An IP address Space Management Model for Wireless Ad-Hoc Networks

Murimo Bethel Mutanga
University of Zululand
Department of Computer Science
KwaDlangezwa, South Africa
bethelmutanga@gmail.com

Tarirai Chani
Monash University
School of Information Technology
South Africa
tarirai@monash.ac.za

Abstract—Wireless technologies are continuously evolving to provide better services with lower setup and running costs. Wireless ad-hoc networking, in particular, has proven to be a promising technology in many application scenarios in today’s world. Manual configuration in these networks is highly impractical hence automatic configuration of IP addresses and other related network parameters is of paramount importance. Many IP address auto-configuration mechanisms have been proposed in literature. These mechanisms can be classified as either being stateful or stateless. Hybrids of stateful and stateless protocols also exist in literature. The purpose of having an address auto-configuration protocol is to manage the IP address space and configure nodes with unique addresses. IP addresses come from a finite domain but due to the management of these addresses is not an easy task. The current approaches fail to address challenges posed by the unpredictable nature of wireless ad-hoc networks. In this paper we proposed an adaptive model to the management of IP addresses. We argue that building components and other auto-configuration parameters should adapt to the changes in the network environment. Adaptation will aid in effective management of the IP address space.

Key words: auto-configuration, wireless ad-hoc networks, Duplicate address detection

1. INTRODUCTION

The concept of self organization is a phenomenon that has its roots in biological and eco-systems is rapidly being adopted in many areas, including computer networks, engineering etc. The field of wireless ad-hoc networks is a typical example of the emergence of self-organized functions. Because of the wide adoption of this technology, it is envisaged that this trend will be accelerated even further due to the rapid proliferation of ubiquitous computing. The need for advanced techniques in self-management will thus become an unquestionable requirement. The complexity in the setup and management of networks, might become a limiting factor for further advancement. It is unambiguous that self-organization will help researchers, network planners and administrator to address these challenges [1]. Actually, self-organization will aid to ease the administration efforts and reduce operation costs considerably.

The autonomous nature of wireless ad-hoc networks requires the deployment of an IP address auto-configuration mechanism to manage the IP address space and configure nodes with IP addresses. It has been argued that the IP address auto-configuration problem can be solved simply by constructing a unique address from the Medium Access Control (MAC) address. However, a big concern with this solution is the question of location privacy [2]. Random generation of IP addresses from a specific range has thus become the defacto solution. However, such a solution has the challenge of coping with a highly dynamic environment and uncertain network conditions in wireless ad-hoc networks [3]. IP address auto-configuration schemes must manage the IP address space in an efficient way. Address duplicate must be avoided. IP addresses come from a finite pool hence there is need to manage them properly. An IP address is one of the most important network parameters. Network nodes cannot take part in unicast communication if they do not have valid and unique IP addresses. Due to the unique characteristics of wireless ad hoc networks, the mechanisms used by infrastructure-based networks cannot be directly applied to manage the IP address space.

The centralized approach is used in wired networks is not practical in ad-hoc hence all the nodes have to collectively manage the address space. The wireless ad-hoc network environment presents challenges that make is difficult to address the issues mentioned above. Conditions in the network change rapidly and membership of the network can also change anytime. The configuration process should manage the address space and guarantee address uniqueness whilst controlling communication overhead and latency. However, achieving low address duplicates is usually associated with high communication overhead and high latency. There are various issues that need special attention when designing and implementing IP address auto-configuration protocols for wireless ad-hoc networks. These include, handling network merging, detecting duplicate addresses, securing the configuration process etc. More control packets are usually used to detect duplicate addresses thereby resulting in high communication overhead.

