International Journal of Information, Control and Computer Sciences
ISSN: 2517-9942
Vol:4, No:10, 2010

Throughput Enhancement in AUDTWMN Using Throwboxes – An Overview

Laveen Sundararaj, Palanisamy Vellaiyan

Abstract**—**Delay and Disruption Tolerant Networking is part of the Inter Planetary Internet with primary application being Deep Space Networks. Its Terrestrial form has interesting research applications such as Alagappa University Delay Tolerant Water Monitoring Network which doubles as test beds for improvising its routing scheme. DTNs depend on node mobility to deliver packets using a store-carry-and forward paradigm. Throwboxes are small and inexpensive stationary devices equipped with wireless interfaces and storage. We propose the use of Throwboxes to enhance the contact opportunities of the nodes and hence improve the Throughput. The enhancement is evaluated using Alunivdtnsim, a desktop simulator in C language and the results are graphically presented.

Keywords**—**Alunivdtnsim – Alagappa University Delay Tolerant Network Simulator, AUDTWMN- Alagappa University Delay Tolerant Water Monitoring Network, DTN - Delay and Disruption Tolerant Networking, LTP – Lick Lider Transmission Protocol.

I. INTRODUCTION

DELAY and Disruption Tolerant Networking (DTN) [8] refers to broad class of Wireless Ad-hoc networks that operate in challenged environments plagued by delays and disruptions [13]. DTN is part of the Inter Planetary Internet, an initiative started at the Jet Propulsion Laboratory (JPL) by Vint Cerf et.al in 1998. DTN has evolved over the years with major research contributions from academicians and Industry. DTN is a network of regional networks. It acts as a overlay on regional networks. DTN supports interoperability by accommodating mobility and low Radio Frequency (RF) power capabilities of the nodes involved. DTN includes RF, Ultra Wide Band (UWB) networks, Optical and Acoustic networks. Though simultaneous connectivity may be absent, a combination of store & forward, along with node mobility makes message delivery possible. The bundle protocol [9] is a DTN protocol based on overlay technique. It can be used on any convergence layer such as TCP, UDP and LTP. The Lick-Lider Transmission protocol [12] is another DTN specific protocol operating at convergence layer. While the bundle protocol moves data packets (bundles) end to end, the LTP is more of a point to point type. While the space applications which are the primary beneficiaries of the DTN [11] have provided ample scope for its research, many terrestrial applications has been conceived that use and contribute to DTN research. Few of such terrestrial applications [12] include:

1. Reindeer herd tracking by the Saami tribesmen in Arctic Circle.
2. Zebra tracking to monitor the movement of zebra and manage their habitat effectively in Africa.
3. Early detection of the invasion of Australian cane toads, a pest and invasive species in non-native regions.
4. Seismic monitoring in Mexico for early warning system against earth quakes, volcano and land slides.
5. SenDT – an initiative by the Trinity College Dublin Ireland to monitor lakes in Ireland.
6. DTN - Simple Text Message application over android OS introduced in Nexus One Cellular phones by Google Inc.

Based on DTN's applicability for a multitude of terrestrial applications, the authors devised upon a DTN [20] based water monitoring network [10][1] named AUDTWMN that will serve the dual purposes of environmental conservation and research. It can be used to monitor critical parameters of water bodies in environmental context and as a test bed for DTN in research context. Monitoring water bodies in district of Sivaganga [23] seemed to be an important environmental need due to the following reasons:

1. It has an area of 4468.11 Sq. Km and is classified as drought prone.
2. It suffers from both poor soil quality and insufficient rainfall.
3. It has forest land and cropped land constituting 5% and 27% respectively of its total area.
4. It has an average population growth of 9.78% every year.
5. Urbanization is increasing by 2% every year.
6. It has no water sampling location, the nearest being Parthibanur (under Madurai surface water quality monitoring [24] station's Jurisdiction) which is upstream of the Vaigai river from Sivaganga. The next one down stream is on the Ramanad big tank on the distributary that does not flow into Sivaganga.

All these factors, combined with absence of perennial rivers (Vaigai and Thenar are seasonal) or dams, the pressure on available water bodies is increasing.

Laveen Sundararaj is a Honeywell Research scholar with the Department of Computer Science and Engineering, Alagappa University, Karaikudi, Tamil Nadu 630003 INDIA. (E-mail: Laveen. Sundararaj@ Honeywell.com; Phone: +91-93422-41464; fax: +91-80-3983-4455).

Dr. Palanisamy Vellaiyan is the Head of the Department of Computer Science and Engineering, Alagappa University, Karaikudi, Tamil Nadu 630003 INDIA. He is also the chairman of the School of Computer Science, Alagappa University, Karaikudi. (Email: vpazhanisamy@yahoo.co.in)

Having done a preliminary design of AUDTWMN, the simulation results pointed that only 38.4% of the message packets from the interior sensor node covered by Cattlenet reached the data centre. Hence it became imperative to devise means to improve the throughput of Cattlenet and hence AUDTWMN. The rest of the paper describes the design considerations for inclusion of Throwboxes [16] in AUDTWMN [1], its simulation using Alunivdtnsim [4] and analysis of the simulation runs.

II. DESIGN CONSIDERATIONS

A. *Overview of AUDTWMN*

AUDTWMN intends to monitor 5 strategically located lakes which are representative of the water bodies in the district. The application is also scalable so as to accommodate as many nodes as there are water bodies. The application must lend itself to be a research tool by enabling simple interface and reconfigurable components. With all these pre-requisites the following design has been conceived by the authors.

