Alternate Representations for Scalable Analysis and Control of Heterogeneous Time Series

Francois Belletti
Alexandre Bayen, Ed.

Electrical Engineering and Computer Sciences
University of California at Berkeley

Technical Report No. UCB/EECS-2017-195
http://www2.eecs.berkeley.edu/Pubs/TechRpts/2017/EECS-2017-195.html

December 5, 2017

Alternate Representations for Scalable Analysis and Control of Heterogeneous Time Series

by

Francois Belletti

A dissertation submitted in partial satisfaction of the
requirements for the degree of
Doctor of Philosophy

in

Computer Sciences

in the

Graduate Division

of the

University of California, Berkeley

Committee in charge:

Professor Alexandre M. Bayen, Chair
Professor Joseph E. Gonzalez
Professor Scott Moura
Professor Laurent El Ghaoui

Fall 2017

Alternate Representations for Scalable Analysis and Control of Heterogeneous Time Series

Abstract

Alternate Representations for Scalable Analysis and Control of Heterogeneous Time Series

by

Francois Belletti

Doctor of Philosophy in Computer Sciences

University of California, Berkeley

Professor Alexandre M. Bayen, Chair

The recent increase in the availability of very large data sets has enabled major breakthroughs in Artificial Intelligence. Automated devices are now able to achieve a higher level of performance in computer vision, playing perfect and imperfect information games, and language processing which now compares to that of humans. Such progress is largely due to improved Single Instruction Multiple Data computing capabilities and higher bandwidth in distributed computing systems and innovative methods to leverage them.

A plethora of algorithms and theories developed in the field of Machine Learning enable better identification of system dynamics and extensive control of the corresponding systems. However, the vast majority of research focuses on problems dealing with homogenous observation data sets or control environment. Additionally, a large amount of work has focused on data sets comprising only of images, videos featuring similar sampling frequencies, and of time series with regular and identical timestamps of observation. Such a setting is not representative of the actual way data sets are collected and problems present themselves to practitioners. The present work delves into a more realistic setting where a unified representation of the data or control problem of interest is not available. We deal with a collection of heterogeneous sub-parts that relate one to another but do not naturally present themselves to practitioners in a homogenous fashion. Our main objective is to design methods that are readily applicable to heterogenous data sets and control problems in the distributed setting. The development of techniques that can be employed without additional pre-processing of the data makes them more practical to use by a broader range of individuals, companies and institutions.

We first focus on large multi-variate time series comprising of observations that are not synchronous in time. We then tackle Partial Differential Equations featuring a spatial continuum of distinct states. After this initial focus on the identification of dynamics in time series – otherwise referred to as time series analysis – we delve into the topic of control. Control leverages the knowledge we gather about a system in order to have it reach a particular desirable state. Part of the new work presented in the current work therefore focuses on the control of distributed cohorts of agents subject to individually unique constraints. Finally, we extend the work conducted separately

to heterogenous time series analysis and control and devise strategies beneficial for neural policies that identify system dynamics and optimal actions as part of the same module.

As we move from system identification to distributed control our aim is to find representations of the initial heterogenous problem that are homogenous and enable communication avoidance. In each of the chapters of the present work, we methodically design an alternate unifying and summarizing representation of the initial data set or control problem of interest.

Our motivation to find small, concise yet expressive representations is that they enable scalable computations in the distributed setting. When utilizing cohorts of computers inter-connected by a medium such as an ethernet or wireless network, communication time presents the main factor in the overall computing time. Decreasing the size of the messages that need to be transmitted therefore minimizes the overhead created by the need to communication information from a machine to another. With less time wasted in communication, we maximize the benefit of having more computational power and memory in the form of a distributed system.

We find by designing communication avoiding homogenous representations that statistical efficiency appears as a natural by-product. To that end, we leverage elements of stochastic process and point process theory, distributed spectral representations of linearized Partial Differential equations, convex duality and stochastic optimization, and finally specific regularization schemes in Reinforcement Learning of deep neural policies.

The present work features novel results on the estimation of cross-correlation of irregularly observed time series with event-driven sampling. A new analysis of the linearized Aw-Rascle-Zhang system of Partial Differential Equations is developed that unravels conditions for travelling waves to expand in the system. A comparative study of a dual splitting algorithm we developed for distributed control reveals new results that highlight how the messages being transmitted are more useful to the cohort of agents for control than for an adversary to eavesdrop on individuals. The regularization scheme we developed for neural control policies enabled extensive and robust control ability that compares with cutting edge parametric control strategies despite that no preliminary calibration is needed with our method.

The applications entailed in our numerical experiments span the fields of quantitative finance, macroscopic traffic modeling, Mobility-as-a-Service, electrical load balancing, and optimal ramp metering for freeways.

The alternate representations we develop are statistically efficient, scale naturally and are readily usable with collections of data-sets or controllers which may not rely on similar representational conventions.

To Shelley and my parents

Contents

List of Figures

List of Tables

Acknowledgments

My thanks go to Professor Alexandre M. Bayen and Professor Joseph E. Gonzales for their advice and guidance through the years. I would also like to thank Professor Ion Stoica, Professor Laurent El Ghaoui, Professor Scott Moura and Professor Michael Franklin for their helpful advice.

Chapter 1

Introduction

Driving vehicles has become one of the most prominent challenges mentioned in mainstream media about progress in Artificial Intelligence. It may seem surprising that such a task – which can be mastered easily by most humans – is still a challenge several decades after a supercomputer beat the worldwide champion of chess [200] – a game that is generally more difficult to master for humans. However, a key difference between playing chess and driving is that although the rules of chess are seemingly complex the perception of the state of the game is transparent and the turn by turn dynamics are fully known. In other words, the system corresponding to the rules underlying the game is simple to represent in a computer, determining the current state of the chessboard is immediate and predicting the next states is straightforward.

Identifying similar rules when it comes to driving vehicles is more complex. In fact, as one learns to drive, one solves two problems. The driver gets used to the behavior of vehicles and how actions on the commands – actuations – are translated into modifications of the state of the vehicle. Furthermore, with experience, the driver learns the variability of such reactions. Along with solving the problem of identifying how the vehicle reacts to actuation, the driver adapts its behavior in order to achieve the goal of driving from a point A to a point B as quickly and safely as possible while respecting traffic rules. In other words, another side of the driving problem consists in learning how to devise the actuation in order to change the state of the vehicle, use perception accurately to drive safely and achieve a particular aim such as commuting.

What makes the problem of learning how to drive difficult to automate is that learning how commands and perception relate to the state of the vehicle is hard. It is also difficult to learn how to actuate the commands finely to drive smoothly and safely. Finally, to complicate things even further, those two problems which are difficult to solve individually are tightly interdependent.

Driving is a simple and relatable example which is quite representative of general difficulties plaguing general purpose automation. As one designs an automaton – term we use to denote an automated device specifying its actions based on a policy translating perception of the environment into controlled actuations [200] – similar issues arise. As the automaton is created to actuate a dynamical system in order to achieve a particular goal two different problems emerge. A first problem consists in identifying the dynamics of the system to actuate while a second problem corresponds to devising optimal actuation strategies in order to achieve the goal. The former problem is generally referred to as system identification [144] while the latter constitutes control theory [31].

In order to better understand the dichotomy we expose, consider the example of automated driving. In this particular instance, the task of system identification corresponds to the evaluation of the mapping between the combined positions of the steering wheel, gearbox, acceleration and brake pedals to the state which corresponds to the position, speed, acceleration and orientation of the vehicle. Deciding how one should progressively accelerate and then perhaps brake in order to stabilize the speed of the vehicle corresponds to the design of a control policy. Similarly, defining by how much one should turn the steering wheel in order to change the orientation of the vehicle by a certain degree corresponds to a control decision specifying the actuation of the car.

In certain simple applications such as controlling an inverted pendulum, some understanding of Newtonian dynamics is sufficient to create a model of the dynamics determining how the system will react to inputs [31]. If one considers a car, a physical model could be used that only relies on the mechanical understanding of the vehicle in order map actuation inputs to state outputs. Unfortunately, even a single vehicle is already too complex. A physical model can hardly accurately account for how wear and tear affect dynamics or predict exactly how tires are going to adhere to the road. In such a case, whenever the dynamics of interest are too complex, data driven identification is employed [25].

State of the art

In order to introduce the challenges addressed in the present work, we first depict a historical perspective on the evolution of research concerned with system identification, progress in control theory and the evolution of the computing devices employed in actual implementations. As we follow the analytical and practical developments of the last decades along those three axis we are lead to consider that a paradigm shift could help better adapt the fields of system identification and control to the distributed heterogenous computing setting which has become the norm today.

Historical perspective on a novel age of pervasive data, large cyber-physical systems and expressive models

We now present key historical developments that relate the well theoretically understood fields of system identification and control to the changing computational landscape of the last few decades.

Progress in dynamical system identification

One may consider the first instances of time series analysis [28, 29, 99] – a subfield of system identification [144, 212, 115] – as the precursors of the field known today as Machine Learning [20, 77, 165] (ML).

The fundamental challenges underlying the estimation of coefficients of a parametrized model are still, to this day, a major problem in the field of Artificial Intelligence [200]. Indeed, as one attempts to build an automaton meant to maximize an expected reward, one needs to be able to understand which fundamental dynamics characterize the environment, how actions affect an eventual reward and how to leverage feedback. A major breakthrough in the field of Artificial Intelligence has been the formulation of optimality in a probabilistic sense and the modeling of environments in a stochastic setting [200]. In such a setting, ML [76, 165, 20] constitutes a powerful tool to estimate dynamics and the consequences of actions in the presence of uncertainty.

Machine Learning is a rather vague and un-intuitive term and therefore it sometimes seems preferable to use the denomination of Statistical Learning instead [77]. Such a name accounts more precisely for the continuity of research that started in statistics in the form of time series analysis [28, 29, 99, 25] and was later framed into sub-fields relating for instance to function approximation such as Gaussian Processes [191], Kernel Learning [43] or Neural Networks [20].

A key element in the progression towards modern Statistical Learning methods was a growth in the size of the data sets, computational power and the dimensionality of the problems under study [29, 76, 84]. More complicated models have progressively been considered and considerably extended the family of vanilla data driven approaches from simple linear predictive models [29] to heavily parametrized function approximators [84]. The idea at the core of data driven system identification has not changed. We let a statistical model learn a mapping between the inputs and outputs of the dynamical system under study. The presence of uncertainty is clearly taken into account in most models today [77]. Hybrid approaches such as state space models [60] help blend some prior knowledge of the system's dynamics with data driven parameter estimation in order to merge physical insights with data driven system identification.

A similar progression towards added statistical and computational complexity has equivalently been observed in the field of Control Theory.

Advances in control strategies

Linear models have also been the first concerns of Control Theoreticians although the field quickly shifted from basic linear systems [31] to non-linear dynamics [231, 122]. Non-linear system analysis presents an added level of complexity but inherently assumes in the first place that perfect knowledge of the dynamics of the system of interest is available. The state of the system itself is considered unknown and inferred on-line as feedback gets available [75, 166] but the model parameters are considered pre-determined.

Approaches that take uncertainty into account for control were developed more recently. Two families popular of approaches can be distinguished. In the first family, Robust Control Theory [250, 16], one assumes that a region of uncertainty exists around the estimated parameter values which accounts for the uncertainty affecting system identification. The second family, Stochastic Control [9], explicitly models uncertainty in the form of stochastic perturbations [9] whose impact is taken into account in the dynamics when it comes to achieving a particular objective.

Although such approaches are vastly successful they are hampered by additional conceptual and computational complexity which may not be rewarded in practical applications as some level of pre-existing knowledge of the system's dynamics is still expected. Non-stationarity of the systems dynamics may therefore lead both robust and stochastic control to fail. More modern approaches merge system identification and control into a single adaptive procedure and optimal behavior is learned by a trial and error process. Such is the premise of Reinforcement Learning (RL) [217] where an actuation policy is progressively improved as the controller acquired knowledge by interacting with the environment.

The absence of the need of prior system identification and the adaptive nature of the RL approach make it very appealing. Recently, it has lead to breakthrough in control problems concerned with RC helicopters [2] where random perturbations affecting the rotation of the rotor are extremely hard to model. Such parametric control policies can be made more complex although they are still learned by trial and error. Deep Q-learning has enabled [160] automata to play video games optimal based on images as input. Policy gradient methods have offered comparable results [69] in robotics although they rely on different adaptation methods. Deep Q-learning and policy gradient methods constitute a very novel field referred to as Deep Reinforcement Learning.

While modern paradigms involving Deep Reinforcement Learning [84, 217] conduct system identification and optimization of a control policy as part of the same model, they still implicitly rely on perception informing actions. Therefore, the same fundamental principle is at work today that was important five decades ago [87]: we need at some point – explicitly or implicitly – to characterize the dynamics of a system of interest in order to understand how our actions will affect it.

Whether such an understanding is granted by statistical methods providing strong guarantees [76] or black box neural models [84] depends on the solution we choose to address the problems of identifying a system and controlling it.

Evolution of computational power

One key element affecting the fields of time series analysis and control has changed over the course of the last five decades however: computational power [161]. Although we can no longer wait for 18 months and get a 2x increase of CPU computing capability, GPUs have computing capabilities that grow considerably each year, both in terms of memory and speed [188]. More importantly, such units of hardware become increasingly better at making data flow rapidly from its storage to where it can be made useful to improve models [121]. Finally, the rapid growth of ethernet network bandwidth [65] makes the interconnection between different machines less of a bottleneck and has enabled researchers to tackle problems such as exploratory data analytics [246] or neural network design [252, 49] with unprecedented computational capabilities.

Being able to leverage distributed networks of machines and the associated computational power has two important consequences for the analysis of systems and control.

First, a cohort of machines belonging to the same data center can read vast amounts of data to train models. As we have more empirical evidence to confront our models to, we can get more robust and accurate modeling abilities. We can also more confidently train complex models involving increasing amounts of parameters.

Second, distributed fleets of controllers are able to control vast systems of interconnected agents. We can now consider problems involving the electric grid in California [129] as a whole and not as separate disconnected sub-problems.

Today's distributed computing setting

Beyond the emergence of faster computational devices residing on a single chip, a major singular characteristic of today's system identification and control problems resides in their distributed heterogenous layout. While great opportunities are associated with the generalized use of distributed estimation and control systems, the challenges created by this new conceptual and computational setting are numerous. Namely, we are no longer computing a single core machine with a unified memory. Entire fleets of computing devices are now used collaboratively as they communicate with each other through a low bandwidth medium such as ethernet or wireless networks [221].

Choreographies of computations involving numerous different machines are far from easy to design and manage [221, 40, 124, 179, 137]. An overwhelming amount of different technologies are available today [209, 48, 223, 246, 74] to obtain the best answer to specific distributed computing need.

Complex computational patterns

The Paxos algorithm [126] is a well known example of an extremely complex algorithm design for the elementary task of electing a leader among a cohort of devices. The complexity of Paxos shows that, in the distributed setting, even a simple problem such as establishing a consensus among a cohort of machines communicating on an ethernet network leads to quite involved solutions in order to get eventual guarantees. One can then easily imagine the level of complexity researchers are exposed to as they design elaborate distributed machine learning pipelines [74] for data-driven system identification.

The complexity of the distributed computing systems asks for time and attention that make it harder to focus on the issue of formulating models with statistical guarantees. The Hogwild algorithm [192] is one famous example of an asynchronous program first discovered empirically as a good solution to compute Stochastic Gradient Descent in the distributed setting. Only later did [155] show that some statistical guarantees could be provided about Hogwild with an assumption of bounded delays in communications. The fact that most statistical guarantees were usually obtained in a setting involving algorithms operating on small data sets [29, 28, 99] has also made the reconciliation computing at scale with formulating well understood estimators quite challenging.

Adapting statistical methods designed for small data sets fitting on a unified memory

Indeed, when data sets comprising of 10^4 observations were the norm [99, 28, 29], the main issue under consideration was that of statistical efficiency. In other words, for a given small amount of data, the concern was to find the estimator that would discover a parameter value as close as possible to the actual one. Also, having been optimized for single core architectures, the corresponding algorithms were not designed to be deployed on cohorts of machines with communication bottlenecks such as an ethernet or wireless network.

Moreover, because memory was limited in size, it was fairly rare to consider jointly data streams sampled at different instants, the optimization of swarms of controllers obeying to different constraints or the collaboration of a cohort of robots. As services such as Amazon EC2 and frameworks such as Apache Spark [246] can make a cluster with several terabytes of RAM available in a matter of minutes, there is an opportunity to leverage a collection of heterogenous data sets jointly in order to make better decisions about the system they all give information about. Another extremely popular strategy consists in iterating fast on the data set under study and progressively fit a model by Stochastic Gradient Descent [170]. Using multiple machines to speed up gradient computation on successive sub-sets of the main data-sets makes such a method practically useful to estimate deep neural networks [49]. The opportunity of using more data to obtain better models and control strategies comes with a challenge though: that of finding a homogenous representation for heterogenous collections of data or control problems so that they can mutually learn and yield a better solution for the whole system.

The hidden assumption of homogeneity is no longer acceptable

The canonical representation of time series employed in classical system identification literature consists in a table of mono-variate or multi-variate measurements indexed regularly in time [29, 25, 99, 28]. However, as it was quickly pointed out in [177], event-driven sampling yields observations irregularly. Such irregular datasets that were inherently heterogenous – because they featured asynchronous measurements – used to be considered an intellectual curiosity. Unfortunately, such heterogeneity has become common today. It relates to two key heterogeneity factors.

As illustrated in Figure 1.1, system identification may face two major challenges when we consider the actual dataset it is based on in an asynchronous distributed sensor network. As larger memories are commonly available on distributed sensors, we can now afford to record timestamps with a high precision more easily. A direct benefit is that according to the Shannon sampling theorem [28] a broader range of frequencies can be accounted for in the digital version of the process of interest. However, a challenge stems from the irregularity and asynchronicity of timestamps across sensors. Each dataset entailed in an individual sensor is indexed differently in time from its counterparts located on other sensors. We are therefore facing conceptual heterogeneity between sub-datasets and therefore sub-problems.

The distribution across a low bandwidth network constituting a communication bottleneck can be considered as another source of heterogeneity. Each sub-dataset is located on a separate memory and each device may use different software and hardware. If the data is communicated as a whole across the network, the corresponding communication overhead will dominate over the time effectively spent computing [19, 221]. We refer to this issue as physical heterogeneity because it is related to the physical layout of the data in the distributed computational system we consider.

