimenting in an actual test-bed. The only existing costs are the simulation software cost and the work hours that will be spent to create the simulation model. Under real, non-simulated, conditions the downsides of testing include longer run-times, a longer set-up phase and hardware costs. The simulation approach also offers repeatability, through which a large number of scenarios can be tested and compared, examining the exact same system under different input or network conditions and user context. Adding to that, by increasing the complexity of the simulation model a better representation of the system can be achieved, leading to simulation results closer to the results that one would expect from a real-world implementation. Lastly, simulation can offer a shorter run-time compared to a test-bed, allowing fast results when changing parameters between tests. This illuminates another advantage of the simulation approach. That is the ability to investigate if a system is future-proof, meaning for example that the network size can be scaled higher than what is seen in a real world environment. Scaling a network is not easily achievable for test-beds since it will require many hardware components, increasing the cost of the implementation. The three main advantages of simulating are presented below:

1. Approximate depiction of reality
2. Insightful system evaluation
3. Future proofing of the system

### 3.1.1 Siafu : What and Why

Siafu is a versatile, large-scale context simulator written in Java. It includes models for agents, places and the context therein. Through those models the scenario can be influenced and also collect contextual information from any of the agents in the network. (NEC Corporation, 2014). Additionally, Siafu provides visualisation for context information. Siafu can also be integrated with an external application to show the effects of context change on it. The simulation in Siafu can include multiple geographical setups (e.g. Urban, Rural, Indoors, etc.) and also take into account user preferences, sensory information, etc. to use as context.

Siafu uses .java file to implement all the necessary components, from user agents to world and context models. In those files the details for each model are defined as well as the evolution of the user context. They are all brought together and combined in the configuration file which is an .xml file. The config.xml tells the simulator where to look for specific files and sets some parameters of the simulation itself, e.g. starting time.

**World Model**
The world model is responsible for modelling the simulation environment and it defines the places of interest and how global events (e.g., public holidays) are generated. The simulated environment is specified using two image files: a background image for visualization

and an alpha map that defines the areas where agents can walk. Places can be permanent (e.g. homes) or temporary (e.g. meeting places) and have properties, changeable by either an agent or global event model. Places can be instantiated using a bitmap when based on a template. (Martin and Nurmi, 2006)

**Agent Model**
The agent model is the decision logic of individuals, and it is responsible for making decisions on what to do and for updating the state of an agent. The agent model is specified as a state machine where context triggers state changes. Also, stochastic state machines can be used. The simulation designer must define the agent model for each agent. (Martin and Nurmi, 2006)

**Context Model**
The context model specifies the context variables that are part of the simulation (e.g. temperature or hotspot coverage), how their values are distributed over the environment and simulated over time. At any time, the context of each agent is defined by its personal context (e.g. activity), and the environment context provided by the context model. The variables in the context model are specified using an overlay image and calibration metadata that defines a mapping between the colour intensity in the overlay and the output value. (Martin and Nurmi, 2006)

## 3.2 Performance metrics

The simulation that was described in the previous section serves the extraction of results so we can quantify the extent of improvement or deterioration in the performance of the service selection decisions. In order to compare the two scenarios a set of common parameters must be established that will provide the base of the comparison and a second set that will vary and differentiate the scenarios. However measurements can be taken from throughout the network about its context. Now, about the context; it evolves and changes, in this case affected by the time of the day and the user behaviour.

### 3.2.1 Mismatch Probability

The end objective in this thesis is to see what happens to the mismatch probability of the user localisation and how is that affected by the context and the performance of the network. **The mismatch probability on the user's location is defined as the probability that a user is not located in the estimated area but out of it.** The radius of the area that is estimated depends on the network context (network performance) and any time constraints on the contextual data that are used to calculate this estimate. While this is the approach also described by the analytical model later on, it is not this the definition of mismatch that is implemented. The implementation defines the mismatch as selecting different services at two points in time based on the spacial proximity. Those two time points are the beginning of the request and the time the user receives the "estimate". At those two times services are selected based on proximity; if those are different it constitutes a mismatch.

It can be assumed that if the service selection decision is based on this location information and estimate then an increase in the mismatch probability of the user's location will cause the inverse effect on the correctness of the service selection decision. This usually results in additional load for the network as a re-estimation must be made and degrades the experience of the user overall.

Thus, the cumulative metric that can encompass the performance of the network and inherently, to a measure, how correct or wrong will be the service selection decision is the mismatch probability on the location of the user.

Mismatch probability is the method of calculating the probability of an event happening within a time frame which is defined stochastically. As seen in the figure below, the time it takes for a position estimate to be calculated based on event 3 is so large that by the time the decision is received, the position has already changed enough and is not able to be approximated based on event 3. So the task of mismatch probability is to calculate the probability of an event happening which will result in a wrong decision.

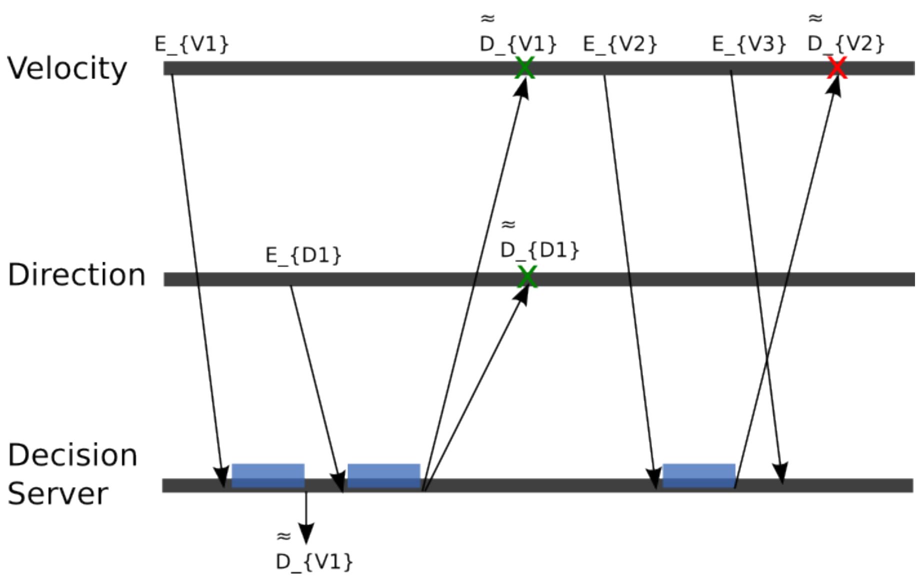


**Figure 3.1:** What has to happen for a mismatch to occur - generic approach

In this thesis the probability of a mismatch during the service selection will coincide with a location mismatch,hence the service selection will be based on the location of the user. Here the mismatch probability will be based on making a wrong service selection. Since that service selection is based on location criteria, mismatching location information of the user in relation to the closest available service will lead to a wrong service selection. That becomes possible by comparing the distance of the user from all the possible candidate parking places around him. The distance of the user from those place is compared on two occasions, when he initiates the service selection task and again when he receives the reply from the remote server. As the user may be in motion throughout this exchange of information, this might lead to a wrong service selection.

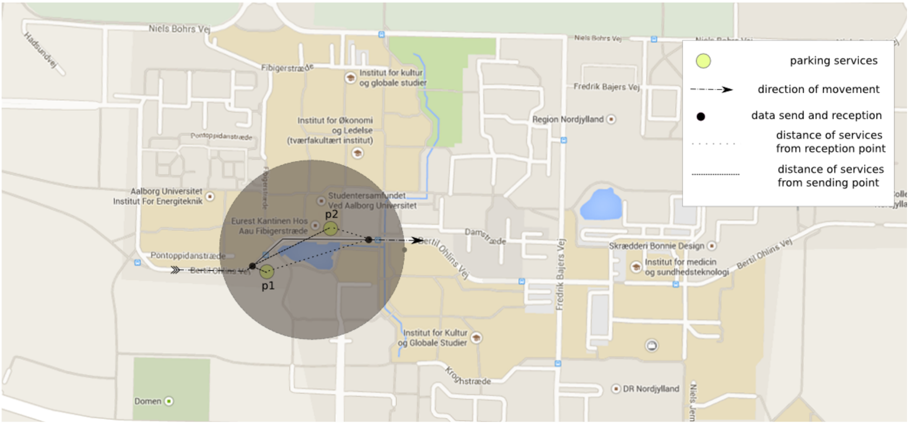


**Figure 3.2:** Definition of mismatch probability in the implemented simulation

Let there be a user moving on the road with a direction from west to east as depicted above. On the path of that user there are two available parking places, p1 and p2. There is also a high network load in that area (dark circle). Let's denote with the two small black circles on the trajectory of the user, the locations where he initiated a service selection and where he received the reply from the server, while the user kept moving. It is easily seen that due to the high delay the user experiences, by the time he receives the reply he has moved quite far away from p1, thus leading to a mismatch of the estimated position, followed by a wrong service selection.
In that sense, the mismatch probability presented in the above figure is one that is mostly affected by distance criteria. However, there are also multiple factors that can affect how or when this distance is measured, for example network load, speed of the user and density of parking places.

## 3.3 Conclusion

This section analysed several topics of interest that affect the design of the custom approach. The simulation approach and the reasons for which it is followed were presented along with a tool for context simulation, Siafu. An outline for the scenario to be implemented was also drawn, stressing the specific characteristics of the mapped area. In the performance metrics section, the parameters that are meaningful for this thesis were analysed.

# 4 Delimitations

*In this chapter, delimitations that were made in the scope of this project will be presented and explained. The delimitations made are related to a) Network Performance (in terms of network load), b) contextual information of the user, c) the connectivity available to the users, d) the introduced delays in regard to the network performance and the process delays from the used architecture, e) and lastly, assumptions concerning the service discovery capabilities of the users. Those delimitations will be used during the implementations phase to simplify certain aspects of the simulated network and its behaviour.*

The project is delimited as to, firstly, provide meaningful results for the specific area that will be investigated but also to fit it into the time frame available for the preparation of this thesis.
The project work will include one scenario through which the correctness of the service selection decisions will be investigated in correlation with the location estimation accuracy for the user and the network performance.

The delimitations done in this project are directly related to objectives of this thesis but also take into account the nature of the designed network. Obviously there are delimitations in the user environment and the network conditions that may exist in such a network. Moreover there are delimitations made that affect the overall performance of the network. An enumeration of the delimitations performed in this project is as:

1. Network Performance
2. Contextual Information of the User
3. Global Connectivity
4. Network Processes and Delays
5. Service Discovery

## 4.1 Network Performance

A major delimitation done in this thesis is one concerning the performance of the network in the mapped area. Since there are not actual measurements through which a realistic network performance can be extracted from, certain assumptions are made about the behaviour of the users and based on that, a performance map is drawn. The performance of the network is assumed to be only a variable of the network load. Subsequently the

network load in specific areas is defined in an absolute manner by the hour of the day. The assumption on the users' behaviour is that during working hours the performance drops in university buildings and around them according to some small radius according to the performance map. By the same token, it is also assumed for the network performance that it peaks in residential areas during the hours before and after work, thus creating an inverse relation between those locations.
This delimitation not only facilitates the creation of a performance map but also in a way, defines the service areas from which the user must select according to his context Steffen R. Christensen (2012).

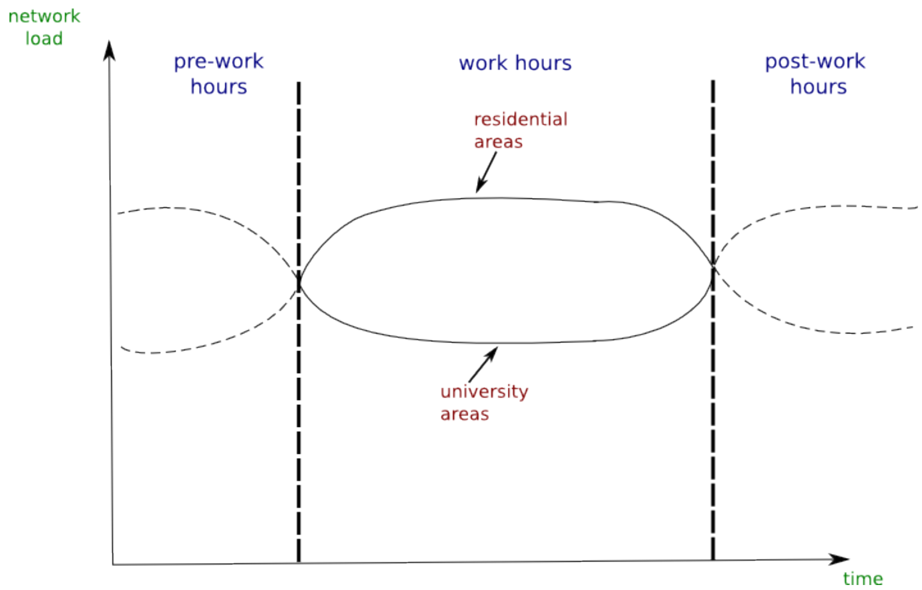


**Figure 4.1:** Assumption on the network performance in correlation with the time of the day

## 4.2 Contextual Information of the User

Another delimitation done in this thesis is the context of the user. While the user's context can include multiple parameters that more fully define it, for the purposes of this thesis we define as context of the user, all information that are relevant to his location and its estimate. That is information that describe his current location and can be used to estimate his next one over a specific time window. Current position, direction and velocity are some of those pieces of information. It is logical to assume that in a realistic case all those parameters, each have a specific "weight" that is taken into account to assert the reliability of the estimate. Additionally, concerning the representation of this information, they are not thought of as a stream of independent packets, each with its own delays but as one packet that can transfer the information from the user to the calculation server reliably.

