

BRICS: A Building-block approach for analyzing Routing protocols in ad hoc networks - a Case Study of reactive routing protocols

Fan Bai, Narayanan Sadagopan, Ahmed Helmy
Department of Electrical Engineering, University of Southern California
{ fbai, narayans, helmy }@usc.edu

Abstract—One of the main challenges in ad hoc networks research is understanding the effect of mobility on the performance of routing protocols. In a previous study, we have shown *Why* mobility affects the performance of routing protocols. In this study, we further extend our approach to analyze the interplay between mobility and the protocol mechanistic building blocks. Through this approach we hope to explain *How* performance varies with mobility by decomposing the protocol into parameterized mechanistic building blocks based on their functionalities. Then, we apply this approach to reactive MANET routing protocols like AODV and DSR, which enables us to build a common building block architecture that encompasses these reactive protocols. The effect of mobility on each building block is evaluated. We are specifically interested in understanding the contribution of each building block to the overall protocol performance. Through simulations, several lessons on protocol design are learnt. For example, in both AODV and DSR, flooding and caching seem to have a great effect on performance, while salvaging in DSR barely seems to have an effect on the protocol performance.

I. INTRODUCTION

Thanks for the availability of small and inexpensive wireless communication devices, research on MANET attracts attention from various research communities. In the near future, different types of mobile ad hoc networks are expected to be deployed in a few scenarios including mobile classroom, battlefield communication and disaster relief activities. The emergence of mobile ad hoc network will change the concept of mobile computing. This field provides several challenging problems that differentiate it from the traditional wired Internet. One main challenge is understanding the effects of mobility on the performance of ad hoc routing protocols.

In Ref. [1], we attempt to use a rich set of mobility models including Random Waypoint(RW), Reference Point Group Mobility(RPGM), Freeway(FW) and Manhattan(MH) models to evaluate the performance of several Ad Hoc network routing protocols like DSR, AODV and DSDV. It is realized that mobility pattern *DOES* influence the protocol performance. The following two observations are acknowledged as well:

- 1) Same MANET routing protocol behave differently under various mobility scenarios;
- 2) Different MANET routing protocols behave differently even for the same mobility scenario.

By introducing several useful metrics which can capture the characteristics of topology graph, Ref. [1] clearly identifies the difference between mobility patterns and provides a reasonable explanation for the 1st observation. However, it fails to give a

This work is partly supported by NSF Career Award 0134650. Ref.[2] is a detailed version of this work.

satisfied explanation for the 2nd observation. In this study, we aim to understand how the protocol mechanism decides protocol performance based on the behavior of protocol mechanistic building blocks. Especially, we are interested at analyzing the interplay between mobility and the protocol mechanistic building blocks which can be used to explain the 2nd observation.

We introduce a novel approach called BRICS to analyze the reactive ad hoc network protocols like DSR and AODV. In this approach, the overall protocol is decomposed into its constituent mechanistic building blocks and evaluated in a systematic way. Each building block is used to implement a specific function. Then, the effect of different mobility patterns on each building block is evaluated. We are specifically interested in understanding the contribution of each building block to the overall protocol performance. Such understanding enables us to reason about the performance difference between different protocols. Our results show that the contribution of these building blocks to the overall performance varies drastically. While flooding and caching have a great effect on performance, route salvaging (in DSR) barely has any effect. Finally, based on the extensive simulation, we learnt several useful lessons on design choice of building blocks and gain a deeper insight into protocol design philosophy. The contributions of our proposed building block approach are three fold:

- 1) Introduce a perspective to understand, investigate and justify the mechanistic functionalities of protocols which may be closely coupled and difficult to be analyzed;
- 2) Provide a methodology to justify the contribution of each mechanistic building blocks on overall protocol performance under various mobility pattern;
- 3) Through extensive simulation, we gain a deeper insight into design philosophy of ad hoc routing protocols.

II. BUILDING BLOCK METHODOLOGY

Extensive research has been done to compare and explain the differences of various routing protocols at the **whole protocol level** through simulation or intuitive analysis [3] [4] [5]. This helped us to rank protocols according to their performance and distinguish the pros and cons of each protocol. But the impact of the fine grained functionality on the whole protocol remains unknown. We propose a methodology to systematically compare and explain the commonalities and differences of routing protocols at a more detailed **building-block level**. This enables us to better understand the interplay between the building blocks and mobility characteristics, a topic that has been ignored in the previous studies.