The main challenge we address in the present thesis is therefore to build methods that can directly solve heterogenous problems concerned with system identification and control while mitigating the

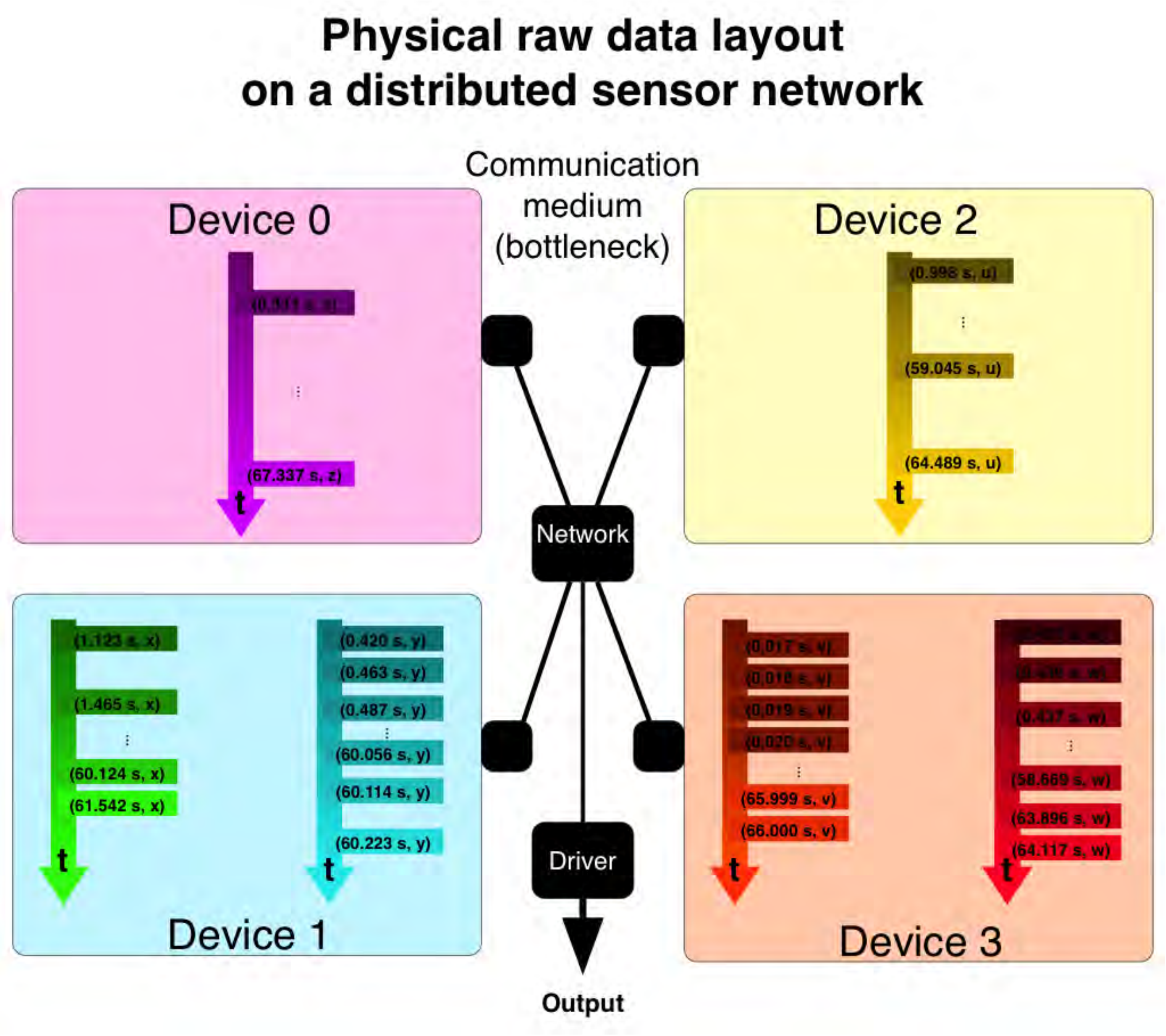


Figure 1.1: Example of irregular measurements in time that make system identification difficult. The asynchronicity of measurements across devices leads to conceptual heterogeneity. The scattering across devices connected by a low-bandwidth communication network leads to physical heterogeneity.

conceptual and computational difficulties arising in such a setting.

Alternate homogenous representations for heterogenous problems

In this thesis, we focus on formulating distributed analysis and control algorithm for settings in which the large scale of the problem under consideration results from the aggregation of heterogenous datasets, dynamics, controllers or systems that can help provide better results if they are considered jointly. Our aim is to develop representations that unravel the common denominator between different yet related data streams or controllers while preserving their idiosyncrasies and avoiding unnecessary communication.

Motivation and approach

While one technically feasible solution to speed up distributed machine learning pipelines consists in focusing on computational system aspects and organizing the hardware and software in order to maximize intercommunication bandwidth [187], such a solution is not readily applicable to a general setting of distributed controllers and sensors that may communicated wirelessly (therefore with low bandwidth). The cost of the hardware involved is high and intercommunication bandwidth may sometimes be subject to hard constraints. Wireless communication across networks of sensors is only one example of a setting where bandwidth is low and few messages should be sent in order to preserve energy [137].

In the present work, we put the burden of effort on the algorithmic side and want to find representations that enable solving heterogenous problems in the general setting of commodity hardware where inter-machine communication presents the main bottleneck.

At the core of our motivation resides the need for end-to-end methods that can directly take as input heterogenous data sets and solve the corresponding estimation or control problem in a scalable manner on commodity hardware.

Unified representations of heterogenous problems in the distributed setting

As we mentioned earlier we identify two sources of heterogeneity in the distributed setting as we consider collections of data sets and controllers as heterogenous problems.

Conceptual and physical aspects of alternate representations

Control problems can be considered as data sets if we reduce them to their objective, constraint set and entailed parameters. We are therefore dealing with distributed collections of data sets $(\mathcal{D}_i)_{i \in I}$. We are concerned with building a computationally tractable mapping $\mathcal{R}$ that is well defined on each $\mathcal{D}_i$ and has its outputs in a homogeneous representational set $\mathcal{H}$.

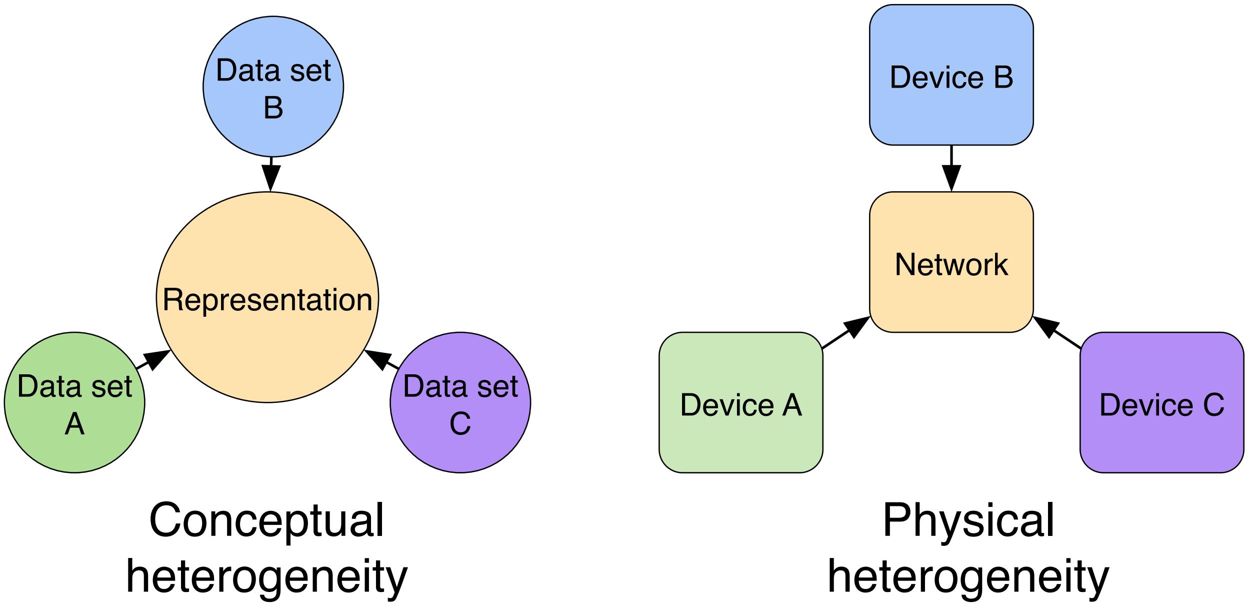


Figure 1.2: We identify two sources of heterogeneity as we consider system identification and control in the distributed setting: conceptual and physical heterogeneity. The problem the thesis poses is that of designing representations that are statistically sound and avoid large communication overhead. In other words, we want our representations to solve both conceptual and physical heterogeneity-related issues as part of their design.

Such a mapping, $\mathcal{R}$, will be referred to as a representation. As we will see in the following developments, naive representational strategies may present statistical issues as well as a large communication overhead. On an earlier example relating to system identification based on asynchronous distributed time series we already delineated two heterogeneity issues. Let us now generalize the two problems we identified. The first issue corresponds to what we identify as conceptional heterogeneity and relates to the diversity of data set domains. The second issue relates on the other hand to physical heterogeneity and the scatter of data across devices connected by a low bandwidth communication medium. We represent such issues in the modern setting of distributed problems in Figure 1.2. Our main challenge consists in designing representations that solve both difficulties at once. We want the representations we design to solve conceptual heterogeneity issues such as statistical artifacts. At the same time, we want our representation to create little communication overhead as it is computed in the distributed setting.

Statistical soundness

Whereas trying to achieve both the objective of scalability and unification of representations in a sound statistical manner may be challenging, it appears as definitively worthwhile attempting. A good understanding of the data generation process or the optimization problem being considered are the key elements leading to meaningful yet concise homogenous representations for homogenous problems.

In the present thesis, we leverage theory regarding second-order stationary time series and their spectral duality, translation invariant operators and their diagonalization in the Laplace domain, convex problems and their dual split equivalents in order to find the first alternate representations we plan to use. A large amount of the work is dedicated to reconciling statistical guarantees with scalable communication avoiding end-to-end data processing involving collections of different sub-problems.

The final chapter caveats the need for good theoretical understanding as, even in a model-free setting, complex perception and control policies can be found to have a natural unifying representation although their individual characteristics differ. No statistical guarantees are given then. However there is a clear advantage to using controllers that are trained with a realistic simulator and can adapt to changing conditions.

Communication avoidance

In the distributed setting communications present the main wall-time bottleneck. We are therefore looking for sets of short, homogenous and parsimonious messages that can uniformly inform us on a heterogenous collection of data. We will demonstrate in the thesis that for such a paradigm to be exploited, an alternate representation can be used thanks to a form of duality (e.g. spectral or convex duality).

The present thesis is in particular focused on the application of such a paradigm to linear stochastic system identification in the presence of irregular random sampling, the analysis of complex infinite-dimensional PDEs, the convergence of swarms of controllers to an optimal scheme and model-free adaptive and collaborative control.

We find that beyond the advantageous scalable properties it enables, communication avoidance forces us to find a homogenous and parsimonious representation that may provide a level of statistical efficiency.

It is indeed expected that if we find a KB-size summary in the context of a problem involving a TB-size data set, the summary we consider entails the very information we need. Noise, redundant and unnecessary information are not communicated. We therefore find in the alternate representations we propose a homogenous summary relating the heterogenous components of the problem of interest together, in a data efficient way.

Design strategies for homogenous communication avoiding representations

As will be shown in each of the following chapters, the applications of the corresponding contributions are many. We extend communication avoiding parameter estimation to the setting of irregularly observed distributed and asynchronous time series. Indeed, irregularly sampled time series are pervasive in stock market [3] and medical data sets [117]. Furthermore, still related to system identification, we use Laplace domain representations to make key properties of coupled PDEs obvious. Such infinite-dimensional PDEs are routinely employed to model the dynamics of freeways [141]. The role of communication avoidance is also key in our control related applications. We are first concerned with distributed control problems are of prime importance Mobility-as-a-Service and load balancing [129] and find that summaries can be designed to mitigate communication overhead without losing performance. Finally, we apply communication avoiding strategies to share experience between ramp metering controllers. Improving such collaborative control schemes is of key importance as they impact the commute of millions [175]. We now delineate the theories and methods we can build upon to better understand the challenges posed by heterogenous problems prior to devise strategies to better address them.

Strategy leveraging Fourier domain duality for second order stationary stochastic processes

The re-emergence of time series analytics triggered by the explosion of large data-sets that are timestamped is currently hindered by the omnipresence of irregularly sampled data in the form of news feeds [149], GPS measurements [105], random high frequency observations on the stock market [3]. Classic time series theory as presented in [29, 146, 99, 100, 25] has indeed mostly been concerned with the analysis of samples located on a regular time grid. Although a body of research focused on irregularly sampled data [177], and novel exploratory data analytics in the frequency domain have been developed [28, 145], no asymptotic theory has been formulated under broad assumptions in the setting of Long Range Dependent time series observed through an event-driven sampling scheme.

The key insight of the proposed strategy stems from a theoretical analysis in continuous time under assumptions that are well adapted to event-driven sampling. Considering correlated time series sampled on asynchronously observed channels leads to challenging problems in modeling, estimation and computation.

We quickly identify that interpolation methods may lead to bias in the estimation of correlation across time series. Such bias is an important issue as it will not be mitigated by leveraging more data in the distributed setting. In order to circumvent interpolation induced bias, we formulate a study in continuous time, which leads to strong guarantees for the kernel smoothed estimator we consider.

One challenge with a continuous time study though resides in its practical implementation. We leverage the duality between time and spectral domain representations for second order stationary time series [29, 47] in order to effectively compute our estimator in the frequency domain. Random Fourier projections are employed – after an appropriate probability distribution over frequencies is devised – in order to compute an estimator with a small vector which should be compared with the billions of samples we consider in practical datasets.

While each collection of observations corresponding to a univariate component has different timestamps, the frequency domain representation we compute is defined on a single set of frequencies. Therefore it appears in this work that the spectral representation unifies a distributed collection of heterogenous time series. The irregular asynchronous sampling appears here as the main source of heterogeneity across data-sets.

Strategy leveraging Laplace duality for linearized diagonalized partial differential equations

Considering time series analysis in continuous time is challenging in the second chapter because we have to deal with infinite-dimensional objects indexed by time. In the third chapter we study a dynamical system taking the form a system of coupled non-linear partial differential equations (PDEs) defined in continuous time and space.

Such complexity being difficult to handle, we linearize the system of interest and then employ a Laplace transform to unveil fundamental properties of the system. Spectral domain representations make the analysis of the system tractable and we quickly find conditions under which traveling waves may appear in the system. The spectral domain also makes the computation of numerical solutions particularly straightforward.

The resulting simplicity contrasts with the initial complexity of the problem. Also, the spectral representation makes the properties of a continuum of heterogeneous states evident as a distributed Laplace transform summarizes temporal information. The spectral representation of transition kernels can furthermore immediately be used for the purpose of controlling the linearized system.

Strategy leveraging Convex duality for distributed control problems

While the first chapters of the thesis deal with collections of heterogenous problems indexed continuously in time and/or space, the following chapters consider discrete collections of heterogenous control problems.

As operators of Mobility-as-a-Service fleets (e.g. Uber, Lyft etc.) try to match a demand with the aggregated presence of multiple drivers or as energy regulators try to balance electric load imbalance at the scale of California, they deal with control problems involving thousands of controllers [129].

While the controllers are all bound by the objective they are trying to achieve collaboratively, they are all subject to idiosyncratic constraints representing their individual availability, location, capacity etc. Such independent constraint sets constitute a heterogenous control problem which is usually solved by centralizing the data corresponding to all the constraints and solving the corresponding massive optimization problem. Serious computational issues arise then. Such a large problem cannot be solved by commodity hardware as it requires very large amounts of memory.

With the modeling assumptions we employ though, convex duality appears directly as an opportunity to formulate summaries of agents' individual constraint sets and share them in the form of small messages thereby enabling a communication-avoiding scheme to find an optimal strategy. The convex dual of the primal problem is therefore the alternate representation we employ here to give homogeneity to the many heterogenous sub-problems we consider at the individual controller level.

Moreover, in order to make the resulting collaborative scheme worthwhile joining for the participants, we analyze the advantageous properties of the algorithms we design in terms of privacy preservation. We find in particular that individual constraints do not need to be communicated. Such a property obtained by enforcing communication avoidance is particularly suitable for participants who do not want to disclose personal information.

Therefore, the analysis demonstrates that the alternate representation we employ in the form of a convex dual enables scalability with communication avoidance which also has important consequences in terms of the data efficiency of the resulting distributed control program. Convex duality gives an equivalent result to the primal optimization program we consider here because Slater's conditions hold. Unfortunately, in the non-convex setting of the last chapter, such strategy has no equivalence guarantees.

Strategy using models parameters as summaries

The fifth chapter is concerned with model-free control of freeways. We no longer assume that system identification has been conducted prior to establishing a control policy. In fact, both system identification and control optimality will be achieved as part of the same model. Optimal neural control policies are learned thanks to policy gradient methods. The controllers effectively learn by trial and error thanks to RL [217].

To be more precise, we consider a group of ramp metering controllers that regulate the flow entering a contiguous portion of freeway. We employ a state of the art simulator for the dynamics of the corresponding stretch of freeway in order to train our policies in conditions close to those of deployment.

Each of the ramp meters serves a particular entrance of the freeway with its particular demand in terms of incoming flow of vehicles. While such particularities make the control problem heterogenous, we want to enable some sharing of experience across controllers as they belong to a common environment and actuate parts of the same system following similar dynamics. We do not employ any form of explicit duality in this setting as the optimization objective we consider is non-convex and the dual solution would be sub-optimal.

The parameters used in the top-most layers of the neural control policies are shared across controllers and we demonstrate that such sharing of experience enables a clear improvement in control performance. Only a small number of parameters are being communicated between distributed controllers on the communication medium. This, in comparison to the amount of data controllers accumulate as they experiment by trial and error, leads to a communication avoiding scheme.

In the last chapter, model parameters of neural perception and control policies appear effectively as homogeneous representations of heterogenous problems. Such property can be interpreted as the consequence of having these small sets of parameters condensate information collected across vast samples of data about control feedback.

Organization of the thesis:

It is worthwhile mentioning that the complexity of the models and problems we consider grows increasingly. We start from linear models in continuous time under random sampling, consider linearized PDEs, study distributed convex programs and finally employ controllers taking the form of expressive neural policies.

To a certain extent, such a progression reflects the succession of trends in ML that have appeared popular in the last few years [76, 21, 165, 84]. A clear paradigm shift has occurred where computing a logistic regression at scale [246] has been superseded by massive Deep Learning (DL) pipeline [1] when it comes to showcasing the latest development of scalable methods.