Many address auto-configuration solutions have been proposed in literature under the umbrella categories of stateless and stateful approaches. These solutions suffer from rigidity; they are not adaptive and they do not attempt to make tradeoffs. For example, the main challenge of protocols utilizing a stateful approach with a distributed address allocation table e.g. ManetConf [4], is reliable state synchronization given the uncertain environments that characterize ad-hoc networks [2]. Whatever changes that
take place at a microscopic level should result in the desired global behaviour hence the need for state information coordination. The cost of explicit coordination is usually higher than the benefits. As the network size increases, explicit coordination can be a challenge and is normally coupled with high communication overhead although it helps in achieving the desired global property of unique IP addresses. The frequency of state synchronization is also one issue that has not yet been addressed in literature. In a fairly stable network, frequent synchronization might result in unnecessary traffic. It is thus imperative to devise adaptive mechanisms to handle state synchronization. In this work we argue that the frequency of state synchronization packets must be decided at runtime. Network conditions must determine how frequent the state information is synchronized.

In stateless approaches eg AROD [5], AIPAC [6] the most important aspect is the design of the Duplicate Address Detection (DAD) procedures. Stateless protocols employ a Duplicate Address Detection (DAD) mechanism during the auto-configuration process. Using this mechanism, new nodes generate their own IP address and broadcast a request packet and set a timer (DAD timeout). When the DAD timeout expires before any node using the requested IP address responds, the new node configures itself. A long period might result in unnecessary delays whilst, a short delay may result in duplicate addresses [7]. The optimal value of DAD for any given network condition is an issue that most protocols in literature do not consider. Short DAD period that is applicable in small networks might not be applicable in large networks. Some solutions resort to repeating DAD for two or three times to guard against DAD trials is an issue that needs further investigation. A few DAD trials are suitable for small networks whilst the same cannot be said about large networks. It is therefore imperative for protocols to adapt to network changes.

The main contribution of the model proposed in this work is the use of network conditions to determine the configurations of the auto-configuration protocol at runtime. The rest of this paper is organised as follows: In section II we discuss some research issues in IP address auto-configuration whilst in Section III, related approaches in the area of IP address auto-configuration are given. Section IV describes our IP address space management model. Finally, Section V concludes this paper and discusses possible future studies.

II. DESIGN ISSUES IN IP ADDRESS AUTO-CONFIGURATION

A protocol for assigning IP addresses in wireless multi-hop networks should meet the following requirements:

A. Network merging and partitioning

IP address auto-configuration protocol must have the capabilities of handling network partitioning and merging. Duplicate addresses might arise if two different partitions merge. Duplicate IP addresses should be detected quickly and resolved.

B. Secure Auto-configuration

Secure auto-configuration must be guaranteed. Wireless networks do not have secure boundaries hence it is difficult to address security and authentication issues. Most auto-configuration protocols proposed in literatures pays less attention to security during auto-configuration. Mechanisms to detect security threats must be implemented in auto-configuration protocols. Security monitoring mechanisms are supposed to be proactive hence they must actively monitor the network conditions for possible attacks.

C. Duplicate Address Detection

An auto-configuration protocol must be able to detect duplicate addresses. A duplicate address detection mechanism is required as continuous process to guard against duplicate addresses caused by erroneous allocation of duplicate addresses. Mechanism for detecting duplicate addresses must be implemented within the auto-configuration protocol. This can be done by analysing routing protocol information for hints that can point to the existence of a duplicate address.

III. CURRENT APPROACHES

A lot of IP address assignment algorithms have been proposed in literature. The mechanisms can be classified according to the way they manage the IP address space. There are basically two main categories, namely, stateful and stateless. The stateless approach does not employ a mechanism of managing the IP address space. The number of free IP addresses are not known hence when new nodes join the network, then choose a random IP address and check for availability through a DAD procedure. On the other hand, stateful approaches employ one or more nodes to manage the IP address space.