Each protocol may be decomposed into a set of parameterized mechanistic “building blocks”. Each “building block” implements a specific well-defined functionality. These building blocks are then organized in a certain way to form the protocol as a whole. Although some protocols consist of building blocks of the same functionality in a similar organization, it is observed that their performance may differ significantly. For example, AODV and DSR behave differently although they belong to the generic class of reactive protocols. One of the reasons for this discrepancy might be that the building blocks are parameterized. Different values for the parameters lead to varying performance across these protocols belonging to the same category.

Thus, the architecture of a protocol consists of the following three elements: *mechanistic building blocks*, *the parameters of the mechanistic building blocks* and *the interaction between the mechanistic building blocks*. For a specific mobility pattern, the behavior of each building block is determined by its parameter setting. At the same time, the interaction between building blocks affects the protocol performance. So the performance behavior of the whole protocol is defined by both the characteristics of building blocks and the interaction between building blocks.

Decomposing a protocol into building blocks in a meaningful way may not be straightforward. As a first attempt, and based on our experience from previous studies, we propose a procedure to systematically decompose routing protocols into their constituent building blocks. We then use the reactive MANET routing protocols as a case study for our procedure. The procedure listed below is an outline to establish the building block architecture for MANET routing protocols:

- 1) *Identification of components*: The first step is to identify the major functionalities and abstract their fundamental mechanisms into building blocks.
- 2) *Organization of components*: The mechanistic building blocks are linked via exchanged messages between them and thus organized to form the whole protocol.
- 3) *Generalization of components*: Then, based on the functionality of the previously chosen building blocks, new, more generalized building blocks may be obtained by merging mechanisms from different protocols that have common structure and achieve similar functionalities.
- 4) *Parameterization of components*: Finally, by changing the setting of the parameters of these generalized building blocks, we can easily represent the different mechanisms.

Thus, each of the protocols under study may be composed of a set of parameterized, generalized building blocks.

III. BUILDING BLOCK ANALYSIS OF REACTIVE MANET ROUTING PROTOCOLS

As an illustration, two reactive protocols, DSR and AODV, are analyzed based on the building block methodology.

A. Decomposition and Organization

The operation of DSR is composed of two phases: *Route Setup phase* and *Route Maintenance phase*. The objective of the former is to find a route to the destination within the network. Two major mechanisms are used to achieve this objective: A

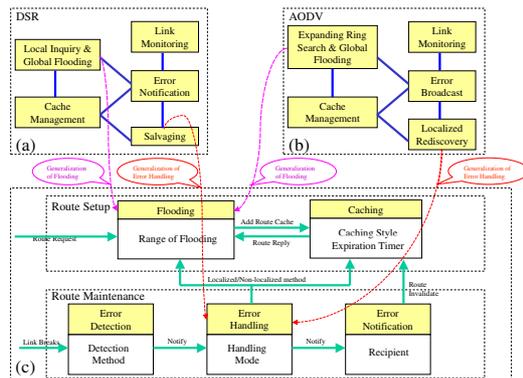


Fig. 1. Diagram of Building Block Framework for Reactive Protocols

mechanism called global flooding is implemented to distribute the *route request* messages within the network once the attempt of source to get a route reply from its direct neighbors fails. Besides, another mechanism is implemented to manage the cached routing information at the nodes, including how to add, invalidate and utilize the cached route entries. Unlike the conventional wired network, the links within the wireless network are highly unstable due to the mobility and wireless propagation losses. The unstable links are dealt with in the *Route Maintenance phase*. DSR monitors the link status at the MAC layer. If a broken link is detected, the salvaging mechanism is used to find an alternative route. At the same time, in DSR, a *route error* message is sent in the direction of the source to eliminate the invalid cache entries.

