ABSTRACT

ZHAO, CHEN. N-Body: Social Mobility Model for Wireless Ad Hoc Network Research. (Under the direction of Mihail L. Sichitiu.)

The contact time between mobile nodes is a fundamental factor that affects performance in Mobile Ad Hoc Networks (MANETs). Previous research on theoretical analysis of contact time distribution for random walk models assumes that the contact events can be modeled as either consecutive random walks or direct traversals, which are two extreme cases. In the first part of this dissertation we bridge the gap between the two extremes. These two cases will result in a power-law or exponential tail in the contact time distribution, respectively. We show that the actual distribution varies between the two extremes: a power-law-sub-exponential dichotomy, whose transition point depends on the average flight duration. Through simulation results we show that this conclusion also applies to random waypoint.

An accurate reproduction of real human movement is essential in simulations of MANETs in order to obtain meaningful performance results. Existing models capturing the features of real world movements often require knowledge of the underlying dynamics in the target scenario, therefore limits their application scope. In the second part of this dissertation we tackle this problem from a different perspective. Rather than considering the details of the target scenario, we extract features from a sample trace, and synthesize traces that match the sample trace in those features. In particular, as human activities are often socially organized, resulting in a tendency of forming groups, we propose an N-body mobility model that extracts the grouping information from sample traces, and reproduces them in the synthesized traces. We show that the N-body model is capable of capturing and synthesizing the tendency to group that matches to those observed from sample traces.

Since the N-body model relies on the inter-nodal relationship data, which can only be obtained from a movement trace of a population no smaller than the target scenario, its application scope is limited. Therefore, in the third part of this dissertation we seek to enhance the N-body model by developing a method to generate such relationship data from the data of a smaller population. We propose an improved configuration model that synthesizes weighted networks according to adjusted statistics extracted from a sample network. This model is validated through simulation.
N-Body: Social Mobility Model for Wireless Ad Hoc Network Research

by

Chen Zhao

A dissertation submitted to the Graduate Faculty of
North Carolina State University
in partial fulfillment of the
requirements for the Degree of
Doctor of Philosophy

Electrical Engineering

Raleigh, North Carolina
2011

APPROVED BY:

__________________________  _________________________
Do Young Eun                  Injong Rhee

__________________________  _________________________
Wenye Wang                    Mihail L. Sichitiu
Chair of Advisory Committee
DEDICATION

To my love,
Xie-Hong,
Derek, and Zoe.

Also to
Grandma, Grandpa,
Papa, Mama, Mom-in-law, and Huan.

Even youths grow tired and weary,
and young men stumble and fall;
but those who hope in the LORD
will renew their strength.
They will soar on wings like eagles;
they will run and not grow weary,
they will walk and not be faint.

—– Isaiah 40:30-31 (NIV)
BIOGRAPHY

Chen Zhao was born in 1984 and spent his childhood in Suchow, a Chinese city often dubbed “Heaven on Earth” and “Venice of the East”. From 2002 to 2006, he attended Shanghai Jiao Tong University where he obtained his B.S. degree in Electrical Engineering with a minor in International Business. Afterwards he moved to United States and joined the Wireless Ad-hoc and Local Area Networks (WALAN) lab in North Carolina State University to pursue a Ph. D. degree in Electrical and Computer Engineering under the direction of Dr. Mihail L. Sichitiu. His research is focused on mobility modeling in wireless ad-hoc networks, particularly group mobility patterns. He is a member of IEEE, ACM, and HKN. He is married to his wife, Xie-Hong Li, and has two children, Derek and Zoe.
ACKNOWLEDGEMENTS

First and foremost, I would like to express my deepest gratitude to my advisor, Dr. Mihail L. Sichitiu. Without his trust, patience, persistence guidance and selfless support, this dissertation would never come into being. Whenever I’m facing difficulties or obstacles, Dr. Sichitiu is always inspirational and encouraging, leading me to breakthroughs one after another. Being not only my advisor, but also a friend for life, I have always wished I could live up to the example he has set: as a researcher, an engineer, and a man.

I would like to express my appreciation to all my committee members: Dr. Injong Rhee, Dr. Wenye Wang and Dr. Do Young Eun, for their comments, critique and suggestions that are invaluable to the success of this research. I would also like to thank Dr. Rong Peng, with whom I worked together in the early stage of this project. She has always been nice, warm-hearted and strict, and it was my honor and pleasure to have such a good mentor during my first year of research.

I would also like to thank Dr. Shiwen Mao (Auburn University), Dr. Rudra Dutta, Dr. Ming Zhao, Dr. Yuh-Ming Chiu, Dr. Han Cai, and Dr. Kyunghan Lee for their insightful comments and useful advice on this research. I would also like to acknowledge the financial support of Research-In-Motion and National Science Foundation (under grant NSF-0626850).

I would also want to thank my lab-mates and friends, Dr. Suyoung Yoon, Dr. Junbum Lim, Dr. Beakcheol Jang, Dr. Xingbin Wang, Dr. Kyuyong Shin, Di Xiao, Mani Pandian, Dr. Yi Xu, Yongchul Kim, Chul-Ho Lee, Hee Won Lee, as well as friends from Raleigh Chinese Christian Church and Colonial Baptist Church, especially Dr. Ben Liu, Dr. Chien-Lung Wu, Larry Hoxie and their families, for their encouragement and support during our difficult times, and sharing the laughter and sorrows in life.

Finally, I’m forever indebted to my grandparents and parents for their endless love, care and support, especially my grandparents who raised me with their unconditional love. Words are not enough to express my gratitude toward them. I also want to mention my mother-in-law, who selflessly took good care of my family when we were in difficulty. I’m so lucky to marry my wife, Xie-Hong, who loves me so much that she is willing to give up all her career and be the woman behind me, taking care of me and our children, Derek and Zoe. This dissertation is dedicated to them.
# TABLE OF CONTENTS

List of Tables ................................................................. vii

List of Figures ............................................................... viii

Chapter 1 Introduction ......................................................... 1
  1.1 Motivation ............................................................... 1
  1.2 Contact Time in Random Walk and Random Waypoint ................. 2
  1.3 Grouping in Mobility Models ........................................ 4
  1.4 Extending to Larger Populations ..................................... 6
  1.5 Outline .................................................................. 7

Chapter 2 Contact Time in Random Walk Mobility Models ..................... 9
  2.1 Model Analysis .......................................................... 9
    2.1.1 Model Settings and Assumptions .................................. 9
    2.1.2 Random Walk with Infinite Flight Lengths ..................... 10
    2.1.3 Tail Behavior of Contact Time Distribution .................... 12
    2.1.4 Random Walk with Finite Flight Lengths ....................... 15
  2.2 Validation ................................................................ 19
    2.2.1 Infinite Flight Lengths without Pause ......................... 19
    2.2.2 Finite Flight Lengths without Pause ............................ 19

Chapter 3 N-Body Mobility Model .............................................. 24
  3.1 Social Metrics ............................................................ 24
    3.1.1 Friendship / Steadiness Matrices ................................. 24
    3.1.2 Sample Traces ...................................................... 26
  3.2 N-Body Mobility Model ................................................... 29
    3.2.1 Basic Dynamics .................................................... 29
    3.2.2 Attraction Forces and Speed Control ............................ 31
    3.2.3 Inter-Nodal Forces ............................................... 32
    3.2.4 Closed Loop Pre-Synthesis ..................................... 34
  3.3 Simulation and Validation ............................................... 37
    3.3.1 Influence of Individual Parameters .............................. 37
    3.3.2 Synthesis from Sample Traces .................................... 38

Chapter 4 Extending Complex Group Structures ................................ 44
  4.1 Network Generation Algorithm ......................................... 44
    4.1.1 Relationship Networks and Testing Scenarios ................... 45
    4.1.2 Configuration Model and Role-Model Enhancement ............ 46
    4.1.3 Weight Distribution ............................................... 47
    4.1.4 Weight Dependent Cluster Coefficient ........................... 51
    4.1.5 Network Generator Algorithm ................................... 55
  4.2 Simulation and Validation ............................................... 58
<table>
<thead>
<tr>
<th>Section</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>4.2.1 Cluster Coefficients and Modularity</td>
<td>58</td>
</tr>
<tr>
<td>4.2.2 N-Body Model Usage</td>
<td>61</td>
</tr>
<tr>
<td>Chapter 5 Conclusions and Future Work</td>
<td>67</td>
</tr>
<tr>
<td>5.1 Conclusion</td>
<td>67</td>
</tr>
<tr>
<td>5.2 Future Work</td>
<td>68</td>
</tr>
<tr>
<td>5.2.1 Geographical Diversity</td>
<td>68</td>
</tr>
<tr>
<td>5.2.2 Performance Analysis</td>
<td>69</td>
</tr>
<tr>
<td>References</td>
<td>71</td>
</tr>
<tr>
<td>Appendices</td>
<td>77</td>
</tr>
<tr>
<td>Appendix A</td>
<td>78</td>
</tr>
<tr>
<td>Appendix B</td>
<td>80</td>
</tr>
<tr>
<td>Appendix C</td>
<td>83</td>
</tr>
<tr>
<td>Appendix D</td>
<td>87</td>
</tr>
<tr>
<td>Appendix E</td>
<td>89</td>
</tr>
<tr>
<td>Appendix F</td>
<td>92</td>
</tr>
</tbody>
</table>
LIST OF TABLES

<table>
<thead>
<tr>
<th>Table</th>
<th>Description</th>
<th>Page</th>
</tr>
</thead>
<tbody>
<tr>
<td>3.1</td>
<td>Influence of Individual Parameters</td>
<td>35</td>
</tr>
<tr>
<td>3.2</td>
<td>Adjustment Strategy (Primary)</td>
<td>36</td>
</tr>
<tr>
<td>3.3</td>
<td>Adjustment Strategy (Secondary)</td>
<td>36</td>
</tr>
<tr>
<td>3.4</td>
<td>Parameters for Simulating Target Scenarios</td>
<td>38</td>
</tr>
<tr>
<td>4.1</td>
<td>RPGM Scenarios Group Structure</td>
<td>50</td>
</tr>
<tr>
<td>Figure</td>
<td>Description</td>
<td></td>
</tr>
<tr>
<td>--------</td>
<td>-------------</td>
<td></td>
</tr>
<tr>
<td>2.1</td>
<td>$N_i$ and $N_j$ are moving at velocity $\vec{v}_i$ and $\vec{v}<em>j$. Both with communication range $r$. From $N_j$'s perspective, $N_i$ is traversing $N_j$'s contact region (the dashed circle) at the relative velocity $\vec{v}</em>{ij}$ along the thick chord $s$.</td>
<td></td>
</tr>
<tr>
<td>2.2</td>
<td>The integral interval in (2.15) assuming $v_{ij} \leq v_{\text{min}}$. When $v_{ij}$ is given, $v_i$ and $v_j$ have to satisfy (2.16) so the integral interval is the gray hexagon. Alternatively, the upper and lower bound of (2.15) can be obtained by integration over the horizontally hatched area and vertically hatched area, respectively.</td>
<td></td>
</tr>
<tr>
<td>2.3</td>
<td>Nodes $N_2$ and $N_3$ are both traversing the contact region of $N_1$. Both events will have the same duration $t_1$ if not interrupted. But the traversal between $N_3$ and $N_1$ is interrupted since $N_1$ finishes its flight before the traversal ends.</td>
<td></td>
</tr>
<tr>
<td>2.4</td>
<td>CCDF of $t_1$ with heterogeneous speeds and without pause, with flight lengths approaching infinity. (a) Results of the RW model, (b) results of the RWP model.</td>
<td></td>
</tr>
<tr>
<td>2.5</td>
<td>CCDF of uninterrupted traversal duration $t_5$. Speed parameter $H = 1$ (homogeneous speeds) for (a) (b) (c) and $H = 10$ for (d) (e) (f). Mobility models: (a) (d) RWP, (b) (e) RWA, (c) (f) RWB. All with contact range $r = 1m$.</td>
<td></td>
</tr>
<tr>
<td>2.6</td>
<td>CCDF of contact times for models with homogeneous speeds ($H = 1$) on logarithm plots (a) (b) (c) and semi logarithm plots (d) (e) (f). The vertical lines are the critical points $t_C$ where the survival rate $\sigma(t_C) = 50%$. Mobility models: (a) (d) RWP, (b) (e) RWA, (c) (f) RWB.</td>
<td></td>
</tr>
<tr>
<td>2.7</td>
<td>CCDF of contact times for models with heterogeneous speeds ($H = 10$) on logarithm plots (a) (b) (c) and semi logarithm plots (d) (e) (f). The vertical lines are the critical points $t_C$ where the survival rate $\sigma(t_C) = 50%$. Mobility models: (a) (d) RWP, (b) (e) RWA, (c) (f) RWB.</td>
<td></td>
</tr>
<tr>
<td>3.1</td>
<td>CDF of Friendship Index $r_{ij}$, (a) for all 4 traces: RPM, RPGM, SFCAB and STBUS, and (b) for flights from the same / different routes in STBUS trace.</td>
<td></td>
</tr>
<tr>
<td>3.2</td>
<td>CDF of Steadiness Index $w_{ij}$, (a) for all 4 traces: RPM, RPGM, SFCAB and STBUS, and (b) for flights from the same / different routes in STBUS trace.</td>
<td></td>
</tr>
<tr>
<td>3.3</td>
<td>Two nodes $n_i$ and $n_j$ with a strong grouping tendency are attracted to each other by the inter-nodal force $F_{ij}$, thus deviating from their normal route and move together for a while, like two friends talking to each other for a while on their way to work.</td>
<td></td>
</tr>
<tr>
<td>3.4</td>
<td>Several nodes are attracted to each other and formed a clog. They will move as a whole and visit the destinations according to their distances.</td>
<td></td>
</tr>
<tr>
<td>3.5</td>
<td>Inter-nodal force as a function of distance ($\epsilon = 1$, $s = 1$, $\sigma = 2$).</td>
<td></td>
</tr>
</tbody>
</table>
Figure 3.6 Influence of individual parameters on the mean $M_{i,j}$ (a)(b)(c) and NSD $S_{i,j}$ (d)(e)(f) of the pairwise distances. Parameter: (a)(d) $\epsilon = 0.5$, (b)(e) $\epsilon = 2$, (c)(f) $\epsilon = 10$. The $\sigma$ axis is transposed in (d)(e)(f) for improved visibility.

Figure 3.7 Trends of convergence during the closed loop pre-synthesis (a)(b)(c) and convergence results at the end of pre-synthesis as target versus the actual mean distance $M_{i,j}$ (d)(e)(f) and NSD $S_{i,j}$ (g)(h)(i). Mobility Models: (a)(d)(g) RPGM, (b)(e)(h) SFCAB, (c)(f)(i) STBUS. Solid lines in (d)-(i) have the slope of the ratio between $M_{\text{ind}}^{\text{ind}}$ and $M_{\text{ind}}$ or between $S_{\text{ind}}^{\text{ind}}$ and $S_{\text{ind}}$. Dashed lines have the slope of 1.

Figure 3.8 Light traffic simulation results for AODV (a)-(f) and DSDV (g)-(l): CCDF of pairwise packet delivery ratio (a)-(c),(g)-(i) and CCDF of pairwise delay (d)-(f),(j)-(l). Mobility models: (a)(d)(g)(j) RPGM, (b)(e)(h)(k) SFCAB, (c)(f)(i)(l) STBUS.

Figure 3.9 Heavy traffic simulation results for AODV (a)-(f) and DSDV (g)-(l): CCDF of pairwise packet delivery ratio (a)-(c),(g)-(i) and CCDF of pairwise delay (d)-(f),(j)-(l). Mobility models: (a)(d)(g)(j) RPGM, (b)(e)(h)(k) SFCAB, (c)(f)(i)(l) STBUS.

Figure 4.1 Groups in the Relationship Network

Figure 4.2 Different scenarios of the group structure when increasing population.

Figure 4.3 Illustration of the distortion function. ($w_0 = 0.3$)

Figure 4.4 Aggregated CDF of Edge Weight in RPGM Traces and Output Networks.

Figure 4.5 Aggregated CDF of Edge Weight in BUS Traces and Output Networks.

Figure 4.6 Weighted cluster coefficients does not scale well.

Figure 4.7 Forming unweighted networks at different thresholds.

Figure 4.8 Selected thresholds based on global cluster coefficients of the RPGM-O input network.

Figure 4.9 Selected thresholds based on global cluster coefficients of the BUS-O input network.

Figure 4.10 Global cluster coefficient in RPGM traces and output networks.

Figure 4.11 Global cluster coefficient in BUS traces and output networks.

Figure 4.12 Optimal number of clusters derived from modularity for RPGM scenarios.

Figure 4.13 Optimal number of clusters derived from modularity for BUS scenarios.

Figure 4.14 Trends of convergence during the closed loop pre-synthesis.

Figure 4.15 Average percentage of population reached by packets over time after they are generated, RPGM Scenarios.

Figure 4.16 Average percentage of population reached by packets over time after they are generated, BUS Scenarios.

Figure 4.17 CDF of pairwise packet delivery ratio (PDR) and end to end delay in RPGM scenarios.

Figure 4.18 CDF of pairwise packet delivery ratio (PDR) and end to end delay in BUS scenarios.
Figure 5.1  The mean pairwise distance assuming independent random placement $M_{i,j}^{\text{ind}}$ versus the actual mean pairwise distance $M_{i,j}$. (a) RPGM, (b) SFCAB, (c) STBUS. ......................................................... 68

Figure 5.2  Influence of mobility model features upon protocol independent or dependent performance metrics. ................................................................. 69

Figure B.1  Probability density of $\vec{v}$ plotted on polar coordinate. The distance represents the speed, and the angle represents the direction. Darker shadow means higher probability density. Speed and direction are assumed to be independent of each other. Here speed is uniformly distributed over $[v_{\text{min}}, v_{\text{max}}]$, and direction is uniformly distributed over $[0, 2\pi]$. .................. 80

Figure B.2  Two plots from Fig B.1 with origin $N_i$ and $N_j$ are shown such that their shadows partially overlap (double hatched area). $A$ is an arbitrary point in the overlapping area. Then $\overrightarrow{N_iA}$, $\overrightarrow{N_jA}$ and $\overrightarrow{N_iN_j}$ will form a vector triangle, representing $\vec{v}_i$, $\vec{v}_j$ and $\vec{v}_{ij}$, respectively. ....................... 81

Figure C.1  A small move in one direction may result in stronger attraction force in the opposite direction. ................................. 84

Figure C.2  A semi-stable balance between four attracting destinations evenly distributed in a circle. ................................. 85

Figure F.1  Comparing global cluster coefficients with / without extension: RPGM-I scenario. ......................................................... 93

Figure F.2  Comparing global cluster coefficients with / without extension: BUS-I scenario. ......................................................... 93
Chapter 1

Introduction

1.1 Motivation

Mobile ad-hoc networks (MANETs) [1,2], as well as delay tolerant networks (DTNs) [3], which feature multi-hop wireless communication between mobile nodes without central coordination, received a significant amount of attention over the recent years due to advances in low cost wireless technologies and to growing demand for infrastructure-less communication. Compared to traditional infrastructure-based broadcast wireless networks, such networks rely on intermediate nodes to forward each others’ traffic, while the communication links may break very often, and the routes are to be established dynamically. Their distributed nature and inherent mobility has made research of these networks extremely difficult. Thus, despite significant research efforts as well as the thousands of papers published so far in the literature, unfortunately, there are very few real deployments, especially large scale ones, of such network applications that are available for the research community. MANETs or DTNs may be deployed in vastly diverse environments, e.g., military applications in the battlefield may have completely different performance from student communications in a campus, just as voice applications may have different requirements from data applications; therefore it is desirable for researchers to be able to validate the research results under highly diverse conditions. Due to both deployment scarcity and diversity requirements, current research on MANETs and DTNs is heavily based on simulation, which are often the only option when a real deployment is not available, while offering convenience to diversify between scenarios. Thus, effective and accurate simulation is of crucial importance to the research community to validate research results. In MANETs and DTNs, in particular, since components of such networks are assumed to be mobile, mobility patterns have to be synthesized under certain mobility models. Accurate reproduction of real life movement patterns for possible deployments of MANETs and DTNs are thus, essential to obtain meaningful results from simulations [4] and to avoid artificial artifacts.
Since human movement patterns are currently far from being well understood [5], realistic mobility modeling over diversified scenarios has been a popular research area over the years and numerous models have been proposed (e.g., [6, 7] are excellent surveys). Early mobility models like random walk and random waypoint [8] or their variants [9,10] are Markovian based statistical models and thus, simple to implement and easy to diversify. However, with a limited state space, they are too simple to synthesize the various spatial-temporal dependencies in real human movements [11,12] as well as inter-nodal interactions. On the other hand, more detailed models [13–18] are often constructed based on the observations from real life scenarios and thus, provide higher realism. However, as these models often have specifically designed model dynamics, which often require detailed knowledge of the mobile behavior in the target scenarios, they are hard to diversify, thus limiting their scope of application.