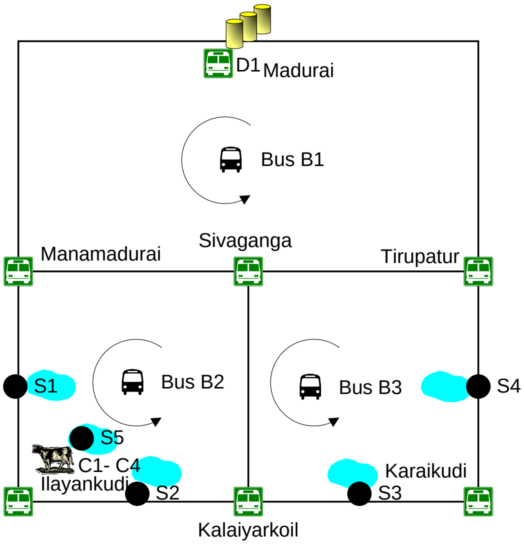

Fig. 1 Schematic of AUDTWMN

The Sivaganga district consists of 6 taluks namely, Sivaganga, Manamadurai, Iilayankudi, Devakottai, Karaikudi and Tirupatur. As the feeder river Vaigai comes under the Madurai water quality monitoring station, the data collected from Sivaganga district through the AUDTWMN shall be delivered at Madurai. The towns in considerations are conveniently connected by National Highways 45B,49,226 and State Highways 28,33,34,35. The Tamil Nadu state transport operates scheduled bus service to these towns and hence is the obvious choice for data mules. Three busses B1-B3 act as data mules. Besides the bus is a superior data mule in technical terms as it has a decent power source from its alternator and battery. As of now, 4 lakes adjoining the road have been chosen as they will be within the 30 meter radio range of the IEEE 802.11 wi-fi which forms the radio link between the sensor nodes and data mules. Four sensor nodes S1-S4 keep collecting the water parameter and transmit them to the data mules. The 5th lake is a special candidate that is located interior and away from the road. They also use special data mules which are 4 DTN node collared cows C1-C4 from a cattle herd that regularly graze the fallow lands between the road and lake. While the hardware design and development of the participating nodes is a parallel effort, this paper focuses more on the overall water monitoring application design. Such a preliminary design is visualized using Alunivdtnsim [4], which is a simple in-house simulator for DTN. The authors developed Alunivdtnsim after they had limited success customizing NS2 [7], NCTUns [3] and the ONE [6] for AUDTWMN. Not much needs to be done on the software side as the authors propose to use dtn2.5 on the nodes. The dtn2.5 running on Ubuntu Linux 9.10 platform serves as excellent DTN nodes [5]. In lab environment, DTN bundles were successfully sent across using the bundle protocol.

B. *Cattlenet*

The Cattle-net is a special portion of the AUDTWMN which makes use of Cattle as Data mule to carry message packets from sensor S5 monitoring an interior lake. By interior lake, we mean that this particular water body is situated away from the road head. Hence there is no possibility of the regular data mules (bus) being in radio range of the sensor node. This capabity of cattlenet enhances the reach of the AUDTWMN into the interior portions of the Sivaganga district. Also as DTN test bed, It makes the DTN [18] part interesting by throwing more challenging scenario & agents (cows) to be part of the network. By challenging we mean the space constraints (the whole node must fit in a radio collar), power constraints (limited solar power available by trickle charge during day time), random grazing habits exhibited by livestock at will, exposure to elements (unlike bus, the radio collar must be designed to with stand the weather). The cattlenet has been proposed in the grazing stretch of fallow land adjoining Ilaiyankudi. It is a square patch measuring 12.5 km on a side. The water body is located approximately 17 km from the Ilaiyankudi in diagonally opposite side. To start with 4 cows C1, C2, C3 and C4 form the cattle-net's data mules. Their synthetic movement has been statistically and behaviorally modeled. In Indian context a typical herd has 12 animals. They are mostly cows as oxen are used for agricultural tasks. The herd assembles at a location close to water body. They graze the whole day from dawn to dusk after which they are herded back home. The Alunivdtnsim uses a cow-director function to synthesize the movement. Although a cow typically moves at 3.5 to 7.6 km per hour, the simulator moves the cow at the upper limit of 7.6 km an hour. The whole 12.5 by 12.5 km grazing patch is divided into a 26 by 26 grid. The radio collar is active once every 250 seconds. The rest of the time it is asleep to conserve power. When a typical cow is radio mapped geographically every 250 seconds, it would have moved 500 meters every time. The movement however is ingeniously decided by the cow-director driven by a pseudo-random generator, the current position of the cow, and its previously covered graze grid points. The last 3 bits of the pseudo random generator give 8 possible values from 0 to 7. Based on these values, the cow-director moves the cow to North-West, North, North-East, West, East, South-West, South, South-East when the pseudo random 3 bits take values of 0,1,2,3,4,5,6 or 7 respectively. When the cow finds itself in the diagonal boundaries, the cow-director restricts the choice of movement in only directions that will take then into the grazing patch to avoid stray away out of the graze perimeter. When the cow finds itself in the lateral boundaries, the cow-director again restricts the choice of movement in only directions that will take it into the grazing patch away from the periphery. The above behaviors are important attributes to grazing cattle as grazing in cultivated land and reserve forest is strictly prohibited. Any such incident will have the animal impounded and will be released only on paying the incurred loss. The incurred loss is assessed by the village local governing body (Panchayat). As Sivaganga district has 27% cropped area and 5% forest land, the cow-director mimics the safe grazing habit of the cattle. In a given day, the cow tries to avoids grazing in the same location it had grazed earlier in the day. The cow-director implements this behaviour by marking the grazing grid points that are reset en-mass everyday (as does the cow when it goes to sleep at the end of the day). The cow-director however does selectively reset graze grid points in a day if it finds itself surrounded by pre-grazed grid points only.

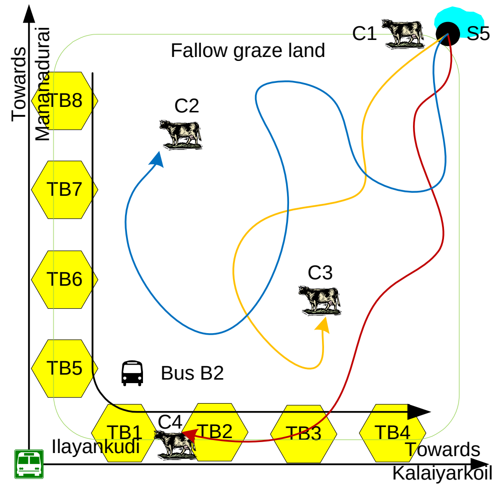

Fig. 2 Schematic of cattle-net of AUDTWMN with Throwboxes.