More expressive models make the collection and use of increasingly massive datasets worthwhile as they enable the training of control policies having to make more and more complex decisions. We show in the upcoming chapters that as we follow such a progression towards complex models, data efficiency and communication avoidance still go hand-in-hand. The alternate representations we formulate enable us to take clear advantage of the resulting computational and statistical advantage in the analysis of the algorithms we formulate and their practical applications.

- **Chapter 2: Fourier domain representations for event-driven irregular sampling.** Most modern large data sets record events with high temporal resolution. As more data is available, practitioners may enjoy the opportunity to estimate larger and more robust multi-variate time series models. However, the event-driven nature of the sampling, the presence of Long Range Dependence and the scale of the datasets present unique challenges which we tackle in the first chapter. We demonstrate that interpolation is not a reliable solution to provide a homogeneous representation for the heterogenous collections of irregularly sampled time series. Interpolation related bias may lead researches to identify significant cross-correlation where there is none. We provide kernel smoothing based estimators and analyze how they do not feature such bias related issues.

  Statistical guarantees are given for second-order estimators in the very general setting of event-driven sampling. While the theoretical analysis occurs in continuous time – under random measures taking the form of Point-Processes – the practical computations occur in the frequency domain. Random projections are designed so as to enable kernel smoothing which provides strong statistical guarantees without falling into the pitfalls that hamper interpolation based methods.

  In our numerical experiments, we consider datasets of several GB in size. We are able to compute accurate cross-correlation estimators while only using KB sized messages. The resulting communication avoidance directly enables scalability for our estimation algorithms. Although our analytical study occurred in continuous time, we identify a clear advantage provided by frequency domain representations for computing.

- **Chapter 3: Spectral representations for control of PDEs.** In systems of PDEs, we consider an infinite continuum of states in continuous time. Each spatial location presents a unique state which leads to a continuity of heterogenous problems that need joint analysis.

  A Laplace domain spectral study of the linearized system of Aw-Rascle-Zhang PDEs provides an efficient and expressive representation of the problem. The resulting analysis of distributed Laplace transforms leads to a straightforward analysis delineating conditions for stability in the system we consider.

We characterize the precise situation under which convective instability pushes the state system away from its steady state. The corresponding traveling waves are important features of the system as they explain how crash may appear in the limit congestion regime for macroscopic freeway dynamics.

Although analysis occurs in the spectral domain, time domain solutions can be computed numerically which leads to predictions of traffic dynamics. Our numerical experiments demonstrate that although we study a linearized approximation of the system of interest, we still obtain an accurate modeling framework.

- **Chapter 4: Convex duality for distributed control.** We consider a problem setting involving agents subject to different heterogenous constraints. The agents need to collaborate as they are bound by a common performance objective at the level of the entire cyber-physical system they actuate.

  Convex duality is employed to formulate a dual split problem which is provably equivalent to the original primal problem but is parallelizable across controllers. The cornerstone of such a separation across agents is the formulation of a consensus Lagrangian multiplier. The multiplier can be interpreted as a price controllers agree on to synchronize their actions.

  The use of a parsimonious representer in the form of a small homogenous price vector presents multiple advantages. Agent-level constraints no longer need to be communicated which protects the privacy of the corresponding participants in the control scheme.

  In the applications we consider, such an desideratum is primordial. We instantiate the control scheme in the setting of Mobility-as-a-Service (MaaS) and electrical flattening in California. In the first problem, drivers should be incentivized to join the collective Mobility-as-a-Service collaborative scheme and protecting privacy is therefore an important aspect. The second problem considers using schedules for electrical vehicles. It is expected that the owners of the corresponding vehicles may want to keep information private.

  We furthermore prove that our scheme is robust, amenable to stochastic obfuscation to better protect privacy, and shares. More information is shared that is useful for the cohort of agent than elements revealing individual level data.

  The distributed communication avoiding scheme we obtain is fast enough to enable a receding horizon control scheme with several thousand vehicles on our Amazon EC2 based simulations. We prove that we can rely on the computational power of a swarm of smartphones although we use a fine temporal and spatial discretization scale.

- **Chapter 5: Distributed cooperative reinforcement learning to control PDEs.** The emergence of complex neural policies in Reinforcement Learning has recently led to great breakthroughs in Artificial Intelligence. The purpose of the last chapter is to leverage such complex models which constitute all-in-one solutions for both system identification and control. Our aim is to control complex discretized PDEs.

  We show that, in the absence of equivalence guarantees with a dual domain (spectral of convex), we are still able to find a common representer for heterogenous controllers. Distributed control of freeways is indeed a heterogenous problem as each controller is located on a ramp with an unique demand profile. We seek a unifying representation of control sub-problems to enable the controllers to share their experience as they learn by trial and errors.

In order to share experience across adaptive controllers while allowing for some specialization we design a regularization scheme leading to model parameter sharing. Model parameter values constitute the messages that are effectively being shared and thereby formulate a unifying representation.

We obtain cutting edge control ability while the neural controllers can dynamically adapt to changing conditions and perturbations such as accidents. Contingencies can be simulated and therefore incorporated in the learnings to the controllers we consider. We show that sharing model parameters reduces the gap between our RL neural control scheme and a state-of-the-art ALINEA model. The ALINEA baseline has perfect knowledge of the parameter values specifying the dynamics of the simulator we employ to train and test our controllers. On the other hand, the novel neural controllers learn the corresponding dynamics by trial and error.

Contributions

The work presented in the thesis has been published in the followed peer reviewed venues.

- **Random projection design for scalable implicit smoothing of randomly observed stochastic processes, AISTATS 2017 (Francois Belletti, Evan Sparks, Alexandre Bayen, Joseph E. Gonzalez):** Standard methods for multi-variate time series analysis are hampered by sampling at *random timestamps*, *long range dependencies*, and the *scale* of the data. In this article we present a novel estimator for cross-covariance of randomly observed time series which identifies the dynamics of an unobserved stochastic process. We analyze the statistical properties of our estimator *without the assumption that observation timestamps are independent from the process of interest* and show that our solution does not suffer from the corresponding issues affecting standard estimators for cross-covariance. We implement and evaluate our statistically sound and scalable approach in the distributed setting using Apache Spark and demonstrate its ability to identify interactions between processes on simulations and financial data with tens of millions of samples.

- **Prediction of traffic convective instability with spectral analysis of the Aw–Rascle–Zhang model, Physics Letters A 379 (38), pp 2319-2330 (Francois Belletti, Mandy Huo, Xavier Litrico, Alexandre M. Bayen):** This article starts from the classical Aw–Rascle–Zhang (ARZ) model for freeway traffic and develops a spectral analysis of its linearized version. A counterpart to the Froude number in hydrodynamics is defined that enables a classification of the nature of vehicle traffic flow using the explicit solution resulting from the analysis. We prove that our linearization about an equilibrium is stable for congested regimes and unstable otherwise. NGSIM data for congested traffic trajectories is used so as to confront the linearized model's predictions to actual macroscopic behavior of traffic. The model is shown to achieve good accuracy for speed and flow. In particular, it accounts for the advection of oscillations on boundaries into the interior domain where the PDE under study is solved.

- **Privacy-preserving dual splitting distributed optimization with application to load flattening in california, CDC (Francois Belletti, Caroline Le Floch, Scott Moura, Alexandre M. Bayen):** This article presents a dual splitting technique for a class of strongly convex optimization problems whose constraints are block-wise independent. The average-based input in the objective is the only binding element. A dual splitting strategy enables the design of distributed and privacy preserving algorithms. Theoretical convergence bounds and numerical experiments show that this method successfully applies to the problem of charging electric devices so as to even out the daily energy demand in California. The solution we provide is a privacy-enforced algorithm readily implementable in a network of smart electric vehicle chargers. It can reach any arbitrary precision for the common optimization goal while relying on randomly perturbed information at the agent level. We show that, provided the community is large enough, an averaging effect enables the group to learn its global optimum faster than individual information is leaked. A limited number of messages are sent out in the distributed implementation which prevents adversary statisticians from having low theoretical Mean Square Errors for their estimates. 3

- **Privacy-preserving MaaS fleet management, ISTTT and Transportation Research Procedia 23, 1000-1019 (Francois Belletti, Alexandre M. Bayen):** On-demand traffic fleet optimization requires operating Mobility as a Service (MaaS) companies such as Uber, Lyft to locally match the offer of available vehicles with their expected number of requests referred to as demand (as well as to take into account other constraints such as driver's schedules and preferences). In the present article, we show that this problem can be encoded into a Constrained Integer Quadratic Program (CIQP) with block independent constraints that can then be relaxed in the form of a convex optimization program. We leverage this particular structure to yield a scalable distributed optimization algorithm corresponding to computing a gradient ascent in a dual space. This new framework does not require the drivers to share their availabilities with the operating company (as opposed to standard practice in today's mobility as a service companies). The resulting parallel algorithm can run on a distributed smartphone based platform.

- **Expert Level control of Ramp Metering based on Multi-task Deep Reinforcement Learning, IEEE-ITS (Francois Belletti, Daniel Haziza, Gabriel Gomes, Alexandre M. Bayen:** This article shows how the recent breakthroughs in Reinforcement Learning (RL) that have enabled robots to learn to play arcade video games, walk or assemble colored bricks, can be used to perform other tasks that are currently at the core of engineering cyberphysical systems. We present the first use of RL for the control of systems modeled by discretized non-linear Partial Differential Equations (PDEs) and devise a novel algorithm to use non-parametric control techniques for large multi-agent systems. We show how neural network based RL enables the control of discretized PDEs whose parameters are unknown, random, and time-varying. We introduce an algorithm of Mutual Weight Regularization (MWR) which alleviates the curse of dimensionality of multi-agent control schemes by sharing experience between agents while giving each agent the opportunity to specialize its action policy so as to tailor it to the local parameters of the part of the system it is located in.

Chapter 2

Linear models in continuous time

Discrete time representation has been dominant historically [29, 25, 99, 229] in time series analysis. Unfortunately, it becomes limited as one tries to build reliable estimation procedures able to deal efficiently with irregular observations recorded on asynchronous sensing channels.

In the absence of a common temporal index available across sensing channels, we therefore study a heterogenous collection of time series. In such a setting, one standard procedure employed to give homogeneity to the timestamped collections consists in interpolating observations on a common time grid. We show in this chapter how such a method leads to concerning statistical issues and spurious estimators for second order properties of multi-variate time series. Having identified bias in interpolation, we seek a different homogenous representation of the heterogenous data streams of interest.

Some literature has been developed in order to provide methods able to estimate models in the irregularly observed setting [177, 145, 28]. Even recent publications in the field of Recurrent Neural Networks [169] present their ability to deal with asynchronous multi-channel sampling as a key advantage of their architecture.

Such studies are primarily concerned with the issue of irregular sampling. Unfortunately, another issue plagues time series analysis: Long Range Dependencies. Most methods designed to deal with Long Range Dependent (LRD) stochastic processes are unfortunately only considered in the discrete time domain [56, 186]. The case of LRD irregularly sampled time series is not clearly considered. Being able to capture LRD components or erase them by appropriate filtering is of paramount importance as otherwise the naive estimators we compute loose their convergence properties [87, 186] and may no longer be trusted. A major challenge is therefore to erase LRD components of a process in the irregularly sampled setting.

Methods focusing on the issue of irregular sampling also do not attempt to produce scalable algorithm. Computational issues arise quickly in standard methods such as the Lombge-Scargle periodogram [145]. Lack of scalability hampers the application of the corresponding methods as modern datasets generally comprise of billions of observations [246].

Finally, we argue that the scope of the theoretical properties revealed in [177] is too limited. A key limiting assumption is that the generation of observations occurs independently from the generation of the value of the process being observed. A core motivation of the chapter is to broaden the field of application of the elegant and extremely insightful study of second order stationary marked point processes presented in [177] and no longer require such limiting assumptions. Our derivations are therefore more general than the proofs in [177] which have been given with the critical assumption that the random timestamps at which observations are available are generated independently from the underlying process of interest. In other words, the properties of the analysis in [177] do not apply to the very general setting of event driven sampling.

Irregularity in many time series datasets available today stems from the fact that sampling is event driven. Such is the case of stock market data where order book information updates are made available when the order book is changed [133]. Recent smart cameras only relay information when a particular event occurs [241]. Geographic triggers such as entering a certain zone also make geo-location time series irregular in an event-driven manner [193]. These are only a few of many settings in which observations are irregular in time and occur in an event-driven manner.

The main objective of the current chapter is therefore to develop a novel and more general analysis of second order estimation for irregularly sampled time series. We want to be able to soundly consider cases where observations are triggered by events related to the process of interest. In order to achieve this goal, we first build a data generation model that leads to the formation of time series data set taking the form of temporally indexed collections (t_i, X_{t_i}). We employ Hawkes processes [47, 101] as a model for the generation of the timestamps (t_i). We further consider the Poisson intensity of the Hawkes process may be correlated with the process of interest (X_t) which we define in continuous time. The correlation accounts for even-driven sampling.

As observation timestamps become themselves a time series, we employ elements of random measure theory [47] carefully use filtrations and conditional expectations in order to give guarantees on the second order estimators we consider. Employing continuous time representations is a crucial point in the analysis, which contrasts with the spectral theory employed for example in [177]. A key consequence is the ability to deal with a event-driven observations seamlessly.

While the analysis of the convergence of the estimator for event-driven irregularly observed time series is conducted in the time domain, computations practically occur in the frequency domain. The dual representation is key to compute on continuous time processes with a parsimonious representer in a Fourier basis comprising of a few thousand floats. Employing a frequency domain representation also makes Long Range Dependence erasure possible with linear time complexity with respect to the number of observations which clearly improves on the standard time domain fractional differentiation method [186]. The latter becomes quickly hampered by its quadratic time complexity with respect to the number of observations available. The standard differentiation operator is also defined in the regularly sampled setting making its application to irregularly sampled time series difficult. To relate the properties of the frequency domain computing representation to the time domain estimator of interest we use random Fourier transforms. The only operation needed to transform the primal time domain representation into a dual frequency domain Fourier decomposition is orthonormal in the space of square integrable functions which preserves the geometric relationships between input signals.

Having highlighted the bias induced by interpolation methods leads, we compare the estimators we formulate with standard interpolation-free estimators specialized for the setting of irregularly sampled Brownian motions [102]. We demonstrate empirically that although our estimator is general purpose, it still performs as well as a specialized estimator such as Hayashi-Yoshida [102]. Our estimator can readily deal with other Long Range Dependent processes such as fractional Brownian motions, which we show is not the case of the Hayashi-Yoshida estimator for cross-correlation.

Organization of the chapter

The present chapter is therefore organized as follows. An introduction to Hawkes Processes is offered to the reader in order to be introduced to elements of the theory of random measures and the underlying modeling insights. Such stochastic point processes are a key component that enables the generality of our analysis.

After this introduction, novel properties of kernel smoothing applied to event-driven irregularly observed continuous time stochastic processes are derived. The chapter presents an appropriate data generation model for event-driven sampling. A core theorem then delineates the second order properties of kernel smoothing applied in the setting of random measures which does not need any assumption of independence between the observation process and the process of interest.

Once continuous time domain properties have been fully exposed, we show alternate representations in the frequency domain make the computation of the corresponding estimators tractable. We also prove that erasing Long Range Dependencies becomes trivial and computationally efficient in the corresponding dual representation space.

A numerical study on both surrogate and actual data shows that our estimator effectively converges and that the communication avoiding computations we can leverage in the frequency domain naturally lead to a scalable algorithm.

2.1 Problem setting and contributions

Cross-covariance estimates are of prime importance in applications ranging from statistical finance [3, 229] to climate studies [163] as asymmetry in cross-covariance is an indicator of a causal relationships between time series [29, 110, 86]. A pair of causally related continuous stochastic processes may be modeled as the solution of a Stochastic Differential Equation (SDE) of the form:

$$\begin{aligned} dX(t) &= dW^X(t) + \int_{s>0} \phi^{YX}(s)dY(t-s) + \phi^{XX}(s)dX(t-s) \\ dY(t) &= dW^Y(t) + \int_{s>0} \phi^{XY}(s)dX(t-s) + \phi^{YY}(s)dY(t-s), \end{aligned} \tag{2.1}$$

where W is a Wiener process [55]. The multidimensional causal kernel $\Phi(\cdot)$ consisting of

$$\phi_{XY}(\cdot), \phi_{XX}(\cdot), \phi_{YX}(\cdot), \phi_{YY}(\cdot)$$

defines [29] the dynamics and the solution to the SDE is given by the stable bi-variate stochastic process, $U(\cdot) = (X(\cdot), Y(\cdot))$. In practice, we observe stochastic processes $U(\cdot)$ at particular points

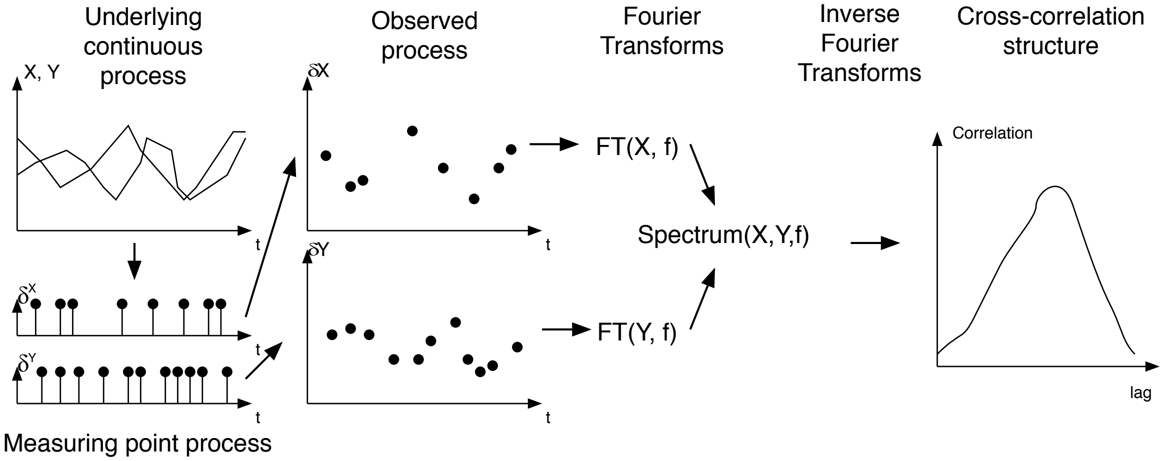


Figure 2.1: Underlying, measuring and observed process.

in time that are determined by a separate measurement process. In this work, we will assume that observations are made on the interval $[0, T]$ with timestamps generated by a bi-variate regular second order stationary Point Process

$$N(\cdot) = (N^X(\cdot), N^Y(\cdot))$$

as defined in [47, 42] referred to as the *measuring process* where $N^X(t)$ and $N^Y(t)$ are the number of observations that have been observed between 0 and time t. We assume that the *measuring process* has a stationary cross-covariance structure, that observations are not accumulated at a singular point in time, and their occurrence may be correlated with the *underlying process*. For example, in the medical setting, the *underlying process* could be the blood pressure of a patient which is defined at all times but only observed during punctual medical which the *measuring process* consists in here. The resulting *observed process*

$$(U\,dN)(\cdot) = ((X\,dN^X)(\cdot), (Y\,dN^Y)(\cdot)) \tag{2.2}$$

conflates the statistical properties of both the underlying process of interest (U) with the measurement process (N). Our aim in this chapter is to infer the statistical properties of the continuous underlying process (U) through the partial observations $(U\,dN)$ determined by the discrete measurement process (N). This data generative process is illustrated in Figure 2.1.