## 4.3 Global Connectivity

Another assumption that delimits the work done in this thesis is that the users can always be connected to a network. The user can at any point send information to the network and receive from it without any restriction. An assumption that was made an also falls under the connectivity category is that this project does not differentiate between network interfaces. Normally, there would be the case of using both cellular and Wi-Fi for outdoor localisation. In this case however the network interface is out of scope and the same performance is assumed for any used interface. However, even if the user is always connected he still subjects to any introduced delays. In this project the delays that the user experiences are analogous to the network load at the user's position.

## 4.4 Network Processes and Delays

Following the previous, the assumptions about the network processes and the introduced delays states that the position estimation calculation time in the remote server is small enough to not affect the total Round-Trip Time (RTT). That leaves only the delays in both the downstream and upstream to affect the RTT and inherently the usefulness of the position estimate. Additionally both downstream and upstream delays are considered to affect the communication in a similar way.
This can be translated to:

- Upstream delay = Delay on user's position at the time of request
- Downstream delay = Delay at the user's position right after the end of estimation calculation

## 4.5 Service Discovery

This thesis does not go into depth regarding the service discovery process. This assumption has the user to know all the available services around him, defined simply as geographical points, and has no restriction of distance. This means that there are no additional delays due to service discovery process before the actual service selection. When the user has to make a selection he is aware of all the possible candidates and the only additional delay is generated solely from the network load at the user's location.

## 4.6 Conclusion

This section went through all the parameters and network aspects that this thesis delimited from. Specific information on the way these delimitations were implemented were given.

Moreover, it became clear as to what those delimitation accommodate in the designed simulation.

# 5 Design

*This chapter gives an overview of the designed approach for the purposes of this thesis. It goes into detail about the design choices that were made in order to simplify the investigated scenario but at the same time be able to extract diversified information on multiple users. The basic contents of the approach are outlined, such as the environment or the expected behaviour of different users.*

## 5.1 System design

In this section the design behind the actual system implementation will be explained. The purpose of this is to make the network and its key components understandable for the reader.
Usually the size of the network has no meaning unless understood through the context of a specific test-case. As this is the case here also, the network must fit the scope of this project. Service selection in small areas has little meaning in correlation with location context. Even though indoor localisation is on the rise during the recent years, the provision of different services in a relatively small area is still maturing. Moreover, the precision required for such a task might be prohibiting depending on the network and the technologies used.
It is the entirely opposite case for outdoor scenarios where the "playground" of the user might be an entire city or even a country.

In order to investigate the case of reliable communication based on the available context information for the user certain goals and tasks are set:

- create an approximate model of network usage based on user behaviour,
- try to identify the extent of influence of location information on service selection tasks and how the network conditions affect this selection task,
- find a equilibrium point or any preferred configurations based on the network or user conditions.

The network examined in this thesis involves a specific area of Aalborg University in the surrounding area of the Department of Electronic Systems. Additionally, in this thesis the service on which a selection must be performed is defined as parking payment. The system is designed to reflect this behaviour of service and users. For the purposes of this thesis the performance of the network (in terms of network load) can be assumed to be known, in the sense that the behaviour of the users and inherently the performance of the network on which they are connecting, can be anticipated. The performance of the network can be seen to follow the same fashion according to the time of the day it is measured. A specific scenario is built around the said mapped area assuming a known variation of the network performance based on the time of the day and the area of the measurement.

### 5.1.1 The Design

In this subsection a more analytical and detailed view of the designed scenario will be provided. This works as a connecting point between the theory that preceded and the implementation of the simulation that will follow.
In this thesis, it has been chosen to map an area of the Aalborg University main campus. This thesis wants to measure the correctness of parking service selection and will only utilise the user behaviour to explain the varying network performance. The users' behaviour creates a differentiation in the performance of the network throughout the day. The network performance is assumed arbitrarily as there are no real data to relate it to. The main principle is that the performance drops in pre-working hours in residential areas. During the working hours the performance drops in university buildings as more users gather in the same geographical area. In the post-work hours the performance increases in university buildings as the majority of the users leaves the network, followed by a drop in the performance of the residential areas networks as people return home and use that network. The relation is clearly inversely analogous depending on the time of the day a measurement is performed. The university and residential areas are out of scope for the system design and they are not included in the mapped areas. However, the performance maps are based exactly on the previously explained behaviour.

In this thesis the network covers a large geographical area that presents with different loads during the day. This allows to identify the effects of varying network performance to a service selection task.

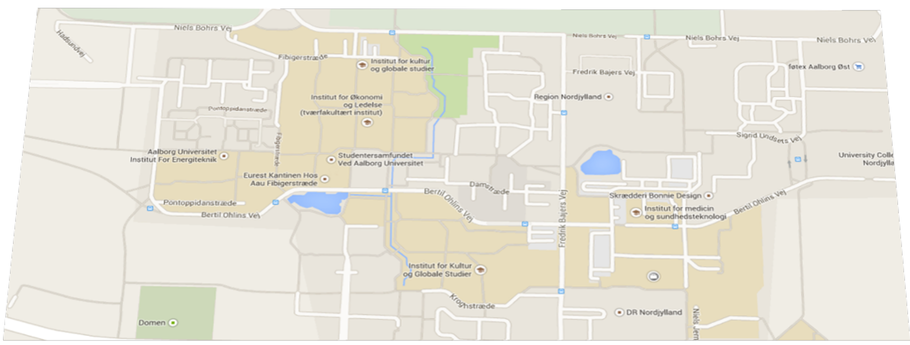


**Figure 5.1:** The University area that was mapped

The image (pixel) coordinates that describe the area that was mapped for this thesis are:

- top right: 1277,144

- bottom right: 1277,692
- top left: 93,144
- bottom left: 93,692

The pixel coordinates were taken at the corner points of the background images as they will also be used to define the "playground" of the simulation. As the background image was captured through Google Maps©the actual map coordinates are also known.
Over the mapped area, there are two kinds of places defined. The first are the road segments in which the users will be moving. The second, small circular areas, define the parking places that the user has to select from based on his location. Due to implementation restrictions the parking areas are not overlapping with each other and this aspect of mismatch will not be investigated. Rather than this, the impact of densely positioned parking places on the performance of the service selection will be investigated. This implementation is in contrast to the one presented in figure 2.4 due to restrictions of the simulation tool. In this implementation the service areas of different sizes are substituted by a dense environment of individual services.

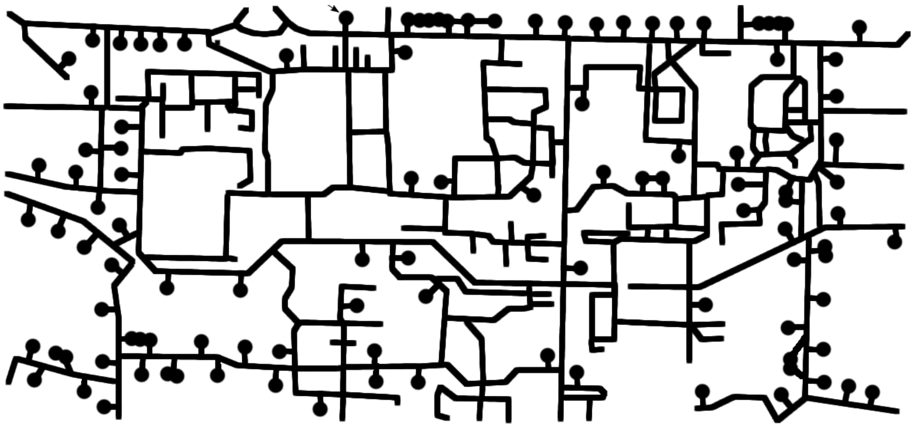

**Figure 5.2:** The Layer that defines the simulation area

The image above illustrates the specific layer used for the simulation. The areas of interest are primarily the small circular areas that define the parking places inside the simulation area. Additionally, there are entrances and exits to define and accommodate the user movement. The goal behind the design of those place was create a long-enough path that the user will follow, so it can be seen over the whole span of the scenario how the differently placed parking spots affect the performance of the selection decision.
In the "playground" created for this scenario there is a set of places defined as to observe the performance of service selection under different network loads. The simulation defines:

1. 2 entrances
2. 2 exits
3. 108 parking spots (dense environment)

4. 75 road points

Apart from the specific places created to accommodate the scenario, users are also introduced to implement a specific behaviour. This behaviour is simple in order to ensure a certain level of randomness to the user's movement. This is important due to the alternative case. This is a predefined route for the user. When the user has a predefined route, the location mismatch probability reduces to a minimum (if not to zero) because the end point is already known and appropriate service can be selected based on that. However when the user has no fixed destination but wanders the map looking for a parking place then the mismatch probability may decrease or increase based on the behaviour of the user.
Those two cases define the two initial scenarios to be investigated.

### 5.1.2 The Scenarios

In this subsection the scenarios that will be investigated will be explained more in depth. There are two scenarios that implement different movement patterns for the user, which is one of the main aspects that can affect his localisation. The performance maps that affect the delay a user experiences in a specific position on the map remain unchanged for both scenarios as they are based on assumed user behaviour and how that evolves over a day span.

**Localisation at Destination**
In this scenario, the user will have a fixed destination and when he reaches that destination, he will then initiate a service selection procedure. Since the user is not moving at the destination point, it can be assumed that his position will not change any more, thus the data that will be used to calculate his position and select the appropriate service will never go out of date. This means that the service selected in this scenario will always be correct and since the user is not moving from that position, the delay of the network is not important. Summarizing this scenario, it specifies:

- multiple randomized destinations for the user.
- fixed user speed
- service selection when at destination

The varying user speed is in relation to the total simulation runs. In each individual run, the agent has a fixed speed. Thus, the only thing that is varying in this scenario is the context of the user when he reaches his destination according to the time of the day.
However, there is an alternative approach to this scenario that will be examined. Instead of having the user not moving during the localisation procedure, what would happen if he was moving? In this approach the user is moving while trying to get a lock on his position, possible correct, through various technologies. The GPS, WLAN positioning and Cellular Network positioning approaches will be examined in both approaches of this scenario. In the alternate case the user is requesting position estimates while moving; due his position changing, sometime drastically, depending on his speed, the specifications of the technology used such as TTFF and accuracy play an important part on the mismatch probability of the user's location and inherently the correct service selection probability.

In the "Localisation at Destination" scenario the context of the user is not translated to network delay as it happens in the "Wandering User" scenario to be described later on. It is rather translated to noise at the destination of the user that causes for a shifted position to be calculated than the true position of the user. While knowing the true position of the user, and being able to shift that position according to the context of the user at that location it becomes possible to compare the service selection that will be made based on both positions, true and noisy.
Even though in the "Delimitation" chapter, through the Global Connectivity clause this thesis delimited from any specific network interfaces, in this case it is really meaningful to examine individual interfaces as each one of them has different accuracies when it comes to location estimation. What will be delimited however is the time it takes for each interface to achieve this accuracy; since the user is assumed to have reached his destination and will no longer move the time it takes for the estimate is thought to be trivial.
Out of the three different interfaces only two are affected by network-related context, those being Wi-Fi and cellular interfaces. The GPS technology can only be assisted by the Wi-Fi and cellular interfaces to increase its response time but network conditions affect only but little the reliability of the estimate itself and only due to receiver-end issues. In summary:

- GPS localisation: 5-10 meters accuracy.
- Wi-Fi localisation: 30-50 meters accuracy.
- Cellular localisation (Triangulation): $\approx$ 100 meters accuracy.

Using the above information on the accuracy of the different interfaces a mapping will provide the matching between the context at the user's location and the shift of the true position in order to adhere to the above boundaries. However, even under the accuracy limitations presented, further specifications were made in order to define additional scenario values and maintain a realistic scope on the simulation. That being said, the following are defined:
The GPS localisation method uses a **hot start** in order to locate the user. A **hot start** is when the GPS device remembers its last calculated position and the satellites in view, the almanac used (information about all the satellites in the constellation), the UTC Time and makes an attempt to lock onto the same satellites and calculate a new position based upon the previous information. This methods has the shortest TTFF among the pure GPS modes (22 sec) (GSM Arena (2014)).
For the WLAN approach, a technology called WPS (Wikipedia (2014d)) was used. Wi-Fi-based positioning system (WPS) are used for positioning with wireless access points. The localisation technique is based on measuring the intensity of the received signal (RSS) and the method of "fingerprinting". The specifications for this approach were drawn by the system designed by a company, SkyHook Wireless ®, that specializes in this field. According to the company's specifications the achievable accuracy of WLAN positioning ranges from 10 - 30 m with a small TTFF of approximately 4 seconds (SkyHook Wireless (2014)).
For the Cellular case there is much controversy about the actual accuracy of the position estimate and the TTFF that it needs. In order to accommodate that a choice was made among the different cellular schemes for localisation. A positioning method called E-OTD

was chosen for that purpose. Enhanced Observed Time Difference (E-OTD) is a standard for the location of mobile telephones. The location method works by multilateration. In suburban areas, as is the one mapped for the simulation, E-OTD has an accuracy from 50 - 250 m. On top of that the TTFF of this methods is 5 - 10 s. By specifying those methods specifications can be drawn for the implementation of the simulation (Wikipedia (2014b)) .