The DTN collar nodes have a circular memory of 8 words. They differ from other nodes by virtue of a Sent Message Packet Memory (SMPM) feature. The SMPM makes a copy of the message packets that are transmitted to the data mule bus which is the next hop towards the destination. The SMPM helps to purge packets that have been passed on to the next stage. The SMPM eliminates the need for implementing acknowledgement feature in the AUDTWMN. In order to confirm the validity of the proposed method, it has been first applied to the simulation experiment. The SMPM is again a circular memory of 8 words. The memory in the DTN collar nodes are kept low due to size constraints of the actual node. The Proposed node shall be a Microcontroller based device complete with onboard solar power system. It must be small enough to fit the cow with least annoyance.

C. Throwbox

Throwboxes [16] are designed to have:

1. High availability
2. Ability to transmit data at high data rates
3. Powerful enough to have sufficient processing to handle high data rates
4. Energy efficient

While many platforms exist to suffice for Throwbox requirements, the authors plan to use modified netbooks with special weather proof enclosures and solar power source as throwbox. The Triage platform used by Wenrui Zhao et.al is also an interesting candidate to be explored. The Triage platform and software consist of a coupled Stargate board and MicaZ mote. The Triage software reduces the amount of time that the device must spend operating the high-power Stargate board, running tasks on the MicaZ whenever possible. Using a hailing radio, passing nodes can trigger the always-on MicaZ to power on the Stargate board and 802.11 CF network interface. In this way, Triage never misses a passing node but remains energy efficient. Solar panels recharge batteries to ensure long-term use. Given the current technological advances in processor and storage the cost and size of such devices is expected to continually decrease, and energy efficiency to improve. Throwboxes are designed to augment DTN by virtue of:

1. Throwboxes are very effective in improving throughput and can also reduce data delivery delay. The improvement in throughput is generally more significant than improvement in delay.
2. Throwboxes are most useful for routing algorithms that use multi-path routing and when nodes follow structured mobility patterns
3. Throwbox deployment that incorporates knowledge about contact opportunities performs better than deployment that ignores this knowledge
4. Additionally, if deployment is customized to existing traffic patterns, the algorithms are more effective than assuming that traffic is equally distributed.

Throwboxes [16] TB1 to TB8 are lined along the roads running on both sides of the Cattlenet. TB1 to TB4 are laid on road from Ilayankudi towards Kalaiyarkoil. They are placed at intervals of 2.5 km starting from Ilayankudi. Similarly TB5 to TB8 are lined along the road from Ilayankudi towards Manamadurai. The Thowboxes again are spaced at 2.5 km. The throwboxes have circular memory of 32 words. They are equipped with IEEE 802.11 wi-fi with range of 30 meters. The Throwboxes do a data pull from the Cow nodes C1-C4 and a data push in to the bus node B2 that runs along that local loop. The Throwboxes are powered by Solar panels which trickle charge a storage battery. The Throwboxes are housed in tamper proof enclosures and placed atop poles. Poles help to prevent tamper by terrestrial beings and provide better Line of Sight range avoiding majority of land based obstacles.

III. SIMULATION CONSIDERATIONS

The Alunivdtnsim [4] is a C language based DTN simulator. It has been deigned to be simple and user friendly keeping in mind the needs of amateur researchers. The Alunivdtnsim is configured with the following simulation set up so as to closely mimic the AUDTWMN. The Alunivstnsim simulates the function and synthetic movement of its constituent agents namely

1. Sensor nodes
2. Data mules (bus and cattle) and
3. Data centre.
4. Throwbox (repeater)

A. Sensor Nodes

The sensor nodes (S1, S2, S3 and S4) simulate the motes that are laid in the water body along side the road. The sensor nodes are within 30 metres (radio range of 802.11b) of the road so that the data mules (buses) can receive the data. The sensor nodes in Alunivdtnsim sample a water parameter every 3 hours. The global tick is scaled to a second each and hence the sampling happens once every 10800 global ticks. The sensor node has a circular memory of size 16 bytes (8 words). Each message packet is an integer which is 2 bytes (1 word) long. The sensor readout function reads and ensembles the message packet from memory location word 1 to word 8 after which it overwrites word 1. The assumption is that the message is taken by the data mule (bus) before being over written. This assumption is valid by a comfortable margin because of the bus schedule followed by the Transport department. Other than collecting data once every 3 hours, the sensor node remains asleep most of the time in order to conserve battery. It is awaken by the presence of data mule which performs a data pull from the sensor node on radio contact.

B. Message Packet

The End Point Identifier (EID) is 3 bit long. Alunivdtnsim uses 5 EIDs from 001 to 101 to represent the nodes S1 to S5. As B1-B3, C1-C4 and D1 do not generate data, their EIDs are reserved, but not used. Message Identifier (MESSGID) is 7 bits long. It can be used to stamp 128 messages from each of 8 nodes. Since only 1 packet is generated every 3 hours, it amounts to only 8 packet a day and 56 packets a week. Since the Time to Live is 7 day max, the old packets will be purged out of the network. So 128 is good to hold messages generated for more than 2 weeks, there by ensuring no 2 different packets have the same EID+MESSGID combo at any given point of time inside of the whole network. The Message (MSG) is 3 bits long. This is the actual scientific data being collected from the water body. Say if we are collecting dissolved oxygen level from a lake, while 000 may represent the acceptable dissolved Oxygen level, all the other bits can be calibrated to represent a quantized value of dissolved Oxygen below the nominal value. Please note that we are not interested in knowing if the dissolved Oxygen level is above the nominal value as it is not of interest to us. The Time To Live (TTL) is 3 bits long. It assumes value of 111 when the packet is created representing the maximum value of 7 days to live. The TTL value gets decremented every 1 day and the packet is eventually purged when TTL becomes 000. Global tick again forms the basis to decrement the TTL every day.