We make three contributions in this work. In Section 2.2, we empirically demonstrate standard cross-covariance estimators designed for regularly observed data almost surely evaluate to 0 resulting in biased estimates of the true correlation structure. We demonstrate that standard solutions to mitigate the resulting bias such as ad-hoc interpolation and Hayashi-Yoshida estimation [102] do not offer unbiased solutions for randomly observed stochastic processes. In Section 2.3 we develop a statistically sound approach based on the use of kernel smoothing that addresses the bias of classical estimators and provide a theoretical analysis demonstrating a bias/variance trade-off depending on the smoothing kernel bandwidth. We show how to compute these estimators efficiently in a scalable manner through random Fourier projections drawn from a specific probability distribution. In Section 2.4 we evaluate the proposed approach on both synthetic and real-world data characterizing the accuracy of our technique by achieving a relative error of 8% or less in linear model recovery with 90% probability. Moreover, our method enables the estimation of the parameters of the generative SDE model for the *underlying process* and using our new method we are able to accurately scale to large financial time series and estimate causal structure.

2.2 Limitations of Standard Methods

In the following, we identify a series of challenges in estimating the cross-covariance structure of a continuous bi-variate *underlying* process observed randomly.

Stochastic Intensity Point Processes

In order to more precisely state the probabilistic properties of the *underlying* and *measuring process*, we introduce useful notation.

The information entailed in the history of the process (X, Y) in the generative model Eq. (2.1) is defined as the filtration [55, 47] $\left(\mathcal{F}_t^{XY}\right)_{t\in[0,T]}$ where $\mathcal{F}_t^{XY}$ is the information available about (X, Y) up to but not including time t. One can also extend the past information to that jointly produced by both the *underlying* process and the *measuring* process which we denote $(\mathcal{F})$.

A stochastic Point Process (N^X, N^Y) [47] defines a random measure over the axis of time and is defined by stochastic local Poisson intensities for N^X

$$\mu_X(t) = \lim_{dt\to 0+} \frac{E\left[N^X(t+dt) - N^X(t)|\mathcal{F}_t\right]}{dt}$$

and similarly for N^Y. In other words, $\mu_X(t)$ and $\mu_Y(t)$ are the number of observations per unit of time, expected given the events that occurred until time t. With the second order stationarity assumption we make, these random measures are characterized by their cross-covariation structure

$$(\gamma^{\mu_X,\mu_Y}(h) = E\left[\mu_X(t)\mu(t+h)\right])_{h\in\mathbb{R}}.$$

The regularity assumption on the other hand guarantees [42] that measurements cannot accumulate at a single unique timestamp. The following proposition is helpful to understand the role of such a measure of continuous stochastic processes and will be used in proofs below:

Proposition 2.2.1 *With a continuous underlying process (X), almost surely,*

$$E\left[X(t)dN^X(t)|\mathcal{F}_t\right] = X(t)\mu^X(t)dt. \tag{2.3}$$

where dt is the Lebesgue measure.

Proof 2.2.1 *This is an immediate consequence of properties conditional expectation and the continuity of X as, almost surely,*

$$E\left[X(t)dN^X(t)|\mathcal{F}_t\right] = X(t+)\mu_X(t)dt = X(t)\mu_X(t)dt$$

where $X(t+)$ is the right limit of X in t.

Statistical Issues

Our aim is to estimate the cross-covariation structure (and the cross-correlation structure, after normalization) of the *underlying* process

$$\gamma^{XY}(h) = E\left[X(t)Y(t-h)\right],$$

as it is a sufficient statistic for Granger causality estimation [86], as well as the estimation of Φ using Yule-Walker equations [29]. A standard estimator in the literature for regularly observed time series [29, 28, 146] is

$$\widehat{\gamma}^{XY}_{\text{regular}}(h) = \frac{1}{T}\iint_{t,s=0}^{T} X(t)Y(s)\delta_0(t-s+h)dsdt$$

where $\delta_0(\cdot)$ is the dirac function centered at 0.

The Epps effect with irregular observation

However, when observations are irregular the naive estimators designed for regular observations take the form

$$\widehat{\gamma}^{XY}_{\text{naive}}(h) = \frac{1}{T}\iint_{t,s=0}^{T} X(t)Y(s)\delta_0(t-s+h)dN^X_t dN^Y_s.$$

and in the case of an observation process which is a non-degenerate point process (N^X, N^Y), this quantity evaluates to 0 almost surely [47] because X and Y are never observed simultaneously.

Theorem 2.2.1 *The estimator $\widehat{\gamma}^{XY}_{\text{naive}}(h)$ evaluates almost surely to 0 for any value of lag h.*

Proof 2.2.2 *As the bi-variate measuring process (N^X, N^Y) is a regular [42] Poisson process, using Landau notation,*

$$\mathbb{P}(\left(N^Y(s+\tau) - N^Y(s)\right)\left(N^Y(s+h\tau) - N^Y(s+h)\right) > 0) = o(\tau),$$

therefore

$$\forall t \in [0,T], \iint_{t,s\in[0,T]} X(t)Y(s)\delta_0(t-s+h)dN^Y_s dN^X_t = 0$$

as $\delta_0(t-s+h)$ is non-zero only at $t-s+h=0$. ■

The bias of the naive estimator for cross-covariance towards 0 is referred to as *Epps effect* in the field of high frequency statistics for finance [3].

To mitigate this issue created by the asynchronous observation process, we develop a novel approach to the estimation of cross-covariance (and therefore cross-correlation) based on kernel smoothing implicitly computed by specifically designed random projection in the Fourier space.

In the time domain, we define our kernel smoothing based cross-correlation estimator

$$\widehat{\gamma}^{XY}_{\text{smooth}}(h) = \frac{1}{T} \iint_{t,s=0}^{T} X(t)Y(s)\kappa(t-s+h)dN^X_t dN^Y_s \tag{2.4}$$

where $\kappa(\cdot)$ is a continuous function expressing a smoothing kernel. In Section 2.3 we will show how to efficiently evaluate this expression through the use of specific random frequency domain projections.

One approach to address the Epps effect is to align observations on a common time grid by interpolation [110] or using an estimator dedicated to the precise setting of the problem under study such as Hayashi-Yoshida [102] for correlated Brownian motions. In the following we show the statistical shortcomings of both pre-existing approaches. The analysis of these shortcomings presented below motivated our proposed approach whose statistical properties we study both theoretically and experimentally.

Bias created by interpolation

In this section, we first review existing techniques for time-domain estimation of second-order statistics for continuous stochastic processes in the context of discrete random sampling in time. Interpolating data is a usual solution in order to be able to use classic time series analysis techniques [177, 239, 77]. Unfortunately, interpolation is not always suitable, as it can create biased estimates which mislead researchers into concluding that there is significant cross-covariance where there is none [110].

A classical way to study the interactions of two asynchronously observed time series is to force the synchronicity of the timestamps by aligning them on a common time grid. While there are many interpolation techniques, a commonly used method is *last observation carried forward* (LOCF) which is not as accurate as linear interpolation or approximation by the nearest point but can efficiently be deployed as it only relies on past data at any point in time. We now consider the causality inference framework introduced in [110] and show how the LOCF interpolation technique creates cross-correlation estimates that may lead to false conclusions regarding the way (X) and (Y) influence each other.

We demonstrate, through simulation, that the asymmetric cross-correlation bias that plagues the LOCF interpolation in [110] does not appear in our proposed method. We consider two synthetic correlated Brownian motions that do not feature any lead-lag (variations in one shortly following variations in the other) and compare the estimation of cross-correlation provided the LOCF interpolation methods and our approach. After having sampled these continuous processes at random timestamps, in Figure 2.2 we compare the cross-correlation estimates obtained by LOCF interpolation and our proposed frequency domain analysis technique confirming that our method

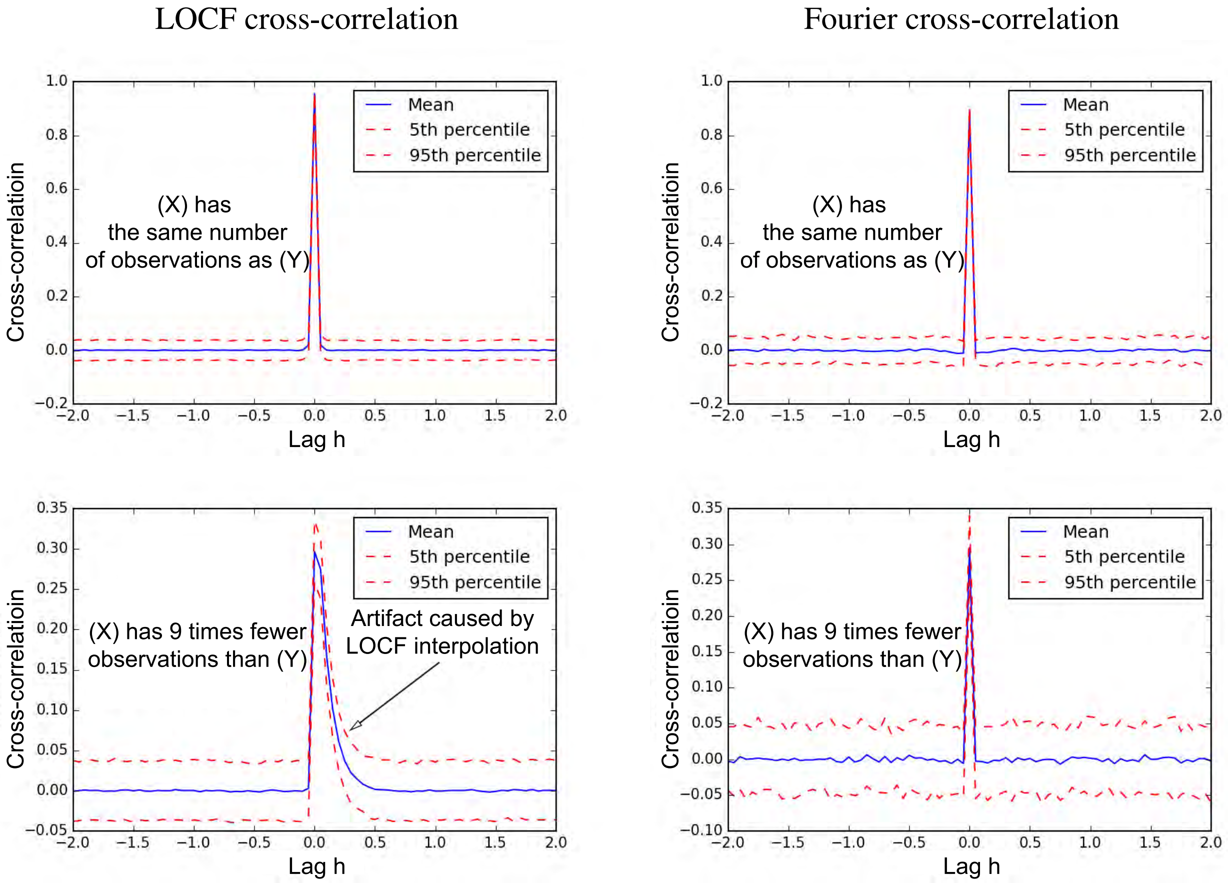


Figure 2.2: Cross-correlograms of LOCF interpolated data versus estimation via compression in frequency domain. The latter estimate does not present any spurious asymmetry due to the uneven sampling frequencies.

does not introduce estimation bias that is characterized by significantly positive cross-correlation for small positive lag values. As the two simulated processes are synchronously correlated, the estimator should find a cross-correlation of 0 except when $h = 0$.

Interpolation-free Causality Assessment

The *Hayashi-Yoshida* (HY) estimator was introduced in [102] to address the LOCF estimators tendency to discover spurious causal structure. The HY estimator of cross-correlation does not require data interpolation and has been proven to be consistent in the context of High Frequency statistics in finance [106].

Correlation of Brownian motions: HY is adapted to measuring cross-correlations between irregularly sampled Brownian motions. Considering the successor operator next for the series of timestamps of a given process, let $[t, \text{next}(t)]_{t \in \text{obs}(X)}$ and $[t, \text{next}(t)]_{t \in \text{obs}(Y)}$ be the set of intervals

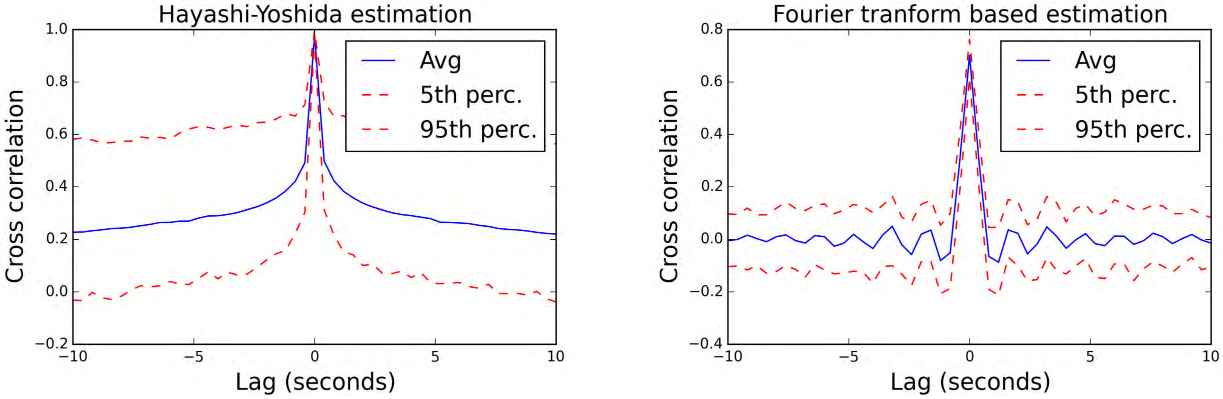


Figure 2.3: **LRD Erasure:** Monte Carlo simulation (100 samples) of two fractional Brownian motions with Hurst exponent 0.8 and simultaneously correlated increments. Spurious slowly vanishing cross-correlation hinders the HY estimation but does not affect our estimation with LRD erasure (see Section 3) as evident by nearly zero cross-correlation for non-zero lag. This issue arises when cross-correlating temperatures or water levels [163].

delimited by consecutive observations of X and Y respectively. The Hayashi-Yoshida cross-covariance estimator over the covariation of (X) and (Y) [118] is

$$\text{HY}(h) = \sum_{\substack{t \in \text{obs}(X),\, s \in \text{obs}(Y) \\ \textbf{s.t:}\ \text{ov}(t, s+h)}} (X_{\text{next}(t)} - X_t) \times (Y_{\text{next}(s)} - Y_s) \tag{2.5}$$

where $\text{ov}(t, s+h)$ is true if and only if $[t, \text{next}(t)]$ and $[s+h, \text{next}(s+h)]$ overlap. The estimator can be normalized by $\text{HY}(0)$ to estimate the cross-correlation.

No interpolation is required with HY but unfortunately this estimator is not applicable outside the context of cross-correlated standard Brownian motions. Figure 2.3 shows how the HY estimator has much more variance than our proposed estimator when estimating cross-correlation of increments on a fractional Brownian motion. Fractional Brownian motions belong to the family of Long Range Dependent (LRD) processes [56] found in general time series in climate sciences [163], finance [154], economics [220] or genomics [138]. When applied to these processes having a long memory of previous perturbations, the technique we present provides a convergent estimator with less variance as opposed to HY. In section 2.3, we show how our frequency domain based analysis naturally handles irregular observations and is able to fractionally differentiate the underlying continuous time process. In the interest of concision, we refer the reader to [70] for the definition of a fractional Brownian motion and fractional differentiation.

2.3 Smoothing kernels in the Fourier domain

In this section, we develop a theoretical analysis of the properties of estimators dedicated to the estimation of the cross-covariance structure of the underlying process $(E[X(t)Y(t+h)])_{h \in H}$

where H is a discrete grid of lags at which we want to evaluate the cross-correlation function. Estimating this second order statistics is of paramount importance as it highlights lead-lag [110] and helps infer linear causation kernels by solving the Yule-Walker equations [29].

Smoothing kernel based estimator

Classically with sparsely observed continuous stochastic processes [177] the *measuring* process is independent from the *underlying* process. We just assume that the compound model of the *measuring* point process and *underlying* process is second order stationary with co-variation function

$$\gamma^{X\mu_X,Y\mu_Y}(h) = E\left[X(t)\mu_X(t)Y(t+h)\mu_Y(t+h)\right].$$

The absence of independence between the *underlying process* and the *measuring process* is a common feature in medical data [211] or financial data [3]. The following theorem shows how this interaction affects smoothed cross-covariance estimators.