**Wandering User**
In this scenario, the user has no fixed destination but rather wanders in the simulation area. Additionally there is not a point where the user will stop moving as to trust the position estimate. This adds more significance to the network delay in comparison to the previous scenario, as the smaller the network delay, the faster the location estimation will be completed and the correct service selection probability will be higher. The specification of this scenario set:

- multiple randomised destinations for the user
- varying user speed
- service selection while on route

The varying user speed is in relation to the total simulation runs. In each individual run, the agent has a fixed speed. In this approach the context of the user is mapped into values that represent network delay, for both upstream and downstream. According to the Delimitations chapter, all possible interfaces are delimited and is assumed that the user is connected to the network with the mapped performance at all times.
A service selection is done in every iteration of the simulation. The iteration represents the estimate request frequency and both upstream and downstream delays that apply to the sending of the information to a centralised authority as well as the reception of the estimate by the user affect the correctness of this selection based on location criteria (service at shortest distance).
The specifications used for this scenario in the simulation were set as to cover as many cases as possible. More specifically they are:

- Estimation request frequency:

  1: once every (simulated) second

  10: six times in every (simulated) minute

  20: three times in every (simulated) minute

  30: twice in very (simulated) minute

- User speed:

  2: slow movement speed

  4: moderate movement speed

  6: fast movement speed

8: very fast movement speed

10: unrealistically fast movement speed

- Delay range:

  0 - 1000 ms

  3000 - 9000 ms

As far as the speed is concerned, speeds 8 and 10 are not realistic from a real-life standpoint at least in an area as the one that was mapped, however those cases too are examined in order to have a complete image of the effect different aspects and their magnitude have on the mismatch probability.
Where the delay is concerned two different delay ranges were tested, one being afar from the other in order of magnitude. That choice was made in order to measure the impact of delay in the mismatch probability through two extreme cases, one best and one worst case.

## 5.2 Conclusion

In this chapter, the specific system design that will also be followed for the implementation section is laid out. Details about the implemented elements and the reasons behind specific choices over them are explained. Moreover, the scenarios that will be implemented are defined with a certain level of detail in order to differentiate and highlight their purpose.

# 6 Analytical Model

*In this chapter the theoretical background of the method that will be used to extract the results from the simulation will be used. Since the metric of choice in this thesis is the mismatch probability between a correct and an incorrect decision, that is also reflected in the analytical approach. Basic parameters that are also used to define the implementation, are used here and their relation and impact to the Probability of Correct Service (PoCS) individually but also together is explained.*

This kind of system can be described in regard to the possible outcomes as a 2-state space. Either a **correct** or an **incorrect** decision can be the possible outcomes in each measurement. As explained before the investigated environment is restricted by certain parameters that belong to the environment, network or the user scope and they affect the probability of the two possible outcomes.
For that purpose the Probability of Correct Service (PoCS) is defined. It expresses the probability that with given information about the user at a specific point in time, a correct service selection will be made.

## 6.1 The Mismatch Probability

As explained in the analysis chapter, the mismatch is described by the probability that a user will not be found in the estimated space for his position but rather out of it. This probability is affected by the speed of the user, the starting position of the user and the context of the user; where context in this case is the delay in which he requests and receives the estimate.

An abstract form of that probability would be:

$$Pr(X(t_1) < X|X_s) \tag{6.1}$$

where $X_s$ is starting point of the estimation area. From that and based on $v(t) = \frac{dx}{dt}$ (displacement equals velocity integrated relative to time) it can be deduced that generically $x = vt$. From that a more relative to the specific case equation can be deduced.

$$X(t_1) = X_s + v(t_1 - t_s) \tag{6.2}$$

where $t_s$ is the time at the beginning of the request and $t_1$ is the time at which he receives an answer. Through equation 6.2, equation 6.1 becomes:

$$Pr(X_s + v(t_1 - t_s) < X|X_s) \tag{6.3}$$

Before going any further this equation should be thought of according to the aspects examined in this master thesis. So, speed is there and displacement is there. The $t_s$ and $t_1$ can be thought of as the delay the user has at the start of the request and the delay he experiences when he has to receive it. Therefore their difference $t_1 - t_s$ is the delay between the request and the reception of the estimate ($t = t1 - t_s$).
Additionally, the $X$ position defines a segment between itself and $X_s$; at this point it should be clarified that $X_s$ and $t_s$ are not connected ($t_s$ is not the time at $X_s$), even though the notation used seems similar they describe different things. Moving forward, previously $t = t_1 - t_s$ was defined as the delay the user experience due to his context from the beginning of the request until the reception of the estimate. The probability forms like

$$Pr(t < \frac{X_1 - X_s}{v}) \tag{6.4}$$

Therefore the delay between the request and the reception of the estimate should be less than the time it takes the user to travel from his starting point ($X_s$) to some point $X$ that defines the segment. If the user is in that segment at the reception of the estimation then equation 6.1 is true and there is no mismatch and vice versa.

That is greatly affected by the distribution of $t$ from the context of the user. For the purposes of this thesis and chapter, let $t = t_1 - t_s$ be exponentially distributed. This means that:

$$t = t_1 - t_s : \lambda \exp^{-\lambda t} \tag{6.5}$$

Let equation 6.1 be the $f(t)$ and exponentially distributed. The probability of equation 6.1 can be expressed by the calculation of the integral of $f(t)$.

$$Pr(t < \frac{X_1 - X_s}{v}) = \int_{\frac{X_s}{v}}^{\frac{X_1}{v}} f(t)dt = \int_{\frac{X_s}{v}}^{\frac{X_1}{v}} \lambda \exp^{-\lambda t} \tag{6.6}$$

This can be generalised by defining a random variable $\overline{X}$ to declare a random position. By splitting the created segment into $n$ parts, $X_s$ can be defined and equation 6.1 turns into:

$$Pr(\frac{\overline{X}}{n} + vt < \overline{X}) \Rightarrow Pr(vt < \overline{X} - (\frac{\overline{X}}{n})) \Rightarrow Pr(t < (\frac{\overline{X}}{v})) - (\frac{\overline{X}}{nv})) \tag{6.7}$$

The probability for the segment to hold a specific value becomes:

$$Pr(t < (\frac{\overline{X}}{v})) - (\frac{\overline{X}}{nv})|\overline{X} = x) = Pr(t < (\frac{x}{v} - \frac{x}{nv}) \times Pr(\overline{X} < x)) \tag{6.8}$$

By calculating those probabilities through their integrals the equation 6.1 becomes:

$$Pr(t < (\frac{x}{v} - \frac{x}{nv}) \times Pr(\overline{X} < x)) = \int_{X_s}^{x} \int_{X_s}^{\frac{x}{nv}} f_t(t)dt \times f_x(x)dx \tag{6.9}$$

## 6.2 Markov Chain Approach

In order to introduce and calculate the PoCS the following one aspect is assumed and exploited.
Firstly, the trustworthiness of the information is not questioned. At every measurement, all the information used to measure the probability of correct service are drawn from their respective sources. This way, the need for evaluation of trustworthiness for older information is eliminated. Additionally, the information are assumed to not have the same impact on the overall PoCS.

A generalised approach for this model could be modelled using the Markov Model approach and defining 2 states in it. The first state defines the correct service selection and the second state the incorrect. There is the probability of transitioning from both states or remaining there. Those probabilities are affected by the parameters used to calculate the PoCS, their values at the time of measurement and the impact they have on the overall probability.

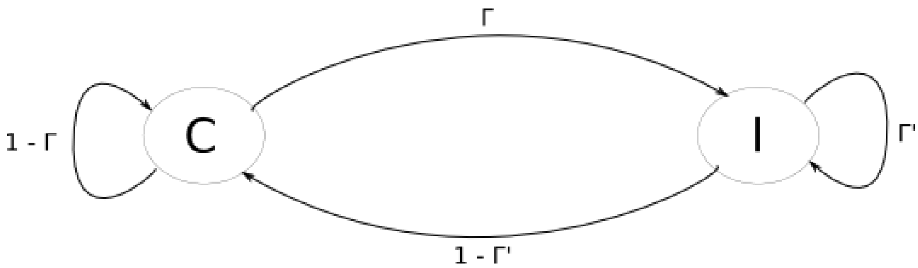


**Figure 6.1:** Discrete time two-event Markov Chain

The image above is a 2-state Markov chain where the state denoted with "C" represents the correct service selection while the state denoted with "I" represents the incorrect service selection. The user, depending on a probability can remain or transit to both states. However as it seen on this depiction the probabilities are inverted.
If $\Gamma$ is the probability of moving from a correct service selection to an incorrect, then the same probability also expresses the chance of remaining to that incorrect state (assuming no parameters change) and that is denoted as $\Gamma'$ in the self-loop of the incorrect state. By the same fashion the self-loop in the correct state coincides with the transition probability of the incorrect state.

## 6.3 The Probabiltiy of Correct Service

Due to the coincidence of location mismatch and wrong service selection decision in the thesis the probability for a correct service decision can be defined as the above probability. That is:

$$PoCS = Pr(t < (\frac{x}{v} - \frac{x}{nv}) \times Pr(\overline{X} < x))$$

The $PoCS$ is calculated as the probability of a location mismatch. It represents the probability of remaining in the correct decision state assuming you started in that and also represents the transition probability from the incorrect decision state to the correct one. Inherently, the transition probability from the correct decision state to the incorrect one and also the self-loop probability in the incorrect state as the remainder probability

$$1 - PoCS.$$

Those four transition probabilities, being based on the generic one presented above and can be represented in a transition matrix as:

$$\begin{bmatrix} PoCS & 1 - PoCS \\ 1 - PoCS & PoCS \end{bmatrix}$$

where each probability in the above matrix respectively is:

$$\begin{bmatrix} P_{CC} & P_{CI} \\ P_{IC} & P_{II} \end{bmatrix}$$

## 6.4 Conclusion

In this chapter the analytical model that mathematically describes the designed approach was explained. The generic Markov Chain approach was outlined explaining the two possible states for the mismatch probability and how the transitional and self-loop probabilities are defined. More into detail, the probability for a location mismatch was investigated taking into account various parameters like the speed and the position of the user. Lastly, a coincidence was claimed between the location mismatch and the wrong service selection which is reflected in the definition of the PoCS. This PoCS was extrapolated and fit into the Markov Chain approach.

# 7 Simulation

*In this chapter, the specific part of the simulation of this project will be presented. The following sections will go through the setup of the simulation, partially presented in the system design section 5.1 as well as simulation-specific details on the approach that was followed to implement the designed system. The setup phase consists of gathering all the necessary components to implement the simulation but also describing the role of those components. The necessary scenarios, as described in the system design section, where implemented using simulation as a tool.*

## 7.1 Building the Simulation

All the previously mentioned functionality is implemented through the Siafu simulator that was presented in the simulation section in the analysis chapter(7). In this section, details about the way those are implemented will be presented. Siafu utilises multiple formats to display and process information. The different formats define different sets of information. Among the formats that are used are:

- .png files
- .xml files
- .java files

The .png files are mostly used to create the areas for the simulation. Those are, *a*) the background image of the simulation, if any, *b*) the walls of the simulation, those are the areas that the agent can or cannot traverse, *c*) the places of the simulation, those are usually user-defined places of interest in the simulated world *d*) and the overlays that define any context that the agents may have. Moreover, there are predefined sprites in Siafu that serve as visualisation for the agents of the simulation. Essentially, the .png files serve as the basis upon which the rest of the functionalities are being built. The figures presented in the Design chapter (chapter 5) are examples of the previously mentioned files for this specific simulation.
Going into more details about the said .png files, one can understand that they constitute different layer of the simulation that, as said, implement different functionalities. However, they can be grouped by the information they represent. This way they can be grouped as "world" images and "context" images. The world images are used to define the world that the agents exists in. Such files include the "background" and "walls" images as well as the images that declare the places that exist inside the simulation playground. Over

them, information about the context of the simulation area (e.g. network performance, Wi-Fi coverage, etc.) are defined also with image files. All of those files are presented in the system design section (5.1).
The .png files that define places are of 4 different types, each one having a different role. The places that define "entrances" are the starting points for all the agents that are in the simulation. Inherently the "exit" places are ending points for some of the agents at least for the first iteration of the simulation. The places defined as "road points" are used in the "Wandering User" scenario where they implement the behaviour of the agents. More into that, the agent changes his destination in every iteration to a random road point, thus producing a wander-like pattern of movement. Ultimately, there is a place layer that defines the parking spots in the simulation "playground". Those are used in order to measure the mismatch probability on the location of the agent and inherently on the service selection,and how that is affected based on the network conditions and the speed of the agent.
Regarding the overlays that describe the cotext that the agent has at each location. They are also made through image files, and they describe the network performance (in terms of network load). This is done due to the delimitation made that users follow a specific pattern over a day span that affects the performance of the network. According to that assumption, three different overlays are created and interchanged depending on the time of the day.

