Data Upload in LTE-A Mobile Networks by Using Shared Segmented Upload

Misagh Tavanpour, Jan Mikhail, Mohammad Moallemi, Gabriel Wainer
Department of Systems and Computer Engineering, Carleton University, Ottawa, Canada
Email: {misagh, janmikhail, moallemi, gwainer}@sce.carleton.ca

Gary Boudreau, Ronald Casselman
Ericsson Canada, Ottawa, Canada
Email: {gary.boudreau, ronald.casselman}@ericsson.com

***Abstract*—There have been ongoing efforts focused on improving mobile networks standards to support the ever-increasing user demands of high data rate services. These efforts are more crucial for cell-edge users where their long distance from their serving Base Station (BS), and the higher interference from the neighbouring cells, degrades their performance. Contemporary communication standards, proposed for Fourth Generation (4G) of mobile telecommunication standards, use different techniques to deal with these bottlenecks. Long Term Evolution Advanced (LTE-A), is a promising standard for 4G mobile networks, and it uses different technologies to enhance users' performance regardless of their location in the coverage area. LTE-A employs Coordinated Multi-Point (CoMP) technique particularly to provide high data rate services for cell-edge users. In this context, we present Shared Segmented Upload (SSU), a novel method for uploading large files from User Equipment (UE) to multiple BSs in a CoMP communication scenario. We use Discrete EVent System Specification (DEVS) formalism to model an LTE-A mobile network using SSU. In addition, we employ DEVS to simulate a conventional non-cooperative algorithm to evaluate the effectiveness of SSU in two scenarios: rural and urban area settings. The simulation results show that, compared to the conventional method, SSU improves cell-edge users' uplink performance and reduces the latency for a UE to upload its data to the network.**

***Index Terms*—CoMP, LTE-Advanced, DEVS**

I. Introduction

Cellular networks have seen many changes since the advent of the first commercial mobile phone network in 1971. Many cellular networks standards were introduced for different generations of mobile communications standards to support the ever-increasing user requirements. Since 1971, the rate of adoption of mobile devices has grown exponentially. Statistics show that in 2013, the number of mobile subscriptions had reached 6.8 billion, equivalent to 95% percent of the world population [1]. In addition, mobile networks are changing the way in which Internet users choose to access the Internet. A study shows that, between May 2011 and May 2012, the proportion of global web page views from mobile devices almost doubled [2]. Consequently, service providers need to address two problems: the large number of UEs that they need to service, as well as their high data rate demands.

There are ongoing research efforts by service providers to improve their mobile networks performance to deal with these two problems. These efforts can be divided into three major categories: improving efficiency of the current resources; providing new hardware; and introducing new standards. In terms of standards, telecommunication service providers focus on increasing the network performance by proposing algorithms or techniques that are more efficient. LTE-Advanced is a mobile communication standard, introduced by 3rd Generation Partnership Project (3GPP) as a candidate for 4G mobile networks. LTE-A is a backward-compatible extension of LTE [3]. LTE-A meets or exceeds the requirements set by the International Telecommunications Union (ITU) in International Mobile Telecommunication-Advanced (IMT-Advanced), and is considered as a candidate for IMT-Advanced systems [4, 5, 6].

One of the main objectives of LTE-A networks is to provide consistent performance for the UEs regardless of their location within the coverage area. Evidently, poor network service is not acceptable in contemporary mobile networks due to users' high data rate requirements. However, providing high quality signals to UEs in all coverage areas is challenging, especially when a UE is located near the cell border. This group of users suffers from two problems: the long distance from the cell center where their serving BS is located, and the higher interference from the neighbouring cells. These problems need to be addressed to allow service providers to meet the

doi:10.4304/jnw.10.4.252-264

expectations of cell-edge users. Modern mobile network standards such as LTE-A propose different techniques to deal with such issues.

Coordinated Multi-Point (CoMP) is one of such techniques employed by LTE-A. CoMP coordinates a set of BSs to decrease the interference and increase the received desired signal power. This method increases the quality of the communication channel between the cell-edge UE and its serving BS. There are two architectures for CoMP in LTE-A: centralized and distributed. Moreover, based on the way that data and scheduling information are shared among transmission points, two CoMP schema can be considered: Coordinated Scheduling/Beamforming (CS/CB) and Joint Processing (JP).

We use a novel algorithm, called Shared Segmented Upload (SSU), for uploading large files from a UE to multiple BSs in a distributed CoMP architecture. SSU has a number of common points with the BitTorrent protocol [7]. The latter is used to speed up the download of large files on the Internet by allowing users to join a swarm of hosts to download and upload from each other, simultaneously. BitTorrent can work over networks with lower bandwidth, and it can be considered as an alternative to the single source, multiple mirror sources technique for distributing data [7].

SSU adapted this technique to improve data upload from a UE to a set of BSs. This technique can solve the bottlenecks caused, for instance, by users uploading large files from the UE to the network, improving the upload performance and quality. This technique is a subset of the Joint Processing method in CoMP. SSU uses small segments to transfer large files from a single UE to the group of BSs that are coordinated dynamically. File segments are transmitted independently and the BSs with better communication channel with the UE can receive more segments of the original data file. This process speeds up the data upload. Finally, the collected segments at different BSs of the coordination set, are gathered and organized at the serving BS, like the pieces of a puzzle.

In order to test and evaluate the performance of SSU and compare it with other method, we compared SSU and a conventional non-cooperative algorithm in two types of scenarios: urban area setting and rural area setting. We used the DEVS formalism to model both SSU and the conventional method. The hierarchal nature of modeling in DEVS allows us to study different aspects of the target problem by providing precise information from different levels of the implemented model.

We used CD++ as a platform to implement the DEVS models. Simulation results reveal that compared to the conventional method, SSU users benefit from more consistent services as their distance from the cell center increases. In addition, these results show that for a given size of data file, cell edge users using SSU required less time to upload their data.

II. BACKGROUND

LTE-A benefits from a number of technologies including Multiple Input Multiple Output (MIMO), Orthogonal Frequency Division Multiplexing (OFDM), enhanced Inter-Cell Interference Coordination (eICIC), and Coordinated Multipoint (CoMP) [8]. These techniques help service providers deliver high data rates for the users, as well as meet the IMT-Advanced requirements. In addition, they are used to overcome the transmission impairments such as Inter-Cell Interference (ICI), a major bottleneck for the performance of cellular networks [9], particularly for users located near the cell borders. ICI is the result of re-using the same radio resources in neighbouring cells in an uncoordinated manner [10], preventing mobile network standards from achieving their theoretical rates [11]. To overcome these problems, different techniques including interference cancellation, interference coordination, and interference randomization have been investigated [2, 6, 12, 13, 14].

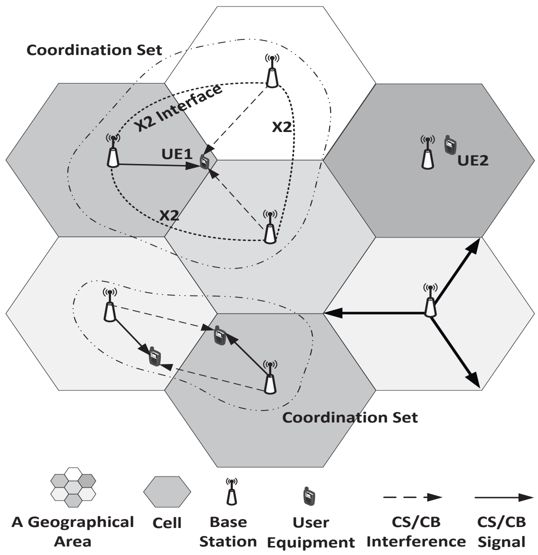

Figure 1. CS/CB transmission in LTE-Advanced

Among the various techniques that LTE-A has employed to improve user quality of experience, CoMP can be considered as a key technique to mitigate co-channel interference and increase per user capacity. CoMP refers to a set of BSs that are coordinated dynamically. BSs in a CoMP network form coordination sets whose main objective is to mitigate interference and enhance the throughput from BSs to UEs, especially for cell-edge users [15]. As shown in Fig. 1, three BSs establish a coordination set to provide better service for a user (UE1) at the edge of the cells. Compared to users in the cell center, there are two main bottlenecks for cell-edge users: lower signal strength due to the longer distance between the UE and the BS, and higher interference levels due to the close proximity to neighbouring cells. High data rates are relatively easy to maintain when the UE is close to the BS (UE2 in Fig. 1), but as distances between the UE and the BS increases, it becomes more challenging [6].