Theorem 2.3.1 *The kernel smoothing estimator does not almost surely evaluate to* 0 *as*

$$E\left[\widehat{\gamma}^{XY}_{\text{smooth}}(h)\right] = \kappa * \left(\gamma^{XdN^X,YdN^Y}\right)(h) \tag{2.6}$$

where $*$ *denotes the convolution operator and* κ *is the same as in Eq. (2.4).*

Proof 2.3.1 *By definition of kernel smoothing estimator*

$$E\left[\widehat{\gamma}^{XY}_{\text{smooth}}(h)\right] = \frac{\iint_{t,s=0}^{T} E\left[X(t)Y(s)\kappa(t-s+h)dN^Y_s dN^X_t\right]}{T}$$

As $X(\cdot)$ *and* $Y(\cdot)$ *are almost surely continuous and* $\kappa(\cdot)$ *is also assumed continuous, we can write the following:*

$$\begin{aligned}
&\iint_{t,s=0}^{T} E\left[X(t)Y(s)\kappa(t-s+h)dN^Y_s dN^X_t\right] \\
&= \iint_{t>s\in[0,T]} E[\\
&E\left[Y(s)E\left[X(t)\kappa(t-s+h)dN^X_t|F_t\right]dN^Y_s|F_s\right]]dsdt \\
&+ \iint_{t<s\in[0,T]} E[\\
&E\left[X(t)E\left[Y(s)\kappa(t-s+h)dN^Y_s|F_s\right]dN^X_t|F_t\right]]dsdt \\
&= \iint_{t,s=0}^{T} E\left[X(t)\mu_X(t)Y(s)\mu_Y(s)\kappa(t-s+h)\right]dsdt
\end{aligned}$$

if we use Proposition 2.2.1 and recombine the two partitions of the double integral. Rewriting the quantity above:

$$\int_{s=0}^{T} E\left[\int_{t=0}^{T} X(t)\mu_X(t)Y(t-s+h)\mu_Y(t-s+h)dt\right]\kappa(s)ds$$
$$=\int_{s\in[0,T]} T\gamma^{XdX,YdY}(h-s)\kappa(s)ds$$

which concludes the proof. ■

Bias analysis for smoothing kernel design

We want to compare the estimators above with of the quantity of interest, $\gamma^{XY}(\cdot) = E\left[X(t)Y(t+h)\right]$.

Corollary 2.3.1 *With the smooth kernel estimator we propose, $\widehat{\gamma}^{XY}_{\text{smooth}}(h)$ the bias term is now*

$$\gamma^{XY}(\cdot) - \kappa * \left(\gamma^{XY} \times \gamma_\mu\right)(\cdot)$$

where

$$\gamma_\mu = E\left[\mu_X(t)\mu_Y(t+h)\right].$$

Let us now analyze this bias term in the simplest setting where the *measuring* process is a bivariate uniform Poisson process with independent components of intensities ν_X and ν_Y respectively and is independent from the *underlying* process.

In such a setting, for any real valued h,

$$\gamma_\mu(h) = E\left[\mu_X(t)\mu_Y(t+h)\right] = \nu_X\nu_Y.$$

Therefore the expected value of the estimator shrinks by a constant factor in

$$E\left[\widehat{\gamma}^{XY}_{\text{smooth}}(h)\right] = \nu_X\nu_Y(\kappa * \gamma^{XY})(h). \tag{2.7}$$

Here, as the sampling intensity decreases, the bias shrinks the estimator for cross-covariance towards 0 as intuition suggests. Further, we see that the smoothing kernel needs to be concentrated about 0 as expected. We cannot reduce this bias to 0, as κ needs to be continuous with a non-empty interior support.

Variance analysis for smoothing kernel design

We conduct of variance analysis in the simple case of a Gaussian smoothing kernel which is an extension of standard results of variance reduction for time series by Gaussian smoothing.

Proposition 2.3.1 *Assuming data is generated by the model in Eq. (2.1), if the smoothing kernel is a Gaussian density centered in 0 with standard deviation σ, the variance of $\widehat{\gamma}^{XY}_{\text{smooth}}(h)$ is $O(\frac{1}{\sigma})$.*

Proof 2.3.2 *Denoting* $\left(\mathcal{F}_t^{X,dN^X}\right)_{t\in[0,T]}$ *the history of the compound process* (X, dN^X)*, we consider*

$$E\left[\left(\widehat{\gamma}^{XY}_{\text{smooth}}(h)\right)^2|\mathcal{F}_T^{X,dN^X}\right] = \int_{t,u=0}^{T} X(t)X(u) \quad \int_{s,v=0}^{T} \kappa(t-s+h)\kappa(u-v+h)A(s,v)dN_t^X dN_u^X$$

$$\text{Where } A(s,v) = E\left[Y(s)Y(v)dN_s^Y dN_v^Y|\mathcal{F}_T^{X,dN^X}\right] = E\left[Y(s)Y(v)\mu_Y(s)\mu_Y(v)|\mathcal{F}_T^{X,dN^X}\right]$$

again by a continuity argument. Conditionally to $\mathcal{F}_T^{X,dN^X}$*, the stochastic process* $Y\mu_Y$ *is second-order stationary. With our assumptions on the generative model Eq. (2.1), the spectral density* $(F(f)_{f\in\mathbb{R}})$ *of* $Y\mu_Y$ *with respect to conditional probability to* $\mathcal{F}_T^{X,dN^X}$ *is such that for any frequency* f *the derivative of the spectral density* $\int_{\mathbb{R}}\frac{log(F'(f))}{1+f^2}df$ *is bounded. Therefore the Karhunen representation theorem [55, 119] guarantees that there exist a second-order stationary process called* (ϵ) *with orthogonal increments under the conditional probability associated with* $\mathcal{F}_T^{X,dN^X}$ *and a functional* ψ *such that*

$$Y(t)\mu_Y(t) = c(t) + \int_0^t \psi(t-s)d\epsilon_s$$

where $c(t)$ *is a deterministic process which will be ignored as it is irrelevant when studying the variance of* $(Y(\cdot)\mu_Y(\cdot))$*. Then,*

$$\int_{s,v\in[0,T]} E[\kappa(t-s+h)\kappa(u-v+h)Y(s)Y(v)\mu_Y(s)\mu_Y(v)|\mathcal{F}_T^{X,dN^X}]dsdv$$
$$= E\left[(\kappa * (Y\mu_Y)(t))^2\middle|\,\mathcal{F}_T^{X,dN^X}\right]$$

where κ *is a Gaussian smoothing kernel with standard deviation* σ *with the stochastic process* $(Y\mu_Y)$*. Convolutions being linear,*

$$\kappa * (Y\mu_Y)(t) = \int_0^t \psi(t-s)(\kappa * d\epsilon)_s$$

and as the increments of ϵ *are de-correlated conditionally to* $\mathcal{F}_T^{X,dN^X}$*,*

$$E\left[\left(\kappa * (Y\mu(Y))\right)^2(t)|\mathcal{F}_T^{X,dN^X}\right] = \int_0^t \psi(t-s)E\left[(\kappa * d\epsilon)_t^2|\mathcal{F}_T^{X,dN^X}\right].$$

The smoothing kernel κ *is here a Gaussian density with standard deviation* σ*, therefore*

$$E\left[(\kappa * d\epsilon)_t^2|\mathcal{F}_T^{X,dN^X}\right] = \frac{E\left[d\epsilon_t^2|\mathcal{F}_T^{X,dN^X}\right]}{\sigma^2}$$

therefore

$$E\left[\left(\kappa * (Y\mu(Y))\right)^2(t)|\mathcal{F}_T^{X,dN^X}\right] = \frac{E\left[(Y\mu(Y))^2(t)|\mathcal{F}_T^{X,dN^X}\right]}{\sigma}$$

and the theorem follows. ■

Bias variance tradeoff

The remarks above clearly delineate some of the design choices that we will take into account when designing the smoothing kernel κ. The theorems we proved already required that the kernel is continuous with a non-empty interior support. To decrease bias we want the kernel to be as concentrated about 0 as possible. However, at least in the Gaussian kernel family, we want to have wide enough a support so as to decrease the variance by pooling asynchronous observations together.

Operating in the frequency domain

A problem we have is that even for standard smoothing kernels the method above is expensive to compute as it requires smoothing an entire irregularly observed time series by a kernel with a theoretically infinite resolution. We solve this by employing randomized Fourier transforms.

Fourier random projection basis

First, consider the Fourier transform of an irregularly observed process $(X_t)_{t \in \text{obs}(X)}$ is defined as

$$FT[X](f) = \int_{t \in [0,T]} e^{-2\pi i f t} X(t) dN_t^X. \tag{2.8}$$

The computation of this quantity can be accelerated using the techniques described in [89].

We assume that we generate frequencies f with a probability distribution F prior to computing the corresponding set of Fourier transforms. We consider the element-wise product of the Fourier transform of Y with the complex conjugate of the Fourier transform of X:

$$SP[Y, X](f) = FT[Y](f) \times \overline{FT[X](f)}. \tag{2.9}$$

Theorem 2.3.2 *Consider the random inverse Fourier transform of this element-wise product.*

$$\begin{aligned} &E_{f \sim F}\left(e^{2\pi i f h} SP[Y, X](f)\right) = \\ &\iint_{t,s \in [0,T]} X(t) Y(s) FT[F](t - s + h) dN_t^X dN_s^Y. \end{aligned} \tag{2.10}$$

where $FT[F](\cdot)$ is the Fourier transform of the distribution of frequencies F.

Proof 2.3.3 *By definition of the Fourier transform*

$$\begin{aligned} &E_{f \sim F}\left[e^{2\pi i f h} SP[Y, X](f)\right] = \\ &E_{f \sim F}\left[\iint_{t,s \in [0,T]} e^{2\pi i f (t-s+h)} X(t) Y(s) dN_t^X dN_s^Y.\right] \end{aligned}$$

and therefore, as sampling of frequencies is independent from both the underlying process generation and the irregular sampling, the expectation of interest equals

$$\iint_{t,s\in[0,T]} E_{f\sim F}\left[e^{2\pi if(t-s+h)}\right] X(t)Y(s)dN_t^X dN_s^Y.$$

■

Corollary 2.3.2 *If F is a Gaussian centered distribution of variance σ, $FT_{f\sim F}(\cdot) = g_{\frac{1}{\sigma}}(\cdot)$ where $g_{\frac{1}{\sigma}}$ is the density of a normal distribution of standard deviation $\frac{1}{\sigma}$. Therefore:*

$$\begin{aligned} &E_{f\sim F}\left(e^{2\pi ifh}SP[Y,X](f)\right) = \\ &\iint_{t,s[0,T]} X(t)Y(s)g_\sigma(t-s+h)dN_t^X dN_s^Y. \end{aligned} \tag{2.11}$$

This enables us to retrieve the estimator in Eq. (2.4), implicitly, via the frequency domain. Therefore we have found a frequency distribution to smoothly combine the observations of irregularly observed processes.

Erasure of memory in the frequency domain

With our method the application of a linear filter such as differentiation or fractional differentiation can also be calculated implicitly in the frequency domain.

Corollary 2.3.3 *Let $A(\cdot)$ a linear filter we aim to apply to the unobserved underlying process X and $B(\cdot)$ to Y. Filtering can be conducted in the frequency domain as*

$$\begin{aligned} &E_{f\sim F}\left[e^{-2\pi ifh}FT[B]FT[Y]\overline{(FT[A]FT[X])}(f)\right] \\ &= \iint_{t,s[0,T]} A * X(t) \times B * Y(s)\kappa(t-s+h)dN_t^X dN_s^Y \end{aligned} \tag{2.12}$$

Pre-processing can therefore be translated in the frequency domain so as to study LRD processes. For a fractional differentiation of level α dedicated to erasing LRD as in Figure 2.3, we use $A(f) = (2\pi if)^\alpha$ [56]. This is the pre-processing step we use to study cross-correlation between Brownian motion increments as in Figure 2.2. It also enables us to show our estimator obtains results comparable to HY in the case of Brownian motions in Figure 2.5 although it can used much more generally.

Choosing the number of basis frequencies

It is practically impossible to project a process on a continuous distribution of elements of the Fourier basis which is what writing $E_{f\sim F}\left[e^{2\pi i f h}SP[Y,X](f)\right]$ implies, we will sample a finite number N_f of frequencies from F and then compute the associated predictions before computing

$$\begin{aligned}\widehat{\gamma}_{\text{f}}^{XY}(h) &= \frac{1}{N_f}\sum_{f=1}^{N_f}\left[e^{2\pi i f h}SP[Y,X](f)\right]\\ &= \iint_{t,s[0,T]} X(t)Y(s)\sum_{f=1}^{N_f} e^{2\pi i f(t-s+h)}dN_t^X dN_s^Y \qquad (2.13)\end{aligned}$$

whose variance will be inversely proportional to M and convergences in probability to $\widehat{\gamma}_{\text{smooth}}^{XY}(h)$ as $M\to\infty$. As the sampling process is independent from the data generation process, with simple Landau notations, in the case of a Gaussian smoothing kernel of standard deviation σ,

$$Var\left(\widehat{\gamma}_{\text{f}}^{XY}(h)\right) = O(\frac{1}{\sigma M}) \qquad (2.14)$$

where the constant depends on the properties of both the *underlying* process and the *measuring* process which we have no control over. It appears therefore that there is a cost to the compression of the information entailed in the Fourier transform sets $(FT\left[X\right](f_i))_{i=1\ldots M}$ in terms of variance. The less data we use for this sufficient statistic, the higher the variance of the cross-covariance estimator.

Communication Avoiding Random Projections

In practice, the data can be distributed across multiple computing devices or sensing platforms linked together by a lower bandwidth communication medium, sorting and collocating data involves too much communication and sufficient statistics cannot be delivered with a reactivity that enables interactive exploratory data analysis [246]. Frequency domain estimation does not require observations to be sorted chronologically and shuffled across multiple nodes of computation. Therefore, the techniques we offer, based on specific frequency domain random projections, enable substantial reduction of communication at the cost of more computations. We empirically demonstrate in Section 2.4 that this enables linear scaling on unsorted data sets scattered across nodes in a data center as we can summarize data sets of several hundreds of millions of timestamps with a few thousand Fourier transforms.

2.4 Experiments

In the following, we will use the notation $\widehat{\gamma}_{f}^{XY}(\cdot)$ for our frequency domain kernel smoothing estimator and demonstrate its properties experimentally.

Linear model estimation

A Monte Carlo experiment in which we have prescribed a certain impulse function Φ in equation Eq. (2.1) demonstrates how the cross-correlation estimates that we provide enable us to reliably retrieve Φ even though the underlying process we simulate is only observed at random times. In this subsection, and only for the sake of this experiment, we simulate a discrete time auto-regressive process with a millisecond time resolution. Once the cross-correlation function between (X) and (Y) and their auto-correlation functions have been estimated as $\left(\widehat{\gamma}_f^{XY}(h)\right)_{h\in H}$ where H is a set finite set of millisecond resolution lags, solving the Yule-Walker equations below [29] gives an estimate of $\widehat{\Phi}_f$ for Φ which we compare to the values we chose for Φ. Solving the Yule-Walker equations is equivalent to a Least Squares Regression approach [29] as it comes down to estimating the precision matrix of a group of random variables based on their covariance matrix [76]. We empirically evaluate our ability to reduce variance by simulating 1000 samples from a single bi-variate linear auto-regressive process as defined in Eq. (2.1). For each sample, we introduced a homogeneous independent random observation process with $\mu_X = \mu_Y = 0.6$. In Figure 2.4, we plot the estimated cross-correlation and estimated model parameters as a function of the lag parameter h along with a 95% confidence intervals. We observe that the cross-correlation estimates have low variance relative to the peak cross-correlation magnitude and that the linear model parameters are accurately recovered with low variance.

Estimating cross-correlations on actual data

In order to highlight significant cross-correlation between pairs of stocks, one needs to consider high frequency dynamics. As we will show in the following, cross-correlation vanishes after a few milliseconds on most stocks and futures. In these settings it is then necessary to use full resolution data which in this instance comes in the form of tables recording record bids, asks and exchanges on the stock market as they happen. The timestamps are therefore irregular and not common to different pairs of stocks. Also, stock prices can very simplistically modeled as Brownian motions and therefore feature long memory. This context is therefore in the very scope of data intensive tasks we consider. We show our novel Fourier compression based cross-correlation estimator provides consistent estimates in this setting.

Checking the consistency of the estimator

Consider ask and bid quotes during one month worth of data. We create a surrogate noisy lagged version of AAPL with a 13ms delay and 91% correlation which is named AAPL-LAG. We study four pairs of time series: APPL/APPL-LAG, AAPL/IBM, AAPL/MSFT, MSFT/IBM. We study the changes in quoted prices (more exactly, volume averaged bid and ask prices). The cross-correlograms obtained below are computed between 10 AM and 2PM for 61 days in January, February and March 2012. For each process, 3000 frequencies were used in the Fourier basis. This is several orders-of-magnitude less than the number of observations that we get per day. These range from 5×10^4 to 1×10^5. We observe an 89% average peak cross-correlation with an 8ms

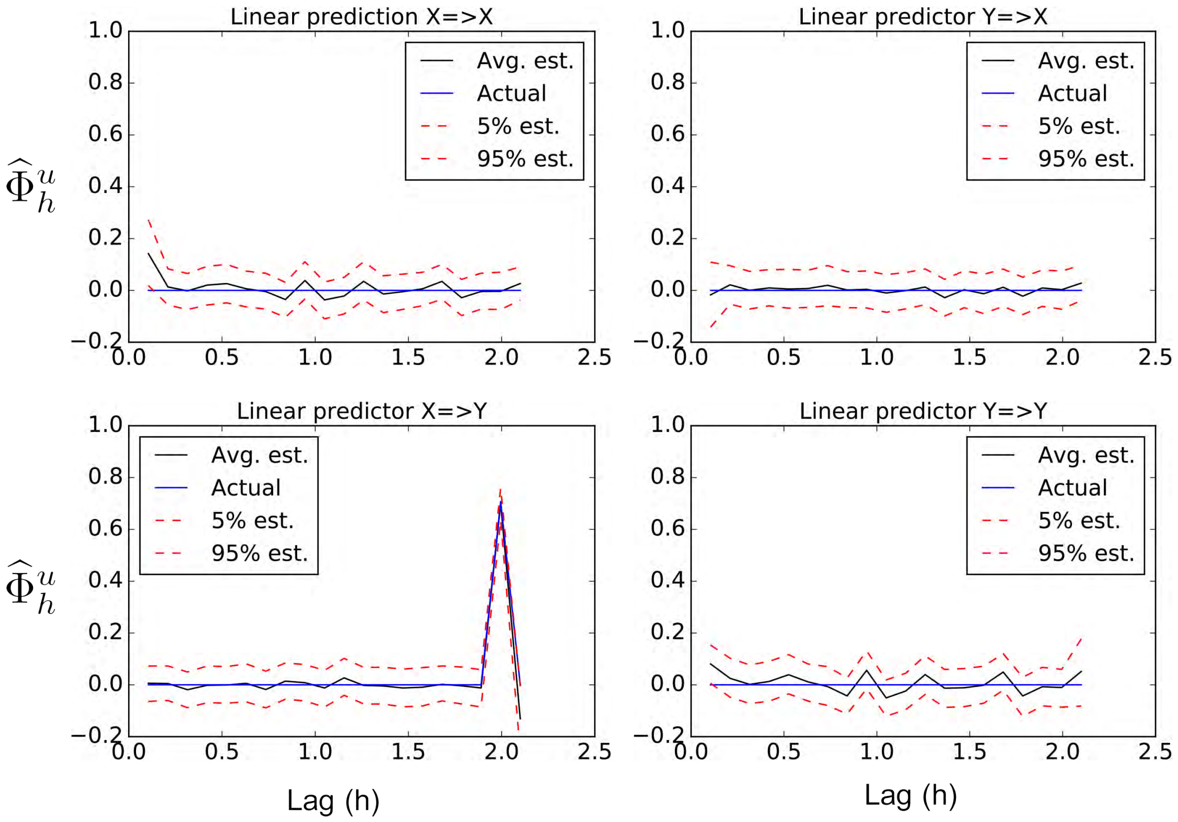


Figure 2.4: A Monte-Carlo experiment used to confirm the validity of our method. An auto-regressive model is generated and then randomly sampled with only 60% of timestamps. The smoothed cross-correlogram estimator we present enabled us to retrieve the parameters of the underlying model without bias and with little variance. Without smoothing, the cross-correlation estimator fails to capture the singularity in 2 and evaluates to 0 everywhere on average. With a probability of 90% the relative error is 8%.

delay for the surrogate pair of AAPL stocks which confirms our estimator is reliable with empirical data. In Figure 2.6 we highlight a taxonomy of causal relationships.