Considering the drawbacks of both kinds of mobility models, in this work we propose a novel mobility model that achieves high realism while easily diversifies to various scenarios. Rather than identifying the detailed behavior of the nodes in specific scenarios and constructing a model for each scenario, this model features a framework that directly extracts necessary information as key parameters or distributions from a sample trace of a small population, and then synthesizes traces, usually of a larger population, using the extracted parameters so that it would have similar behaviors to the sample trace. In particular, since human movements are often socially correlated, our mobility model focuses on the group-forming behavior in real human movements, which to the best of our knowledge, none of the existing works is capable of doing so without prior knowledge of the underlying social structure of the target scenario.

In order to achieve a high degree of realism in the proposed mobility model, the influence of mobility parameters to the actual network performance should be well understood. Therefore, in addition to the mobility model, we conduct a preliminary study on the statistical behavior of contact time (link duration) of random walk and random waypoint models in Chapter 2. The purpose of this study is to gain a deeper understanding in the subtlety of the underlying dynamics between the model parameters (like flight length, speed and contact range) and the protocol-independent performance metrics like the contact time.

### 1.2 Contact Time in Random Walk and Random Waypoint

Random Walk (RW) and Random Waypoint (RWP) are widely used mobility models in the research community [6, 7] due to their simplicity, despite the fact that many researchers have pointed out their many drawbacks [12,19,20], and proposed more sophisticated ones like [12, 16,17,21,22]. However, even for the simple models like RW and RWP, the relationship between their input parameters (speed, pause, flight length, flight directions, etc.) and the corresponding impact on network performance is not yet quantitatively well understood.
For dense MANETs with dynamic routing protocols, network performance depends on both the mobility and the protocols. In [4] the authors proposed several protocol independent metrics including the link change rate and link duration, allowing the impact of mobility models to be evaluated through those metrics without reference to any specific protocol. For those with sparser nodes, e.g., Pocket Switched Networks (PSN) [23], there are also such protocol independent metrics like the inter-contact time and the contact time [11].

In both scenarios the contact time (or alternatively, link duration, link lifetime, link expiration time, etc.) has been an important performance metric in evaluating the impact of mobility. In the research presented in Chapter 2 we focus on the distribution of contact times of RW and RWP. Several papers studying this distribution have been published [24–32]. Some of them are based on empirical analysis [26, 30], while others are based on theoretical derivation [24, 25, 27–29, 31, 32].

Studies based on empirical analysis have the advantage of being accurate. In [26] the authors examined the PDFs of contact time through simulation and concluded that the PDFs are significantly different among different models. Among them the PDF of RW exhibits a single peak. The authors of [30] fit the PDF of contact time from RWP traces against several common distributions. The results showed that the lognormal distribution is the best fit for their traces.

Since it is very hard for the empirical analysis to analyze all parameter spaces, theoretical derivation is necessary to better understand the underlying dynamics between the model parameters and the contact time, even though such derivation usually imposes simplifying assumptions. In RW and RWP, nodal movements are consecutive flights along straight lines. When the communication range is small in comparison to the flight length, it is reasonable to assume nodes do not stop or change directions during contact events. Thus the contact events are modeled as direct traversals [25, 27–29]. In an early work [25] using this model, the duration distribution of two-hop paths with static senders and receivers was studied. In [27], the authors derived the contact time distribution of RW using the direct traversal model, assuming all nodes move at the same speed. In their later work [28] they extended the results with heterogeneous nodal speed. Both papers did not derive any closed form and all results were obtained numerically. In [29], the authors did a similar analysis as in [27] but derived a closed form for homogeneous speeds. They also obtained the contact time distribution numerically for two nodes with different fixed speeds.

On the other hand, when the communication range is large in comparison to the flight length, nodes often stop or change directions multiple times during contact events. Thus the contact events should then be modeled as the sum of consecutive random walks inside the nodes’ communication range (usually modeled as circles) [24, 31, 32]. In an early work [24] using this

---

1 All these terms refer to the same metric. In this paper the term contact time is used.
model, the authors derived the probability of link availability with different initial conditions. In [31] the authors proposed a two-state Markovian framework that can be used to approximate the contact time distribution of any mobility model. They also stated that the “direct traversal” model is a special case in their framework. A comprehensive analysis of contact times using this model was done in [32], where the authors concluded that the contact time distribution can be approximated as exponential. In [32] the communication range is a random variable and the mobility model was a “smoothed” variation of RW [22]. As a special case, their conclusion also applies to ordinary RW with constant communication range.

However, both assumptions, direct traversal and consecutive random walk, are essentially two extremes in regarding the ratio of communication range and flight length. In general, the actual behavior of RW models lies in between the two extremes. In Chapter 2 we conduct a comprehensive analysis that bridges these two extreme assumptions in previous works. We focus on their difference in the tail of the distribution of their contact times. We first show that when flight lengths are infinite, which is equivalent to the direct traversal assumption, the PDF of contact time has a power law tail with both homogeneous and uniform speed distribution. Moreover, when flight lengths are no longer infinite, the contact time distribution shows a power-law-sub-exponential dichotomy, with the transition point being a function of the flight time distribution. As the average flight length becomes shorter, the transition takes place earlier. When, finally, the flight length is short enough in comparison to the communication range, which is equivalent to the consecutive random walk assumption, the dichotomy degenerates into a single exponential tail, which conforms to the conclusion in [32].

1.3 Grouping in Mobility Models

Most mobility models proposed in the literature so far assumes that all nodes move independently without any inter-nodal correlation, e.g., the widely implemented random walk (RW) and random waypoint (RWP) [8]. In real life, however, human activities are often socially organized. People are related with each other through complex inter-personal relationships. These relationships form the social network, which is known to be highly clustered [33] as well as showing small world effects [34, 35]. Such inter-personal relationships also affect human mobility so that on many occasions people tend to move in groups, rather than individually, e.g., students attending lectures, platoons moving on the battlefield, families walking around at the state fair, or rescue squads searching for victims. On the other hand, because of the complicated nature of human behavior, such groups may be dynamic and of different strengths. People may stay in different groups at different times or even linger among several groups at the same time. Some groups may be tight and have fixed members, while others are loose and temporary. Therefore, real human movements can be quite heterogeneous. In recent works [36, 37] such
complex heterogeneity is well observed in real traces and is shown to affect the performance of both MANETs and DTNs.

Mobility models aware of such social aspects and the heterogeneity among nodes have been proposed in recent years [21, 38–53]. The Reference Point Group Model (RPGM) [38] is perhaps the earliest work taking such grouping tendency into account. In RPGM nodes are divided into groups and move randomly around the group centers, which are also moving. It is a direct and intuitive way to synthesize such tendency. Several variants of RPGM were proposed later in [39–44] with refined patterns of group-wise movements [39, 40] or intra-group movements [40–43], or with fine-grained structures inside groups [44]. However, all these models require explicit division of the population into groups, which do not represent situations where inter-personal relationships may be complex.

Another class of models [21,45–48] was proposed with the dynamics similar to early preferential attachment models in [54,55]. These models require an $N \times N$ numerical matrix for an $N$ node system, with each entry representing the strength of the social bond between the two corresponding nodes. Such matrices are usually referred to as social matrices. Although they share the preferential attachment principle that a node tends to go where its friends go, the actual dynamics differ from one another: The model in [45] derives a fixed meeting schedule for each node according to the cliques in the social matrix, and the strength of relationship corresponds to the frequency and duration of meetings. In models proposed in [21,46], the selection of nodes' destinations is influenced by their friends, with the social matrix corresponding to the influence level. In addition, the model in [46] also incorporates flock behavior first studied in [56]. The model in [47] behaves similar to RPGM, but determines the pause time based on the nodes' relationship with its neighbors. Finally, in [48] nodes are attracted to their friends, and they repulse each other once they’ve been together for enough long time. All these models provide means to incorporate a social network into the model dynamics. However, the derivation of such social networks requires an in-depth understanding of the underlying social structure in the simulation scenario, which is not apparent due to its complex interaction with the model dynamics. Therefore these models all have limited application scope. In fact, most of their actual simulations implement highly clustered networks\(^2\) where they are essentially reduced to variants of RPGM.

A third class of models tackles the network heterogeneity from a different point of view as a result of geographical heterogeneity between individual nodes [49,50]. Although these models do produce heterogeneous and even correlated movements if properly configured, they have the same weakness as the models similar to RPGM by being unable to synthesize the behavior with complex social interactions. Yet another class of models proposed in [51–53] implement

\(^2\)With the exception of [45,48] which focused on scale-free networks, however social networks are usually clustered rather than scale-free [33].
detailed role models to produce group movements. These models share the same shortcomings of those preferential attachment based models regarding the limited application scope.

In this research, aware of the shortcomings of existing models, we propose the *N-Body Mobility Model* inspired by N-body simulations [57]. **Rather than modeling the underlying social relationship, we simply try to capture the heterogeneity in inter-nodal distances by observing real movement trajectories and reproduce them in the synthesized traces.** Inter-nodal attraction and repulsion forces based on modified Gay-Berne potential [58] are implemented as to adjust the inter-nodal distances, which will be explained in detail in Chapter 3.

Using forces or similar means to adjust the nodal movements in mobility models has been considered in the literature. In an early work [41] the authors implemented speed adjustments in an ad hoc manner to mimic the flock behavior first studied in [56]. Similar dynamics were also implemented in later works [42, 46]. In [43] the authors proposed to use attraction and repulsion forces to mimic the flocking behavior [56] as well as providing obstacle-avoiding capabilities. However, these previous works used such dynamics as a means of smoothing the nodal movements on a fine level, i.e., to avoid collisions with other nodes or obstacles, rather than coordinating individual nodes as to synthesize correlated movements. Therefore, these models either require explicit nodal division into groups [42, 43] or have to synthesize group movements through other means like correlating the selection of destinations [46] or pre-defined patterns [41]. The proposed N-body model differs from earlier works in that all correlated movements are completely and quantitatively controlled by inter-nodal forces.

Compared with previous models, the advantage of our model is three-fold:

- First, it does not require detailed knowledge of the social interaction of the target scenario; therefore it’s simpler and has a wider application scope.

- Second, by synthesizing the heterogeneous inter-nodal distances, the extent of such group-forming tendencies is tractable and therefore, can be quantitatively studied.

- Finally, the dynamics based on Gay-Berne potential mimics the speed matching, group centering and collision avoidance behavior of flock movements studied in [56], thus the nodal movements are more realistic on the fine level.

### 1.4 Extending to Larger Populations

Since our proposed N-body model relies on the numerical inter-nodal relationship matrices, which can only be obtained from a movement trace of a population whose size is no smaller than the target scenario, its application scope is limited. It is desirable that such relationship matrices could be synthesized by extending those obtained from a small population. Such
an improvement could significantly enhance the applicability of N-body model: in order to synthesize the mobility trace for a particular scenario, we can sample the movement information from a small number of individuals, and use N-body to generate the traces for a much larger population. Therefore, following the development of N-body model, we propose a novel method to generate larger relationship matrices based on observations from smaller populations. Specifically, since the relationship matrices are essentially weighted complete networks, and group-moving tendencies are shown as clusters in these networks, we develop an improved configuration model that generates weighted networks according to given clustering and weight information extracted from sample networks.

Network generation and analysis has been a topic of considerable interest in the research community. The earliest network generation algorithm proposed in [59, 60] generates pure random unweighted networks. Such networks, however, are far too simple in comparison to real networks [61], thus are often used as null models in validating certain structural features of real life networks.

Since real life networks, including social networks, metabolic networks and communication networks, exhibit complex structural characteristics such as scale-free [62], small-world phenomenon [34], degree correlation [63] and high cluster-coefficient [33], more sophisticated models have been proposed to generate networks showing those features. A good survey [61] presents an overview of these models.

In [34,35] the authors proposed a simple model to generate unweighted networks with high cluster coefficient and low average distance, thus showing the small-world effect. That model allows only adjusting clustering on a coarse level without fine control on individual nodes.

The configuration model [64,65] was originally developed to generate unweighted networks with given degree distributions. In [66] such a configuration model was extended to generate weighted networks. Later, in [67], a sophisticated configuration model was proposed to generate unweighted network with arbitrary degree distribution and tunable global clustering. Recently a more refined model [68] was developed to generate unweighted network with given degree-degree correlation and degree dependent clustering.

To the knowledge of the authors, there has not been any work done in generating a weighted network with tunable clustering. Therefore, the method proposed in this dissertation is the first attempt in achieving this goal.

1.5 Outline

The rest of the dissertation is organized as follows: In Chapter 2 we conduct the research on the contact time distribution in random walk and random waypoint mobility models. In Chapter 3 we first quantify the metrics for group-forming tendency in mobile environments. We then
present and validate the N-body mobility model. In Chapter 4 we present and validate a novel network generator to extend the group structures observed from small populations to describe larger populations. We conclude this dissertation in Chapter 5.
Chapter 2

Contact Time in Random Walk Mobility Models

In this chapter we conduct a comprehensive study of the contact time distribution in simple mobility models. The theoretical modeling is presented in Section 2.1 and the conclusions are validated in Section 2.2. Although our theoretical modeling is mainly based on Random Walk, the results are shown shown to hold sufficiently well for Random Waypoint in Section 2.2. The conclusion of this research bridges the gap between the two extremes in previous works as surveyed in Section 1.2.

2.1 Model Analysis

In this section the mathematical analysis of the contact time distribution is presented. In Section 2.1.1 we present the basic settings and assumptions. In Section 2.1.2 we briefly review the general derivation of contact time distribution in [27–29] assuming infinite flight lengths. In Section 2.1.3 the power-law tail behavior is investigated assuming both homogeneous and uniform speed distributions. In Section 2.1.4 we consider the impact of finite flight length and reach the conclusion of the power-law-sub-exponential dichotomy.

2.1.1 Model Settings and Assumptions

Random Walk

In the RW model all nodes randomly walk along straight line segments. These walks are called “flights”. For each flight a node travels in a direction $\phi$ at speed $v$ for distance $u$. Afterwards it pauses for time $t_p$, and starts the next flight.
Parameters $\phi$, $v$, $u$ and $t_p$ are independent random variables for each flight. Flights of different nodes are also independent of each other. Usually, the direction $\phi$ is uniformly distributed over $[0, 2\pi]$, and the speed $v$ is uniformly distributed over $[v_{\min}, v_{\max}]$ ($v_{\min} > 0$ in order to obtain a stationary speed distribution [19]). The flight length $u$ and pause time $t_p$ usually follow some common distributions like uniform, exponential or even power-law [12] [5].

Throughout this paper, all these assumptions are applied to RW models, except the pause time $t_p$ is always ignored (fixed to zero). In addition, nodes are confined to a torus, which provides uniform node density and edge wrapping.

Random Waypoint

RWP differs from RW in that for each flight every node selects a waypoint in a confined area as the destination instead of selecting the direction and the distance. Thus in RWP the node density is not uniform [69], the flight directions may not be uniformly distributed, and the flight length distribution is determined by the shape of the confined area [70]. Directions and distances of consecutive flights of the same node are not independent either. Therefore RWP is far more difficult to analyze than RW.

All analysis in this paper is based on RW models, but the conclusions will also be validated against RWP through simulation in Section 2.2. The results show that the conclusions obtained from RW hold sufficiently well for RWP.

Contact Event

Every node is assumed to have the same communication range $r$, which is reasonable under most circumstances. When the distance between two arbitrary nodes $N_i$ and $N_j$ is smaller than $r$, the two nodes are considered to be able to communicate with each other. If during the time interval $[t_0, t_1]$ the nodes $N_i$ and $N_j$ are able to communicate, while at the time instants $t_0^-$ and $t_1^+$ they are not, then the time interval $[t_0, t_1]$ is referred to as a contact event, and its duration $t_1 - t_0$ is defined as the contact time.

2.1.2 Random Walk with Infinite Flight Lengths

We start with the assumption that all flights have infinite flight lengths so that the contact events are modeled as direct traversals as in [27–29]. More accurately, this is the limit case as the nodal flight lengths approach infinity.$^1$ Figure 2.1 shows such a contact event. Nodes $N_i$ and $N_j$ are moving at velocity $\vec{v}_i$ and $\vec{v}_j$, respectively. From $N_j$’s point of view, $N_i$ is moving

$^1$There is subtle difference between infinite and “approaching infinite” flight lengths. With strict infinite flight lengths, nodes never change their speeds so (2.8) does not hold. Since we are more interested in finding the limit when flight lengths approach infinity, we obtain the results here and in Section 2.1.3 assuming (2.8) holds.
at the relative velocity:

\[ \vec{v}_{ij} = \vec{v}_i - \vec{v}_j. \]  

(2.1)

The duration of \( N_i \) traversing \( N_j \)'s contact region (the dashed circle in Fig. 2.1) is:

\[ t_1 = \frac{s}{v_r} = \frac{s}{v_{ij}}, \]  

(2.2)

where \( s \) is the length of the chord, and \( v_r \) is the traversing speed *during a contact event* which equals to the relative speed \( v_{ij} \) (but with different distributions [27–29]), and \( t_1 \) is the contact time assuming infinite flight lengths.

![Figure 2.1: \( N_i \) and \( N_j \) are moving at velocity \( \vec{v}_i \) and \( \vec{v}_j \). Both with communication range \( r \). From \( N_j \)'s perspective, \( N_i \) is traversing \( N_j \)'s contact region (the dashed circle) at the relative velocity \( \vec{v}_{ij} \) along the thick chord \( s \).](image)

Derivation of the distribution of \( t_1 \) requires the distribution of both \( s \) and \( v_r \). Its general derivation is already done in [27–29]. Due to page limit we just show the results here (with slightly different notation that conforms to ours):

The PDF of the traversal duration \( t_1 \) assuming infinite flight lengths:

\[ f_{T_1}(t_1) = \int_0^{2v_{\text{max}}} f_{v_r}(v_r) f_S(v_r t_1) v_r \, dv_r, \]  

(2.3)

where \( f_S \) is the PDF of the chord length \( s \):

\[ f_S(s) = \frac{s}{2r \sqrt{4r^2 - s^2}}, \]  

(2.4)
and $f_{V_i}$ is the PDF of the traversal speed $v_i$:

$$f_{V_i}(v_i) = \frac{v_i f_{V_{ij}}(v_i)}{\int_0^{2v_{\text{max}}} v_i f_{V_{ij}}(v_i) dv_{ij}}, \quad (2.5)$$

where $f_{V_{ij}}$ is the PDF of the relative speed $v_{ij}$ in (2.1) and (2.2):

$$f_{V_{ij}}(v_{ij}) = \int_{v_{\text{min}}}^{v_{\text{max}}} \int_{v_{L}}^{v_{H}} \frac{f_V(v_i) f_V(v_j)}{\pi} \left| \frac{\partial \Delta \phi}{\partial v_{ij}} \right| dv_i dv_j, \quad (2.6)$$

where $f_V$ is the speed distribution, $v_H$ and $v_L$ are integral limits, and $v_i, v_j, v_{ij}, \Delta \phi$ has the relationship:

$$v_{ij} = \sqrt{v_i^2 + v_j^2 - 2v_i v_j \cos(\Delta \phi)}. \quad (2.7)$$

The $f_V$ in (2.6) is the instantaneous speed distribution. Usually there is no closed form solution for (2.6)$^2$, and it is obtained by sampling nodes from the model. In RW or RWP there is also another widely used definition for speed distribution: the one that nodes follow when choosing speeds at the beginning of each flight (usually uniform distribution for RW and RWP). We denote the former with $f_V$ and the latter with $g_V$. They have the following relationship which can be easily derived from the results in [19] (A detailed derivation is provided in Appendix A):

$$f_V(v) = \frac{1}{\int_{v_{\text{min}}}^{v_{\text{max}}} g_V(v) dv} g_V(v), \quad (2.8)$$

### 2.1.3 Tail Behavior of Contact Time Distribution

Generally the PDF of contact time in RW can only be obtained numerically through (2.3). However, most implementation of RW models assume homogeneous speeds or uniform speeds, thus theirs the contact time distribution can be shown to have a power-law tail.

