December 2017

Ant-inspired Interaction Networks For Decentralized Vehicular Traffic Congestion Control

Andreas Kasprzok
Clemson University, akasprzok@gmail.com

Ant-inspired Interaction Networks For Decentralized Vehicular Traffic Congestion Control

---

A Dissertation
Presented to
the Graduate School of
Clemson University

In Partial Fulfillment
of the Requirements for the Degree
Doctor of Philosophy
Automotive Engineering

---

by
Andreas Kasprzok
December 2017

Accepted by:
Dr. Beshah Ayalew, Committee Chair
Dr. Andrej Ivanco
Dr. Ardalan Vahidi
Dr. Pierluigi Pisu

Abstract

Mimicking the autonomous behaviors of animals and their adaptability to changing or foreign environments lead to the development of swarm intelligence techniques such as ant colony optimization (ACO) and particle swarm optimization (PSO) now widely used to tackle a variety of optimization problems. The aim of this dissertation is to develop an alternative swarm intelligence model geared toward decentralized congestion avoidance and to determine qualities of the model suitable for use in a transportation network. A microscopic multi-agent interaction network inspired by insect foraging behaviors, especially ants, was developed and consequently adapted to prioritize the avoidance of congestion, evaluated as perceived density of other agents in the immediate environment extrapolated from the occurrence of direct interactions between agents, while foraging for food outside the base/nest. The agents eschew pheromone trails or other forms of stigmergic communication in favor of these direct interactions whose rate is the primary motivator for the agents' decision making process. The decision making process at the core of the multi-agent interaction network is consequently transferred to transportation networks utilizing vehicular ad-hoc networks (VANETs) for communication between vehicles. Direct interactions are replaced by dedicated short range communications for wireless access in vehicular environments (DSRC/WAVE) messages used for a variety of applications like left turn assist, intersection collision avoidance, or cooperative adaptive cruise control. Each vehicle correlates the traffic on the wireless network with congestion in the transportation network and consequently decides whether to reroute and, if so, what alternate route to take in a decentralized, non-deterministic manner. The algorithm has been shown to increase throughput and decrease mean travel times significantly while not requiring access to centralized infrastructure or up-to-date traffic information.

Acknowledgment

I would like to express my gratitude to Dr. Beshah Ayalew for his supervision and support.

This work would not be possible without the financial support and guidance provided by Chad Lau and the Harris Corporation.

Lastly I want to thank Meagan Skelton for her support during the last few years.

Table of Contents

List of Tables

List of Figures

Chapter 1

Introduction

This thesis can be roughly divided into two parts: first, the study of social insects and development of a multi-agent model inspired by their foraging behaviors. This model was subsequently optimized for emergent behaviors that foster congestion avoidance while retaining individual agents' abilities to successfully complete their tasks. Several aspects of the resulting behaviors were then adapted to the second part of the thesis, a connected vehicle simulation utilizing V2V technology for decentralized congestion avoidance. The resulting model's performance is analyzed and several possible improvements evaluated.

1.1 Motivation

Connected vehicle technology has the potential to greatly increase both the efficiency and safety of travel by vehicle, with the capability to reducing accidents in the US by over 500,000 per year [1] as well as cut waiting times at traffic lights in half [2]. Additional travel costs directly attributable to congestion amounted to 6.9 billion hours and 3.1 billion gallons of fuel in 2014, amounting to $160 billion in losses [3]. Reducing congestion goes hand in hand with reducing these costs, and connected vehicle technology promises grand strides in that direction by efficiently leveraging existing road infrastructure instead of costly expansions. Paramount to the adaption of the technology is cheap integration to both entice OEMs to integrate radios into their vehicles and consumers to see value in the increased vehicle costs, estimated to be around $329 per automobile [1]. This added cost however dwarfs the costs of RSU deployment which are estimated to be around $17,600 per site [4]. We therefore decided to focus on developing a decentralized V2V application to ensure functionality under most circumstances, without relying on infrastructure or competing with a slew of other applications by not adding any bandwidth of our own.

1.2 Swarm intelligence

The study of decentralized, self-organized, often biologically-inspired systems is relatively young. Boids [5], a simulation of bird flocking behavior, was first published in 1987, while the term swarm intelligence was introduced in 1989 [6]. Swarm intelligence models usually consist of elementary agents, limited in memory and processing power. Following simple rules, interactions between agents lead to emergent behaviors whose complexity surpasses the capabilities of its unaware executors. Prominent examples of such systems include ant colony optimization (ACO) [7], a probablistic optimization technique for finding good paths in graphs based on the foraging behavior of pheromone-laying ants, and particle swarm optimization (PSO) [8], a metaheuristic which optimizes problems by iteratively moving candidate solutions around a search space using simple rules inspired by bird flocking behaviors.

ACO draws on the observation that ants distribute pheromone trails between areas of interest, such as food sources, and the nest. Other ants who encounter such a path are likely to follow it and reinforce it if they also find food. As the pheromone evaporates over time, paths which are both short and frequently traveled increase in pheromone density while longer paths are less frequently traveled and fade away, optimizing the route. Applied to graphs, it has been used to tackle a variety of issues such as the traveling salesman problem and transportation systems. A drawback of the model is that it relies on stigmergy - communication using the environment - instead of direct exchange of information between agents. Consequently it is unusable in environments that rapidly change or cause pheromones to evaporate quickly and implementation in a transportation system would be difficult.

PSO was first intended as a simulation of the movement of groups of animals such as bird flocks or fish schools and was later observed to be performing optimization. A survey of the wide variety of use cases was published by Poli [9]. Agents in the simulation represent candidate solutions and update their velocity by both the entire swarm's best known position as well as their own best known position until a good solution is found. This requires every agent to be able to communicate with every other agent, referred to as swarm communication structure. This topology however may be adjusted in various ways, for example to only share information with a close subset of particles. This variant bears several similarities to the congestion avoidance model discussed in this work.

1.3 VANETs and ITS

Intelligent transportation systems (ITS) have been formally defined as "are advanced applications which without embodying intelligence as such aim to provide innovative services relating to different modes of transport and traffic management and enable various users to be better informed and make safer, more coordinated and smarter use of transport networks" [10]. ITS provide a tool to reducing congestion alongside expanding infrastructure, by using existing roads more efficiently. Specific applications range from traffic signal control and emergency notification systems to collision avoidance systems and other cooperative efforts. One aspect of these systems is the use of dedicated short range communications (DSRC) to enable radio-equipped vehicles to communicate both with each other, referred to as vehicle-to-vehicle communications (V2V) and with infrastructure (vehicle-to-infrastructure or V2I) such as traffic signals or other roadside units (RSUs).

Mobile ad-hoc networks (MANETs), self-configuring infrastructure-less networks consisting of mobile devices, such as smart phones, can be used as an alternative to cellular networks for data exchange in an immediate environment. Vehicular ad-hoc networks (VANETs) are a type of MANET comprised of radio-equipped vehicles and aforementioned RSUs to enable safety applications such as electronic brake lights or even provide entertainment such as discovery services. Vehicular networks pose a unique challenge as their topology rapidly changes due to high speeds, yet also offers certain constraints as the vehicles are defined to roads. Wireless access in vehicular environments (WAVE), specified by IEEE 802.11P and IEEE1609, is an effort to meet the technical challenges of these networks to achieve low latency usable for collision avoidance systems even in high doppler spread situations as well as function in high density environments such as crowded cities. Maximum transmission power ranges from 0 dBm to 28.8 dBm, roughly equating to a distance of 10m to 1km in unobstructed environments [11].

Use cases for VANETs often include both V2V and V2I communications, especially concerning congestion relief: traffic lights as RSUs can relay the amount of vehicles traveling toward another traffic light in order to optimize signal timings, or RSUs distributed over an area may collect traffic data to be analyzed by a server cluster, with relevant information relayed back to individual vehicles. The use of RSUs however requires a significant investment in infrastructure and will most likely only be seen in major cities and highways initially, constraining the use of VANET technology. Applications which only rely on V2V communications, by way of comparison, may entice faster adoption of the technology.

1.4 Contributions of the thesis

The National Highway Traffic Safety Administration (NHTSA) and the Department of Transportation (DOT) issued a Notice of Proposed Rulemaking in 2016 concerning V2V communications technology. Meanwhile, General Motors has moved forward in offering connected vehicle technology in certain Cadillac models. At this point it seems inevitable that the integration of connected vehicle technology will move forward in the coming years, alongside smarter, semi-autonomous vehicles. An extensive set of disruptive applications for the technology is already in development.

We do not intend to add to the multitude of bandwidth-consuming applications but to achieve the highest impact with the lowest amount of added bandwidth and required data set with a V2V application for both autonomous and non-autonomous vehicles. Our congestion avoidance model can therefore be utilized by a significant cross-section of connected vehicles, and our decentralized approach which does not require a cellular connection ensures functionality in a wide range of environments.

1.5 Organization of the thesis

The thesis is organized in roughly chronological order of the conducted research. Section 2 introduces and elaborates on the ant-inspired model for multi-agent interaction networks without stigmergy and analyze its general performance. Section 4 details the congestion avoidance mechanisms inherent in the model and examines those behaviors in detail. Together these sections cover the first half of the dissertation. Section 5 proposes a congestion avoidance algorithm for transportation networks consisting of vehicles equipped with short-range radios which inherits its many of its features from the ant-inspired multi-agent model, adapted from its 2D environment to the vehicular network's graph-like structure. Finally, section 6 explores a multitude of alterations and possible improvements to the model as well as their impact on performance.

Chapter 2

An ant-inspired model for multi-agent interaction networks without stigmergy

2.1 Abstract

The aim of this chapter is to construct a microscopic model of multi-agent interaction networks inspired by foraging ants that do not use pheromone trails or stigmergic traces for communications. The heading and speed of each agent is influenced by direct interactions or encounters with other agents. Each agent moves in a plane using a correlated random walk whose probability distribution for heading change is made adaptable to these interactions and is superimposed with probability distributions that emulate how ants remember nest and food source locations. The speed of each agent is likewise influenced by a superposition of impetus and resistance effects that arise from its recent interactions. Additionally, the agents use a quorum sensing mechanism to trigger a non-deterministic decentralized congestion avoidance scheme. A discrete time non-deterministic recruitment model is adopted and incorporated to regulate the population of foraging agents based on the amount of food perceived to exist in the environment. Simulation experiments were conducted to evaluate and demonstrate how agents employ the interaction network when foraging in open and closed environments as well as in scenarios with narrow pathways that trigger congestion.

2.2 Introduction

Social insects such as ants, bees, and wasps are known to exhibit complex group-level problem solving capabilities, despite the simplicity of the capabilities and behaviors of the individual insect. This observation has inspired a surge of research efforts to derive computational models that abstract the behavior of social insects. Such models for ant colonies have received significant attention following their successful application to many different types of combinatorial optimization problems. The first of these models and algorithms were proposed by [7]. Today, various refinements of ant colony optimization (ACO) algorithms have been applied to optimization problems ranging from the traveling salesman problem [12] to data network routing and scheduling [13] to protein folding [14]. Further applications influenced by models of the behaviors of ants and other social insects include multi-agent systems for swarm robotics [15]; [16], reconfigurable manufacturing systems [17] and other systems relying on self-organizing principles, such as sensor networks, computing grids and software for business processes and network security [18]. Most discussions on ant-based foraging are based on mimicking ants that communicate through pheromone secretions and create probabilistically optimal trails between places of interest, like the nest (or a home base) and a food source. This kind of indirect communication by altering the environment is known as stigmergy. However, depositing environmental markers may be costly or impractical for artificial systems such as transportation networks and swarms of drones in which we aim to incorporate features of our model. For example, using roadside infrastructure for indirect communication between vehicles is expensive and too restrictive for transportation networks. Therefore, instead of stigmergy, we lean on the fact that many species of insects, especially of the Hymenoptera order, also possess spatial abilities enabling them to navigate their environments, including returning to the nest without the use of a trail [19]. These primarily exploit the guidance abilities afforded by the interactions between agents.

The direct encounters or interactions between agents may be interpreted and used in different ways. In some ant species, such as the ant *Lasius niger*, the encounters manifest as repulsive interactions where ants turn away from each other to prevent congestion [20]. We explore an alternative concept: in lieu of turning away from each agent encountered, the individual agent counts interactions from the recent past and modifies its walk accordingly. When registering many interactions, indicating congestion, agents will move both faster and more directly in order to clear the congested area. Behaviors observed in *Cataglyphis velox*, which have been shown to move more erratically close to the nest [21], contradict our model in this case. We alter several other naturally occurring behaviors as well to emphasize congestion avoidance. For that reason, our model also employs quorum sensing, a mechanism used by the ant *Temnothorax albipennis* [22] and certain honeybee species [23], to choose between future nest sites for purposes of emigration [24]. The quorum sensing (QS) mechanism is also used for similar purposes by honey bees [25] and by the bacteria *Vibrio harveyi* [26]. A good discussion of the non-deterministic, yet resilient and robust characteristics of QS can be found in [27]. In this chapter, we apply the QS mechanism to trigger a non-deterministic congestion avoidance scheme. By estimating the density of other individuals in their environment, the agents can collectively change their behavior [28]. We use this to trigger a congestion avoidance behavior in our agents wherein they try to estimate the center of the congested area and avoid it, finding alternate routes to their destination. This behavior is similar to that observed in pheromone-laying ants, which form additional trails under crowded conditions [29]. A related study of lane formation and congestion avoidance with pheromone trails, specific to the behavior of the army ant *Eciton burchelli*, was conducted by [30]. With pheromone trails absent in our model, the mechanism we propose enables a similar behavior especially in environments consisting of narrow pathways.

There have been many other efforts at modeling multi-agent traffic. The simplest models adopt the total asymmetric simple exclusion process (TASEP) [31] offered for modeling ants, cars and pedestrians. Therein, the agents are treated as particles. In the TASEP process, the particles enter a one-dimensional lattice, composed of individual cells, from one direction. A particle can only advance if the next cell is empty. If the cell is occupied, it must wait. The process can be extended to an n-dimensional grid of cells forming a cellular automaton, in which the cells are updated according to some rule and with regards to the states of both the cell being updated as well as its surrounding cells. By using a 2-dimensional cellular grid, for example, various types of traffic such as trails of ants and pedestrians [32], as well as of vehicular traffic of whole cities [33], may be simulated. Since cells are updated only based on their own state and that of cells in their immediate surroundings, these models are computationally efficient and lend themselves to large-scale computation. The model we propose in this chapter differs from these cell-based approaches in that instead of cell states, our model updates the motion state of each individual agent at each time-step, regardless of its spatial location. This is similar to one-dimensional trail models as those explored by [34], though our model relies on direct touch instead of distance information. Additionally, each agent in our model makes probabilistic decisions about its heading and speed based on its interactions with others in its immediate surroundings.

The behaviors to be outlined in this chapter are loosely based on those described in Gordon's book "Ant Encounters" [35] for red harvester ants. We have recently come across yet another work [36] that attempts to simulate the seed foraging behavior of the ants described in Gordon's book. The model described therein draws heavily on cell-grid based modeling notions that complicate congestion modeling. For example, in that model, ants occupying a grid cell are assumed to walk over each other in heavy traffic to model an agent's/ant's memory of the location of the base and of the last location at which food was found. We will be referring to the objects that the agents are to retrieve as "food", but they can of course be any number of things. The "task" to be completed by the agents is to retrieve all the food items and return them to the base. We combine this with a decentralized congestion and wall avoidance mechanism based on quorum sensing and a notion of an avoidance sector that each agent estimates. In addition, we implement the recruitment algorithm model proposed in [38] to regulate the rate at which agents leave the base. A model following similar rules was evaluated by [39], with a focus on the geometry of the foraging environment and the decision-making process between trail bifurcations. Additionally, Garnier's model was evaluated using Alice micro-robots instead of a computer simulation.

The simulation model of the interaction network proposed here is a discrete-time non-deterministic model that captures the decentralized decisions by individual agents, which subsequently create emergent collective behaviors such as the establishment of chains of agents to food sources, avoiding congested areas and walls, and regulation of the foraging population based on the availability of food. However, we don't claim that the model is faithful to the behavior of any particular species of ants or insects, but rather combines features that have been observed across various species of insects with extrapolations and designs of our own. Our eventual goal is to abstract observations from this microscopic model of a multi-agent interaction network to offer insights for the practical design and management of decentralized networks with minimal needs for extra infrastructure, stigmergy, memory, and bandwidth.

The rest of the chapter is organized as follows: section 2 describes the modeling framework in detail, including navigation and communication between ants, recruitment strategy, and congestion and wall avoidance. Section 3 applies the model to several environments and scenarios and analyzes the results. Section 4 offers some conclusions, including the future envisaged applications of the model.

Chapter 3

Modeling

This section details the different components of the model. We start with a brief description of the modeling environment and list some basic assumptions in subsection 3.1, followed by the model for the navigation and communication in 3.2, the recruitment model in 3.3, and the decentralized avoidance mechanisms in 3.4. Finally, pseudocode of the agent's decision loop is given in subsection 3.5.

3.1 Environment

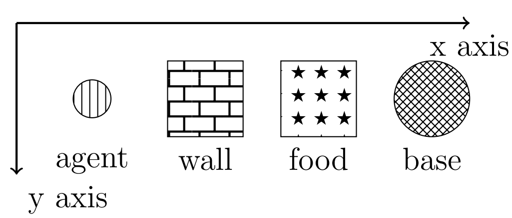


Figure 3.1: Objects in the simulation

The environment in our model is a two-dimensional plane of infinite length and width, though square wall segments may be used to create obstructions, hallways, and other navigational features. Geometrically, agents and the base are modeled as circles, while walls and food items are modeled using squares. The relative size of the main objects in the model is illustrated in Figure 3.1. Agents are homogeneous in size, while the lengths of the sides of wall segments, food items, and the diameter of the base are twice that of an agent's.

Additionally, the following assumptions are made: the base is an unlimited source of agents. Agents that return to the base with food, and therefore memory of that food source's location, are placed in a FIFO (first in, first out) queue. If the queue is empty, a new agent is recruited to forage. Otherwise, the agent at the front of the queue is released in accordance with the recruitment algorithm to be detailed later. Recruited agents possess a random initial heading selected from a uniform distribution. All objects in the model (agents, food, etc.) are represented as rigid, inelastic bodies, unable to move through each other. Collisions between an agent and other agents or objects constitute the basic interactions to be used by the rest of the model. Each collision event is recorded by each agent. An agent that collides with the base while returning to it is removed from the 2D plane and put in a queue to be released again in accordance with the recruitment algorithm.

3.2 Navigation and communication

The motion of the agents is modeled using a correlated random walk. . The change in direction from step i to step $i+1$, denoted by θ, is sampled from the von Mises distribution with probability density function given by [40]:

$$f(\theta) = M(\theta;\mu;\kappa) = \frac{1}{2\pi I_0(\kappa)} e^{\kappa cos(\theta-\mu)} \tag{3.1}$$

where θ takes values in $[0°, 360°)$, μ is the mean heading change, κ is the concentration parameter, which varies the intensity of the peak of the distribution about the mean, and I_0 is the modified Bessel function of the first kind of order 0. Early uses of the von Mises distribution include the modeling of the movement of mammals such as foxes [41], but it has since been applied to model a wide range of biological motions, including those of ants [42].

A key feature of this distribution that we exploit for modeling the persistence of an agent's movement in a specific direction is the concentration parameter κ, whose effect on the distribution is shown in Figure 3.2. A low value of κ leads to an erratic walk, causing an agent to, on average, remain in its immediate surroundings. A high value leads to a highly correlated walk in which the agent likely continues in its current direction.