Random projections enable scalability

A primary goal of this work is to enable practical scalable causal inference for time series analysis. The aim of the approach we present here is to enable analytics with large numbers of records. To evaluate scalability in a real-world setting we assess the relation between AAPL and MSFT over the course of 3 months. In contrast to our earlier experiments (shown in Figure 2.6), we no longer average daily cross-correlograms in and therefore only leverage concentration in the inverse Fourier transform step of the procedure. With only 3000 projections for 5×10^6 observations

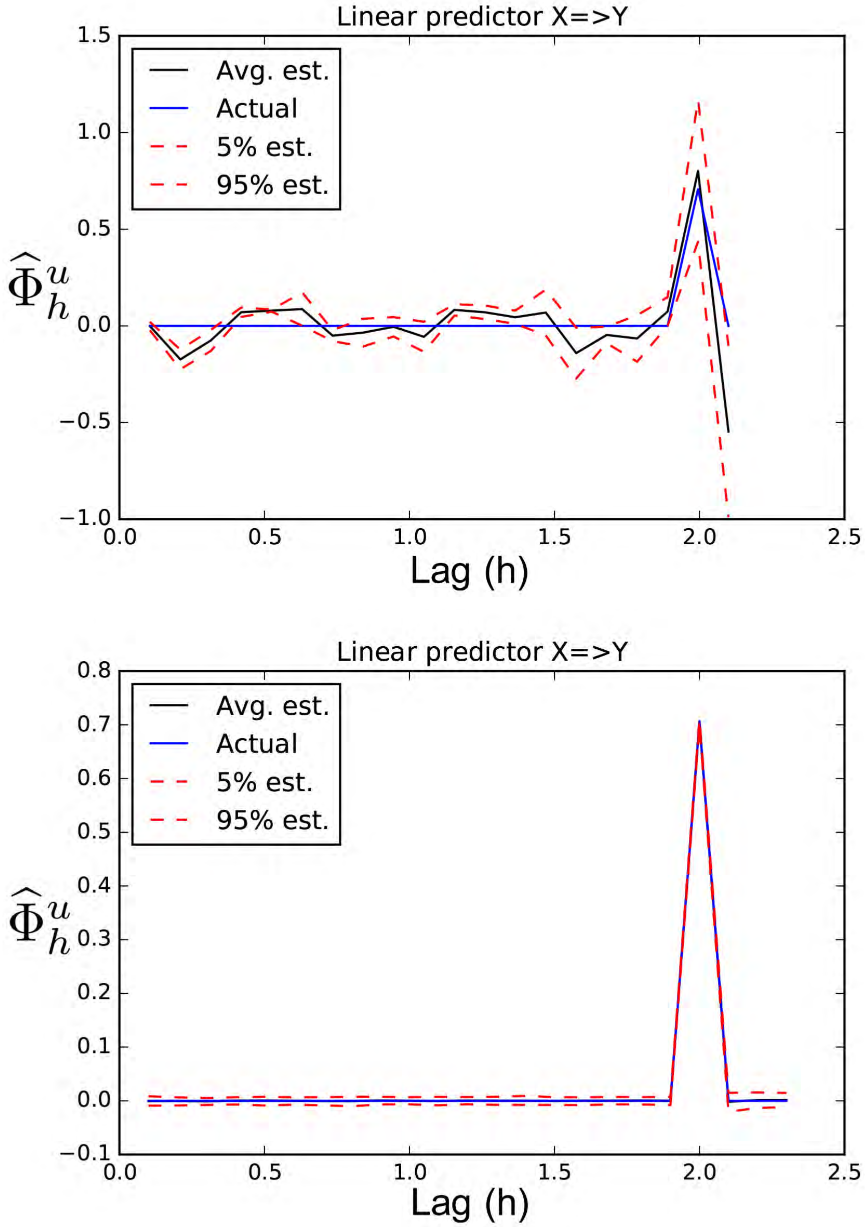


Figure 2.5: **Top:** HY estimation on synthetic Brownian motions. **Bottom:** Estimation with frequency domain differentiation. Although it has more variance, our estimator has similar bias to HY without being specifically designed to Brownian motions.

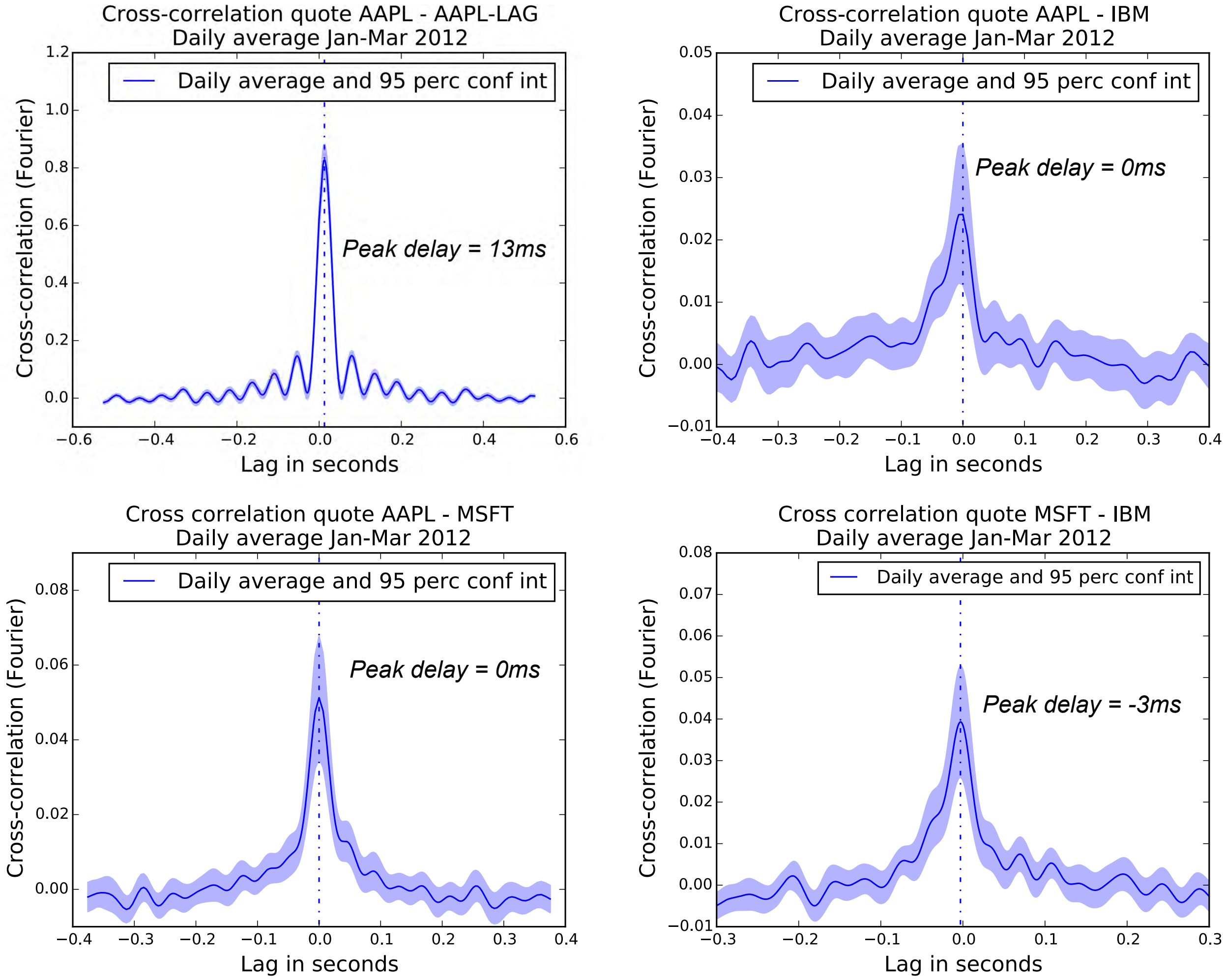


Figure 2.6: Average of daily cross-correlograms pairs of stock trade and quote data. Compression ratio is $< 5\%$. We retrieve lag and correlation accurately on surrogate data. The daily averaged cross-correlogram of AAPL and IBM is strongly asymmetric, therefore AAPL can predict IBM. The symmetry between AAPL and MSFT shows there is no such relationship between them. Symmetric and offset in correlation peak show that variations in IBM can be predicted $3ms$ ahead by observing MSFT.

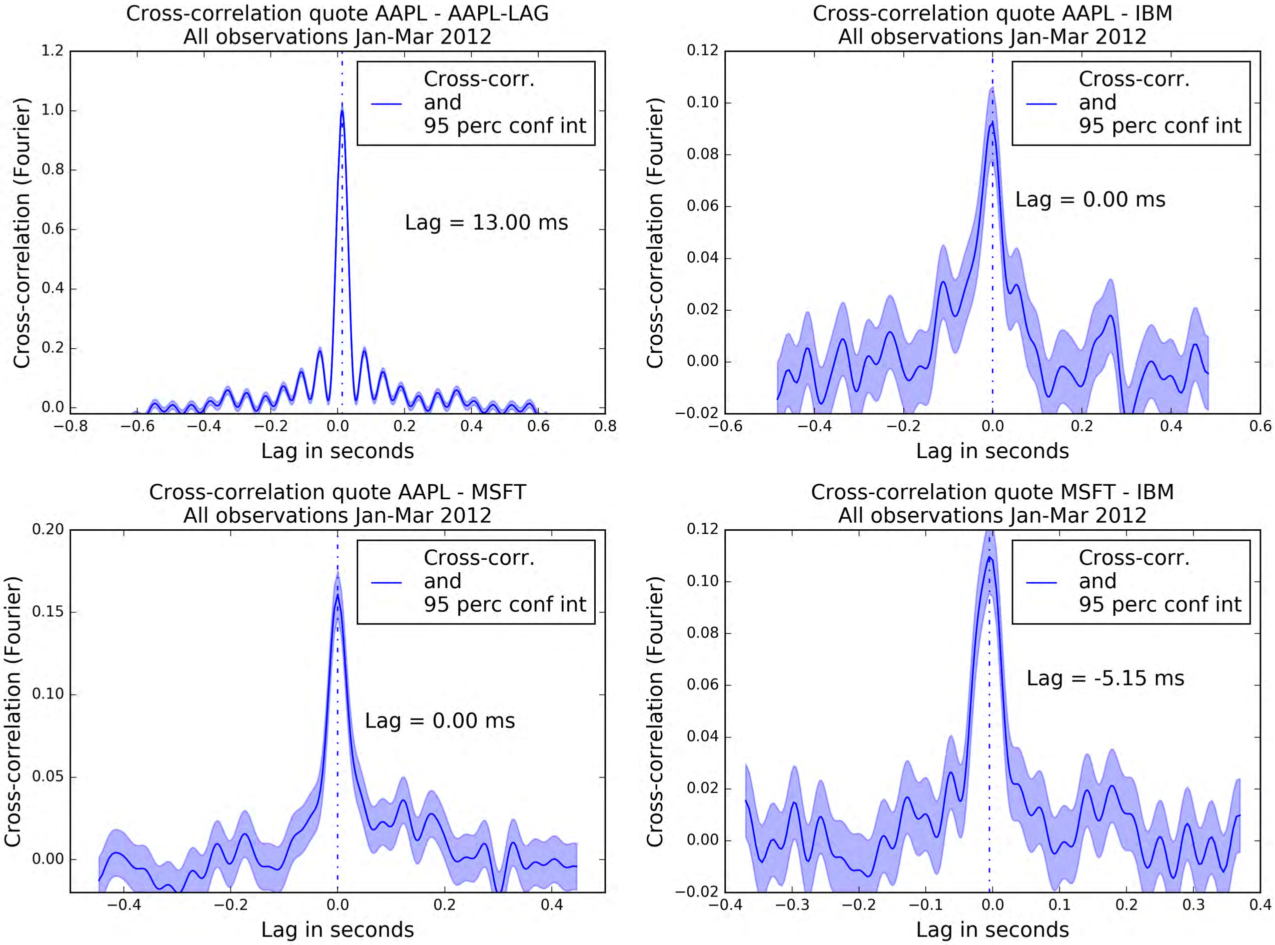


Figure 2.7: Compression ratio is $< 1\%$. On the entire data set we retrieve results similar to 2.6 therefore validating the use of our estimation of cross-correlograms in a scalable manner thanks to Fourier domain compression. Confidence bounds are computed using the asymptotic independence of spectral components [28].

per time series, the results we obtain on Figure 2.7 reveals the causal relation between AAPL, AAPL-LAG, IBM and MSFT consistently with Figure 2.6.

Choosing the number of projections:

We compute an empirical standard deviation of the daily cross-correlogram obtained in January 2012 (19 days) for MSFT (Microsoft) and INTC (Intel) with 10, 100, 1000 and 10000 projections to confirm our theoretical conclusions. Figure 2.8 shows that, as expected, the variance decreases linearly with the number of projections and we can obtain reliable estimates with 1000 projections.

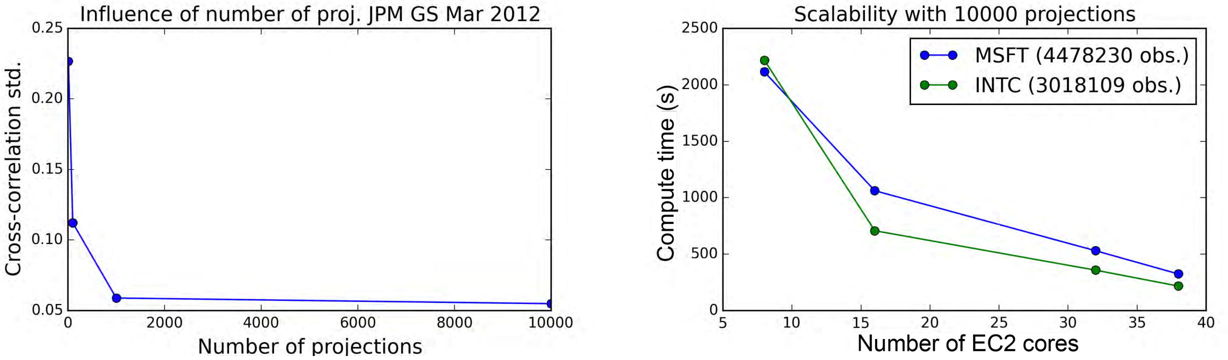


Figure 2.8: On the left we plot the empirical standard deviation of daily cross-correlograms (Figure 2.7) with respect to the number of projections showing that the variability decreases rapidly. On the right we plot the run time performance of our algorithm versus the number of Apache Spark EC2 cores showing approximately linear speedup. The small number of projections (10^4) relative to the size of the data set (10^7 records) avoids communication.

Scalability:

In order to assess the scalability of the algorithm in a situation where communication is a major bottleneck, we run the experiment with Apache Spark on Amazon Web Services EC2 machines of type r3.2xlarge. In Figure 2.8 we show that even with a large number of projections (10000) which is enough to considerably reduce the variance of the estimator the communication burden is low enough to achieve nearly linear speedup.

2.5 Conclusions of the chapter

In this chapter, we address the issues hampering cross-covariance estimation for randomly observed continuous stochastic processes. Pre-existing methods suffer from bias issues that can mislead researchers into identifying linear causality where there is none, lack strong statistical guarantees with general assumptions on the process of interest and rely on high communication requirements to be computed in the distributed setting. After having defined a new estimator for cross-covariance that does not systematically evaluate to 0, we analyze the bias/variance trade-off related to the kernel smoothing our method relies on. To enable scalability, we leverage the careful design of random Fourier transforms to implicitly compute kernel smoothing between asynchronously observed time series in the frequency domain. We show with simulated data that the estimator we propose reliably retrieves model parameters. We then demonstrate that it enables a scalable study of stock market pair cross-correlation with tens of millions of high frequency recordings.

In the present chapter, frequency domain representations appear as very advantageous when computing kernel smoothing on irregularly observed time series. The main theoretical contribution resides in the proofs we give under general assumptions on the consequence of kernel smoothing for event-driven irregularly observed time series. Such contribution is made practically usable at scale by computing effectively in the dual frequency domain which allows linear time complexity fractional differentiation. The homeomorphic properties of the Fourier transform in the case of translation invariant operators such as high-pass filters make the corresponding alternate representation a very appealing candidate to analyze time series in a scalable manner.

The idea of using an alternate basis to effectively compute will remain core to the rest of the thesis. We seek alternate concise representations to enable scalability while finding similar terms to express the properties of heterogenous time series. The chapter introduces a key pattern: the approach leads to data efficient and reliable estimation strategies. Enabling homogenous studies in a scalable manner thanks to communication avoidance leads us naturally to consider statistically reliable methods.

The next chapter introduces another duality in the form of Laplace transform whose definition is causal and better suited for control theory. The study of an infinite dimensional system is rendered elegant and simple thanks to the alternate representation we leverage. The next chapter will aim at getting a better theoretical understanding of the system we consider: an infinite dimensional model for macroscopic traffic on a freeway. Even in a setting where computations are not a key challenge, we find that employing a dual representation is extremely useful.

Chapter 3

Dual representations: from analysis to control

The previous chapter exposed multiple strategies to overcome the issues of scale, irregular sampling, Long Range Dependence while enabling expressiveness in the analysis of time series. The core focus of the thesis remains in the present chapter on the estimation of dynamical models otherwise referred to as system identification [31]. One of the key incentives to identify a system resides in the ability we get to control the system based on the knowledge of how it will react to input commands and perturbations.