The operation of AODV is similar to DSR, the routing function is achieved in *Route Setup phase* and *Route Maintenance phase*: Expanding ring search and global flooding schemes are in charge of distributing the *route request* message in the network while a caching mechanism is used to maintain routes and to reply the route requests based on the cached routing information. Hello messages are used by AODV to monitor the link status. If a broken link is detected, a localized route discovery mechanism is re-initiated by the upstream node to repair the broken route in some scenarios. Nodes within the network are notified about the error so that the stale cache entries can be removed.

The part(a) and part(b) of Figure 1 show the architecture for DSR and AODV, respectively, based on the BRICS approach. This kind of architecture is one of many possible candidates.

B. Generalization and Parameterization

From the previous section, we observe that both DSR and AODV can be decomposed into a similar set of basic mechanistic building blocks. By comparing and analyzing the commonalities and differences of the functionalities of these components, we can construct generalized building block that encompasses the building blocks of each protocol. The part(c) of Figure 1 shows a generalized building block architecture for both DSR and AODV. As a example shown in Figure 1, salvaging from DSR and localized error recovery from AODV are compared and merged as the *error handling* building block whose major functionality is to find alternative route to replace the invalid route. The parameter of this generalized building

block is the mode of error handling used. Once the building blocks are generalized, the key parameters of building blocks can be identified by studying the difference in the functionality of various protocols. We now discuss the design choices (parameter settings) of each of the identified building blocks.

Route Setup Phase is initiated if there is no cached route available to the destination.

Flooding: Flooding is mainly used for route discovery if the route to the destination does not exist in the sender's cache. Here, the key parameter is *the range of flooding*, generally described by TTL field in the IP header. One optimization is to implement the localized controlled flooding before global flooding. This may be useful if the probability of finding an appropriate cache in the neighborhood is high.

Caching: Several parameters affect the behavior of the caching building block. One parameter is *whether aggressive caching is allowed*, i.e. whether multiple cache entries are allowed for the same destination and whether a node can cache the route information it overhears? Generally speaking, aggressive caching scheme increases the possibility of finding an appropriate route without re-initiating a route discovery.

Route Maintenance Phase takes the responsibility of detecting broken links and repair the corresponding routes.

Error Detection: It is used to monitor the status of the link of a node with its immediate neighbors. Several methods to monitor the link status between neighbors can be used: MAC level acknowledgement, network-layer explicit Hello message or network-layer passive overhearing scheme. Here, the parameter is *the mode of error detection used*.

Error Handling: It finds alternative routes to replace an invalid route after a broken link is detected. One of the parameters to this block is *whether localized recovery should be used*. In a localized recovery, the node detecting the broken link will attempt to find an alternative route in its own cache or do a localized flooding before asking the source to re-initiate the route discovery.

Error Notification: It is used to notify the nodes in the network about invalid routes. The key parameter to this building block is *the recipient of the error message*. Either only the source is notified or the entire network is notified.

C. Comparison of parameters of DSR and AODV

Having identified the generalized building blocks, we now study the specific parameter settings for these building blocks for DSR and AODV. We pose some questions about the utility of the various design choices made by these protocols. In section IV, we attempt to answer these questions.

- 1) *Flooding:* For the *range of flooding*, both DSR and AODV use the two-step controlled flooding. DSR conducts a non-propagating direct-neighborhood inquiry (TTL=1) first before the global flooding (TTL=D, D is network diameter). Similarly, AODV uses the expanding ring search (TTL=1,3,5,7) before the global flooding is initiated. Here, we want to answer the following question: *How useful are non-propagating route requests?*
- 2) *Caching:* DSR uses *aggressive caching*, while AODV does not. Another parameter is the *expiration timer for the cache entry*. However, for caching, We are only interested in the following questions: *How useful is caching?*

and Is aggressive caching better than non-aggressive caching?

- 3) *Error Detection:* For *mode of error detection*, both DSR and AODV can use either of the three choices mentioned in section III-A. So, we do not investigate this building block in our analysis.
- 4) *Error Handling:* For *localized recovery*, in DSR, on detecting a broken link, the upstream node will first search its cache to replace the invalid route, even though the found alternative route may also be invalid in some scenarios. While in AODV, the upstream node detecting the broken link will initiate a localized flooding. For this building block, we are interested in the following question: *Which is a better scheme for localized error handling: cache lookup or localized flooding?*
- 5) *Error Notification:* For *recipient of error notification*, both DSR and AODV notify the error to the source. So, we do not investigate this block during our simulations.