**Random Walk with Homogeneous Speeds**

When every node is moving at the same speed $v_0$, the derivation is greatly simplified. A closed form of the PDF of contact time has been derived in [29]:

$$f_{T_1}(t_1) = \frac{1}{4} \left( \frac{1}{t_0} + \frac{t_0}{t_1^2} \right) \log \frac{t_0 + t_1}{|t_0 - t_1|} - \frac{1}{2t_1}, \quad (2.9)$$

$^2$Alternatively, there is a more intuitive way to derive the PDF of $f_{V_{ij}}(v_{ij})$ through polar coordinates. See Appendix B for details.
where \( t_0 = r/v_0 \), \( r \) is the contact range. The PDF (2.9) has a maximum at \( t_1 = t_0 \). By applying Taylor series the PDF (2.9) will follow a power law distribution in its tail:

\[
f_{t_1}(t_1) \approx \frac{1}{4} \left( \frac{1}{t_0} + \frac{t_0}{t_1^2} \right) \left( 2 \frac{t_0}{t_1} + 2 \frac{t_0}{t_1^3} \right)^3 - \frac{1}{2t_1} \\
\approx \frac{2t_0^2}{3t_1^2} \quad \text{when} \quad t_1 \gg t_0. \tag{2.10}
\]

**Random Walk with Uniform Speeds**

In most RW models nodes choose their speeds according to a uniform distribution on \([v_{\text{min}}, v_{\text{max}}]\) at the beginning of their flights. Therefore from (2.8) the instantaneous nodal speed distribution is:

\[
g_V(v) = \frac{1}{v_{\text{max}} - v_{\text{min}}}, \tag{2.11}
\]

\[
f_V(v) = \frac{1}{v \log H}, \quad \text{where} \quad H = \frac{v_{\text{max}}}{v_{\text{min}}}. \tag{2.12}
\]

From (2.7) the partial derivative in (2.6) can be determined:

\[
\Delta \phi = \cos^{-1} \frac{v_i^2 + v_j^2 - v_{ij}^2}{2v_i v_j}, \tag{2.13}
\]

\[
\frac{\partial \Delta \phi}{\partial v_{ij}} = \frac{2v_{ij}}{\sqrt{-v_i^4 - v_j^4 - v_{ij}^4 + 2v_i^2 v_j^2 + 2v_{ij}^2 v_i v_j + 2v_{ij}^2 v_i^2 v_j^2}}. \tag{2.14}
\]

Substitute (2.12) and (2.14) into (2.6), the PDF of the relative speed \( v_{ij} \) is:

\[
f_{V_{ij}}(v_{ij}) = \int_{v_{\text{min}}}^{v_{\text{max}}} \int_{v_L}^{v_H} \frac{2v_{ij}dv_jdv_i}{\pi v_i v_j \log^2 H \sqrt{-v_i^4 - v_j^4 - v_{ij}^4 + 2v_i^2 v_j^2 + 2v_{ij}^2 v_i v_j + 2v_{ij}^2 v_i^2 v_j^2}}. \tag{2.15}
\]

The integral limits \( v_L \) and \( v_H \) in (2.15) are determined by not only \( v_{\text{min}} \) and \( v_{\text{max}} \), but also the triangle law among \( v_i, v_j \) and \( v_{ij} \):

\[
\begin{align*}
\{ & \ v_i + v_j \geq v_{ij}, \\
\ v_i + v_{ij} \geq v_j, \\
\ v_j + v_{ij} \geq v_i. \}
\end{align*} \tag{2.16}
\]

Therefore the interval varies as \( v_{ij} \) varies. Since only the tail behavior of the PDF of contact time is interested, \( v_{ij} \) is assumed to be small \((v_{ij} \leq \min\{ (v_{\text{max}} - v_{\text{min}})/2, v_{\text{min}} \})\) and the corresponding interval is the gray hexagon in Fig. 2.2. To facilitate the derivation, the upper and lower
bound of (2.15) are obtained by integration over the vertically hatched parallelogram and the horizontally hatched parallelogram in Fig. 2.2, respectively:

\[
\begin{align*}
    f_{V_{ij}}(v_{ij}) < & \int_{v_{min}}^{v_{max}} \int_{v_{j}-v_{ij}}^{v_{j}+v_{ij}} f_{V_i, V_j, V_{ij}}(v_i, v_j, v_{ij}) \, dv_i \, dv_j \\
    & \frac{\log \frac{v_{max}^2 - v_{ij}^2}{v_{min}^2 - v_{ij}^2} - 2 \log H}{2v_{ij} \log^2 H},
\end{align*}
\]

(2.17)

\[
\begin{align*}
    f_{V_{ij}}(v_{ij}) > & \int_{v_{min}+v_{ij}}^{v_{max}+v_{ij}} \int_{v_{j}-v_{ij}}^{v_{j}+v_{ij}} f_{V_i, V_j, V_{ij}}(v_i, v_j, v_{ij}) \, dv_i \, dv_j \\
    & \frac{\log \frac{v_{max}^2 - 2v_{ij} v_{max}}{v_{min}^2 + 2v_{ij} v_{min}} - 2 \log \frac{v_{max} - v_{ij}}{v_{min} + v_{ij}}}{2v_{ij} \log^2 H}.
\end{align*}
\]

(2.18)

Applying the Taylor series:

\[
\begin{align*}
    f_{V_{ij}}(v_{ij}) < & \frac{v_{ij} \left( \frac{1}{v_{min}^2} - \frac{1}{v_{max}^2} \right)}{2 \log^2 H} + O(v_{ij}^3),
\end{align*}
\]

(2.19)

\[
\begin{align*}
    f_{V_{ij}}(v_{ij}) > & \frac{v_{ij} \left( \frac{1}{v_{min}^2} - \frac{1}{v_{max}^2} \right) - 2v_{ij}^2 \left( \frac{1}{v_{min}^2} - \frac{1}{v_{max}^2} \right)}{2 \log^2 H} + O(v_{ij}^3).
\end{align*}
\]

(2.20)

Substituting (2.19) and (2.20) into (2.5), the corresponding upper and lower bound for the PDF of traversal speeds during contact events \( f_{V_t}(v_t) \) are:

\[
\begin{align*}
    f_{V_t}(v_t) < & K_1 v_t^2 + O(v_t^4), \\
    f_{V_t}(v_t) > & K_1 v_t^2 - K_2 v_t^3 + O(v_t^4),
\end{align*}
\]

(2.21) \hspace{1cm} (2.22)

where \( K_1 \) and \( K_2 \) are constants. As \( v_t \to 0 \), the upper and lower bounds converge to:

\[
f_{V_t}(v_t) \approx K_1 v_t^2 + O(v_t^3).
\]

(2.23)

Substituting (2.23) and (2.4) into (2.3), the asymptotic distribution of contact time \( t_1 \) as \( t_1 \to \infty \) is:

\[
f_{T_1}(t_1) \approx \frac{K}{t_1^4},
\]

(2.24)

where \( K \) is a constant.

From (2.24) and (2.10) the contact time \( t_1 \) follows a power-law of \( t_1^{-4} \) with uniform speed distribution, while a power-law of \( t_1^{-3} \) with homogeneous speed. Simulation in Section 2.2.1 will show that the actual tail of the PDF of \( t_1 \) with uniform speed distribution will first follow a power-law of \( t_1^{-3} \), then switches to \( t_1^{-4} \). With larger \( v_{min} \) and \( v_{max}/v_{min} \) ratio, the transition
takes place sooner.

2.1.4 Random Walk with Finite Flight Lengths

The analysis in previous sections is based on the assumption that nodal flight distances approach infinity such that contact events can be modeled as direct traversals. However, for most real scenarios and practical RW and RWP implementations, the flight lengths are finite. Typical RW models may have flight lengths following common distributions, like uniform, exponential, power-law, etc. Recent studies [5, 12] suggest that real world human movements can be approximated by truncated levy walks.

Regardless of how flight lengths are distributed, some traversing events will be interrupted because one or both nodes finish their current flights and change directions during the traversal. Intuitively, the longer a traversing event is, the more likely it be interrupted. When such interruption rate has reached a certain extent, the direct traversal assumption no longer holds and the contact events have to be modeled differently, e.g., as the sum of random walks.

Traversal Survival Rate

When all nodes have infinite flight lengths, assume a traversing event has duration $t_1$. When flight lengths are no longer infinite, we define the survival rate as the probability that such a traversal not be interrupted, which is a function of $t_1$, denoted with $\sigma(t_1)$.

At the beginning of such a traversal, both nodes are on their own flights. The traversal will “survive” only if $t_1$ is no greater than both residual flight times, where by residual flight time we refer to the remaining time in the current flight. As shown in Fig. 2.3, $N_1$ is on its flight with distance $u$ and speed $v_0$, while $N_2$ and $N_3$ are traversing the contact region of $N_1$. Both traversals have the same duration $t_1$ if not interrupted. We further assume $N_2$ and $N_3$ have infinite flight lengths. Therefore, the traversal between $N_3$ and $N_1$ is interrupted since $N_1$ finishes its flight before the traversal ends, while the one between $N_2$ and $N_1$ is “survived”. We denote such a survival rate when only one node has finite flight lengths with $\mu(t_1)$. Since nodes move independently, we have:

$$\sigma(t_1) = \mu^2(t_1).$$

Intuitively as shown in Fig. 2.3, all such traversals with duration $t_1$ will be interrupted because of $N_1$ only if $N_1$ is in the gray segment at the beginning of the traversals. Assuming such traversals occur at the same frequency along that flight, the survival rate along that flight

---

3This assumption is true for RW that has uniform node density, but not strictly true for RWP, which has higher density in the center [69]. Simulation results show that this assumption still holds sufficiently well for RWP.
Figure 2.2: The integral interval in (2.15) assuming \( v_{ij} \leq v_{\text{min}} \). When \( v_{ij} \) is given, \( v_i \) and \( v_j \) have to satisfy (2.16) so the integral interval is the gray hexagon. Alternatively, the upper and lower bound of (2.15) can be obtained by integration over the horizontally hatched area and vertically hatched area, respectively.

Figure 2.3: Nodes \( N_2 \) and \( N_3 \) are both traversing the contact region of \( N_1 \). Both events will have the same duration \( t_1 \) if not interrupted. But the traversal between \( N_3 \) and \( N_1 \) is interrupted since \( N_1 \) finishes its flight before the traversal ends.
with flight time $\tau = u/v_0$ is:

$$\mu(t_1|\tau) = \begin{cases} \frac{\tau-t_1}{\tau} & t_1 < \tau \\ 0 & t_1 \geq \tau \end{cases}.$$  \hspace{1cm} (2.26)

Further, we assume that the occurrence density of traversing events of the same duration $t_1$ is the same everywhere regardless of flight times or speeds. We denote this density with $\eta(t_1)$, which is proportional to the PDF of $t_1$ in (2.3) ((2.3) is in fact the normalized form of $\eta(t_1)$). Therefore the survival rate from a single node’s perspective $\mu(t_1)$ can be derived:

$$\mu(t_1) = \frac{\int_{t_1}^{\tau_{\text{max}}} \eta(t_1) (\tau-t_1) f_\tau(\tau) d\tau}{\int_{t_1}^{\tau_{\text{max}}} \eta(t_1) f_\tau(\tau) d\tau} = \frac{\int_{t_1}^{\tau_{\text{max}}} (\tau-t_1) f_\tau(\tau) d\tau}{E(\tau)},$$ \hspace{1cm} (2.27)

where $f_\tau$ is the PDF of flight time. Substitute (2.27) into (2.25) the survival rate is:

$$\sigma(t_1) = \mu^2(t_1) = \left( \frac{\int_{t_1}^{\tau_{\text{max}}} (\tau-t_1) f_\tau(\tau) d\tau}{E(\tau)} \right)^2.$$ \hspace{1cm} (2.28)

It is hard to directly verify (2.28) through simulation. Practically (2.28) can be verified more accurately through the duration distribution of the “survived” traversals, which is denoted with $t_S$:

$$f_{T_S}(t_S) = \frac{f_{T_1}(t_S) \sigma(t_S)}{\int_0^{\tau_{\text{max}}} f_{T_1}(t_S) \sigma(t_S) dt_S}.$$ \hspace{1cm} (2.29)

**Dichotomy in the Tail**

The survival rate $\sigma(t_1)$ calculated in (2.28) is a monotonically decreasing function. When the traversal duration $t_1$ reaches a critical point $t_C$, the survival rate $\sigma$ falls below a certain threshold $\gamma$:

$$\sigma(t_C) = \gamma.$$ \hspace{1cm} (2.30)

For RW with homogeneous speeds $v_0$, the critical point $t_C$ has closed form when the flight length $u$ follows some common distributions:

$$t_C = \begin{cases} \frac{u_0(1-\sqrt{\gamma})}{v_0} & u \text{ fixed to } u_0, \\ \frac{2u_0}{v_0} (1 - \sqrt{\gamma}) & f_U(u) = \frac{1}{2u_0}, \ 0 \leq u \leq 2u_0, \\ -\frac{u_0}{2v_0} \log \gamma & f_U(u) = \frac{1}{u_0} e^{-\frac{u}{u_0}}. \end{cases}$$ \hspace{1cm} (2.31)

---

4This assumption is strictly true only for RW with homogeneous speeds. But the results are shown to be good enough for general RW and RWP models in simulation.
The threshold $\gamma$ is determined empirically. Our simulation results suggest $\gamma \leq 50\%$ for most RW and RWP models.

Most of the traversals with duration $t_t \geq t_C$ are interrupted and their actual contact time $t$ may follow some other distribution. Some of them may have the contact time $t < t_C$. Since the PDF of $t_t$ is fast dropping ($t_t^{-3}$ or $t_t^{-4}$), the total number of events with traversal duration $t_t \geq t_C$ is small enough in comparison to the shorter ones. Therefore the distortion to the power-law distribution of contact times $t \leq t_C$ can be neglected. Thus, the tail behavior of contact times will show a dichotomy: power-law before $t_C$, and something else thereafter.

In [71] the authors used a finite time renewing process to obtain the upper bound of inter-contact time distribution. The same technique can also be applied here to obtain the upper bound of the PDF of contact time $t$ when $t > t_C$. Assume nodes are moving on a finitely bounded surface with average flight time $E[\tau]$. Consider a node $N_1$ at the position $X_1(t_0)$ at time $t_0$. After a period $t_R \gg E[\tau]$, it is reasonable to conclude that $N_1$’s position $X_1(t_0 + T)$ is independent of $X_1(t_0)$. From the stationary nodal distribution the probability that two nodes are within the contact range of one another is $p_c$. Thus the probability that two nodes $N_1$ and $N_2$ kept contact with each other for time $T$ has an upper bound:

$$P(t \geq T) \leq p_c^{T/t_R},$$

which follows an exponential distribution. The parameter $t_R$ is determined by both the boundary of the surface and the average flight time $E[\tau]$. The probability $p_c$ is determined by $E[\tau]$ and the communication range $r$. Therefore, we conclude that the PDF of contact time for general RW models will exhibit a \textit{power-law-sub-exponential dichotomy} in the tail. In fact, as simulation results shown in Section 2.2.2, the second half of the dichotomy can be well approximated by an exponential distribution.

Qualitatively, as average flight length $E[\tau]$ gets smaller, the power-law-sub-exponential transition takes place earlier. When the average flight length become so small (usually smaller than the contact range) that the power-law part is no longer distinguishable, the dichotomy degenerates to a single exponential tail. In this case the contact events can be modeled as the sum of consecutive random walks, and a more accurate result is obtained in [32], where the PDF of contact times follows the exponential distribution:

$$f_T(t) \cong e^{-\lambda t},$$  \hspace{1cm} (2.33)

where $\lambda = \frac{E[v]}{r}$.  \hspace{1cm} (2.34)

$E[v]$ is the average nodal speed, and $r$ is the contact range.
2.2 Validation

In this section the results in Section 2.1 are validated through simulation. In Section 2.2.1 we validate the conclusion of power-law tail assuming infinite flight lengths, and in Section 2.2.2 we validate the conclusion of dichotomy.

2.2.1 Infinite Flight Lengths without Pause

To validate the conclusion of power-law tail in Section 2.1.3, we implemented an RW and an RWP model. Since the assumption of infinite flight lengths is equivalent to all direct traversals not being interrupted, for each traversing event we recorded the traversing chord length $s$ and the relative speed $v_r$, and calculated its traversal time $t_I$ through (2.2). By doing this instead of actually implementing infinite flight lengths, the implementation is greatly simplified. In addition, it ensures that (2.8) holds since nodes change their speeds between flights, therefore truly reflects the limit of the PDF of contact time as flight lengths approach infinity.

The RW model was on a $10 \times 10$m torus and the RWP model was on a $10 \times 10$m square. Both models have 100 nodes with communication range $r = 1m$ run for 10 hours without pause. In the RW model all flights have fixed flight length $u = 10m$. In both models nodes select their speeds for each flight uniformly from $[v_{\min}, v_{\max}]$. Both models were run multiple times with different $H = \frac{v_{\max}}{v_{\min}}$, but with the same average speed $v_{\text{avg}}$ derived in [19]:

$$v_{\text{avg}} = \int_{v_{\min}}^{v_{\max}} v f_V(v) dv = \frac{v_{\max} - v_{\min}}{\log H}.$$  \hspace{1cm} (2.35)

In our simulation $v_{\text{avg}} = 1m/s$. When $H = 1$ the models are equivalent to homogeneous speeds with speed $v_0 = 1m/s$.

The CCDFs of traversal duration $t_I$ are shown in Fig. 2.4 on logarithmic scale. It is clearly shown that all CCDFs have power-law tail. With heterogeneous speeds the distribution functions first approach $t_I^{-3}$ as with homogeneous speeds ($H = 1$), then switch to $t_I^{-4}$ at some time. As $H$ becomes larger, the distribution functions switch to $t_I^{-4}$ earlier.

2.2.2 Finite Flight Lengths without Pause

The same models are used as in the previous section to validate the dichotomy conclusion in Section 2.1.4. Since the flight lengths are finite, we no longer calculate the traversal duration $t_I$, but instead directly measure the contact times $t$. We also diversify the random walk model by implementing different flight lengths: one RW model with uniform flight lengths on $[0, 2u]$, denoted with RWA; and one with exponential flight lengths with mean $u$, denoted with RWB. For the RWP model, we diversify its flight lengths by varying its area $m \times m$. In addition, the
contact range \( r \) is also set to different values to compare the results.

**Traversal Survival Rate**

We start by validating the traversal survival rate \( \sigma(t_I) \) through the uninterrupted traversal duration \( t_S \), whose PDF is derived in (2.29). The CCDFs of \( t_S \) are shown on logarithmic scale in Fig. 2.5 together with the theoretical results. It is shown that for all RW models with homogeneous speeds, the simulation results are very close to the theoretical predictions; but for RW with heterogeneous speeds or RWP models, there are slight differences between the theoretical and simulation results. That is due to the simplifying assumptions do not strictly hold true for RWP or heterogeneous speeds as discussed in Section 2.1.4. However, from the results shown, (2.28) still holds sufficiently well to reach the conclusion of dichotomy even with RWP or heterogeneous speed.

**Dichotomy in the Tail**

The CCDFs of the contact time are plotted in Fig. 2.6 and Fig. 2.7 for models with different speed parameter \( H \), contact range \( r \) and flight length \( u \) in both logarithmic scale and semi logarithmic scale. The critical points \( t_C \) where the survival rate \( \sigma(t_C) = 50\% \) are also plotted as the vertical lines. From the logarithm plots the dichotomy is clearly shown in the tail of the CCDFs. When the same plots are shown on semi logarithm plots, the straight lines in the tail suggest the sub-exponential half of the dichotomy may be very close to exponential.
Figure 2.5: CCDF of uninterrupted traversal duration $t_S$. Speed parameter $H = 1$ (homogeneous speeds) for (a) (b) (c) and $H = 10$ for (d) (e) (f). Mobility models: (a) (d) RWP, (b) (e) RWA, (c) (f) RWB. All with contact range $r = 1m$.  

21
Figure 2.6: CCDF of contact times for models with homogeneous speeds ($H = 1$) on logarithm plots (a) (b) (c) and semi logarithm plots (d) (e) (f). The vertical lines are the critical points $t_C$ where the survival rate $\sigma(t_C) = 50\%$. Mobility models: (a) (d) RWP, (b) (e) RWA, (c) (f) RWB.
Figure 2.7: CCDF of contact times for models with heterogeneous speeds ($H = 10$) on logarithm plots (a) (b) (c) and semi logarithm plots (d) (e) (f). The vertical lines are the critical points $t_C$ where the survival rate $\sigma(t_C) = 50\%$. Mobility models: (a) (d) RWP, (b) (e) RWA, (c) (f) RWB.
Chapter 3

N-Body Mobility Model

In this chapter we present the N-body mobility model that extracts group-forming tendencies in real life movements from sample traces and reproduces them in synthesized traces. Such social aspects are quantified in Section 3.1, and reproduced using the N-body model introduced in Section 3.2. Results are validated in Section 3.3.

3.1 Social Metrics

To synthesize mobility traces that mimic the heterogeneity of real life movements, the nodes’ group-forming tendencies need to be quantified from observations of real movement traces for later reproduction. Intuitively, for nodes moving in certain groups, their relationships can be reflected in the statistics of the pairwise inter-nodal distances among the group members and between the other nodes.