There is actually only one file in xml format but a very important one at that. The config.xml is the configuration file for the whole simulation. It includes from the simplest information, such as the name of the simulated world to the packages of the models used in the simulation. Those models are used to design the behaviour of the simulated world, the agents that are in it and their context. Also, in the configuration file the different context information are defined in addition to the overlays and their type is also defined. The type also defines the format of the extracted context information. Adding to that, the startup time of the simulation is also defined in the configuration file as it can be used to dynamically modify the agents' context. Lastly, the image or map coordinates of the simulation "playground" are defined here to mark and calibrate the background map to the other layers. An example of the configuration file for the designed simulation is given below:

```xml
<worldname>TestLand</worldname>
<!-- What are the calibration coordinates associated to the
        background map? Use decimal degrees.
        Hint: http://www.fcc.gov/mb/audio/bickel/DDDMMSS-decimal.html  -->
<calibration> <!-- fix the coordinates based on map -->
        <topright    latitude="144.03" longitude="1277.03"/>
        <bottomright latitude="692.03" longitude="1277.03"/>
        <bottomleft  latitude="93.03"  longitude="692"/>
</calibration>
<!-- How many seconds does each simulation iteration represent-->
<iterationstep>1</iterationstep>
<!-- Simulation time at startup (World creation date) -->
<starttime>
        <year>2014</year>
        <month>1</month>
        <day>1</day>
        <hour>11</hour>
        <minute>0</minute>
</starttime>

<spritepath>sprites/19x11</spritepath>

<models>
        <agentmodel>de.nec.nle.siafu.testland.AgentModel</agentmodel>
        <worldmodel>de.nec.nle.siafu.testland.WorldModel</worldmodel>
        <contextmodel>de.nec.nle.siafu.testland.ContextModel</contextmodel>
</models>
<!-- For each overlay, what is the tag the simulator should associate to the
        pixel value? -->
<overlays>
        <Performance type ="real" />
</overlays>
```

**Figure 7.1:** Configuration file for the Simulation

Last, but not least, building block of the simulation are the .java files. In the java files, the whole of the behaviour of the agents is implemented along with any dynamic changes in the world model and in the context model if they are needed. Moreover, through the .java files the handling of the information provided from the previously mentioned .png and .xml files becomes possible. The code that is included in the .java files is rather lengthy and not all of it is relevant to explain the concepts that were designed for the purposes of this thesis. The code itself will be included in the appendices at the end of this thesis but will not be further analysed unless this is deemed necessary further on for specific parts of it.

## 7.2 Preparing the Simulation

It became apparent from the beginning of the project that the two approaches that are handled would be diverse. On one hand, the "Localisation at Destination" scenario treats the user's context as possible noise at the receiver end. On the other hand the wandering user scenario, where the context of the user is translated to transmission delay investigates an entirely different possibility for the case of location-based service selection.
In the wandering user approach, agent, environment and network parameters that can affect the probability for a mismatch and degrade the performance of the service selection process, include:

- speed of the agent
- direction of the agent
- network delay (at the location of the agent)
- parking place density

All of these parameters are in control of the developer of the simulation and their impact on a specific task performed by the agent (e.g. service selection) can be investigated.

### 7.2.1 Agent parameters

Speed and direction are two parameters that fall under the agent category because they do describe the behaviour of the agent, in the simulation and hereby known as agent. The speed of the agent can affect in a great degree the service selection process as the process itself is based on the localisation of the agent, and possibly an accurate one at that. The higher the speed of the agent, the faster his location changes.
The case for the agent's direction is somewhat different as it is limited and bounded by physical restrictions. That is, the inability of the agent to have a completely free movement in his environment. Generally. the collectivity of pedestrians, cars, etc., are bounded by pavements, roads and other structures that guide but also restrict their movement and their direction. This thesis takes this as a fact and does not go into affecting the direction of the agent directly, rather than leaving to be set by the road layout that was designed.
Specifically for this simulation the only agent parameter that allows for modification and can affect the performance of the service selection is the speed of the agent. This is in correlation to the network delay (explained below) as with the same delay and higher speeds a mismatch in the agent's location, resulting in a wrong service selection, is more likely.
Specifically for this simulation, the agent speed can vary in a range from 2 to 10 with a step of 2. Normally, 2 is the slowest speed and 10 the fastest. An additional parameters on the side of the agent is the estimation frequency, that is how often a request a position estimate is placed on the server. As this is a simulation, it is not directly represented by the tool. Instead, there exists a variable that defines the number of seconds simulated in each iteration of the simulation. In the implemented simulation a location estimate is requested in every iteration and by the way it is implemented we can virtually increase or decrease the frequency of the requests. Specifically, the iteration step grows from $1 \rightarrow 10 \rightarrow 20 \rightarrow 30$ sec. This means that the requests can be as frequent as 1 every second to 2 per minute.

### 7.2.2 Environment parameters

The environment that the agent moves in, plays an important part on the performance of the service selection. That is always in correlation with the kind of service that an agent must select from and in what grade that service is present in the agent's environment. This thesis does not go into specifics about the service discovery but assumes that the agent is aware of all the available services in his region and has to make a correct service selection based on his location. A centralised service selection would possible become a major handicap for the agent since the position estimate the server generates and will base the service selection on, can be wrong depending on the network conditions, agent's speed, estimation frequency and density of available services. The general case would be to associate the location accuracy with the size of the service area and how this affects the performance of the service selection. However, due to restrictions of the simulation tool the aforementioned approach has to be simplified and seen from a different angle. In this thesis, the concept of multiple service areas of different size is restricted by the used simulation tool (Siafu). Due to this a workaround was formulated. The multiplicity of places is kept, however each individual place defines a different service. This way, the location mismatch can almost coincide with a service mismatch. By increasing the density of the places (and the services), the effect the previously mentioned parameters can be seen more easily and accurately. This can be depicted easily through the following figure:

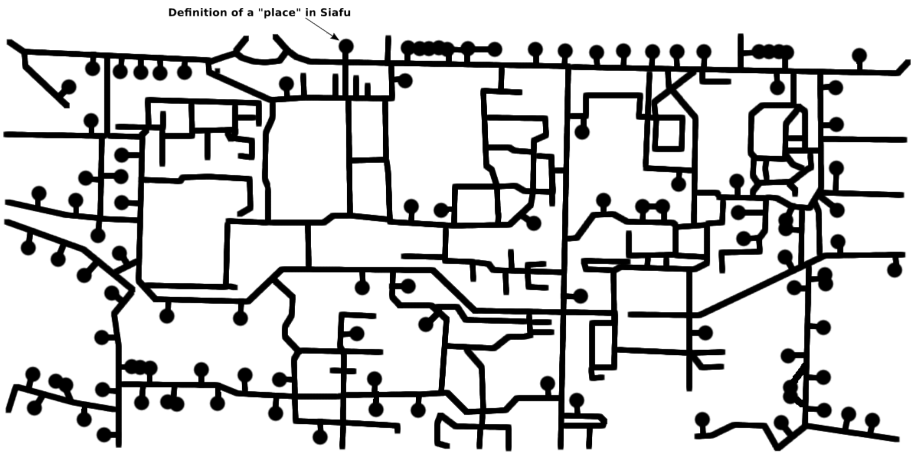


**Figure 7.2:** Definition of a simulated place

This is in direct correlation with the agent parameters, presented above, as the speed of the agent can also greatly affect the performance of the service selection when there is a dense environment of service areas close to each other.
Additionally, the service selection can be affected by the network conditions at the agent's position.

**Emulating Random Movement - Road Points**
In the "Wandering User" scenario a wander-like movement had to be implemented for the agent. Instead of the agent immobilizing itself and then requesting for a position estimate, the agent now wanders around the map and asks for the correct service selection based on his location. This was achieved by taking advantage of a feature of the simulation tool. There has been implemented a set of "places", namely "Road Points" that the agent, upon creation has as destinations. Once a point has been reached a new one is assigned causing the desirable wander-like movement. Those point were set near or on cross-roads on the map in order to create a more life-like movement when the destination of the agent was re-assigned.
It was chosen to be implemented like this because ideally the control of the final transaction remains for the agent. This is against the centralised approach were everything is taken

care of by the central authority (server) because it is more susceptible to delay constraints.

### 7.2.3 Network parameters

In the simulation the context of the user is split into three different periods, pre-work, work and post-work. Over those periods the context of user is changing for the mapped area. The principle behind this separation is the assumption, as previously mentioned, that the agent follow a specific moving pattern during the day. Due to this, three different overlays were designed to express the load for these three periods. Namely they are:

- pre-work overlay
- work overlay
- post-work overlay

The information that the above overlays represent are generic in the sense that by themselves they do not have a meaning; meaning is given to them by the mapping performed on those values. The extracted values of the user's context can be mapped to the network aspect e.g. delay, noise that fits to the respective scenario.

#### Pre-work Overlay

The pre-work overlay defines a time-span of roughly two hours in the simulation. That is from 6 a.m. to 8 a.m. where the users are located to their homes, waking up, getting ready for work. While this happens the network load in residential areas increases during that hours. This behaviour, while is not directly shown during the simulation, it is what the network load for that time frame is based on. As defined in Siafu, the darker the colour the lower the value. These values can be extracted and handled. More on how this is achieved here will follow later.

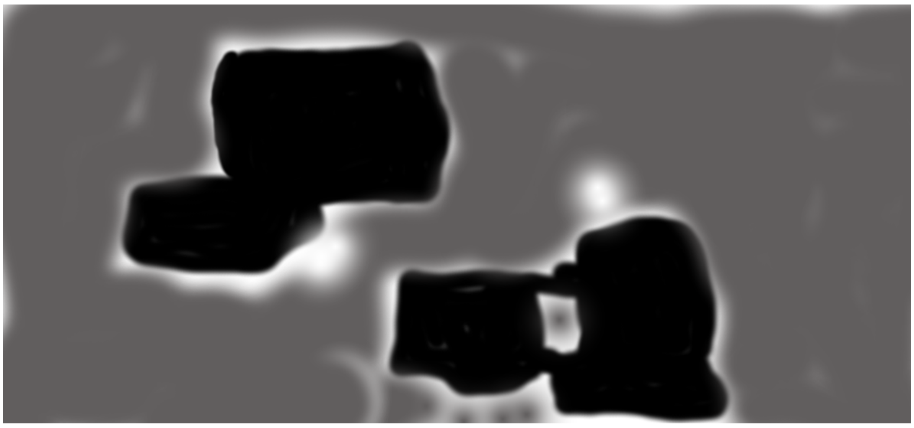

**Figure 7.3:** Sample overlay that depicts the network performance of the time period between 6 - 8 a.m.

#### Work Overlay

The work overlay defines the longest time frame in the simulation. The work hours define a period between 8 a.m. and 4 p.m. when the majority of the users are at work. This results in a drastic decrease of the network load (the performance increases) in the residential areas of the map while the network load increases drastically in areas near and at work areas (e.g. university buildings). The same principle applies regarding the representation of the values.

**Figure 7.4:** Sample overlay that depicts the network performance of the time period between 8 a.m. - 4 p.m.

### Post-work Overlay

In the post-work overlay a similar to the pre-work network load layout is defined. This happens due to the inverse behaviour of the users which causes the same effect regarding the performance of the network. When the users leave work and head towards the residential areas, there happens the same phenomenon; The load in the university building drops and the load in the residential areas increases. The post-work overlay is the last overlay used in the simulation and it defines a time-frame of again two hours from 4 p.m. to 6 p.m. when the simulation is terminated. The same value representation applies as before.

