Performance Analysis of Vertical Handover Algorithms

Supervisor: Dr. Amitabha Chakrabarty

Members: 1. Brian D' Rozario [06201002]

2. Shaikh Mohammad Farabi [08301006]

__Declaration__

We hereby declare that this thesis is based on the results we found by our work. Contents of work found by other researcher(s) are mentioned by reference. This thesis has never been previously submitted for any degree neither in whole nor in part.

Signature of Supervisor: Signature of Authors:

_____   _____

Dr. Amitabha Chakrabarty   Brian D' Rozario

[Student Id.: 06201002]

_____

Shaikh Mohammad Farabi

[Student Id.: 08301006]

Acknowledgement

Abstract

The world that we live in today comprises of many different technologies, each with its special characteristics. We now live in a world where technology is expanding rapidly and every time it is being made even better or something new is being gifted to us. The world is now converging to become "one" under the influence of new networking technologies. Most of our electronic devices that we use now-a-days have networking capabilities. Some devices have the ability to communicate with various types of networks at the same time, e.g. GSM, WLAN, 3G, 4G, WiMAX, etc.

Quite often we use a particular network for a specific task and hardly do we use both networks in parallel as this can consume more power from the source and drain the batteries sooner than expected. Hence we resort to means by which we may switch networks as required. This switching process is known as **handover.**

Table of Contents

List of Figures

List of Tables

Chapter 1

Introduction

In this chapter, we shall describe what horizontal and vertical handovers are, the goal of this research, the research methods used and the scope of implementation.

1.1: Brief Description of Horizontal and Vertical Handover Systems

Basically, there are two types of handover technologies that are described below

Horizontal handover is the scenario when a device is switching between similar types of networks, e.g. one GSM cell to another GSM cell or one WLAN access point to another WLAN access point. Figure 1 depicts a horizontal handover.
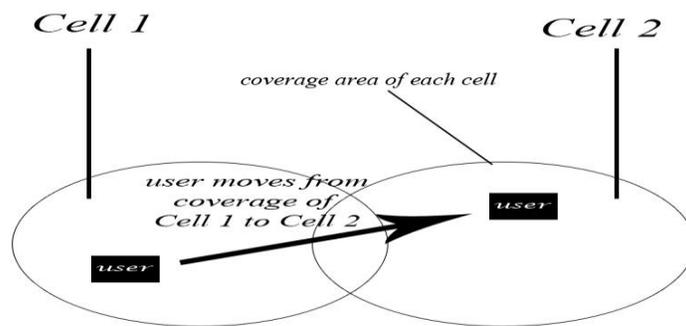
Figure 1. **Horizontal Handover**

Vertical handover on the other hand is the scenario where a device is switching between different types of available networks [1,2], e.g. GSM to WLAN. Figure 2 depicts a vertical handover.
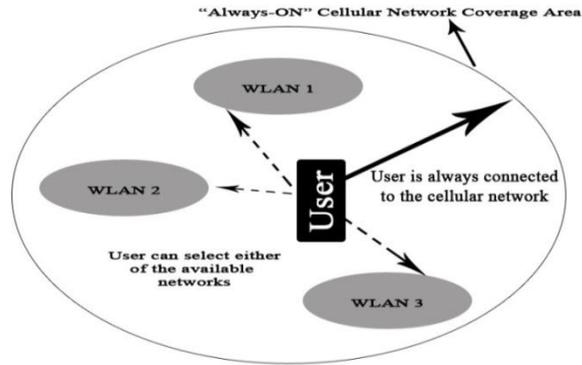
Figure 2. **Vertical Handover**

The handover process is simply a switching process, but what causes this switching? Due to availability of different types of networks and signals around us, we are being offered different types of facilities for using each type of network. For instance, we avail the GSM technology to have voice conversations where as we resort to WLAN for performing tasks that require huge volumes of data, e.g. browsing the Internet, video conferencing, audio streaming, uploading pictures, etc. All these could be done using the GSM network too, but we opted for WLAN maybe because it was offering us with free connectivity and greater bandwidth as opposed to the GSM network, which would otherwise apply charges for availing those services. The handover process takes place on the basis of some policies. The policies may be **cost-function** [1,2,5,6] based or simply deciding which network is providing greater bandwidth or better signal strength, etc.

The cost-function [5,6] is a mathematical equation that takes account some variables, e.g. available bandwidth, signal strength (RSSI), power consumption, packets lost. All these variables are assigned a value and calculated for each type of network available to the device. The network interface that gets the better cost-function value is engaged and the other interface is dropped. The process continues until the device user decides to stop.

1.2: Goal of the Research

The main purpose of this research is to study some of the available algorithms for **vertical handover.** Through this research, we can get an overview of how an algorithm performs under certain working or operating environment(s). This research will focus on the **cost function** based vertical handover algorithms [5,6] and also on vertical handover systems algorithms that take only certain environment variable into account for making a vertical handover between available networks, for example: receive signal strength (RSSI), available bandwidth, signal-to-noise ratio (SNR).

This research mainly focuses on the performance analysis of some of the available vertical handover algorithms. By performance analysis, we mean to **simulate** some of the available vertical handover algorithms under certain environment conditions and comprehend the effect of the environment on the type of algorithm used. Most of the time, the user(s) of portable devices having access to various types of network, are on the move which means the devices will certainly encounter changes in the RSSI as the user is either moving away or getting closer to a base station or stationary. Also, the user may experience changes in bandwidth while on the move or being stationary. This is because, the user might have entered a coverage zone where there are too many users of the same type of network, thereby allocating a smaller share of bandwidth to the newly added user under the coverage area that particular access point (AP).

1.3: Research Methods

The research was conducted on some selected cost-function based vertical handover algorithms. The selection was based on the ability and availability of the hardware and software available to us since some of the proposed algorithms were implemented in real-time systems using complex hardware and software systems.

The cost-function based vertical handover algorithms were implemented using the Java programming language. The environment for the simulation was created by using the various variables used for a specific algorithm, e.g. bandwidth, RSSI, device power consumption, etc.

The selected algorithms were simulated under varying environments over a time period of 5 minutes, 15 minutes and 30 minutes. The timer was implemented in Java programming as well.

The reference values and thresholds used for the variables / cost functions were scaled to resemble the values used by the selected research papers. Next, the obtained values were extracted and a graph was plotted.

1.4: Scope of Implementation

Numerous works have been done in the field of vertical handover systems, where a user is seamlessly switched into another network while on the move or on the basis of availability of options. For instance, in public places, such as a subway train station, a user may require to access the Internet, so the data volume is being transacted through the WLAN access point available for use, since it has a greater data volume capacity. Now considering the user has got aboard a train and has started to move, eventually he/she is going to lose connection to the already connected access point. But the connectivity cannot be stopped. Instead, ways have be sought to seamlessly switch the user to the available cellular network (which is an "Always-ON" network) and the connectivity for the task being performed by the user is not hampered. There are many ways by which this can be achieved and few are: cost function based policies to do a vertical handover, implementation of Fuzzy Logic [9] to do a vertical handover, comparing factors such as bandwidth or receive signal strength of cellular network with another available network and then deciding on which network to use.

Through our research, we shall try to observe how cost function based policies perform in doing a vertical handover and at the same time use some of the factors which are used in the calculation of a cost function(s) to do vertical handovers. With this, research we, attempt to analyze the performance of some of the vertical handover algorithms.

1.5: Outline of the Report

This report is divided into **three** chapters. Chapter 1 discusses the fundamentals of vertical and horizontal handover along with the goal of this research, the scope of implementation and the research methods. Chapter 2 consists of the descriptions of the selected vertical handover algorithms and the simulation results for various environment conditions (i.e. varying levels of bandwidth, RSSI or usage cost). This chapter also contains the **assumptions** made for the simulation and analysis of each selected algorithm. Performance analysis of the selected algorithms is at the end of this chapter. Chapter 3 concludes this research and also the future prospect of this research in indicated at the end.