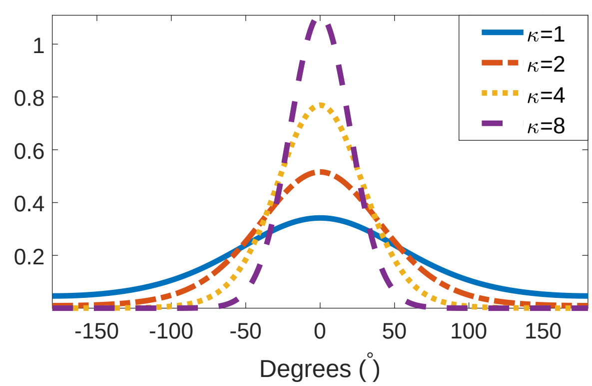


Figure 3.2: Examples of different κ values used in the von Mises distribution

We use the above observation to implement an important feature: *the effect of agent encounters*. The concentration parameter κ is made dependent on recent encounters with other agents. The only information transmitted during the interaction is whether or not the encountered agent is carrying food, and, if it is, its current heading. To model the limited memory of such agents, the impact of each encounter is modeled to decrease exponentially with time. From these recent interactions, an agent computes its κ with:

$$\kappa(t) = \kappa_{min} + \sum_{i=1}^{n} Ae^{-T(t-t_i)} \tag{3.2}$$

where n is the number of interactions and t_i is the time of each interaction's occurrence. A and T are parameters, while k_{min} denotes the minimum concentration parameter of an agent and is used to retain an amount of directionality in the absence of interactions. Note that the above continuous time equation is implemented in discrete time after discretizing it at the sampling rate of the simulation.

In our model, agents who are in densely traveled areas and experience more interactions travel in a straighter line, leading to a subsequent decrease in the density of agents. On the other hand, agents who are further away from the base and therefore, less likely to experience many interactions, will move more erratically. By doing so, they are more likely to remain in their current area and not travel too far without further interactions. The result is a more even distribution of agents compared to a static random walk with a constant concentration parameter. This specific property of the model is discussed further in Section 3.7. If the agent encountered is carrying food, then the foraging agent will change its heading to the opposite of the food-bearing agent in an effort to locate the source of the food. Given that multiple agents are returning to the base with food, the foraging agent will be able to follow this impromptu trail followed by returning agents to the neighborhood of the food source. These encounters simultaneously increase the concentration parameter of the foraging agent, making it less likely to change direction.

Agents return to the base either because they found food or to save energy. To do so, they continuously integrate their path in order to maintain an internal vector of the base's location, similar to ants [43]. This effective base compass is integrated into our model as a second von Mises distribution whose mean always points in the direction of the base relative to the agent's current heading. The two von Mises distributions are added and normalized, and the agent chooses its next heading from the resulting distribution. Figure 3.3 illustrates the implementation of this concept for an agent whose current heading is at a 90° angle to the base. The

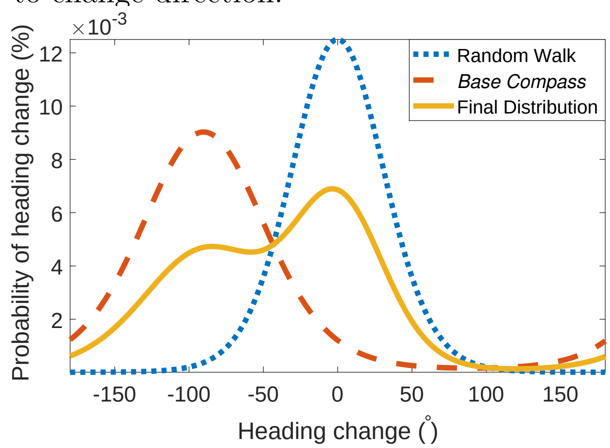


Figure 3.3: Example combination of the von Mises probability distributions for the base compass and the correlated random walk

combination of multiple sources of information pertaining to the agents' random walk with homing trajectories has been observed in *Cataglyphis fortis* [44] and is similar to cue integration, wherein multiple sources of information are averaged based on their relative importance [21].

In several simulation experiments that used a fixed base compass, combined with an interaction-adapted random walk, individual agents exhibited odd behaviors such as overshooting the base or becoming stuck along long obstacles. In these cases, the correlated random walk dominated the agent's navigation at high interaction rates. Therefore, in our model, the base compass distribution's concentration parameter is adapted with interactions as above, while that of the correlated random walk is held at a minimum value k_{min} when agents are returning to the base. This configuration emphasizes returning to the base over exploration and area coverage. Since the density of agents, and therefore encounters, generally increases as distance to the base decreases, the returning agents' confidence in their heading should also increase accordingly.

Once an agent has found food and successfully returned to the base, it is very likely to be sent out again in the near future [35], when the likelihood of retaining memory of the food's location is high. We model this behavior in the form of another von Mises distribution oriented toward the location of the food. It operates just like the base compass, and will be referred to as the food compass. However, unlike the base compass, the concentration parameter κ is not influenced by interactions. Instead, interactions still affect the concentration parameter of the correlated random walk as described above. The food compass's distribution and that of the correlated random walk are added and normalized. The agent chooses its next heading from the resulting distribution. In this manner, the food compass embeds a small, consistent bias toward the food source. If an agent with a food compass arrives at the general area in which it has previously found food, the feature ceases to have any impact on the agent's navigation. This helps the agent avoid staying in the area after the food source has been consumed.

Just like its heading, an agent's speed is impacted by its recent interaction memory. When encountering other agents, an agent desires to move faster either to cover more space, or in order to reach a food source faster. We refer to this as the agent's *impetus*. Yet at the same time, its speed is impeded by the interaction itself: the agents need to shuffle past each other to continue, and touch each other for the interaction to occur. We refer to this opposing effect as *resistance*. We model the combined effect as a sharp decrease in speed followed by a prolonged, but smaller increase in speed :

$$v(t) = v + \sum_{i=1}^{\infty} I e^{-\lambda_I (t-t_i)} - \sum_{i=1}^{\infty} R e^{-\lambda_R (t-t_i)} \tag{3.3}$$

where v is the default speed at which an agent travels without any interactions, and

t_i is the time for the i^{th} interaction. The impetus and resistance parameters, I, λ_I, R, and λ are selected such that $I < R$ and $\lambda_I < \lambda_R$. The *Resistance* parameters R and λ_R are made larger than the impetus parameters I and λ_I in order to achieve an initial decrease in speed. These effects of the interactions are continually added to the default speed v as the interactions happen. Note that the effect of older interactions on the speed diminishes exponentially as time goes on and that the above continuous time equation is implemented in discrete time in the multi-agent simulation

3.3 Recruitment

Since the multi-agent simulation model is primarily motivated by the foraging behavior of ants, the recruitment of foraging agents from the base is abstracted via a discrete time recruitment model proposed by [38]:

$$\alpha_k = max(\alpha_{k-1} - qD_{k-1} + cA_k - d, \underline{\alpha}), \alpha_0 = 0 \tag{3.4}$$

$$D_k \sim Poisson(\alpha_k) \tag{3.5}$$

where α_k is the rate at which agents are recruited from the base at the k^{th} time step, A_k is the number of returning food-bearing foragers at the k^{th} time step, and the actual number of agents departing the base, D_k, is set to a Poisson random variable with α_k as its mean. The coefficient q, c, and d control the contributions of other terms already defined, while $\underline{\alpha}$ denotes the minimum recruitment rate from the base.

Agents leave the nest at a minimum rate $\underline{\alpha}$ and return to the base after a certain period if they don't find any food in order to conserve energy. We refer to this period as a *timeout*. Agents that return to the base without food do not count toward A_k. Therefore, in the absence of a food source, the population of foraging agents will eventually stabilize as ants return at the same rate at which they are sent out. When an agent encounters food, it immediately attempts to return to the base. The number of returning food-bearing agents A_k increases the amount of outgoing foragers D_k. As already noted, food-bearing foragers are assumed to retain memory of the food source's location; and they are likely to be sent back out first in order to retrieve more [35].

3.4 Decentralized avoidance mechanisms

The correlated random walk, the *Base* and food compass and speed modifications described above are sufficient for the multi-agent model to simulate foraging behaviors in an unrestricted environment up to a certain agent density. However, scenarios that constrict agent movement such as walls and tunnels require additional behavior modifications. Large concentrations of agents can lead to traffic jams and pile-ups. We propose a mechanism whereby agents sense or estimate the density of agents based on interaction rates and subsequently the center of the congested area, and use this information to make decisions that help them avoid high-traffic areas. Similarly, agents can be made to avoid contact with a wall or obstruction for a certain time after an initial encounter. These behaviors are entirely decentralized actions consistent with the rest of the model. We detail this congestion and obstruction avoidance mechanisms and behaviors in the following subsections.

3.4.1 Congestion avoidance

As mentioned in the introduction, the quorum sensing mechanism is known to be used by certain species of ants, bees and bacteria for the purposes of collective site selection and emigration. Since the agents in our simulation model already count interactions for navigation purposes (to update their correlated random walk's heading and speed), we use quorum sensing as a trigger for their avoidance strategy. Agents also keep track of the directions from which they have experienced recent contacts/interactions for a certain period of time. Then, each agent estimates an average direction which ideally points toward the center of the congestion as experienced by it. This direction is computed using:

$$\theta_C = atan(\frac{\sum_{i=1}^{m} \sin(\theta_i)}{\sum_{i=1}^{m} \cos(\theta_i)}) \tag{3.6}$$

where θ_C is the heading pointing to the perceived center of congestion, θ_i the direction of contact of a given recent interaction and m is the total number of recent interactions retained during a small time frame. Through experimentation we arrived at 4 seconds as an acceptable value for our simulations. Higher numbers of interactions enable the agent to more precisely estimate both the level and the perceived center of the congestion. Once the agent estimates the direction of the congestion, it creates an avoidance area/sector of a certain number of degrees equally distributed on either side of the congestion direction vector. To be consistent with the non-determinism of the modeling approach, the avoidance sector itself is estimated as a function of the interaction rates as described below. This sector is then excluded from possible headings that the agent might choose at its next decision step, in order to get away from the area of congestion. The probabilities of the remaining directions are renormalized, and used for further heading decisions by the agent.

In Figure 3.4, a foraging agent (vertical lines) has recently encountered several other agents (horizontal lines). Given the directions of the interactions, the foraging agent estimates the average direction of the congestion (dots) and decides to avoid the area. The estimation of the avoidance sector is randomized using a von Mises distribution, whose mean is estimated from recent interactions according to Eq. 3.6, and the arc of the congestion avoidance sector is also made dependent on the perceived amount of congestion experienced by the agent by selecting the sector angle as $\sphericalangle_S =$

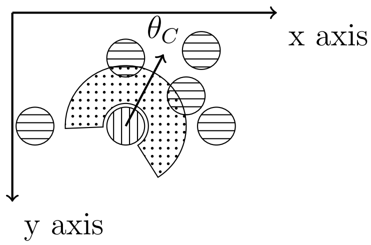


Figure 3.4: Example congestion avoidance sector

2κ, where κ is the concentration parameter (variance) of its heading distribution which is already updated with interactions as discussed before. Consequently, an agent experiencing mild congestion will estimate a small congestion avoidance sector with a random mean, while an agent experiencing high congestion will estimate a large congestion avoidance sector whose mean is more likely to be true to the actual center of congestion. As we illustrate later, this feature has been found to significantly increase the performance of the avoidance mechanism in constricted areas.

3.4.2 Behavior near obstructions

Collisions with obstructions, obstacles or walls, are handled in a similar manner to congestion. When an agent comes into contact with a wall, it will avoid possible further contact with the wall for a certain period of time . It avoids the wall by creating a 180° avoidance sector whose central angle is perpendicular to the encountered wall. Consequently the agent will, for the near future, only move parallel to or away from the wall. Upon wall contact, the direction parallel to the wall and closest to the agent's heading upon contact is the most likely to be chosen as the next heading. Due to the peaked shape of the von Mises distribution, when parallel to the wall, an agent is most likely to continue moving alongside it or away at a slight angle. For our model, we chose 2 seconds as the time span for which this behavior persists. At base speed, an agent will traverse the length of 2.5 wall segments during that time before resuming its normal behavior.

3.5 Pseudocode

The individual agent's decision loop is outlined in Algorithm 1. The decision loop combines all elements of the modeling approach described in the preceding sections. The agent makes decisions several times a second, but continues moving in between decisions. Therefore, interactions and encounters with obstacles can occur in between decision points.

Algorithm 1 Agent Decision Loop

```
procedure DECISION LOOP
    repeat
        calculate κ                                         ▷ Equation 2
        calculate Speed                                     ▷ Equation 3
        if time spent foraging > timeout then
            returning = true
        end if
        calculate heading change                            ▷ See function
CALCULATEHEADINGCHANGE
        move
    until agent has returned
end procedure

function CALCULATEHEADINGCHANGE
    if returning or carrying food then
        add base compass distribution
    else if has food memory then
        add food compass distribution
    end if
    if recently contacted walls then
        avoid walls
    end if
    if κ > congestion threshold then
        estimate mean congestion angle                      ▷ Equation 6
        avoid congestion
    end if
    choose heading change from final distribution
end function
```

3.6 Experiments and results

The simulation model was built in Unity3D 5. It uses Unity's 2D physics system for purposes of collision detection. The physics time step is set to 0.02 seconds, and agents make decisions every 0.1 seconds. Experiments were run for 600 seconds and repeated 100 times for each scenario and averaged results are reported, unless otherwise noted. At every time step, we log data about each agent: its speed, position, ID and concentration parameter. Additionally, time stamps of agents leaving the base and picking up food as well as interaction events are logged for further analysis.

The values of model parameters used in the experiments are listed in Table 3.1. The recruitment algorithm is tuned to provide a steady growth of foraging agents as long as food is available. Navigational parameters are set to have agents move erratically in the absence of encounters, but move with purpose upon encountering other agents. The memory of an interaction is set to decay to negligible amounts after 10 seconds, and each agent's *timeout* is set to 180 seconds. Many of these parameter values were chosen through numerous trials in our simulation environment defined earlier. If different simulation time and spatial scales (agent size, and object/tile sizes, etc) are selected, one could arrive at other parameter combinations that work with the main modeling components.

For the simulation, samples that satisfy the Von Mises distribution (for the correlated random walk, base and food compasses, as well as the randomized congestion avoidance sector) are obtained using the following procedure: The possible heading change of an agent was discretized to 360 degree values. The probability values corresponding to each degree heading change are assembled in the probability vector $P = (p_1, p_2, ..., p_d)$ based on the von Misses distribution. From this, the cumulative probability vector $C = (c_1, c_2, ..., c_d)$ is defined as $c_j = \sum_{j=1}^{i} p_i$ where $j \in 1, 2, ...d$, where d is the index of the degree heading change under consideration. Then, a uniform random value $0 < x < 1$ is found and j determined s.t. $c_{j-1} < x <= c_j$.

In the following sections, we describe the following scenarios which showcase certain behaviors of the multi-agent interaction network as captured by our proposed simulation model: coverage of the search space, foraging behavior and congestion avoidance evaluated in open and restricted environments.

3.7 Coverage of search space

We first compare the multi-agent interaction network with a network created using agents with fixed/static concentration parameters. The latter is a network in which agents are indifferent to interactions. The minimum concentration parameter for the interaction network is selected experimentally as $\kappa_{min} = 3.5$, forming a relatively peaked distribution that can still be adapted substantially by interactions to higher values of κ. Agents are released from the base at a rate of 1 agent per second in unrestricted environment. As there is no food source in this scenario, the rate of recruitment does not increase. The tile size used to quantize the data constitutes a square with side length equal to an agent's radius. The environment is devoid of obstructions for these experiments.

Figure 3.5, which portrays the coverage of the area around the base, shows that the agents cover a mostly regular circular region, decreasing in density inversely to the distance from the base. This coverage map looks visually the same both for the random walk with static κ and the interaction network with adaptive κ (coverage map not shown for the former). However, there are major differences in how coverage is achieved with and without the interaction network. To analyze this, experiments were run at a static $\kappa = 6.4312$, 5.0, 6.0 and 7.0 to compare with the interaction network with the adaptive κ. The value $\kappa = 6.4312$ is the average concentration parameter of the agent population that employs the interaction network.

Table 3.1: Parameters used in the implementation

Umbrella	Parameter	Symbol	Value
Recruitment	Minimum Recruitment Rate	α	1
	Foraging Parameter	q	0.05
	Forager Return Parameter	c	0.005
	Decay Parameter	d	0
Navigation	Minimum κ	κ_{min}	3.5
	Minimum Compass κ	κ_{radar}	4.5
	Interaction Amplitude	A	2
	Interaction Decay	T	20
	timeout (seconds)		180
Speed	Impetus Amplitude	I	1.5
	Congestion Amplitude	C	3.5
	Impetus Decay	λ_I	2
	Congestion Decay	λ_C	1
	Default Speed	v	20
Memory	Interaction Memory		10s
	Wall Memory		2s
	Congestion Memory		4s
	Quorum Threshold	T_q	40
	Decision Interval		0.1s

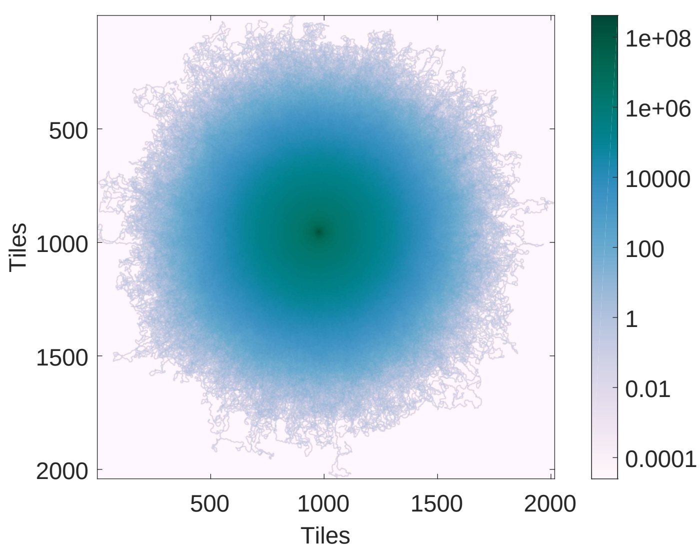
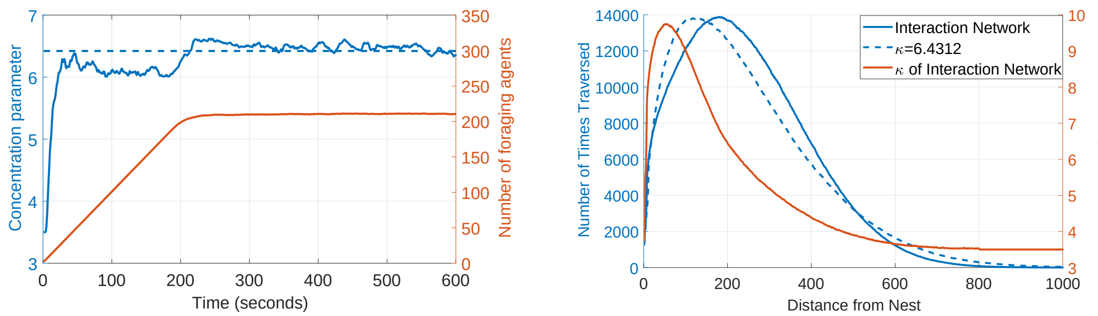


Figure 3.5: Agent density with the interaction network

Figure 3.6a shows that the average concentration parameter of the foraging agents stabilizes quickly. As the population levels off due to agents returning to the base, the average concentration parameter slightly increases due to the increase in interactions between the returning and foraging agents. The distribution of agents around the base is similar between the interaction network and one where agents do not interact but instead possess a concentration parameter κ equal to the mean κ of the interaction network (κ= 6.4312), as seen in Figure 3.6a. However, as depicted in Figure 3.6b, most of the interactions, and therefore, the highest agent concentration parameters are found closer to the base. That is, the interactions near the base effectively push the agents away from it as can be seen in the shifted location of the most traversed tiles for the case with the interaction network. This behavior clearly differs from that with the static case that uses the average $\kappa = 6.4312$.