Beside these three questions about the design choices, we are also interested at the explanation for the observation we made in [1]: DSR outperforms AODV in most scenarios except Freeway and Manhattan Models with high mobility.

IV. PERFORMANCE EVALUATION AND DISCUSSION

We identified parts of codes in the network simulator (*ns-2*) with the CMU Wireless Ad Hoc networking extension which implement these building blocks and profiled them during our simulations [6]. Our mobility scenario generator [1] produced the different mobility patterns following the Random Waypoint (RW), RPGM, Freeway (FW) and Manhattan (MH) models according to the format required by *ns-2*. In all these patterns, 40 mobile nodes moved in an area of 1000m x 1000m for a period of 900 seconds. For RPGM, we used 2 different mobility scenarios: single group of 40 nodes and 4 groups of 10 nodes each moving independently of each other and in an overlapping fashion. The maximum speed V_{max} was set to 1, 5, 10, 20, 30, 40, 50 and 60 m/sec to generate different movement patterns for the same mobility model. The traffic pattern is consisted of randomly chosen 20 Constant Bit Rate (CBR) sources and 30 connections. The data rate used was 4 packets/sec and the packet size was 64 bytes.

A. Analysis of DSR and AODV

1) *Flooding:* Figures 2 and 3 show the ratio of non propagating route requests to the total number of route requests issued by the DSR and AODV respectively. This metric measures the likelihood of finding a route to the destination from the source's neighbors. Through simulations, we find that the non-propagating route request is frequently used (more than 30% for DSR and more than 10% for AODV in most scenarios).

It is also observed that this ratio increases from RW to MH to FW mobility models. This is because the geographic constraints on movement are greater in FW than MH, which are in turn greater than in RW. Thus, the likelihood of finding a route to the destination from the source's neighbors increases from RW to MH to FW. RPGM (Single and Multiple groups) has a high average degree of spatial dependence. Hence, it is expected that this ratio will be high. However it turns out that

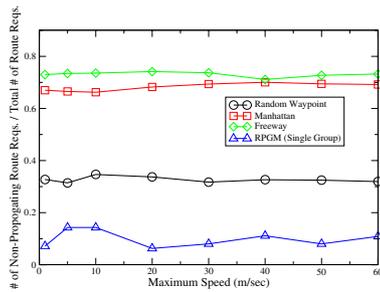


Fig. 2. Ratio of Non Propagating Route Requests to Total Number of Route Requests for DSR

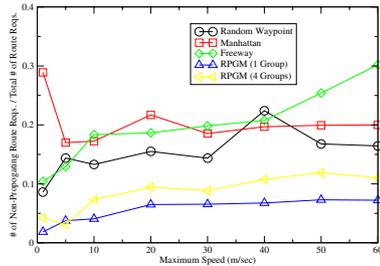


Fig. 3. Ratio of Non Propagating Route Requests to Total Number of Route Requests for AODV

this is not the case. As Figure 3 shows, the ratio is the lowest for RPGM. This is because, in RPGM, the propagating route requests at the first time make up a large percentage of the total route requests. Since the route remains stable for long periods of time, very few route requests retries are done by the source.

On the other hand, the ratio for DSR is almost twice as large as that for AODV across all mobility models. A possible reason for this might be the fact that DSR uses aggressive caching as compared to AODV. When such a caching scheme is coupled with the mechanism of non propagating route requests, it translates to low routing overhead and high throughput as was shown in [1] and several other comparative studies. Thus, it seems that caching has a significant impact on the performance of DSR and AODV. Hence we study it next.

2) *Caching*: To measure the effectiveness of caching, we evaluate the ratio of the number of route replies coming from the cache to the total number of route replies. Figures 4 and 5 show that this ratio is high for RW, MH and FW models, which implies that most of the route replies for these mobility models come from the cache. On the other hand, since RPGM (Single and Multiple Groups) has a high degree of spatial dependence, it would be expected that most of the route replies can be found in the cache of the nearby nodes. However, it turns out that the ratio is lower for RPGM. This is because, most of the route replies come from the destination when the first route request is sent by the source to the destination. After this, the route remains stable during the entire simulation and thus very few route requests retries are done. Since the stability of the routes in the case of multiple groups is lesser than that in the single group case, the ratio is higher in the former case than the latter one.