Chapter 2

Vertical Handover Algorithms

In this chapter, we shall perform the simulation of three different types of vertical handover algorithms: Cost function policy-based vertical handover algorithm, Decision-based vertical handover algorithm and User-centric cost function policy-based vertical handover algorithm. Each of these algorithms will be simulated over a period of 5 (five) minutes, 15 (fifteen) minutes and 30 (thirty) minutes or 300 (three hundred) seconds, 900 (nine hundred) seconds and 1800 (eighteen hundred) seconds respectively, under the environments as mentioned in the respective researches. It is worth noticing that the research on these various algorithms were done in real-time hardware and software systems, hence our simulation methods needed some considerations due to lack of such powerful hardware and/or software. So, we had to omit some factors that were included in the original research.

2.1 - 1: Automated Cost Function Policy-Based Vertical Handover

As mentioned previously, this research shall focus on the work of certain algorithms due to unavailability of hi-tech hardware and software environments, since most of the algorithms were implemented in real-time. Software simulation of such complex systems are beyond our scope or research, hence we shall work with the algorithms that were feasible to simulate on the hardware and software available to us.

In the algorithm proposed in [6], the decision to perform a vertical handover was based on the "policy" set by a cost function based algorithm. The cost function took into account the factors such as the bandwidth available to the user at each type of network interface, the receive signal strength (RSSI) available to each type of network interface and the average power consumption of a device while using a particular network interface on the same device.

The cost function in [6] took into account the weighted functions of bandwidth, power consumption and RSSI , and they are $N(B) = \dfrac{W_B * (B)}{\max(B_1...B_n)}$, $N(\dfrac{1}{P}) = \dfrac{W_P * (\frac{1}{P})}{\max(\frac{1}{P_1}...\frac{1}{P_n})}$, $N(RSS) = \dfrac{W_{RSS} * (RSS)}{\max(Rss_1...RSS_n)}$ respectively.

The weighted functions $N(B)$, $N(\frac{1}{P})$, $N(RSS)$, mentioned in [6] are network dependent and are allocated to each connected device independently by the access point or base station. The algorithm proposed in [6] is a vertical handover algorithm based on cost function policy and works in the environment of heterogeneous networks.

Vertical handover decision cost function [6] is a measurement of the benefit obtained by handing over to a particular network. It is evaluated for each network *n* that covers the service area of a user. It is a sum of weighted functions of specific parameters. The general form of the cost function f_n of wireless network *n* is given by:

$$f_n = \sum_s \sum_i w_{s,i} \cdot p^n_{s,i}$$

Since each of the factors has different units, [6] used the following normalized cost function,

$$f_n = W_B * N(B) + W_P * N(\frac{1}{P}) + W_{RSS} * N(RSS).$$

The policy of [6] is such that the f_n value for whichever available network turns out to be greater, the communication process is switched to that particular network.

Assumptions: Although, [6] took into account some of the vital factors that seem relevant to calculate the cost function for an available network, it has been assumed that the mobile device is always within the coverage of an "Always-ON" cellular network. This has been assumed that the device is always connected to a particular network even if other types of networks are unavailable. Also, the weighted function for the device's power consumption, W_P, is assumed to be device independent since each device has a different power consumption rating and battery life under operating conditions. The values of the weighted functions lie within the range of 0

and 1 or as $0 \leq W_B, W_P, W_{RSS} \leq 1$. The **monetary cost** of using a network has not been taken into account. If it were considered, the vertical handover algorithm in [6] would require user interventions to input the cost for the unit usage of a certain network. The proposed cost function [6], is more of an **automated** nature as opposed to **user centric** vertical handover algorithms.

2.1 - 2: Simulation and Analysis of the Algorithm

The cost function policy based vertical handover algorithm in [6], was simulated in an environment within time periods of 5 (five) minutes, 15 (fifteen) minutes and 30 (thirty) minutes or 300 (three hundred) seconds, 900 (nine hundred) seconds and 1800 (eighteen hundred) seconds respectively.

The environment for the simulation was setup using the reference values from [6]. Initially, the weighted functions W_B, W_P, W_{RSS} were set as 0.3, 0.5 and 0.8 respectively. The algorithm is depicted by the flowchart illustrated by Figure 4.

The simulation of **f_n** in [6] provided the output shown in Table 1.

Time (seconds)	Number of Vertical Handovers
300	173
900	529
1800	1017

Table 1. **Output of simulation for Automated Cost Function Policy-based Vertical Handover Algorithm.**
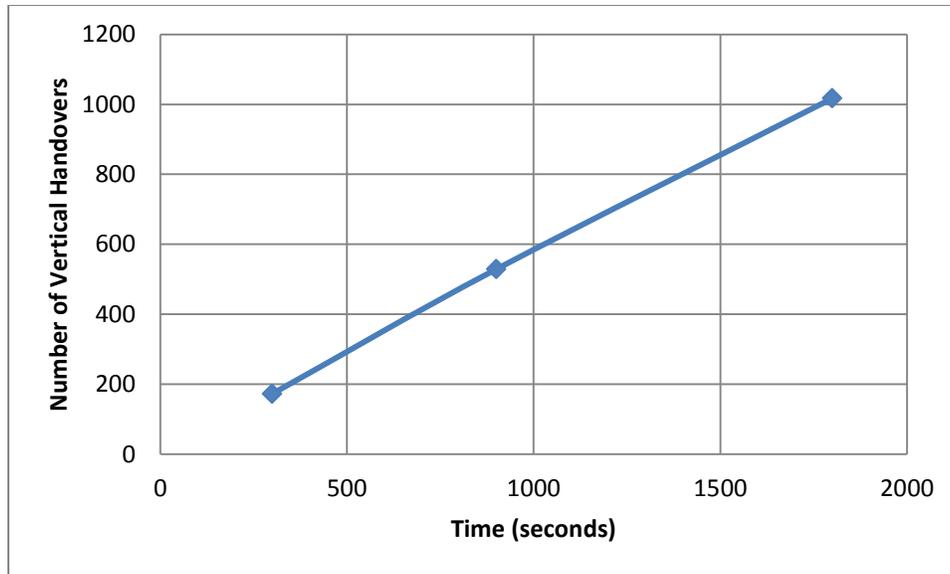
Figure 3. **Number of Vertical Handovers against Time for algorithm simulate in section 2.1 - 1**