(a) κ and foraging population over time

(b) Agent density and κ over distance

Figure 3.6: Characteristics of coverage using the interaction network

Table 3.2: Statistics of tiles discovered over 600 seconds

Algorithm	κ value	Tiles discovered	Standard deviation
	5.0	1120086.01	33644.11
Static κ	6	138621.00	43038.18
	6.4312	1386542.73	42644.36
	7.0	1484031.90	44276.23
Interaction network	variable	1171721.83	30156.08

When comparing the statistics of the tiles covered with the interaction network to those of several static κ cases, as shown in Table 3.2, the interaction network is not as efficient. Although its mean κ is high at 6.4312, the cumulative coverage with the interaction network over the 600 seconds is close to that of the static case with $\kappa = 5.0$. However, this outcome can be attributed to the sharp drop-off in the concentration parameter of the agent population in the interaction network as the distance from the base increases (see Figure 3.6b). The drop-off leads to more random motion of agents and to a slightly denser spread of agents as illustrated in Figure 3.6b. This behavior may be beneficial when responding to environmental factors, e.g in the desert for the harvester ants mentioned earlier. Furthermore, as will be evident next, the interaction network is crucial to successful foraging and congestion avoidance.

3.8 Foraging behavior

To study the foraging behavior captured by the model, we situated the base in an open environment without obstructions and located a food source at three distances from the base: at 20, 40, or 60 times the agent radius. The food piles are comprised of 20 stacks arranged in a 3x3 rectangle. Here, the recruitment algorithm outlined in Section 3.3 becomes important as the rate of food-bearing agents returning to the base causes more agents to be sent out in response to the availability of food, as seen in Figure 3.7. Once the food source has been used up, the population of foraging agents returns to a steady state as more agents are still recruited at the minimum rate and they return to the base after their timeout of 180 seconds.

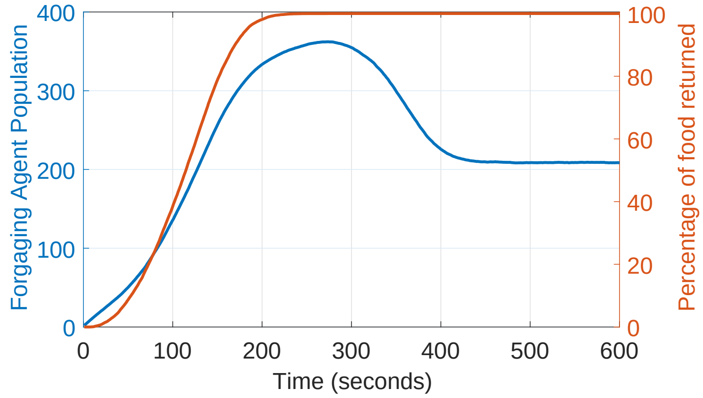


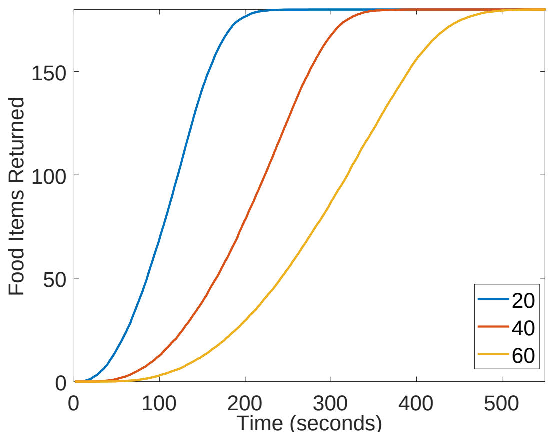
Figure 3.7: Foraging population compared to perceived availability of food over time

Figure 3.8a shows the impact of distance of the food source from the base. We observe that a linear increase in distance of the food source leads to a roughly linear increase in task completion time. The shape of the curves is reminiscent of a sigmoid or Gombertz function: As individual agents establish a memory of the food source and point more agents toward the food source on their return trip; this leads to higher recruitment and the rate of food pickup increases. As the food source diminishes, the rate again decreases as the stacks closest to the base are used up, and the chance of an agent encountering a food stack diminishes.

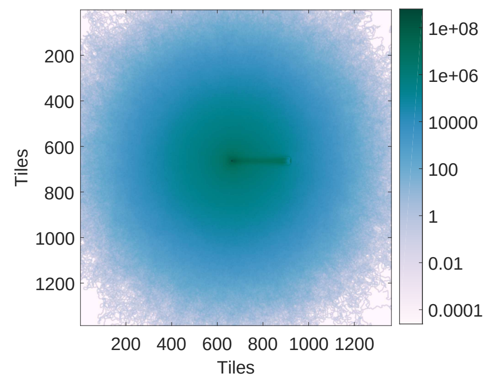
The coverage map in Figure 3.8b shows a high concentration of agents between the base and the food source at a distance of 60 times the agent radius to the right of the base. There is a clear interaction chain created by the direct contact between food-bearing and foraging agents, both with and without food compass. However, a part of the agent population still covers other areas around the base in search of additional food sources.

3.9 Congestion avoidance

As described in subsection 3.4, after the congestion direction is estimated by an agent, it establishes an avoidance sector, a set of directions that it will not consider as long as its concentration parameter is above the quorum sensing threshold required to trigger the congestion avoidance behavior. In the non-deterministic (randomized) implementation, the avoidance sector is modeled using a von Mises distribution with



(a) Food retrieval over time for various distances of food source



(b) Foraging heatmap showing the interaction chain as a dark line from base (middle) to food source (right)

Figure 3.8: Foraging behavior with interaction network in open environment

the mean estimated from recent interactions and its sector $\sphericalangle S = 2\kappa$. In order to evaluate the congestion avoidance feature of the interaction network, we consider the bottleneck scenario shown in Figure 3.9, which is designed to encourage congestion with narrow pathways. In this scenario, the base is on the left and the food is on the right.

We compare the non-deterministic (randomized) avoidance scheme with an implementation that uses a static avoidance sector. The latter does not vary the center of the sector around the mean but it computes a variable sector size based on the κ parameter. Figure 3.10a shows that with the non-deterministic version, there is a significant reduction in interactions and therefore of the concentration parameter after the first 150 seconds compared to the static case. This enables the agents to return food even after the population increased to a point where the simulation without a randomized sector is starting to reach deadlocks around 200 seconds. This is indicated by the sharp decrease in food returns. When evaluating κ with respect to distance from the base, as shown in Figure 3.10b, we note a high amount of congestion close to the base and at the beginning of the pathways for the case with the static avoidance sector, while the randomized version shows a more even distribution of interactions.

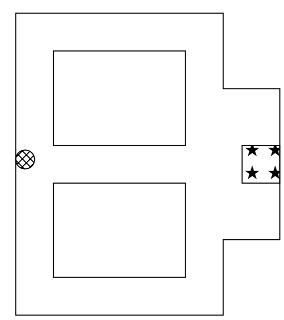
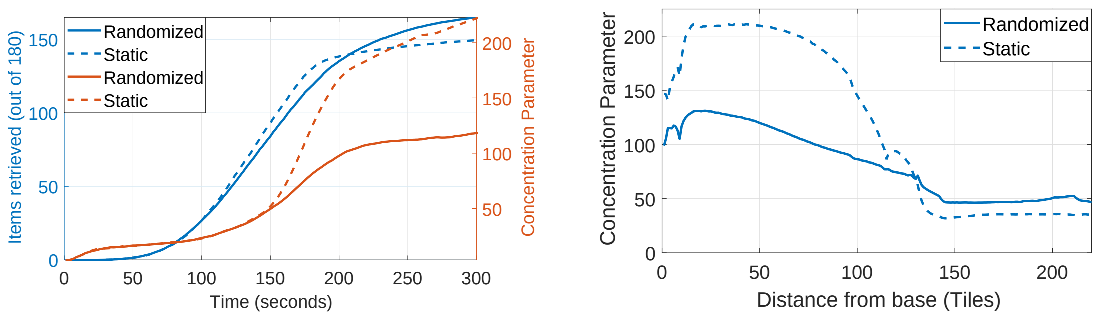


Figure 3.9: Bottleneck scenario

We have omitted discussion of simulation results without congestion avoidance, as they lead to deadlocks in all 3 possible pathways between the base and the food. As a result, the agents are unable to complete retrieval of all food available.



(a) κ and foodpick up over time (b) κ with respect to distance from the base

Figure 3.10: Decentralized congestion avoidance in the bottleneck scenario

3.10 Conclusions

This chapter detailed a non-deterministic multi-agent simulation model comprised of agents that do not use stigmergy for mutual communication. Instead agents primarily interact via direct physical contact. The model has three main components: 1) navigation and communication, which rely on an interaction-adapted correlated random walk superimposed with notions of a base compass and food compass as well as speed updates due to *Impetus* and *Resistance* effects arising from the same interactions, along with decaying memory of these interactions, 2) a decentralized congestion and wall avoidance scheme which is triggered with quorum sensing, and 3) a previously proposed recruitment model that regulates the foraging population. The performance of the overall multi-agent model is illustrated with simulations of search space coverage and foraging in open environments and congestion avoidance in constricted scenarios.

The main conclusion is that it is possible to create common collective behaviors using only the rates of direct encounters/interactions between agents, as opposed to reliance on pheromone trails or other stigmergic cues for communication. The interaction network is able to achieve behaviors such as targeted food retrieval with interaction chains as well as traffic balancing across multiple pathways in a completely decentralized manner. However, unlike traditional stigmergic models, the present model requires superpositions of the correlated random walk of each agent with other secondary distributions of, for example, the *Food* and *Base Compass* to achieve these behaviors. Furthermore, the communication requirements are minimal - for a majority of encounters/interactions the fact of their occurrence is enough, without the need to exchange any information. Only in certain circumstances, like interactions between food-bearing and foraging agents, is any additional information exchanged on encounter between the agents. A non-deterministic collision avoidance scheme is also proposed in which each agent uses the same interaction rates to probabilistically estimate a congestion sector and avoid the area. Finally, we remark that while the key parameters of the multi-agent simulation model we used in our illustrations are given in Table 1, one can easily scale these parameters to examine the collective coverage, foraging and avoidance behaviors in different time and length scales.

Some interesting analyses can be pursued as future work on the proposed model. For example, the trade-off between food retrieval and exploration of the environment can be further characterized in relation to individual agents - like what percentage of agents form the interaction chains from the base to the food sources, compared to the percentage that explore the rest of the environment. By extension we might explore the percentage of agents actually retrieving food, given that agents who have previously returned with food are recruited again for subsequent trips. The response of the interaction network to a changing environment, such as appearing and disappearing food sources or obstacles, would need to be characterized as well. Finally, it is necessary to do a systematic robustness analysis with respect to the parameters required for the different aspects of the model to function - base speed, span rate, memory decay rates, agent geometry, etc. - and to establish specific relationships between these parameters that will result in a successful, optimal interaction network.

We aim to abstract observations from this microscopic model of a multi-agent interaction network and apply them to the design and management of decentralized mobile radio networks such as those that could be deployed on individual vehicles and field robots. Attractive scenarios for this include those where infrastructure such as roadside units or cell towers are not readily available or are very costly. These are similar to environments unsuitable to pheromone trails or other stigmergic traces. We envision such networks to provide services as alternative to those employing vehicle-to-infrastructure communication. Specific aspects include decentralized congestion management of the radio networks and/or of the traffic of the mobile agents utilizing such a network. A preliminary example of extending the basic ideas of the presented model for decentralized vehicular traffic congestion management appears in our recent work [45].

Chapter 4

Decentralized congestion control in multi-agent networks

4.1 Abstract

Interaction networks formed by foraging ants are among the most studied self-organizing multi-agent systems in nature that have inspired many practical applications. However, the vast majority of prior investigations assume pheromone trails or stigmergic strategies used by the ants to create foraging behaviors. We first review an ant network model where the direction and speed of each ant's correlated random walk are influenced by direct and minimalist interactions, such as antennal contact. We incorporate basic ant memory with nest and food compasses, and adopt a discrete time, non-deterministic forager recruitment strategy to regulate the foraging population. The chapter's main focus is on decentralized congestion control and avoidance schemes that are activated with a quorum sensing mechanism. The model relies on individual ants' ability to estimate a perceived avoidance sector from recent interactions. Through simulation experiments it is shown that a randomized congestion avoidance scheme improves performance over alternative static schemes.

4.2 Background

Ants and other social insects exhibit complex problem solving capabilities in groups despite the limited capabilities of the individual insect/agent. They can share information by secreting pheromones into the environment, which can later be picked up by other ants. This mechanism of communication through the environment is referred to as stigmergy and is the focus of a multitude of algorithms and heuristics such as ant colony optimization or ACO [7]. Similar algorithms have since been applied to many optimization problems such as the traveling salesman problem (TSP) [12] to network routing and scheduling [46],protein folding [14] and others [15] [17].

Ants can also communicate directly by exchanging hydrocarbons on antennal contact. In habitats unsuited for environmental markers, ants like *Pogonomyrmex barbatus* and *Lasius Niger* instead rely on encounter rates with fellow ants for navigational purposes: by counting interactions, these ants are able to discern basic information about distance to the nest or points of interest as well as initiate behavioral changes in the case of nest migration. The ant species *Temnothorax albipennis* [22] uses encounter rates to choose new nesting sites: when enough ants congregate at a certain site, increasing encounter rates above a certain threshold and signaling a consensus [28], individual ants change from nest searching to migration. This mechanism is referred to as quorum sensing [24] and is used for similar purposes in honey bees [23] and the bacteria *V. harveyi* [47]. A good discussion of the non-deterministic, yet resilient and robust characteristics of quorum sensing are summarized in [27].

In this chapter, we briefly describe a microscopic model for ants that use encounter rates for the purposes of navigation and congestion avoidance. Many aspects of the model are discussed in further detail in our journal paper [48] and are inspired by the detailed descriptions of ant interaction networks by biologist D. Gordon [35]. Here, we specifically focus on decentralized congestion control by first implementing the quorum sensing mechanism to congestion control instead of colony migration. Each ant is considered to also estimate the perceived center of, and sector around, the congestion when detecting a concentration of ants above the quorum threshold. Three possible implementations of a decentralized congestion control scheme are proposed and evaluated. To regulate the foraging ant population, we adopt the recruitment model proposed by Gordon [38], where the rate of spawning of the ants from the nest is related to the rate of return of food bearings ants.

This chapter is motivated by possible applications of ant-inspired decentralized congestion control that could use minimalist direct communication without using external infrastructure or other analogues of stigmergy. Congestion control problems are prevalent in the management of transportation networks [49], data networks [50], and other multi-agent resource/information dissemination/consumption applications, such as social networks [51].

The rest of this chapter is organized as follows: Section 4.3 reviews the modeling assumptions and the navigation scheme used by each ant, while section 4.4 details the decentralized congestion control scheme. Section 4.5 briefly reviews the recruitment model adopted and section 4.6 presents results and discussions focusing on variations of the congestion control scheme. Section 4.7 concludes the chapter.

4.3 Navigation

The ants' (the agents') movement across a 2-dimensional plane is modeled as a correlated random walk updating at fixed time steps. During their movement, agents may physically contact other agents, which triggers an interaction during which the ants exchange whether or not they are carrying food, and, if they are, their current heading. However, the simple occurrence of the encounter/interaction is the most important information for the agents. The change in heading at each step, θ, is sampled from the von Mises distribution [40] in Equation 4.1,

$$f(\theta) = M(\theta; \mu; \kappa) = \frac{1}{2\pi I_0 \kappa} e^{\kappa cos(\theta - \mu)} \tag{4.1}$$

where κ is the concentration parameter. It varies the dispersion of the distribution: at $\kappa = 0$, the distribution is uniform; when κ is large, the distribution becomes concentrated about its mean μ. I_0 denotes the modified Bessel equation of the first kind. This distribution is commonly applied to model a wide range of biological motions including those of ants [42]. Note that θ denotes the change in the ant's heading direction from step i to step $i+1$. We model each ants' response to recent encounter with other ants or interactions by updating its concentration parameter using equation 3.2.

The speed of each ant is also made directly dependent on the number of recent interactions it experiences as seen in Equation 3.3.

The goal of each foraging ant is to acquire food and return to the nest from which it is spawned. Using the mechanisms described above, the ants travel across the search space. When an ant encounters food or decides to return to the nest after a certain time of foraging unsuccessfully it it does so using a *Nest Compass*. In nature, ants continuously integrate their walk in order to maintain an internal vector of the nests location [43]. We integrate this behavior into our model using a second von Mises distribution whose mean points in the direction of the nest relative to the agent's current heading. This is superimposed onto the ant's correlated random walk. The result is then normalized and used for heading decisions. While an ant is returning to the nest, interactions are used to modify the *Nest Compass* instead of the random walk: as the ant travels closer to the nest, it is more likely to encounter other ants and in return reaffirmed in its general direction. When a foraging ant encounters an ant carrying food, it sets its heading to the opposite of the encountered ant's heading before making its next decision. The assumption made is that the food source will be located in the general area opposite to where the food-carrying ant is headed, since its *Nest Compass* urges it to return. Similar to the *Nest Compass*, ants who have previously found food and successfully returned it to the nest possess a *Food Compass*, which is modeled via an additional von Mises distribution added to the random walk that points to the location at which food has recently been found by the ant. With the above model components, given a certain number of food-bearing and foraging ants, if the food source is concentrated in an area, a interaction chain is eventually formed where ants influence each other in their relative directions and speeds.

4.4 Decentralized congestion avoidance strategy

While the navigational attributes mentioned above enable the ants/agents to successfully forage in open environments, constricted scenarios such as those incorporating corridors and closed spaces greatly hinder the ants ability to forage: they get stuck in corners, are often unable to move past each other when interacting along walls, and may be unable to return to the nest depending on the geometry of the environment. Additionally, congestion in bottlenecks may cause a standstill for the colony and total loss of productivity.

We first describe how we model the behavior near walls or obstacles. When an ant contacts an obstacle other than an ant, the component of its heading perpendicular to the wall is nullified and the ant slows down, covering less distance. Additionally, we have found in computational experiments that this behavior encourages the coalescence of groups of ants along the walls; often times ants keep trying to move into opposite directions yet get stuck as they are unable to efficiently move past each other. To combat this behavior, we endow ants to avoid contact with a wall after an initial encounter with it. All directional changes that would move an ant closer to the direction of an encountered wall are avoided for a short time frame. This avoidance region is taken to be a nearly 178° sector directly perpendicular to the direction of the encountered wall. This simple behavior assumes that the wall extends a certain distance in either direction, but is found to increase the colony's productivity.