A. The Stateless paradigm

Stateless protocols free IP addresses are not known in advance because address allocation tables are not kept. All the network nodes collectively manage the IP address space by participating in duplicate address detection. New nodes generate their own IP addresses from an allowed range and check for possible conflicts. If an address conflict is detected, the new node will repeat the process until a free address is obtained. The process of verifying the uniqueness of the address is called a Duplicate Address Detection (DAD) procedure. Generally the DAD process is categorized as being either Strong DAD [8] or Weak DAD [9]. Weak DAD makes use of a key-address combination that must always match if there is no address conflict. Nodes analyse routing protocol packets for signs of address conflicts. Strong DAD is a time-based DAD that checks if there is an address conflict in a network within a finite bounded time interval. Strong DAD configures nodes after the DAD procedure has been successfully completed or after a specific time interval called a DAD timeout period.
AIPAC [6], AROD [5] and the scheme proposed in [4] and are based on Strong DAD. In [4], a new node chooses two addresses, a temporary address and the actual address to use. The temporary address is used only once during the address negotiation phase. The network is then flooded with an address request packet containing the actual address. A node that uses the requested IP address an address reply message to defend its address. If no AREPs are received by the originator after a certain time interval and after multiple tries, the node concludes it can use the chosen address. In AIPAC[6], a new node periodically broadcasts a request message until a reply is received from at least one neighboring node (initiator). The initiator then performs DAD on behalf of the new node.

AROD [5] extends Strong DAD [9] by including address reservation in as a mechanism to reduce the communication overhead. The authors argue that it is difficult to guarantee uniqueness of allocated addresses without performing a DAD thus they proposed a distributed auto-configuration scheme that uses address reservation and optimistic DAD. Reserved addresses were introduced to help reduce allocation latency and the DAD mechanism guarantees the uniqueness of address with much smaller communication overhead than traditional DAD approaches.

Stateless approaches are prone to duplicate IP addresses because of the unpredictable nature of ad-hoc networks. When network size increase, number, the probability of a failed DAD also increases hence resulting in delay and communication overhead. Determining the parameters of DAD like the timeout period, is an issue that needs investigation. A static value might not be the best since network conditions are not static. It is therefore imperative to employ adaptive mechanisms to respond to network conditions.

B. Stateful Auto-configuration

There are many variations of stateful auto-configuration but the basic concept is that there is at least one node that is responsible for managing the IP address space. When new nodes join, the nodes managing the IP address space can easily issue free IP addresses since they are known in advance. To guard against address leakages, nodes that run stateful auto-configuration synchronise the address allocation tables. Stateful approaches following the stateful paradigm can be further classified according to the way they manage the IP address allocation table. The address allocation table can either be centralized or distributed. In case of distributed allocation table, there are two alternatives: Distributing a common table managed by all the nodes or distributing multiple disjoint allocation tables where each node manages its own pool of IP addresses.

Auto-configuration solutions that use a centralized allocation table must guarantee that the central node has up-to-date state information to avoid address leakages and conflicts. Node departures and arrivals should be reflected promptly. In CAC [10], a central node called the address agent periodically broadcasts verify-packets which contain the address-list and a time stamp. Every node checks whether or not it is included in the address list. Traffic to the central node must be well managed so that it does not get overloaded since it is the only one with the responsibility of managing the IP address allocation table. If the address agent is temporarily unavailable, there must be a mechanism of selecting a new AA. In CAC, if a node does not receive verify packets from the address agent anymore, it assumes that the network is partitioned and elects itself as the new AA.

MANETConf uses Distributed Address Allocation Table ie all nodes in the network keep a list of IP addresses that are currently in use. The management of the IP address space is thus distributed to all the nodes. If a node (Requester) wants to join the network, it has to rely on an already configured node. Then the Initiator selects a free IP address from the address allocation table, and checks for its availability through a DAD procedure.

Prophet [11], uses a novel approach that follows the stateful paradigm but does not make use of an IP allocation table. The basic idea behind Prophet is to predict the allocation table using a function f(n) that is distributed on all the nodes. The first node in the network chooses the function parameters. As other nodes join the network, the function f(n) and a state value to generate IP addresses are passed on to them so that they can also allocate new node with IP addresses. In this approach, the IP address space is known by all nodes in the network.