One approach to ensure high data rates regardless of the distance from the base station is to coordinate signals from multiple BSs. When a UE is at the cell edge, we can use the neighbouring BSs signals in a coordinated way, and improve transmission quality. To do so, the BSs and UEs need to exchange information to create a coordination set. This information includes scheduling, Hybrid Automatic Repeat Request feedback, Channel State Information (CSI), and control information [5]. As seen in Fig. 1, BSs are connected through 3GPP's high-speed interfaces, called X2, and they share the messages received from their UEs with other BSs in the coordination set over these interfaces [5].

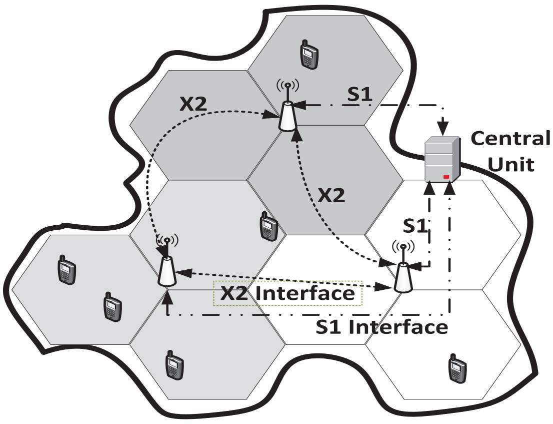

Figure 2. Centralized CoMP transmission in LTE-Advanced

Based on the way that the control information is shared among the transmission points, two CoMP implementations can be considered: centralized and distributed. In centralized CoMP, there is a central entity where all the UEs' data and channel information is available. Each serving BS forwards its UE's CSI feedback and other control information, as well as the UE's data, to this central unit through the network's low latency S1 interfaces (see Fig. 2). The central unit is responsible for performing the scheduling operations and forwarding the scheduling decisions to the BSs in the coordination set, which act based on these decisions. This architecture has a high signalling overhead on the backhaul because all the BSs must send all of the UEs' status information and data to the central unit. In addition, the central unit needs to send the scheduling results to the BSs over the backhaul. In distributed CoMP, UEs share their channel status with their serving BSs and each serving BS forwards this information to the other BSs in the coordination set. As such, all the BSs in the coordination set can perform the scheduling operations independently. The scheduling operations are performed using identical schedulers and the same set of inputs, leading to the same results at each BS. The resulting distributed CoMP implementation has a reduced infrastructure cost and less complexity [3, 4, 6, 11, 16].

There are two schemas for CoMP in LTE-Advanced based on the way in which the data and scheduling information are made available at the BSs: Coordinated scheduling/Beamforming (CS/CB) and Joint Processing (JP). In the former, each UE is only served by its serving BS that is one of the BSs in the coordination set (Fig. 1). The scheduling decisions are made to reduce interference among the BSs in the coordination set. In this method, the exchange of scheduling information is required. However, the UE data does not have to be shared among the BSs in the coordination set. The UE1 in Fig. 1 communicates with its coordination set using Coordinated Scheduling. In JP, the data to be transmitted to a single UE is transmitted from the BSs in the coordination set simultaneously (UE1 in Fig. 3). This increases the received signal strength at the UE and decreases interference levels, resulting in a higher data rate. However, in this method, the amount of data exchanged over the backhaul can be very large [3, 5, 6, 8, 11, 17] and latency of data exchange between BSs can limit the achievable data rates.

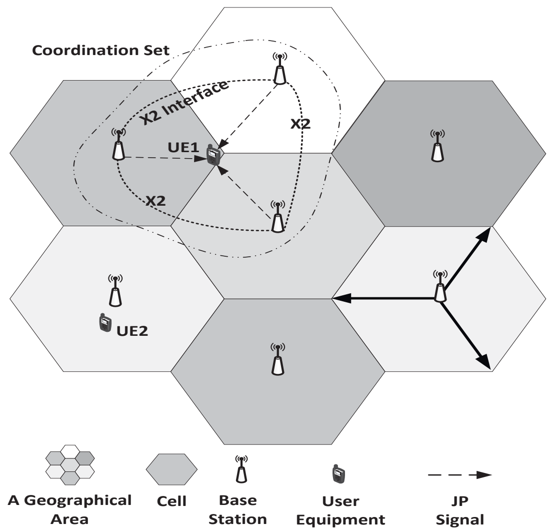

Figure 3. JP transmission in LTE-Advanced

There are numerous researchers working on Modeling and Simulation (M&S) of cellular networks, a useful method to test and evaluate new techniques. M&S can be used to study different aspects of a problem by changing the test configurations and analyzing how the model reacts. For instance, in [18], the authors used M&S for testing in LTE networks. In [19], the authors focused on discontinuous reception in the LTE networks. This approach leads to better battery power usage of UEs (with potential increase in latency). They used OPNET as their platform for M&S. In [20], the authors investigated user equipment's quality of service (QoS) requirements in the LTE uplink and proposed a QoS-aware resource allocation paradigm for the LTE uplink scheduling. The authors used NS3 to evaluate and compare the performance of the proposed approach with two other time domain paradigms. In [21], the authors studied the handover procedure in LTE networks using NS2 as the simulation tool to investigate the effects of user data forwarding on the user connections. In [22], the authors presented an OPNET simulation model to investigate the uplink performance of LTE FDD and TDD modes regarding the latency and channel utilization. The LTE-A standard supports carrier aggregation by integrating contiguous or non-contiguous carriers at the base station. In [23], NS3 was used to implement a carrier aggregation module to study scalable video multicast to LTE-A user groups.

In our case, we used the DEVS formalism [24] to model the cellular network to test and evaluate SSU and compare it to the conventional non-cooperative algorithm. DEVS theory is a methodology used to represent models, providing an abstract description of the system of interest [24]. Within DEVS, the Coupled components maintain the hierarchical structure of the system, while Atomic components represents the behavior of different parts of the system. Atomic components can be considered as the basic building blocks of the system, which are composed of I/O ports and a finite state timed automaton representing the behavior of the model [6, 24, 25]. The CD++ toolkit has been used as the framework to implement the DEVS models. This toolkit provides a built-in specification language for implementing Atomic models using C++. A Model file (MA file) is used to define the hierarchical structure of the Coupled models and to initialize the atomic model's parameters [6, 25].

III. The SSU Algorithm

Uploading large files to a mobile network can be challenging. The limited availability of bandwidth in a single communication link between a UE and its serving BS reduces the achievable data rates, particularly for cell-edge users where the reception is weak. SSU is designed to mitigate these issues by spreading the data transfer over a number of BSs that participate in a CoMP coordination set. The algorithm is based on the BitTorrent download algorithm over the web [7].

Initially, the UE creates a "*MetaInfo*" file that defines the large data file to be uploaded to its serving BS. This file is relatively smaller, and therefore, can be transferred quickly to the UE's serving BS. Table 1 shows the format of a *MetaInfo* file.