3.1.1 Friendship / Steadiness Matrices

In the following scenario we assume a total of \( N \) nodes \( n_i, \ i = 1, \ldots, N \), moving in the vector space \( \Omega \) with their positions at time \( t \) being \( \vec{P}_k(t) \in \Omega \). The pairwise distance between two arbitrary nodes \( n_i \) and \( n_j \) is then the following random process:

\[
D_{ij}(t) = \left| \vec{P}_i(t) - \vec{P}_j(t) \right|,
\]  

whose mean \( M_{i,j} \) and normalized standard deviation (NSD) \( S_{i,j} \) can be estimated from the recorded traces, which often consist of position information at discrete time points, e.g., positions sampled at a total of \( L \) time points \( t_l = l\Delta t, \ l = 1,2,\ldots L \) with equal intervals \( \Delta t \). Assuming the nodal movements are stationary, the mean and NSD can be estimated through
the sample mean and variance:

\[ M_{i,j} \approx \bar{M}_{i,j} = \frac{1}{L} \sum_{l=1}^{L} |\bar{P}_i(t_l) - \bar{P}_j(t_l)|, \quad (3.2) \]

\[ S_{i,j} \approx \bar{S}_{i,j} = \sqrt{\frac{\sum_{k=1}^{L} |\bar{P}_i(t_l) - \bar{P}_j(t_l)|^2}{(L-1)\bar{M}_{i,j}^2}} - \frac{L}{L-1}. \quad (3.3) \]

To capture the group-forming tendency, we compare \( M_{i,j} \) and \( S_{i,j} \) with those of independent random placement, which is the distance between nodes when they are independently placed according to the stationary nodal distribution. The corresponding mean and NSD is denoted by \( M^{\text{ind}} \) and \( S^{\text{ind}} \):

\[ M^{\text{ind}} = \int \int_{\Omega^2} |\vec{P} - \vec{P}'| \lambda(\vec{P})\lambda(\vec{P}')d\vec{P}d\vec{P}', \quad (3.4) \]

\[ S^{\text{ind}} = \sqrt{\frac{\int \int_{\Omega^2} |\vec{P} - \vec{P}'|^2 \lambda(\vec{P})\lambda(\vec{P}')d\vec{P}d\vec{P}'}{M^{\text{ind}}^2}} - 1, \quad (3.5) \]

where \( \vec{P} \) and \( \vec{P}' \) are two independent random vectors following the stationary nodal distribution \( \lambda \). For convenience, we define the friendship index \( r_{i,j} \):

\[ r_{i,j} = 1 - \frac{M_{i,j}}{M^{\text{ind}}}, \quad r_{i,j} \in (-\infty, 1], \quad (3.6) \]

and the steadiness index \( w_{i,j} \):

\[ w_{i,j} = \frac{S_{i,j}}{S^{\text{ind}}}, \quad w_{i,j} \in [0, \infty). \quad (3.7) \]

By inspecting the friendship and steadiness index \( r_{i,j} \) and \( w_{i,j} \) the static and dynamic aspects of the pairwise group-forming tendency can be revealed. When \( r_{i,j} > 0 \), the two nodes \( n_i \) and \( n_j \) are closer to each other than one would expect from two independent nodes of the same stationary distribution. Therefore they show a positive tendency of moving in a group. On the contrary, when \( r_{i,j} < 0 \), the negative tendency could be interpreted as the two nodes are trying to avoid contact with each other. On the other hand, when two nodes are moving in a group, while the mean of their distance indicates the geographical size of the group, the NSD reflects the steadiness of the group. For example, node pairs with \( w_{i,j} < 1 \) are likely to stick to their positions inside the group rather than wander around, and vice versa.

Since both the pairwise distance \( M_{i,j} \) and that of independent random placement \( M_{i,j}^{\text{ind}} \) are of the same stationary distribution, any simple population concentration, e.g., geo-
graphical hotspots like airports or shopping centers in a city, do not affect $r_{i,j}$, while certain restrictions that affect movement patterns do, e.g., road restrictions force the nodes to follow certain routes, thus nodes on the same direction of the same route have shorter pairwise distances. It is shown in Section 3.3.2 when comparing performance results between the original traces, the N-body models and modified random waypoint models with empirical hotspots, that simply incorporating hotspots does not account for the pairwise heterogeneity.

For an $N$ node system, the friendship and steadiness index from all pairs of nodes comprise the $N \times N$ friendship matrix and steadiness matrix, respectively:

$$\mathbf{R} = \begin{pmatrix} r_{1,1} & \ldots & r_{1,N} \\ \vdots & \ddots & \vdots \\ r_{N,1} & \ldots & r_{N,N} \end{pmatrix}, \quad \quad (3.8)$$

$$\mathbf{W} = \begin{pmatrix} w_{1,1} & \ldots & w_{1,N} \\ \vdots & \ddots & \vdots \\ w_{N,1} & \ldots & w_{N,N} \end{pmatrix}. \quad \quad (3.9)$$

Both matrices are symmetric with the diagonal elements undefined.

### 3.1.2 Sample Traces

We extracted these matrices from RWP and RPGM as well as two real life movement traces: the bus movement traces in Seattle from Ad-Hoc city project [72] and taxicab movement traces in San Francisco from Cabspotting project [73]. For all four models, we consider 48-50 nodes running for 16 hours with the sample rate of 4 seconds per sample. For real traces, samples are obtained through linear interpolation.

The Seattle bus trace (STBUS) contains GPS logs of more than 1200 buses running 239 routes for around 20 days. Bus locations are updated about every 2 minutes while the buses are running. We extracted flights out of the traces, which are continuous records of a bus (with a maximum time gap of 4 minutes between consecutive records) running on a single route. Among them we chose 48 longest flights running 7 routes for at least 16 hours. These flights are then time-shifted as if they are running at the same time. Therefore, such a trace will contain a higher level of group-forming tendency between buses from the same routes.

The SF cab trace (SFCAB) contains GPS logs of 536 yellow cabs for over 30 days. Cab locations are updated almost every minute if the cabs stay online. Among the traces we selected 50 cars that stayed online at the same time for at least 16 hours. Most of the cabs should move independently, however, due to the geographical restraints (like highways in the city), there exists some modest level of group-forming tendencies.
For the RWP and RPGM model, we set them as both running on a 20 × 20 km plane, which resembles the dimensions of a downtown like Seattle or SF. For RWP we had 50 individual nodes choosing their speeds uniformly from 3 to 12 m/s, which is the typical city speeds. For RPGM, we divided the 48 nodes into 4 groups. Each group has 12 nodes with a radius of 500 m for two groups and 2 km for the other two. All groups move at 5 m/s and nodes randomly drift within the groups’ radius also at 5 m/s. No pause was implemented for both RWP and RPGM.

Figure 3.1: CDF of Friendship Index \( r_{ij} \), (a) for all 4 traces: RPM, RPGM, SFCAB and STBUS, and (b) for flights from the same / different routes in STBUS trace.

We plotted the cumulative distribution functions (CDF) of the pairwise friendship index \( r_{ij} \) in Fig. 3.1(a), and all four traces show results as expected: RWP has its friendship indexes concentrated around 0, indicating homogeneous pairwise relationship among nodes without any group-forming tendency. For RPGM, from the upper right steps it is clear that node pairs from the same groups have different levels of positive group-forming tendency, which is related to the group radius. Node pairs from different groups, however, have their \( r_{ij} \) concentrated around 0 just like in RWP\(^1\). The SFCAB trace is similar to RWP, but with an increased level of heterogeneity as well as higher group-forming tendency because of the geographical restrictions as mentioned above. The STBUS trace shows an even higher level of heterogeneity. To better understand this trace, we separated its \( r_{ij} \) data into those running the same routes and those running different routes, and plotted their CDFs in Fig. 3.1(b). Compared to the buses running different routes, those running the same routes have an average much higher group-

\(^{1}\)It also shows some steps around 0, which is due to the fact that the group radius is comparable to the simulation area, therefore result in different levels in average pairwise distance depending on the group membership of both nodes.
forming tendencies, although there are some pairs showing negative tendencies, which is caused by imperfect trace processing as some flights are not correctly aligned in time. On the other hand, those running different routes show the highest heterogeneity among all traces, while on average, they maintain a balance between positive and negative group-forming tendencies. This is due to the fact that different routes are restricted to different portions of the city, and therefore, there are heterogeneous, but evenly distributed average distances between routes.

Figure 3.2: CDF of Steadiness Index $w_{ij}$, (a) for all 4 traces: RPM, RPGM, SFCAB and STBUS, and (b) for flights from the same / different routes in STBUS trace.

Similar to the friendship index, we also plotted the CDFs of the pairwise steadiness index $w_{i,j}$ in Fig. 3.2(a). The figure shows that RWP has its steadiness indexes converging towards 1, which is to be expected from a homogeneous model. RPGM has similar distribution to RWP, due to its similar dynamics with that of RWP for both nodes from different groups and nodes from the same groups, regardless of the group radius. Both SFCAB and STBUS traces have the steadiness indexes $w_{i,j}$ spanning a wider range, which indicates higher diversity in their NSD than RWP or RPGM. Similar to Fig. 3.1(b), Fig. 3.2(b) shows the CDFs of steadiness index $w_{i,j}$ for STBUS trace with buses running the same routes and those running different routes. From the figure the buses running different routes show smaller NSD in their distances, which is again due to the fact that different routes are restricted to different geographical regions, therefore their distances would often have a rather small variance when compared to the average distances. For buses running the same routes, if they are not correctly aligned in time, their distance could have relatively large variance compared to their average distance.
3.2 N-Body Mobility Model

To quantitatively reproduce the social structures as observed in the target scenario, which can be either real or synthesized traces, in this section we propose the N-body mobility model. Unlike the previous works [21,38–53], the proposed N-body model does not require any understanding of the underlying social structure of the target scenario, but rather, it takes the friendship matrix \( R \) and steadiness matrix \( W \) extracted from the target trace as inputs, and generate movement traces that converge to the target matrices.

3.2.1 Basic Dynamics

The dynamics of N-body model are based on that of RWP [8] (may incorporate empirical results into parameters like speed, pause, hotspots as in [16–18]) but with movements regulated by competing attractions towards destinations as well as mutual forces between nodes. By implementing such a multi-body interaction, based on N-body dynamics [57], users can fine-tune the group-forming tendencies between nodes towards the desired targets.

![Figure 3.3: Two nodes \( n_i \) and \( n_j \) with a strong grouping tendency are attracted to each other by the inter-nodal force \( F_{i,j} \), thus deviating from their normal route and move together for a while, like two friends talking to each other for a while on their way to work.](image)

At the start of the simulation, all nodes randomly select their destinations like in RWP (or according to empirical popularity as in [16–18]). The nodes are then attracted towards their destinations. They also interact (via forces) with other nodes such that their actual trajectories may deviate from their normal straight routes as shown in Fig. 3.3. Once the nodes reached the destinations they pause for some time and move on to their next destinations.

Since nodal movements are force driven, the N-body model is implemented as a time stepping
simulator using the improved Euler’s method \[74\] with step time $\Delta t$. The basic dynamics are described below:

1. At time $t = 0$ each node $n_i$ is randomly placed at $\vec{P}_i$ inside the simulation area and is marked as ready (nodes can be in one of the three states: ready, in-flight, in-pause).

2. Randomly (or according to some distribution) select the next waypoint for each node marked ready. Mark these nodes as in-flight.

3. Calculate the forces imposed on each node $n_i$ marked in-flight:

$$\vec{F}_i = \vec{G}_i + \sum_{j \neq i} \vec{F}_{j,i} = g(\vec{P}_i, i = 1 \ldots N),$$

where $\vec{G}_i$ is the attraction force from the destination, and $\vec{F}_{j,i}$ is the inter-nodal force from another node $n_j$. Update its position $\vec{P}_i$ and velocity $\vec{v}_i$ over this time step using the improved Euler’s method:

$$\vec{v}_i(t + \Delta t) = f(\vec{v}_i(t), \vec{F}^{**}_i(t)),$$

$$\vec{P}_i(t + \Delta t) = \vec{P}_i(t) + 0.5\vec{v}_i(t)\Delta t + 0.5\vec{v}_i(t + \Delta t)\Delta t,$$

where $\vec{F}^{**}_i(t)$ is the average force over the time step:

$$\vec{F}^{**}_i(t) = 0.5\vec{F}_i(t) + 0.5\vec{F}^*_i(t + \Delta t),$$

where $\vec{F}^*_i(t + \Delta t)$ is the estimated force at the end of the step:

$$\vec{F}^*_i(t + \Delta t) = g(\vec{P}^*_i(t + \Delta t), i = 1 \ldots N),$$

$$\vec{P}^*_i(t + \Delta t) = \vec{P}_i(t) + 0.5\vec{v}_i(t)\Delta t + 0.5\vec{v}^*_i(t + \Delta t)\Delta t,$$

$$\vec{v}^*_i(t + \Delta t) = f(\vec{v}_i(t), \vec{F}_i(t)).$$

Compared to the classical Newton’s method, the improved Euler’s method provides better accuracy at the same time granularity \[74\]. Derivation of the velocity in (3.11) and (3.16) will be further discussed in Section 3.2.2.

4. If any in-flight nodes have reached their destinations (or within some threshold distance), mark them as in-pause. Randomly choose a pause time for each node.

5. If any in-pause nodes have reached their pause time, mark them as ready.

6. Step the time forward $t = t + \Delta t$. Go to step 2.
3.2.2 Attraction Forces and Speed Control

Figure 3.4: Several nodes are attracted to each other and formed a clog. They will move as a whole and visit the destinations according to their distances.

For an arbitrary node $n_i$, we denote the attraction force from its destination as $\vec{G}_i$, and the inter-nodal force from another node $n_j$ as $\vec{F}_{j,i}$ ($\vec{F}_{j,i} = -\vec{F}_{i,j}$ since they are symmetric). If the friendship index $r_{i,j}$ is positive, the inter-nodal forces $\vec{F}_{j,i}$ and $\vec{F}_{i,j}$ will pull the nodes together so that they will be closer to each other as shown in Fig. 3.3. Intuitively, higher inter-nodal forces will result in lower average distance and higher $r_{i,j}$. However, for very high friendship indexes $r_{i,j}$, the inter-nodal forces could become so strong that the two nodes will “stick” to each other and cannot be separated by the attractions to their destinations, and they will move together in a group as a result of the attraction forces to the destinations competing with each other. To avoid tie-ups, the destination attraction forces are set to be inverse proportional to the square of the distance between the nodes and their destinations$^2$:

$$\vec{G}_i = G_0 \frac{|\vec{d}_0|}{|\vec{d}_i|} \hat{d}_i,$$

(3.17)

where $\vec{d}_i$ is the distance vector from $n_i$ towards its destination and $\hat{d}_i$ is the corresponding unit vector, $G_0$ is the unit force, and $d_0$ is a pre-determined reference distance where nodes receive unit attraction $G_0$, i.e., the mean nodal distance of random placement.

When a group of nodes are locked-in and the attraction forces are competing against each

$^2$In the original version of this model published in [75], the attraction forces were inverse proportional to the distance, rather than the square of the distance. In our later simulations the original setting turned out to be causing node clogging under certain circumstances. Therefore the attraction force is changed to the current form in (3.17). A detailed analysis of this issue is provided in Appendix C.
other, the nearest destination will often win out. If so, the group will visit the destinations in the sequence according to their relative distances as shown in Fig. 3.4. Near-ties are easily broken since the balance is unstable: a small move in any direction will enhance the forces towards that direction while weakening the opposite ones. Once a group reaches a destination and the winning node is pausing, the group may even pause as a whole if the inter-nodal forces between the winning node and the rest of the group are strong.

To control speeds, each node is assigned a maximum speed \( v_{iM} \). A node may accelerate, decelerate or change its direction due to attractions and inter-nodal forces, but its speed never exceeds its limit. Therefore when calculating speeds in (3.11) and (3.16) during each time step, if the speed of a node exceeds its limit, it is truncated to the limit (assuming all nodes have the same unit mass 1):

\[
\vec{v}_i(t + \Delta t) = f(\vec{v}_i(t), \vec{F}_i) \\
= \begin{cases} 
\vec{v}^*_i & \text{if } |\vec{v}^*_i| \leq v_{iM}, \\
v_{iM} \frac{\vec{v}^*_i}{|\vec{v}^*_i|} & \text{if } |\vec{v}^*_i| > v_{iM},
\end{cases}
\]

(3.18)

where \( \vec{v}^*_i = \vec{v}_i(t) + \vec{F}_i \Delta t \),

(3.19)

All nodes may share the same speed limit, may have different ones, or even choose one at the beginning of each flight as in RWP.

Hence, when no inter-nodal forces exists, after some period of acceleration, the node will be moving at its maximum speed \( v_{iM} \) towards the destination. If \( G_0 \) is chosen sufficiently large so that the acceleration time is negligible, and no inter-nodal forces are considered, the N-body model reduces to RWP.

### 3.2.3 Inter-Nodal Forces

The inter-nodal forces must be very carefully determined since they play a central role in adjusting the statistics of pairwise nodal distances. In a recent paper [76] the author pointed out that flock behaviors like collision avoidance, velocity matching and flock centering, which were studied in [56] and applied to mobility models in [41, 43, 46], can be synthesized by implementing the Gay-Berne potential [58], which had been used in modeling inter-molecular forces in chemistry. Inspired by this conclusion, we implemented the inter-nodal forces in a simplified form of the Gay-Berne potential:

\[
\vec{F}_{j,i} = 4G_0 \epsilon \left[ \frac{s}{d_{i,j}} + s - \sigma - \frac{s^2}{(d_{i,j})^2 + s - \sigma} \right] \hat{d}_{i,j},
\]

(3.21)
where $\vec{d}_{i,j}$ denotes the distance vector from node $n_i$ towards $n_j$, and parameters $\epsilon$, $s$ and $\sigma$ shape the relationship between $|\vec{F}_{i,j}|$ and $|\vec{d}_{i,j}|$ as shown in Fig. 3.5.

![Figure 3.5: Inter-nodal force as a function of distance ($\epsilon = 1$, $s = 1$, $\sigma = 2$).](image)

Figure 3.5 shows the features of the inter-nodal force $\vec{F}_{j,i}$ as well as the influence of parameters $\epsilon$, $s$ and $\sigma$. When nodes $n_i$ and $n_j$ are far from each other, i.e., $|\vec{d}_{i,j}| > \sigma$, there is a weak attraction force between them; as they get closer, the force becomes stronger, which implies that nodes are more likely attracted to friends nearby rather than those far away. The attraction force reaches its maximum of $\epsilon G_0$ when $|\vec{d}_{i,j}| = s + \sigma$. Afterwards it decreases, and is reduced to 0 when $|\vec{d}_{i,j}| = \sigma$. As the two nodes get even closer, the attraction turns to repulsion, and approaches infinity when the nodes get too close as $|\vec{d}_{i,j}| \to (\sigma - s)^+$ if $\sigma > s$. Thus $\sigma$ is said to be the *comfort distance* between the two nodes since at $\sigma$ there is neither attraction nor repulsion.

Intuitively, the positive forces between far away nodes pull them together so as to mimic velocity matching and flock centering, while the repulsion forces between closer nodes avoid collisions [56, 76]. While $\sigma$ sets the optimal distance between two nodes, $s$ indicates the maximum span of this attraction-repulsion force, thus they are directly related to the mean and NSD of the pairwise nodal distances. The parameter $\epsilon$ sets the strength of such force in influencing nodal movements. Therefore, to finely control the pairwise nodal distances such that they may approach the desired mean and NSD, a unique set of the parameters $\epsilon$, $s$ and $\sigma$ need to be assigned to each pair of nodes, namely, $\epsilon_{i,j}$, $s_{i,j}$ and $\sigma_{i,j}$. For convenience, we define the *interaction vector*:

$$\eta_{i,j} = (\epsilon_{i,j}, s_{i,j}, \sigma_{i,j}).$$  \hspace{1cm} (3.22)
All such vectors in the \( N \) node system comprise a \( N \times N \) vector matrix:

\[
Q = \begin{pmatrix}
\eta_{1,1} & \cdots & \eta_{1,N} \\
\vdots & \ddots & \vdots \\
\eta_{N,1} & \cdots & \eta_{N,N}
\end{pmatrix}.
\]  

(3.23)

\( Q \) is referred to as the interaction matrix. Similar to the relationship matrix \( R \) in (3.8) and steadiness matrix \( W \) in (3.9), it’s symmetric with diagonal elements undefined.

### 3.2.4 Closed Loop Pre-Synthesis

To synthesize the desired behavior in pairwise nodal distances, when all other parameters of the N-body model have been specified or extracted from the target scenario, the problem is reduced to finding the appropriate interaction matrix \( Q \) such that the actual friendship and steadiness matrices (denoted by \( R' \) and \( W' \)) approximate the ones from the trace:

\[
(R', W') = \mathcal{F}(Q) \approx (R, W),
\]

(3.24)

where \( \mathcal{F} \) maps \( Q \) to the resulting \((R', W')\) given all the other parameters (hotspots, speeds, pause times, etc) in the N-body model.

Ideally, it is straightforward to find a numeric solution for (3.24) if there is a one-to-one relationship between individual elements of \( R', W' \) and \( Q \):

\[
(r'_{i,j}, w'_{i,j}) = \mathcal{G}(\eta_{i,j}),
\]

(3.25)

where \( \mathcal{G} \), if exists, maps the individual interaction vector \( \eta_{i,j} \) to the corresponding friendship and steadiness indices \((r'_{i,j}, w'_{i,j})\), independent of the other elements in \( Q \).