The graph for the values obtained in Table 1 is plotted with the x-axis representing time and the y-axis representing the number of vertical handovers for [6]. It is worth noticing that the graph is linear by nature hereby we may conclude that the cost function algorithm in [6] has a linear performance.
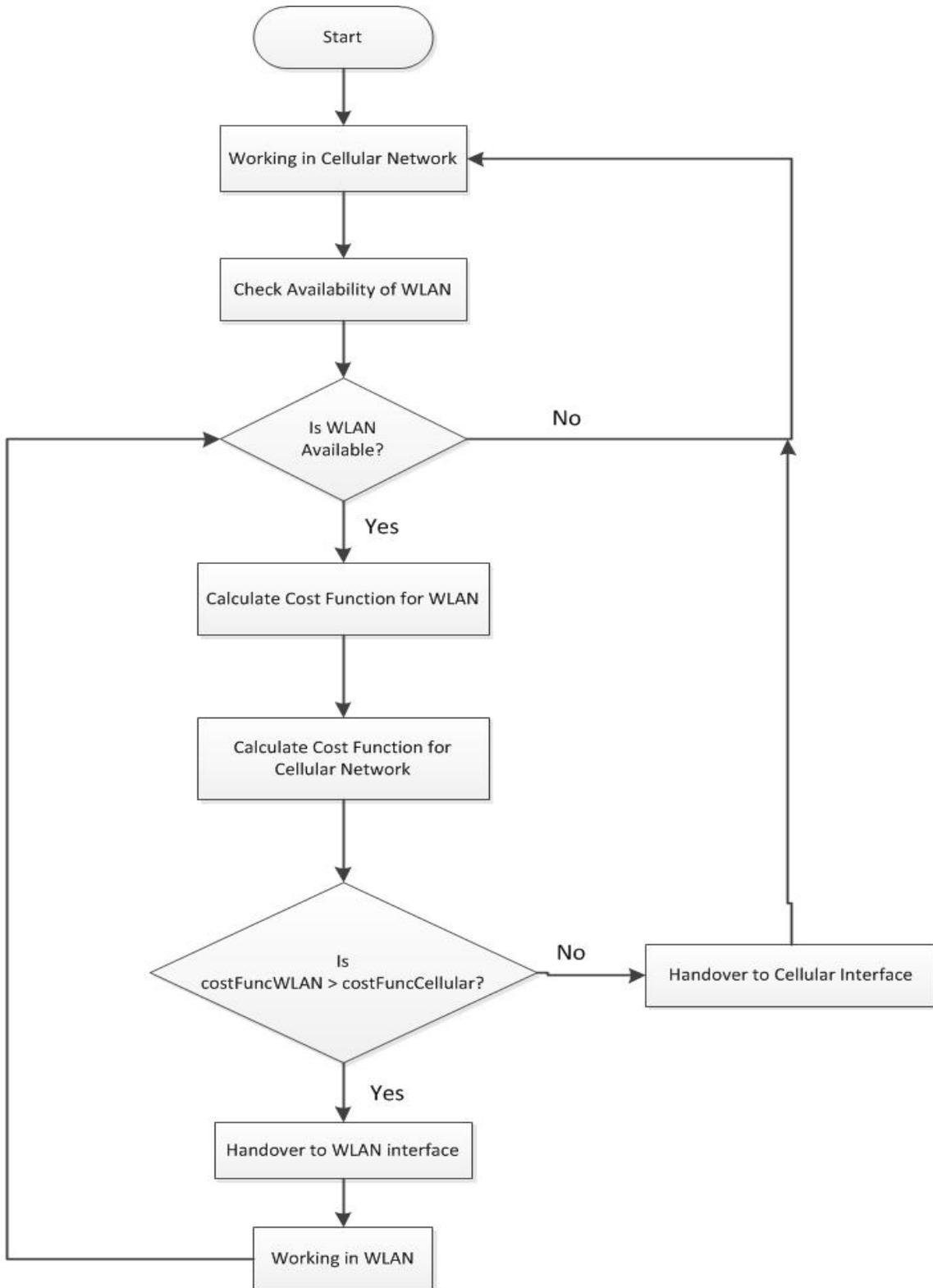
Figure 4. **Algorithm for vertical handover process for algorithm simulated in section 2.1 - 1**

The results obtained by simulating f_n in [6] in Table 1 is lacking accuracy in the measure of the number of vertical handovers due to the working environment. The research conducted in [6] was performed under environments where advanced real-time hardware and software systems were implemented. It is worth noticing that the simulated values will always provide in consistent outputs since the values of the bandwidth and RSSI are **generated randomly.** A random generation of environment variables and weighted functions typically represents that the mobile node (MN) is continuously in movement in different zones of network coverage by different types of network technology. But in reality, the values of bandwidth or RSSI do not change so **rapidly** as in the case of the simulation. For instance, the value of the receive signal strength (RSSI) of a MN may change after an hour or so even if the MN is in motion. Moreover, the number of instructions that may be processed by the machine used for the simulation is also a factor as a computer having higher computational power will be capable of giving more outputs per second of time as opposed to a slower computer.

The algorithm in [6] is an automated cost function policy based vertical handover system where the user has no option to intervene with assigning weights to the parameters of a particular network interface during the cost function calculation.

When considering algorithms that are automated by nature or take values from the available types of accessible networks, the number of handovers relies on the weights and values obtained from an access point (AP) or base station (BS). The values of the factors taken into account by [6] are obtained from the AP or BS and then the cost function is calculated hence, considering that a MN is in continuous motion where it's entering different zone of network coverage, the cost function values keep on changing continuously and therefore the number of vertical handovers change accordingly. For the simulation, a **threshold** of **5.10** was found to work best in a scenario where the factors were changing randomly over a period of time. This threshold of 5.10 was assigned for the WLAN interface of the device, i.e. while doing the vertical handover operation, if the WLAN interface's cost function value was **strictly less** than 5.10, only then the communication process would be switched to the cellular network if the communication was already in the WLAN mode. Else it would keep on working in then WLAN interface.

2.2 – 1: Decision-based Vertical Handover Algorithm using Bandwidth

Decision-based vertical handover algorithms work almost in the same manner as the cost function policy-based vertical handover systems, only with the exception of considering one factor that determines if a vertical handover is necessary. For instance, the bandwidth available to a network interface of a device may be considered as a **deciding** factor. If the bandwidth available to a certain network interface is greater than the other at a certain instance of time, the communication procedure can be switched to the interface which has greater amount of bandwidth available.

Assumptions: In algorithms that use only a certain factor as the deciding element, we do not take into account the other factors that may be affecting the usage of a particular available network. Some of the factors that are **not** considered: monetary cost of using a certain network, receive signal strength (RSSI), power consumption, signal-to-noise ratio (SNR), packet drop, latency due to switching. The only factor we put our focus on is the available bandwidth for an available network interface of the mobile node (MN).

2.2 -2: Simulation and Analysis of the Algorithm

The algorithm for vertical handover had only **one** deciding factor, i.e. bandwidth available to each network interface of a MN. Using the bandwidth as the only decision-maker we shall try to observe how the number of vertical handovers vary over the periods 5 (five) minutes, 15 (fifteen) minutes and 30 (thirty) minutes or 300 (three hundred) seconds, 900 (nine hundred) seconds and 1800 (eighteen hundred) seconds respectively.

The setup for the simulation was made to resemble real-time scenario, even though the results are not quite close to researches already performed since real-time hardware and complex software systems were not available to us. The vertical handover method takes into consideration the changes in bandwidth available to each network interface at **intervals** of 5 (five) seconds. If the bandwidth available to the WLAN interface is greater than that of the cellular interface, the ongoing communication procedure is switched to the WLAN interface if it was working in the cellular network or stays with WLAN interface if already it was working in the WLAN interface.

A **threshold** for the bandwidth available to the cellular interface is assigned at 2000 kbps (two thousand kilobits per second). If the bandwidth available to the cellular interface is greater than that of the WLAN interface, the algorithm checks if the bandwidth available for the cellular interface is greater than the 2000 kbps threshold. If so, then the communication is handed-off to the cellular interface else WLAN takes over. The vertical handover algorithm in [6] was assigning weights and considering other factors to calculate the cost function value and due to the complexity of the calculation method and our hardware limitations, we were unable to make periodical checks on the cost function values, such as that of the bandwidth checking at 5 second intervals.

We shall also compare results for the number of vertical handovers for using bandwidth as the deciding factor with the algorithm running **without** the interval checks. Figure 5 and Figure 6 shows the output of the vertical handover algorithm with and without interval checking respectively. Figure 7 depicts the algorithm of vertical handover using only available bandwidth for the decision with 5 second interval checks for available bandwidth. Table 2 shows the number of vertical handovers performed when the 5 second interval checking was enabled and Table 3 shows the outputs when 5 second interval checking was disabled as the simulation was run for 5 (five) minutes, 15 (fifteen) minutes and 30 (thirty) minutes or 300 (three hundred) seconds, 900 (nine hundred) seconds and 1800 (eighteen hundred) seconds respectively.