C. Data Mule (Bus)

The buses B1, B2 and B3 form the data mules. Based on the route they ply (local loop or district loop), they have slightly varying degrees of functionality. B2 and B3 are local buses that ply the local loop. All 3 buses are from the Sivaganga depot and hence it forms the base terminus for their operation. Bus B2 starts from Sivaganga, travels to Manamadurai, Ilaiyankudi, Kalaiyarkoil and reaches back Sivaganga. It travels at 32 km per hour taking 5 minute stops at Manamadurai, Ilaiyankudi and Kalaiyarkoil. It however takes a 30 minutes stop at Sivaganga terminus before it resumes the same loop route all over. The bus travels 10 meteres per second of the global tick, thereby clocking 36 km per hour. The navigation of the bus in Alunivdtnsim is controlled by an Inertial Reference System (IRS), a technology used in Aircraft. It is also augmented by Global Positioning System (GPS) that confirms the co-ordinates of locations before making turns. The DTN node on board bus has a circular memory of 56 words. This is because of buses having better physical space and power source (alternator/battery) to handle more data unlike the sensor nodes. Whenever the bus B1 comes within radio range of sensor nodes S1, S2 or data mules C1, C2, C3 and C4 of the Cattle-net, on its routes, it performs a data pull operation of copying the message packets from sensor node on to itself. In exchange it purges those packets off the memory in senor nodes to avoid duplicates and to save the memory on sensor nodes. Hence at this point of time, the routing is of first contact [18] kind. If the memory in bus is full for some reason, the data pull operation does not over write but drops the packets instead. Before doing a data pull, the routing algorithm also ensures that the radio contact is fresh so as to avoid multiple contacts which result in wastage of battery power. The bus B3 operates similarly but in a different route. It starts from Sivaganaga, travels to Kalaiyarkoil, Karaikudi, Tirupatur and reaches back Sivaganga. It takes pit stops of 5 minutes each in Kalaiyarkoil, Karaikudi, Tirupatur while taking a 30 minutes stop over at Sivagana terminus before it resumes its loop route once again.

The bus b3 performs data pull from sensor nodes S3 and S4 that are en-route. The buses B2 and B3 have capability to exchange data as in flooding routing algorithm. However, this is an optional feature. The bus B1 operates slight differently by virtue of its district loop route. It travels at 20 meters per second of global tick clocking 72 km per hour. The bus travels point to point from Sivaganga to Madurai and back with 30 minutes stop at both places. The bus B1 initiates data pull from bus B2 and B3 whenever it comes in radio contact with them. Here again the radio range and freshness of contact are ensured before commencing packet transfer. It also has a circular memory of 56 words capacity. Only non-duplicate packets are fetched from B2 and B3 and written in free memory locations of B1. On successful transfer, the packet in B2 and/or B3 is purged in order to make room for new packets. An embedded TTL functionality monitoring the memory in buses performs a TTL check to decrement TTL and eventually purge expired packets. The bus B1 does not collect data from any sensor directly by design. The synthetic movements of the bus have been statistically modeled based on their operating schedules.

D.Data Centre

The data centre D1 at Madurai [24] keeps checking for radio contact with bus B1. Once in radio range, it establishes contact and on contact being fresh does a data pull from bus B1. The data centre D1 has a linear memory of 560 words. The data centre on successfully performing a data pull from bus B1, purges the packet off B1's memory in order to relinquish the space for new packets. The data centre has sufficient memory to store packets generated by all 5 sensor nodes for 2 weeks.

E. Disruption events

Of the many disruption possibilities, the authors have chosen only 2 scenarios as part of the preliminary design consideration. They are 1. Physical hiding of the DTN node by other un-known agents such as road vehicles. As the wireless portion of AUDTWMN uses IEEE 802.11 which operates at 2.4 GHz, it is a line-of-sight propagation. Hence any physical obstruction is going to cause disruption of the packet transfer. 2. Radio noise in the medium causing radio interference. Noise may be caused by various reasons such as lightening (AUDTWMN is an outdoor application), electric arcing (the roads are lined with power lines), and spark plug discharge in internal combustion gasoline engines (majority of vehicles are two-wheelers which have spark plugs). While noise may be spread across a wide frequency spectrum, its role in disruption increases significantly when it's amplitude is strong around the 2.4 GHz frequency (that of IEEE 802.11).

IV. SIMULATION RUN

The Alunivdtnsim is run typically run for 2 weeks, 1 day and 6 hours (1317600 Global ticks). The results of the simulations are captured in one or more data log file(s). The data logs record all parameters against the global tick as time reference. The results of the simulations are graphically analyzed using a tool called Gnu plot [2]. Gnu plot is a widely used open source program for plotting and visualizing data. Graphical analysis involves the following basic steps:

1. Plot the data.
2. Inspect it, trying to find some recognizable behavior.
3. Compare the actual data to data that represents the hypothesis from the previous step.
4. Repeat.

The data log generated by Alunivdtnsim can be used to carry out Data analysis and visualization. Some of these analyses include but not limited to

1. Graphical analysis
2. Presentation graphics
3. Control charts
4. Reality representation
5. Image analysis
6. Statistical analysis
7. Exploratory data analysis

The simulator is run with various combinations of operational parameters so as to analyze the performance.

TABLE I
THROUGHPUT OF SINGLE BEST THROWBOX

Used Single Throwbox	Number of message packets received at Data Centre
None Used	491
TB1	494
TB2	515
TB3	522
TB4	470
TB5	479
TB6	469
TB7	478
TB8	498

AUDTWMN uses first contact for the routing from source layer to hop layer1, from hop layer 1 to hop layer 2 and from hop layer 2 to the destination layer. However, the routing scheme from source layer to 0.5 hop layer and from 0.5 hop layer is set to (1) First contact [18] (2) Epidemic [22]. Also the pseudo-random generator is set to choose various 3 bits from amongst its integer string resulting in cow-director exhibiting interesting grazing patterns.