Unfortunately, (3.25) almost never holds. The distance between two nodes depends on interactions not only between themselves, but also with their friends, since increasing the attraction force between a pair of nodes would also shorten distances between their mutual friends. Thus, it is not feasible to obtain any direct solution of (3.24). A practical way to solve the problem is to use a closed loop pre-synthesis before the actual trace generation: start with a pre-specified \( Q \), synthesize the trace for some period of time \( T_c \), calculate the actual matrices \( R' \) and \( W' \) and adjust \( Q \) according to their difference to the target \( R \) and \( W \), then repeat this epoch with the new \( Q \) until the results are sufficiently close to the target. In this way, despite of the complex interactions between nodes, an appropriate interaction matrix \( Q \) may be approximated on an individual pair basis, e.g., for each pair of nodes like \( n_i \) and \( n_j \), adjust only the corresponding \( \eta_{i,j} \) accordingly during each cycle if there is a discrepancy between their actual and target pairwise nodal distance, namely, the mean \( M_{i,j} \) and NSD \( S_{i,j} \).
Thus, to effectively adjust $\eta_{i,j}$, we need to understand the way the individual parameters $\epsilon_{i,j}$, $s_{i,j}$ and $\sigma_{i,j}$ influence $M_{i,j}$ and $S_{i,j}$. Inferring from the characteristics of $\vec{F}_{i,j}$ shown in Fig. 3.5 as well as from the discussion in Section 3.2.3, we have the following conclusions (also summarized in Table 3.1):

- Adjusting $s_{i,j}$ or $\sigma_{i,j}$ has no effect if $\epsilon_{i,j} = 0$. Therefore $s_{i,j}$ and $\sigma_{i,j}$ should be adjusted only when $\epsilon_{i,j}$ is significantly large, i.e., the resulting inter-nodal force being comparable to the average gravity, thus having a substantial influence on $M_{i,j}$ and $S_{i,j}$.

- Increasing $\epsilon_{i,j}$ will push the two nodes towards the comfort distance $\sigma_{i,j}$ and restrain them from deviation. As $\epsilon_{i,j} \rightarrow \infty$, the two nodes will be “locked” with $M_{i,j} \rightarrow \sigma_{i,j}$ while $S_{i,j} \rightarrow 0$.

- $\sigma_{i,j}$ is directly proportional to the average distance $M_{i,j}$. $M_{i,j}$ is often at the close neighborhood of $\sigma_{i,j}$. However, when $\sigma_{i,j} < s_{i,j}$ and $\epsilon_{i,j}$ is not very large, decreasing $\sigma_{i,j}$ may reduce the influence of $\vec{F}_{i,j}$ and thus, lost control over $M_{i,j}$.

- Increasing $s_{i,j}$ increases the span of the inter-nodal force and therefore, releases the pair of nodes from being “locked” at the distance $\sigma_{i,j}$, thus will likely increase $S_{i,j}$.

### Table 3.1: Influence of Individual Parameters

<table>
<thead>
<tr>
<th>Adjustment</th>
<th>$M_{i,j}$</th>
<th>$S_{i,j}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\epsilon_{i,j}$ $\uparrow$</td>
<td>Closer to $\sigma_{i,j}$</td>
<td>$\downarrow$</td>
</tr>
<tr>
<td>$\epsilon_{i,j}$ $\downarrow$</td>
<td>Away from $\sigma_{i,j}$</td>
<td>$\uparrow$</td>
</tr>
<tr>
<td>$\sigma_{i,j}$ $\uparrow$</td>
<td>$\uparrow$</td>
<td>N/A</td>
</tr>
<tr>
<td>$\sigma_{i,j}$ $\downarrow$</td>
<td>$\downarrow$</td>
<td>N/A</td>
</tr>
<tr>
<td>$s_{i,j}$ $\uparrow$</td>
<td>N/A</td>
<td>$\uparrow$</td>
</tr>
<tr>
<td>$s_{i,j}$ $\downarrow$</td>
<td>N/A</td>
<td>$\downarrow$</td>
</tr>
</tbody>
</table>

Based on these conclusions, during the closed loop pre-synthesis, $\eta_{i,j}$ is then adjusted according to the following steps. For each pair of nodes $n_i$ and $n_j$, denote their target mean and NSD of distance as $M_{i,j}$, $S_{i,j}$, and the actual ones $M'_{i,j}$, $S'_{i,j}$. The details are described in pseudo code provided in Appendix D.

1. Before the pre-synthesis starts, assign the initial values for $\eta_{i,j}$: $\epsilon_{i,j} = G_0/(N - 1)$, $\sigma_{i,j} = M_{i,j}$ and $s_{i,j} = S_{i,j} \cdot M_{i,j}$, where $N$ is the total number of nodes. Since $\epsilon_{i,j}$ has to be sufficiently large before adjusting $s_{i,j}$ or $\sigma_{i,j}$, it is assigned the above value with which the maximum total attraction upon each node is $G_0$. 

35
2. During each pre-synthesis cycle, linearly adjust all three parameters of $\eta_{i,j}$ using a strategy derived from Table 3.1 and summarized in Tables 3.2 and 3.3. In both tables it is considered $M_{i,j}' \approx M_{i,j}$ when $M_{i,j}'$ is within a percentage threshold $\pm M_{TH}$ of $M_{i,j}$, and so does $S_{i,j}'$ with threshold $S_{TH}$.

Table 3.2: Adjustment Strategy (Primary)

<table>
<thead>
<tr>
<th>$S_{i,j}' &lt; S_{i,j}$</th>
<th>$M_{i,j}' &lt; M_{i,j}$</th>
<th>$M_{i,j}' \approx M_{i,j}$</th>
<th>$M_{i,j}' &gt; M_{i,j}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$s_{i,j} \uparrow \sigma_{i,j} \uparrow$</td>
<td>$s_{i,j} \uparrow$</td>
<td>$s_{i,j} \uparrow \sigma_{i,j} \downarrow$</td>
<td>$s_{i,j} \uparrow$</td>
</tr>
<tr>
<td>$S_{i,j}' \approx S_{i,j}$</td>
<td>$\sigma_{i,j} \uparrow$</td>
<td>$s_{i,j} \uparrow$</td>
<td>$\sigma_{i,j} \downarrow$</td>
</tr>
<tr>
<td>$S_{i,j}' &gt; S_{i,j}$</td>
<td>$\epsilon_{i,j} \uparrow$</td>
<td>$s_{i,j} \downarrow$</td>
<td>$\epsilon_{i,j} \downarrow$</td>
</tr>
</tbody>
</table>

3. During the linear adjustment, first use the primary methods in Table 3.2. Essentially, we tune $s_{i,j}$ and $\sigma_{i,j}$ to adjust the corresponding $M_{i,j}'$ and $S_{i,j}'$, except when $S_{i,j}' > S_{i,j}$ and $M_{i,j}'$ is far from $M_{i,j}$ at the same time, which indicates an insufficient $\epsilon_{i,j}$, we increase $\epsilon_{i,j}$ instead.

Table 3.3: Adjustment Strategy (Secondary)

<table>
<thead>
<tr>
<th>$S_{i,j}' &lt; S_{i,j}$</th>
<th>$M_{i,j}' &lt; M_{i,j}$</th>
<th>$M_{i,j}' \approx M_{i,j}$</th>
<th>$M_{i,j}' &gt; M_{i,j}$</th>
</tr>
</thead>
<tbody>
<tr>
<td>$\epsilon_{i,j} \uparrow$</td>
<td>$\epsilon_{i,j} \uparrow \downarrow$</td>
<td>$\epsilon_{i,j} \uparrow$</td>
<td>$\epsilon_{i,j} \uparrow$</td>
</tr>
<tr>
<td>$S_{i,j}' \approx S_{i,j}$</td>
<td>$\epsilon_{i,j} \uparrow$</td>
<td>$\epsilon_{i,j} \uparrow$</td>
<td>$\epsilon_{i,j} \uparrow$</td>
</tr>
<tr>
<td>$S_{i,j}' &gt; S_{i,j}$</td>
<td>$\epsilon_{i,j} \uparrow$</td>
<td>$\epsilon_{i,j} \uparrow$</td>
<td></td>
</tr>
</tbody>
</table>

4. It is possible that $s_{i,j}$ and $\sigma_{i,j}$ may diverge and be unable to effectively adjust $M_{i,j}'$ and $S_{i,j}'$. It is often caused by the $\epsilon_{i,j}$ being either too weak or too strong. Thus during the linear adjustment $s_{i,j}$ and $\sigma_{i,j}$ is confined to $[M_{i,j}S_{i,j}/2, 2M_{i,j}S_{i,j}]$ and $[0, 2M_{i,j}]$, respectively. If they exceed these limits, the corresponding $\epsilon_{i,j}$ is adjusted using the secondary methods in Table 3.3.

5. Continue the pre-synthesis cycles until either all $M_{i,j}'$ and $S_{i,j}'$ are on average within a percentage threshold to their targets, or the average percentage improvement of each iteration is less than a threshold.

---

3When $S_{i,j}' < S_{i,j}$ while $M_{i,j}'$ is far from $M_{i,j}$, the mean would require increasing $\epsilon_{i,j}$ but the NSD would prefer reducing it. Under such circumstances the mean has higher priority than the NSD so $\epsilon_{i,j}$ is increased. Therefore the simulation results in Section 3.3.2 show better convergence and linearity in mean than in NSD.
3.3 Simulation and Validation

We validate our model by comparing its performance with the target traces. The N-body mobility model is currently implemented in Matlab.

3.3.1 Influence of Individual Parameters

![Figure 3.6: Influence of individual parameters on the mean $M_{i,j}$ (a)(b)(c) and NSD $S_{i,j}$ (d)(e)(f) of the pairwise distances. Parameter: (a)(d) $\epsilon = 0.5$, (b)(e) $\epsilon = 2$, (c)(f) $\epsilon = 10$. The $\sigma$ axis is transposed in (d)(e)(f) for improved visibility.](image)

Before synthesizing the traces for target scenarios, we first validate the influence of the individual parameters $\epsilon_{i,j}$, $s_{i,j}$ and $\sigma_{i,j}$ on the actual mean and normalized standard deviation (NSD) of the pairwise distances as described in Section 3.2.3 and 3.2.4. We run an N-body simulation with two nodes on a 200m $\times$ 200m plane with no hotspots and no pause. The unit force (acceleration) is set to $G_0 = 1m/sec^2$ with the reference distance $d_0 = 140m$, which is approximately the average distance between two nodes in RWP calculated using (3.4) with the density function $\lambda$ set to the one of RWP from [70]. Each node has the same maximum speed 5m/s. The simulator proceeds at time step $\Delta t = 1s$, and node locations are sampled every 8
seconds. Each simulation lasts 4 hours.

For each run we specify the $\epsilon$, $s$ and $\sigma$ of the inter-nodal force between the two nodes. The resulting mean and NSD of their distances are shown against $\sigma$ and $s$ in Fig. 3.6 with $\epsilon = 0.5, 2, 10$m/s$^2$, respectively. When $\epsilon$ is low ($\epsilon = 0.5$) as shown in Fig. 3.6(a) and 3.6(d), adjusting $s$ and $\sigma$ does not make a large difference. When $\epsilon$ is significantly high ($\epsilon = 10$), from Fig. 3.6(c) it is shown that the mean is mainly determined by $\sigma$ while almost not affected by $s$, and from Fig. 3.6(f) the NSD is shown to be proportional to $s$ with given $\sigma$. Increasing $\sigma$ also decreases NSD since it increases the mean.

### 3.3.2 Synthesis from Sample Traces

<table>
<thead>
<tr>
<th>Parameters</th>
<th>RPGM</th>
<th>SFCAB</th>
<th>STBUS</th>
</tr>
</thead>
<tbody>
<tr>
<td>Node Num</td>
<td>48</td>
<td>50</td>
<td>48</td>
</tr>
<tr>
<td>Area (km × km)</td>
<td>20 × 20</td>
<td>61 × 47</td>
<td>24 × 30</td>
</tr>
<tr>
<td>Ref Distance $d_0$ (m)</td>
<td>7287</td>
<td>4785</td>
<td>9184</td>
</tr>
<tr>
<td>Unit Force $G_0$ (m/sec$^2$)</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Avg Speed (m/sec)</td>
<td>6.5</td>
<td>6.3</td>
<td>4</td>
</tr>
<tr>
<td>Avg Pause (sec)</td>
<td>0</td>
<td>56</td>
<td>297</td>
</tr>
<tr>
<td>Step Time $T_{step}$ (sec)</td>
<td>1</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Total Simulation Time (hour)</td>
<td>16</td>
<td></td>
<td></td>
</tr>
<tr>
<td>Contact Range (m)</td>
<td>2500</td>
<td>2000</td>
<td>3000</td>
</tr>
</tbody>
</table>

To validate the N-body model we first use the model to synthesize traces showing social structures similar to those extracted from the traces in Section 3.1, namely, the RPGM, SFCAB and STBUS traces. The parameters of the N-body models are set according to the target scenarios as summarized in Table 3.4. Assuming all nodes move with the same maximum speed and follow an exponential pause time distribution, the average speeds, pause times and total simulation times are directly obtained from the traces. Reference distances $d_0$ used in (3.17) are set to the mean distances of independent random placement $M_{i,j}^{ind}$ which are calculated using (3.4) in Section 3.1. For better synthesis accuracy, hotspots, obtained through empirical node density from the target traces, are also implemented. The whole simulation area is divided into a total of $K$ square grids (1km × 1km for SFCAB and STBUS, 200m × 200m for RPGM), and a probability of selection $\mu_k$ ($k \in [1 \ldots K], \sum \mu_k = 1$) is calculated for each grid from the sample traces as the average percentage of node population inside that grid. When each node is selecting its next destination (Section 3.2.1, step 2), it first randomly selects a grid following...
Figure 3.7: Trends of convergence during the closed loop pre-synthesis (a)(b)(c) and convergence results at the end of pre-synthesis as target versus the actual mean distance $M_{i,j}$ (d)(e)(f) and NSD $S_{i,j}$ (g)(h)(i). Mobility Models: (a)(d)(g) RPGM, (b)(e)(h) SFCAB, (c)(f)(i) STBUS. Solid lines in (d)-(i) have the slope of the ratio between $M_{\text{ind}}'$ and $M_{\text{ind}}$ or between $S_{\text{ind}}'$ and $S_{\text{ind}}$. Dashed lines have the slope of 1.

Before synthesizing traces, we run the closed loop pre-synthesis as described in Section 3.2.4 to obtain the appropriate interaction matrix $Q$.\footnote{When the social structure in the target scenario is simple, e.g., similar to that of RPGM, the interaction matrix $Q$ may be manually set without the pre-synthesis described in Section 3.2.4.} For the closed loop adjustments, we set the percentage thresholds $M_{\text{TH}}, S_{\text{TH}}$ and linear slope $\delta_e$ (explained in Appendix D) to be 5%, 10%
and $1 \times 10^{-6} \text{m}^{-1}$, respectively. The maximum speeds are also adjusted during the closed loop cycles to approach the desired average speeds. The cycle period $T_c$ is set to 160 hours, which yields statistically significant results.

We show the trends of convergence during the closed loop as well as the convergence results by the end of the pre-synthesis in Fig. 3.7. The trends of convergence are shown in Fig. 3.7(a)-(c) as the average absolute difference between the actual and target mean (solid lines) and standard deviation (dashed lines) plotted against number of iterations. The convergence results are presented in Fig. 3.7(d)-(i) as the target mean $M_{i,j}$ and NSD $S_{i,j}$ versus the actually generated results $M'_{i,j}$ and $S'_{i,j}$, together with the solid lines having the slope of the ratio between the actual and target mean or NSD of independent random placements $M_{\text{ind}}'/M_{\text{ind}}$ or $S_{\text{ind}}'/S_{\text{ind}}$. Ideally, in Fig. 3.7(a)-(c) all lines should converge to 0, and in Fig. 3.7(d)-(i) the solid lines should have a slope close to 1, which are the dashed lines, and the dots be closely distributed around the solid lines. Although the discrepancies from ideal cases suggests that there are limitations in our pre-synthesis and the target social matrices are approximated on a best effort basis, the linearity shown in the final convergence results in Fig. 3.7(d)-(i) indicates that the heterogeneity of inter-nodal distance in the target traces are sufficiently well captured and reproduced.

To further validate the N-body model, we ran MANET simulations on the ns-2 simulator [77] using two representative routing protocols: the reactive AODV [78] and the proactive DSDV [79] on both the target and the synthesized traces. The contact ranges listed in Table 3.4 are heuristically determined according to the dimensions of the simulation area and the average node density. For comparison, we also show the results for Improved Random Waypoint Models (IRWP), which are RWP models that incorporate hotspots in waypoint selections in the same way as we implemented in N-body. The IRWP models use the same parameters (node number, simulation area, speed, pause, total time) and the same hotspot probabilities $\mu_{i,j}$ as the corresponding N-body models. Therefore, the IRWP models keep all the features of the corresponding N-body models except the attraction and inter-nodal forces, which are replaced with straight-line movements at constant speeds.

Simulations are run with both light and heavy traffic. During light traffic simulation, data packets are only generated and sent between one pair of nodes. Packet size is set to 500 kB and 1 packet is generated each second. Simulations are run for each pair of nodes and the average packet delivery ratio (PDR) and end-to-end delay are collected. The CCDF of the pairwise PDR and delay for all 1128 pairs (1225 pairs for STBUS) of nodes are shown in Fig. 3.8. For heavy traffic simulation, all pairs of nodes simultaneously send out packets at the same time and the packet rate is reduced to 1 packet per 3 seconds. Since there are more than 1000 flows at the same time, the traffic load is much heavier despite the reduced packet rate; as shown in Fig. 3.9 the average packet delivery ratios are lower on all scenarios.
Figure 3.8: Light traffic simulation results for AODV (a)-(f) and DSDV (g)-(l): CCDF of pairwise packet delivery ratio (a)-(c),(g)-(i) and CCDF of pairwise delay (d)-(f),(j)-(l). Mobility models: (a)(d)(g)(j) RPGM, (b)(e)(h)(k) SFCAB, (c)(f)(i)(l) STBUS.
Figure 3.9: Heavy traffic simulation results for AODV (a)-(f) and DSDV (g)-(l): CCDF of pairwise packet delivery ratio (a)-(c),(g)-(i) and CCDF of pairwise delay (d)-(f),(j)-(l). Mobility models: (a)(d)(g)(j) RPGM, (b)(e)(h)(k) SFCAB, (c)(f)(i)(l) STBUS.
From Figs. 3.8 and 3.9, the curves of the N-body models are always closer to those of real traces than the IRWP models, as well as showing similar trends of heterogeneity, which is particularly clear in the STBUS and RPGM simulations. On the other hand, although the IRWP models exhibit different results over different scenarios, all those results approximately follow a normal distribution in their pairwise average values, which indicates a homogeneous inter-nodal relationship. Therefore, the N-body model captures the existing pairwise heterogeneity in sample traces, which mainly comes from the group-forming tendencies among mobile nodes, while the improved RWP models that incorporates empirical hotspots, speeds, pause times, etc. like in [16,18] fail to capture this and result in homogeneous behavior.
Chapter 4

Extending Complex Group Structures

In this chapter we present our method that extends the group structure observed in small populations to larger populations. The detailed implementation of the model is presented in Section 4.1. The model is then validated in Section 4.2 both as a stand alone generator as well as combined with N-body mobility model.

4.1 Network Generation Algorithm

In this section we introduce our method of generating those networks to be used in N-body model as described in Section 3.2. In particular, we synthesize a weighted network consisting of $N$ nodes, called the output network, that shows statistical features similar to the given weighted network of $M$ nodes, called the input network, where usually $M \ll N$. Inspired by the work in [68], which generates unweighted networks with given degree distribution and degree dependent clustering using a configuration model, in our method we extract certain parameters such as weight distribution and cluster coefficients from the input network, then generate a weighted network that shows similar features using a modified configuration model.

The content of this section is organized as follows: In Section 4.1.1 we briefly describe the structural features of the relationship networks, as well as the testing scenarios we will use in the generation process. We then introduce the basic configuration model and our enhancement based on role-model mechanism in Section 4.1.2. In Section 4.1.3 and 4.1.4 the two major statistics, weight distribution and cluster coefficients, are discussed in detail, respectively. In Section 4.1.5 we present the detailed operation of our network generator.
4.1.1 Relationship Networks and Testing Scenarios

In the N-body model, the relationship network $R$ of all the nodes defined in (3.8) is a fully connected weighted network. The edge connecting two arbitrary nodes $n_i$ and $n_j$, is denoted the relationship index $r_{i,j}$ and defined in (3.6):

$$r_{i,j} = 1 - \frac{M_{i,j}}{M_{\text{ind}}}, \quad r_{i,j} \in (-\infty, 1],$$  \hspace{1cm} (4.1)

where $M_{i,j}$ is the mean distance calculated using (3.2), and $M_{\text{ind}}$ is the average distance assuming independent random placement calculated using (3.4). For convenience in calculation, in this chapter we normalize this weight to $[0,1]$:

$$r_{i,j} = \frac{\max(M) - M_{i,j}}{\max(M)}.$$  \hspace{1cm} (4.2)

As shown in Fig. 4.1, nodes from the same group will have heavy edges between each other, while nodes from different groups will be connected with light edges. The weights of the intra-group edges corresponds to the closeness of the group.

![Figure 4.1: Groups in the Relationship Network](image)

To quantitatively describe the structural features of such fully connected weighted networks, we extract two sets of parameters on a per-node basis: the weight distribution and weight dependent cluster coefficient, which will be discussed in detail in the following Sections 4.1.3 and 4.1.4.