Time (seconds)	Number of Vertical Handovers with Interval Checking
300	3
900	7
1800	19

Table 2. **Simulation output with interval checking**

Time (seconds)	Number of vertical Handovers without Interval Checking
300	67709
900	210159
1800	417566

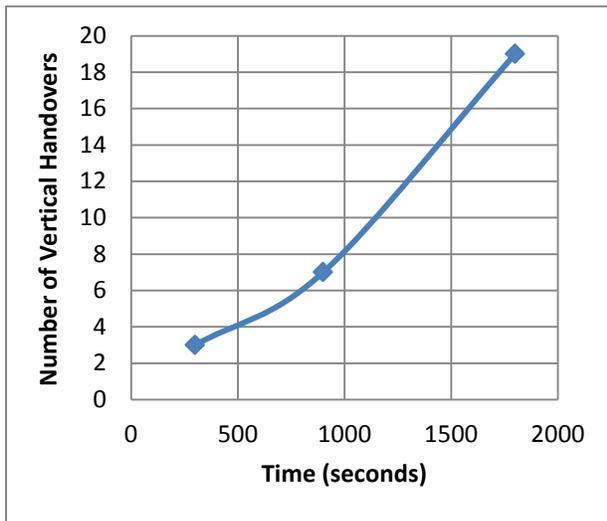Table 3.**Simulation output without interval checking**

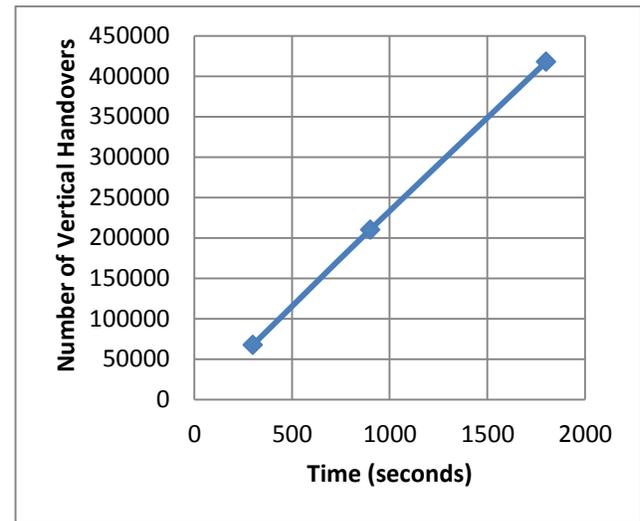Figure 5. **Graph for simulation results in Table 2**   Figure 6. **Graph for simulation results in Table 3**

From the outputs with the interval checking enabled, it is clearly visible that the number of vertical handovers can be dramatically reduced to a reasonably small amount even over a larger period of time. With the interval checking disabled, the number of vertical handovers is absurdly **large** and **unrealistic** to be used or implemented at this may lead to greater handover **latency**. Data packets have a greater chance of being **dropped** as too many vertical handovers are being performed; since the switching process takes some time, although a fraction of a second, rapid or frequent switching will increase the latency and communication will eventually be **delayed.**

The 5 second interval checking is particularly useful as it gives the device some time to evaluate the network better before switching and brings some realism to the simulation model. In the real world, allocated bandwidth doesn't vary so frequently, hence a check for bandwidth at regular intervals is more of a realistic approach to a decision-based vertical handover algorithm with available bandwidth as a deciding factor.
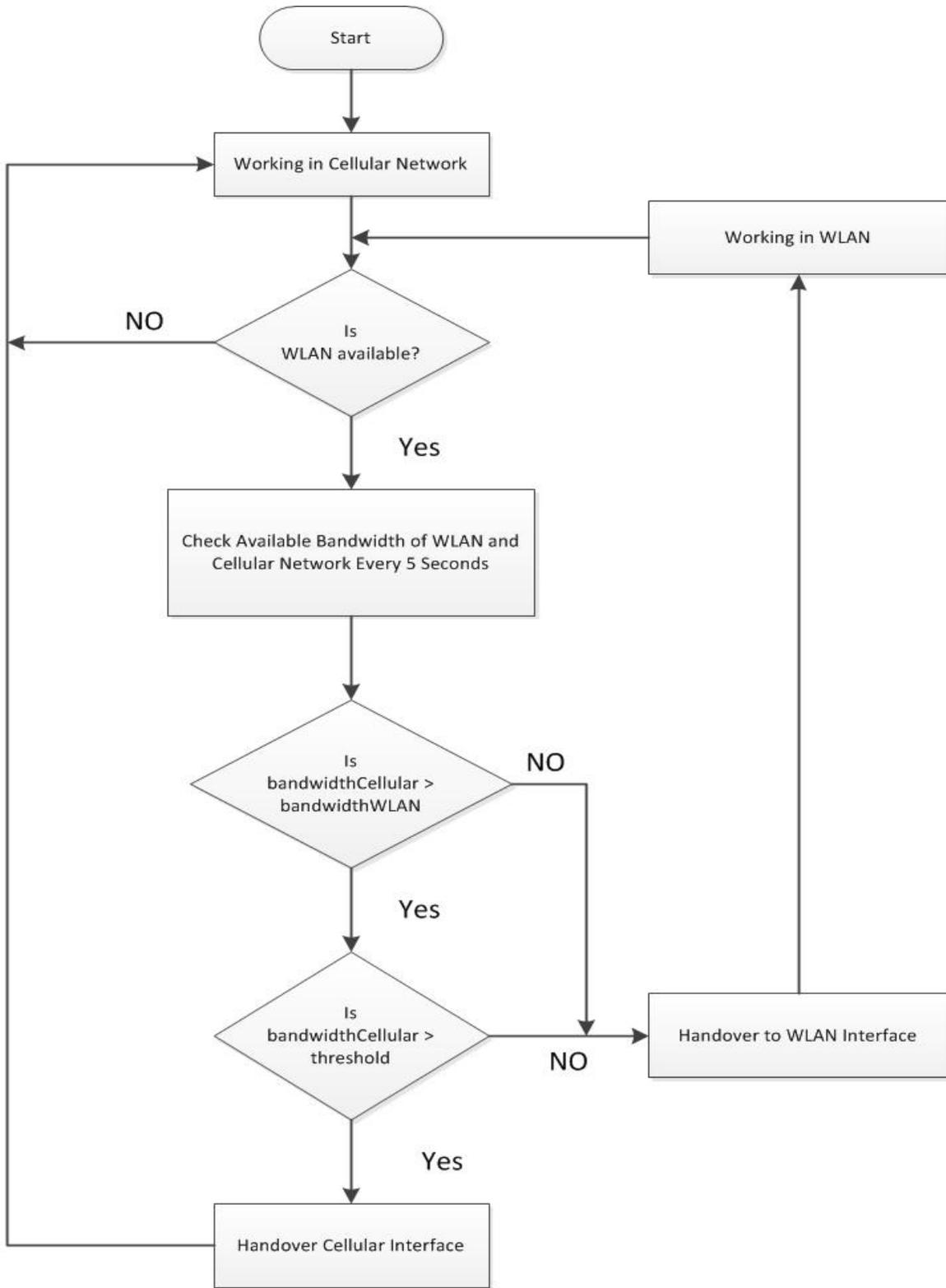
Figure 7. **Algorithm for vertical handover process simulated in section 2.2 - 1**

2.3 – 1: User Centric Cost Function Policy-Based Algorithm

There are scenarios, where the user has preferences for using or switching to a certain type of network. For instance, let use assume that the mobile node (MN) has access to WLAN and cellular networks but it can use only one network at a time. The user may need to check e-mails or simply want to stream videos from the Internet. Both WLAN and cellular networks give access to these resources but the user may decide to use the cellular network just to check mails since it will use that network for a very limited amount of time and use very little data volume as opposed to streaming videos, which consumes huge volumes of data. In the case of streaming video, WLAN may be the best choice. Cellular networks, as we know, come with data plans for which the user has to pay. WLAN on the other hand is mostly available at public places such as airports, bus stations, train stations or near a café and usually they're free to use while offering high bandwidth. Hence the preference to use a particular network arises. We shall now look into a cost function policy-based vertical handover algorithm that allows the user to interact with the system and how the system responds to the user's preference by performing vertical handovers for the given inputs. The user is given the options to assign weights to the bandwidth weighted function and the cost of using the network, W_B and W_C respectively. The bandwidth has is directly proportional to the cost function whereas the cost of using a network, W_C, is inversely proportional. The reason for this is that when an user sees high cost(s) for using a network, there is less interest to use that network since and user's main target is to gain more out of a network by paying as less as possible.