TABLE II
THROUGHPUT OF CUMULATIVE THROWBOXES

Used Cumulative Throwboxes	Number of message packets received at Data Centre
None Used	491
TB1	494
TB1,TB2	539
TB1,TB2,TB3	551
TB1,TB2,TB3,TB4	526
TB1,TB2,TB3,TB4,TB5	511
TB1,TB2,TB3,TB4,TB5,TB6	515
TB1,TB2,TB3,TB4,TB5,TB6,TB7	530
TB1,TB2,TB3,TB4,TB5,TB6,TB7,TB8	553

The authors use First Contact [18], Disrupted, Pseudo Random Bit Shift at 4 delivering the highest number of packets 491 of 560 as the default and go on to enhance the through put using the throwbox. The thowbox TB1 to TB8 for an intermediate hop layer 0.75 between the hop layer 0.5 and hop layer 1. The first set of simulation attempts to find the single throwbox which provides the highest possible throughput. The results as tabulated in Table I.

The second set of simulation attempts to find the cumulative effectiveness of throwboxes demonstrating the increased throughput. The results as tabulated in Table II.

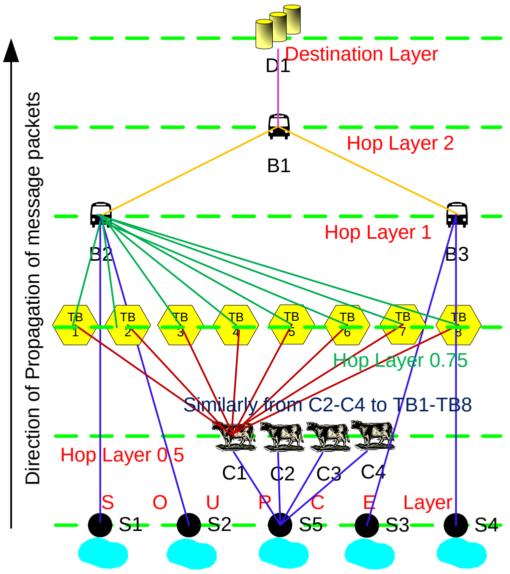

Fig. 3 AUDTWMN represented as Graph

V. SIMULATION RESULTS

Eight Throwboxes were selectively included or excluded in individual and collective manner yielding 2 sets of simulation results that have been tabulated. In individual Throwbox, the highest throughput of 522 message packets gets delivered when TB3 is deployed. Out of 112 message packets generated by S5, 74 are delivered at datacenter. This amounts to 66% of the message packets of S5 as against 38.4% without Throwboxes. It is also interesting to observe that deploying TB4,TB5,TB6 or TB7 decreases the throughput, an example where Thorowbox works counter-productive.

In the cumulative category, the highest throughput of 553 message packets gets delivered when all 8 Throwboxes TB1-TB8 are deployed. Out of 112 message packets generated by S5, 105 are delivered at datacenter. This amounts to 93.7% of the message packets of S5 as against 38.4% without Throwboxes. It is also interesting to observe that linear increase in number of Throwbox may not bring a linear increase in Throughput. Hence TB1+TB2+TB3 may be better than TB1+TB2+TB3+TB4 or TB1+TB2+TB3+TB4+TB5 or TB1+TB2+TB3+TB4+TB5+TB6 or TB1+TB2+TB3+TB4+TB5+TB6+TB7. From cost perspective, the best value for money seems to be adding that one individual Throwbox that gives additional throughput. Although, given the random grazing pattern of the Cattlenet, multiple Throwboxes ensure that interesting message packets do not get lost.

The rest of this section presents the graphical output of Alunivdtnsim's data log generated by gnu plot [2] for highest cumulative throughput as example for analysis. The total simulation time is divided into 2 halves for ease of presentation. The output progresses along with the propagation of the message packets from source layer to the destination layer passing through the intermediate hop layer.

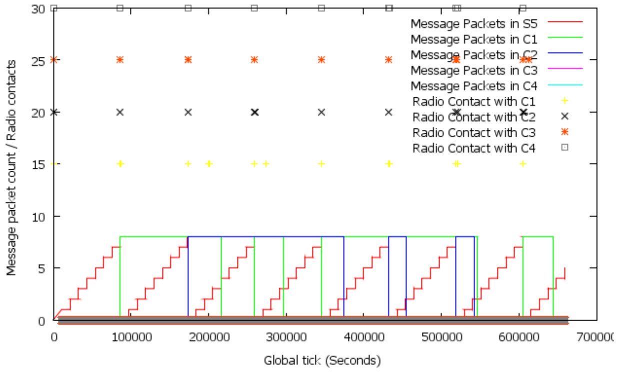

Fig. 4 Message Packet counts in S5,C1-C4 and Radio Contact of C1-C4 with S5. (First half of Simulation)

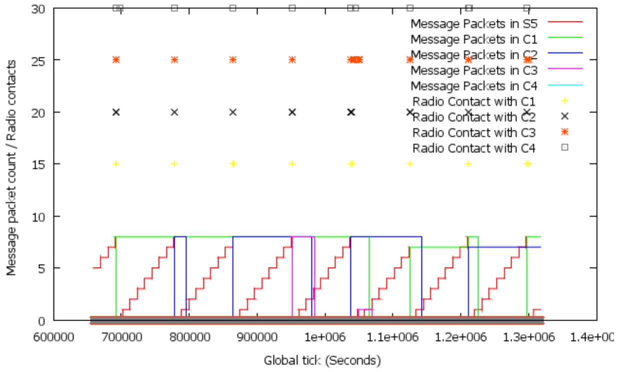

Fig. 5 Message Packet counts in S5,C1-C4 and Radio Contact of C1-C4 with S5. (Rest of Simulation)

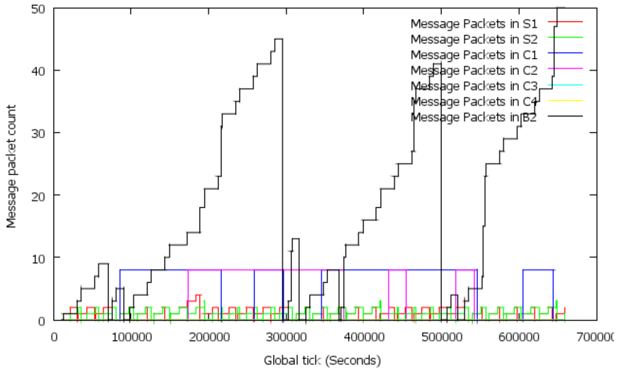

Fig. 6 Message Packet counts in S1, S2 and C1-C4. (First half)