Before we start delving into discretely separated problems in the following chapters, let us show alternate representations in the spectral domain help make critical control-related properties of systems of 2×2 coupled Partial Differential Equations (PDEs) evident although the system under study consists of a continuous of spatial elements evolving under different yet related conditions. The current chapter leverages the elegance of Laplace spectral representations in order to analyze an infinite dimensional dynamical system: the Aw-Rascle-Zhang system of differential equations.

The motivation to devise such a spectral representation for a PDE is providing a better understanding of the traffic dynamics commuters are exposed to and particularly dangerous traveling waves of braking vehicles. The system of PDEs under study provides cutting edge modeling ability for macroscopic freeway traffic – in particular the prediction of traveling waves. Our aim is two-fold. First, we delineate a stable regime in which linearizing the non-linear system of interest constitutes a reliable approximation. Second, we explore the source of traveling waves which lead to exponentially growing divergence in the linearized system. We employ the Laplace transform of the time series of interest because our aim is to gain the ability to control the system by actuation and therefore we want to use a causal transform that does not depend on infinite past information. The Laplace transform LP clearly answer this need as it is defined by:

$$\forall f \in L^2(\mathbb{R}),\; z \in \mathbb{C} \to LP[f](z) = \int_{t=0}^{\infty} f(t) e^{-izt} dt.$$

The chapter is organized as follows. We start from the classical Aw-Rascle-Zhang (ARZ) model for freeway traffic and develop a spectral analysis of its linearized version. We then show how a counterpart to the Froude number in hydrodynamics can be defined that enables a classification of the nature of vehicle traffic flow using the explicit solution resulting from spectral analysis of the linearized system. We prove that our linearization about an equilibrium is stable for congested regimes and convective-unstable otherwise. NGSIM data for congested traffic trajectories is used to compare the linearized model's predictions with actual macroscopic behavior of traffic. The model is shown to achieve good accuracy for speed and flow while frequency domain representations enable fast and elegant calculation of the solution. The predicted patterns replicate the propagation of boundary conditions' oscillations into the interior resolution domain of the PDE under study which simulates numerically how information propagates within the continuous time and space system.

From a physics standpoint, one goal of this research (on physics of traffic) is to gain a better understanding of the core dynamics of the system so as to design new coordination strategies for ramp metering and varying speed limits.

Traffic macroscopic models

First-order traffic macroscopic models such as *Lighthill–Whitham–Richards* (LWR) model [141, 196] have remained very popular since the 1950's as they became more accurate thanks to diversified and more realistic fundamental diagrams [90, 171, 46, 234, 174, 73, 227, 79] and numerical methods that were simple to implement such as the Godunov scheme ([82, 173, 46, 45]). However, they have inherent shortcomings most of which are discussed at length in [44] such as failure to capture accurately shock structure, light traffic dynamics, and stop-and-go behavior, otherwise known as traffic oscillations. Oscillations have attracted increasing attention in transportation research [104]. For example jamitons, traffic jams that appear without the presence of a bottleneck, have been reproduced in contained experiments [215, 72] and explained theoretically as the result of both particular configurations of the traffic system [73] and fuzzy fundamental diagrams [207].

Non-linear second order models such as Payne-Whitham (PW) [178, 238] were first presented as a compelling alternative to first order models that accounted for many of these empirical features. As pointed out in [44] the PW approach has issues in both the derivation of its equations, and its predictions. The model relies on the assumption that spacing and speed vary slowly, yielding negligible second and third derivatives for these quantities. This is contradictory with the observations of Newell [172] given that the car-following model predicts sharp changes in these quantities. Moreover, [247] proved that, with a non-zero viscosity coefficient, the PW model violates the anisotropy of traffic flow and can predict negative speeds.

Criticism towards that first generation of second order models led to a more realistic second generation. In [248], Zhang proposed a modification of the momentum equation of the PW model to handle the issue of backward-propagating traffic. Soon after, Aw and Rascle [10] presented a model with the following momentum equation:

$$(v + p(\rho))_t + v(v + p(\rho))_x = \frac{V(\rho) - v}{\tau}. \tag{3.1}$$

Including the pressure term, $p(\rho)$, in a convective derivative guarantees that no information travels faster than the speed of the cars. Aw and Rascle demonstrated in [10] that "with a suitable choice of function p," the above class of models avoids inconsistencies of earlier second-order models. Zhang proposed in [247] the same model with $p(\rho) = -V(\rho)$. With this choice of $p(\rho)$, the model is referred to as the *Aw–Rascle–Zhang* (ARZ) model, which is the model used in the present chapter.

This model has since been thoroughly studied. Rascle later proved in [190] that the relaxed model converges towards the LWR model as $\tau \to 0$. Conditions for solution to the Riemann problem to exist in all conditions were studied in [131]. The derivations in [162] and [80] showed how to embed this model in a network of junctions. Godunov discretization scheme and numerical comparison showed in [153] that ARZ model fits real data better than the LWR model. Relaxed approximations with constant wave have also been used to provide numerical solutions in [51].

More recent models such as Gupta-Katiyar (GK) (see [95]) have introduced an approximate expression of headway as a perturbation series in car following models and do not feature the shortcomings of information travelling faster than traffic does or negative speeds. This asymptotically anisotropic model presents realistic shock wave structures as studied in [92] and allows practitioners to predict reaction of traffic to varied on-ramp flow structures on freeways [182] or bottlenecks [98]. This new model also interestingly extends to the case of multiple lanes ([93]) and more importantly to entire traffic network consisting of several sections ([94]).

New models are also developed now as in [113] that take into account reduced reaction times and radio communication in platoons of computer assisted vehicles and interestingly describe how smart cars can change traffic behavior. An acceleration term appears here, akin to that added to the AR model in [50] which gave more realism to the model. The idea of modeling traffic as a set of agents interacting as a stochastic game and the resulting kinetic theoretic model developed in [14] contributed to bridging the gap between microscopic and macroscopic models. However, we choose to study and linearized ARZ model here as it offers a good compromise between simplicity and realism.

Models appropriate for control

Instead of trying to generate a switching model as in phase transition models [39, 228, 23], our aim is to design a single representation that paves the way to designing generic control strategies readily usable in all traffic conditions. Therefore we use the ARZ model linearized around arbitrary nominal conditions.

Laplace transform and spectral analysis are powerful tools for analyzing linear systems and take into account oscillations. In that regards, other oscillation generating models could have been linearized such as [88]. Behavioral models such as in [172, 225], and more recently [35, 81], depict the effects of car-following and lane-changing on freeway dynamics, effects often cited as the cause of oscillations [156, 38, 6]. However, second-order macroscopic models are most suited for our method. Other approaches have used spectral transforms such as wavelets as in [249] or the Fourier-Galerkin transform as in [222]. Our Laplace transform based study is different in that it is fundamentally physics driven and does not rely on a finite difference scheme.

Our analysis of the ARZ equations is strongly inspired by the pioneering analysis of Litrico and Fromion for the Saint-Venant equations [143]. Linearizing the corresponding two equation PDE system around an equilibrium point enables the use of spectral methods to design efficient control strategies for canals [142].

Approach and contributions

The present chapter extends the corresponding spectral framework of [143] to the case of the ARZ equations so as to achieve a two-fold objective.

We aim to develop strategies that enforce ease of use of the ARZ model for stability analysis and control. Analytical solutions to these non-linear equations are difficult to derive but linearization facilitates design of efficient control schemes with multiple inputs and outputs. Such schemes have for example been set up in this framework in [125] assuming Riemann invariants are proportional on the left boundary. We do not make such an assumption but pay particular attention to the formulation of boundary conditions to guarantee the well-posedness of the problem.

We assess the quality of the model by comparing its output with actual data collected in the field (using the NGSIM dataset).

The contributions of this chapter are as follows.

Modeling: We derive the characteristic form by linearization and diagonalization of the ARZ model. This lead to the definition of a counterpart to the Froude number in hydrodynamics [143], which separates free-flow and congested regimes.

Spectral analysis: From the charateristic form we derive the spectral form: a distributed transfer function [143]. Time domain responses derived from the spectral transfer matrices show that the linearized system is unstable in the free-flow regime. These waves lead the linearized system away from its equilibrium point in the free-flow regime. This complements the empirical and theoretical findings presented in [226, 235] about convective instability propagating upstream in congested regime as well as theory and numerical experiments conducted in [97, 96] about on the GK model.

Numerical validation: A numerical experiment using NGSIM data is conducted to validate the overall procedure. Previous studies focused on averaged errors and only displayed predictions at a couple of points along the freeway [153, 66]. Here, we present an entire map of the states and conduct model assessment in a holistic manner. This procedure demonstrates that the linearized model successfully accounts for traffic oscillations and also provides simple and consistent methods to calibrate the relaxation time, τ.

Organization of the chapter

The rest of this chapter is organized as follows. In Section 3.1 we present the characteristic form of the ARZ model in several state variables, leading to the derivation of the spectral form of the flux and velocity system in the following section. We focus on these states in particular as they are the most easily observed and controlled in traffic. Properties of the spectral form in the two flow regimes are also analyzed. The last section Section focuses on numerical analysis. We present estimation procedures for (v, q, ρ) and the model parameters, comparing empirical estimates with numerical predictions of the linearized model.

3.1 The ARZ model

We consider the ARZ model with relaxation term. The model is shown here:

$$\rho_t + (\rho v)_x = 0, \tag{3.2}$$

$$(v - V(\rho))_t + v(v - V(\rho))_x = \frac{V(\rho) - v}{\tau}, \tag{3.3}$$

where ρ is the density, v is the velocity, τ is the relaxation time, and $V(\rho) = Q(\rho)/\rho$ is the equilibrium velocity profile, where $Q(\rho)$ is the density-flow relation given by the fundamental diagram. We assume that V is C^1 derivable over its domain. Without the relaxation term cars never reach the maximum allowable speed [190] and the steady-state relation between density and speed is broken in the presence of road junctions [114]. Note that at the equilibrium velocity this term is zero.

In vector form the ARZ model is

$$\begin{pmatrix}\rho\\ v\end{pmatrix}_t + \begin{pmatrix} v & \rho \\ 0 & v + \rho V'(\rho)\end{pmatrix}\begin{pmatrix}\rho\\ v\end{pmatrix}_x = \begin{pmatrix} 0 \\ \dfrac{V(\rho) - v}{\tau}\end{pmatrix}. \tag{3.4}$$

With the appropriate variable change, we can rewrite the model in the density-flow and velocity-flow forms, the latter of which is most useful to us for practical control purposes. Using the flow relation $q = \rho v$ and (3.4), the density-flow form is

$$\begin{pmatrix}\rho\\ q\end{pmatrix}_t + \begin{pmatrix} 0 & 1 \\ -\frac{q}{\rho}\left(\frac{q}{\rho} + \rho V'(\rho)\right) & 2\frac{q}{\rho} + \rho V'(\rho)\end{pmatrix}\begin{pmatrix}\rho\\ q\end{pmatrix}_x = \begin{pmatrix} 0 & 0 \\ \frac{V(\rho)}{\tau} & -\frac{1}{\tau}\end{pmatrix}\begin{pmatrix}\rho\\ q\end{pmatrix} \tag{3.5}$$

In the same manner we arrive at the velocity-flow form,

$$\begin{pmatrix}v\\ q\end{pmatrix}_t + \begin{pmatrix} v + \frac{q}{v}V'\left(\frac{q}{v}\right) & 0 \\ \frac{q}{v}\left(v + \frac{q}{v}V'\left(\frac{q}{v}\right)\right) & v\end{pmatrix}\begin{pmatrix}v\\ q\end{pmatrix}_x = \frac{1}{\tau}\begin{pmatrix} V\left(\frac{q}{v}\right) - v \\ Q\left(\frac{q}{v}\right) - q\end{pmatrix}. \tag{3.6}$$

Although the (ρ, q) and (v, q) forms are less common in transportation engineering, they bear interesting similarities to hydrodynamical systems. Moreover, they are promising for sensing problems as loop detectors typically sense (ρ, q) while GPS measurement generally yield estimates for v.

Linearization

We are interested in small deviations, $(\tilde{\rho}(x,t), \tilde{v}(x,t))$, from a given nominal profile. Consider the nominal solution $(\rho^*(x), v^*(x))(V(\rho^*) = v^*)$ satisfying $v_t = \rho_t = 0$. Then (3.4) becomes

$$v^* \rho^*_x + v^*_x \rho^* = 0, \tag{3.7}$$

$$(v^* + \rho^* V'(\rho^*)) v^*_x = \frac{V(\rho^*) - v^*}{\tau} = 0. \tag{3.8}$$

Then we must have $v^*_x = \rho^*_x = 0$, so the solution is uniform along the road.

Linearizing the ARZ model (3.4) around the nominal solution described above, we obtain

$$\begin{pmatrix} \tilde{\rho} \\ \tilde{v} \end{pmatrix}_t + \begin{pmatrix} v^* & \rho^* \\ 0 & v^* + \rho^* V'(\rho^*) \end{pmatrix} \begin{pmatrix} \tilde{\rho} \\ \tilde{v} \end{pmatrix}_x = \begin{pmatrix} 0 & 0 \\ \frac{V'(\rho^*)}{\tau} & -\frac{1}{\tau} \end{pmatrix} \begin{pmatrix} \tilde{\rho} \\ \tilde{v} \end{pmatrix}. \tag{3.9}$$

Similarly for the density-flow system (3.5), we linearize around the equilibrium $(\rho^*, q^*)(\rho^* V(\rho^*) = q^*)$ with deviations $(\tilde{\rho}(x,t), \tilde{q}(x,t))$. The linearized system is as follows:

$$\begin{aligned} \begin{pmatrix} \tilde{\rho} \\ \tilde{q} \end{pmatrix}_t + \begin{pmatrix} 0 & 1 \\ -\left(\frac{q^*}{\rho^*}\right)^2 - q^* V'(\rho^*) & 2\frac{q^*}{\rho^*} + \rho^* V'(\rho^*) \end{pmatrix} \begin{pmatrix} \tilde{\rho} \\ \tilde{q} \end{pmatrix}_x = \\ \begin{pmatrix} 0 & 0 \\ \dfrac{V(\rho^*) + \rho^* V'(\rho^*)}{\tau} & -\dfrac{1}{\tau} \end{pmatrix} \begin{pmatrix} \tilde{\rho} \\ \tilde{q} \end{pmatrix}. \end{aligned} \tag{3.10}$$

Finally, for the velocity-flow system, we have

$$\begin{pmatrix} \tilde{v} \\ \tilde{q} \end{pmatrix}_t + A \begin{pmatrix} \tilde{v} \\ \tilde{q} \end{pmatrix}_x = B \begin{pmatrix} \tilde{v} \\ \tilde{q} \end{pmatrix}. \tag{3.11}$$

where

$$A = \begin{pmatrix} v^* + \frac{q^*}{v^*} V'\left(\frac{q^*}{v^*}\right) & 0 \\ \frac{q^*}{v^*}\left(v^* + \frac{q^*}{v^*} V'\left(\frac{q^*}{v^*}\right)\right) & v^* \end{pmatrix} \tag{3.12}$$

$$B = \begin{pmatrix} -\dfrac{(v^*)^2 + q^* V'\left(\frac{q^*}{v^*}\right)}{(v^*)^2 \tau} & \dfrac{V'\left(\frac{q^*}{v^*}\right)}{v^* \tau} \\ -\dfrac{q^*\left((v^*)^2 + q^* V'\left(\frac{q^*}{v^*}\right)\right)}{(v^*)^3 \tau} & \dfrac{q^* V'\left(\frac{q^*}{v^*}\right)}{(v^*)^2 \tau} \end{pmatrix} \tag{3.13}$$

Characteristic form

We diagonalize the linearized equations to obtain a more useful form of the model, which will then be treated in the spectral domain. For all systems of coordinates (density-speed, density-flow and speed-flow), the eigen values of the linearized PDE are $\lambda_1 = v^*$ and $\lambda_2 = v^* + \rho^* V'(\rho^*)$. Note that $V'(\rho^*) \leq 0$ so $\lambda_2 \leq \lambda_1 = v^*$. Therefore this is consistent with the physical dynamics of the system as no waves travel faster than the equilibrium vehicle speed.

We have proceeded with the diagonalization of the three systems of equations. For the sake of concision we only write below the derivations corresponding to the most interesting system of coordinates to us: (v, q). Computing the eigenvalues of A and a diagonalization vector basis yields

$$\begin{pmatrix}\xi_1\\ \xi_2\end{pmatrix}_t + \underbrace{\begin{pmatrix}\lambda_1 & 0\\ 0 & \lambda_2\end{pmatrix}}_{\tilde{A}}\begin{pmatrix}\xi_1\\ \xi_2\end{pmatrix}_x = \underbrace{\begin{pmatrix}-\frac{1}{\tau} & 0\\ -\frac{1}{\tau} & 0\end{pmatrix}}_{\tilde{B}}\begin{pmatrix}\xi_1\\ \xi_2\end{pmatrix}, \tag{3.14}$$

where the characteristic coordinates are

$$\begin{pmatrix}\xi_1\\ \xi_2\end{pmatrix} = \begin{pmatrix}\dfrac{\rho^*\lambda_2}{\lambda_1-\lambda_2}\tilde{v}+\tilde{q}\\ \dfrac{q^*}{\lambda_1-\lambda_2}\tilde{v}\end{pmatrix} = \underbrace{\begin{pmatrix}\dfrac{\rho^*\lambda_2}{\lambda_1-\lambda_2} & 1\\ \dfrac{\rho^*\lambda_1}{\lambda_1-\lambda_2} & 0\end{pmatrix}}_{R}\begin{pmatrix}\tilde{v}\\ \tilde{q}\end{pmatrix} \tag{3.15}$$

The Traffic Froude Number

In fluid mechanics, the Froude number is a dimensionless number which delineates the boundary between flow regimes [214, 143]. Using the eigenvalues of the system in the characteristic form, we are able to define a useful counterpart to this number. Since $V(\rho)$ is non-increasing function, we have $V'(\rho^*) \leq 0$. Assuming $V'(\rho^*) \neq 0$ there are two flow regimes: one in which $\lambda_1\lambda_2 < 0$ and one characteristic line travels downstream whereas the other characteristic line travels upstream, and one in which $\lambda_1\lambda_2 > 0$ and both characteristic lines travel downstream. We define the *Traffic Froude Number* (TFN) as

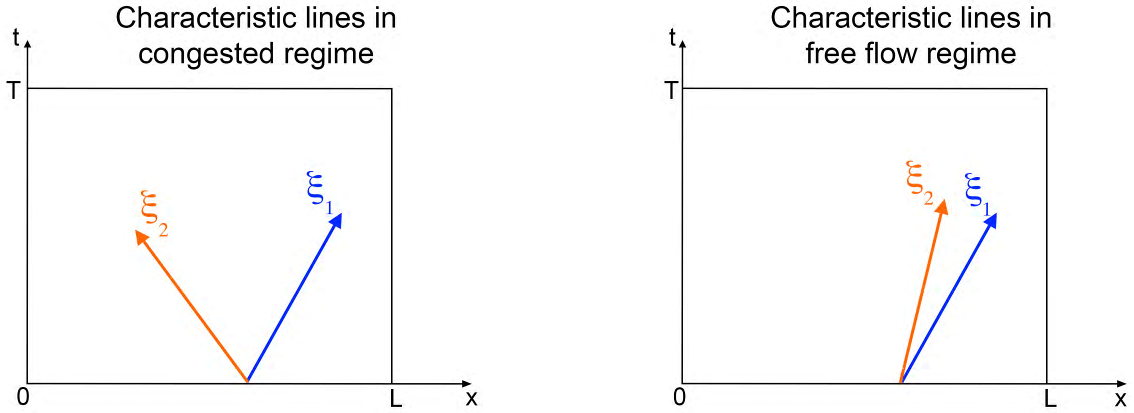
$$F = \left|\frac{\rho^* V'(\rho^*)}{v^*}\right|. \tag{3.16}$$

Then we have

$$\begin{cases} F > 1 & \Rightarrow |\rho^* V'(\rho^*)| > v^* & \Rightarrow \lambda_2 < 0\\ F < 1 & \Rightarrow |\rho^* V'(\rho^*)| < v^* & \Rightarrow \lambda_2 > 0 \end{cases}.$$

Note also that $\lambda_2 = v^* + \rho^* V'(\rho^*) = \dfrac{Q(\rho^*)}{\rho^*} + \dfrac{\rho^* Q'(\rho^*) - Q(\rho^*)}{\rho^*} = Q'(\rho^*)$. Hence the system is in free-flow when $F < 1$ and congestion when $F > 1$. In hydrodynamics these regimes are referred to as the subcritical and supercritical regimes, respectively [143]. The direction of characteristic lines is illustrated in Figure 3.1.