**Figure 7.5:** Sample overlay that depicts the network performance of the time period between 4 p.m. - 6 p.m.

### 7.2.4 Wandering User

In the "Wandering User" scenario, the network aspects that can directly affect the location estimation's reliability and inherently the service selection is network delay and that is what the context of the user is mapped into. This is due to the importance of a timely upload, calculation and download of the necessary information from and to the agent, especially in a centralised architecture for the position estimation that is assumed here.
In Siafu this representation of delay is performed through a matching between pixel values and delay groups having as a criteria the number of digits (how great is the pixel value). To improve the understanding of the previous clause, here is a visualization effort:

| Example Pixel Value | Group | Mapped Value | Explanation |
|---|---|---|---|
| 165777299 | 8 digits | random value between 800 - 1000 (ms) | As dictated by Siafu, the larger the pixel value, the larger the delay. The mapping complies. |
| 4577923 | 7 digits | random value between 600 - 800 (ms) | |
| 569213 | 6 digits | random value between 400 - 600 (ms) | |
| 45612 | 5 digits | random value between 200 - 400 (ms) | |
| 0 | single digit | random value between 0 - 200 (ms ) | |

**Table 7.1:** Sample mapping of the pixel value (extracted by Siafu) to meaningful context for the user

The method used in the implementation for this matching is Java's Math.random(). The usage of the Math.random() method provides a uniform distribution of the mapped values according to the calling frequency.
By this matching, a delay that is representative of the pixel value it originates from but not bound to it is produced. This delay is utilised afterwards to simulate the network condition at the agent's position.
Adding to that, three different context overlays, describing the network load in pre-work, work and post-work hours, are used in order to see how different network loads in the same areas can affect the mismatch probability.
The aforementioned matching was based on the image files that were created to depict the network load in the simulation in each different time during the day.

| Technology | Accuracy | TTFF |
| --- | --- | --- |
| GPS | 5-10 m | 22 s (Hot Start) |
| WLAN | 10-30 m | 4 s |
| Cellular | 50 - 100 m | 7.5 s (E-OTD) |

**Table 7.2:** Specification used for the separate localisation interfaces

### 7.2.5 Localisation at Destination

In the "Localisation at Destination" scenario, the context of the user extracted from the simulator is mapped in such a way that it provides an estimate of the magnitude of position shift certain levels of noise in the network can produce. Additionally, the magnitude of the position shift can change according to the interface used for the localisation. Not all different localisation methods are equally prone to error due to network noise and can produce the same accuracy due to technical limitations. In Siafu, as before, this representation of position shift is achieved through a matching between the extracted pixel value at the user's position and a position shift according to that value. Later on that shift, that is in terms of pixels, is added to the user's current position to produce the final shifted position and compare the service selection of the two positions based on the service that has the shortest distance to both positions.
Now, since the coordinate system used in Google Maps ® and the coordinate system of Siafu do not match a solution had to be found. It was found after verification with the online map that the scale of the map that was used in the simulation is 1:10.000; this means that 1cm on the map represents 100 m in the real world. Now if this knowledge is inserted to the existing map we get that for the scaled image's shortest dimension, measuring 13cm, we have 549 pixels. Since every centimetre represents 100 m and there are 14 cm in the vertical dimension the total in meters is 1400 m. If this was to be extrapolated in map coordinates the points based on the vertical axis would represent the latitude of the user. Based on this scaling and the number of pixels:

$$1400/549 \approx 2.5m/px$$

Since a position shift means a shift in both longitude and latitude, there had to be a calculation for the longitude coordinate as well. In the same fashion the mapped can be measured to be 30 cm long on a 1:10000 map scale. This means that there is a 3000 m distance described by 1186 pixels. This means that:

$$3000/1186 \approx 2.5m/px$$

It is seen that the conversion for both dimensions is approximately the same. This conversion is needed due to the coordinate system the Siafu simulator uses. In Siafu the coordinates are set based on the pixel size of the image starting from (0,0) and going up to (x,y) according to the set values, in this case (1186, 549). Since the position shift is done by modifying the coordinates of Siafu this conversion allows for a more accurate shift of the position according to the technology used. Research that was conducted for the three separate technologies used for the localisation for the user lead to the following specifications for each one. Those specifications will be used in the simulation to evaluate their performance. The accuracies of the three technologies used are presented below:

While the TTFF for GPS and WLAN is explicitly defined, the cellular interface has a TTFF that can range from $5 - 10\,\text{sec}$. In order to remedy this situation the arithmetic mean of the two edge values is used instead; that is $(5 + 10)/2 = 7.5\,\text{sec}$.This way the shift can be designed to match each different technology under the user's context. Additionally, since in this case the differentiation is based on the technology used for localisation and the fact that it happens only when the user is at a destination point, the estimate request frequency has no meaning here. A depiction of the how things happen in this case is:

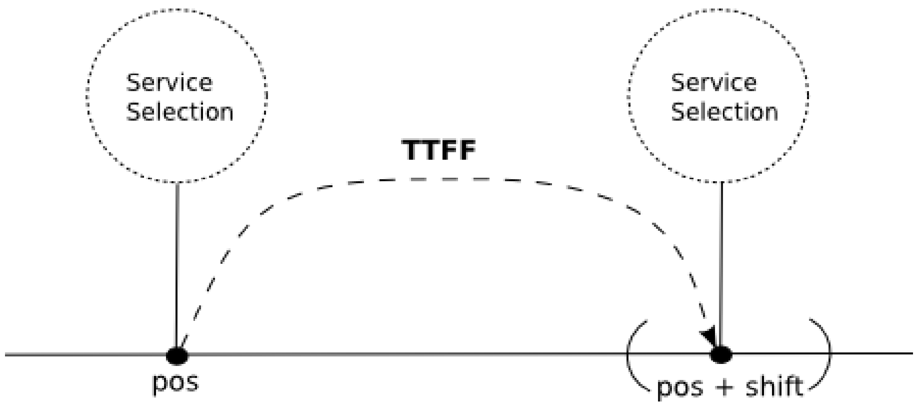


**Figure 7.6:** How the positions and their respective services are calculated in the "Localisation at Destination" scenario

### 7.2.6 Application Emulation - Multiple Threads

Going into more detail about the implementation, multi-threading was implemented to emulate the application layer of the simulation. Multi-threading additionally facilitates the multiple requests for location estimate according to the pre-set frequency. Since Siafu itself has a limited set of commands to interact with the simulated world and more than that, its context, the multi-threading approach was used to improve the utilization of this set of commands.
This is achieved by introducing a pool of threads that the simulation can pick from to assign the execution of the function that acts as the application layer. At each iteration step, the execution assigns a thread to the called function and it is executed in parallel with the rest of the simulation. This parallelism allows for the mapped network delay to be implemented.
The specific definition of the set-up used for emulation is:

- the emulator function
- thread pool consisting of 16 threads
- one executor service

The emulator function emulates the application layer in the implemented simulation. The measured agent is passed as a parameter to the emulator function.
The emulator function uses one of the 16 available threads in the pool at any given time.

If all the threads are unavailable at the time of the method invocation an exception is thrown. This handling of the threads is executed by the executor service.
In this parallel environment, the agent's context is extracted and manipulated, as explained before in the Network Parameters subsection, to simulate the delay at the agent's position. This representation of the delay is possible since the threads that are handling each task become separate during execution time. The thread that emulates the application layer is interrupted by an equal amount of milliseconds to the extracted context of the agent. Since the simulation continues to run regardless of the state of the emulator it is possible to know the context of the user at each position and use it to simulate downstream or upstream delay.
Lastly, the internal implementation of the application layer has the advantage of enabling the monitoring of the impact the estimate request frequency has on the correct service decision, a thing that was not possible with the external application approach.

## 7.3 Conclusion

In this chapter, specific information about the implemented simulation were presented. The Siafu tool was a bit more extensively explained in regard to the ways certain things are implemented through it. Additionally, those methods were extrapolated to the specific case investigated in this thesis and the details on how these were implemented were explained. Following that, relevant variables in the Siafu simulation (e.g. agent speed, mapped network delay) were presented. It was done so due to their relevance to the extracted results and their impact on them. The following chapter will present the results extracted from the aforementioned designed simulation and draw conclusions from them.

# 8 Results

*In this chapter the results extracted from the simulation runs will be presented. Following, a comparison of the results will be performed.The extracted results are based on the delimitations made throughout the construction of the simulation. The mismatch probability calculated for each implemented scenario is presented and is compared both in an internal setup to identify the impact of each parameters on the specific scenario but also in regard to the other scenarios to identify the different approaches and how they improve or deteriorate the performance of this specific metric.*

Calculating the PoCS is extremely useful especially in cases where multiple context information that can affect it are available. Based on the PoCS, an autonomous system or even the user of the system could decide what actions could be taken in order to improve the performance of the said system. That is, try to maximise the PoCS according to the available context. Since the location accuracy and in turn the performance of service selection strongly depends on the user's context, the PoCS can:

- give the ability to categorize context according to the effect it has on PoCS in each individual case.
- provide the user with options for performance increase according to his environment.
- provide a numeric value that represents the degree to which an estimate can be trusted thus the degree of probable correctness of service.

The results presented below are used in order to evaluate the designed scheme. Afterwards, and based on those results, conclusions will be drawn as to what degree the context of the user can affect the performance of the service selection. Based on the results, proposals may be made in order to keep the PoCS at high levels under any context.

## 8.1 The Localisation at Destination Case

In the "Localisation at Destination" case, another form of the user's context is explored. Now, instead of network delay, noise is assumed for the receiver end of the communication.The noise on the receiver's end results in shift in the user's location. This shift however is greatly affected by the interface used for the user's localisation. Due to the accuracy of different localization interfaces, the user cannot be positioned as accurately given a specific noise level according to his context.

The results are illustrating this high difference in the localization interfaces as the difference in accuracy is evident (as is described in the Design chapter).

The first approach is decoupled from the estimation request frequency as the user checks for service selection based on his location only when at a destination. Moreover, the speed of the user does not directly affect the localisation due to the same fact that the localisation happens only at the destination. It indirectly affects the rate at which the user reaches his respective destinations. The parameter that is heavily linked to the shift of the position of the user and is examined in this approach is the interface used for localisation. The results on the table below were derived with the following characteristics:

- user speed: 5 (fast)
- request estimate: at destination

| | | |
|---|---|---|
| GPS | Prework | 50 % |
| | Work | 27 % |
| | Postwork | 20 % |
| WLAN | Prework | 64 % |
| | Work | 63 % |
| | Postwork | 90 % |
| CELL | Prework | 72 % |
| | Work | 80 % |
| | Postwork | 92 % |

**Table 8.1:** Mismatch probability of the different technologies under all the different contexts separately for the Location at destination case.

In table 8.1 above however, the difference in accuracy is illustrated for all the different contexts of the user in the different periods of time. The GPS interface suffers the least from noise on the receiver end as it is mostly based on the signals sent by the satellites and the noise can be dealt with through correction techniques to maintain the accuracy into a certain margin. The WLAN on the other hand exhibits a higher error rate as it is restricted by the technical limitations of the WLAN network, for example, the range of an access point. Cellular localisation shows the largest mismatch probability under all the different context and that is caused by the low accuracy of this method in exchange for the best coverage.

### 8.1.1 Localisation at Destination - Moving User Case

Adding to this scenario, a separate simulation was built to examine the impact different technologies have on the reliability of the location estimate. In this simulation, two subcases are examined; results are extracted for both the cases where the interfaces adhere to their respective TTFF and where they do not. For all the different speeds of the user results are drawn for $TTFF = 0$ and for $TTFF =$ (assigned value for each interface). The assigned values and the reason they were chosen as such are presented in the Design chapter (chapter 5). By this comparison it can be seen how feasible it is to use a specific interface under certain conditions.
In those simulations three different speeds for the user were used, slow, medium and fast.

Moreover, instead of tracking the user when at a destination, the moving user is being tracked constantly with a very fast interval. This was done for all the different contexts but the results were grouped and presented on an Interface / Movement speed basis. Following are the extracted results of this simulation:

In the 8.1 graph it is easily seen that, regarding the mismatch probability and its correlation with the position shift, a pattern is evident. This is mainly caused by the network load distribution in the context overlays. Due to that, the context value that covers the biggest part of the mapped area gets the largest part of the sampling as well. While it is expected it is not the rule, there can be variations and for those responsible is the randomised movement of the user to each destination.
Moreover it can be seen that the mismatch probability differs, in most cases, the least between the "TTFF applied" and "no TTFF" values calculated.

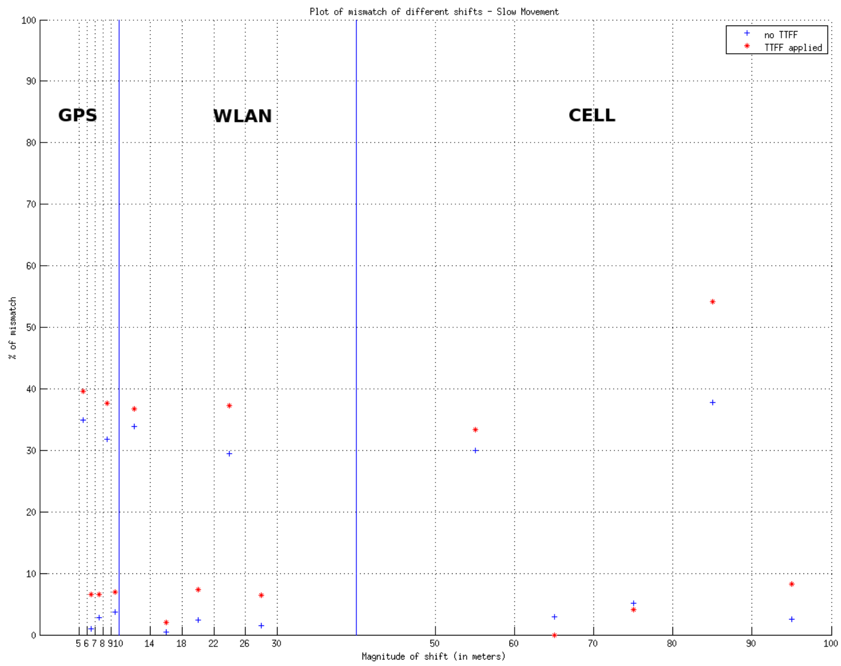


**Figure 8.1:** Comparison of the mismatch probability for different interfaces for the slow speed

For the medium movement speed case represented in graph 8.2 a scaling in the mismatch probability is apparent as the speed of the user increases. For GPS the "no TTFF" mismatch probability goes a bit over 50 % for the "8-9 m" gap as well as the "22-26 m" gap for WLAN goes up a little below 50 % which is a huge increase compare to its value in the slow movement graph. What also increased in some cases is the gap between the mismatch probabilities calculated for "TTFF applied" and "no TTFF". For instance the biggest difference can be found for the cellular interface in the "50-60 m" gap which is a bit more than 20 %. Additionally what one can observe on this graph is an inversion that happens between the "no TTFF" and "TTFF applied" mismatch probabilities. Normally, it would be expected for the "TTFF applied" to be always higher due to the lock time that is introduced, which is what happens for most cases. The cause for this is mainly the accuracy that each individual interface provides. It is less evident (by magnitude) in the case of GPS because the shift margin is a lot tighter than it is for the WLAN and cellular interfaces. As the accuracy lowers and there is no introduced delay the more probable it is for a mismatch to occur.