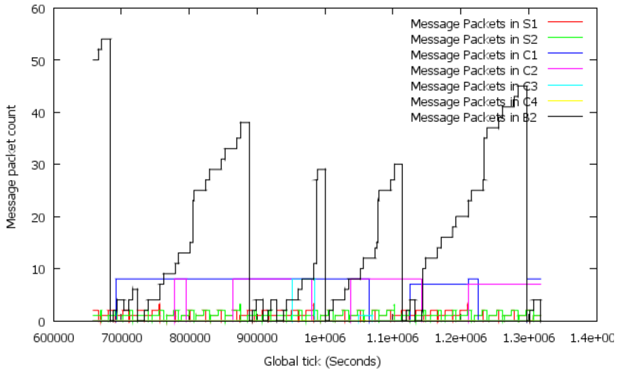

Fig. 7 Message Packet counts in S1, S2 and C1-C4. (Rest of Sim)

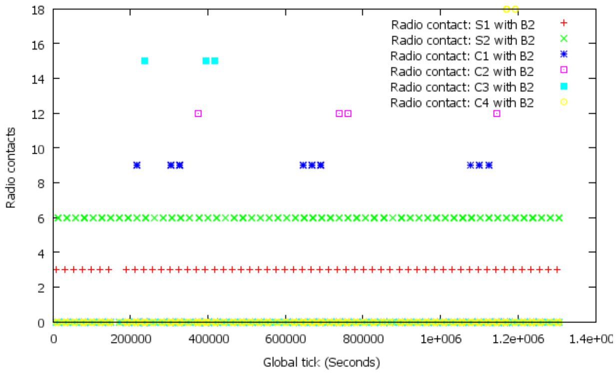

Fig. 7 Radio contacts in S1, S2, C1-C4 and B2. (Full Simulation).

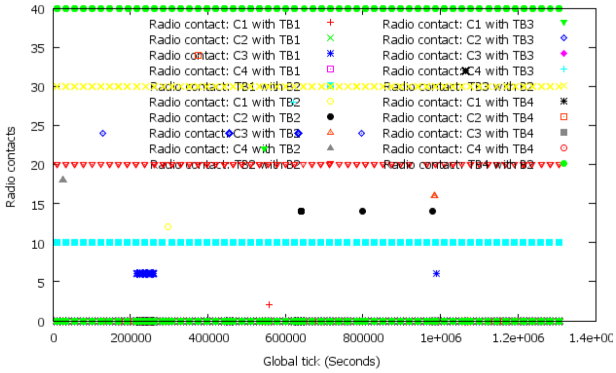

Fig. 9 Radio contacts of TB1-TB4 with C1-C4 and B2. (Full Sim)

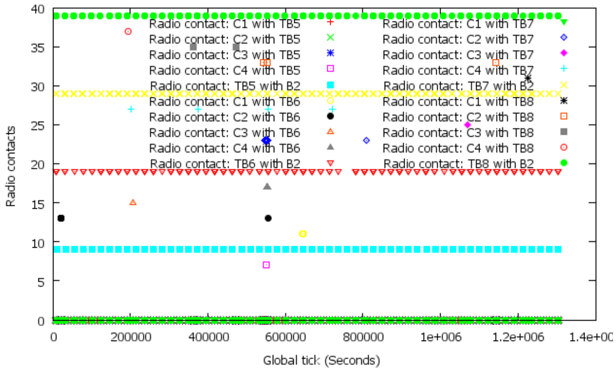

Fig. 10 Radio contacts of TB5-TB8 with C1-C4 and B2. (Full Sim)

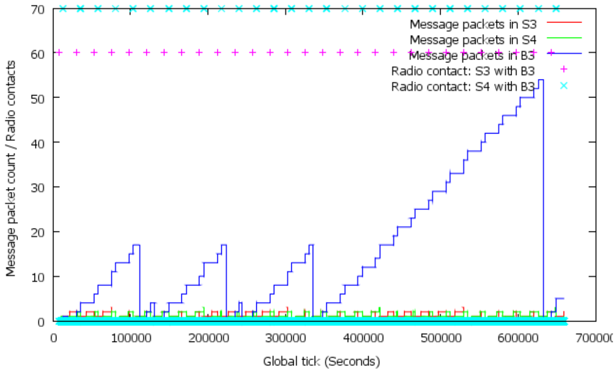

Fig. 11 Message Packet counts in S3, S4 & B3 and Radio Contact of S3, S4 with B3. (First half of Simulation)

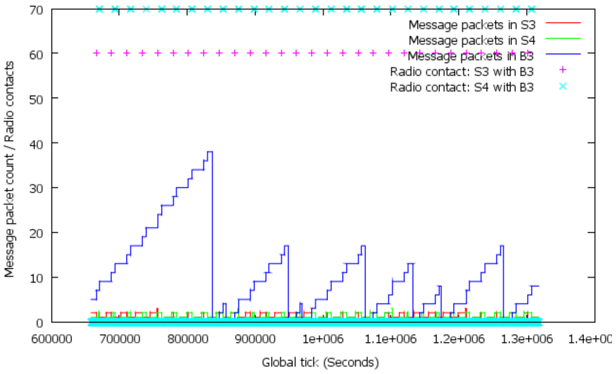

Fig. 11 Message Packet counts in S3, S4 & B3 and Radio Contact of S3, S4 with B3. (Rest of Simulation)

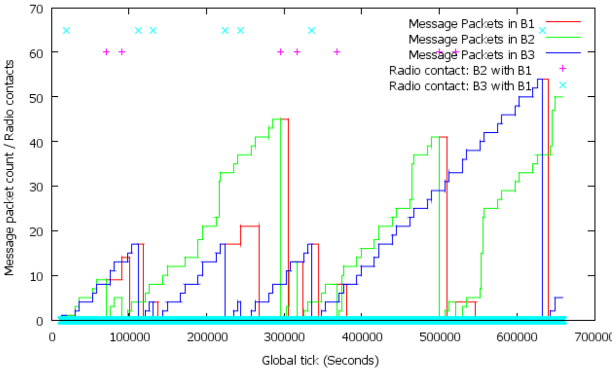

Fig. 12 Message Packet counts in B1-B3 and Radio Contact of B2, B3 with B1. (First half of Simulation)