Away from obstacles, a decentralized congestion control scheme is implemented starting with the quorum sensing mechanism: when ants sense a high density of other ants around them, indicated by the concentration parameter κ reaching or exceeding a threshold value T_C, the ants change their behavior to prioritize the avoidance of further interactions [52]. To model this, we assume each ant to not only count recent interactions but also the directions from which they occurred. Then, each ant estimates the average direction to the center of the congestion using:

$$\theta_C = atan(\frac{\sum_{i=1}^{m} \sin(\theta_i)}{\sum_{i=1}^{m} \cos(\theta_i)}) \tag{4.2}$$

where θ_C is the heading pointing to the perceived center of congestion and θ_i the direction of contact of a given recent interaction. Higher numbers of interactions enables the ant to more precisely estimate the center of congestion, as well as the level of congestion. Once the ant estimates the direction to the center of the congestion, it creates an avoidance area/sector of a certain number of degrees equally distributed on either side of the congestion direction vector. This sector is then excluded from possible headings that the ant might choose at its next decision step, in order to avoid the congested area. The probabilities of the remaining directions are normalized and then used for further heading decisions. This behavior is illustrated in Figure 3.4, where a foraging ant is avoiding a cluster of ants surrounding it.

The size of the avoidance area can be made directly dependent on the concentration parameter. This is a notion consistent our model of the formation of the ant network formed by interactions.

$$\beta_C = X_C * \kappa \tag{4.3}$$

where β_C is the size of the congestion area, X_C is a scaling constant, and κ is the concentration parameter. When the avoidance area is dependent on the number of recent interactions, a small amount of congestion will result in less evasive action which leads to a less drastic change in ant behavior and higher throughput in moderately traveled bottlenecks.

Instead of the static estimates of the congestion area sector offered above or the one varying with the concentration parameter, we also consider a randomized estimate. This can be done by using the von Mises distribution with a mean estimated as above and a concentration parameter κ of the random walk (indicator of encounter rates). By looking up the direction of the congestion's center as perceived by the agent from this distribution, we introduce some randomness to prevent standstills which may occur when groups of ants traveling in opposite directions encounter each other in corridors or bottlenecks. We shall refer to this case as randomized variable sector in the results below.

For an in-depth discussion on the decision loop of the agents, see [48].

4.5 Recruitment

The recruitment of foragers in the nest is abstracted using a discrete time recruitment model proposed by Prabhakar, Dektar, and Gordon [38]:

$$\alpha_n = max(\alpha_{n-1} - qD_{n-1} + cA_n - d, \underline{\alpha}), \alpha_0 = 0 \tag{4.4}$$

$$D_n \sim Poisson(\alpha_n) \tag{4.5}$$

where α_n is the rate at which ants are spawned from the nest at time n, A_n is the number of returning, food-bearing foragers at time n, and the actual number of ants departing the nest, D_n, is set to a Poisson random variable using the spawn rate as its mean. q, c, and d are parameters for the variables already discussed, while $\underline{\alpha}$ denotes the minimum spawn rate of the nest.

Ants that have previously found food and therefore possess a *Food Compass* are recruited first since according to Gordon [35], a relatively small amount of the foraging population does a majority of the work. The parameters of equations 4.4 and 4.5 are tuned as to recruit ants at a rate similar to the number of ants returning plus the minimum rate. This leads to steady growth of the population until the food source is fully consumed.

4.6 Results and discussion

We implemented the model described above in the modeling environment Unity t 3D using C#. Unity's built-in physics system was used for collision detection between ants as well as their navigation around each other and obstacle. Values of important parameters used for generating the simulation results presented in this section are listed in our journal paper [48]. Each experiment was executed 100 times.

We consider two geometric scenarios to primarily evaluate the decentralized congestion avoidance scheme. Scenario 1, shown in Figure 4.1, depicts the nest area and food area connected by 3 paths. This was chosen to encourage interactions and create congested areas. It is a version of the double bridge experiment [53], though with two longer routes instead of one. The direct route between the two areas will be most traveled due to the ants' *Nest Compass*. We hypothesized that a correctly working congestion avoidance strategy would divert a number of ants from the shortest paths to the two alternate paths. Scenario 2 adds two additional pathways at the north and south ends of scenario 1. We run all tests in this scenario as well in order to evaluate the scalability of our strategy. The task in all experiments is to collect 180 food items arranged in a 3x3 square in the back of the food area. The results for scenario 2 were found to be in agreement with observations from scenario 1, and most of its results were therefore not included here, except where noted (discussion of Table 4.1), for lack of space.

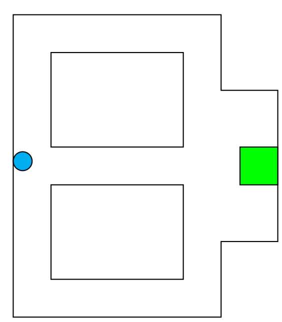
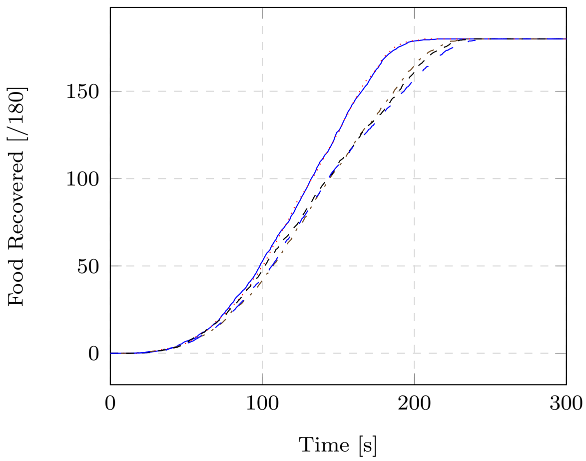


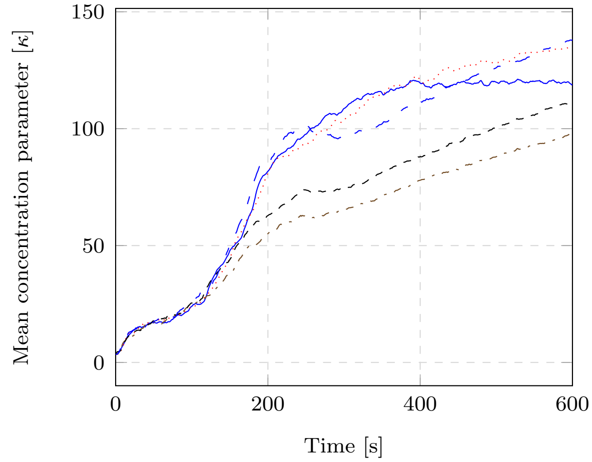
Figure 4.1: Scenario 1

Figure 4.2a shows the task completion times on scenario 1 when using three static sectors (with deterministically estimated perceived center, Equation 4.2; and sectors of 120°, 180°, and 240°), varying the congestion area/sector in relation to the concentration parameter (Equation 4.3) or randomized variable sector. It can be seen that the variable and randomized sectors lead to improved task completion times by about 40 seconds. Furthermore, there appears to be only minute differences between the results for the experiments using static congestion avoidance sectors.

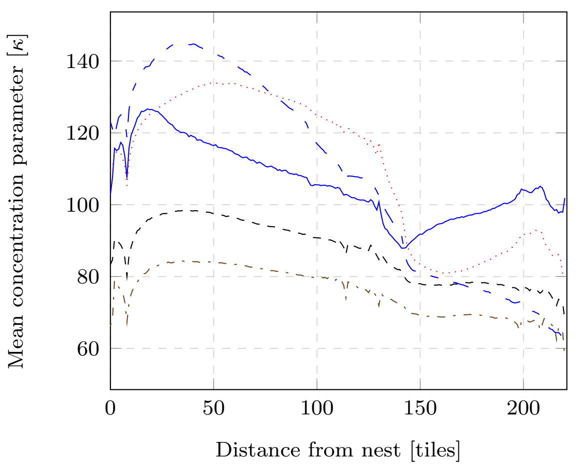
More information about the interactions in the network under the different congestion avoidance settings can be gleaned from Figure 4.2b, which shows the history of the average concentration parameter of the network during the experiment. The concentration parameter is a direct result of the recent interactions experienced by each ant. While a wider static congestion avoidance area setting results in less interactions/second on average, the ants in these experiments are unable to resolve the congestion created by the constricted pathways: the rate of interaction increases



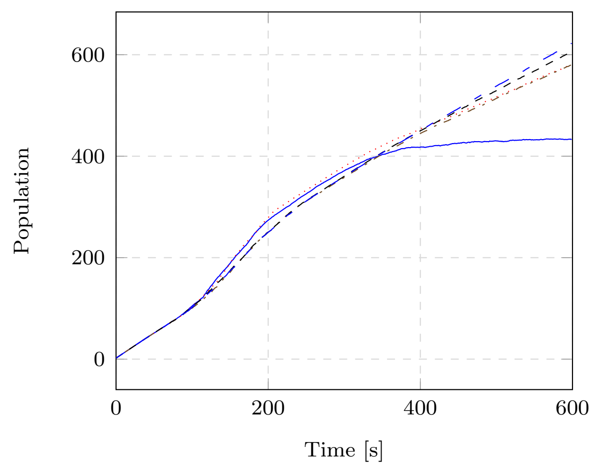
(a) Task Completion Times



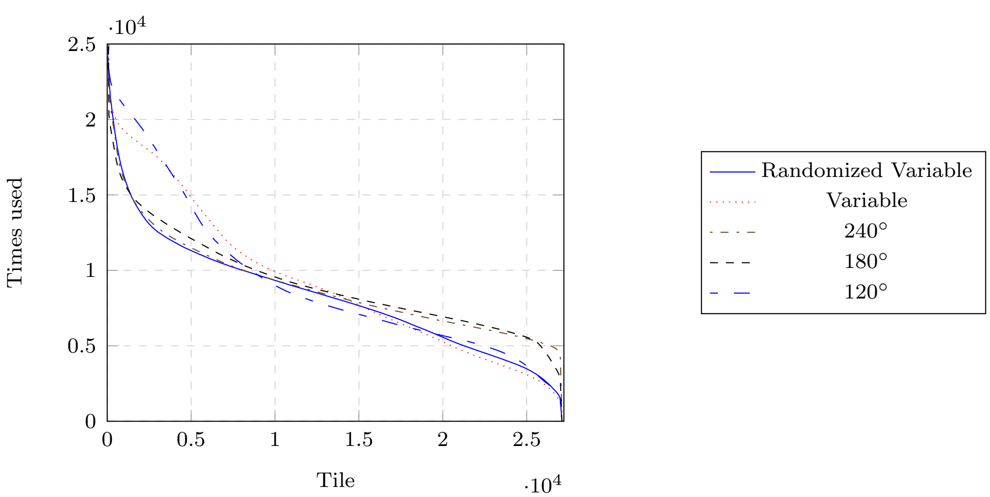
(b) Concentration parameter over time



(c) Concentration parameter over distance



(d) Foraging ant population over time



(e) Rate of tile useage in descending order

Figure 4.2: Experimental data comparing several congestion avoidance strategies

even after task completion (240s, in Figure 4.2a). Varying the avoidance area using κ shows a reduced trend, yet the number of interactions is still steadily increasing with time. However, randomizing the perceived center of congestion seems to allow the ants to move past each other in the congested pathways and reduce both the number of interactions which is reflected in the concentration parameter.

Table 4.1: Mean Avoidance Sector sizes for both scenarios and congestion avoidance strategies

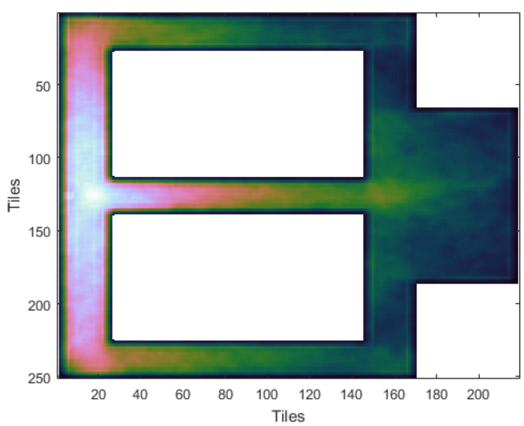
Scenario	Avoidance Strategy	Mean Avoidance Sector (°)
1	Variable	243.3
	Randomized Variable	232.6
2	Variable	229.5
	Randomized Variable	214.0

Table 4.1 references the average sizes of congestion avoidance sectors for both scenarios and the two congestion avoidance strategies using a variable sector size. The size of the congestion avoidance sector for each ant in these experiments was set to 2*κ, or $T_C = 2$. While the difference between the variable and randomized variable sectors are small relative to how differently they perform, the results do support the observations made earlier that the randomized variable strategy leads to less interactions and consequently smaller concentration parameters, thereby reducing the size of the congestion avoidance sector and increasing the efficiency of the network via a faster task completion time as in Figure 4.2a. Additionally, the increased number of pathways in scenario 2 compared to scenario 1 leads to less interactions as the ants are able to efficiently spread out across the increased search space in scenario 2.

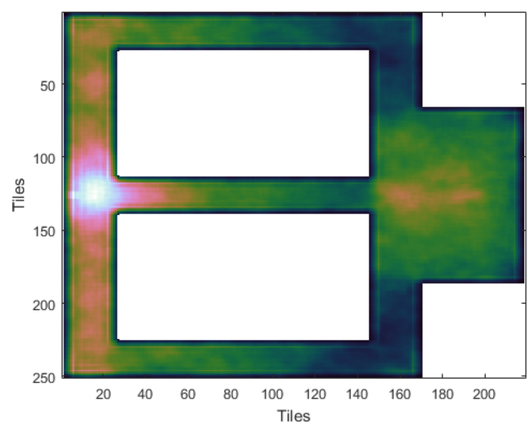
Another way to evaluate the effectiveness of each congestion avoidance strategy or setting might be to measure how well it diffuses ants across the environment. To this effect, Figure 4.2c illustrates the average κ for the agents with respect to their distance from the nest. What this data shows is that a high congestion avoidance angle is beneficial in avoiding interactions, lowering the concentration parameter and indicating even diffusion. The randomized variable strategy performs poorly by comparison, yet we can conclude that even distribution does not necessarily indicate higher throughput, supported by Figure 4.2b. Instead the higher, static congestion avoidance sectors simply leads to a standstill, restricting individual ants' mobility and confining them to their immediate areas.

Figure 4.2d illustrates the foraging population of the simulated ant colony over the length of the experiment. Ants are spawned at the minimum rate according to the recruitment algorithm until the first ants return with food around 100 seconds into the simulation, at which point additional ants are spawned to keep the population growing at a steady pace while food is available. Once the food source has been consumed, the population should stabilize as foraging ants will return to the nest after their 180 second timeout. The population only stabilizes in the case of the randomized variable congestion avoidance area, indicating that in the other cases ants are unable to return to the nest, an observation that is also indicated by Figure4.2b. This confirms our conjecture above that the randomized variable sector gives a realistic decentralized congestion control scheme for the random interaction network created by the foraging ants.

A visual comparison of this trend is shown in Figure 4.3, comparing coverage in scenario 1 between a static 120° avoidance sector and a randomized variable congestion avoidance sector. Note how most of the activity in (a) is confined to the starting area while (b) shows a more even distribution of activity. This visual approximation is supported by Figure 4.2e, which shows that congestion avoidance sectors of size 180° and 240° as well as the randomized variable sector lead to more even tile coverage compared to the smaller 120° sector and the variable sector size.



(a) A static 120° avoidance sector



(b) Randomized variable avoidance sector

Figure 4.3: Examples of cumulative coverage maps for different congestion avoidance strategies

4.7 Conclusions

This chapter detailed a non-deterministic multi-agent simulation model comprised of agents that do not use stigmergy for mutual communication. Instead, agents primarily interact via direct physical contact. The model has three main components: 1) navigation and communication, which rely on an interaction-adapted correlated random walk superimposed with notions of a base compass and food compass as well as speed updates due to impetus and resistance effects arising from the same interactions, along with decaying memory of these interactions; 2) a decentralized congestion and wall avoidance scheme which is triggered with quorum sensing; and

3) a previously proposed recruitment model that regulates the foraging population. The performance of the overall multi-agent model is illustrated with simulations of search space coverage and foraging in open environments and congestion avoidance in constricted scenarios.

The main conclusion is that it is possible to create common collective behaviors using only the rates of direct encounters/interactions between agents, as opposed to reliance on pheromone trails or other stigmergic cues for communication. The interaction network is able to achieve behaviors such as targeted food retrieval with interaction chains as well as traffic balancing across multiple pathways in a completely decentralized manner. However, unlike traditional stigmergic models, the present model requires superpositions of the correlated random walk of each agent with other secondary distributions of, for example, the food compass and base compass to achieve these behaviors. Furthermore, the communication requirements are minimal — for a majority of encounters/interactions the fact of their occurrence is enough, without the need to exchange any information. Only in certain circumstances, like interactions between food-bearing and foraging agents, is any additional information exchanged on encounter between the agents. A non-deterministic collision avoidance scheme is also proposed in which each agent uses the same interaction rates to probabilistically estimate a congestion sector and avoid the area. Finally, we remark that while the key parameters of the multi-agent simulation model we used in our illustrations are given in Table 1, one can easily scale these parameters to examine the collective coverage, foraging and avoidance behaviors in different time and length scales.

Some interesting analyses can be pursued as future work on the proposed model. For example, the trade-off between food retrieval and exploration of the environment can be further characterized in relation to individual agents — such as what percentage of agents form the interaction chains from the base to the food sources, compared to the percentage that explore the rest of the environment. By extension, we might explore the percentage of agents actually retrieving food, given that agents who have previously returned with food are recruited again for subsequent trips. The response of the interaction network to a changing environment, such as appearing and disappearing food sources or obstacles, would need to be characterized as well. Finally, it is necessary to do a systematic robustness analysis with respect to the parameters required for the different aspects of the model to function — base speed, span rate, memory decay rates, agent geometry, etc. — and to establish specific relationships between these parameters that will result in a successful, optimal interaction network.

Observations from this microscopic model of a multi-agent interaction network can be abstracted and applied to the design and management of decentralized mobile radio networks such as those that could be deployed on individual vehicles and field robots. Attractive scenarios for this include those where infrastructure such as roadside units or cell towers are not readily available or are very costly. These are similar to environments unsuitable to pheromone trails or other stigmergic traces. We envision such networks to provide services as alternative to those employing vehicle-to-infrastructure communication. Specific aspects include decentralized congestion management of the radio networks and/or of the traffic of the mobile agents utilizing such a network.

Chapter 5

Decentralized traffic rerouting using minimalist communications

5.1 Abstract

Vehicular ad-hoc networks have the potential to greatly decrease travel times while increasing traffic safety. Potential applications range from simple turn assist to complex traffic management via integration with city infrastructure like traffic lights. In this chapter, inspired by observed behaviors of certain insects, we propose a decentralized congestion avoidance scheme. The approach measures the level of congestion in a road transportation network by the amount of wireless network traffic generated by vehicle-to-vehicle communications. When experiencing high congestion, a vehicle re-routes non-deterministically using a modified K shortest path algorithm whose paths are weighted using a Logit model. To illustrate the workings of the proposed solution, we use a microscopic vehicular traffic simulator coupled with a communication network simulator. The resulting decentralized congestion avoidance scheme requires only a minute amount of bandwidth and it can be implemented on top of existing vehicle-to-vehicle communication applications by both autonomous and non-autonomous vehicles alike. Significant improvements were obtained in traffic throughput with the proposed approach.