To better illustrate the network generating process, similar to Section 3.1.2, we obtained two testing input networks from different scenarios. The first one, denoted RPGM-O scenario, is from the classical RPGM model [38]. In that model we have 48 nodes divided into 4 groups with 12 nodes each, moving on a $20 \times 20$km$^2$ plane for 16 hours. Two of the groups have a radius of 500m (“tight” groups), while the others have a radius of 2km (“loose” groups). Each group moves at 5m/s, and each node randomly moves inside its group at 5m/s. No pause was
implemented.

The second one, denoted BUS-O scenario, is a real bus GPS trace in Seattle from the Ad-Hoc City project [72], which contains 239 routes (more than 1200 nodes) recorded for about 20 days. Like in Section 3.1.2, since there are frequent discontinuities in the trace, we extracted continuous records for each bus (referred to as flights), out of the trace. The flights are time-shifted so they all start at the same time, thus, buses running the same routes will seem to move in groups. In addition, since every bus sticks to only the area its route covers, to isolate the groupness from the geographical preferences of each route, we also spatially shifted and resized the routes when necessary such that all routes now cover approximately a same geographical area. We selected 40 flights, 4 from each of 10 different routes, to form our testing input scenario, each of the flight lasts for at least 5.3 hours.

4.1.2 Configuration Model and Role-Model Enhancement

The original configuration model [64] is an effective way to generate unweighted networks with an arbitrary degree distribution. Its basic operation consists of first assigning a certain number of “half edges”, called stubs, to each node according to the desired degree distribution, then randomly connecting the stubs from unconnected nodes to form edges until all stubs are connected. The advantage of this model is that it can be easily customized to impose fine-grained control onto individual nodes, e.g., degree distribution, weight distribution and cluster coefficient as demonstrated in [67, 68]. In this paper we use this model as a starting point to generate the output networks, but rather than assigning unweighted stubs, we assign weighted stubs with desired weights to each node, and connect them accordingly.

To better preserve the structural features of the input networks, we propose a role-model approach to enhance the configuration model. As illustrated in in Fig. 4.1, groups may have different tightness and sizes, thus nodes from different groups would have different weight distributions and cluster coefficients. Such heterogeneity can not be sufficiently described using only aggregated statistics. On the other hand, since the input network can be regarded as a small sample of the actual target population, it is reasonable to expect that there are behavioral similarities between nodes in the input network and nodes in the output network. Therefore, to accurately reflect such heterogeneity in the input network, each node in the output network follows a node in the input network as its role-model. The basic idea is that each node in the output network will inherit the individual statistics such as stub weight distribution and weight dependent cluster coefficients from its role model in the input network (the exact mechanism will be discussed in detail in the following sections). If the role-models of two nodes are connected via a heavy link in the input network, then these corresponding nodes in the output network are also likely to be connected via a similarly heavy link in the output network, and vice versa.
In other words, if the role-models of a node belongs to a certain group in the input network, then that node is likely to belong to a similar group in the output network.

### 4.1.3 Weight Distribution

As an analogy to the degree distribution in unweighted networks, the weight distribution in our fully connected weighted network is an important parameter in describing its structural features. For each particular node, assuming the node belongs to a certain group, its weight distribution largely reflects what kind of group the node belongs to: whether large or small, tight or loose, etc.

However, when applying the weight distribution extracted from a small population to a larger population, there are different scenarios that should be considered, since groups may grow in different ways as the population grows. As shown in Fig. 4.2, the original network contains 6 nodes forming 3 groups. When the node population doubles, there may still be 3 groups with each group containing twice as many nodes, as depicted in scenario 1; at the other extreme, each group may continue to have 2 nodes, but the number of groups doubles as shown in scenario 3; between the two extremes: there can be more groups, and each group can have more nodes, as shown in scenario 2.

![Figure 4.2: Different scenarios of the group structure when increasing population.](image)

Therefore, the weight distributions of the nodes in the output network may differ from those in the input network due to the different ways the groups grow. Assuming the networks depicted in Fig. 4.2 have heavy edges between nodes from the same group, and light edges (not shown) between those from different groups, the percentage of heavy edges is 20% in the input network, while it is 27% in scenario 1 (“larger groups”), 18% in scenario 2 (intermediate), and only 9.1% in scenario 3 (“more groups”). Generally, with more groups and fewer nodes in each group, the percentage of heavy edges decreases, resulting in a weight distribution that becomes
distorted in favor of the light edges.

Thus, before the extracted weight distribution of each node is applied to the output network, it is adjusted according to the different group growth behaviors mentioned above. In order to quantitatively describe the different scenarios, we define the distortion parameter $r \in [0, 1]$, where $r = 0$ indicates larger groups, $r = 1$ indicates more groups, and $0 < r < 1$ indicates intermediate scenarios. A simple adjustment method is to multiply the weight histogram of the incremental edges by a distortion function whose shape is controlled by $r$. Assuming there are $M$ nodes in the input network and $N$ nodes in the output network, the weight distribution is adjusted following the steps listed below:

1. For each node $m_i$ in the input network, extract the weight histogram $F_I(i, w)$ of its connected edges such that
   \[ \sum_{w=w_{\text{min}}}^{w_{\text{max}}} F_I(i, w) = M - 1. \]  
   (4.3)

2. As the population increases, there are $N - M$ more edges connected to each node. Their weight histogram, $\Delta F(i, w)$, follows the input weight distribution multiplied by a distortion function $f(w, r)$:
   \[ \Delta F(i, w) = \frac{F_I(i, w) \cdot f(w, r)}{\sum_{w=w_{\text{min}}}^{w_{\text{max}}} F_I(i, w) \cdot f(w, r)} (N - M). \]  
   (4.4)

The shape of the distortion function $f(w, r)$ is controlled by parameter $r$. When $r = 0$ it should be a flat function $f(w, 0) = 1$, which implies that edges of all weights grow proportionally when there are larger groups\(^1\). As $r$ increases the distortion function $f(w, r)$ should bend to favor inter-group (light) edges in order to decrease the number of intra-group (heavy) edges and reduce group sizes. As $r$ approaches 1, the distortion function should become an impulse centered at the average inter-group edge weight, therefore favoring inter-group (light) edges to an extreme extent. Based on these requirements we choose the following asymmetric quadratic function with a peak at $w_0$ as illustrated in

---

\(^1\)When groups grow, it is the population of each group that grows proportionally rather than the number of edges, and the number of heavy intra-group edges connected to each node is always equal to the group population minus one. For example, assuming there are 4 nodes in each group in the input network, as the population triples there should be 12 nodes nodes in each group. The number of heavy edges connected to each node increases from $3 = 4 - 1$ to $11 = 12 - 1$. Thus, to correctly estimate the number of additional heavy edges, when given that of the input network which is 3, we need to add 1 to that 3 to recover the group population, then increase it proportionally. In our example it is $(3 + 1) \times 2 = 8$. This also explains the fact that in the networks depicted in Fig. 4.2, the “larger group” scenario has higher percentage of heavy edges (27%) compared to the original network (20%). Therefore, in (4.4) the input histogram $F_I(i, w)$ needs to be slightly modified before calculating $\Delta F(i, w)$: $F_I(w_h) \leftarrow F_I(w_h) + 1$, where $w_h$ is the highest weight where $F_I(i, w_h) > 0$. The input histogram $F_I(i, w)$ in (4.8) remains the same.
\[ f(w, r) = \max \left[ 1 - \frac{r}{1-r} \left( \frac{w-w_0}{w_m(w)} \right)^2 , 0 \right], \quad (4.5) \]

where \( w_m(w) \) is the normalization factor:

\[ w_m(w) = \begin{cases} 1 - w_0 & w > w_0 \\ w_0 & w \leq w_0 \end{cases}, \quad (4.6) \]

such that \( f(w, r) \) will decrease proportionally on both sides of \( w_0 \). Theoretically the peak \( w_0 \) should be the average weight of the inter-group edges, since as \( r \) increases the inter-group edges should be favored over intra-group (heavy) edges. In practice we use the normalized average distance assuming independent random placement as discussed in Section 3.1.1, which works well in our testing scenarios:

\[ w_0 = \frac{\max(M) - M_{\text{ind}}}{\max(M)}, \quad (4.7) \]

where \( M_{\text{ind}} \) is defined in (3.4), and can be estimated based on the population distribution of the input trace.

---

**Figure 4.3:** Illustration of the distortion function. \((w_0 = 0.3)\)

3. Finally the weight histogram in the output network is:

\[ F_O(i, w) = F_I(i, w) + \Delta F(i, w). \quad (4.8) \]

The processed weight histogram \( F_O(i, w) \) is applied to every node in the output network.
who follows node $m_i$ as its role model.

Table 4.1: RPGM Scenarios Group Structure

<table>
<thead>
<tr>
<th>Trace</th>
<th>RPGM-L</th>
<th>RPGM-I</th>
<th>RPGM-M</th>
</tr>
</thead>
<tbody>
<tr>
<td>Number of Tight Groups</td>
<td>2</td>
<td>4</td>
<td>10</td>
</tr>
<tr>
<td>Nodes in Each Tight Group</td>
<td>60</td>
<td>30</td>
<td>12</td>
</tr>
<tr>
<td>Number of Loose Groups</td>
<td>2</td>
<td>3</td>
<td>10</td>
</tr>
<tr>
<td>Nodes in Each Loose Group</td>
<td>60</td>
<td>40</td>
<td>12</td>
</tr>
</tbody>
</table>

We applied the above process to both RPGM-O and BUS-O testing scenarios. In the RPGM scenario, the population is increased from 48 to 240. To validate the different group growth behavior, three additional RPGM traces with 240 nodes are generated: the RPGM-L trace corresponding to “larger groups” scenarios, the RPGM-M trace corresponding to the “more groups” scenario, and the RPGM-I trace corresponding to the “intermediate” scenario. The group structure of these traces are listed in table 4.1. In correspondence to these three scenarios, we processed the weight distribution using heuristically determined $r$ values: \( r = 0 \) for “larger groups”, \( r = 0.65 \) for “more groups”, and \( r = 0.475 \) for the “intermediate” scenario\(^2\). We plot the overall weight distribution of the RPGM traces and the output networks in Fig. 4.4. From the CDF plot it is clear that there are differences between the different RPGM scenarios as expected, and the corresponding distorted distributions match well with the extended scenarios.

Similarly, in the BUS scenario, the population is increased from 40 to 140 and three additional BUS traces with 140 nodes are collected: the BUS-L trace containing 10 routes with 16 nodes each, the BUS-I trace containing 20 routes with 8 nodes each, and the BUS-M trace containing 40 routes with 4 nodes each. The weight distribution of output networks are adjusted using the following $r$ values: \( r = 0 \) for “larger groups”, \( r = 0.57 \) for “more groups”, and \( r = 0.46 \) for the “intermediate” scenario. We plot the overall weight distribution of the BUS traces and the output networks in Fig. 4.5. Although the differences between different scenarios are not as apparent as in the RPGM traces, the subplot shows the different scenarios are clearly distinguishable at the heavyweight end, and the distorted distributions closely matched those of the extended scenarios.

\(^2\)Though in theory $r = 0.99 \cdots$ should be used for “more groups” scenarios as it is the extreme case of favoring inter-group edges, in practice it is often not a good choice. When $r$ gets close to 1, the actual weight distribution would have a steep step over $w_0$ (derived using (4.7)), which is not the case in real traces as illustrated in Figs. 4.4 and 4.5, as the inter-group edges might actually span a wide spectrum of weights. Thus in our test scenarios we heuristically used $r = 0.65$ and 0.57 for RPGM and BUS scenarios with “more groups”, respectively.
4.1.4 Weight Dependent Cluster Coefficient

Although the edge weight distribution of each node does reflect important information about group movements such as size and geographical radius of the groups, however, weight distribution alone is not a clear indicator that the nodes are actually moving in groups. If a group exists in a relationship network, group members should be connected to each other via heavyweight edges. In other words, if two nodes both have a heavyweight connection to a third node, it is very likely that the connection between the two nodes is also heavyweight. The cluster coefficient has been a successful measure in quantifying such characteristics.

The cluster coefficient (CC) was originally proposed to measure the tendency of nodes to form clusters in unweighted networks. Its global version (denoted by $C$) was first proposed in [80] to measures the overall level of clustering in a network. For any three nodes in an undirected and unweighted network, they form a connected triplet if there are at least two edges connecting them, and a triangle if there are three edges. The global CC is defined as the
ratio between the number of triangles and connected triplets:

\[ C = \frac{\text{Number of Triangles}}{\text{Number of Connected Triplets of Nodes}}. \] (4.9)

Basically, the global CC indicates the probability of two nodes being connected given that they have a common neighbor. With \( C = 1 \), there is always an edge between two nodes when they have at least one common neighbor.

The local CC was introduced in [34] to measure such tendency in the neighborhood of a particular node. For a node \( n_i \) its local CC, denoted by \( C_i \), is defined as:

\[ C_i = \frac{2 \sum_{n_j, n_k \in N_i} e_{j,k}}{K_i(K_i - 1)}, \] (4.10)

where \( K_i \) is the degree of \( n_i \), \( N_i \) is the set of all nodes connected to \( n_i \), and \( e_{j,k} = 1 \) if there is an edge between \( n_j \) and \( n_k \), 0 if otherwise. Similar to the global CC, the local CC indicates the probability of any two neighbors of node \( n_i \) to be connected to each other. If \( C_i = 1 \), all neighbors of \( n_i \) are connected to each other, thus forming a tightly knitted cluster; while \( C_i = 0 \) means there is no edge between any of \( n_i \)'s neighbors.

Several methods have been developed to extend the cluster coefficient to weighted networks. The most popular ones are surveyed in [81], and a recent one is proposed in [82]. Although these methods differ in their actual calculation, all of them are based on a similar idea to their unweighted counterparts: while in unweighted networks all the triangles are counted, in weighted networks the “strengths” of all the triangles are summed up. Usually triangles that consist of only heavyweight edges are assigned very large “strengths” compared to other ones, therefore summing up these “strengths” reflects the level of clustering in weighted networks. As shown in Fig. 4.6, a triangle consisting of three thick links is considered a strong triangle. In subplot (a) there is one strong triangle, while in (b) there is none. If the strengths of strong triangles and other ones are properly defined, there will be a significant difference between the cluster coefficients in the two scenarios.

However, in our fully connected relationship networks, such weighted cluster coefficients do not scale well as the population increases. The reason is that no matter how small the strengths of regular triangles are compared to those of strong triangles, they should not be zero as long as all edges have positive weights. As shown in Figs. 4.6(c) and 4.6(d), when the population increases, the number of regular triangles increases quadratically, while the number of strong triangles may increase differently depending on the group growth behavior discussed in Section 4.1.3. In the “more groups” scenario, the number of strong triangles increases linearly. Thus, when the population is large enough, eventually the total strength of regular triangles will overwhelm that of the strong triangles, resulting in little distinction in cluster coefficients.
between highly clustered networks and random networks\(^3\).

Therefore, to provide a better quantitative description of the level of clustering in our relationship networks, we propose to use *weight dependent* cluster coefficients. The idea is best explained by analogy to tuning the brightness of a monochrome monitor: the monitor starts completely black, as the brightness increases, first the brightest spots show up, forming the skeleton of the image, then gradually the rest of all the other parts start to appear one by one. As illustrated in Fig. 4.7, the original network shown in subplot (a) consists of heavy (thick black lines), medium (thick gray lines) and light edges (thin dashed lines). In other subplots we form three different unweighted networks by including only edges heavier than a certain threshold: only heavy edges in (b), both heavy and medium ones in (c) and all of them in (d). Thus at each of these thresholds the corresponding global and local cluster coefficients as defined in (4.9) and (4.10) can be calculated, reflecting the clustering levels of the weighted network on different weight thresholds.

By tying the cluster coefficients to different levels of weight, the issue of regular triangles overwhelming strong ones disappears since regular triangles do not exist when the weight thresholds are high. On the other hand, calculating CCs at multiple levels of weight makes it possible

\(^3\)Negative triangle strengths could be used to counter this problem if the average strength of weak triangles is 0. However, it is very difficult to properly define the positive / negative triangle strengths even when negative edge weights are allowed. Furthermore, this approach does not work well for networks that do not have a significant distinction between heavy and light edges.
to describe the complex group structures, e.g., tightness of groups, in weighted networks. By properly selecting the weight thresholds, the structural information can be preserved to a great extent, such that it is sufficient to generate relationship networks to be used in the N-body model.

To select the optimal weight thresholds, we first transform the weighted network to un-weighted ones using a dense series of weight thresholds, and examine their global cluster coefficients as shown in Figs. 4.8 and 4.9. Among the series of thresholds, we choose those whose corresponding global CCs are local maxima. The reason for this choice is that when the global CC reaches a local maxima, it is very likely that at that particular threshold the group structure of the heavy edges is best revealed, while the global CC may be reduced with lower or higher
thresholds, as some light edges are added or some heavy edges are removed\textsuperscript{4}.

We plot the global CCs of both the RPGM-O and BUS-O input networks in Figs. 4.8 and 4.9, respectively, and mark the selected thresholds. From Fig. 4.8 it is clear that thresholds 0.79 and 0.95 corresponds to the tight and loose groups. In Fig. 4.9 the thresholds 0.78, 0.91, 0.96 and 0.97 also reflects different groups of buses.

![Figure 4.8](image1.png)

**Figure 4.8:** Selected thresholds based on global cluster coefficients of the RPGM-O input network.

![Figure 4.9](image2.png)

**Figure 4.9:** Selected thresholds based on global cluster coefficients of the BUS-O input network.

### 4.1.5 Network Generator Algorithm

Based on the basic configuration model introduced in Section 4.1.2, we incorporate the improvements and parameters discussed in previous sections, and proposed our improved network

\textsuperscript{4}Sometimes the dense series of thresholds could cause excessive local maximas in global CCs (as shown in Fig. 4.9) which does not truly reflect the group structure. To smooth out the “jitters” only the maximas who are also the maximums in their left neighborhoods are selected. In our test cases we used a neighborhood size of 0.05.
generator (an optional extension to this generator that favors cluster coefficients over degrees is introduced in Appendix F.). The detailed operation follows the steps listed below:

1. Normalize the input network using (4.2), such that the weights \( r_{i,j} \) follow \( r_{i,j} \in [0,1] \) where \( r_{i,j} = 1 \) indicates strongest grouping tendency.

2. For each threshold \( b_{th} = 1, 0.99, 0.98, \ldots, 0.01, 0.00 \), convert the weighted network into an unweighted one, whose edges, denoted \( r'_{i,j} \), follow \( r'_{i,j} = 1 \iff r_{i,j} \geq b_{th} \) and \( r'_{i,j} = 0 \iff r_{i,j} < b_{th} \). Then calculate its global cluster coefficient, denoted \( C_g(b_{th}) \), using (4.9). Find the optimal thresholds, from the highest weight to the lowest weight denoted \( b_o(k), k = 1, 2, \ldots, K \), where each of their corresponding global CCs \( C_g(b_o(k)) \) reaches a local maxima. The optimal thresholds divide the edge weight space into weight levels.

3. For each optimal threshold \( b_o(k) \), calculate the local cluster coefficient, denoted \( C_l(i, k) \), for each node \( m_i \) in the corresponding unweighted input network using (4.10).

4. For each node \( m_i \) in the input network, derive its stub weight histogram for the output network, denoted \( F_O(i, w) \), using the method described in Section 4.1.3.

5. For each node \( n_i \) in the output network, randomly choose a node in the input network as its role model, denoted \( m_{\theta(i)}, \theta(i) \in \{1, 2, \ldots, M\} \). Assign \( N-1 \) stubs to \( n_i \). Assign a desired weight to each stub according the role model’s output weight histogram \( F_O(\theta(i), w) \).

6. Choose the optimal threshold of the highest weight as the current threshold \( b_o(1) = b_o(k) \).

7. Mark all the stubs whose weights are no lower than \( b_o(k) \) as eligible. For each node \( n_i \), denote the number of its eligible stubs as \( G(i, k) \) and the number of currently connected edges as \( L(i) \), use its role model’s corresponding local cluster coefficient \( C_l(\theta(i), k) \) to calculate its target number of local triangles:

\[
T_{tgt}(i, k) = \frac{C_l(\theta(i), k)}{2} \frac{[G(i, k) + L(i)] [G(i, k) + L(i) - 1]}{[G(i, k) + L(i)] [G(i, k) + L(i) - 1]}.
\]  

(4.11)

Since the algorithm starts with the highest threshold and goes all the way down to 0, once two nodes are connected on a higher weight level, they will always remain connected throughout the lower weight levels, so do the triangles formed on higher levels. Therefore, we keep the actual number of local triangles for each node in \( T_{act}(i) \) throughout all levels. At the beginning all nodes have \( T_{act}(i) = 0 \), while by the end \( T_{act}(i) = (N-1)(N-2)/2 \). We also keep the number of common neighbors between each pair of unconnected nodes \( n_i \) and \( n_j \) in \( B(i, j) \). When \( n_i \) and \( n_j \) are connected, they will form \( B(i, j) \) triangles in each other’s neighborhood.
8. In the output network, select one node \( n_i \) with the maximum number of eligible stubs, denoted as the *starting node*.

9. Find all the other nodes that are not yet connected to the starting node \( n_i \) but have eligible stubs. Store these candidates in \( Q \). If \( Q \) is non-empty, choose the best candidate using steps listed below; Otherwise, go to step 11.