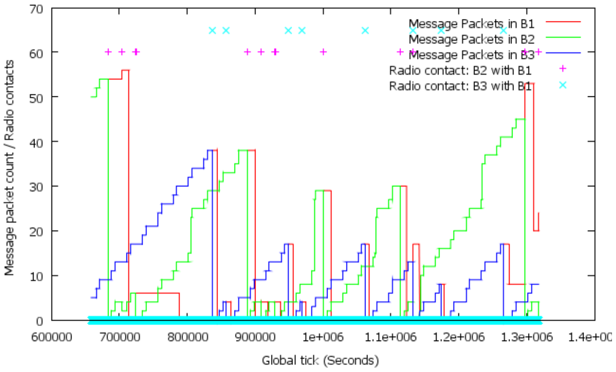

Fig. 13 Message Packet counts in B1-B3 and Radio Contact of B2,B3 with B1. (Rest of Simulation)

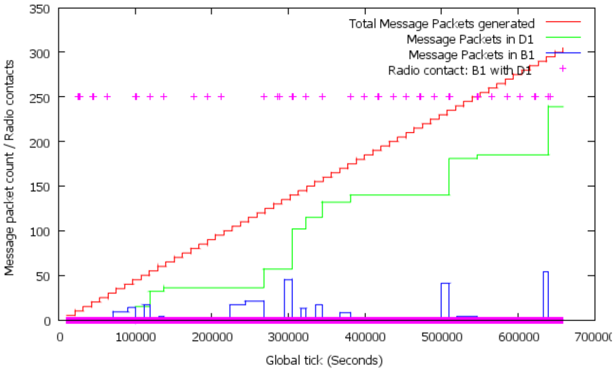

Fig. 14 Message Packet count in B1,D1 and Radio Contact of B1,D1.

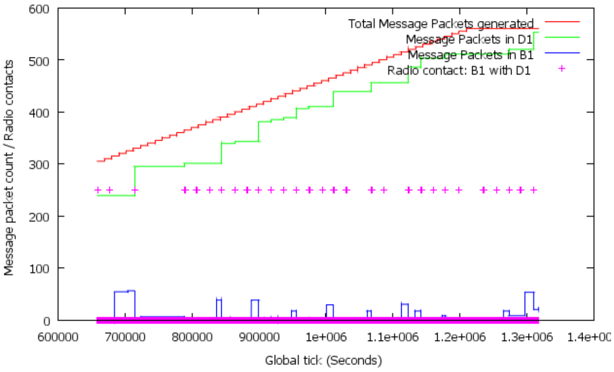

Fig. 15 Message Packet counts in B1, D1 and Radio Contact of B1&D1. (Rest of Simulation)

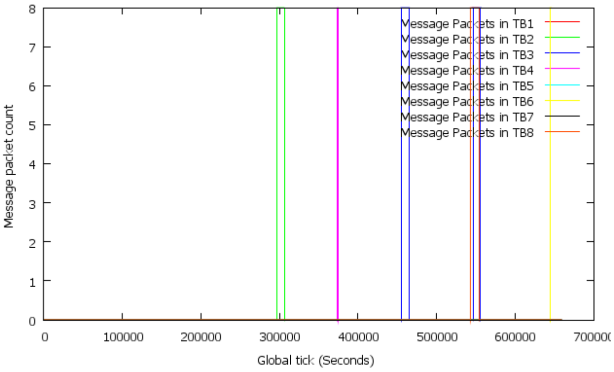

Fig. 16 Message Packet counts in TB1-TB8 (First half of Simulation)

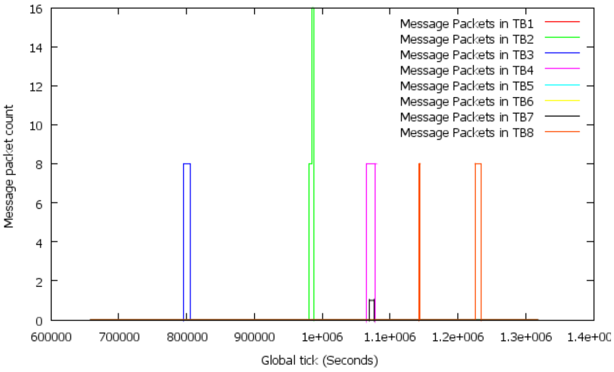

Fig. 17 Message Packet counts in TB1-TB8 (Rest of Simulation)

VI. CONCLUSION

The enhancement of AUDTWMN using Throwboxes has been presented. Individual and cumulative use of Throwboxes has been simulated and both the set of simulation results has been presented. A maximum throughput of to 93.7% has been observed in simulation by using Throwboxes as against 38.4% of the S5 message packets without Throwboxes. Future scope of work includes:

1. Careful study of the non-linear throughput increase/decrease against a linear increase of the Throwboxes count using Network flow algorithms.
2. Exploring other techniques [15] to achieve a 100% throughput of the message packets generated by S5.
3. Testing the delivery performance using other DTN routing [19] schemes such as (a) Spray and Wait [17], (b) PRoPHET [21] and (d) MaxProp [14].

ACKNOWLEDGMENT

The authors would like to express their cordial thanks to Dr. Manikantan Ramadas, Er. Raghu Raman for their valuable advice and support.

REFERENCES

[1] Laveen Sundararaj, Palanisamy Vellaiyan. “An Overview of Alagappa University Delay Tolerant Water Monitoring Network”. International Journal of Computer Science and Network Security, Korea, Volume 10, Number 5, May 2010. ISSN: 1738-7906.

[2] Philipp K. Janert, ‘Gnuplot in Action - Understanding data with graphs”, Manning Publications, 2010.

[3] Prof. Shie-Yuan Wang, Chih-Liang Chou, and Chih-Che Lin, “The GUI User Manual for the NCTUns 6.0 Network Simulator and Emulator”, Network and System Laboratory, Department of Computer Science, National Chiao Tung University, Taiwan, 2010.