TABLE I.
MetaInfo File Format

Key	Description
length	Length of file (bytes)
name	Filename
piece size	Number of bytes in each piece
pieces	String consisting of the concatenation of all 20-byte SHA1 hash values, one per piece

The piece size is usually an exponent of two, and it is selected based on the file size. There is a trade-off between the piece size and the efficiency of SSU. A large piece size makes SSU less effective, as it becomes similar to uploading the large file using traditional techniques; on the other hand, a very small piece size will result in a very large MetaInfo file, increasing the overhead. The optimal piece size depends on a number of factors, such as the number of BSs involved in the CoMP coordination set, and the number of handovers expected to happen during the file transfer. Therefore, the piece size varies depending on the conditions of the uplink channels, which can be adjusted in different simulation scenarios to be investigated for each situation. In the BitTorrent protocol, the most common piece sizes used are 256 KB, 512 KB, and 1 MB [7]. All file pieces are of equal length, except for the final piece, which is irregular. The number of pieces is determined by dividing the total length of the file by the piece size. Each piece is identified by a SHA1 hash code generated from the data contained within that piece. These hash values are each 20 bytes long and are concatenated together to form the pieces value dictionary in the MetaInfo file.

Fig. 4 shows the steps for uploading a large file from a UE to its serving BS.

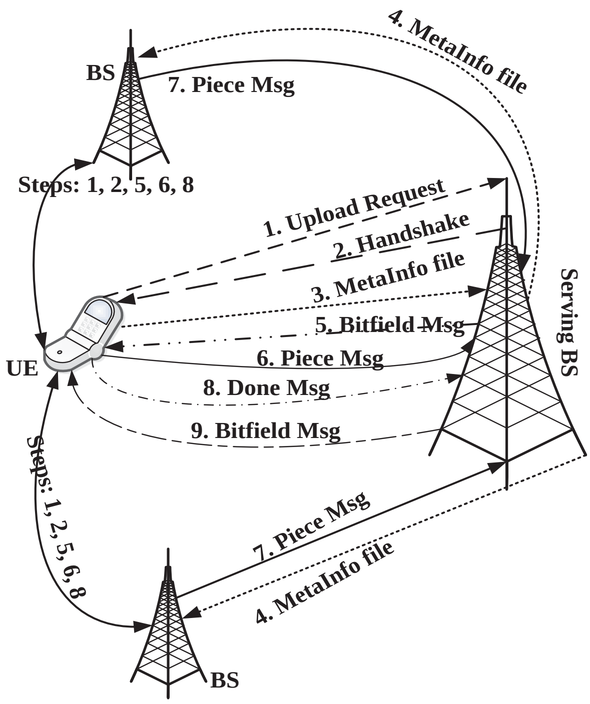

Figure 4. Segmented Upload Algorithm Message Transfer

The steps are as follows:

i. The UE sends an *UploadRequest* message to all the BSs in its CoMP set.

ii. The BSs reply by sending a *Handshake* message.

iii. The UE sends the *MetaInfo* file to the serving BS.

iv. The Serving BS forwards the *MetaInfo* file to other BSs in the CoMP set.

v. The BSs acknowledge the receipt of this file by sending the *Bitfield* message, which also tells the UE about the pieces available on the BSs.

vi. The UE sends the pieces by sending the *Piece* message to all the BSs in its CoMP set (since the messages are sent via TCP, they do not need to be acknowledged as the transmission is guaranteed by TCP).

vii. The BSs send the received pieces to the serving BS through the *Piece* message, once they receive them.

viii. The UE stops the data transfer by sending the *Done* message, as soon as all the pieces are sent.

ix. The Serving BS acknowledges the receipts of all the pieces by sending a *Bitfield* message.

x. If the *Bitfield* message does not acknowledge the receipt of all the pieces, the UE continues sending the missing pieces until completion, and repeats from step *viii*.

As mentioned in step *vii*, the non-serving BSs in the UE's CoMP coordination set forward the pieces they receive from the UE to the UE's serving BS. The cost of the additional data transfer is the overhead incurred on the mobile network's backhaul, where BSs exchange data through the X2 interfaces. In the conventional non-cooperative method, each UE sends the entire data file to only its serving BS and data transfer over the backhaul is not required. As a result, the SSU algorithm imposes more overhead on the network's backhaul X2 links.

IV. MODELING OF THE MOBILE NETWORK IN DEVS

We have developed a DEVS model for studying a mobile network employing the proposed algorithm. The model consists of various sub-coupled models and the atomic models.

As seen in Fig. 5, the top level is called Area. This coupled model includes one atomic model, Log Manager, and three other coupled models (Switch, UE Manager, and BS Manager) and the interconnections among these models. Log Manager is responsible for gathering statistics during the simulations. The Switch coupled model in Fig. 5 consists of two atomic models and the required interconnections. It models the communication between each pair of BSs and UEs. Instead of defining interconnections for each pair (which can grow quickly as the size of the model increases), the Switch is used to receive all the sent messages from each UE and BS, and broadcasts them to all the other models. The BSs and UEs can then recognize their messages based on the destination address field of the received message. Both the BS Manager and UE Manager (as the coupled models) almost have the same structure. Each of them has a Registry unit (as a DEVS atomic model), which is responsible for some management and control actions for BSs and UEs, respectively. In addition, BS Manager and UE Manager can have any number of BS and UE models, depending on the defined size of Area. The number of UEs in UE Manager is usually between four or five times the number of BSs (this is the reason that explains how the structure of UE Manager and BS Manager can be different). Both UE and BS models are coupled models, and they are composed of two atomic models: Queue and Processor. The arrival of a message at a Queue is processed based on the message's delay time. Therefore, among all the messages in the queue, the one with the least delay time leaves the queue first. The Processor of the UEs and the BSs operates based on the definition of SSU.

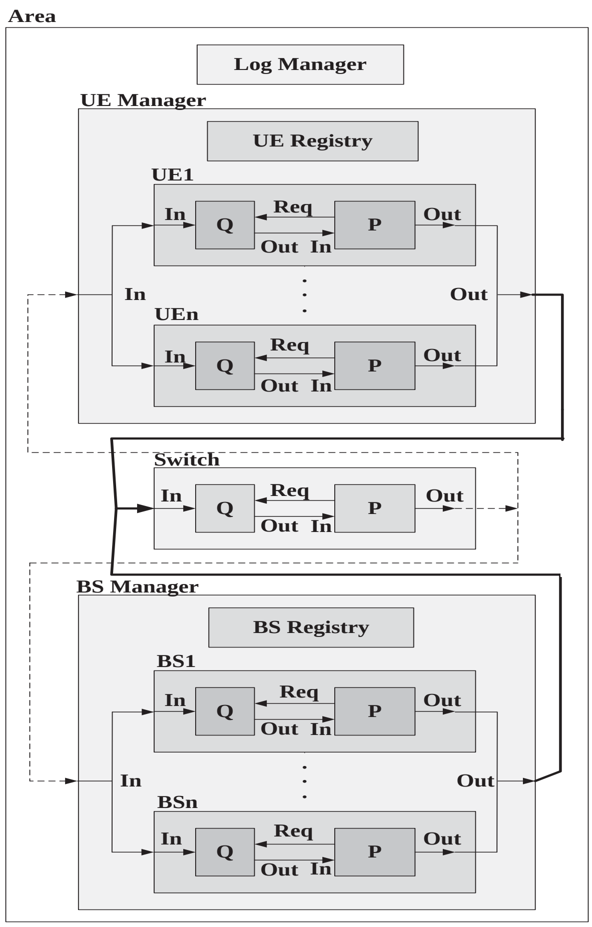

Figure 5. Simplified DEVS model hierarchy for a mobile network model (Q: Queue, P: Processor)