The proposal in [12], 2008 uses a distributed address assignment system that uses address coordinators as the agents for configuration of IP addresses. New nodes listen for a while for hello messages coming from the closest coordinator. The coordinator stores a pool of free IP addresses that is uses to configure the new node before it releases half of the addresses to the new node. A new node can also become a coordinator if it is more than two hops from its own coordinator. If a node wants to leave the network, it leases its IP address(es) to the nearest coordinator. In case of abrupt disconnection of nodes, state synchronization is necessary. Each node periodically broadcasts a message to the first coordinator (c-root).

The main challenge of stateful approaches is the design of reliable state synchronization mechanisms. Frequent state synchronization messages result in high communication overhead yet it helps to reduce duplicate addresses. On the other hand reducing the frequency of state synchronization might result in less communication overhead and increase in address duplicates. Fairly stable networks might not require frequent state synchronization.

C. Hybrid Approaches

Hybrid auto-configuration solutions combine characteristics stateful and stateless approaches to manage the IP address space. These protocols combine DAD with an address allocation table that is either centrally

In Wise-DAD [16], all the nodes passively collect state information but still performs DAD when a new node wants to join the network. An unconfigured selects one of its neighbors node to act as its negotiating agent (initiator). The initiator then generates a random IP address from the allowed addresses and checks its allocation table if there is no node in the network that have requested for or used the same IP. If the address is not known, the initiator then performs a DAD (using an address request message). All nodes receiving an address request packet update their tables and add their IP addresses to the packet before broadcasting it. If any node is using the requested address, it defends it with an IP conflict message and this process is repeated. If no IP conflict message is received after a certain time interval, the address is assumed to be free and the initiator will send an address reply message to the new node. The address reply message will have the IP address for the new node, the network identifier and the state information (allocation table). If a node leaves the networks gracefully, it broadcasts a goodbye message and all the nodes delete its IP address from their allocation tables. If a node leaves abruptly, immediate address reclamation is not performed. Since the node will not be sending or forwarding any data packets, other nodes will remove all passive nodes from their allocation tables. Allocation tables are not actively synchronized; they are used only as an estimate of the state information. If a node does not take part in an IP address allocation process for a long time, its IP address will be deleted when the size of the allocation table reaches a certain level because it will be assumed that the node left the network abruptly.

IV. THE NEW IP ADDRESS SPACE MANAGEMENT MODEL
It is clear that there is no single approach or protocol that is exceptionally superior to the other. This is so because in meeting all the design criterion for an IP address auto-configuration protocol, the following performance metrics goals usually contradict each: low latency, high probability of address uniqueness and low communication overhead. For example, the best way of making sure that the allocated IP address is unique is to perform DAD, but on the other hand, the best way to avoid high communication overhead is to eliminate or avoid performing a DAD procedure. The ideal situation is getting maximum benefits (desirable characteristics) while keeping the costs (undesirable properties) associated with attaining those conditions as low as possible. Therefore the design must consciously make tradeoffs between these contradictory factors. Schemes proposed in literature suffer from rigidity; they are not adaptive and they do not attempt to make tradeoffs between these contradictory factors. Optimally choosing the best parameters and organizing them to achieve the overall goal of automatic configuration leads us an optimization problem which is a difficult problem to solve given uncertain conditions in ad hoc networks. To address these challenges, we propose an adaptive paradigm. We argue that the building blocks of IP address auto-configuration protocols must adapt to network conditions. This is motivated by the work in [1] that gives general guidelines on self-configuration. In this work we adapt the model in [1] specifically for IP address auto-configuration.

A. IP addresses should be distributed to all nodes
We need to explicitly define a mechanism of delegating the responsibilities of assigning IP addresses to all the nodes in the network. There is also a need to explicitly define how this decentralized system will be managed. i.e. Localized behavior rules or functions that, if applied in all nodes at a microscopic level (within their local neighborhoods), automatically lead to the desired network behavior at a macroscopic level. Network nodes must have only a local view of the network and interact with their neighbors as much as possible whilst the whole network follows the desired global property. This will reduce both allocation time and minimize communication overhead since communication will be done locally. We need to define the following building blocks of a framework that follows the adaptive paradigm (Figure 1).

i. Mechanisms, rules or functions for managing the local IP addresses. These rules can be implemented as functions that govern the behavior of individual nodes in their own neighborhoods. The behaviour of nodes should yield the desired global properties if applied consistently. It must be clear how the IP addresses will be managed locally without adversely affecting the global pool of addresses.

ii. Functions or rules governing the delegation of the responsibilities of assigning IP addresses to all the network nodes in such a way that the rules defined in (i) above can be applied with ease. The responsibility of address allocation can be assumed by either all or a set of selected nodes. Central allocation results in delays and high communication overhead.