5.2 Introduction

Congestion is a widespread problem in today's traffic networks [54], escalating from an already impressive estimated cost of \$37.5 billion in 2004 [55] to \$121 billion in 2015 [56]. While the building of additional transportation infrastructure like highways, streets, subways, and trains can alleviate congestion, it is a very costly solution. Intelligent use of the existing road infrastructure can decrease travel times, CO_2 emissions, fuel consumption, and driver frustration. This can be achieved, for example, by connecting vehicles using cheap short-range radios so that they share information about traffic status on nearby roads.

Vehicular ad-hoc networks, or VANETs, are a type of mobile ad-hoc networks specifically consisting of vehicles communicating with each other (vehicle-to-vehicle or V2V) or roadside units (RSUs, vehicle-to-infrastructure or V2I) [57]. These networks enable the development of applications ranging from active safety like left turn assist to improved traffic efficiency like the dissemination of traffic information and smart route planning [58]. However, the high mobility of nodes in the network, and therefore constantly changing topology, leads to challenges with regard to stability and reliability of the connections [59]. As a consequence, roadside units are often assumed as an integral part of these networks. However, RSUs may not be present in smaller cities or emerging regions.

For this reason, we have decided to eschew V2I and focus on creating a universally useful V2V application. Furthermore, we draw inspiration from certain insects that form interaction networks chiefly by counting physical encounters to achieve complex tasks such as navigation and foraging for food, area coverage and decentralized congestion avoidance [60] [61]. In analogous fashion, we explore the possibility that by measuring the amount of network traffic in the VANET, either as created by other applications such as file sharing or active prediction [62], or by beacon messaging [63], and using it to estimate the congestion in the transportation network, individual vehicles make routing decisions in a completely decentralized manner. Using non-deterministic route selection, we illustrate that it is possible to relief congestion and balance the traffic load without requiring any additional information exchange between vehicles. In the version discussed in this chapter the rate of interaction, not the information itself, is the deciding factor. While the performance of such a decentralized approach may not compare favorably with a cloud-based solution like Google Maps or WAZE, we note that the approach could function even in the absence of a data connection and we surmise that it may be able to respond to changing conditions with less latency. The low bandwidth requirement of our model and integration into other applications means that it can be realized using the IEEE 802.11p Wireless Access for Vehicular Environment (WAVE) protocol using the Dedicated Short Range Communications (DSRC) spectrum in the 5.9 GHz band [64].

The rest of the chapter is organized as follows: Section 5.3 describes the inspiration for the work further, and reviews options for practical congestion measurement and traffic routing. Section 5.8 discusses the routing algorithm we adopt for our purposes. Section 5.9 elaborates on how we use the microscopic traffic simulator SUMO, coupled with the OMNet++ network simulator using the VEINS framework. Section 5.10 evaluates the model using a base case scenario. Finally, Section 5.11 summarizes the chapter.

5.3 Background

5.4 A Microscopic model for multi-agent interaction networks without stigmergy

As mentioned above, we draw inspiration from interaction networks created by certain insects like harvester ants, in which the basis for navigation and routing decisions are minimalist encounters between agents [35] [20] [22]. In our previous work [60], we developed a microscopic model for such multi-agent networks that create complex foraging behaviors with simplistic physical interactions in which barely any information is exchanged. No pheromones or environmental/infrastructure markers (stigmergy) are involved. Instead, the rate of interactions inform the individual agents' modifications of its correlated random walk. The model also includes a decentralized congestion avoidance scheme which uses the direction of recent interactions to create an arc of potential headings which are to be avoided. This mechanism is explored in depth in our subsequent paper [61]. We adopt the non-deterministic nature of the agents' heading decisions as well as their reliance on interaction rates for decision making as the basis of the model explored in this chapter.

5.5 Measuring congestion

Congestion, whether caused by extraordinary factors such as accidents or as a manifestation of everyday traffic in urban areas, presents an important problem increasing fuel consumption, emissions, and travel times. Its measurement and detection can be realized in a multitude of ways.

The simplest method is to define congestion as traffic that is moving at a slower rate than permitted by the road [65]. Traffic intensity may also be measured by carefully deployed surveillance systems such as traffic cameras [66], forming sensor networks whose data can once again be analyzed by central servers. However, infrastructure such as this is costly to deploy and maintain, making it available only to the busiest of intersections in major cities. VANETs reduce the need for expensive sensor networks. Another method is to correlate the amount of congestion with the density of vehicles in a certain area [67]. We will adopt a version of the latter approach: by estimating the density of vehicles in the immediate vicinity using wireless network traffic between vehicles, we compute a *congestion parameter* that is then used as the basis of our model's behavior. Ways this parameter can be derived include control channel (CCH) messages assuming DSRC/WAVE, other beacon messaging applications, or even applications which send information non-periodically.

5.6 Routing

Route assignment in order to improve traffic flow and reduce travel times can be implemented in a variety of ways. A few examples which route cars as they enter the network are listed below, however none of them re-route the vehicle as it travels. The simplest and most inefficient form of routing discussed here is to find the shortest path from source to destination - *shortest path routing*. In a transportation network topology, consisting of major roads connecting areas of varying traffic demands, congestion is likely to occur using this approach. *One-shot routing*, by contrast, incrementally assigns vehicles to routes as they enter the network according to current travel times. The resulting trips do not represent a stochastic user-equilibrium, but drastically improve throughput and travel times in most networks over shortest path routing. We chose this method as a basis of comparison for our model due to its simplicity, though it does require route and travel time information for each vehicle in the network at the time of departure, while the vehicles in our model only possess limited up-to-date information about their immediate vicinity. *Dynamic User Assignment (DUA)* is an iterative process in which the simulation is run and then a number of vehicles are chosen to be assigned alternative routes. This is repeated for a number of steps and also does not guarantee an equilibrium state - and can not be applied to live scenarios due to its iterative nature - but can be useful in analyzing transportation networks. *Stochastic User Equilibrium (SUE) Assignment* routes vehicles by determining a number of k shortest paths to the destination and then weighing them according to a choice model, for example Logit, C-Logit, Probit, or modified Lohse Logit [68]. Our model functions similarly, except vehicles begin by computing the shortest path and only using the k shortest path algorithm and Logit model upon encountering a congested area as measured using the *congestion parameter*.

5.7 Related work

A study using beacon messaging by vehicles subject to accidents in order to re-route other vehicles was conducted by [69]. [70] outlines a similar approach to ours in which vehicles send beacons when approaching an intersection, asking for information in routes of interest in order to evaluate congestion and make an informed routing decision. CoTEC (Co-Operative Traffic congestion detECtion) is a traffic congestion quantification model using traffic density acquired via beacon messaging and the vehicle's speed and processed via fuzzy logic to detect congestion [71]. Once a vehicle detects congestion, it exchanges traffic estimation messages with nearby vehicles to further estimate the extent and duration thereof. SOTIS (Self-Organizing Traffic Information System) uses a decentralized V2V approach by disseminating periodic data packets between vehicles, which is then used to analyze the traffic situation by every individual agent [72]. The analysis is then re-distributed to surrounding vehicles. StreetSmart Traffic has vehicles disseminate information about roads on which they had to travel with slower speeds than anticipated [73]. These approaches function well, but require the distribution of information between vehicles which is specific to their application.

5.8 Congestion-triggered rerouting

Once the *congestion parameter* (see section 5.5) reaches or exceeds the threshold T_C, the vehicle makes a decision about whether or not to reroute, and, if so, which route it should take. The threshold is implemented to avoid any and all route alterations without sufficient information to indicate that a route change may have a positive impact on travel time. Additionally, the rerouting response will not trigger again until the *congestion parameter* has decreased below a reset threshold, T_R, where $T_R < T_C$. Yen's Algorithm [74] is used to find K shortest paths, A^1 through A^K, from the vehicle's current location to its destination. However, in order to facilitate the relief of congestion near the vehicle's current location, the amount of spur nodes used to determine additional short paths was reduced: instead of ranging from the first node to the next to last node, only up to the first 3 spur nodes were used to determine alternative paths. Consequently, A does not contain K shortest paths anymore, but the shortest path A^1 and collection of the shortest paths which deviate from the overall shortest path within the next 3 nodes, A^2 through A^K. The modification prevents paths which do not immediately contribute to congestion relief by omitting those who do not differ from the current path on which congestion is encountered, until the vehicle has already endured the congestion it is currently experiencing. The modified Yen's algorithm is outlined in Algorithm 2.

These options are weighed against each other and assigned a probability using the modified Lohse Logit model [75], similar to SUE assignment. However, the vehicle does not possess any information about the routes of other vehicles and can therefore only use the cost of the paths as stored in its memory.

$$p(k) = \frac{exp[-(\beta X_k)^2]}{\sum_{h \in R_y} exp[\![-(\beta X_h)^2]\!]} \qquad \forall k \in R_{ij}, \forall i \in I, \forall j \in J \tag{5.1}$$

where $X_k = \frac{C_k}{C_{min,ij}}$ and $C_{min,ij}$ is the travel cost of the shortest route of origin-destination pair ij. R_ij is the set of routes from i to j. β is the dispersion parameter of the perception of travel time among drivers as empirically devised by Lohse and in form of $\beta = \frac{12}{1+exp(0.7-0.015C_{min,ij})}$.

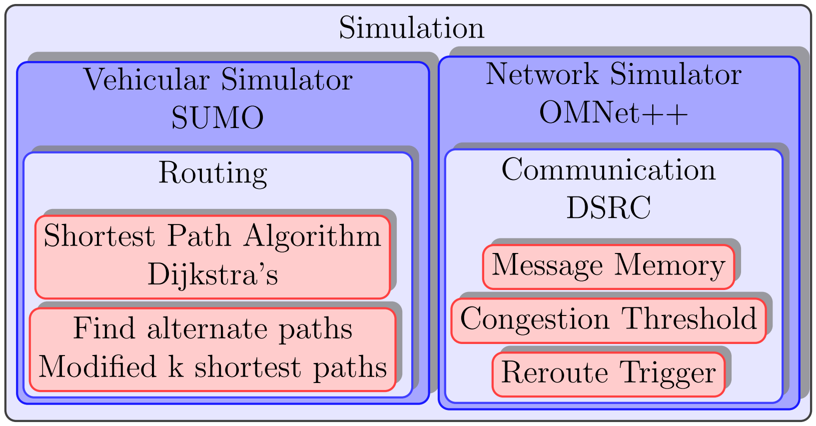


Figure 5.1: Model implementation overview

5.9 Network and traffic modeling

Our simulation consists both of a network simulator, OMNet++, and a traffic simulator, SUMO, coupled using the open-source framework VEINS. We equip each vehicle with a radio transmitter to send multi cast messages using the IEEE1609 WAVE standard.

SUMO (Simulation of Urban MObility) is a GPL-licensed traffic simulator written in C++ [76]. The simulator is able to import networks from a multitude of sources and possesses a host of utilities for creating traffic demands. Its feature set includes a TCP client/server architecture dubbed Traffic Control Interface, TraCI for short, enabling the setting and retrieval of variables in SUMO.

OMNet++ is a event-based network simulation framework written in C++ [77]. It is able to simulate a wide variety of networks, but our use is confined to wireless communication networks. Alternatives include J-Sim [78], NS-3, and OPNET. OMNet++ was chosen for its ease of integration with SUMO via the VEINS framework, its maturity, and our familiarity with the programming language. OMNet++ also features built-in data collection, which was used extensively for this project.

VEINS is an open-source framework that couples OMNet++ and SUMO along with several other tools, enabling realistic vehicular network simulations [79]. Bidirectional coupling via SUMO's TraCI TCP server enables us to influence a vehicle's routing based on its communications with surrounding vehicles. Messages received are stored in the OMNet++ node, which can send a reroute command to SUMO via TraCI. The modified Yen's Algorithm and Lohse Logit model have both been implemented in SUMO. Each simulator's roles are outlined in Figure 5.1.

5.10 Experiments

The proposed congestion avoidance approach was evaluated using the reduced scenario in Figure 5.2. A east/west main road is divided by an intersection with traffic lights and surrounded by capillary roads which may be used by vehicles to avoid the congestion created by the intersection. While the main roads are two lanes wide, the capillary roads are one-way streets: east to west to the north of the main road, and vice versa, to avoid left turns which could increase the variance encountered in the experiment. Using SUMO's randomTrips.py script, 3600 trips were created with a period of $1/sec$. Important to note is that if a vehicle cannot be inserted on an edge due to heavy traffic, it waits until insertion is possible, in effect extending the road segment as far as needed. Potential starting edges include the west- and east-most edges as well as the edges to the south and north of the intersection. Trips have been generated to be 10 times more likely to start on the east or west edges, and all trips terminate on one of those two edges. This creates heavy congestion designed to create a traffic jam on both sides of the intersection. The traffic light favors east/west traffic with a 31 second long green period, while the north/south road is green for 6 seconds, and the length of the yellow light period is 4 seconds.

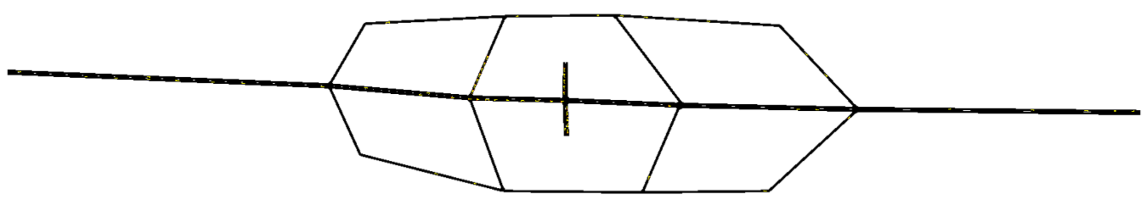


Figure 5.2: The reduced scenario

We compare our model with a base case in which all vehicles simply take the shortest route to their destination as well to the one-shot routing algorithm (see section 5.8). In the case of the reduced scenario used here, this means that, using shortest path routing, the vehicles only use the main east/west road and avoid the capillary roads around the intersection, while both our model and one-shot routing will use the capillary roads depending on the amount of traffic on the main road.

Vehicles are equipped with a 20mW short-range radio and communicate using WAVE with an empty frame body field. The payload may instead by used for other applications. Even though beacon periods of down to 0.1 seconds are supported by modern DSRC technology, beacon messaging with a period of 3 seconds is used. Vehicles retain each message for 3 seconds, and then a reroute is triggered as the message memory exceeds 20 messages, meaning $T_C = 20$. $T_R = 10$ was chosen for the reset threshold.

An example of the utilization of the capillary roads for a single experiment using the rerouting algorithm we propose is illustrated in Figure 5.3. Uneven utiliza-

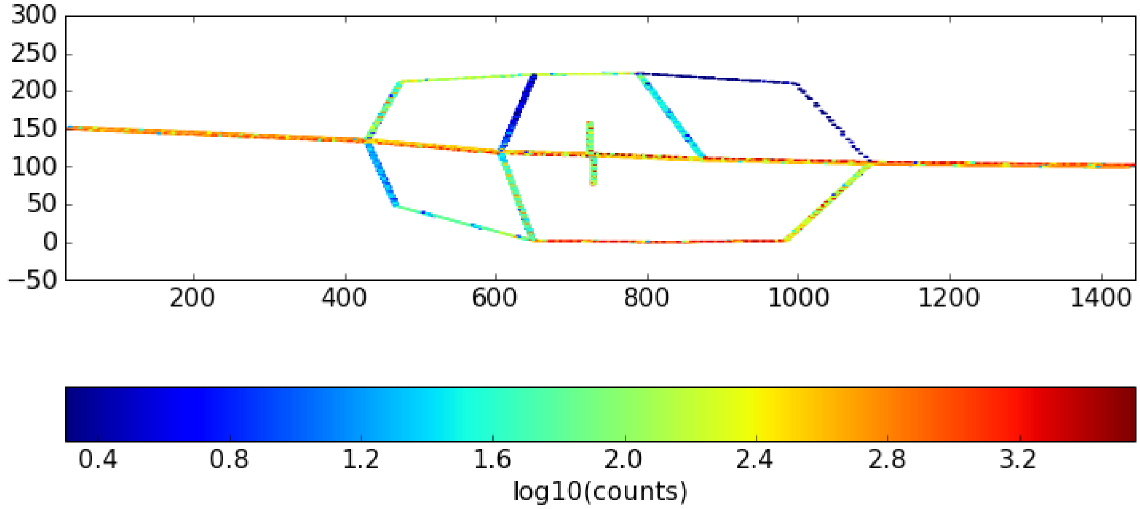


Figure 5.3: Heatmap of vehicle density using rerouting algorithm

tion indicates that the algorithm's parameter may be tuned even more aggressively, as the majority of vehicles still use the main road. This is a result of the use of the Lohse Logit model, which always examines the shortest path as the most desirable, which may not be the case in heavily congested scenarios.

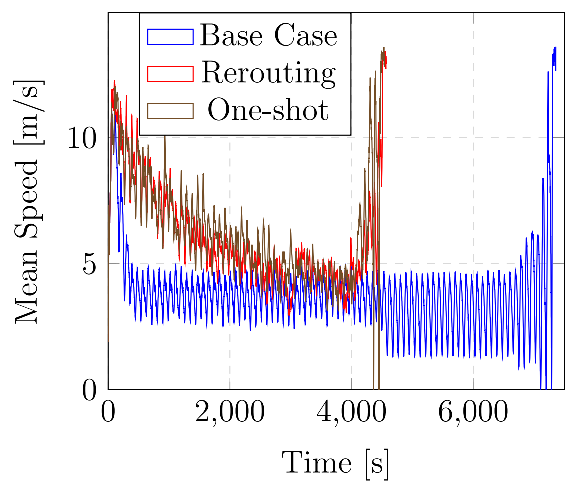


Figure 5.4: Mean speed over time for vehicles using different routing algorithms

Figure 5.4 shows that the base case quickly reaches a bottleneck, while both our congestion avoidance model and one-shot routing utilize the capillary roads, increasing mean speeds and throughput. The regular pattern of the stop-and-go traffic is reflected in Figure 5.5, showing that most vehicles have a very regular mean velocity in the base case simulation, but about 31.9% lower than our rerouting algorithm, as seen in Table 5.1. The mean speeds in the rerouting case decrease over time as the capillary network reaches capacity, and then sharply increase as the last of the cars leave the network. Meanwhile, speeds in the base case remain low due to its low throughput. Since the mean distance traveled by each vehicle only increases by about 10%, we achieve an overall reduction in travel time averaging 33.4%. The reduction in CO_2 emissions of about 18.5% is attributable to vehicles on the capillary roads which are not subject to the stop and go traffic caused by the intersection. Most importantly, using our model, all 3600 vehicles are able to traverse the road network in 4567 seconds, an increase in throughput of 37.9% over the base case, which required 7359 seconds for all vehicles. Our model actually compares very favorably to one-shot routing in the main metrics discussed in the table.