   (a) From each node \( n_j \) in \( Q \), check if the number of local triangles of both \( n_i \) and \( n_j \) will exceed the desired amount of triangles if they are connected:

   \[
   B(i, j) + T_{\text{act}}(i) \leq T_{\text{tgt}}(i, k), \quad (4.12)
   \]

   \[
   B(i, j) + T_{\text{act}}(j) \leq T_{\text{tgt}}(j, k). \quad (4.13)
   \]

   Select only the nodes that satisfy both conditions. If no node satisfies both conditions, select those who will exceed the least number of triangles. Remove the rest of the nodes from \( Q \). Go to the next step if there are more than one node in \( Q \).

   (b) To generate as many triangles as possible while not exceeding the target number, from the remaining nodes in \( Q \) keep those that will generate the most number of triangles if connected to \( n_i \), which is equivalent to those with highest \( B(i, j) \). Remove the other nodes from \( Q \). Go to the next step if there are more than one node in \( Q \).

   (c) For each remaining node \( n_j \), find its role model \( m_{\theta(j)} \) and that of the starting node \( m_{\theta(i)} \). Check the edge weight between the two role models in the input network. If that weight is within the current *weight level* (between the current threshold \( b_0(k) \) and the previous one \( b_0(k-1) \)), keep \( n_j \) in \( Q \). Keep all nodes if no one satisfies this condition. If there are more than one node in \( Q \), go to the next step.

   (d) Among all the remaining nodes in \( Q \), choose from all their eligible stubs the one whose desired weight is closest to that of the heaviest stub of \( n_i \), and keep only the corresponding node.

In actual implementation, the above selection process is simplified into calculating a likelihood value for each node and choosing the highest one. All factors from the above steps are added into the likelihood with different weights to insure proper precedence of these conditions.

10. Once the best candidate \( n_j \) is found, connect the highest weighted stub on \( n_i \) with a stub on \( n_j \) whose weight is the closest to that of the \( n_i \)'s. Form an edge between \( n_i \) and \( n_j \) with a weight of the average of the two stubs. Update the number of generated triangles in \( T_{\text{act}} \) and the number of common neighbors in \( B \). If there are still eligible stubs on node \( n_i \), go back to step 9; otherwise, go to next step.
11. If there are still eligible stubs on node $n_i$, no other nodes will have stubs to connect to them on the current threshold. Therefore the remaining stubs will be moved to the next lower weight level in which $n_i$ has stubs, and their desired weights will be changed according to the weight distribution of that lower level. Choose the next starting node and go back to step 9: first pick the one with the most eligible stubs among the nodes that are connected to $n_i$; if all neighbors of $n_i$ have no eligible stubs then pick from the rest of the nodes. If there are no more eligible stubs on the current threshold $b_0(k)$, move to the next lower threshold $b_0(k + 1)$ and go to step 7.

4.2 Simulation and Validation

In this section we perform simulations to validate the network generator we proposed in Section 4.1 (with the optional extension described in Appendix F). In the following Section 4.2.1 we validate our generator by comparing the cluster coefficients and modularity in original networks and synthesized ones. In Section 4.2.2 our network generator is combined with N-body model to synthesize the mobility traces. Both protocol independent and dependent performance metrics are also validated against original traces.

4.2.1 Cluster Coefficients and Modularity

To validate the group structures in the synthesized network, similar to the threshold selection process described in Section 4.1.4, for both RPGM and BUS traces, we plot the unweighted global CC using a dense series of weight thresholds and compare them against the original networks in Figs. 4.10 and 4.11, respectively. From the figures it is clear that although there are some discrepancies at the lower weight end with RPGM traces, and in the more groups scenario with BUS traces, the global cluster coefficients in the output networks match well with the real networks, especially at the higher weight end that represents positive group forming tendencies.

For both weighted and unweighted networks with groups, a measure called modularity was proposed in [83] to indicate the quality of a particular group division: a division of a network is considered “high quality” when intra-group edges have heavy weights while inter-group edges have light weights, and vice versa. Thus by measuring the difference of edge weights between inter-group edges and intra-group edges, one can quantitatively tell the quality of a particular division. For a network with $N$ nodes and pairwise edge weights $r_{i,j}$ ($r_{i,j} = \{0, 1\}$ for unweighted networks), the modularity $Q$ is defined as:

$$Q = \frac{1}{4m} \sum_{i,j} \left( r_{i,j} - \frac{k_i k_j}{2m} \right) \delta_{i,j},$$  \hspace{1cm} (4.14)
where,

\[ k_i = \sum_j r_{i,j} \]  
(4.15)

\[ m = \frac{1}{2} \sum_i k_i \]  
(4.16)

and \( \delta_{i,j} = 1 \) when \( n_i \) and \( n_j \) belong to the same cluster, and \( \delta_{i,j} = -1 \) otherwise. Basically the modularity measures the discrepancy between the actual average inter-cluster edge weight and the average edge weight assuming edges are connected randomly. The modularity reaches the maximum when a particular cluster division is optimal.

Since weighted modularity is based on comparing and summing up the weights of heavy and
light edges, we found it also suffers from the scaling problem discussed in Section 4.1.4. Therefore, inspired by the idea of weight dependent cluster coefficients, we obtain the unweighted modularity values using the same weight thresholds used in extracting cluster coefficients, and compare the weight dependent modularity of the output networks against the real ones. For each weighted network we first run a minimum spanning tree clustering algorithm [84] to obtain a series of candidate clustering scenarios, then for each weight threshold extracted in Section 4.1.4, we calculate the unweighted modularity for each candidate clustering scenario and find the optimal number of clusters under that threshold. We plot the optimal number of clusters against the weight thresholds for both RPGM and BUS scenarios in Figs. 4.12 and 4.13.

From Fig. 4.12 it is clear that the optimal cluster numbers in the output networks almost exactly matched those of corresponding RPGM scenarios. In Fig. 4.13 we see some discrepancies between the output networks and corresponding BUS scenarios, partly due to the fact that the group structures in the BUS traces are more complicated, and actual group division is not as clear as RPGM traces. Despite that, the optimal number of clusters still match sufficiently well in all three scenarios.
4.2.2 N-Body Model Usage

The output networks is fed into the N-body model to synthesize mobility traces, and network performance of these traces is compared against the original RPGM and BUS scenarios to validate our network generator. We run simulations in Matlab and the ns-2 simulator [77] and collected performance metrics in both DTN and MANET environments.

N-Body Closed Loop Convergence

We plot the average percentage difference between the actual and target mean and standard deviation of pairwise inter-nodal distances during the closed loop pre-synthesis in the N-body model in Fig. 4.14. From the figures we see that the synthesized networks converged quickly during the closed loop stages, which indicates that these networks are good depictions of real life group mobility.

![Figure 4.14: Trends of convergence during the closed loop pre-synthesis.](image)

DTN Epidemic Performance

We further validate the performance of our synthesized traces using DTN epidemic performance: assuming every node generates a packet periodically, and floods it throughout the
Figure 4.15: Average percentage of population reached by packets over time after they are generated, RPGM Scenarios.
Figure 4.16: Average percentage of population reached by packets over time after they are generated, BUS Scenarios.
entire network. To simplify our analysis we assume nodes have infinite buffers, and packets are forwarded instantly once any two nodes are in communication range. For each packet we track the percentage of population it has reached over time, and we plot the average percentage for all generated packets in Figs. 4.15 and 4.16. Such a metric is protocol independent, and reflects the upper bound of DTN forwarding performance.

From both figures we can see that the results from synthesized traces match well with those from original traces. Generally as the communication range increases, the packets get forwarded faster. In all scenarios we found that synthesized traces are more sensitive to changes in the communication range, that’s mainly due to the fact the in synthesized traces the edge of the groups are not as sharp as in the original traces. As nodes occasionally wander away from the groups, increased communication range has a greater impact on packet forwarding. Despite that, the performance of synthesized traces is fairly close to the original traces.

In [85] the author studied the epidemic spreading characteristics of clustered networks. Their conclusion is that for clustered networks, the epidemic spreads faster in the beginning compared to regular random networks due to a rapid infection on nodes within the same cluster, but are slower over time since the chance of an inter group contact is low. As we plot the different scenarios (larger groups, more groups or intermediate) together in Figs. 4.15(j), 4.15(k), 4.16(j) and 4.16(k), we can see such a characteristic in both synthesized and original traces. The “larger groups” scenarios have faster packet spreading in the beginning than the “more groups” scenarios, but slows down later on.

**MANET Performance**

To further validate the synthesized traces, we also used ns-2 network simulator to run MANET simulations using both AODV [78] and DSDV [79] routing protocols. We set up the simulation as each node sends to every other node a packet of 500 bytes every 15 seconds. We set the communication / interfere ranges to 1000 / 2000 m in RPGM scenarios, and 2000 / 3000 m in BUS scenarios. We collected the average end-to-end delay and packet delivery ratio (PDR) for each pair of nodes and compared the results from synthesized traces against the corresponding original ones. The cumulative distribution function (CDF) for both scenarios is shown in Figs. 4.17 and 4.18. From the figures it is clear that in all scenarios with both routing protocols the performance of synthesized traces are close to the original traces.
Figure 4.17: CDF of pairwise packet delivery ratio (PDR) and end to end delay in RPGM scenarios.
Figure 4.18: CDF of pairwise packet delivery ratio (PDR) and end to end delay in BUS scenarios.
Chapter 5

Conclusions and Future Work

5.1 Conclusion

In this paper we conducted research in mobility modeling for wireless ad-hoc networks. We proposed a mobility framework that bridges the gap between the current models by offering high realism and convenience to diversify. In particular, our model is focused on capturing the group-forming tendency in real human movement traces and reproducing them in synthesized traces.

In Chapter 2 we conducted a preliminary research with focus on mathematical analysis of contact time distribution in random walk models, in the hope of bridging the gap between two existing approaches: the direct traversal model and the consecutive random walk model. We show that with uniform speed distributions under the direct traversal model the PDF of contact times has a power-law tail, while previous works show an exponential tail under the consecutive random walk model. We conclude that for general random walks with uniform speed distribution, the PDF of contact times has a tail that is actually between the two extremes: a power-law-sub-exponential dichotomy, which degenerates into the extremes as the flight lengths vary. This conclusion is also validated against RWP models.

In Chapter 3 we studied the group-forming tendency of human movements in mobile ad-hoc networks. We surveyed the existing models and proposed the N-body mobility model that is capable of synthesizing such tendency as observed from target traces. Compared to existing models, our model has the advantage of not requiring any detailed knowledge of the underlying social interaction of the target scenario, therefore has wider scope of application. Our model is validated against the target traces both from real life and synthesized. Simulation results show that our model is capable of capturing the heterogeneity as observed in the target traces.
5.2 Future Work

In the future, the N-body mobility model could be improved in several directions. In the following sections we propose two possible extensions together with some preliminary results.

5.2.1 Geographical Diversity

From the validation results Fig. 3.7, 3.8 and 3.9 it is clear that N-body model does not perform equally well on all three sample traces. Specifically, N-body performs best on RPGM, slightly worse on STBUS, and worst on SFCAB. The reason is that the heterogeneity in the pairwise nodal relationship in the three sample traces is not the exclusive result of the group-forming tendency, but also the geographical diversity, as nodes have individual preference in their movement area. To illustrate this, we calculate the average pairwise distance assuming independent random placement (IRP) but using individual nodal distribution:

\[ M_{i,j}^{\text{ind}} = \int \int_{\Omega} \left| \vec{P} - \vec{P}' \right| \lambda_i(\vec{P})\lambda_j(\vec{P}')d\vec{P}d\vec{P}', \]  

(5.1)

which is similar to (3.4), but differs in \( \lambda_i \) and \( \lambda_j \) corresponding to the individual nodal distributions of node \( n_i \) and \( n_j \), rather than the global distribution \( \lambda \) in (3.4). The IRP distance \( M_{i,j}^{\text{ind}} \) of each pair of nodes is plotted against their corresponding actual distance \( M_{i,j} \) in Fig. 5.1. If there is a strong correlation between \( M_{i,j} \) and \( M_{i,j}^{\text{ind}} \), the heterogeneity in pairwise distance mainly comes from geographical diversity.

![Figure 5.1: The mean pairwise distance assuming independent random placement \( M_{i,j}^{\text{ind}} \) versus the actual mean pairwise distance \( M_{i,j} \). (a) RPGM, (b) SFCAB, (c) STBUS.](image)

From Fig. 5.1 it is clear that in RPGM the IRP pairwise distances have little influence on the actual distances, while in SFCAB the relationship is almost completely linear. In STBUS the
IRP distances do affect the actual distances linearly, but there are also discrepancies. Therefore, Fig. 5.1 reveals that the heterogeneity in pairwise nodal relationships comes from different sources: for the RPGM trace, it mainly comes from group movements; for SFCAB trace, it mainly comes from geographical diversity; and for STBUS trace, it’s partly group movements and partly geographical diversity.

Although N-body model is primarily designed to synthesize the group movements, it does capture a certain level of heterogeneity brought by geographical diversity as in SFCAB trace and STBUS trace. However, such geographical diversity in the sample traces is lost during synthesis. Therefore, an improvement of the current model that distinguishes between the two different kinds of heterogeneity is desirable and will be part of our future work.

5.2.2 Performance Analysis

![Figure 5.2: Influence of mobility model features upon protocol independent or dependent performance metrics.](image)

Although the impact of such group-forming tendency upon performance heterogeneity is already shown in Section 3.3.2 through ns-2 simulation, a more comprehensive and quantitative analysis of such impact of correlated movements should be done. In addition to the group-forming tendency, other common factors may be also analyzed, including but not limited to:

- Speed
- Pause Time
- Flight Length
- Node Arrival/Departure Pattern (Variable Population Size)
• Hotspots (Nodal Population Concentration)

• Geographical Diversity (Section 5.2.1)

The performance measures used may be protocol dependent (end-to-end delay, packet delivery ratio, throughput, overhead, etc) or protocol independent (contact/link duration, inter-contact time, path duration, etc.). Chapter 2 can be considered as preliminary work for such an analysis. The purpose of this analysis is to distinguish the influence of common factors in mobility models upon the system performance, as shown in a simplified example in Fig. 5.2. In the figure we briefly illustrate the relationship between the mobility model features such as speed, pause time, node density, and the performance metrics including link duration, inter-contact time and the packet delivery ratio. The node density is directly related to the average packet delivery ratio in various routing protocols, but does not affect the link duration or inter-contact time between any specific pair of nodes. On the other hand, speed and pause time are parameters of the model dynamics and thus, closely related to link duration or inter-contact time, while they have only limited influence on packet delivery ratio, which largely depends on the connectivity of the network. Similarly, the protocol independent metrics like link duration and inter-contact time has also limited impact on packet delivery ratio.

By doing such an analysis and providing an understanding of the relationship between the mobility model features and their performance implications, it is possible for the research community to tailor the mobility models so that they exhibit heterogeneous performance characteristics, thus be able to easily verify any ad hoc network implementations in diverse mobile environments.
REFERENCES


[71] Han Cai and Do Young Eun. Crossing over the bounded domain: From exponential to power-law inter-meeting time in MANET. In *Proc. ACM MOBICOM’07*, Montréal, Canada, September 2007.


Appendix A

In this appendix we derive (2.8), the relationship between the two definitions of speed distribution. To facilitate the derivation, we start from a discrete speed distribution with $M$ values, that is, $v \in (\xi_1, \xi_2, \ldots, \xi_M)$. For simplicity assume there is only one node in the model, and the node has gone through $N$ flights. We denote the length of the $n$th flight with $s_n$, and the speed driving that flight with $v_n$. We denote the probability that the node selects $\xi_m$ as its speed at the beginning of each flight with $P_g(\xi_m)$, and the probability that at any given time the node is traveling at the speed $\xi_m$ with $P_f(\xi_m)$.

For each $\xi_m$, we define the set of flight indices that the flight is driven at the speed $\xi_m$:

$$\Xi_{m,N} = \{i | i \in \{1, \ldots, N\} \text{ and } v_i = \xi_m\},$$  \hfill (A.1)

such that the probability that the node is traveling at speed $\xi_m$ is:

$$P_f(\xi_m) = \lim_{N \to \infty} \frac{\sum_{i \in \Xi_{m,N}} s_i / v_i}{\sum_{n=1}^{N} s_n / v_n}$$

$$= \lim_{N \to \infty} \frac{1}{\xi_m} \sum_{i \in \Xi_{m,N}} s_i$$

$$= \lim_{N \to \infty} \frac{1}{\xi_m} \sum_{k=1}^{M} \frac{1}{\xi_k} \sum_{j \in \Xi_{k,N}} s_j$$

$$= \frac{1}{\xi_m} \sum_{k=1}^{M} \frac{1}{\xi_k} \frac{1}{\xi_m} \sum_{j \in \Xi_{k,N}} s_j$$

$$= \frac{1}{\xi_m} \frac{1}{\xi_m} P_g(\xi_m)$$

$$= \frac{1}{\xi_m} \frac{1}{\xi_m} \sum_{k=1}^{M} \frac{1}{\xi_k} P_g(\xi_k),$$  \hfill (A.2)

where from the definition of $P_g(\xi_m)$:

$$P_g(\xi_m) = \lim_{N \to \infty} \frac{\sum_{i \in \Xi_{m,N}} s_i}{\sum_{n=1}^{N} s_n}.$$  \hfill (A.3)
Thus, at the limit the discrete (A.2) transforms to the continuous (2.8):

\[ f_V(v) = \frac{\frac{1}{v}g_V(v)}{\int_{v_{\text{min}}}^{v_{\text{max}}} \frac{1}{v}g_V(v) \, dv}. \]
Appendix B

In this appendix we present an alternative method to derive the relative speed distribution in (2.6) through polar coordinates, which is more intuitive and easier to calculate numerically.

Figure B.1: Probability density of $\vec{v}$ plotted on polar coordinate. The distance represents the speed, and the angle represents the direction. Darker shadow means higher probability density. Speed and direction are assumed to be independent of each other. Here speed is uniformly distributed over $[v_{\text{min}}, v_{\text{max}}]$, and direction is uniformly distributed over $[0, 2\pi]$.

It is intuitive to present the probability density of velocity $f_{\vec{V}}(\vec{v})$ on the polar coordinate as shown in Fig B.1, especially when directions are assumed to be independent of speed, and uniformly distributed over $[0, 2\pi]$ (as usually does in RW models). We denote the speed distribution with $f_{V}(v)$. The probability density of velocity $\vec{v}$ of an arbitrary node $N$ as shown in Fig B.1 is:

$$f_{\vec{V}}(\vec{v}) = \frac{1}{2\pi v} f_{V}(v) \quad \text{where} \quad v = |\vec{v}|.$$  \hfill (B.1)
Figure B.2: Two plots from Fig B.1 with origin $N_i$ and $N_j$ are shown such that their shadows partially overlap (double hatched area). $A$ is an arbitrary point in the overlapping area. Then $\overrightarrow{N_iA}$, $\overrightarrow{N_jA}$ and $\overrightarrow{N_iN_j}$ will form a vector triangle, representing $\vec{v}_i$, $\vec{v}_j$ and $\vec{v}_{ij}$, respectively.

When two of such plots, corresponding to two arbitrary nodes $N_i$ and $N_j$, are placed side by side, as shown in Fig B.2, an arbitrary point $A$ in the overlapping area forms a vector triangle with the two origins $N_i$ and $N_j$, where $\overrightarrow{N_iA}$ is $\vec{v}_i$, $\overrightarrow{N_jA}$ is $\vec{v}_j$, and $\overrightarrow{N_iN_j}$ is the relative velocity $\vec{v}_{ij} = \vec{v}_i - \vec{v}_j$. Thus, if the distance between the two origins $N_i$ and $N_j$ is set to their relative speed $v_{ij}$, the PDF of $v_{ij}$ is proportional to the integral of the joint probability density of $\vec{v}_i$ and $\vec{v}_j$ over the overlapping area (the double hashed area in Fig B.2):

$$f_{V_{ij}}(v_{ij}) = C_V \int_{\mathcal{A}(v_{ij})} f_{V_i}(|\overrightarrow{N_iA}|)f_{V_j}(|\overrightarrow{N_jA}|)dA,$$

(B.2)

where $\mathcal{A}(v_{ij})$ is the overlapping area, which is determined by $v_{ij}$. $C_V$ is a normalization factor.

With (B.1) and (B.2) the PDF of the relative speed $v_{ij}$ can be written in terms of the nodal speed distribution $f_V(v)$:

$$f_{V_{ij}}(v_{ij}) = C_V \int_{\mathcal{A}(v_{ij})} \frac{f_V(v_i)f_V(v_j)}{4\pi^2 v_i v_j} dA,$$

(B.3)

where $v_i = |\overrightarrow{N_iA}|$ and $v_j = |\overrightarrow{N_jA}|$.