Fig. 6 shows a simplified segment of a model file representing the DEVS model hierarchy for our system (based on the one presented in Fig. 5). The model file is used to define a DEVS Coupled model and its hierarchical structure using the CD++ tool [25]. In the model file, Coupled models list their components and links between them, and Atomic models can list some/all their parameters.

```
[top]
components : logManager@LogManager  switch
components : UEmanager BSmanager
...
Link : out@switch in@UEmanager
Link : out@switch in@BSmanager
Link : out@UEmanager in@switch
Link : out@BSmanager in@switch

[logManager]
bsCounter : 16
ueCounter : 64
...

[switch]
Components: switchQueue@SwitchQueue
Components: switchProcessor@SwitchProcessor
in : in
out : out
Link : in in@switchQueue
Link : req@switchProcessor req@switchQueue
Link : out@switchQueue in@switchProcessor
Link : out@switchProcessor out
...
[switchQueue]

[switchProcessor]

[UEmanager]
components : UEregistry@UERegistry UE1 UE2... UE64
...

[UEregistry]
areaConfiguration : rural
...

[UE1]
components: UE1Queue@UEQueue  UE1Processor@Node
...
[UE1Queue]
UEId : 1

[UE1Processor]
UEId : 1
...

[BSmanager]
components : BSregistry@BSRegistry  BS1 ... BS16
...

[BSRegistry]
...

[BS1]
components : BS1Queue@BSQueue BS1Processor@BS
...

[BS1Queue]
...

[BS1Processor]
BSId : 1
areaConfiguration : 0  #0 = rural, 1 = urban.
Dhb : 15
Hb : 45
frequency : 900
...
```

Figure 6. Simplified Model file of an rural area

As seen in Fig. 6, we started by defining Area coupled model (from Fig. 5) as the top model. We introduced the components of Area model and the required interconnection among these sub models based on the depicted model in Fig. 5. As the next step, we required to define Log Manager, Switch, UE Manager and BS Manager in Model file. As we mentioned in the previous section, Log Manager is a DEVS atomic model. Among all the parameters of a DEVS atomic model, the ones that we want to pass a value for them are being defined in the Model file. After introducing Log Manager, we defined the Switch coupled model. This coupled model includes two atomic components (switch Queue and switch Processor) and the interconnections (Fig. 5). We needed to define switch components in the model file as well. After defining Switch model and its components completely, we needed to define other sub-coupled models of the top model. These sub coupled model themselves may include one or more atomic/coupled models. Therefore, we need to define them in turn as well. By following these steps until defining all the models, we can implement the hierarchical structure of our model in Fig.5 in a Model file.

Aside from the atomic model components described earlier, a few other passive classes have been added to complete the model. A simplified UML class diagram presenting these classes is shown in Fig. 7. The BS class represents a BSProcessor using id, type, coordinates, height from ground, height from the average rooftop, carrier frequency, transmission power, antenna gain, and a list of connections with the UEs in range. The Node class characterizes a UEProcessor with an id, current coordinates, destination coordinates, speed, transmission power, antenna gain, and a list of connections to the in-range BSs. The BS and Node classes define the behaviour of the corresponding component based on the algorithm being simulated. The UE and BS parameters mentioned above can be initialized using the DEVS model file in order to construct the simulation scenario.

The UELink class defines a list held by every BS that contains the downlink parameters of the communication link to each of the UEs within range. These parameters include the separation distance, path loss, and the received power. Similarly, the BSLink class is a list held by every Node object, and it contains parameters similar to those in the UELink class for the uplink connection. The two respective classes have methods to calculate link parameters such as propagation model, path loss, and the received power in rural area settings. The Nodes and BSLink classes hold pointers to the head of linked lists of UEs and BSs in the area respectively.

The UE and BS parameters such as transmission power, gain, UE speed, BS antenna height and operating frequency can be initialized to specific values in the model file to create a simulation scenario. These parameters can also be set automatically by defining the area type in the model file to either rural or urban simulation scenarios. This allows us to change the simulation scenario rapidly for different simulation executions. Other simulation properties such as BS power, noise figure and noise power are automatically set by choosing one of the following communication standards defined in [26]: UTRA FDD, UTRA 1.28MHz TDD, UTRA 3.84MHz TDD, and E-UTRA FDD and E-UTRA TDD.

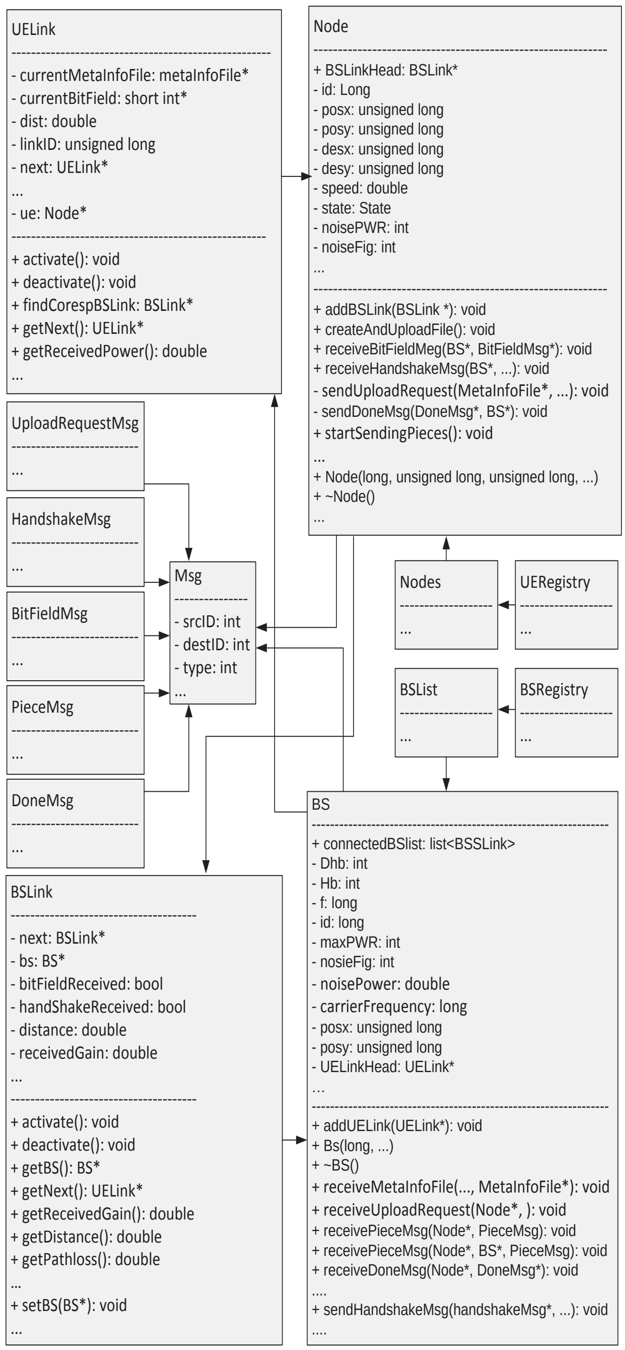

Figure 7. Simplified class diagram of the model

UERegistry, an Atomic component in the UE Manager Coupled model, is triggered in 100 millisecond intervals to update the state of the wireless network. It makes use of the list of UEs in Nodes to update their current position based on the previous position, and the predefined destination coordinates and speed, and the elapsed time since the last update. Moreover, UERegistry periodically updates the status of the communication channels between UEs and BSs. The status includes the validity of the links depending on the distance between the corresponding UE and BS, as well as the uplink and downlink channel parameters discussed earlier.

A class hierarchy for messages has been implemented to encapsulate the contents of the messages used by SSU and allow the model components to communicate with fewer messages. Msg objects include IDs of the source and destination components as well as the size of the message object and its type. Subclasses of the Msg superclass define fields specific to the message, as defined by the algorithm. The Msg class hierarchy is shown in Fig. 7.

In this model, a UEProcessor only handles the upload of one file at a time, and therefore, its state transitions directly correspond to the different steps of the proposed algorithm. The UEProcessor states are: *Idle*, *CreateAndUpload*, *UploadRequest*, *RcvHandshakeWait*, *RcvHandshake*, *SendMetaInfo*, *RcvBitFieldWait*, *RcvBitField*, *SendPiece*, *SendDone*, *RcvDoneBitFieldWait*, and *RcvDoneBitField*. Fig. 8 shows the UEProcessor DEVS graph.