Table 5.1: Comparison of rerouting and one shot algorithm to shortest path routing

	Base Case	Rerouting	One Shot
Mean Time (s)	299.6(+33.4%)	224.5	216.5(−3.57%)
Mean Speed (m/s)	3.79(−31.9%)	5.57	5.79(+3.9%)
Mean Distance (m)	1137.1(−9.0%)	1250.1	1252.9(+0.2%)
CO_2 (mg/s)	506.3(+18.5%)	427.3	405.2(+0.2%)
Completion time (s)	7359(+37.9%)	4567	4541(−0.6%)

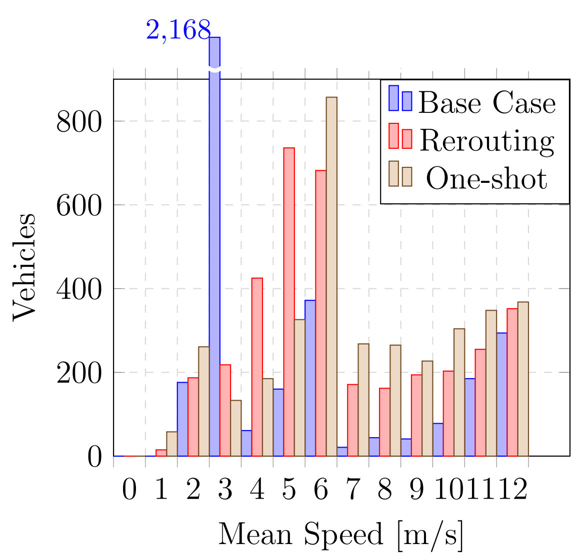


Figure 5.5: Speed over time comparing the three models

Figure 5.6 shows the same regularity in the base case as Figure 5.5, this time with regard to travel time. While mean travel time is significantly reduced with our decentralized rerouting algorithm, the histogram shows a drawback to our current model: travel time for a small number of vehicles increases noticeably compared to the base case. We are currently evaluating the model in more scenarios to verify that this is a consistent result, but no matter how infrequent, highly negative experiences for a small group of drivers and passengers should be avoided. However, One-shot routing shows an even greater disposition toward this behavior than our model.

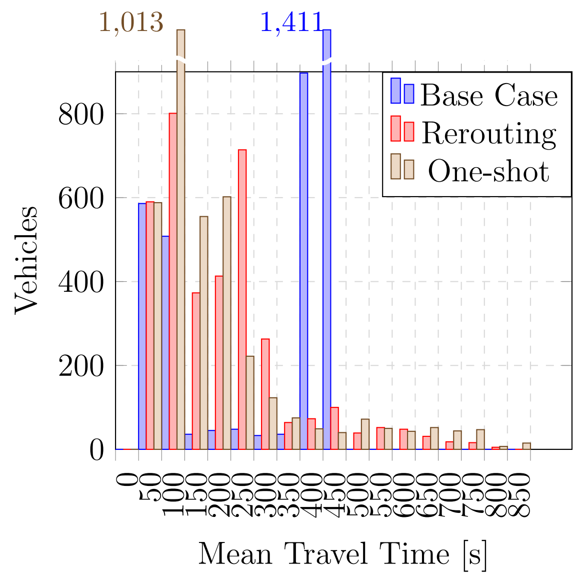


Figure 5.6: Travel time histogram

5.11 Conclusions

This chapter described a decentralized congestion avoidance strategy for connected vehicles. By correlating traffic on the wireless network to congestion in the transportation network, agents are able to make non-deterministic routing decisions which reduce congestion. This is done merely based on the rate of communications received by an agent, without the requirement to communicate its decisions to neighboring vehicles. The performance of the strategy is investigated using bi-directionally coupled vehicular and network simulators in a reduced scenario. We illustrated that compared to ideal shortest path routing, the proposed approach is much more efficient in increasing throughput and mean speed for the vehicles in the network, similar to using one-shot routing which requires significantly more information about the transportation network.

The current model offers a binary response to congestion: once the threshold is exceeded, the vehicle potentially reroutes, then waits until congestion has dropped below a limit before another reroute may be triggered. An alternative approach we are pursuing is a proportionate response to congestion, where, instead of rerouting upon exceeding the threshold, a chance of rerouting could be assigned depending on the amount of perceived congestion, evaluated either over time or as a function of distance. The higher the congestion, the more likely the vehicle is to choose to reroute. Consequently, the vehicle is also able to evaluate its path options differently: in highly congested areas, paths of higher cost should be more attractive than in low congestion.

Algorithm 2 Modified Yen's K shortest path algorithm

```
procedure MODIFIEDKSP(graph, source, sink, K, n) ▷ where n is the number of nodes on the path, starting from the source, to be considered
    A[0] ← Dijkstra(graph, source, sink)
    B ← []
    for k ← 0, K do
        for i ← 0, min(n, A[k − 1]) do                ▷ modification
            spurNode ← A[k − 1].node(i)
            rootPath ← A[k − 1].nodes(0, i)
            for all path p ∈ A do
                if rootPath == p.nodes(0, i) then
                    remove p.edge(i, i + 1) from graph
                end if
            end for
            for all node rootPathNode ∈ rootPath except spurNode do
                remove rootPathNode from graph
            end for
            spurPath ← Dijkstra(graph, spurNode, sink)
            totalPath ← rootPath + spurPath
            append totalPath to B
            restore edges to graph
            restore nodes ∈ rootPath to graph
            if B is empty then
                break
            end if
            sort B
            A[k] ← B[0]
            pop B
        end for
    end for
    return A
end procedure
```

Chapter 6

Modifications to congestion control algorithm

6.1 Abstract

Vehicular ad-hoc networks, or VANETs, are mobile networks formed by moving vehicles. Along with autonomous vehicle technology they are poised to revolutionize transportation by improving safety and decreasing travel time. This paper builds on a proposed decentralized congestion avoidance algorithm using only the rate of messages exchanged between vehicles as an indicator of congestion in the transportation network in order to dynamically reroute vehicles and reduce traffic jams. The algorithm only requires a high-level map of the road network and the rate of incoming messages disseminated by other V2V applications such as forward collision warning or intersection movement assist, and does not add any bandwidth of its own. Additionally, it can be employed by both autonomous and non-autonomous vehicles alike to improve traffic throughput and decrease travel times. In this chapter we explore a range of possible improvements to our congestion avoidance algorithm ranging from filtering of incoming messages to several alterations in route choice determination and decision making. All of these are evaluated using the SUMO traffic simulator and OMNet++ network simulator.

6.2 Introduction

Congestion is a widespread problem in US traffic networks [54], escalating from an estimated cost of \$37.5 billion in 2004 [55] to \$121 billion in 2015 [56]. While building additional transportation infrastructure like highways, streets, subways, and trains can alleviate congestion, it is a very costly solution. Intelligent use of the existing road infrastructure can decrease travel times, CO_2 emissions, fuel consumption, and driver frustration. This can be achieved, for example, by connecting vehicles using cheap short-range radios so that they share information about traffic status on nearby roads.

Vehicular ad-hoc networks, or VANETs, are a type of mobile ad-hoc networks specifically consisting of vehicles communicating with each other (vehicle-to-vehicle or V2V) or roadside units (RSUs, vehicle-to-infrastructure or V2I) [57]. These networks enable the development of applications ranging from active safety like left turn assist to improved traffic efficiency like the dissemination of traffic information and smart route planning [58]. However, the high mobility of nodes in the network, and therefore constantly changing topology, leads to challenges with regard to stability and reliability of the connections [59]. As a consequence, roadside units are often assumed as an integral part of these networks. However, RSUs may not be present in smaller cities or emerging regions.

For this reason, we have decided to eschew V2I and focus on creating a universally useful V2V application. Furthermore, we draw inspiration from certain insects that form interaction networks chiefly by counting physical encounters to achieve complex tasks such as navigation and foraging for food, area coverage and decentralized congestion avoidance. In analogous fashion, we explore the possibility that by measuring the amount of network traffic in the VANET, either as created by other applications such as file sharing or active prediction [62], or by beacon messaging [63], and using it to estimate the congestion in the transportation network, individual vehicles make routing decisions in a completely decentralized manner. Using non-deterministic route selection, we illustrate that it is possible to relieve congestion and balance the traffic load without requiring any additional information exchange between vehicles. In the decentralized approach we discuss in this paper the rate of interaction, not the information itself, is the deciding factor. While the performance of such a decentralized approach may not compare favorably with a centralized, cloud-based solution like Google Maps or WAZE, we note that the approach could function even in the absence of a data connection and we surmise that it may be able to respond to changing conditions with less latency. The low bandwidth requirement of our model and integration into other applications means that it can be realized using the IEEE 802.11p Wireless Access for Vehicular Environment (WAVE) protocol using Dedicated Short Range Communications (DSRC) [64].

The rest of the chapter is organized as follows: Section 6.2.1 describes the inspiration for the work further, and reviews options for practical congestion measurement in and traffic routing. Section 6.3 expands on adjacent literature presented in the previous chapter. Section 6.4 evaluates the model using a reduced scenario, while Section 6.5 discusses and evaluates improvements to the algorithm. Finally, Section 6.6 summarizes the paper.

6.2.1 Background

Measuring congestion Congestion, whether caused by extraordinary factors such as accidents or as a manifestation of everyday traffic in urban areas, increase fuel consumption, emissions, and travel times. Its identification and measurement can be realized in a multitude of ways.

The simplest method is to define congestion as traffic that is moving at a slower rate than permitted by the road [65]. Traffic intensity may also be measured by carefully deployed surveillance systems such as traffic cameras [66], forming sensor networks whose data can once again be analyzed by central servers. However, infrastructure such as this is costly to deploy and maintain, making it available only to the busiest of intersections in major cities. VANETs reduce the need for expensive sensor networks. Another method is to correlate the amount of congestion with the density of vehicles in a certain area [67]. We will adopt a version of the latter approach: by estimating the density of vehicles in the immediate vicinity using wireless network traffic between vehicles, we compute a *congestion parameter* that is then used as the basis of rerouting decisions by each vehicle. Ways this parameter can be derived include control channel (CCH) messages assuming DSRC/WAVE, other beacon messaging applications, or even applications which send information non-periodically. Adaptive rate control of basic safety messages (BSM) in WAVE networks based on channel load [80], for example, would reduce the amount of messages sent in congested networks, but the channel load measurement in itself could be used to help derive a useful indicator of congestion in the transportation network.

Routing In order to improve traffic flow and reduce travel times, route assignment can be implemented in a variety of ways. A few appraoches which route cars as they enter the network are listed below, however, none of them re-route the vehicle as it travels. The simplest and most inefficient form of routing discussed here is to find the shortest path from source to destination - *shortest path routing*. In a transportation network topology, consisting of major roads connecting areas of varying traffic demands, congestion is likely to occur using this approach. *One-shot routing*, by contrast, incrementally assigns vehicles to routes as they enter the network according to current travel times. The resulting trips do not represent a stochastic user-equilibrium, but drastically improve throughput and travel times in most networks over shortest path routing. We chose this method as a basis of comparison for our model due to its simplicity, though it does require route and travel time information for each vehicle in the network at the time of departure, while the vehicles in our model only possess limited up-to-date information about their immediate vicinity. *Dynamic User Assignment (DUA)* is an iterative process in which the simulation is run and then a number of vehicles are chosen to be assigned alternative routes. This is repeated for a number of steps and also does not guarantee an equilibrium state - and can not be applied to live scenarios due to its iterative nature - but can be useful in analyzing transportation networks. *Stochastic User Equilibrium (SUE) Assignment* routes vehicles by determining a number of K shortest paths to the destination and then weighing them according to a choice model, for example Logit, C-Logit, Probit, or modified Lohse Logit [68]. Our model functions similarly, except vehicles begin by computing the shortest path and only using the K shortest path algorithm and Logit model upon encountering a congested area as measured using the *congestion parameter*.

6.3 Related work

A study using beacon messaging by vehicles subject to accidents in order to re-route other vehicles was conducted by [69]. [70] outlines a similar approach to ours in which vehicles send beacons when approaching an intersection, asking for information in routes of interest in order to evaluate congestion and make an informed routing decision. CoTEC (Co-Operative Traffic congestion detECtion) is a traffic congestion quantification model using traffic density acquired via beacon messaging and the vehicle's speed and processed via fuzzy logic to detect congestion [71]. Once a vehicle detects congestion, it exchanges traffic estimation messages with nearby vehicles to further estimate the extent and duration thereof. SOTIS (Self-Organizing Traffic Information System) uses a decentralized V2V approach by disseminating periodic data packets between vehicles, which is then used to analyze the traffic situation by every individual agent [72]. The analysis is then re-distributed to surrounding vehicles. StreetSmart Traffic has vehicles disseminate information about roads on which they had to travel with slower speeds than anticipated [73]. These approaches function well, but require the distribution of information between vehicles which is specific to their application.

The core of the rerouting algorithm discussed herein has been laid out in Chapter 5. It is expanded upon in Section 6.5 of this chapter, and alter fundamental aspects of the model to measure their impact on overall performance.

6.4 Experiments

The proposed congestion avoidance approach was evaluated using the reduced traffic scenario shown in Figure 5.2. A east/west main road is divided by an intersection with traffic lights and surrounded by capillary roads which may be used by vehicles to avoid the congestion created by the intersection. While the main roads are two lanes wide, the capillary roads are one-way streets. To the north of the main road, the one-way streets run east to west, and vice versa to the south, to avoid left turns which could increase the variance encountered in the experiment. Using SUMO's randomTrips.py script, 3600 trips were created with a period of $1/sec$. Important to note is that if a vehicle cannot be inserted on an edge due to heavy traffic, it waits until insertion is possible, in effect extending the road segment as far as needed. Potential starting edges include the west- and east-most edges as well as the edges to the south and north of the intersection. Trips have been generated to be 10 times more likely to start on the east or west edges, and all trips terminate on one of those two edges. This creates heavy congestion which will lead to a traffic jam on both sides of the intersection. The traffic light favors east/west traffic with a 31 second long green period, while the north/south road is green for 6 seconds, and the length of the yellow light period is 4 seconds.

We compare our model with a base case in which all vehicles simply take the shortest route to their destination (see Section 6.2.1) without any adjustments during the trip. In the case of the reduced scenario, using shortest path routing, the vehicles only use the main east/west road and avoid the capillary roads around the intersection, while our model and its modifications will utilize the capillary roads to reduce congestion.

Vehicles are equipped with a 20mW short-range radio and communicate using WAVE with an empty frame body field. The payload may instead by used for other applications. Even though beacon periods of down to 0.1 seconds are supported by modern DSRC technology, beacon messaging with a period of 3 seconds is used. Vehicles retain each message for 3 seconds –same as the beacon period- and trigger a reroute when the message memory exceeds 20 messages ($T_C = 20$), unless noted otherwise. $T_R = 10$ was chosen for the reset threshold.

An example of the utilization of the capillary roads for a single experiment using the rerouting algorithm we propose is illustrated in Figure 5.3. Uneven utilization indicates that the algorithm's parameter may be tuned even more aggressively, as the majority of vehicles still use the main road. This is a result of the use of the Lohse Logit model, which always weighs the shortest path as the most desirable, which may not be the case in heavily congested scenarios.

Figure 5.4 presents the history of the mean travel speed for the vehicle vehicles under two routing algorithms (base case - shortest route, and congestion triggered rerouting proppposed here). It shows that the base case quickly reaches a bottleneck, while both our congestion avoidance model utilizes the capillary roads, increasing mean speeds and throughput. The mean speeds in the rerouting case decrease over time as the capillary network reaches capacity, and then sharply increase as the last of the vehicles leave the network. The regular pattern of the stop-and-go traffic is reflected in Figure 5.5, showing that most vehicles have a rather constant mean velocity ($3.79m/s$) in the base case simulation, but about 31.9% lower than our rerouting algorithm ($5.57m/s$). These and other performance statistics are summarized in Table 6.1. Meanwhile, speeds in the base case remain low, resulting in low throughput Since the mean distance traveled by each vehicle only increases by about 10%, we achieve an overall reduction in travel time averaging 33.4%. The reduction in CO_2 emissions of about 18.5% is attributable to vehicles on the capillary roads which are not subject to the stop and go traffic caused by the intersection.

Figure 5.6 shows the same regularity in the base case as Figure 5.5, this time with regard to travel time. While mean travel time is significantly reduced with our decentralized rerouting algorithm, the histogram shows a drawback to our current model: travel time for a small number of vehicles increases noticeably compared to the base case. We are currently evaluating the model in more scenarios to verify that this is a consistent result, but no matter how infrequent, highly negative experiences for a small group of drivers and passengers should be avoided. Generally, the modifications to the algorithm which will be discussed further down tend to increase the standard deviation of all metrics as a trade off of decreasing mean travel times.

6.5 Modifications

Table 6.1: Modification statistics

	Base Case	Rerouting	Randomization	No Shortest	Chance	Modified Lohse	Filtering
Mean Time (s)	299.6	224.52	243.14	223.41	211.16	406.30	214.13
Mean Speed (m/s)	3.79	5.57	5.14	6.58	7.21	3.85	5.86
Mean Distance (m)	1137.1	1250.1	1248.20	1463.02	1519.67	1563.83	1254.48
Mean Emissions (mg/s)	506.3	427.3	452.92	456.18	435.41	690.1	405.75

Table 6.2: Modification standard deviations

	Base Case	Rerouting	Randomization	No Shortest	Chance	Modified Lohse	Filtering
Mean Time (s)	170.60	143.39	158.26	219.54	204.32	353.45	164.86
Mean Speed (m/s)	5.31	5.86	5.72	5.69	5.65	5.10	5.94
Mean Distance (m)	360.11	329.28	328.21	272.95	228.43	271.48	332.15
Mean Emissions (mg/s)	255.40	208.16	227.42	261.21	236.99	400.058	223.43

6.5.1 Randomization

Our algorithm thus far assumes periodic communications to derive the congestion parameter. This assumption is made to simplify and lower the amount of variables in the model. In order to test the algorithm under non-periodic circumstances, we modified the period for sending messages to be random with a normal distribution using a standard deviation of 1 second, while the mean remained at 3 seconds. Introducing this change without additional modifications leads to an increase in mean travel times of about 8.3%, but still offers an improvement of 23.2%

compared not using the congestion avoidance algorithm at all. The corresponding metrics are summarized in tables 6.1 and 6.2. Given the binary nature of the base algorithm - reroute above the threshold, do nothing below - a small reduction in the threshold T_C for this modification should most likely result in similar performance to the basic rerouting algorithm.

6.5.2 Threshold

For most of the experiments, the threshold at which a car decides to reroute was set to receiving 20 beacons from different vehicles within a 3 second period. Tables 6.3 and 6.4 summarize the performance metrics when adjusting the rerouting threshold. Lowering the threshold leads to vehicles potentially rerouting under less congested conditions, which leads to negative individual experiences. Halving T_C from 20 to 10 led to minute improvements in the major metrics, with the expected increase in standard deviations. However halving the threshold again resulted in mean speeds even slower than that of the initial, high threshold, and worse performance, indicating that there exists an optimal setting somewhere around $T_C = 10$ for this specific scenario.

Table 6.3: Statistics summary for various rerouting thresholds

	$T_C = 20$	$T_C = 10$	$T_C = 5$
Mean Time (s)	224.52	214.13	230.94
Mean Speed (m/s)	5.57	5.86	5.41
Mean Distance (m)	1250.1	1254.48	1248.66
Mean Emissions (mg/s)	427.3	405.746	435.41

Table 6.4: Statistics summary for various rerouting threshold standard deviations

	$T_C = 20$	$T_C = 10$	$T_C = 5$
Mean Time (s)	143.39	150.061	144.14
Mean Speed (m/s)	5.86	5.33	5.85
Mean Distance (m)	329.28	270.81	328.28
Mean Emissions (mg/s)	208.16	223.18	208.27

6.5.3 Eliminating the shortest path

When a vehicle decides to reroute, it uses the modified Yen's Algorithm to find a set of short paths that quickly diverge from the shortest path, which the vehicle is currently using, but still includes said shortest path. Therefore, a vehicle which makes the decision to reroute may yet choose to remain on its current path, which is actually the most likely option using the Modified Lohse Logit model. Eliminating the shortest, current path as a viable option when deciding to reroute should, in theory, serve to increase the traffic on the congested stretch of road to a number of vehicles at or below the threshold. In the reduced traffic scenario we analyzed, the improvement in mean travel time when introducing this variation is minimal: even though mean speeds increase noticeably, a corresponding rise in mean travel distances negates any improvements in travel time while increasing emissions. See tables 6.1 and 6.2, column "No shortest".