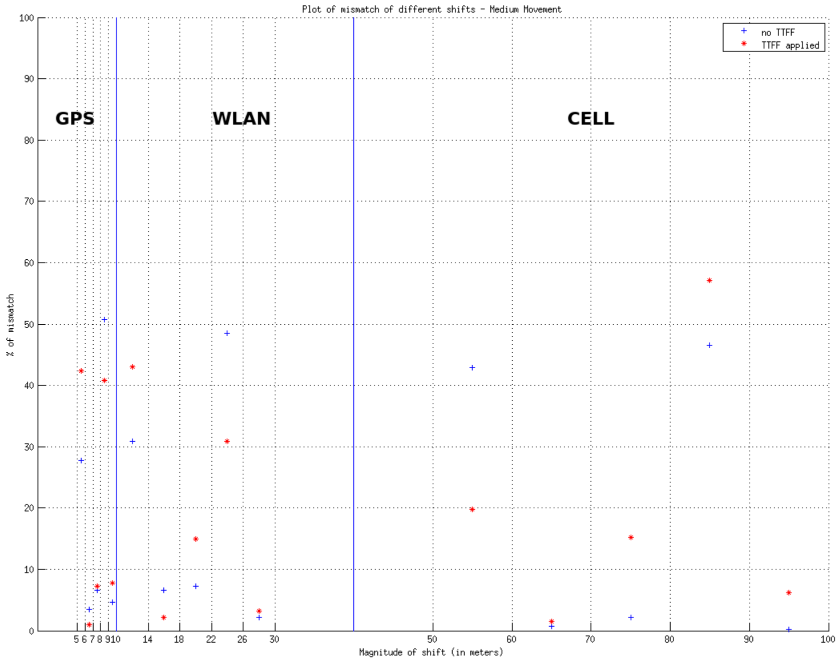


**Figure 8.2:** Comparison of the mismatch probability for different interfaces for the medium speed

For the fast movement speed case the results can be found in graph 8.3 while the maximum values for the mismatch probabilities remain in more or less the same levels as in the medium movement speed case (that is between the 40-50 % margin) the difference between the two cases ("TTFF applied and "no TTFF") is smaller. While observing the values for the cellular interface it is easily seen that the mismatch for the two cases are very close, in some cases almost coinciding. This happens due to the speed and the provided accuracy. The speed of the user in this case is so fast and the tracking accuracy of the cellular interface is so low that adding delay to the process affects it only a little. This however does not show so evidently in the case of the GPS interface. The GPS interface, as explained in design section has a 22 sec. lock time which quite large, so even though the user is moving very fast this long lock time prevents the two probabilities from coinciding.Another thing that might have been expected but isn't so evident in these graphs is the increase of mismatch probability per distance grid as the distance increases, both in count and magnitude. This is two-fold; firstly the count is not equally distributed due to the context overlays that were designed. Since the context value is weighted by its digit count (pixel value) and some values or range of values are favoured over the performance map it leads to this behaviour. Secondly, the implementation of calculating the position shift of the user's position according to interface used is taking into account both directions due to the fact that it is impossible to know where the estimation will be calculated. So, assuming the estimation area is a circle around the user, two positions are calculated, one for the hemisphere in front of the user and one for the hemisphere behind the user.

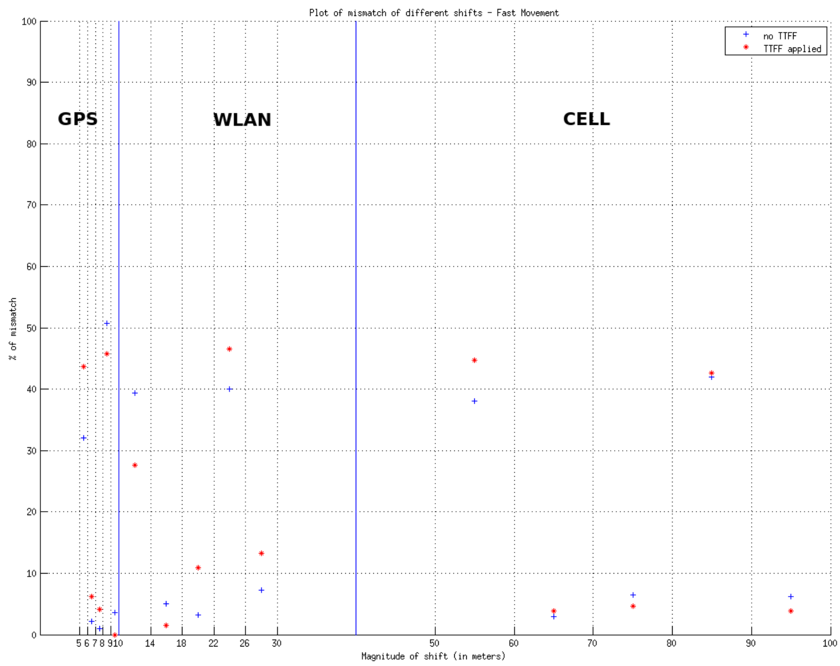


**Figure 8.3:** Comparison of the mismatch probability for different interfaces for the fast speed

In graph 8.4, it can be seen that there is indeed an expected trend for the overall mismatch over a day span for each interface as the speed of the user increases. The high lock time of GPS and the low accuracy of cellular localisation are keeping their "TTFF applied" values close to and at 100 %. The WLAN interface has the least steep slope as it provides medium accuracy (10 - 30 m) and relatively fast lock time (4 sec). However, one might argue that GPS with no lock time and such high accuracy has a quite high mismatch probability ($\approx 74\%$) but this is where the service density comes in. Many services (108) were created onto the mapped area to investigate the effect of service density on the mismatch probability. As they are all different, a mismatch in location inherently leads to a wrong service selection. From that it can be deduced that even with a high accuracy (and possibly low lock time) in the presence of a dense service environment the mismatch probability cannot be reduced as easily.

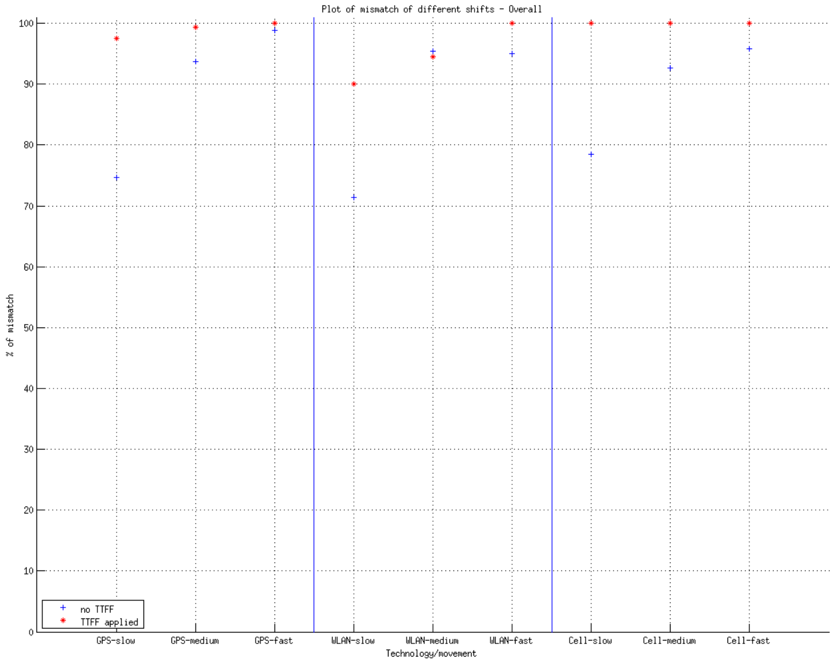


**Figure 8.4:** Comparison of the collective mismatch probability for different interfaces overall the different speeds

## 8.2 The Wandering User Case

The biggest challenge comes from the "Wandering User" scenario where the user is in constant movement while he is requesting position estimates with a predefined frequency. The three variables that are examined in the simulation are:

1. the speed of the user
2. the context of the user
3. the frequency of the estimation requests.

The speed of the user, as stated in the previous chapter, varies between the values of 2-10. The context of the user has fixed values for specific intervals and is separated into three different time-zones, namely pre-work, work and post-work hours. The network load that is mapped to network delay values is assumed not to change in each time-zone. Lastly, the frequency with which the user is requesting the position estimates can go from once every second to twice per minute.

The tables below are categorized based on the estimate request frequency that was used. The mismatch probability represents the whole day span that was measured, from pre-work to post-work hours.

In the simulations that have a position estimation frequency of 1 (that is every second), the mmPr is at the lowest levels of all 4 estimation request frequency settings. This happens due to the better image that a frequent update offers about the position of the user; combined with the relatively low delay introduced produces the lowest mismatch. What is not taken into account is the amount of messages that will potentially flood the network by such a high request rate.

| Speed | Estimate Request Frequency | mmPr |
|---|---|---|
| 2 | 1 | 22.9% |
| 4 | 1 | 27.2% |
| 6 | 1 | 26.9% |
| 8 | 1 | 24% |
| 10 | 1 | 26.2% |

**Table 8.2:** Mismatch Probability over a day span for the different speeds of the user given a very high estimate request frequency for the best case (1 second)

| Speed | Estimate Request Frequency | mmPr |
|---|---|---|
| 2 | 1 | 92.6% |
| 4 | 1 | 96.6% |
| 6 | 1 | 95.6% |
| 8 | 1 | 96.5% |
| 10 | 1 | 95.7% |

**Table 8.3:** Mismatch Probability over a day span for the different speeds of the user given a very high estimate request frequency for the worst case (1 second)

On the other hand when a position estimation is requested less frequently, the mismatch probability has a growing trend according to the speed of the user. However, it can be seen that this notion is not always true. In the cases where the speed increases but the mismatch drops, the service density comes into play.

| Speed | Estimate Request Frequency | mmPr |
|---|---|---|
| 2 | 10 | 26.6% |
| 4 | 10 | 28.2% |
| 6 | 10 | 29.6% |
| 8 | 10 | 32.9% |
| 10 | 10 | 29.4% |

**Table 8.4:** Mismatch Probability over a day span for the different speeds of the user given a high estimate request frequency for the best case (10 seconds)

| Speed | Estimate Request Frequency | mmPr |
|---|---|---|
| 2 | 10 | 89.3% |
| 4 | 10 | 98.8% |
| 6 | 10 | 89.5% |
| 8 | 10 | 99.2% |
| 10 | 10 | 86.9% |

**Table 8.5:** Mismatch Probability over a day span for the different speeds of the user given a high estimate request frequency for the worst case (10 seconds)

In this thesis, the designed approach tries to make the two concepts of location and service mismatch coincide since the concept of a service area is restricted by the simulation tool. Under that context, the denser the environment, the higher the possibility for a mismatch and vice versa. So the factor that can affect indirectly the user is his movement in regard to his environment. In the "Wandering User" scenario the user is moving randomly on the map.

| Speed | Estimate Request Frequency | mmPr |
|---|---|---|
| 2 | 20 | 28.7% |
| 4 | 20 | 33.4% |
| 6 | 20 | 35.7% |
| 8 | 20 | 33.2% |
| 10 | 20 | 33.4% |

**Table 8.6:** Mismatch Probability over a day span for the different speeds of the user given a relatively low estimate request rate for the best case (20 seconds)

| Speed | Estimate Request Frequency | mmPr |
|---|---|---|
| 2 | 20 | 90.5% |
| 4 | 20 | 100% |
| 6 | 20 | 100% |
| 8 | 20 | 100% |
| 10 | 20 | 100% |

**Table 8.7:** Mismatch Probability over a day span for the different speeds of the user given a relatively low estimate request rate for the worst case (20 seconds)

Additionally, the mismatch gets higher as the estimate request frequency increases. That is expected as the change in the user's location is greater when the frequency of the position estimate requests is lower. But even in this case, the density of the services and the movement of the user can affect the results to some extent.

| Speed | Estimate Request Frequency | mmPr |
|---|---|---|
| 2 | 30 | 22.9% |
| 4 | 30 | 27.8% |
| 6 | 30 | 35.7% |
| 8 | 30 | 35.5% |
| 10 | 30 | 38.2% |

**Table 8.8:** Mismatch Probability over a day span for the different speeds of the user given a very low estimate request rate for the best case (30 seconds)

| Speed | Estimate Request Frequency | mmPr |
|---|---|---|
| 2 | 30 | 100% |
| 4 | 30 | 100% |
| 6 | 30 | 100% |
| 8 | 30 | 98.1% |
| 10 | 30 | 100% |

**Table 8.9:** Mismatch Probability over a day span for the different speeds of the user given a very low estimate request rate for the worst case(30 seconds)

The following graph that includes all the measured data based on the estimate request frequency. In the graph, it can be seen that the greatest influence on the location mismatch is from the frequency with which the location estimate is requested.Additionally, a correlation between the speed and the request frequency can be seen. As this is not always the case the density of the services in a specific area affects the mismatch but this parameter is entirely based on the movement of the user.
In the case where the position estimate is requested most frequently (blue line) , the mismatch is maintained at a reasonable level in correlation with the speed when the speed has the maximum value as the position estimate and its frequency can better represent the user and his changing positions despite the introduced delay.