The algorithm in [5] proposes a user centric model where the user has the option to assign weights as inputs to the user-centric cost function policy-based vertical handover algorithm. The algorithm for [5] is depicted by Figure 10. There are typically two types of network operations performed at mobile node (MN), real-time operations and non-real-time operations. Real-time operations include browsing the Internet, uploading a picture to a web location, etc. Non-real-time operations include checking mails automatically at certain intervals, updating the weather widget, fetching latest news from CNN, etc. The difference between real-time and non-real-time operations is basically that in the former, the user is directly **interacting** with the MN (e.g. clicking on the "log-in" button of a webpage or telling the browser to open www.flickr.com in a new tab), while in the latter, the MN does as it has been directed (e.g. update weather widget

after every 30 minutes). The factors considered in [5] are Cost of service (C): The cost of the different services to the user is a major issue, and can sometimes be the decisive factor in the choice of a network. For different networks, there would be different charging policy, therefore, in some situation the cost of a network should be taken into consideration in making handover decisions; Security (S): For some applications, confidentiality or integrity of the transmitted data can be critical. For this reason, a network with higher security level may be chosen over another one which would provide lower level of data security; Power consumption (P): Vertically handing off to a high power consuming network is not desirable if the mobile terminal's battery is nearly exhausted or the battery's lifetime is relatively short; Network conditions (B): Available bandwidth is used to indicate network conditions and is a major factor, especially for voice and video traffic. Available bandwidth is a measure of available data communication resources expressed in Kbit/sec. It is a good indicator of the traffic conditions in the access network and Network performance (F): In some cases interference or unstable network connections might discourage a handoff decision. The weighted functions for the factors considered in [5] are W_c , W_s, W_p, W_b and the values of these weighted functions lie within the range of 0 (zero) and 1 (one) or $0 \leq W_c$, W_s, W_p, $W_b \leq 1$. Since W_c , W_s, W_p, W_b lie between 0 and 1

then,

$$W_c + W_s + W_p + W_b = 1$$

The cost function in [5] is given by the formula

$$Q_i = f\left(W_c \left(\frac{1}{C_i}\right), W_s\left(S_i\right), W_p\left(\frac{1}{P_i}\right), W_b\left(B_i\right) \right).$$

Assumptions: For simulation of this algorithm [5], we have omitted the usage of the factor of security (S) since this factor being randomly generated or set will eventually lead to a misleading set of outputs because from the perspective of our research, we do not have any in-sight about the amount of security level deployed by each type of accessible network. Also, we are assuming that the WLAN access point (AP) is either an open public network that is free for use or pre-authenticated. Hence we are omitting the security (S) factor from the selected user-centric algorithm. Therefore, as per our assumptions,

we now have:

$$Q_i = f \left(W_c \left(\frac{1}{C_i} \right), W_p \left(\frac{1}{P_i} \right), W_b (B_i) \right).$$

Therefore,

$$0 \leq W_c, W_p, W_b \leq 1.$$

And

$$W_c + W_p + W_b = 1.$$

2.3 – 2: Simulation and Analysis of the Algorithm

The algorithm in [5] was run over the period of 5 (five) minutes, 15 (fifteen) minutes and 30 (thirty) minutes or 300 (three hundred) seconds, 900 (nine hundred) seconds and 1800 (eighteen hundred) seconds respectively. The simulation was setup such that the weighted factors for cost of usage (W_c), available bandwidth (W_b) and power consumption (W_p) were varied **manually** to represent that when an user changed the parameters, how would the algorithm react to the values, i.e. how many vertical handovers would be made over the period of time (in this case 5 minutes, 15 minutes and 30 minutes).

Time (seconds)	Number of Vertical Handovers
300	69
900	167
1800	437

Table 4. **Simulation results for User-centric Cost Function Policy-based Vertical Handover Algorithm**

For the simulation results in Table 4, Table 5 shows the weighted functions used:

	Bandwidth Weight (W_b)	Cost of Usage (W_c)	Power Consumption (W_p)
WLAN	0.7	0.2	0.1
Cellular	0.4	0.5	0.1

Table 5. **User input used for simulation of algorithm in section 2.3 - 1**

Figure 8 represents the graph for the values obtained during the simulation in Table 4 for the weighted functions in Table 5.
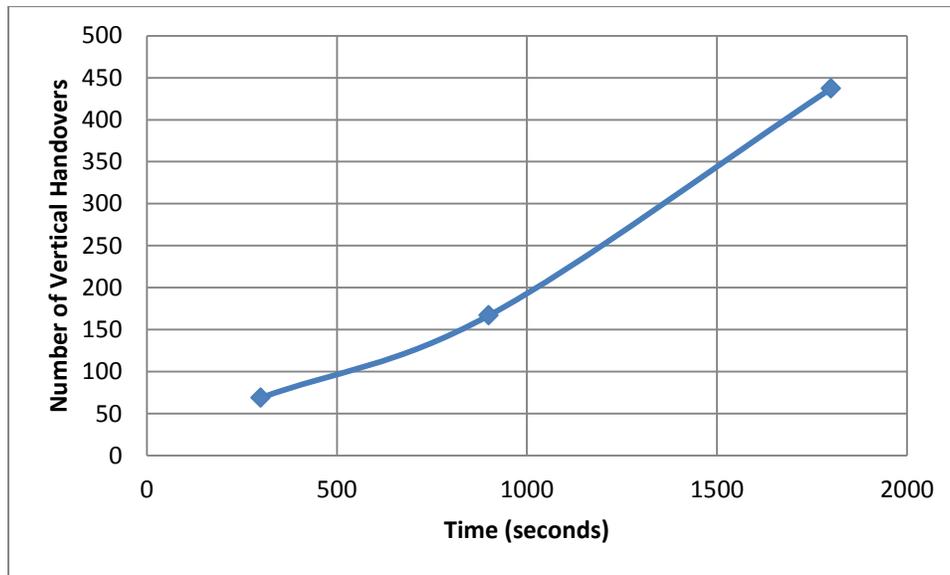

Figure 8. **Graph for simulation results in Table 4**

The simulation was run for [5] and this time with a different set of weighted functions. The output from the second simulation is show in Table 6.

Time (seconds)	Number of Vertical Handovers
300	69
900	363
1800	715

Table 6. **Simulation results for User-centric Cost Function Policy-based Vertical Handover Algorithm**

Table 7 shows the set of weighted function values that were considered for the second run of [5].

	Bandwidth Weight (W_b)	Cost of Usage (W_c)	Power Consumption (W_p)
WLAN	0.1	0.8	0.1
Cellular	0.6	0.3	0.1

Table 7. **User input used for simulation of algorithm in section 2.3 - 1**

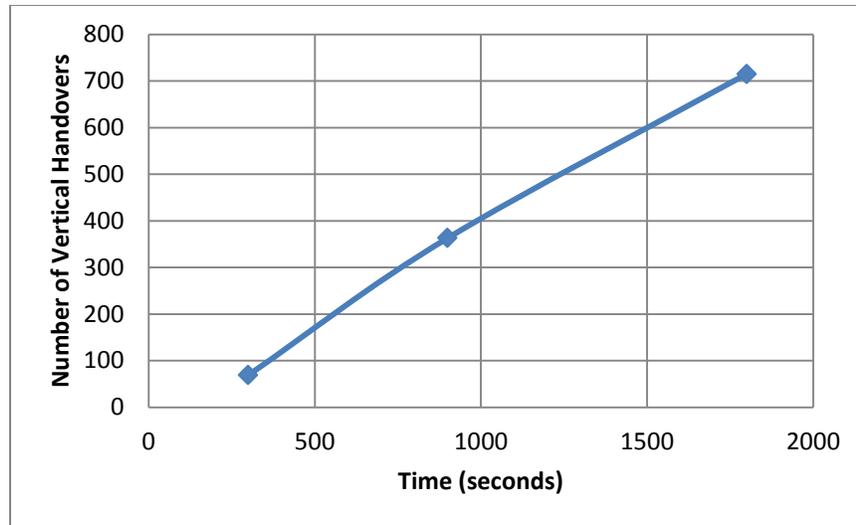Figure 9 shows the graphical output for the simulation for Table 6.