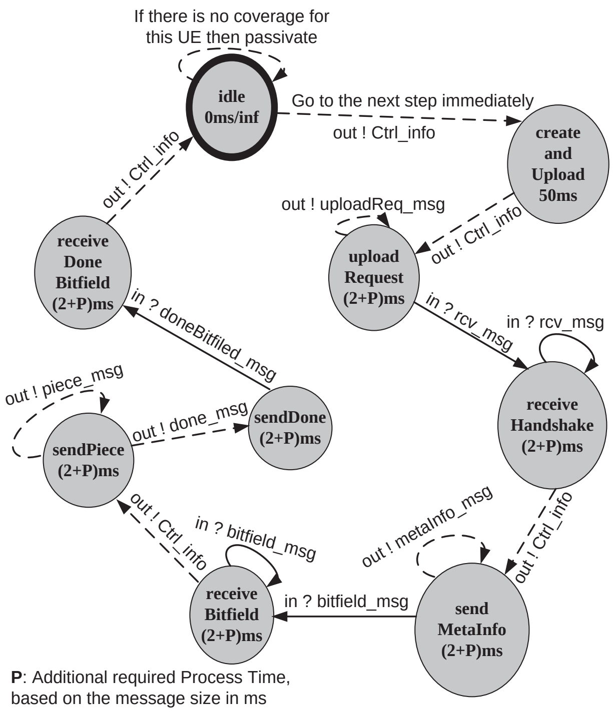

Figure 8. UEProcessor DEVS graph

On the other hand, since a BSProcessor handles incoming and outgoing messages to neighbouring BSs and multiple UEs, its state machine tends to be complex. To simplify its state transitions, a BSProcessor only cycles through four states, namely, *Idle*, *Receive*, *Process*, and *Send*, once for each external message received. Fig. 9 shows a DEVS graph for BSProcessor. For example, when a BS in idle state and receives an upload request, it replies with a Handshake message, and it changes to the idle state again. Thus, the behavior of the model depends on the state of the UELink between the BS and the corresponding node at which the received message originated. UELink is not an Atomic DEVS model. Rather, it is just a data structure, which keeps track of events between each pair of UE and BS. The UELink states are *idle*, *receiveUploadRequest*, *sendHandshake*, *receiveMetaInfo*, *sendMetaInfo*, *receivePiece*, *sendPiece*, *receiveDone*, and *sendBitField*. Fig. 10 shows a state diagram for UELink.

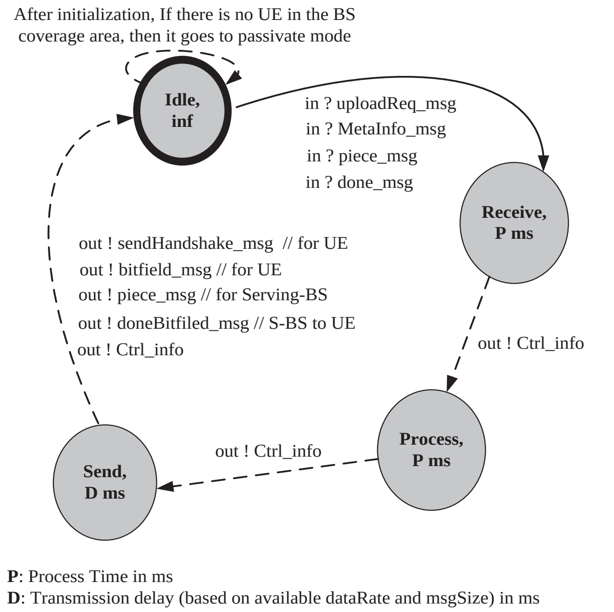

Figure 9. BSProcessor DEVS graph

In order to evaluate the performance of SSU, another conventional non-cooperative algorithm was implemented. This non-cooperative algorithm represents a simplistic upload model where a UE only communicates with its serving BS. The upload process begins with an upload request message from the UE (similar to SSU) to the BS, which is acknowledged by the BS with a Handshake message. A file is then uploaded in a stream of variable sized packets, or segments. The size of the packets depends on the available bandwidth and data rate of the communication channel. The upload processes ends when the UE sends the BS a Done message following the last data Piece message.

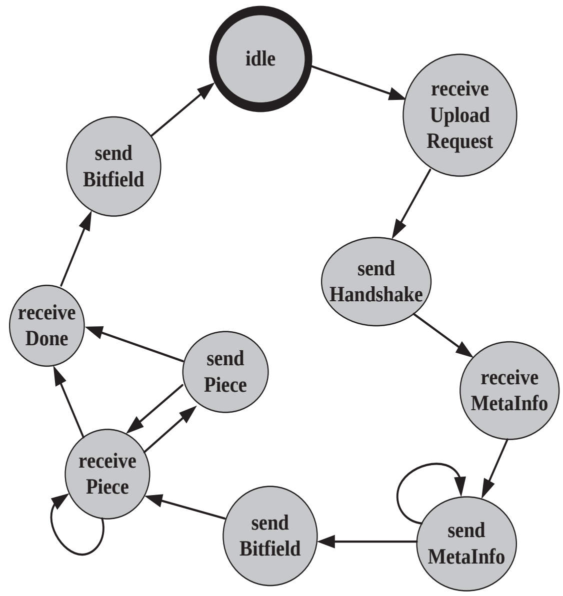

Figure 10. UELink state diagram

V. Simulation Scenario and Results

To assess the potential of SSU, we ran a series of system level simulations. The algorithm aims to improve the throughput and data rates for cell-edge users; therefore, the effectiveness of the algorithm needs be evaluated as a function of distance from the cell center. In each iteration of the simulation, the UEs are randomly positioned in the system area within a narrow predefined range of distances, measured from the center of the serving BS cell. The simulations are carried out in both rural area and urban area settings. 900 MHz is used as the operating carrier frequency in the rural setting. In the urban area setting, two operating carrier frequencies are considered in these simulations: 900 MHz and 2000 MHz. 5 MHz is used as transmission bandwidth for both rural and urban setting. In addition, the noise density is assumed to be fixed at -174 dBm/Hz and the log-normally distributed shadowing (LogF) is set to a standard deviation of 10dB. The other detailed simulation parameters are listed in Table 2 and 3 [26]. Table 2 includes required simulation parameters for rural setting. In case of urban setting, the same parameters with the same values are used, unless in Table 3 there is another value for a parameter. The UERegistry atomic model is responsible for periodically updating the UEs' locations based on their current locations, their predefined random destinations, and speeds. This periodically updates the propagation model (L) for the links between each pair of BSs and UEs. The updated propagation model is needed to calculate received signal power at the receiver side. The available data rate at the link between UEs and BSs can then be calculated. The following formulas show the required steps to calculate link data rate.

TABLE II.
KEY SIMULATION PARAMETERS IN A RURAL SETTING

Parameter	Value
Frequency	900 MHz
Transmission bandwidth	5 MHz
Noise Density	-174 dBm/Hz
MCL (R)	80 dB
BS Antenna Gain	15 dB
BS Antenna Height above rooftop (Dhb)	15 meters
BS Antenna Height above ground (Hb)	45 meters
LogF	10dB
File size	0.5MB-64MB
Maximum BS power	43 dBm
Maximum power per DL traffic channel	30 dBm
Minimum BS power per user	15 dBm
BS Noise figure	5 dB
Maximum UE power	21 dBm
Minimum UE power	-50 dBm
UE Noise figure	9 dB

TABLE III.
KEY SIMULATION PARAMETERS IN AN URBAN SETTING

Parameter	Value	
Frequency	900 MHz	2000 MHz
BS Antenna Gain	12 dB	15 dB
MCL	70 dB	

Let's assume that R is the BS-UE separation in kilometers, f is the carrier frequency in MHz, Dhb is the base station antenna height in metres, measured from the average rooftop level and Hb is the BS antenna height above ground (in meters). Then, the Macro cell propagation model for rural urban areas is given by the following formulas, (1) and (2) [26].