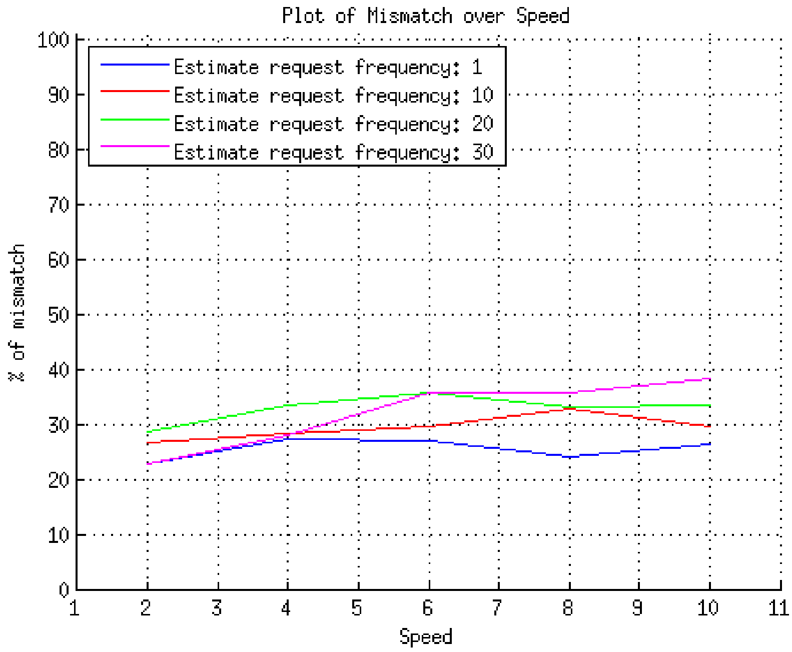


**Figure 8.5:** Mismatch Probability all the whole day span for all the discrete user speeds and for all separate estimate request frequencies

Against that, in the second scenario it is observed in the figures 8.5 and 8.6 that a significant increase on the delay drives the mismatch probability over 80% thus plummeting the reliability of a service selection based on the user's location. The behaviour for each discrete estimation request frequency is roughly the same with the lowest frequent (30: twice / minute) producing the largest mismatch probability in both cases.

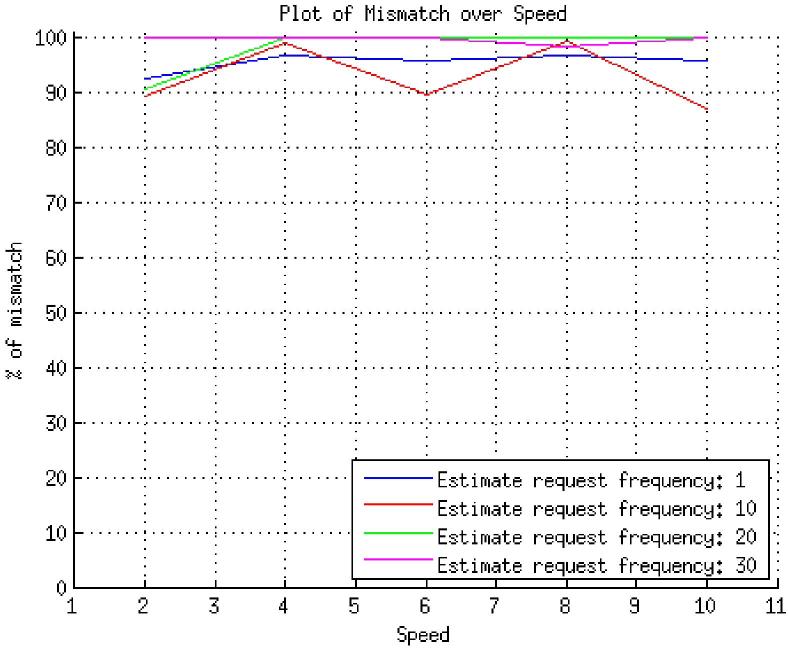


**Figure 8.6:** Mismatch Probability all the whole day span for all the discrete user speeds and for all separate estimate request frequencies for the second case

In the following graphs 8.7 and 8.8, the variation of mismatch over the three different time periods defined in the simulation for a specific speed (in this case the slowest speed used, 2) for the two separate delay ranges that were introduced. In graph 8.7 it can be observed that for the slowest measured speed all the probabilities across the three different contexts maintain an almost linear, slightly downward trend in at least 3 out of 4 different frequencies. Since in this graph the mapped delay is low the slow speed of the user produces a lower mismatch as the shift in position is not so great over time. Generally the mismatch probability is kept at low levels on a per context basis.

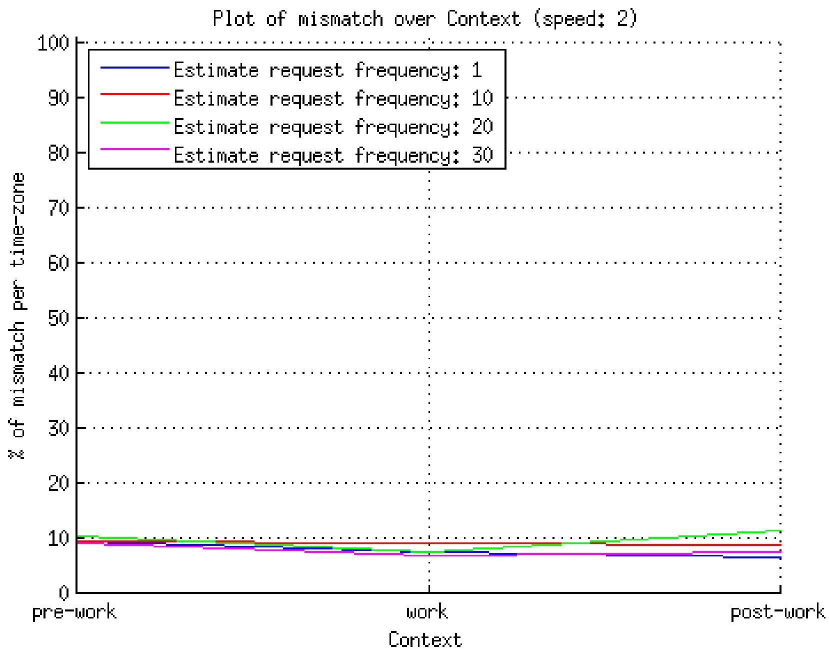


**Figure 8.7:** Evolution of mismatch over all context for a fixed speed and for all the different estimate request frequencies

Against that, in the second scenario seen in graph 8.8 it is observed that even though in the first case the lowest speed presents with the lowest mismatch values for all iterations the same cannot be verified for the second case where the delay margin is much higher. This argues for the clause that with a high enough delay in the user's environment the speed of the user, no matter how low, has little if any, impact on the reliability of the user's location and inherently to the performance of the service selection based on that location.

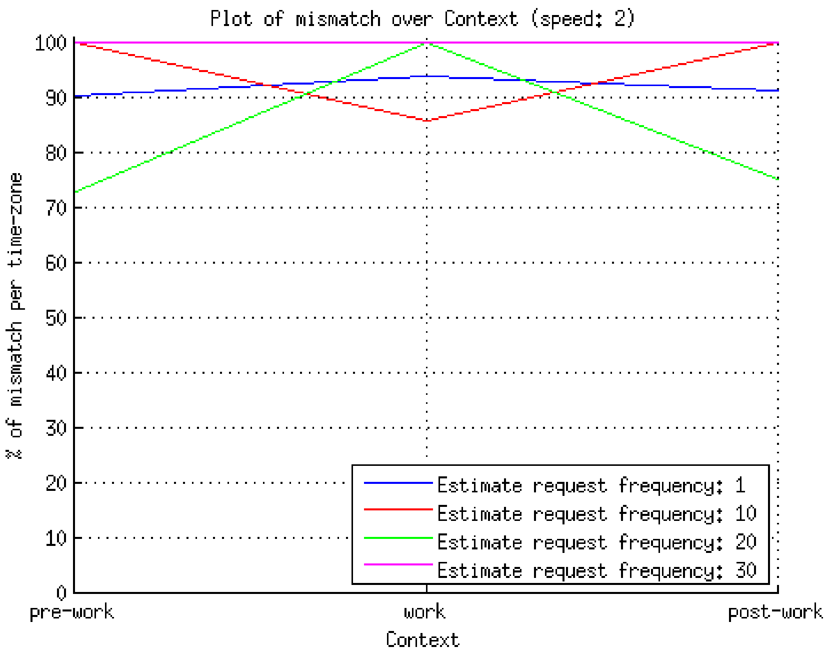


**Figure 8.8:** Evolution of mismatch over all context for a fixed speed and for all the different estimate request frequencies for the second case

In the two graphs below, 8.9 and 8.10, the average mismatch probability is represented over all iterations. The average was taken over all the measured speeds. In graph 8.9 it can be seen that the average mismatch over all the speeds creates an upward trend for all the context; as the speed increase the mismatch follows an increasing trend as well but a few exceptions and this is reflected in the average. The average mismatch probability generated by each context and across all iterations seems to stabilize around 10%. The context that seems more linear than the others is the work context and that is mainly cause by its high duration it has in the simulation (8 a.m. - 4 p.m.) which produces more samples and a more accurate mismatch probability representation.

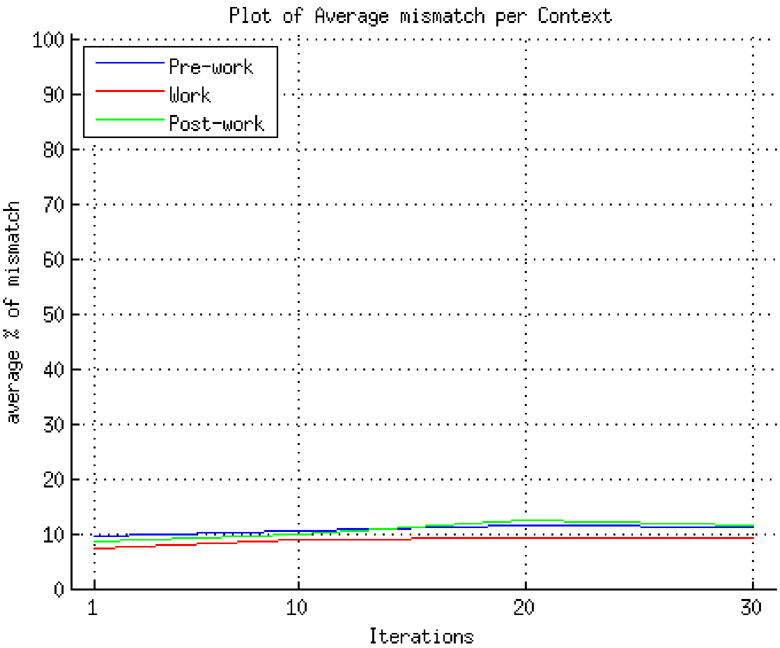


**Figure 8.9:** Average Mismatch for each context and iteration step

Against that, in the second scenario, in graph 8.10 it is observed that even though there is a slight upward trend on the average mismatch in the first graph across all iterations for all the available contexts the impact delay has on them is presumably minimal. Something different is observed in the second graph; the delay experience in that environment is so large that the behaviour of the mismatch probability seems erratic and with quite some magnitude.
This is an argument against the impact estimate request frequency has on the mismatch probability. With the mismatch probability of all the speeds averaged and plotted against each iteration little change is observed in the average mismatch probability( in graph 8.9 for the first case). However this doesn't happen in the second case (graph 8.10). There the delay is so high that even the average of all speeds per context fluctuates too much among the iterations. The only visible pattern for the second case is seen for the lowest frequency where all average mismatches converge to 100%.

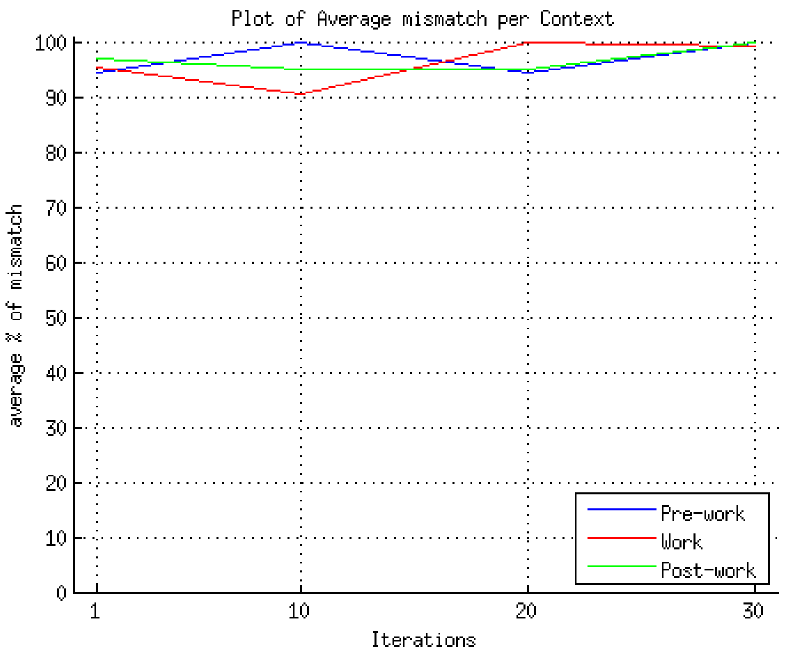


**Figure 8.10:** Average Mismatch for each context and iteration step for the second case

In summary of this chapter certain corollaries can be drawn for the individual cases that were presented. For the first case examined "Localisation at Destination" it can be observed that the two approaches presented (localisation of the user when at a destination ONLY and continuous tracking of a moving user respectively) differ greatly in their results and the magnitude of those.
In the first case, where there is no lock time and the user is considered to be unmoving when getting his position, the accuracy of each interface is the determinant factor for the evolution and magnitude of the mismatch probability (table **??**).
In the case of continuous tracking of the user the speed of the user as well as the existence or not of a lock time is of key importance to the evolution of mismatch probability. An overview of this situation can be seen in graph(graph 8.4))

In the second examined case, "Wandering User", it can be argued that the context of the user (in terms of network delay) greatly affects the mismatch probability. Given a large enough difference on the magnitudes of introduced delay to the user, the mismatch probability almost triples over the same setup.
It can also be seen that other parameters, not belonging exclusively to the network domain, like the speed of the user or the estimation request frequency behave entirely different under different context and different mapping.