Figure 9. **Graph for simulation results in Table 6**

By comparing the outputs in Figure 8 and Figure 9 and the values for the number of vertical handovers performed, we can draw a conclusion that, it is up to the user's inputs that determine the performance of the user-centric cost function policy-based vertical handover algorithm. If more emphasis is given to the cellular weighted parameters, then the number of vertical handovers will depend on the availability of WLAN since WLAN has also been configured by the user for certain thresholds. Without the presence of WLAN, the MN will continue to operate in the cellular network and vice versa. Otherwise, vertical handover is performed on the basis of set policies.
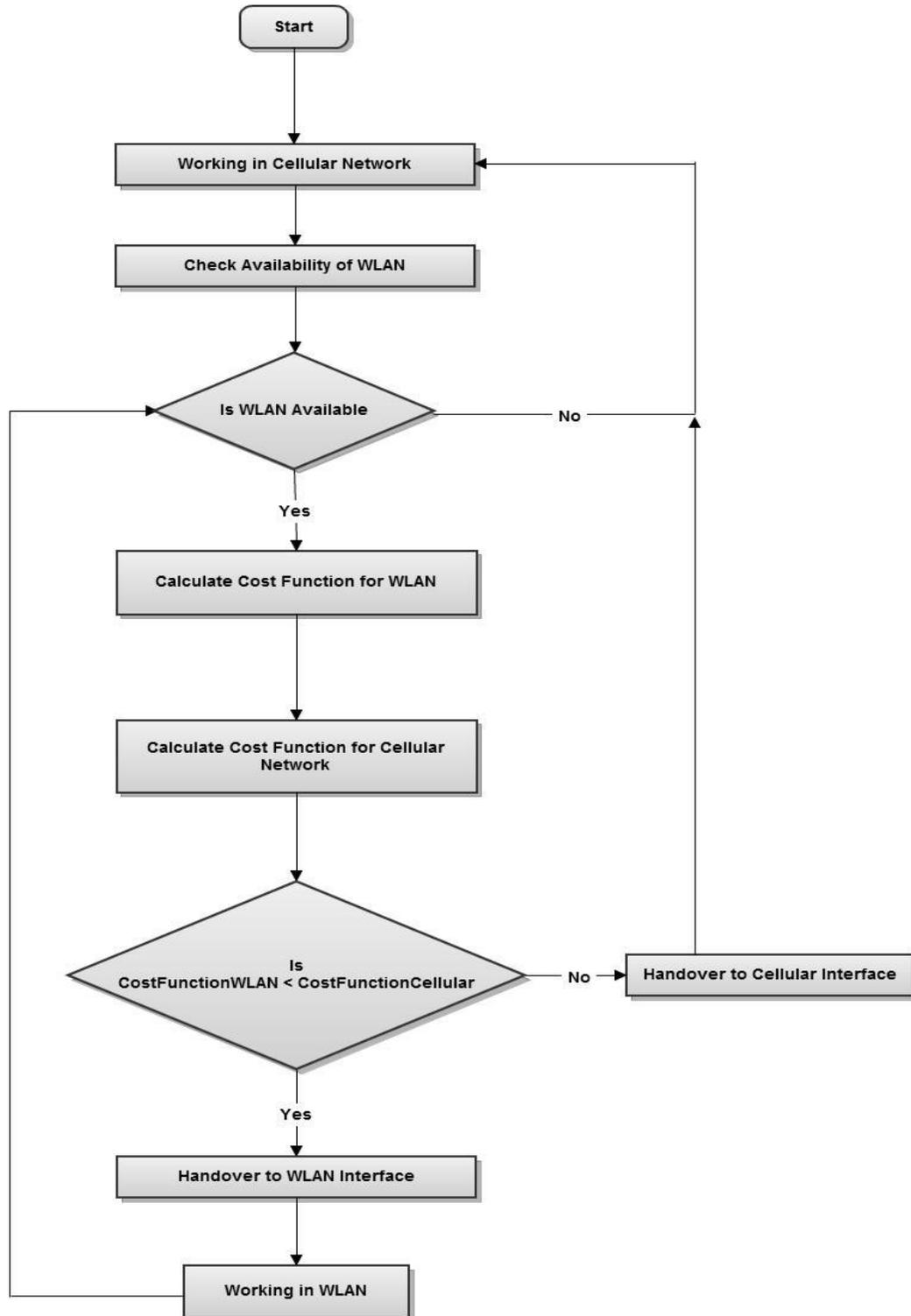
Figure 10. **Algorithm for User-centric Cost Function Policy-based Vertical Handover**

2.4: Further Simulation of Algorithms

In the previous sections of chapter 2, we have shown the results of three different types of vertical handover algorithms. In this section, the algorithms in [5] and [6] will be simulated and analyzed for the same environment as that of the decision-based algorithm in section 2.2 – 1, i.e. [5] and [6] will be simulated and the outputs for calculating the cost function at 5 (five) seconds will be observed.

Figure 11 and Figure 12 shows the output for the simulation of [5] **without** and **with** the 5 (five) second interval checking respectively.

Figure 13 and Figure 14 shows the output for the simulation of [6] **without** and **with** the 5 (five) second interval checking respectively.
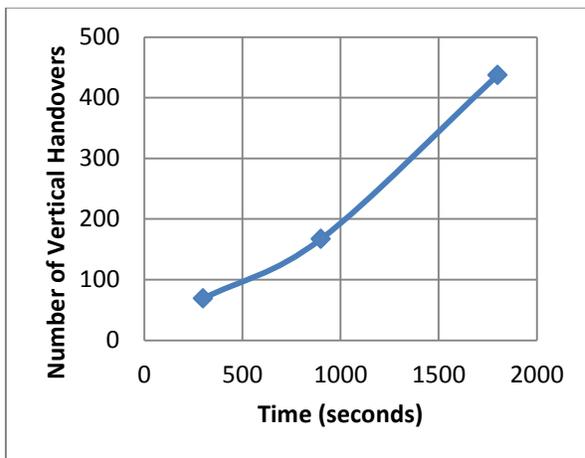

Figure 11. Output for User-centric cost function policy-based algorithm **without** interval checking
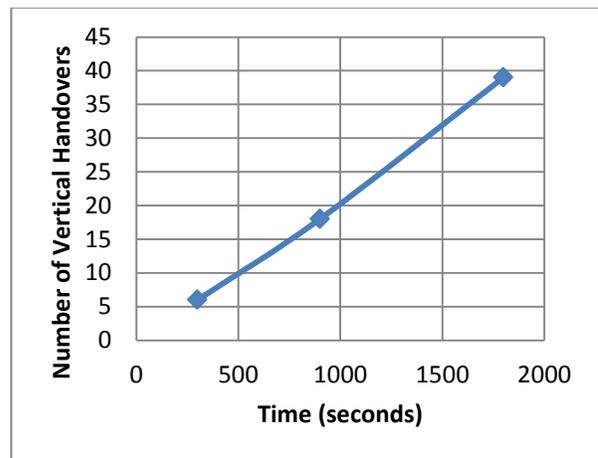
Figure 12. Output for User-centric cost function policy-based algorithm **with** interval checking
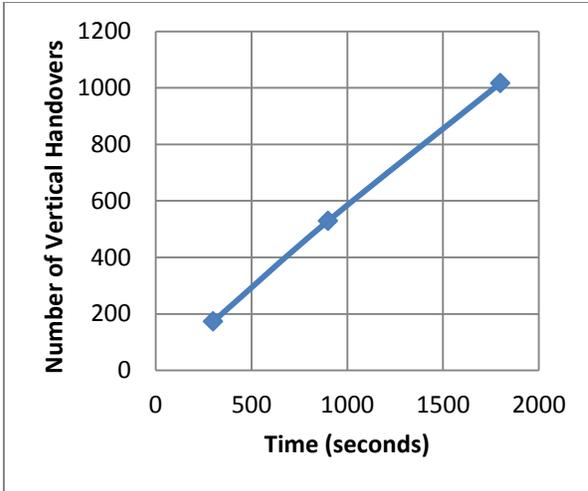
Figure 13. Output for automated cost function policy-based algorithm **without** interval checking
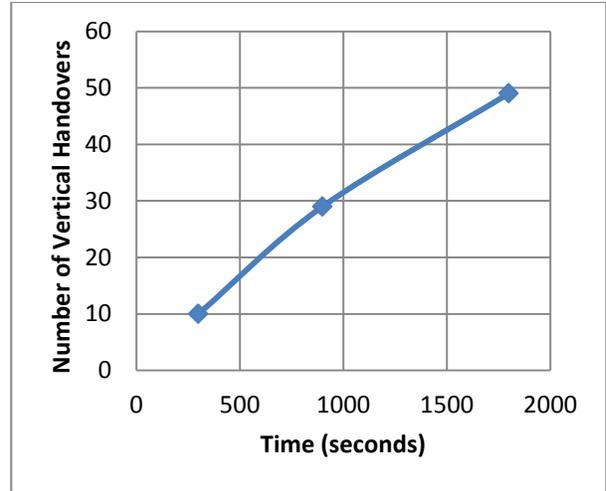
Figure 14. Output for automated cost function policy-based algorithm **with** interval checking