[4] Laveen Sundararaj, Palanisamy Vellaiyan, “Alunivdtnsim – An Overview of a simple Delay Tolerant Network simulator”, National Conference on Information, Communication & Networking, SRM EEC Chennai, April 2010.

[5] Laveen Sundararaj, Palanisamy Vellaiyan, “DTN Work Update”, DTNRG meeting March 2009, a NASA event, Google HQ, California, United States of America.(http://down.dsg.cs.tcd.ie/dtnrg-at-google/)

[6] Ari Keränen,Jörg Ott, Teemu Kärkkäinen (TKK), "The ONE simulator for DTN protocol evaluation",SIMUTools 2009, Rome, Italy.ISBN:978-963-9799-45-5.

[7] Teerawat Issariyakul, Ekram Hossain, “Introduction to Network Simulator NS2” , Springer Science Media, 2009, ISBN: 978-0-387-71759-3.

[8] Kevin Fall, Senior Member, IEEE, and Stephen Farrell, "DTN: An Architectural Retrospective", IEEE Journal on Selected Areas in Communications, Vol.26, No 5, June 2008.

[9] K. Scott and S. Burleigh, “Bundle Protocol Specification,” Internet RFC 5050, Nov 2007.

[10] Laveen Sundararaj, Palanisamy Vellaiyan, “Planned DTN Work”, DTNRG meeting May 2007, Trinity College Dublin, Republic of Ireland. (http://down.dsg.cs.tcd.ie/misc/DubDTN/)

[11] V. Cerf, S. Burleigh, A. Hooke, L. Torgerson, R. Durst, K. Scott, K. Fall, and H. Weiss, “Delay–Tolerant Networking Architecture,” Internet RFC 4838, April 2007.

[12] S. Farrell and V. Cahill, Delay– and Disruption–Tolerant Networking. Artech House Publishers, 2006, ISBN: 1-59693-063-2.

[13] Stephen Farrell, Vinny Cahill, Dermot Geraghty, Ivor Humphreys and Paul McDonald, “When TCP Breaks, Delay- and Disruption-Tolerant Networking", IEEE Internet Computing July-August 2006.

[14] J. Burgess, B. Gallagher, D. Jensen, and B. N. Levine, “MaxProp: Routing for Vehicle-Based Disruption-Tolerant Networks”, Proceedings of IEEE Infocom, April 2006.

[15] Zhensheng Zhang, “Routing in intermittently connected mobile ad hoc networks and delay tolerant networks: overview and challenges” Communications Surveys & Tutorials, IEEE, Volume 8, Issue 1, First Quarter 2006 Page(s):24 – 37.

[16] Wenrui Zhao, Yang Chen, Mostafa Ammar, Mark Corner, Brian Levine, Ellen Zegura, "Capacity Enhancement using Throwboxes in DTNs",In Proc. IEEE Intl Conf on Mobile Ad hoc and Sensor Systems,2006.

[17] Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S. Raghavendra, “Spray and Wait: An Efficient Routing Scheme for Intermittently Connected Mobile Networks”, Proceedings of ACM SIG-COMM Workshop on Delay-Tolerant Networking (WDTN), 2005.

[18] Thrasyvoulos Spyropoulos, Konstantinos Psounis, and Cauligi S. Raghavendra, “Single-copy routing in intermittently connected mobile networks”, Proc. Sensor and Ad Hoc Communications and Networks SECON, pages 235-244, October 2004.

[19] Sushant Jain, Kevin Fall, and Rabin Patra, “Routing in a delay tolerant network”, SIGCOMM '04: Proceedings of the 2004 conference on

Applications, technologies, architectures, and protocols for computer communications, pages 145-158, New York, NY, USA, 2004. ACM.

[20] K. Fall, “A Delay-Tolerant Network Architecture for Challenged Internets,” in Proc. ACM SIGCOMM ’03. New York, NY, USA: ACM Press, 2003, pp. 27–34.

[21] A. Lindgren et al., “Probabilistic Routing in Intermittently Connected Networks”, Mobile Comp. and Commun. Rev., vol. 7 no.3, July 2003.

[22] A. Vahdat and D. Becker, “Epidemic routing for partially connected ad hoc networks”, Technical Report CS-200006, Duke University, April 2000.

[23] "The Environment Profile for Sivagangai district" published by Environmental Information System Centre, Department of Environment, Government of Tamil Nadu, Sponsored by Ministry of Environment and Forest, Government of India. http://www.tnenvis.nic.in/DtProfiles/sivagangai.pdf

[24] "Surface Water Quality Monitoring Stations – Tamil Nadu" published by State Ground and Surface Water Resources Data Centre, Water Resources Department, Government of Tamil Nadu. http://www.groundwatertnpwd.org.in/

Laveen Sundararaj received the B.E., degree from Bharathiar University in 2001. While working at Honeywell India, he also received the M.Sc., and M.Phil., degrees from Alagappa University in 2005 and 2008 respectively. He has worked at Honeywell Aerospace, Olathe, USA from 2004 to 2005. He has also worked at Honeywell Aerospace, Phoenix, USA, from 2008 to 2009. He is currently a Honeywell research scholar at Alagappa University, Karaikudi. His research interests include Digital Image Processing and Ad-hoc Networks. He is a member of the DTNRG and has given tech talks at DTNRG 2007, Trinity College Dublin, Ireland and DTNRG 2009, a NASA event at Google HQ, USA.

Dr.Palanisamy Vellaiyan received the B.Sc., degree from Bharathidasan University in 1987. He also received the M.C.A., and Ph.D., degrees from Alagappa University in 1990 and 2005 respectively. After working as Lecturer in AVVM Sri Pushpam College, Poondi from 1990 to 1995, He joined Alagappa University as Lecturer in 1995. He is currently the Head of Department of Computer Science and Engineering. He also received the M.Tech degree from Bharathidasan University in 2009. His research interests include Computer Networks & Security, Data mining & Warehousing and Algorithms. Since 2008, he is also the Chair person of the School of Computer Science, Alagappa University, Karaikudi, India.