$$L_{rural} = 69.55 + (26.16 * log_{10} f) - (13.82 * log_{10} hb) + \left(\left(44.9 - (6.55 * log_{10} hb)\right) * log_{10} R\right) - (4.78 * (log_{10} f)^2) + (18.33 * log_{10} f) - 40.94 \quad (1)$$

$$L_{urban} = (40 * \left(1 - (4 * 10^{-3} * Dhb)\right) * log_{10} R) - (18 * log_{10} Dhb) + (21 * log_{10} f) + 80dB \quad (2)$$

Considering the log-normally distributed shadowing (LogF) with standard deviation of 10dB, the pathloss is given by (3) [26].

$$pathloss = L_{rural/urban} + LogF \quad (3)$$

The received signal power at each UE and BS is calculated by (4) [26]. The Max method in (4) returns the greater value between two parameters.

$$RX\_PWR = TX\_PWR - Max(pathloss - G\_TX - G\_RX, MCL) \quad (4)$$

where RX_PWR is the received signal power, TX_PWR is the transmitted signal power, G_TX is the transmitter antenna gain, G_RX is the receiver antenna gain, and MCL is the minimum coupling loss.
The link data rate can then be calculated taking into account Additive White Gaussian Noise (AWGN), using (5), where B is the transmission bandwidth and N_0 is the noise variance.

$$data\ rate = B\ log_2(1 + \frac{RX\_PWR}{N_0 * B}) \quad (5)$$

A set of simulations was performed in a rural area setting with an operating area of 8 km by 8 km, and BS-UE distances increasing in increments of 800 meters up to 8 kilometers. In other words, the first simulation positioned UEs within the first 800 meters around their serving BS, the second simulation had UEs placed between 800 and 1600 meters from the BS, and so on. In the case of urban area simulations, we use 17 BSs to provide radio coverage over a geographical area of 2800 meters by 3000 meters. Similar to the rural area simulation, in each of the urban area simulations, the UEs are located at a predefined distance range from their serving BSs. The width of this distance range in which UEs are located initially is 50 meters. This means that in the first simulation the UEs are within the first 50 meters around their serving BSs, and in the second iteration, they are located between 50 and 100 meters from the serving BS, and so on.

In both simulation categories, there are 64 active UEs and each the UEs uploads one file during the simulation. The simulations were allowed to run until all the file uploads were complete and the simulation statistics were collected. These files were then analyzed and some of the chosen results are shown in the following figures. In all of these figures, the horizontal axis shows the average UEs distance from their serving BSs.

A user can only communicate with its serving BS when traveling in a cellular network that uses a conventional non-cooperative algorithm. This case is true even when the user is in the cell edge areas. On the other hand, the SSU algorithm provides higher data rate by allowing the UEs to communicate with the BSs in the coordination set while they are close to the cell edge area. Fig. 11 shows the average number of BSs that each UE communicates with during the uploading process in rural area setting. Fig. 12 and Fig. 13 show the same thing during the upload process in an urban configuration with carrier frequencies of 900 MHz and 2000 MHz respectively.

With the conventional non-cooperative algorithm in use, a UE only communicates with its serving BS resulting in an average number of connected BSs of one. When the UE is close to the center of the cell, SSU behaves in a similar manner to the conventional algorithm in terms of the number of BSs the UE communicates with. However, as the UE moves closer to the cell's edge, it is more likely that the UE will receive signals of multiple BSs. This explains why, as seen in the rest of the figures, SSU provides better performance for UEs close to the cell's edge. According to Fig. 11, Fig. 12, and Fig. 13, a UE starts receiving signals from multiple BSs around 150 meters and 2400 meters from the cell center in rural and urban areas, respectively.

As seen in Fig. 11, when a UE employing SSU is close to the cell border (around 7500 meters from the cell center), the UE is able to communicate with an average of 2.7 BSs. Similarly, in an urban area with operating frequencies of 900 MHz and 2000 MHz, a UE using SSU is able to communicate with an average of 2.73 and 2.64 BSs respectively (Fig. 12 and Fig. 13).

Fig. 14 shows the average upload time for a data file as a function of distance for the SSU algorithm, as well as the conventional non-cooperative algorithm in a rural area setting. Fig. 15 and Fig. 16 show the results of the same parameter for both the SSU algorithm and the conventional non-cooperative algorithm in an urban area setting with the two selected operating frequencies. The upload process starts with the UploadRequest message from the UE and it ends when the UE receives the BitField message from its serving BS. In rural areas, a UE employing SSU is able to complete the upload process in 5.6 seconds when it is 800 meters from its serving BS and in 284 seconds when it is 8000 meters from the cell center. In contrast, a UE using the conventional non-cooperative algorithm requires an equal amount time when it is close to the serving BS. However, near the cell edge, the upload process required 546.8 seconds. As seen in Fig. 15, the upload process using both algorithms took 3.85 seconds close to the cell center when operating at 900 MHz in an urban area. At 500 meters away from the UE's serving BS, a UE using the conventional algorithm uploaded the file in 34.49 seconds while a UE using SSU uploaded the file in 17.29 seconds. Similarly, in an urban network with an operating frequency of 2000 MHz, SSU and the conventional algorithm required 5.22 seconds close to the serving BS, and 37.5 and 67.9 seconds at 500 meters from cell center, respectively. It is clear that in urban areas, signal attenuation is higher at 2000 MHz compared to 900 MHz, leading to a longer average upload time. These figures reveal that as the UEs' distance from the cell center increases, the rate of increase in average upload time for the conventional algorithm is higher than that of the SSU algorithm. This means that the SSU algorithm provides services that are more consistent to its users, regardless of their location within their cell. The effects of SSU on the average upload process time can be seen more clearly, when the UEs moves further away from their serving BSs. Specifically, the effectiveness of SSU becomes apparent when the UE is around 4400 meters away from the cell center in rural areas. In an urban area setting, the effectiveness of SSU is clear when the UE is 200 meters and 250 meters away from the cell center with frequencies of 900 MHz and 2000 MHz respectively. Closer to the center of the cell, SSU slightly increases the overhead, caused by the additional control messages.

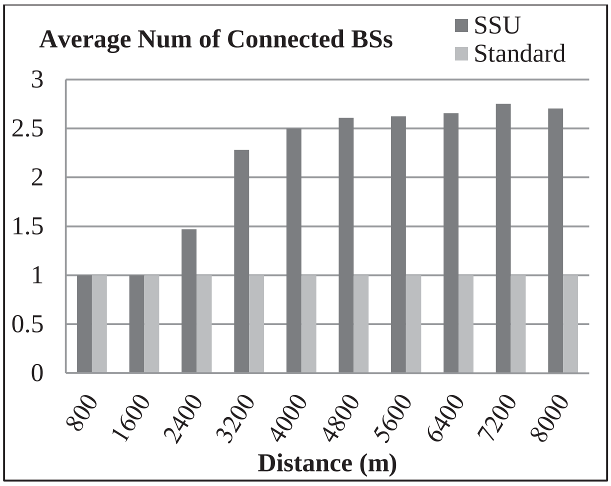

Figure 11. Average number of connected BSs vs. Distance from BS (rural area)

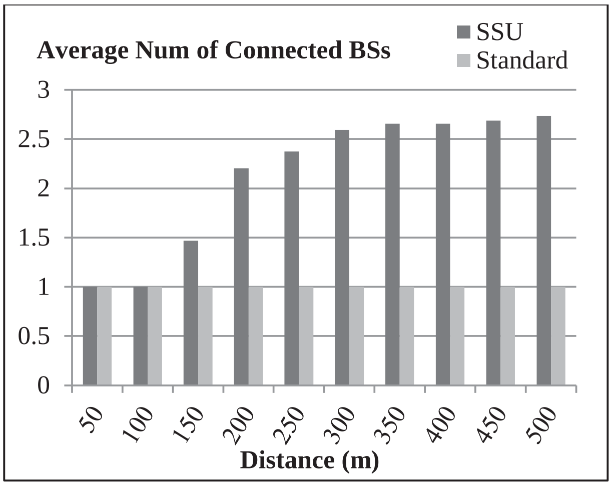

Figure 12. Average number of connected BSs vs. Distance from BS (urban area, 900 MHz)