B. State information should not be explicitly coordinated or synchronized
State information should not be explicitly coordinated because of the high communication overhead that may result. We define three rules for managing state information synchronization:

i. All possible states and how they affect network behavior must be defined. Mechanisms of how to respond to each state can then be defined to react to the identified state changes.

ii. The level of state information inconsistencies that can be tolerated must be defined. When the network increase in size, coordination can be very difficult. Also due to the unpredictable nature of
ad hoc networks, coordination can be bandwidth consuming if network conditions change frequently.

iii. State information must be passively synchronized. Mechanisms of passively obtaining state information must be defined. This can be achieved by using routing protocol control packets such as hello messages.

C. Adapt to changes

Due to the ad-hoc nature of multi-hop networks, the environment in which the nodes operate may change unexpectedly. An IP address auto-configuration protocol should adapt to different triggers to change. This will increase robustness of IP address allocation schemes. The desired performance of an IP address allocation scheme is measured at a macroscopic level hence there is need to monitor if the distributed address management is achieving the desired global goal. To avoid address leakages, acceptable levels of non-coordination defined in (b) above needs too be monitored and corrective action needs to be taken if need arises.

Other scenarios that require protocols to employ a monitoring mechanism include the merging of two or more independently configured networks, network partition and the exhaustion of local IP addresses. All unforeseen occurrences need to be monitored and corrective action taken.

We propose that an IP address protocol following this approach should have the following functions for the purpose adapting to different triggers for change.

i. A performance monitoring mechanism that monitors the changes that might require corrective action. This can be implemented using monitoring algorithms or functions that can either be proactive or reactive in nature. Increase in address allocation latency, communication overhead, network merging, security threats, address duplicates etc are some of the things that should be monitored. We propose that the design of the monitoring mechanisms must be bandwidth conscious so as to minimize communication overhead. For the network to be able to respond to changes, the protocol must first detect the changes.

ii. A response management strategy that takes action should a change that requires nodes to behave differently is detected by mechanisms employed in (i) above. The nodes should take action that is relevant to the changes observed. A DAD procedure should also be part of an IP address auto-configuration scheme. It can be defined under the adaptation rules to guard against erroneous address allocation. If the current settings or configurations of the protocols are no longer yielding the desired results, We need to determine the optimal values that will give the desired results e.g. the optimal value of the DAD timeout period and the number of DAD trials. Network conditions might change anytime hence these values must be calculated based on the network conditions. Setting static values will not result in an optimal protocol since network conditions can change at any time.

Figure 1. The Adaptive Model

V. CONCLUSION AND FUTURE WORK

The advent of wireless networking has significantly reduced the costs of setting up computer networks. Wireless ad hoc networks in particular have the potential to expand but a lot of research is still needed to realize this dream. Automatic configuration of IP addresses in such networks greatly reduces administration efforts and costs. Given the constraints in the wireless ad-hoc networking environment, managing the IP address space is not an easy task. In this paper we presented an approach of managing the IP address space in Wireless Ad-hoc networks. In this approach we propose the distribution of IP addresses and went on to propose local management of these addresses. Protocols following this paradigm are likely to be more robust than both stateless and stateful approaches. The biggest research challenge is coming up with the best building components for constructing the monitoring and adaptation modules. We hope that our contributions will stimulate further research in this direction. The future focus of this work will be on designing protocols that follow this paradigm.

ACKNOWLEDGEMENTS

The authors appreciate the support given to the Department of Computer science at the University of Zululand by industry partners.

REFERENCES