The difference in the ratio for DSR and AODV is greater than

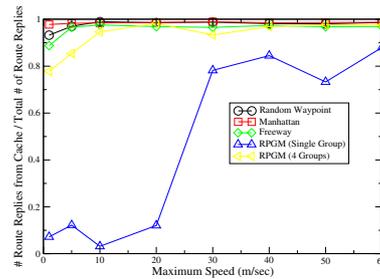


Fig. 4. Ratio of the Number of Route Replies from the Cache to the Total Number of Route Replies for DSR

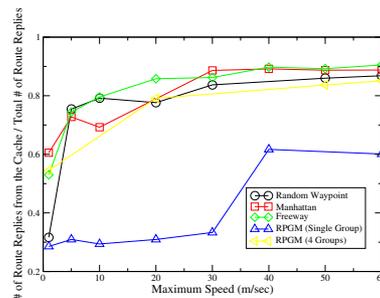


Fig. 5. Ratio of the Number of Route Replies from the Cache to the Total Number of Route Replies for AODV

20% for all mobility models. DSR uses aggressive caching as compared to AODV. Thus, the likelihood of a route reply coming from a cache is higher in DSR than in AODV. Thus, fewer route requests will be needed and thus the routing overhead of DSR is lower than AODV as was concluded in several comparative studies. Thus, aggressive caching seems to be a good design choice.

To completely evaluate the caching strategy, we also need to examine the validity of the cache entries. We evaluate the ratio of invalid cache entries to the total number of cache entries for DSR. As shown by Figure 6, the ratio increases from RPGM to RW to FW to MH mobility models. Also, it was observed that the ratio of route replies from the cache to the total number of route replies was higher for MH than FW which in turn was higher than RW. Thus caching may have adverse effects in mobility models with a high relative speed. Packets may be sent on invalid routes which might lead to packets being dropped and route request retries. This leads to a lower throughput and higher overhead for DSR for the RW, FW and MH models as was shown in [1].

On the other hand, in mobility models with very high relative speed like MH and FW, AODV seems to achieve as good a throughput as DSR (and sometimes better), as was shown by [1]. AODV does not use aggressive caching, thus the ratio of the number of route replies coming from the cache to the total number of route replies is lesser for AODV than DSR. Thus, the likelihood of getting invalid routes from the cache is lesser for AODV than for DSR. Moreover, at high relative speeds, the number of routes broken is greater. Thus, a protocol which has a better error handling mechanism at higher relative speeds might perform better in such situations. This line of reasoning

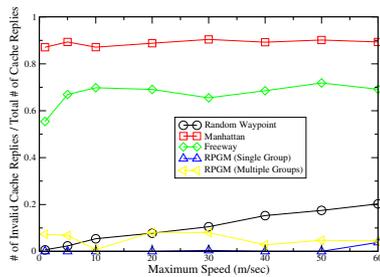


Fig. 6. Ratio of the Number of Invalid Route Replies from the Cache to the Total Number of Route Replies from the Cache

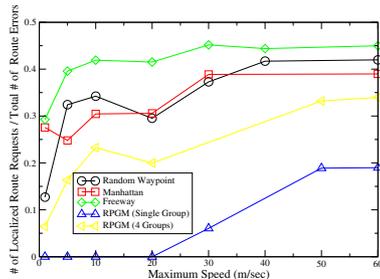


Fig. 7. Ratio of the Number of Localized Route Recovery Requests to the Total Number of Route Errors for AODV

leads us to evaluate the next building block of interest - Error Handling.

3) *Error Handling*: To study the effectiveness of error handling, we focus on localized error handling. We evaluate the ratio of the number of localized error handling to the total number of route errors for both DSR and AODV. For DSR, we notice that salvaging accounts for less than 2% of the total number of route errors. Moreover, if we take invalid cache entries into account, the effect of salvaging on the protocol performance is further lowered. On the other hand, in AODV, a route request is initiated by the upstream node which detects the broken link if it is closer to the destination. As Figure 7 shows, the ratio is between 40%-50% for FW and MH models. Moreover the routes obtained by this mechanism are more up to date than those from the cache. This is probably another factor which explains the better performance of AODV as compared to DSR in the FW and MH models.