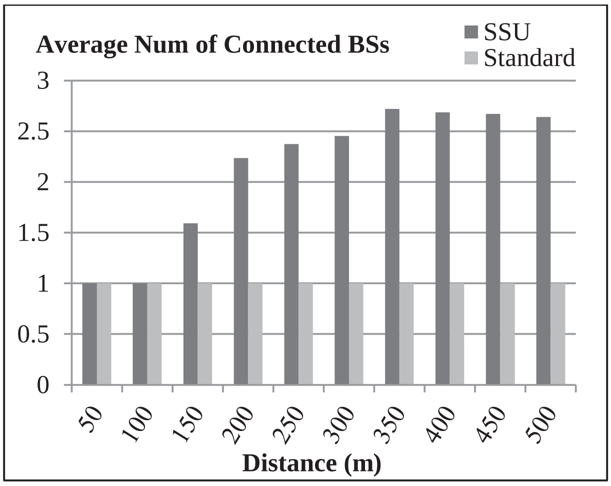

Figure 13. Average number of connected BSs vs. Distance from BS (urban area, 2000 MHz)

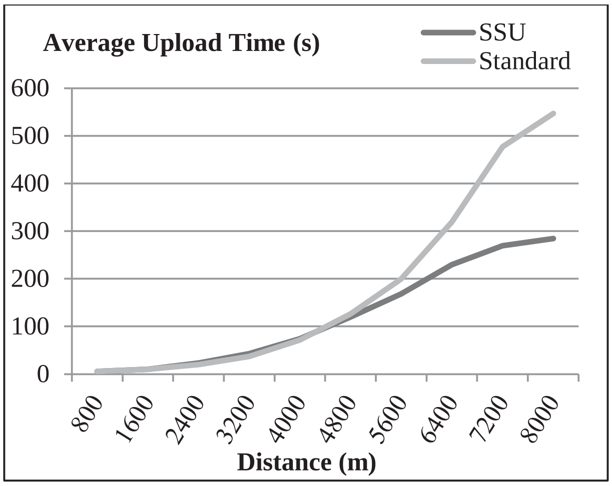

Figure 14. Average file upload time vs. Distance from BS (rural area)

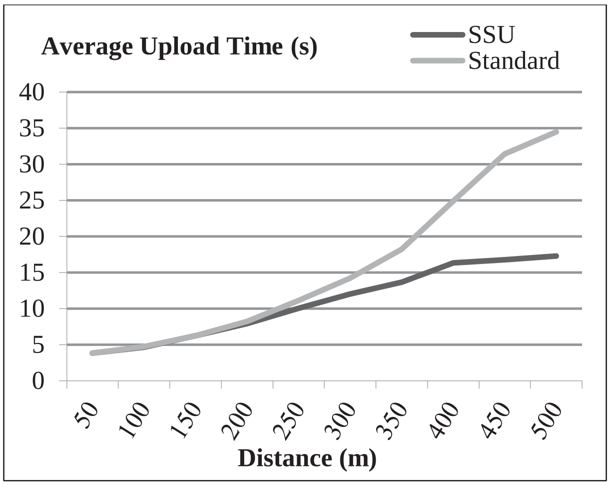

Figure 15. Average file upload time vs. Distance from BS (urban area, 900 MHz)

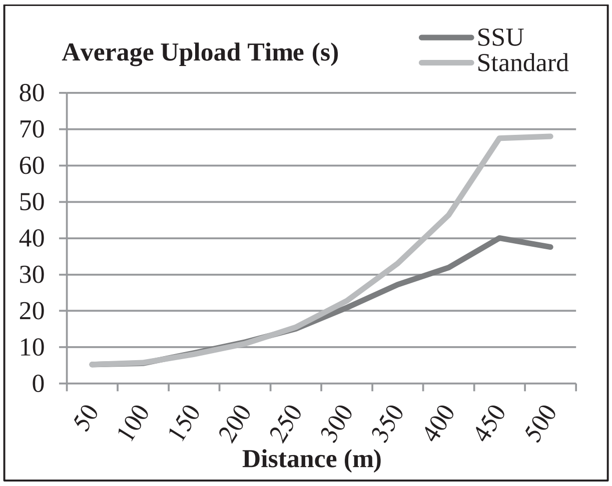

Figure 16. Average file upload time vs. Distance from BS (urban area, 2000 MHz)

Finally, Fig. 17 shows the comparison between the SSU algorithm and the conventional algorithm with respect to the average data rate they provide for the UEs during the simulations in a rural area setting. Fig. 18 and Fig. 19 present the same parameter in the urban area setting. These figures depict results similar to what is seen in Fig. 14, Fig. 15, and Fig. 16. According to these figures, close to the cell edge, SSU provides nearly twice the data rate compared to the conventional non-cooperative scheme, in both rural and urban areas. Similar to the pervious results, the effectiveness of SSU becomes clear at approximately the same distance from the cell's center (4400 meters in rural networks and 225 meters in urban networks). At distances closer to the BS, the performance of SSU is nearly equal to that of the conventional non-cooperative algorithm but imposes a small overhead, slightly decreasing the data rate provided to the UE.

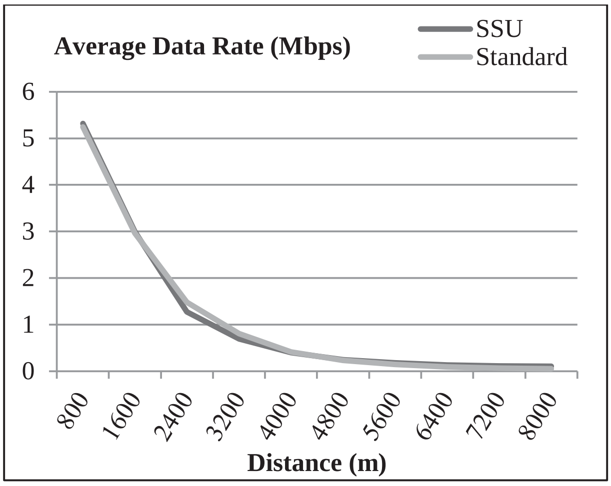

Figure 17. Average data rate vs. Distance from BS (rural)

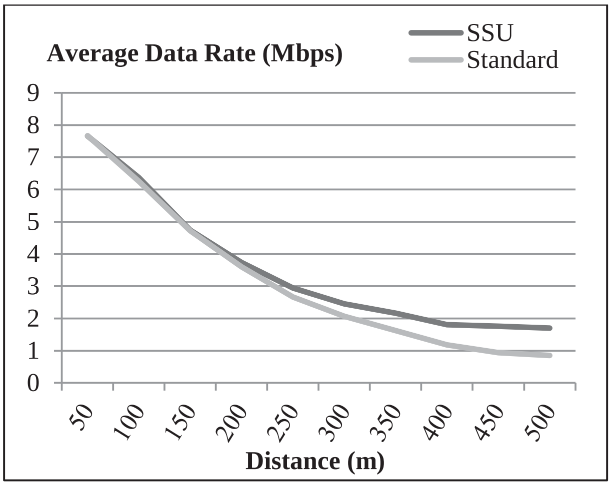

Figure 18. Average data rate vs. Distance from BS (urban area, 900 MHz)