# 9 Conclusion

By the end of this section it should be possible to provide an answer for the initial question set at the end of the introductory chapter. That is:

**What is the impact of varying network performance, in the course of a day, on the reliability of the user's position estimate and how does this in turns affects a location-based service selection?**
**Additionally, how this network performance, if perceived as context, affects the reliability of the user's location estimate and how is that reflected in the performance of service selection tasks (correct selection of service)**

How this context is perceived and in turn utilised to evaluate the trustworthiness of the position estimate is conditional. This is why in this thesis 4 different scenarios, belonging to two distinct cases were run in order to verify the role and impact of the user's and the network's context on specific operations, such as location estimation. In order for this to be realised a major assumption, driven by technical limitations, was made. That there are many services scattered around the mapped area but none of them is grouped. Every and each one of them is a distinct service. Since location reliability still is a major issue in order to differentiate between all the different services existing on the map, the case was investigated through simulation.
The reliability of a position estimate however relies on at least two parameters, the ones investigated in this thesis at least, and those are:

- how quickly the estimate is received.
- how accurate this estimate is, assuming a specific technology is used to calculate it.

These two parameters were used to answer the initial question as best possible. In those two approaches the context of the user is translated into delay and position shift accordingly.

## 9.1 Conclusions on the "Wandering User" scenario

In regard to the extracted conclusions from this first scenario in relation to the specific question asked at the beginning of the report. In this first scenario the context of the user (in terms of network load) was translated into a possible delay that the user experiences. In the two separate cases the delay ranges in two extreme margins. It can be argued that the speed of the user has a smaller effect on the reliability of the estimated location than delays does. If the delay is large enough no matter how slow the user is going there will always be a high chance for location mismatch and therefore wrong service selection.

### 9.1.1 Discussion on the "Wandering User" scenario

For the first case and according to produced results of the simulation one can see that there is a great gap between the service selection performance of the two schemes. Where the first case produces an acceptable mismatch probability for the mapped delay even for the higher speeds, in the second case the mapped delay has such a heavy impact that even the slowest speed is not enough to accommodate for the produced fault. This leads to a mismatch probability that across the board is prohibiting as for the user to trust the estimated position. This situation is itself an argument against a centralised system where the server takes care of all the transactions, from position estimation to billing the customer for the respective service. If a position estimation has such a high chance of mismatch even with a very frequent request rate, then the final decision for the trustworthiness of the estimate should be left to the user instead.
Even though this premise is based on a very high delay difference, this high difference highlights the importance and impact of network conditions on the mismatch probability. One might argue that even with a high delay environment, the mismatch probability can be equalised by an equally slow user to maintain a certain level of probability. This gap in measures works against that clause by showing that given a high enough delay, even the slowest speed will produce a high probability for not trusting the location estimation.

## 9.2 Conclusions on the "Localisation at Destination" scenario

For the second case and in relation to the initial question at the beginning of the report it was seen that when the context of the user is mapped to receiver-end noise it can greatly affect the reliability of a location estimate and in turn of a service selection. The lower mismatch probability was produced by the GPS interface as it is the most accurate, however the density of the services was so high that it was again prohibiting to deem it reliable. Since this is just a claim, a new simulation that includes a less dense environment would be the requirement in order to resolve it. If the less dense environment produces a lower mismatch probability under the same specifications then it can be argued that service density plays an important role in the magnitude of the mismatch probability.
The issue here is that it is hard to find an equilibrium solution for the trade-off accuracy vs. time-to-first-fix. It is entirely dependent on the environment the user is in. What can be said with certainty is that it is extremely hard for all three interface to get a reliable lock on an already moving user. In order to remedy this in real-life situations during the setup phase while the used interface is acquiring a lock on the user, the user is mostly keeping still or moving very slowly; in other cases the estimated position might differ from the true position of the user.

### 9.2.1 Discussion on the "Localisation at Destination" scenario

For the second case were multiple accuracies are introduced on the position estimate one can see the possible effect of receiver-end noise to the trustworthiness of the position as to make a correct service selection based on that estimate. In this case where multiple interfaces are used, someone could easily extrapolate the results to financial issues. Obviously the GPS interface is the dominating solution in the world over the last decade and it does work amazingly well for outdoor scenarios given certain conditions. However there are places where GPS is not functional or has reduces performance, for example indoor localisation is based on WLAN, in urban localisation WLAN also gain formidable ground or maybe in places where GPS cannot get a lock e.g. in an urban environment. However it could be very costly to implement a WLAN network for localisation purposes outdoors, mainly due to its size that is affected by the limited range of access points in comparison to other existing interfaces. Additionally, to get accurate measurements (fingerprinting) for the whole layout of an area and the signal attenuation in it a large amount of time is required. In urban scenarios the Wi-Fi hotspots of freely available are utilised and they are a lot, in cellular networks, pre-existing infrastructure is used to calculate the position of a user and the GPS utilises satellites that are already there (and continuously upgraded). Point being here that there is not an interface that can cover every possible terrain, outdoors or indoors due to possibly prohibiting developmental costs. There are many to cover this multitude.

## 9.3 Future Work

Even if this thesis is concluded, there are certain aspects of it that can be improved and further developed. These include the presented delimitations chapter as well as improvements to the already implemented methods. These abstractions can comprise subjects of future work in order to make the system more accurate, and consequently obtaining even more meaningful results from it. Going into more details about the future work, it can be divided into the following cases, which will be explained in detail later.

- Network performance
- Global connectivity
- Improvement of the simulation
- Additional setups

Even if the network performance is based on a previously made work that tracked user behaviour and has a rational basis, the performance map that is used in this simulation is not as sophisticated. This is due to the fact that the network performance is mapped from pixel values. One important aspect is to find a way to more accurately represent the network performance in the simulation. The target is a more detailed mapping in both values and distribution of those .
Adding to that, in the delimitations chapter connectivity is delimited and there is no differentiation of service provider. It is plainly assumed that the user is connected at all

times. To provide a more realistic view on this matter, data about the coverage and the network load could be used by Danish carriers.

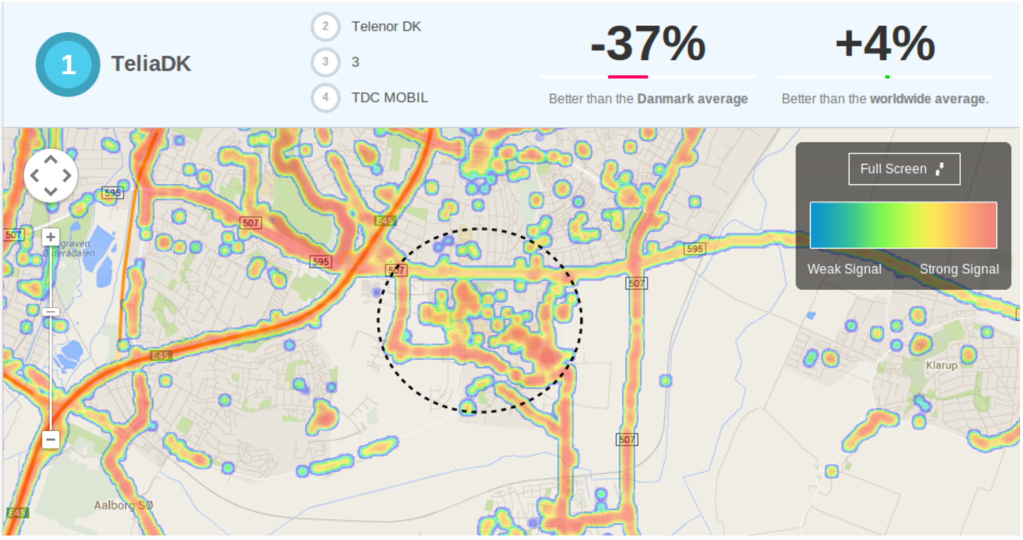


**Figure 9.1:** Potential base for a more accurate performance map according to real-life data (source: www.opensignal.com/coverage-maps/Denmark)

Another area that allows for improvement is the simulation as a whole. For one, the user-side application. At the current level the application has ultimate knowledge of the environment the user is in and provides no prediction for the user's movements (the movement is always constraint by the road system). This improvement would provide a more pragmatic set of results.
Another characteristic of the currently implemented simulation that affects the results and could be further improved is the movement of the user. The movement of the user in the implemented simulation is fairly random. Using behavioural data a predefined set of destinations or routes that adhere more to life-like could be implemented to enhance the results. Adding to that the internal way that the user is deciding for which direction to follow could also be improved, however that would require in-depth knowledge of the whole simulator.
A network aspect of the simulation that is not taken into account is the dynamic nature of network performance and how that can be heavily influenced by the user. In this simulation the context over a mapped area was considered to be static over a fixed period of time. The realistic case would be that user activity affects the performance of the network (for example, very frequent estimate requests flood the network and worsen the performance). By getting closer to this behaviour, making the context of the user dynamic a more realistic view of network can be observed and how that affects the service selection tasks.

A further improvement of the simulation could be the addition of more setups that are diverse as to explore as many as possible of the potential cases that can rise in an environment. This way a more accurate image can be acquired about how different context affects the user in each individual case.

# Bibliography


Cisco Systems Inc. (2014). The evolving context-aware ecosystem: Location, proximity and privacy. `http://newsroom.cisco.com/feature/1157689/The-Evolving-Context-Aware-Ecosystem-Location-Proximity-Privacy`. [Online; accessed 7-March-2013].

Cuddy, S., Katchabaw, M., and Lutfiyya, H. (2005). Context-aware service selection based on dynamic and static service attributes. In *Wireless And Mobile Computing, Networking And Communications, 2005. (WiMob'2005), IEEE International Conference on*, volume 4, pages 13–20 Vol. 4.

GSM Arena (2014). Gps (global positioning system). `http://www.gsmarena.com/glossary.php3?term=gps`. [Online; accessed 31-May-2014].

Martin, M. and Nurmi, P. (2006). A generic large scale simulator for ubiquitous computing. In *Mobile and Ubiquitous Systems: Networking Services, 2006 Third Annual International Conference on*, pages 1–3.

NEC Corporation (2014). Siafu: An open source context simulator. `http://http://siafusimulator.org/`. [Online; accessed 7-March-2013].

Nielsen, J. J. (2013). Fundamentals of localisation in wireless networks. University Lecture - Networks and Ambient Intelligence.

SkyHook Wireless (2014). Skyhook wireless. `http://www.skyhookwireless.com/`. [Online; accessed 31-May-2014].

Steffen R. Christensen, Martin S. Dam, L. M. M. M. S. P. (2012). Mismatch probability modeling of central decision process with multiple information providers and empirical network models. Technical report, Aalborg University. Local copy; accessed 5-April-2014.

Tanasescu, V., Roman, D., and Domingue, J. (2009). Service selection via extreme geotagging. In *Advanced Geographic Information Systems Web Services, 2009. GEOWS '09. International Conference on*, pages 189–194.

Wikipedia (2014a). Context-aware networks. `http://en.wikipedia.org/wiki/Context_aware_network`. [Online; accessed 7-March-2014].

Wikipedia (2014b). Enhanced observed time difference. `http://en.wikipedia.org/wiki/E-OTD`. [Online; accessed 29-May-2014].

Wikipedia (2014c). Intelligent transporation systems. `http://en.wikipedia.org/wiki/Intelligent_transportation_system`. [Online; accessed 7-March-2014].

Wikipedia (2014d). Wi-fi positioning system. http://en.wikipedia.org/wiki/Wi-Fi_positioning_system. [Online; accessed 29-May-2014].

# List of Figures

# List of Tables

# A Used acronyms

**ITS** Intelligent Transportation System

**GPS** Global Positioning System

**WLAN** Wireless Local Area Network

**QoS** Quality of Service

**QoE** Quality of Experience

**CAN** Context-Aware Network

**mmPr** Mismatch Probability

**RTT** Round-Trip Time

**PoCS** Probability of Correct Service

**E-OTD** Enhanced Observed Time Difference

**TTFF** Time To First Fix

**WPS** Wi-Fi Positioning System

**RSS** Received Signal Strength