6.5.4 Chance-based rerouting

To achieve a gradual, probabilistic response to congestion that parallels the insect inspiration [60], we implemented a simple chance-based rerouting trigger: the congestion threshold is lowered from 20 to 5, in order to still prevent rerouting in very light traffic. But for every message received above the threshold of 5 messages every 3 seconds, there is a 1% chance that the vehicle decides to reroute. As the density of messages increases with the density of vehicles, the chance of a vehicle deciding to reroute in a given time frame also increases. A single vehicle's probability to reroute will reach about 82% at a congestion parameter of $\kappa = 20$, the previously used threshold at which the rerouting is triggered.

This change to the algorithm to a gradual response to the perceived congestion experienced by vehicles shows an improvement in mean travel time of about 5.9%, and is the most beneficial single improvement upon the original algorithm discussed herein. However, the modification also increases standard deviations markedly, which indicates an increased risk of heavily negative singular experiences for some vehicles. See 6.1 and 6.2, column "Chance".

6.5.5 Modifying route selection

Once a vehicle decides to reroute and finds a set of suitable routes, they are weighed using the Modified Lohse Logit model, which assigns weights to the paths irrespective of the congestion the vehicle finds itself in. In other words, a slightly longer path has the same attractiveness in light and heavy congestion. However, during heavy congestion, longer paths should be weighted more highly than shorter ones, as they may be less congested, and less vehicles may choose them as alternatives. To test this, we modified the Lohse Logit model's dispersion parameter as follows:

$$\beta = \frac{5}{\kappa} * \frac{12}{1 + exp(0.7 - 0.015C_{min,ij})} \tag{6.1}$$

which diminishes the difference between the β of paths of different costs as the congestion parameter κ increases, lessening the impact of cost on route selection which should in turn promote the selection of longer paths as perceived congestion increases. The congestion threshold T_C was also lowered from 20 to 5, where the response is equal to the unmodified dispersion parameter. The results show an increase in mean travel times, even compared to only using shortest path routing and no congestion avoidance - the base case.

6.5.6 Message domain filtering

Thus far the content of the messages received by vehicles in this model has been treated as irrelevant, ensuring flexibility, privacy, and adaptability. However, messages used for certain applications may include information relevant to the congestion avoidance algorithm, such as position and direction of travel of other vehicles. This additional information may be used to filter out incoming messages sent from vehicles which do not have an immediate impact on the congestion experienced by the receiving vehicle, such as those originating from vehicles traveling in the opposite direction or on adjacent streets. To test this hypothesis we included the rough direction an agent is traveling in as a byte in the payload of the WAVE beacon. Upon the receipt of a message, those that lie outside of a 90° cone of the agent's current direction of travel are then discarded. Experiments show a slight improvement when using domain filtering without the increase in standard deviations exhibited by several of the previously mentioned strategies as seen in Table 6.1 and Table 6.2.

In addition to testing the domain filtering in the outlined scenario, in which traffic flow is relatively equal in both main directions, we modified the scenario to send 5 times as much traffic in one direction compared to the other, allowing us to analyze the impact of the filtering method on *light* traffic when exposed to heavy network traffic that has no bearing on the actual transportation network congestion experienced by the vehicles as they are originating from a large number of vehicles traveling in the opposite direction (*heavy* traffic). The results in Tables 6.5 and 6.6 show that under these circumstances, the unmodified rerouting algorithm actually performs slightly better than the message domain filtering modification, which performed better in the scenario with equal traffic flow in both directions. The rerouting trigger threshold T_C is reduced by half from 20 to 10 due to the assumption that, on average, half of all messages will be filtered results in a barely measurable increase in performance.

Table 6.5: Message filtering in uneven scenario

	Base Case	Rerouting	Filtering	Filtering w/ $T_C = 10$
Mean Time (s)	376.97	285.57	291.21	290.20
Mean Speed (m/s)	4.01	5.38	5.27	5.29
Mean Distance (m)	1511.69	1535.5	1531.47	1534.57
Mean Emissions (mg/s)	667.43	563.124	569.67	568.84

Table 6.6: Message filtering standard deviations in uneven scenario

	Base Case	Rerouting	Filtering	Filtering w/ $T_C = 10$
Mean Time (s)	150.06	111.47	111.32	112.05
Mean Speed (m/s)	5.33	5.82	5.80	5.81
Mean Distance (m)	270.81	266.31	267.46	268.16
Mean Emissions (mg/s)	223.17	182.81	183.223	183.89

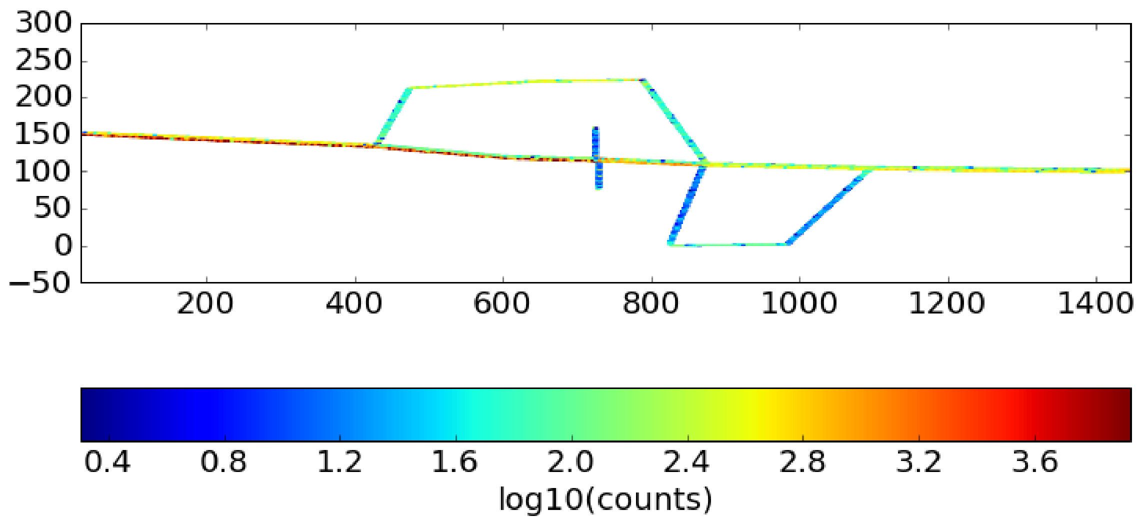


Figure 6.1: Heatmap of uneven scenario without filtering

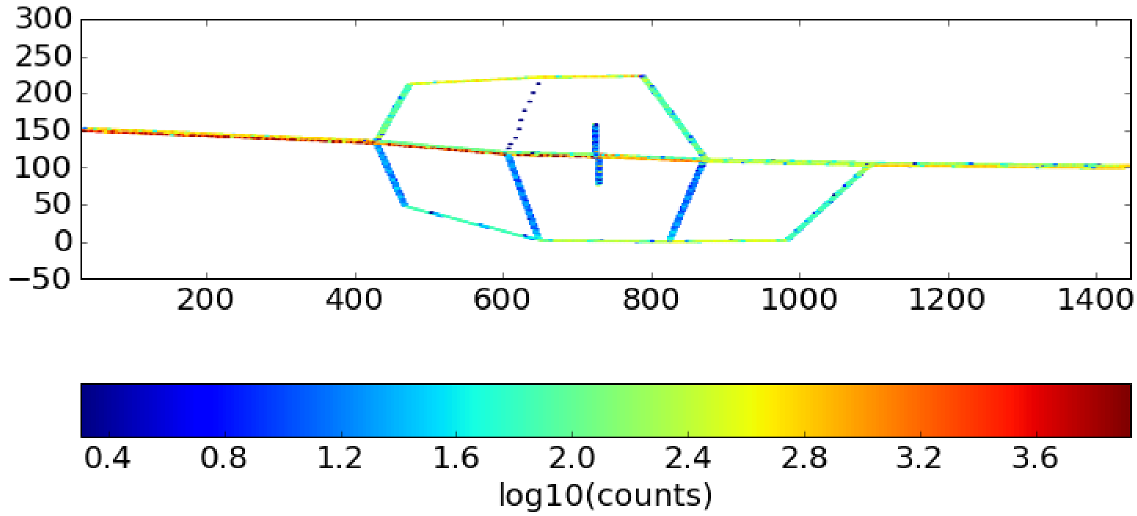


Figure 6.2: Heatmap of uneven scenario with filtering

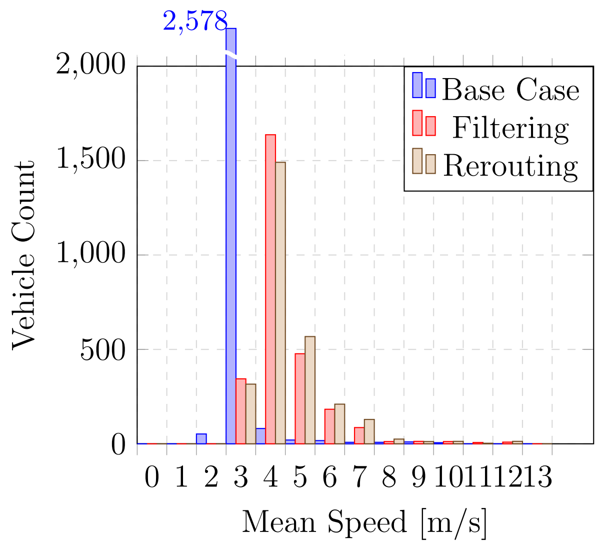


Figure 6.3: Mean speeds - heavy traffic

When dissecting the different approaches (filtering using $T_C = 10$ has been omitted due to a lack of variance between it and the standard $T_C = 20$) by direction, we note that the heavy traffic's mean speed is highly regular in the base case, illustrated in Figure 6.3. This is to be expected from steady stop-and-go traffic at an intersection. Using rerouting and extending it with message filtering increases mean speeds and also variance due to the increased route choices available. Interestingly, the light, unobstructed traffic that we hypothesized would be adversely affected by the messages originating from the heavy traffic in the opposite direction does not show nearly as much variance between approaches (Figures 6.4) as the lack of congestion results in consistently high mean speeds and the reduced scenario's compact nature does not allow for highly substandard route choices. The corresponding figures for travel time, Figures 6.5 and 6.6, show a reversal of the mean speed distributions, which reinforces the notion that the length of the chosen route in the reduced scenario has less of an impact as the amount of traffic it is experiencing. Given the lack of performance increase using filtering over the basic rerouting algorithm in this specific scenario, the violation of privacy by probing messages for information which may be used to filter them does not outweigh the loss of privacy thereby created. Figures 6.2 showcases the increase in route diversity using domain filtering compared to the standard algorithm in Figure 6.1.

6.6 Conclusions

This chapter described a decentralized congestion avoidance strategy for connected vehicles. By correlating traffic on the wireless network to congestion in the

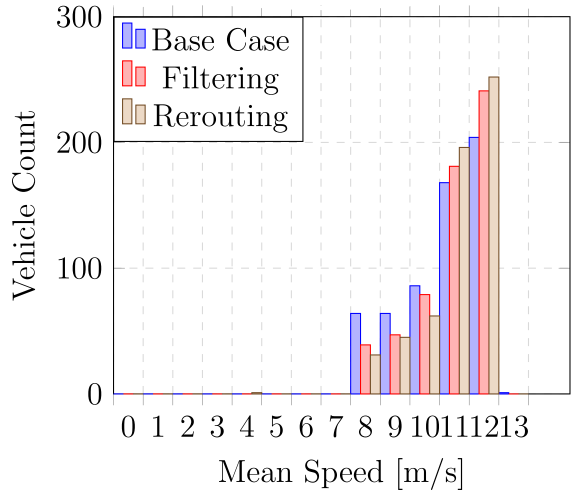


Figure 6.4: Mean speeds - light traffic

transportation network, individual vehicles are able to make non-deterministic routing decisions to reduce congestion. This is done passively, based on the rate of communications received by an agent, without any dissemination of aggregated information to neighboring vehicles. The performance of the strategy is investigated using bi-directionally coupled vehicular and network simulators in a reduced traffic scenario. We illustrated that compared to ideal shortest path routing, the proposed approach is decreases mean travel times in the network. Several promising modifications to the algorithm are also discussed and evaluated: while reducing the network congestion threshold at which rerouting occurs does not necessarily improve performance, modifying the route choices by eliminating the current path as an option or implementing a chance-based rerouting trigger so that likelihood of rerouting increases with network congestion improves overall performance at the expense of consistency and an increase in standard deviations. Using additional information to filter unnecessary messages is also evaluated positively, with the drawback that it violates the privacy afforded by the basic model. We also attempted to modify the attractiveness of alternate paths in relation to the amount of network congestion, however, no significant performance improvements were obtained in the test scenario.

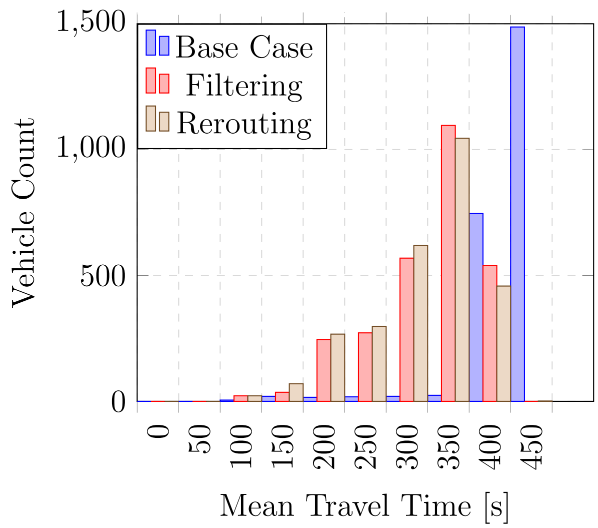


Figure 6.5: Uneven scenario - mean speeds in heavy traffic

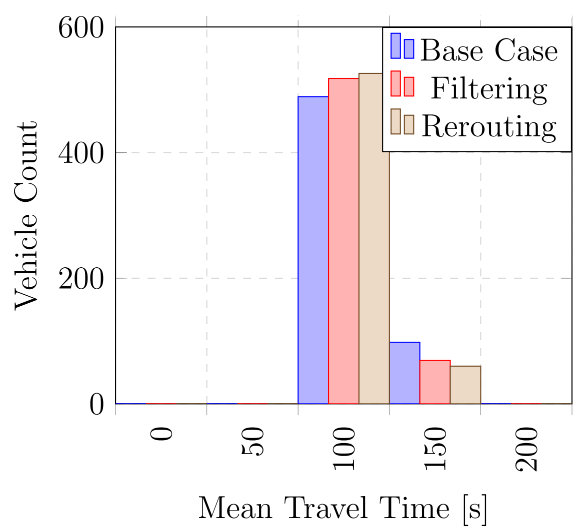


Figure 6.6: Uneven scenario - mean speeds in light traffic

Chapter 7

Conclusions

7.1 An ant-inspired model for multi-agent interaction networks without stigmergy

Chapters 2 and 4 outlined a non-deterministic multi-agent simulation model inspired by insect foraging behaviors, especially those of ants. Agents in the model rely on direct physical interactions with other agents and their environment to make decisions and do not rely on other mechanics such as pheromone trails. Collective emergent behaviors were tailored to favor congestion avoidance while still forming interaction chains to efficiently return food to the colony and preserve agent mobility. The effectiveness of the model is illustrated both in open and enclosed environments and under varying food source availability and placement. We have used the multi-agent interaction network to show that

- it is possible to create emergent, decentralized behaviors between simple agents using only minimal information - the rate of encounters between agents - without relying on infrastructure like pheromone trails or other modes of communication.

- using a small set of additional information, such as the location of the nest and a previously encountered food source, we can achieve additional behaviors such as targeted food retrieval with interaction chains.

- by using the encounter rate between agents we created a non-deterministic collision avoidance scheme using probabilistic estimation of congestion sectors over a range of encounters.

- the quorum sensing mechanism used by ants for nest site selection can be adapted to function as a decentralized congestion avoidance mechanism.

Future work

The multi-agent network built in Unity is self-contained and fully functional at this point. The values of several variables however have been determined experimentally. A thorough exploration of the variable space may be beneficial to gain a more intimate understanding of the interplay between variables, though the sheer number of experiments required to achieve sensible averages of metrics given the deep pool of variables and their possible combinations has made this a time-prohibitive effort thus far. Additionally, we have not explored the response of the network to changing environments, for example food sources which appear a certain time in the simulation has already elapsed. After some refactoring the application is able to be sped up considerably at the expense of immediate visual feedback and given a sufficiently powerful CPU. Several comments have been received indicating that it may be useful to implement pheromone trails as a way to provide a contrast to the model, however the main motivation of this work is to eschew stigmergy in favor of decentralized, direct interactions between agents.

7.2 Decentralized traffic rerouting using minimalist communications

The second half of the thesis examines a decentralized congestion avoidance strategy for connected vehicles. Several of its features are adapted from the ant-inspired multi-agent network:

- Direct interactions between agents are replaced by short-range beacon messages sent between vehicles and used to infer and estimate the congestion a vehicle is experiencing in the transportation network.

- The quorum sensing mechanism, used by the ant-like agents to trigger additional congestion avoidance behaviors, is used to trigger the congestion avoidance scheme for vehicles.

- The ant-like agents make heading decisions at each time step, which is unrealistic for vehicles traveling in a graph-like environment. Furthermore, limiting the number of reroutes positively impacts the experience of drivers in non-autonomous vehicles. Instead, once a reroute is triggered, the vehicle adjust its "heading" by examining a number of short paths which quickly diverge from the current area.

- Paths are chosen non-deterministically after having been assigned a chance of selection based on their cost. This mirrors how ants evaluate their heading choices using the von Mises distribution.

- Agents still only rely on a small amount of information: encounter rates and a map of its surroundings.

The model is analyzed using bi-directionally coupled vehicle and network simulators and found to improve mean travel times by over 25%, performing similarly to one shot routing but requiring far less information and no centralized infrastructure. Several modifications to the model are subsequently evaluated:

- Randomizing the beacon period to more closely equal real-life network conditions leads to small, acceptable losses in performance.
- Varying the encounter density threshold at which the congestion avoidance behavior triggers does impact performance, though not so drastically as to outweigh the algorithm's benefits over shortest path routing.
- Eliminating the shortest path as a possible rerouting decision promises minimal performance improvements at the cost of consistency.
- Assigning a set chance to trigger a reroute after each message beyond a low threshold greatly improves performance. A higher density of vehicles leads to more messages, and a higher chance of an individual vehicle choosing to reroute within a given time frame, naturally scaling the congestion response to the amount of congestion present in the transportation network.
- Evaluating paths differently depending on the amount of congestion present has only lead to performance losses given the method used.
- Filtering messages sent by vehicles which may not actively impact the congestion experienced by the recipient based on message contents may minimally improve performance at the cost of privacy.