From the figures above, we observe that the introduction of the 5 (five) second interval checking has brought down the number of vertical handovers performed by [5] and [6]. Next we shall compare the outputs of [5] and [6] with 5 (five) second interval checking with the decision-based vertical handover using bandwidth as the **deciding** factor. We shall compare the decision-based vertical handover algorithm that had the 5 (five) second interval checking.

In figure 15, we have compared the output of the decision-based vertical handover algorithm of section 2.2 – 1 with [5], both with 5 (five) second interval checking.
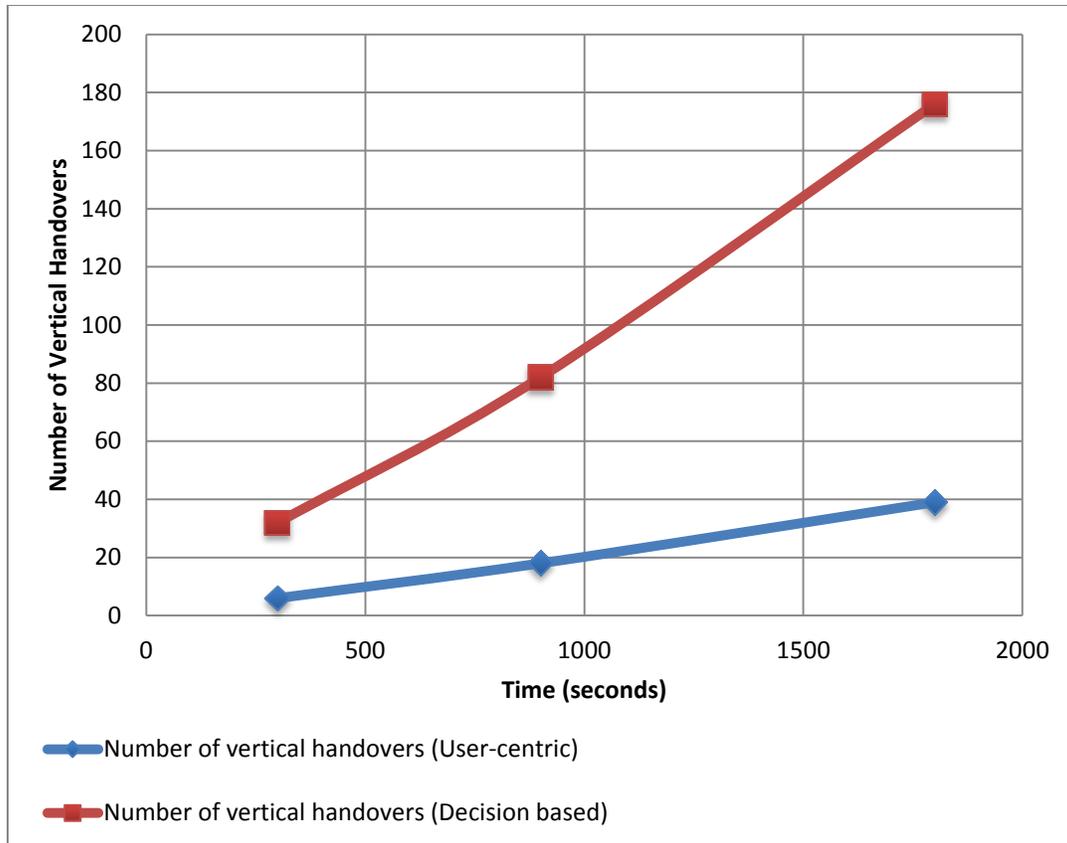
Figure 15. Performance comparison of decision-based and user-centric cost function policy-based vertical handover algorithm with 5 second interval checking

From the graph in Figure 15, we can clearly see that the user-centric vertical handover algorithm [5] outperforms the decision-based algorithm, since over the same periods of simulation, [5] makes less number of vertical handovers while the decision-based algorithm makes many vertical handover for the same.

Figure 16 will depict the performance of [6] and the decision-based vertical handover algorithms with the 5 (five) second interval checking.
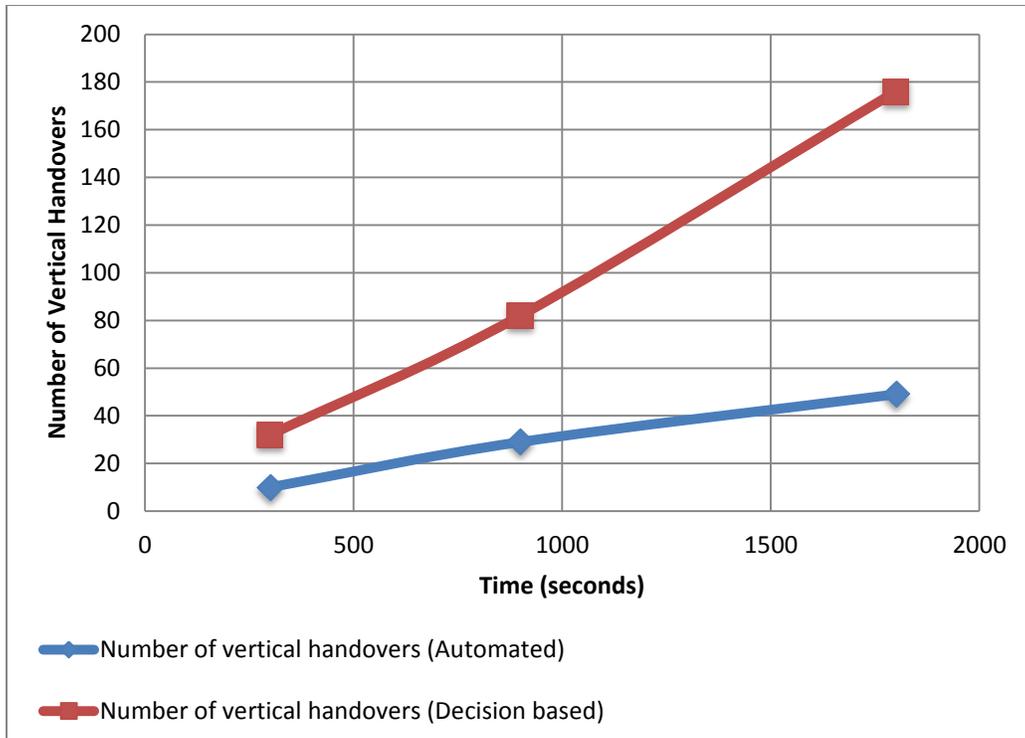
Figure 16. Performance comparison of decision-based and automated cost function policy-based vertical handover algorithm with 5 second interval checking

From Figure 16, it is evident that [6] performs better with the 5 (five) second interval checking as opposed to the decision-based vertical handover algorithm with 5 (five) second interval checking since [6] makes fewer number of vertical handover that the decision-based algorithm for the same periods of simulation.