For most cases it is more intuitive and easier to obtain numerical results from (B.3) than directly through (2.6). For some special nodal speed distributions $f_V(v)$ even a closed form can be easily derived through (B.3). For example, one of those special speed distributions is the linear distribution:

$$f_V(v) = \frac{2v}{v_{max}^2} \quad \text{where} \quad 0 \leq v \leq v_{max}.$$  

(B.4)
Substituting (B.4) into (B.3), the closed form of \( f_{V_{ij}}(v_{ij}) \) is derived:

\[
f_{V_{ij}}(v_{ij}) = \frac{C_V}{\pi^2 v_{max}^4} \int_{a(v_{ij})} dA \\
= \frac{C_V}{\pi^2 v_{max}^4} \left( 2v_{max}^2 \cos^{-1} \frac{v_{ij}}{2v_{max}} - v_{ij} \sqrt{v_{max}^2 - v_{ij}^2} \right). \tag{B.5}
\]

Further, through (2.5) the PDF of \( v_r \) is:

\[
f_{V_{r}}(v_{r}) = \frac{2v_{r}}{\pi v_{max}^3} \left( 2v_{max}^2 \cos^{-1} \frac{v_{r}}{2v_{max}} - v_{r} \sqrt{v_{max}^2 - v_{r}^2} \right). \tag{B.7}
\]
Appendix C

This appendix includes a fix to the original N-body model proposed in [75] in order to avoid node clogging under certain circumstances.

In the original N-body model the attraction force on an arbitrary node \( n_i \) from its destination is defined as to be inverse proportional to its distance towards the destination \( \vec{d}_i \):

\[
\vec{G}_i = G_0 \frac{d_0}{|\vec{d}_i|} \hat{d}_i.
\]  
(C.1)

The reason for this choice is to avoid tie-ups between competing attractions from multiple destinations when the corresponding nodes are strongly attracted to each other and forming a group. When a node gets closer to its destination, its attraction force gets stronger, and vice versa. Therefore any balance between the competing forces will be unstable: a small move in one direction will always increase the total force towards that direction.

However, such intuition may not be always correct on a 2-D surface. Consider the situation shown in Fig. C.1: a node makes a small move of distance \( \Delta d \) from position \( p \) to \( p' \), moving further away from its destination marked by the star. The distance to the destination changes from \( L \) to \( L' \), and the attraction force changes from \( \vec{F} \) to \( \vec{F}' \) correspondingly. There is also an angle between the direction towards the destination and the direction of movement, which is denoted \( \theta \) and \( \theta' \), respectively.

The attraction force from the destination pulls the node back in its movement direction. When the node is at position \( p \), the pull back force denoted \( F_b \) is:

\[
F_b(p) = F \cos(\theta) = G_0 \frac{d_0}{L} \cos \theta.
\]  
(C.2)
When the node moves to $p'$, the pull back force becomes:

$$F_b(p') = F' \cos \theta' = G_0 \frac{d_0}{L} \cos \theta'$$

$$\approx G_0 \frac{d_0}{L + \Delta d \cos \theta} \cos(\theta - \frac{\Delta d \sin(\theta)}{L})$$

$$\approx G_0 \frac{d_0}{L^2} (L - \Delta d \cos \theta) \left( \cos \theta + \frac{\Delta d \sin^2 \theta}{L} \right)$$

$$\approx G_0 \frac{d_0}{L^2} (L \cos \theta - \Delta d \cos^2 \theta + \Delta d \sin^2 \theta).$$  \hspace{1cm} (C.3)

So the decrease in pull back force is:

$$\Delta F_b = F_b(p) - F_b(p') \approx \frac{G_0 d_0 \Delta d}{L^2} (\cos^2 \theta - \sin^2 \theta)$$

$$\approx F \frac{\Delta d}{L} (\cos^2 \theta - \sin^2 \theta).$$  \hspace{1cm} (C.4)

When $\theta > \pi/4$, $\Delta F_b$ will become negative, thus moving away from the destination will actually increase the pull back force instead of decreasing it. The reason is although the attraction force decreases as the node moves away from its destination, the angle $\theta$ is also reduced, thus a larger part of the attraction force is pulling back the node, resulting in an overall increased pull back force.

Therefore, under certain circumstances, competing attraction forces may actually form a semi-stable balance rather than an unstable one. For example, as shown in Fig. C.2 a group of nodes are attracted to four destinations evenly distributed in a circle centered at the group. With such a configuration whenever the group makes a small move in any direction, the additional forces pulling forwards will always be balanced out by additional forces pulling backwards.

In order to fix this problem, we simply change the attraction force to be inverse proportional
to the square of the distance:

\[ \vec{G}_i = G_0 \frac{d_0}{d_i^2} \hat{d}_i. \]  
(C.5)

With the new definition in (C.5), in the situation shown in Fig. C.1, the decrease in the pull back force is:

\[ \Delta F_b \approx F \frac{d}{L} (2 \cos^2 \theta - \sin^2 \theta). \]  
(C.6)

Although this decrease may still be negative when \( \theta \) is large enough, it will never reach a (semi-)stable balance as shown in Fig. C.2: whenever the competing forces reach a balance, such a balance will be unstable in at least some directions. Assuming we have \( N \) competing attraction forces reaching a (semi-)stable balance in direction \( \phi = 0 \), denote the distances to these destinations as \( L_i \) and direction angles as \( \theta_i \), the pull back force decrease should be non-positive:

\[ \Delta F_b(0) \propto \sum_{i=1}^{N} \frac{1}{L_i^3} (2 \cos^2 \theta_i - \sin^2 \theta_i) \leq 0. \]  
(C.7)

In the orthogonal direction \( \phi = \pi/2 \), the pull back force decrease would be:

\[ \Delta F_b(\pi/2) \propto \sum_{i=1}^{N} \frac{1}{L_i^3} (2 \cos^2 (\theta_i + \pi/2) - \sin^2 (\theta_i + \pi/2)) \]
\[ \propto \sum_{i=1}^{N} \frac{1}{L_i^3} (2 \sin^2 \theta_i - \cos^2 \theta_i). \]  
(C.8)
Add the two forces together we have:

\[
\Delta F_b(0) + \Delta F_b(\frac{\pi}{2}) \propto \sum_{i=1}^{N} \frac{1}{L_i^3} > 0 \tag{C.9}
\]

\[
\Rightarrow \Delta F_b(\frac{\pi}{2}) > 0. \tag{C.10}
\]

Thus the balance is *unstable* in direction \( \phi = \pi/2 \).
Appendix D

This appendix includes the pseudo code for the closed loop pre-synthesis. $M_{TH}$ and $S_{TH}$ correspond to percentage thresholds within which the actual statistics $M'_{i,j}$ and $S'_{i,j}$ are considered sufficiently close to the target values; $\delta$ is the linear slope of adjusting $\epsilon_{i,j}$ set by the user. Procedure 1 presents the main steps of the closed loop pre-synthesis:

**Procedure 1** Closed Loop Pre-synthesis

1. $\epsilon_{i,j} \leftarrow G_0/(N-1)$, $\sigma_{i,j} \leftarrow M_{i,j}$, $s_{i,j} \leftarrow S_{i,j} \cdot M_{i,j}$ // Initialize parameters
2. repeat
3. Run N-body simulation for $T_{cycle}$
4. for $i, j \in [1..N], i \neq j$ do
5. Calculate $M'_{i,j}$ and $S'_{i,j}$
6. $\Delta M_{i,j} \leftarrow M'_{i,j} - M_{i,j}$, $\Delta S_{i,j} \leftarrow S'_{i,j} - S_{i,j}$
7. Update the parameters $\eta_{i,j}$ ($\epsilon_{i,j}$, $s_{i,j}$ and $\sigma_{i,j}$)
8. end for
9. until avg($\{|\Delta M_{i,j}|\}$) < $M_{TH}$ and avg($\{|\Delta S_{i,j}|\}$) < $S_{TH}$ or stopped by user

The adjustment in line 7 is presented in detail in Procedure 2.

Alternatively, since the initial values of $\sigma_{i,j}$ and $s_{i,j}$ specified in Section 3.2.4 such that $\sigma_{i,j} = M_{i,j}$ and $s_{i,j} = S_{i,j} \cdot M_{i,j}$ are in fact reasonably good candidates, a simpler adjustment algorithm for line 7 in Procedure 1 that adjusts only $\epsilon_{i,j}$ could be used, as presented in Procedure 3.
Procedure 2 Update $\eta_{i,j}$

1: // Adjust $\epsilon_{i,j}$, $s_{i,j}$ and $\sigma_{i,j}$ according to Tables 3.2 and 3.3
2: if $|\Delta M_{i,j}| > M_{i,j} \cdot M_{\text{TH}}$ and $\Delta S_{i,j} > S_{i,j} \cdot S_{\text{TH}}$ then // $\epsilon_{i,j}$ may not be large enough
3: $\epsilon_{i,j} \leftarrow \epsilon_{i,j} + \delta_{\epsilon} |\Delta M_{i,j}|$
4: else // Assume $\epsilon_{i,j}$ is large enough
5: if $|\Delta S_{i,j}| > S_{i,j} \cdot S_{\text{TH}}$ then
6: $s_{i,j} \leftarrow s_{i,j} - \Delta S_{i,j} \cdot M'_{i,j}/2$ // Adjust $s_{i,j}$ to fix $S_{i,j}$
7: if $s_{i,j} < S_{i,j} \cdot M_{i,j}/2$ then // $s_{i,j}$ exceeds lower limit
8: $s_{i,j} \leftarrow S_{i,j} \cdot M_{i,j}/2$
9: $\epsilon_{i,j} \leftarrow \epsilon_{i,j} + \delta_{\epsilon} |\Delta S_{i,j} \cdot M'_{i,j}|$
10: end if
11: if $s_{i,j} > 2S_{i,j} \cdot M_{i,j}$ then // $s_{i,j}$ exceeds upper limit
12: if $|\Delta M_{i,j}| < M_{i,j} \cdot M_{\text{TH}}$ then
13: $s_{i,j} \leftarrow 2S_{i,j} \cdot M_{i,j}$
14: $\epsilon_{i,j} \leftarrow \epsilon_{i,j} - \delta_{\epsilon} |\Delta S_{i,j} \cdot M'_{i,j}|$
15: if $\epsilon_{i,j} < 0$ then // $\epsilon_{i,j}$ needs to be non-negative
16: $\epsilon_{i,j} \leftarrow 0$
17: end if
18: end if
19: end if
20: if $|\Delta M_{i,j}| > M_{i,j} \cdot M_{\text{TH}}$ then
21: $\sigma_{i,j} \leftarrow \sigma_{i,j} - \Delta M_{i,j}$ // Adjust $\sigma_{i,j}$ to fix $M_{i,j}$
22: if $\sigma_{i,j} < 0$ then // $\sigma_{i,j}$ exceeds lower limit
23: $\sigma_{i,j} \leftarrow 0$
24: $\epsilon_{i,j} \leftarrow \epsilon_{i,j} + \delta_{\epsilon} |\Delta M_{i,j}|$
25: else if $\sigma_{i,j} > 2M_{i,j}$ then // $\sigma_{i,j}$ exceeds upper limit
26: $\sigma_{i,j} \leftarrow 2M_{i,j}$
27: $\epsilon_{i,j} \leftarrow \epsilon_{i,j} + \delta_{\epsilon} |\Delta M_{i,j}|$
28: end if
29: end if
30: end if
31: end if

Procedure 3 Update $\eta_{i,j}$ (Alternative method)

1: // Adjust only $\epsilon_{i,j}$ according to Tables 3.2 and 3.3
2: $\epsilon_{i,j} \leftarrow \epsilon_{i,j} + \delta_{\epsilon} \left( |\Delta M_{i,j}| + \Delta S_{i,j} \cdot M'_{i,j} \right)$
3: if $\epsilon_{i,j} < 0$ then
4: $\epsilon_{i,j} \leftarrow 0$
5: end if
Appendix E

This appendix briefly describes two optional enhancements to N-body mobility model: over-adjustment prevention in closed loop and adaptive stepping time control.

Over-adjustment Prevention in Closed Loop

During the closed loop pre-synthesis, the linear adjustment parameters, such as $\delta_\epsilon$ introduced in Appendix D, has to be carefully determined since adjusting the interaction parameters have complicated impact on nodal behavior. When $\delta_\epsilon$ is too large, it could result in over-adjustment which would lead to fluctuation in the convergence process; while when $\delta_\epsilon$ is too small, it would take a long time to converge. However, even with very small $\delta_\epsilon$, it is still possible over-adjust the interaction parameters at some point of the closed loop pre-synthesis. Therefore, we implemented an optional adaptive parameter scheme to alleviate the over-adjustment problem.

The basic scheme of this enhancement is to for each iteration during the closed loop pre-synthesis to compare its result (average difference in mean and deviation, denoted $\Delta M$ and $\Delta S$) against that of the previous iteration (denoted $\Delta M_p$ and $\Delta S_p$). If the result is not as good as that of the previous iteration, keep re-running the current iteration with $\delta_\epsilon$ cut half each time, until the results are better or a maximum number of re-iteration is reached. The pseudo code is listed in Procedure 4 and can be used to replace Procedure 1 in Appendix D.

Adaptive Stepping Time Control

To increase simulation efficiency of the N-body model, an adaptive stepping time control scheme is also developed. Basically, if the total force a node receives does not change or changes very slowly, larger time steps could be used to improve simulation performance while not seriously affecting results. This scheme is described in detail in Procedure 5.
Procedure 4 Closed Loop Pre-synthesis with Over-adjustment Prevention

1: $\epsilon_{i,j} \leftarrow 0, \sigma_{i,j} \leftarrow M_{i,j}, s_{i,j} \leftarrow S_{i,j} \cdot M_{i,j}$ // Initialize parameters
2: $\Delta M_p \leftarrow$ NULL, $\Delta S_p \leftarrow$ NULL
3: $RIT\_CNT \leftarrow 0$ // Number of re-iterations
4: repeat
5: Run N-body simulation for $T_{cycle}$
6: for $i, j \in [1..N], i \neq j$ do
7: Calculate $M'_{i,j}$ and $S'_{i,j}$
8: $\Delta M_{i,j} \leftarrow (M'_{i,j} - M_{i,j})/M_{i,j}, \Delta S_{i,j} \leftarrow (S'_{i,j} - S_{i,j})/S_{i,j}$
9: end for
10: $\Delta M \leftarrow \text{avg}(|\Delta M_{i,j}|), \Delta S \leftarrow \text{avg}(|\Delta S_{i,j}|)$
11: if $\Delta M_p =$ NULL, $\Delta S_p =$ NULL or $\Delta M + \Delta S < \Delta M_p + \Delta S_p$ or $RIT\_CNT \geq 6$ then // Has better results or reached maximum re-iteration number
12: Reset $\delta$ to original value
13: $RIT\_CNT \leftarrow 0$
14: Update the parameters $\eta_{i,j}$ ($\epsilon_{i,j}, s_{i,j}$ and $\sigma_{i,j}$)
15: $\Delta M_p \leftarrow \Delta M, \Delta S_p \leftarrow \Delta S$, save current node status into $pStatus$
16: else // Prepare for re-iteration
17: if $RIT\_CNT = 0$ or $\Delta M + \Delta S - \Delta M_p - \Delta S_p < \Delta_{opt}$ then
18: $\eta_{opt} \leftarrow \{\eta_{i,j}\}$ // Keep track of the the best result
19: $\Delta_{opt} \leftarrow \Delta M + \Delta S - \Delta M_p - \Delta S_p$
20: end if
21: $RIT\_CNT \leftarrow RIT\_CNT + 1$ // Increase re-iteration count
22: if $RIT\_CNT \geq 6$ then // Re-iterated 6 times, use the best result
23: $\{\eta_{i,j}\} \leftarrow \eta_{opt}$
24: else
25: $\delta \leftarrow \delta/2$ // Exponentially reduce adjustment parameter
26: Update the parameters $\eta_{i,j}$ ($\epsilon_{i,j}, s_{i,j}$ and $\sigma_{i,j}$)
27: end if
28: Restore current node status from $pStatus$
29: end if
30: until $\Delta M < M_{TH}$ and $\Delta S < S_{TH}$ or stopped by user
Procedure 5 N-body Simulation with Adaptive Stepping Time
1: $t \leftarrow 0$, initialize nodes (Steps 1 and 2 in Section 3.2.1)
2: MAX_STEP $\leftarrow 8$, MIN_STEP $\leftarrow 0.0625$ // Constants specifying the maximum / minimum stepping time
3: $\{STEP.T(i)\} \leftarrow$ MIN_STEP // Start from minimum stepping time
4: for $i \in [1..N]$ do // Queues to hold force change history
5: $Q_\Delta F(i) \leftarrow$ empty fifo queue, size = 10
6: $Q_{F_p}(i) \leftarrow$ empty fifo queue, size = 10
7: end for
8: Calculate total forces $\{\vec{F}(i)\}$ as in Sections 3.2.2 and 3.2.3.
9: $\{\vec{F}_p(i)\} \leftarrow \{\vec{F}(i)\}$ // Record previous forces
10: while $t \leq$ SIM_TIME do // Main loop through the whole simulation
11: $ENDT \leftarrow t + MAX\_STEP$ // Each loop lasts MAX_STEP
12: $\{CHECK\_TIME(i)\} \leftarrow \{t + STEP.T(i)\}$ // Time to update each node
13: while $t \leq ENDT$ do
14: $t \leftarrow$ min($\{CHECK\_TIME(i)\}$) // Find nodes to update
15: for $i \in \{i : CHECK\_TIME(i) = t\}$ do
16: Update node position (Steps 3 to 5 in Section 3.2.1)
17: $CHECK\_TIME(i) \leftarrow$ min ($t + STEP.T(i), ENDT$)
18: Calculate total force $\vec{F}(i)$
19: enqueue($Q_\Delta F(i), |\vec{F}(i) - \vec{F}_p(i)|$)
20: enqueue($Q_{F_p}(i), |F_p(i)|$) // Record force change history
21: $\vec{F}_p(i) \leftarrow \vec{F}(i)$
22: // Increase stepping time if past force change is slow
23: if avg($Q_\Delta F(i)/avg(Q_{F_p}(i)) < 0.2$ then
24: $STEP.T(i) \leftarrow$ min($STEP.T(i) \cdot 2, MAX\_STEP$)
25: else
26: $STEP.T(i) \leftarrow$ max($STEP.T(i)/2, MIN\_STEP$)
27: end if
28: end for
29: end while
Appendix F

This appendix includes an optional extension to the network generating method described in Section 4.1. This extension gives users the option of preferring cluster coefficients over degrees when generating networks.

The network generator described in Section 4.1 generates networks based on weight dependent degrees and cluster coefficients derived from an input network. Basically, the generator would first try to ensure the actual degrees match the desired values, then try to achieve the desired cluster coefficients on a best effort basis. Usually when the cluster coefficients are not very high it is possible to satisfy both parameters. However, in many of our networks representing the social interaction in the N-body model, the cluster coefficients are often high (close to 1), sometimes making it impossible to satisfy both conditions. For example, assuming the input network consists of 1 group of 8 nodes and 2 groups of 4, thus half nodes have degrees of 7 while the other half have degrees of 3, and every node has a cluster coefficient of 1. In the output we decided to double the total population while increasing the group size by 50%. After the distorting process described in Section 4.1.3, we have 32 nodes, half with degrees of 11 and half with degrees of 5, and every node has a cluster coefficient of 1. Although it is desired to generate a network consisting 1 group of 12 nodes, 2 groups of 6 nodes, and 2 groups of 4 nodes (or 1 group of 8 nodes), it would require violating the degrees to satisfy cluster coefficients, which is not possible with out current generator.

Therefore, in order to generate a desired network under these circumstances, we developed an optional cluster coefficient fixing phase after generating the network on each weight level described below:

1. On each weight level, after generating the network using the steps described in Section 4.1.5, for each pair of nodes check their potential impact on the global cluster coefficient:
   (a) If they are not connected, record the change of global cluster coefficient if they are connected.
   (b) If they are connected on the current level, record the change of global cluster coefficient if their connection is removed.
(c) If they are connected but not on a higher level, their connection cannot be removed on the current level.

2. Among all the pair of nodes examined, choose the pair that could change the global cluster coefficient closest to the desired value. Add or remove the connection and adjust the stubs accordingly. Record the net increase or decrease in the number of edges on the current level. If the net increase / decrease reaches a certain threshold, in the next round consider only removing / adding edges.

3. If the global cluster coefficient is within a threshold (denoted $GCC_{th}$) from the desired value, or if no edges can be added or removed to improve global cluster coefficient, end the process. Otherwise, go back to the first step.

![Graph](image1.png)

Figure F.1: Comparing global cluster coefficients with / without extension: RPGM-I scenario.

![Graph](image2.png)

Figure F.2: Comparing global cluster coefficients with / without extension: BUS-I scenario.
In our testing scenarios in Section 3.3 we set the threshold $GCC_{th}$ to 0, which means to adjust the network until the global cluster coefficient exactly matches the desired value or no more improvements can be made. We allowed a maximum of 10% and 20% of net increase / decrease in the number of edges on each weight level for the RPGM and BUS scenarios, respectively. We run the network generator without the extension described here on the two intermediate scenarios and plot the global cluster coefficients together with the ones taken from Fig. 4.10(b) and Fig. 4.11(b), in Fig. F.1 and Fig. F.2, respectively. From the figures it is clear that this extension makes a big improvement on the cluster coefficients of the generated networks.