B. Discussion

The above study of the building blocks has given us greater insight into the design of the reactive routing protocols for MANETs. Decomposing a protocol into building blocks and evaluating these building blocks have shown us the scenarios in which the chosen parameters can give a better performance. From the above study, we learnt the following principles of protocol design:

- 1) Caching helps reduce the protocol overhead. However, whether aggressive caching should be used depends on the scenarios in which the protocol will be deployed. For low mobility scenarios, aggressive caching might be useful, while for higher mobility scenarios, the stale cache entries might affect the protocol throughput.

- 2) Non Propagating route requests, when combined with caching also reduce the protocol overhead. If caching is widely done in the network, it may be more advantageous to do non propagating route requests (or expanding ring search) than globally flooding the route request. In DSR, due to aggressive caching, it may be more useful to do expanding ring search (from the source) on a route error than doing a global flooding (from the source). Again this might work well only for low mobility scenarios.
- 3) The nature of localized error handling also has a significant impact on protocol performance. Re-initiating a route request from an intermediate node can be more advantageous than doing a local cache lookup in high mobility scenarios, while a cache lookup might be more advantageous for low mobility scenarios.

Thus, no particular parameter setting of these building blocks is the most optimal for all scenarios. This further strengthens our conclusion in [1] that there is no clear winner among the protocols across all mobility scenarios.

V. CONCLUSION AND FUTURE WORK

From previous studies, we observed that performance of MANET routing protocols vary significantly with mobility. Moreover, it was observed that mobility impacts different protocols differently even if they follow similar mechanisms as in the case of DSR and AODV. In an attempt to answer such performance discrepancies, we carried our research beyond the whole protocol level. This led to the building block based approach for analyzing routing protocols. Using this approach, we analyze the reactive MANET routing protocols as a case study. We demonstrate the possible benefits of such an approach by relating the performance of the building blocks to the protocol performance. Through experiments, several lessons that may help in designing protocols with better performance are learnt.

As part of the future work, we would like to extend this approach to pro-active protocols like DSDV, among others. While proposing the building block architecture, we used our intuition about the mechanisms of DSR and AODV. To accomplish the same in a generalized manner, using an algorithm, would be one of our future research directions.

REFERENCES

- [1] F. Bai, N. Sadagopan and A. Helmy. IMPORTANT: A framework to systematically analysis the Impact of Mobility on Performance of Routing protocols for Adhoc NeTwork. in Proceedings of IEEE INFOCOM 2003.
- [2] F. Bai, N. Sadagopan and A. Helmy. BRICS: A Building-block approach for analyzing Routing protocols in ad hoc networks - a Case Study of reactive routing protocols. USC-CS-TR-02-775.
- [3] J. Broch, D. A. Maltz, D. B. Johnson, Y-C. Hu, and J. Jetcheva. A Performance Comparison of Multi-Hop Wireless Ad Hoc Network Routing Protocols. In Proceedings of the Fourth Annual ACM/IEEE International Conference on Mobile Computing and Networking, ACM, Dallas, TX, October 1998.
- [4] S-J. Lee, M. Gerla, and C-K. Toh. A Simulation Study of Table-Driven and On-Demand Routing Protocols for Mobile Ad Hoc Networks. IEEE Network, vol. 13, no. 4, Jul/Aug 1999, pp. 48-54.
- [5] S. R. Das, C. E. Perkins and E. M. Royer. Performance Comparison of Two On-demand Routing Protocols for Ad Hoc Networks. Proceedings of INFOCOM 2000 Conference, Tel-Aviv, Israel, March 2000.
- [6] L. Breslau, D. Estrin, K. Fall, S. Floyd, J. Heidemann, A. Helmy, P. Huang, S. McCanne, K. Varadhan, Y. Xu and H. Yu. Advances in Network Simulation, IEEE Computer, vol. 33, No. 5, p. 59-67, May 2000.