Figure 17 shows all thee algorithms' performances with 5 (five) second interval checking.
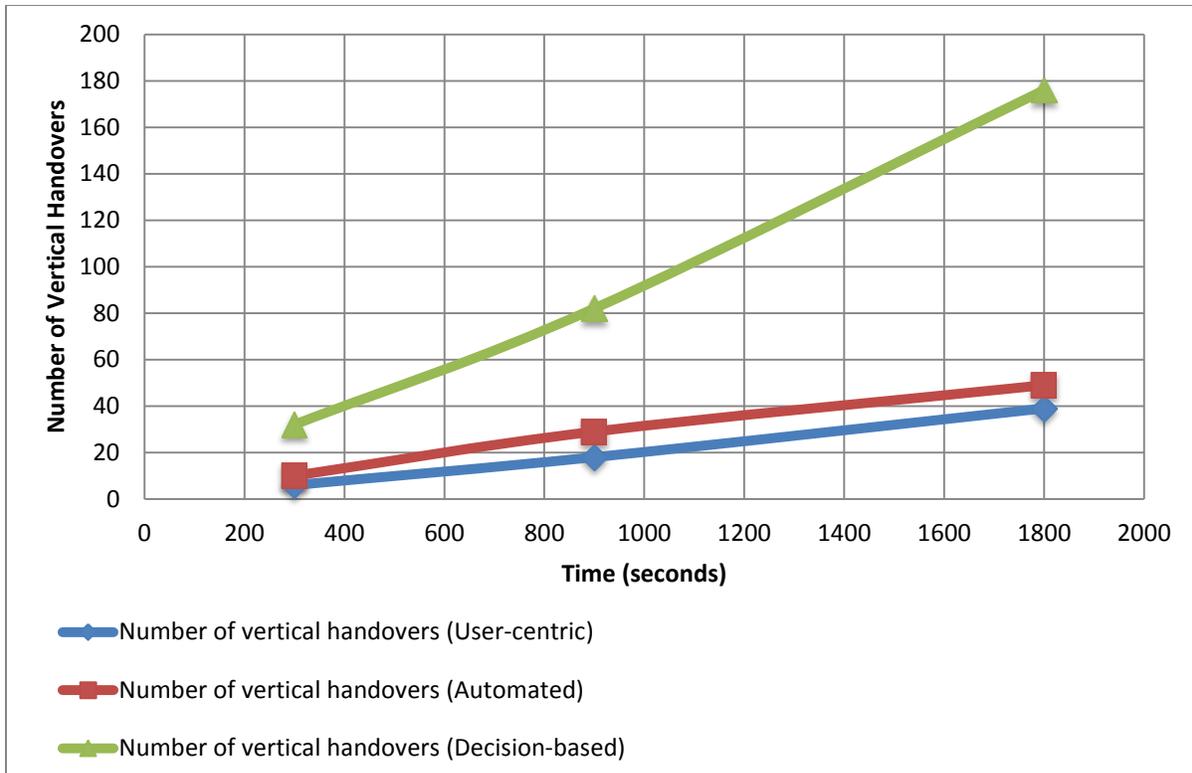
Figure 17. Performance comparison of automated and user-centric cost function policy-based vertical handover algorithms with decision-based vertical handover algorithm.

2.5: Performance Analysis

From the Figure 15 and Figure 16 we see that both the cost function policy-based vertical handover algorithms perform better that the decision-based vertical handover algorithm. This is primarily because, the cost functions take various environment factors such as bandwidth, RSSI, usage cost, power consumption, etc. into account and then decide whether to switch to a different type of available network by comparing the cost function's value against a **threshold** value.

Eventually, Figure 17 shows us that the user-centric cost function policy based vertical handover algorithm [5] performs better that the automated cost function policy based vertical handover algorithm [6]. This is because, [5] takes more environment variables into account hence the precision of the calculation for real-time operating environment is greater. Moreover, the user(s) are able to interact with the system in [5]. They can assign the **weight** (0 being the lowest and 1

being the highest) of the usage cost of a network manually and also define a weight to the bandwidth required, since a user may not always want to consume higher bandwidth for an operation.

Chapter 3

Conclusion

In conclusion, we may state that there is nothing called the "perfect" vertical handover algorithm although it may be cost function policy-based, decision-based on a single variable or Fuzzy Logic vertical handover system. Every algorithm has its advantages and disadvantages.

3.1: Concluding the Research

The main focus of this research was to **analyze** some of the available cost function policy-based vertical handover algorithms. We have found that cost function policy-based vertical handover algorithms perform better than decision-based vertical handover algorithms which take only one factor (bandwidth in this case) for performing vertical handovers.

Cost functions may be automated or user-centric (i.e. users can assign *weights* to certain factor, such as bandwidth or usage cost of a network), but their performance lies in the selection of the various environment factors. Figuring the **right** threshold is also a **determinant** of the performance since the values of the cost function(s) for each type of network is compared against a set threshold and vertical handover is performed accordingly.

.

3.2: Future Works

In this research, we have seen the performance analysis of three different variations of vertical handover algorithms. For the future we intend to work on developing an algorithm that is decision based, but considers more than one factor, e.g. quality of service (QoS), RSSI, packet delay, as being the **decision makers**. The user(s) may also interact and assign **weight** to a factor that might seem important to them during usage. Then, we can make a better performance

analysis of the other available types of vertical handover algorithms with our work and suggest which type of algorithm to use for a specific type of environment, e.g. an environment where factors change rapidly or the user is in a location where rarely the RSSI changes but the available bandwidth is changing quite often as more users are being added to an AP, etc.

References

[1]   SungHoon Seo, SeungChan Lee, and JooSeok Song. 2007. *Policy Based Intelligent Vertical Handover Algorithm in Heterogeneous Wireless Networks*. In *Proceedings of the 2007 International Conference on Convergence Information Technology* (ICCIT '07). IEEE Computer Society, Washington, DC, USA, 1900-1905.

[2]   Kok-Kiong Yap, Rob Sherwood, Masayoshi Kobayashi, Te-Yuan Huang, Michael Chan, Nikhil Handigol, Nick McKeown, and Guru Parulkar. 2010. *Blueprint for introducing innovation into wireless mobile networks*. In *Proceedings of the second ACM SIGCOMM workshop on Virtualized infrastructure systems and architectures* (VISA '10). ACM, New York, NY, USA, 25-32.

[3]   Andrea Calvagna and Giuseppe Di Modica. 2004. *A user-centric analysis of vertical handovers*. In *Proceedings of the 2nd ACM international workshop on Wireless mobile applications and services on WLAN hotspots* (WMASH '04). ACM, New York, NY, USA, 137146.

[4]   Lucian Suciu, Karine Guillouard, and Jean-Marie Bonnin. 2006. *A methodology for assessing the vertical handover algorithms in heterogeneous wireless networks*. In *Proceedings of the 2006 workshop on Broadband wireless access for ubiquitous networking* (BWAN '06). ACM, New York, NY, USA, Article 4 .

[5]   Nikhil Patel, Kiran Parmar *Quality Dependent Vertical Handover Decision Algorithm for Fourth Generation(4G) Heterogeneous Wireless Networks* , IJCEM International Journal of Computational Engineering & Management, Vol. 15 Issue 2, March 2012

[6]   Meriem Kassar, Brigitte Kervella, Guy Pujolle *An overview of vertical handover decision strategies in heterogeneous wireless networks* (2008) *Computer Communications*, 31 (10), pp. 2607-2620

[7]   Mrs.Chandralekha, Dr.Praffula Kumar Behera *Minimization of Number of Handoff Using Genetic Algorithm in Heterogeneous Wireless Networks*, International Journal of Latest Trends in Computing (E-ISSN: 2045-5364), Volume 1, Issue 2, December 2010

[8]   Aaron Carroll and Gernot Heiser. 2010. *An analysis of power consumption in a smartphone*. In Proceedings of the 2010 USENIX conference on USENIX annual technical conference (USENIXATC'10).

[9]   Hongwei Liao, Ling Tie, and Zhao Du *A Vertical Handover Decision Algorithm Based on Fuzzy Control Theory* Computer and Computational Sciences, International Multi-Symposiums on, Vol. 2 (2006), pp. 309-313