Based on the performance of the model, we can conclude the following:

- Wireless network traffic can be used to indicate congestion in a transportation network of connected vehicles. The beaconing scheme used herein was simple in order to reduce overall complexity, but has been shown to be robust to varying messaging rates. Other indicators such as channel load can also be substituted for the method used in our model.
- Efficient congestion avoidance can be achieved
  - without the need for centralized infrastructure or roadside units.
  - with only limited information supplied by other vehicles in the immediate vicinity.
  - without the need to disseminate private and/or identifying information.

– without competing for bandwidth with other V2V and V2I applications.

This makes our model an attractive alternative in environments where cellular service or infrastructure is not available such as smaller towns, disaster zones, or simply as a backup solution. Additionally, users can remain anonymous as the model simply relies on the time at which a message is received and not its contents, and does not add any bandwidth of its own.

Future work

The congestion avoidance algorithm presented herein has mainly been evaluated in the reduced scenario described in Figure 5.2, providing a small, self-contained approach to tuning and evaluating the algorithm. Given the computer hardware available for the project, we were able to achieve a ratio of 1.4:1 to 3:1 real-time to simulation-time ratio. Most experiments analyzed herein had a simulation time of about 1.5 to 3 hours. Consequently, parameters and experiments needed to be chosen carefully to gather a compelling data set. Evaluating the model in real-life networks such as TAPAS Cologne [81] and LuST (Luxembourg SUMO Traffic [82]) were initially planned and several tests conducted, however the excessive real-time runtime per experiment, often one to three days, proved to be prohibitive for gathering a statistically significant sample size given the many inherently non-deterministic aspects of the model. As a compromise the model was also tested in a 40x40 grid network consisting of two-way road segments 400m in length. In this instance, creating a traffic flow that created sufficient congestion given the many route choices available proved to be problematic.

Bibliography

[1] J. Harding, G. Powell, R. Yoon, J. Fikentscher, C. Doyle, D. Sade, M. Lukuc, J. Simons, and J. Wang, "Vehicle-to-vehicle communications: Readiness of v2v technology for application," Tech. Rep., 2014.

[2] E. Uhlemann, "Introducing connected vehicles [connected vehicles]," *IEEE Vehicular Technology Magazine*, vol. 10, no. 1, pp. 23–31, 2015.

[3] D. Schrank, B. Eisele, T. Lomax, and J. Bak, "2015 urban mobility scorecard," 2015.

[4] J. Wright, J. K. Garrett, C. J. Hill, G. D. Krueger, J. H. Evans, S. Andrews, C. K. Wilson, R. Rajbhandari, and B. Burkhard, "National connected vehicle field infrastructure footprint analysis," Tech. Rep., 2014.

[5] C. W. Reynolds, "Flocks, herds and schools: A distributed behavioral model," *ACM SIGGRAPH computer graphics*, vol. 21, no. 4, pp. 25–34, 1987.

[6] G. Beni and J. Wang, *Swarm Intelligence in Cellular Robotic Systems*. Springer Berlin Heidelberg, 1993, pp. 703–712.

[7] M. Dorigo, "Optimization, learning and natural algorithms," *Ph. D. Thesis, Politecnico di Milano, Italy*, 1992.

[8] J. Kennedy and R. Eberhart, "Particle swarm optimization," in *Neural Networks, 1995. Proceedings., IEEE International Conference on*, vol. 4, Nov 1995, pp. 1942–1948 vol.4.

[9] R. Poli, "An analysis of publications on particle swarm optimisation applications," 2007.

[10] European Parliament, "Directive 2010/40/eu," 2010, http://eur-lex.europa.eu/LexUriServ/LexUriServ.do?uri=OJ:L:2010:207:0001:0013:EN:PDF.

[11] Y. J. Li, "An overview of the dsrc/wave technology," in *International Conference on Heterogeneous Networking for Quality, Reliability, Security and Robustness*. Springer, 2010, pp. 544–558.

[12] T. Stützle and M. Dorigo, "Aco algorithms for the traveling salesman problem," *Evolutionary Algorithms in Engineering and Computer Science*, pp. 163–183, 1999.

[13] G. Di Caro and M. Dorigo, "Antnet: Distributed stigmergetic control for communications networks," *Journal of Artificial Intelligence Research*, vol. 9, pp. 317–365, 1998.

[14] A. Shmygelska, R. Aguirre-Hernández, and H. H. Hoos, *Ant Algorithms: Third International Workshop, ANTS 2002 Brussels, Belgium, September 12–14, 2002 Proceedings*. Berlin, Heidelberg: Springer Berlin Heidelberg, 2002, ch. An Ant Colony Optimization Algorithm for the 2D HP Protein Folding Problem, pp. 40–52.

[15] E. Şahin, "Swarm robotics: From sources of inspiration to domains of application," in *International workshop on swarm robotics*. Springer, 2004, pp. 10–20.

[16] M. Dorigo, M. Birattari, and M. Brambilla, "Swarm robotics," *Scholarpedia*, vol. 9, no. 1, p. 1463, 2014, revision #138643.

[17] P. Leitão, J. Barbosa, and D. Trentesaux, "Bio-inspired multi-agent systems for reconfigurable manufacturing systems," *Engineering Applications of Artificial Intelligence*, vol. 25, no. 5, pp. 934–944, 2012.

[18] N. Foukia and S. Hassas, "Managing computer networks security through self-organization: a complex system perspective," *Engineering Self-Organising Systems*, vol. 2977, pp. 124–138, 2004.

[19] M. Collett, L. Chittka, and T. S. Collett, "Spatial memory in insect navigation," *Current Biology*, vol. 23, no. 17, pp. R789–R800, 2013.

[20] A. Koutsou and S. He, "Study of ants' traffic organisation under crowded conditions using individual-based modelling and evolutionary computation," *2009 IEEE Congress on Evolutionary Computation (CEC 2009)*, pp. 3330 – 7, 2009. [Online]. Available: http://dx.doi.org/10.1109/CEC.2009.4983367

[21] A. Wystrach, M. Mangan, and B. Webb, "Optimal cue integration in ants," in *Proc. R. Soc. B*, vol. 282. The Royal Society, 2015.

[22] S. C. Pratt, E. B. Mallon, D. J. Sumpter, and N. R. Franks, "Quorum sensing, recruitment, and collective decision-making during colony emigration by the ant leptothorax albipennis," *Behavioral Ecology and Sociobiology*, vol. 52, no. 2, pp. 117–127, 2002. [Online]. Available: http://dx.doi.org/10.1007/s00265-002-0487-x

[23] T. D. Seeley and P. K. Visscher, "Quorum sensing during nest-site selection by honeybee swarms," *Behavioral Ecology and Sociobiology*, vol. 56, no. 6, pp. 594–601, 2004.

[24] S. C. Pratt, "Quorum sensing by encounter rates in the ant temnothorax albipennis," *Behavioral Ecology*, vol. 16, no. 2, pp. 488–496, 2005.

[25] R. Foss, "A self-organising network model of decision making by the honey bee swarm," *Kybernetes*, vol. 45, no. 3, pp. 358–70, 2016.

[26] H. Withers, S. Swift, and P. Williams, "Quorum sensing as an integral component of gene regulatory networks in gram-negative bacteria," *Current Opinion in Microbiology*, vol. 4, no. 2, pp. 186 – 193, 2001.

[27] J. Hamar and R. Dove, "Quorum sensing in multi-agent systems," *INSIGHT*, vol. 15, no. 2, pp. 35–37, 2012.

[28] N. R. Franks, J. P. Stuttard, C. Doran, J. C. Esposito, M. C. Master, A. B. Sendova-Franks, N. Masuda, and N. F. Britton, "How ants use quorum sensing to estimate the average quality of a fluctuating resource," *Scientific reports*, vol. 5, 2015.

[29] A. Dussutour, V. Fourcassie, D. Helbing, and J.-L. Deneubourg, "Optimal traffic organization in ants under crowded conditions," *Nature*, vol. 428, no. 6978, pp. 70–73, 2004.

[30] I. D. Couzin and N. R. Franks, "Self-organized lane formation and optimized traffic flow in army ants," *Proceedings of the Royal Society of London B: Biological Sciences*, vol. 270, no. 1511, pp. 139–146, 2003.

[31] M. Cremer and J. Ludwig, "A fast simulation model for traffic flow on the basis of boolean operations," *Mathematics and Computers in Simulation*, vol. 28, no. 4, pp. 297–303, 1986.

[32] A. Schadschneider, A. Kirchner, and K. Nishinari, "From ant trails to pedestrian dynamics," *Applied Bionics and Biomechanics*, vol. 1, no. 1, pp. 11–19, 2003.

[33] D. A. Rosenblueth and C. Gershenson, "A model of city traffic based on elementary cellular automata," *Complex Systems*, vol. 19, no. 4, p. 305, 2011.

[34] D. Chaudhuri and A. Nagar, "Absence of jamming in ant trails: Feedback control of self-propulsion and noise," *Physical Review E 91, 012706*, 2015.

[35] D. Gordon, *Ant Encounters: Interaction Networks and Colony Behavior*, ser. Primers in Complex Systems. Princeton University Press, 2010. [Online]. Available: https://books.google.com/books?id=MabwdXLZ9YMC

[36] G. Greenfield, "On simulating seed foraging by red harvester ants," in *Artificial Life (ALIFE), 2013 IEEE Symposium on*. IEEE, 2013, pp. 105–112.

[37] S. Benhamou, "Detecting an orientation component in animal paths when the preferred direction is individual-dependent," *Ecology*, vol. 87, no. 2, pp. 518–528, 2006. [Online]. Available: http://dx.doi.org/10.1890/05-0495

[38] B. Prabhakar, K. N. Dektar, and D. M. Gordon, "Anternet: The regulation of harvester ant foraging and internet congestion control," in *Communication, Control, and Computing (Allerton), 2012 50th Annual Allerton Conference on*, 2012, pp. 1355–1359, iD: 1.

[39] S. Garnier, M. Combe, C. Jost, and G. Theraulaz, "Do ants need to estimate the geometrical properties of trail bifurcations to find an efficient route? a swarm robotics test bed," *PLoS*, vol. 9, no. 3, 2013.

[40] E. A. Codling, "Biased random walks in biology," Ph.D. dissertation, The University of Leeds, 2003.

[41] D. B. Siniff and C. Jessen, "A simulation model of animal movement patterns," *Advances in ecological research*, vol. 6, pp. 185–219, 1969.

[42] G. A. Fink, J. N. Haack, A. D. McKinnon, and E. W. Fulp, "Defense on the move: ant-based cyber defense," *IEEE Security & Privacy*, vol. 12, no. 2, pp. 36–43, 2014.

[43] S. Wohlgemuth, B. Ronacher, and R. Wehner, "Ant odometry in the third dimension," *Nature*, vol. 411, no. 6839, pp. 795–798, 2001.

[44] T. S. Collett and M. Collett, "Memory use in insect visual navigation," *Nature Reviews Neuroscience*, vol. 3, no. 7, pp. 542–552, 2002.

[45] K. Andreas, A. Beshah, and L. Chad, "Decentralized traffic rerouting using minimalist communications," in *IEEE International Symposium on Personal, Indoor and Mobile Radio Communications*. IEEE, 2017.

[46] M. Dorigo and T. Stützle, *Ant Colony Optimization*, ser. A Bradford book. BRADFORD BOOK, 2004. [Online]. Available: https://books.google.com/books?id=_aefcpY8GiEC

[47] X. Liu, P. Zhou, and R. Wang, "Small rna-mediated switch-like regulation in bacterial quorum sensing," *IET systems biology*, vol. 7, no. 5, pp. 182–187, 2013.

[48] A. Kasprzok, B. Ayalew, and C. Lau, "A microscopic model for multi-agent interaction networks without stigmergy," *Swarm Intelligence (under review)*, 2016.

[49] A. Downs, “The law of peak-hour expressway congestion,” *Traffic Quarterly*, vol. 16, no. 3, 1962.

[50] S. Floyd and V. Jacobson, “Random early detection gateways for congestion avoidance,” *IEEE/ACM Transactions on networking*, vol. 1, no. 4, pp. 397–413, 1993.

[51] Y.-P. Tian and C.-L. Liu, “Consensus of multi-agent systems with diverse input and communication delays,” *IEEE Transactions on Automatic Control*, vol. 53, no. 9, pp. 2122–2128, 2008.

[52] D. M. Gordon, R. E. Paul, and K. Thorpe, “What is the function of encounter patterns in ant colonies?” *Animal Behaviour*, vol. 45, no. 6, pp. 1083–1100, 1993.

[53] J.-L. Deneubourg and S. Goss, “Collective patterns and decision-making,” *Ethology Ecology & Evolution*, vol. 1, no. 4, pp. 295–311, 1989.

[54] D. Schrank, *Urban mobility report (2004)*. DIANE Publishing, 2008.

[55] C. Winston and A. Langer, “The effect of government highway spending on road users’ congestion costs,” *Journal of urban Economics*, vol. 60, no. 3, pp. 463–483, 2006.

[56] D. of Transportation DOT, “Beyond traffic: The blue paper,” Tech. Rep., feb 2015.

[57] K. Pandey, S. K. Raina, and R. S. Rao, “Hop count analysis of location aided multihop routing protocols for vanets,” in *2015 International Conference on Signal Processing, Computing and Control (ISPCC)*, Sept 2015, pp. 68–73.

[58] E. Schoch, F. Kargl, and M. Weber, “Communication patterns in vanets,” *IEEE Communications Magazine*, vol. 46, no. 11, pp. 119–125, November 2008.

[59] S. Y. Wang, “Predicting the lifetime of repairable unicast routing paths in vehicle-formed mobile ad hoc networks on highways,” in *2004 IEEE 15th International Symposium on Personal, Indoor and Mobile Radio Communications (IEEE Cat. No.04TH8754)*, vol. 4, Sept 2004, pp. 2815–2819 Vol.4.

[60] A. Kasprzok, B. Ayalew, and C. Lau, “A microscopic model for ant interaction networks without stigmergy,” *Swarm Intelligence (under review)*, 2016.

[61] A. Kasprzok and B. Ayalew, “Decentralized congestion control in random ant interaction networks,” in *The Eight International Conference on Swarm Intelligence (ICSI 2017)*, Fukuoka, Japan, July 2016.

[62] A. Broadhurst, S. Baker, and T. Kanade, “A prediction and planning framework for road safety analysis, obstacle avoidance and driver information,” 2004.

[63] H. Noori and B. B. Olyaei, "A novel study on beaconing for vanet-based vehicle to vehicle communication: Probability of beacon delivery in realistic large-scale urban area using 802.11p," in *2013 International Conference on Smart Communications in Network Technologies (SaCoNeT)*, vol. 01, June 2013, pp. 1–6.

[64] "Standard specification for telecommunications and information exchange between roadside and vehicle systems -5 ghz band dedicated short range communications (dsrc) medium accesscontrol (mac) and physical layer (phy) specifications," ASTM Int'l, Tech. Rep., July 2003.

[65] A. Downs, *Still Stuck in Traffic: Coping with Peak-Hour Traffic Congestion*. Brookings Institution Press, 2004.

[66] G. Palubinskas, F. Kurz, and P. Reinartz, "Detection of traffic congestion in optical remote sensing imagery," in *IGARSS 2008 - 2008 IEEE International Geoscience and Remote Sensing Symposium*, vol. 2, July 2008, pp. II–426–II–429.

[67] A. Tympakianaki, A. Spiliopoulou, A. Kouvelas, I. Papamichail, M. Papageorgiou, and Y. Wang, "Real-time merging traffic control for throughput maximization at motorway work zones," *Transportation Research Part C: Emerging Technologies*, vol. 44, pp. 242 – 252, 2014. [Online]. Available: http://www.sciencedirect.com/science/article/pii/S0968090X1400103X

[68] E. Cascetta, F. Russo, and A. Vitetta, "Stochastic user equilibrium assignment with explicit path enumeration: comparison of models and algorithms," vol. vol.3, Oxford, UK, 1997//, pp. 1031 – 7.

[69] S. T. Rakkesh, A. R. Weerasinghe, and R. A. C. Ranasinghe, "A decentralized vehicle re-routing approach using vehicular ad-hoc networks," in *2016 Sixteenth International Conference on Advances in ICT for Emerging Regions (ICTer)*, Sept 2016, pp. 201–207.

[70] M. T. Garip, M. E. Gursoy, P. Reiher, and M. Gerla, "Scalable reactive vehicle-to-vehicle congestion avoidance mechanism," in *2015 12th Annual IEEE Consumer Communications and Networking Conference (CCNC)*, Jan 2015, pp. 943–948.

[71] R. Bauza, J. Gozalvez, and J. Sanchez-Soriano, "Road traffic congestion detection through cooperative vehicle-to-vehicle communications," in *IEEE Local Computer Network Conference*, Oct 2010, pp. 606–612.

[72] L. Wischoff, A. Ebner, H. Rohling, M. Lott, and R. Halfmann, "Sotis - a self-organizing traffic information system," in *The 57th IEEE Semiannual Vehicular Technology Conference, 2003. VTC 2003-Spring.*, vol. 4, April 2003, pp. 2442–2446 vol.4.

[73] S. Dornbush and A. Joshi, "Streetsmart traffic: Discovering and disseminating automobile congestion using vanet's," in *2007 IEEE 65th Vehicular Technology Conference - VTC2007-Spring*, April 2007, pp. 11–15.

[74] J. Y. Yen, "Finding the k shortest loopless paths in a network," *management Science*, vol. 17, no. 11, pp. 712–716, 1971.

[75] D. Lohse, H. Teichert, B. Dugge, and G. Bachner, "Ermittlung von verkehrsströmen mit n-linearen gleichungssystemen–verkehrsnachfragemodellierung," *Schriftenreihe des Instituts für Verkehrsplanung und Straßenverkehr der TU Dresden*, vol. 5, 1997.

[76] D. Krajzewicz, J. Erdmann, M. Behrisch, and L. Bieker, "Recent development and applications of SUMO - Simulation of Urban MObility," *International Journal On Advances in Systems and Measurements*, vol. 5, no. 3&4, pp. 128–138, December 2012.

[77] A. Varga and R. Hornig, "An overview of the omnet++ simulation environment," in *Proceedings of the 1st international conference on Simulation tools and techniques for communications, networks and systems & workshops.* ICST (Institute for Computer Sciences, Social-Informatics and Telecommunications Engineering), 2008, p. 60.

[78] E. Butterworth, B. Jardine, G. Raymond, M. Neal, and J. Bassingthwaighte, "Jsim, an open-source modeling system for data analysis [version 3; referees: 2 approved]," *F1000Research*, vol. 2, no. 288, 2014.

[79] C. Sommer, R. German, and F. Dressler, "Bidirectionally Coupled Network and Road Traffic Simulation for Improved IVC Analysis," *IEEE Transactions on Mobile Computing*, vol. 10, no. 1, pp. 3–15, January 2011.

[80] B. Kim, I. Kang, and H. Kim, "Resolving the unfairness of distributed rate control in the ieee wave safety messaging," *IEEE Transactions on Vehicular Technology*, vol. 63, no. 5, pp. 2284–2297, Jun 2014.

[81] C. Varschen and P. Wagner, "Mikroskopische modellierung der personenverkehrsnachfrage auf basis von zeitverwendungstagebüchern," *Integrierte Mikro-Simulation von Raum-und Verkehrsentwicklung. Theorie, Konzepte, Modelle, Praxis*, vol. 81, pp. 63–69, 2006.

[82] L. Codeca, R. Frank, and T. Engel, "Luxembourg sumo traffic (lust) scenario: 24 hours of mobility for vehicular networking research," in *Vehicular Networking Conference (VNC), 2015 IEEE.* IEEE, 2015, pp. 1–8.