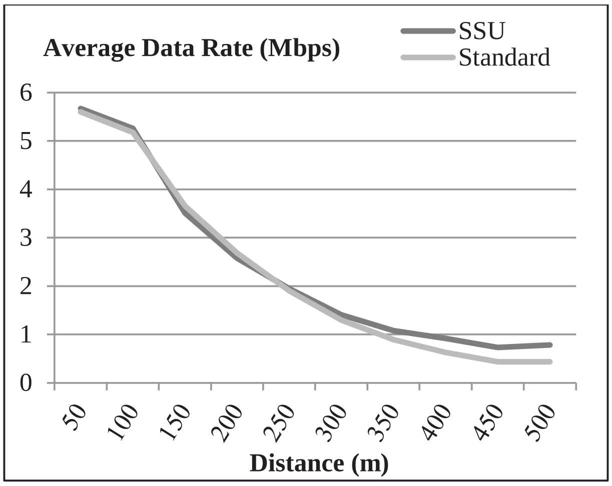

Figure 19. Average data rate vs. Distance from BS (urban area, 2000 MHz)

VI. Conclusion

We have used Discrete Event System Specification (DEVS) formalism to model and simulate LTE-A mobile networks using two approaches: Shared Segmented Upload algorithm (SSU) and a conventional non-cooperative method. The SSU is an uplink schema for LTE-Advanced networks. CD++ software was used as the platform to model and implement the cellular network for both rural and urban area settings. The simulation results show that SSU provides services that are more consistent to the users as their distance increases from the cell center. Compared to the conventional method, SSU provides higher data rate for the users and reduces the required time for a UE to upload its data to the network. Considering the large amount of data that required to be transmitted over a mobile network, further investigation is required to study its influence on the backhaul. Moreover, we need to extend the proposed SSU algorithm to reduce such overhead.

References

[1] ICT Data and Statistics Division, Telecommunication Development Bureau, International Telecommunication Union, “The World in 2013, ICT Facts and Figures,” http://www.itu.int/en/ITU-D/Statistics/Documents/facts/ICTFactsFigures2013-e.pdf. Accessed on November 2014.

[2] MobiThinking Latest Mobile Stats, http://mobithinking.com/mobile-marketing-tools/latest-mobile-stats/a. Accessed on October 2014.

[3] I. F. Akyildiz, D. M. Gutierrez-Estevez, E. C. Reyes, “The Evolution to 4G cellular system: LTE-Advanced.” Physical Communication, Volume 3, No 4, pp. 217-244, 2010.

[4] M. Sawahashi, Y. Kishiyama, A. Morimotol, D. Nishikawa, M. Tanno, “Coordinated Multipoint Transmission/Reception Techniques for LTE-advanced.” IEEE Wireless Communication Journal, Volume 17, No 3, pp. 26-34, June 2010.

[5] S. Parkvall, D. Astely, “The Evolution of LTE Towards IMT-Advanced.” Journal of Communications, Volume 4, No 3, pp. 146-154, April 2009.

[6] M. Tavanpour, G. Wainer, G. Boudreau and R. Casselman, “DEVS-based Modeling of Coordinated Multipoint Techniques for LTE-Advanced.” 16th Communications and Networking Symposium, San Diego, CA. April 2013.

[7] B. Cohen, "The BitTorrent Protocol Specification," http://bittorrent.org/beps/bep_0003.html. Accessed October 2014.

[8] T. Tran, T. Shin, O. Shin, “Overview of Enabling Technologies for 3GPP LTE-Advanced.” EURASIP Journal on Wireless Communications and Networking, Volume 2012, No 1, February 2012.

[9] V. Jungnickel, A. Forck, S. Jaeckel, f. Bauermeister, S. Schiffermuller, S. Schubert, S. Wahls, L. Thiele, T. Haustein, W. Kreher, J. Mueller, H. Droste, G. Kadel, “Field Trials Using Coordinated Multi-Point Transmission in the Downlink.” IEEE 21st International Symposium on Personal, Indoor and Mobile Radio Communications (PIMRC) Workshops, Istanbul Turkey, 2010.

[10] A. Osseiran, J. Monserrat, W. Mohr, “Mobile and Wireless Communication for IMT-Advanced and Beyond.” Wiley Publications, 2012.

[11] R. Irmer, H. Droste, P. Marsch, M. Grieger, G. Fettweis, S. Brueck, H, Mayer, L. Thiele, V. Jungnickel, “Coordinated Multipoint: Concepts, Performance and Field Trial Results.” IEEE Communications Magazine, Volume 49, No 2, pp. 102 – 111, 2011.

[12] X. Zhnag, M. A. Khojastepour, K. Sundaresan, S. Rangarajan, K. G. Shin “Exploiting Interference Locality in Coordinated Multi-Point Transmission Systems.” IEEE International Conference on Communications, Ottawa, ON, 2012.

[13] S. Mehryar, A. Chowdhery, W. Yu, “Dynamic Cooperation Link Selection for Network MIMO Systems with Limited Backhaul.” IEEE International Conference on Communications, Ottawa, ON, 2012.

[14] S. Brueck, L. Zhao, J. Giese and M.A. Amin, “Centralized Scheduling for Joint Transmission Coordinated Multi-Point in LTE-Advanced.” International ITG Workshop on Smart Antennas, Bremen, Germany, 2010.

[15] Xianqi LU, Si Chen, Wen-an Zhou and Junde Song, “Analysis of Signaling Design in LTE-Advanced Coordinated Multipoint Transmission/Reception System.” IEEE Conference on Computer Science and Automation Engineering (CSAE), Zhangjiajie, China. 2012.

[16] V. Jungnickel, L. Thiele, T. Wirth, T. Haustein, S. Schiffermuller, a. Forck, S. Wahls, S. Jaeckel, S. Schubert, H. Gabler, C. Juchems, F. Luhn, R. Zavrtak, H. Droste, G. Kadel, W. Kreher, J. Mueller, W. Stoermer, G. Wannemacher, "Coordinated Multipoint Trials in the Downlink." IEEE GLOBECOM Workshop, Honolulu HI, 2009.
[17] D. Lee, H. Seo, B. Clerckx, E. Hardouin, D. Mazzarese, S. Nagata, K. Sayana, "Coordinated Multipoint Transmission and Reception in LTE-Advanced: Deployment Scenarios and Operational Challenges." IEEE Communications Magazine, Volume 50, No 2, pp 148–155, February 2012.
[18] Q. Qiu, J. Chen, L. Ping, Q. Zhang, X. Pan, "LTE/SAE Model and its Implementation in NS 2." 5th International Conference on Mobile Ad-hoc and Sensor Networks (MSN '09), Fujian China, 2009.
[19] A. Koc, S. Jha, R. Vannithamby, M. Torlak, "Device Power Saving and Latency Optimization in LTE-A Networks Through DRX Configuration," IEEE Transactions on Wireless Communications, Volume 13, No 5, pp 2614-2625, March 2014.
[20] H. Safa, W. El-Hajj, K. Tohme "A QoS-Aware Uplink Scheduling Paradigm for LTE Networks," IEEE 27th International Conference on Advanced Information Networking and Applications, Barcelona, Spain, 2013.
[21] L. Bajzik, P. Horváth, L. K_rössy, C. Vulkán, "Impact of Intra-LTE Handover with Forwarding on the User Connections," Mobile and Wireless Communications Summit, Budapest, Hungary, 2007.
[22] J. Brown and J Y. Khan, "Performance Comparison of LTE FDD and TDD Based Smart Grid Communications Networks for Uplink Biased Traffic," IEEE Third International Conference on Smart Grid Communications (SmartGridComm), Tainan, Taiwan, November 2012.
[23] R. Sivaraj, A. Pande, P. Mohapatra, "Spectrum-aware Radio Resource Management for Scalable Video Multicast in LTE-Advanced Systems," IFIP Networking Conference, Brooklyn, NewYork, May 2013.
[24] Zeigler, B., T. Kim, and H. Praehofer, "Theory of Modeling and Simulation," Academic Press, 2000.
[25] G. A. Wainer, "Discrete-event modeling and simulation: a practitioner's approach," CRC Press, 2009, ISBN 978-1-4200-5336-4.
[26] 3rd Generation Partnership Project, "Technical Report 36.942, V10.2.0," 2010-2012.