For traffic, the interpretation of the different regimes is somewhat different. Free flow regime corresponds to these situations where drivers are not slowed down by heavy traffic and go as fast as the desired speed. The congested regime arises when traffic is denser and, because too many cars are present on the same freeway section, drivers slow down and eventually form traffic jam. This number is therefore a dimensionless quantity that clearly delineates different physical regime of vehicular traffic.


Characteristic lines in congested regime
t
T
0
L
x
ξ2
ξ1
Characteristic lines in free flow regime
t
T
0
L
x
ξ2
ξ1

$F > 1$, $\lambda_2 < 0$, $\lambda_1 > 0$
Inputs $\xi_1(0,t)$ and $\xi_2(L,t)$

$F < 1$, $\lambda_1 > \lambda_2 > 0$
Inputs $\xi_1(0,t)$ and $\xi_2(0,t)$

Figure 3.1: Illustration of characteristic lines in congested (supercritical) and free-flow regime (subcritical) ξ_1 and ξ_2 propagate along.

3.2 Spectral analysis of the linearized ARZ model

We now consider the (v,q) system for the frequency domain analysis for practical control purposes.

State-transition matrix

Taking the Laplace transform of the diagonalized form (3.14) we obtain

$$\frac{\partial\hat{\xi}(x,s)}{\partial x} = \mathscr{A}(s)\hat{\xi}(x,s) + \mathscr{B}\xi(x,t=0^-), \tag{3.17}$$

where $\mathscr{A}(s) = \tilde{A}^{-1}(\tilde{B} - sI)$ and $\mathscr{B} = -\tilde{A}^{-1}$. The general solution to this ordinary differential equation is

$$\hat{\xi}(x,s) = \Phi(x,s)\hat{\xi}(0,s) + \Phi(x,s)\int_0^x \Phi(\nu,s)^{-1}\mathscr{B}\xi(\nu,0^-)\mathrm{d}\nu, \tag{3.18}$$

where $\Phi(x,s) = e^{\mathscr{A}(s)x}$ is the state-transition matrix. Assuming zero initial conditions we have

$$\hat{\xi}(x,s) = \Phi(x,s)\hat{\xi}(0,s). \tag{3.19}$$

To compute the exponential we diagonalize the matrix as

$$\mathscr{A}(s) = \mathscr{X}(s)\mathscr{D}(s)\mathscr{X}^{-1}(s) \tag{3.20}$$

where

$$\mathscr{X}(s) = \begin{pmatrix} 0 & \dfrac{\lambda_2 - (\lambda_1 - \lambda_2)\tau s}{\lambda_1} \\ 1 & 1 \end{pmatrix}, \mathscr{D}(s) = \begin{pmatrix} -\frac{s}{\lambda_2} & 0 \\ 0 & -\frac{1+\tau s}{\tau\lambda_1} \end{pmatrix}. \tag{3.21}$$

Hence

$$\Phi(x,s) = \mathscr{X}^{-1}(s)e^{\mathscr{D}(s)x}\mathscr{X}(s) = \begin{pmatrix} \phi_{11}(x,s) & \phi_{12}(x,s) \\ \phi_{21}(x,s) & \phi_{22}(x,s) \end{pmatrix}, \tag{3.22}$$

with

$$\phi_{11}(x,s) = e^{-\frac{x}{\tau\lambda_1}}e^{-\frac{x}{\lambda_1}s}, \tag{3.23a}$$

$$\phi_{12}(x,s) = 0, \tag{3.23b}$$

$$\phi_{21}(x,s) = \frac{\lambda_1\left(e^{-\frac{x}{\tau\lambda_1}}e^{-\frac{x}{\lambda_1}s} - e^{-\frac{x}{\lambda_2}s}\right)}{\lambda_2 - \tau(\lambda_1-\lambda_2)s}, \tag{3.23c}$$

$$\phi_{22}(x,s) = e^{-\frac{x}{\lambda_2}s}. \tag{3.23d}$$

Let $\alpha = -\dfrac{\lambda_2}{\tau(\lambda_1-\lambda_2)}$. It is clear that $\phi_{11}(x,s)$ is composed of the product of a distributed delay corresponding to information propagating at speed λ_1 and an exponential attenuation where $\tau\lambda_1$ plays the role of a characteristic spatial length. Similarly ϕ_{22} is a delay corresponding to information propagating at speed λ_2. The interpretation of ϕ_{21} is more difficult. In low frequencies ($|s| \ll |\alpha|$), this transfer function takes a much more transparent form. Indeed, in the expression $\phi_{21}(x,s) = -\frac{\lambda_1}{\lambda_2}\frac{\alpha}{s+\alpha}e^{-\frac{x}{\lambda_2}s}\left(1 - e^{-\frac{x}{\lambda_1\tau\alpha}(s+\alpha)}\right) \simeq -\frac{\lambda_1}{\lambda_2}e^{-\frac{x}{\lambda_2}s}\left(1-e^{-\frac{x}{\lambda_1\tau}}\right)$ the transfer function appears as the combination of a distributed delay where λ_2 is the propagation speed and a distributed gain where $\lambda_1\tau$ is a characteristic distance.

Free-flow case ($F < 1$)

Consider the system in the free-flow regime. With $\xi_1(0,t)$ and $\xi_2(0,t)$ as the inputs and $\xi_1(L,t)$ and $\xi_2(L,t)$ as the outputs, the distributed transfer matrix is exactly the state-transition matrix $\Phi(x,s)$. Using (3.15), we can write

$$\begin{pmatrix} \widetilde{v}(x,s) \\ \widetilde{q}(x,s) \end{pmatrix} = \underbrace{\begin{pmatrix} \dfrac{\rho^*\lambda_2}{\lambda_1-\lambda_2} & 1 \\ \dfrac{\rho^*\lambda_1}{\lambda_1-\lambda_2} & 0 \end{pmatrix}^{-1} \Phi(x,s) \begin{pmatrix} \dfrac{\rho^*\lambda_2}{\lambda_1-\lambda_2} & 1 \\ \dfrac{\rho^*\lambda_1}{\lambda_1-\lambda_2} & 0 \end{pmatrix}}_{\Psi(x,s)} \begin{pmatrix} \widetilde{v}(0,s) \\ \widetilde{q}(0,s) \end{pmatrix} \tag{3.24}$$

with

$$\psi_{11}(x,s) = \frac{\alpha e^{-\frac{x}{\lambda_1}\left(s+\frac{1}{\tau}\right)} + s e^{-\frac{sx}{\lambda_2}}}{s+\alpha}, \tag{3.25a}$$

$$\psi_{12}(x,s) = \frac{1}{\rho^*\tau}\frac{e^{-\frac{sx}{\lambda_2}} - e^{-\frac{x}{\lambda_1}\left(s+\frac{1}{\tau}\right)}}{s+\alpha}, \tag{3.25b}$$

$$\psi_{21}(x,s) = -s\rho^*\tau\alpha\frac{e^{-\frac{sx}{\lambda_2}} - e^{-\frac{x}{\lambda_1}\left(s+\frac{1}{\tau}\right)}}{s+\alpha}, \tag{3.25c}$$

$$\psi_{22}(x,s) = \frac{s e^{-\frac{x}{\lambda_1}\left(s+\frac{1}{\tau}\right)} + \alpha e^{-\frac{sx}{\lambda_2}}}{s+\alpha}. \tag{3.25d}$$

It could appear at first sight that $-\alpha$ (here a positive real) is a singularity of the transfer functions and the system is not bounded-input/bounded-output stable. However, we have $\frac{1}{\lambda_1}\left(-\alpha+\frac{1}{\tau}\right) = \frac{1}{\tau(\lambda_1-\lambda_2)} = \frac{-\alpha}{\lambda_2}$, thus a Taylor expansion about $-\alpha$ shows that numerators and denominators cancel each other out for $s \to -\alpha$. It follows that $-\alpha$ is not a pole of any transfer function. The spectrum of $\Psi(x,s)$ is identical to that of $\Phi(x,s)$ which is lower triangular and has eigenvalues $e^{-\frac{x}{\tau\lambda_1}}e^{-\frac{x}{\lambda_1}s}$ and $e^{-\frac{x}{\lambda_2}s}$ whose module is trivially bounded by 1. This proves the H_∞ norm of Ψ is 1 and therefore the system is Bounded-Input/Bounded-Output (BIBO) stable. This proves that the output remains bounded for a given value of x.

We will show below that a conic region of the $[0,T]\times[0,L]$ domain features exponential growth in free-flow regime. This arises when changing t and x simultaneously and complements the conclusion formulated above (in which t varies and x remains constant).

Low frequency approximation for physical variables in free-flow regime

Analyzing the expressions above becomes easier when approximating them for $|s| \ll |\alpha|$. This corresponds to traffic flow varying slowly and smoothly. We find the following approximate expressions for the transfer functions:

$$\psi_{11}(x,s) \simeq e^{-\frac{sx}{\lambda_2}}e^{-\frac{x}{\tau\lambda_1}}, \tag{3.26a}$$

$$\psi_{12}(x,s) \simeq \frac{1}{\rho^*\tau\alpha}e^{-\frac{sx}{\lambda_2}}\left(1-e^{-\frac{x}{\tau\lambda_1}}\right), \tag{3.26b}$$

$$\psi_{21}(x,s) \simeq -s\rho^*\tau e^{-\frac{sx}{\lambda_2}}\left(1-e^{-\frac{x}{\tau\lambda_1}}\right), \tag{3.26c}$$

$$\psi_{22}(x,s) \simeq e^{-\frac{sx}{\lambda_2}}. \tag{3.26d}$$

Interpreting the low frequency expressions is fairly straightforward. In ψ_{11}, $e^{-\frac{sx}{\lambda_2}}$ is a distributed delay with propagation speed λ_2 and $e^{-\frac{x}{\tau\lambda_1}}$ a distributed gain with characteristic distance $\tau\lambda_1$. In ψ_{12} and ψ_{21} we can notice the combination of a distributed delay whose characteristic speed is λ_2 and a distributed gain whose characteristic distance is $\tau\lambda_1$. It is also remarkable that $\widetilde{q}(x,s)$ appears as the result of a derivator applied to $\widetilde{v}(0,s)$. The approximate expression for ψ_{22} highlights the presence of a distributed delay where information propagates at speed λ_2. This low frequency analysis therefore highlights fundamental mechanisms core to the physics of traffic that can be decomposed as modular elements here.

Bode plots for free-flow regime

We generate Bode plots using the following parameters taken from [107]: q_{max} = 1300 veh/h, ρ_{max} = 0.1 veh/m, and L = 100 m. The Greenshields fundamental diagram, where the equilibrium flow is $Q(\rho) = 4\frac{q_{\text{max}}}{\rho_{\text{max}}^2}\rho(\rho_{\text{max}} - \rho)$, is used to approximate the fundamental diagram. For inhomogeneous second-order models, the relaxation time, τ, falls in the range of about 14-60 seconds [66]. A relaxation time of τ = 15 s is used for the following simulations. We simulate for $\rho^* = 0.01$ veh/m. Here the characteristic frequency of the system, $|\alpha|$, equals 0.53 Hz which is indeed sensible for traffic flow modeling.

The Bode plots for the physical variables are displayed in Figure 3.3. For the Riemann invariants only $\phi_{21}(x, s)$ and $\phi_{22}(x, s)$ are represented in Figure 3.2 ($\phi_{11}(x, s)$ and $\phi_{12}(x, s)$ are only delay functions).

For transfer functions featuring $1 - e^{-\frac{x}{\lambda_1\tau\alpha}(s+\alpha)}$ as a factor (that is to say ϕ_{21}, ψ_{12}, and ψ_{21}) one can observe in the corresponding Bode plots that the value of the log-gain in high frequency tends to vary very sharply. Indeed, with $s = jw$,

$$\left|1 - e^{-\frac{x}{\lambda_1\tau\alpha}(s+\alpha)}\right| = e^{-\frac{x}{\lambda_1\tau}}\sqrt{\left(e^{\frac{x}{\lambda_1\tau}} - \cos\left(\frac{w}{\lambda_1\tau\alpha}x\right)\right)^2 + \sin^2\left(\frac{w}{\lambda_1\tau\alpha}x\right)}.$$

Therefore, if the spatial pseudo-period $\tilde{L} = \frac{2\pi}{w}\lambda_1\tau\,|\alpha|$ is low enough, near zero values appear when x is a multiple of $\tilde{L}$. This explains the irregular shape of the distributed Bode plots of ϕ_{21}, ψ_{12}, and ψ_{21} for frequencies $w \gg 2\pi\frac{\lambda_1\tau|\alpha|}{L} = 6.53$ Hz. This does not impact the stability of the system. Bode plots only look irregular about such points because of the logarithmic scale.

Step responses

We analyze the behavior of the system given step inputs $\tilde{v}(0, t) = v_{\text{step}}H(t)$ and $\tilde{q}(0, t) = q_{\text{step}}H(t)$, where $H(\cdot)$ is the Heaviside function. The step responses can be explicitly computed

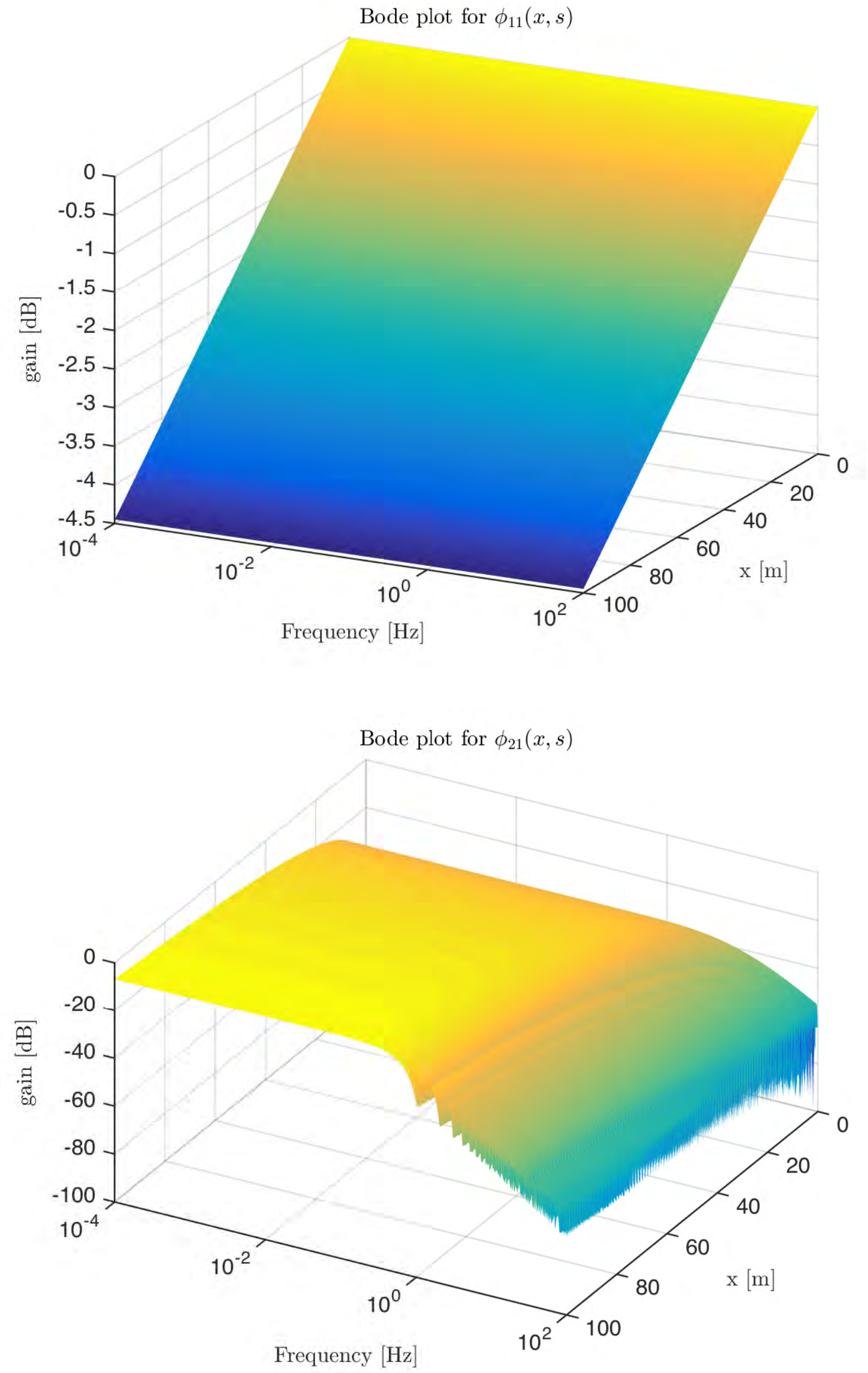
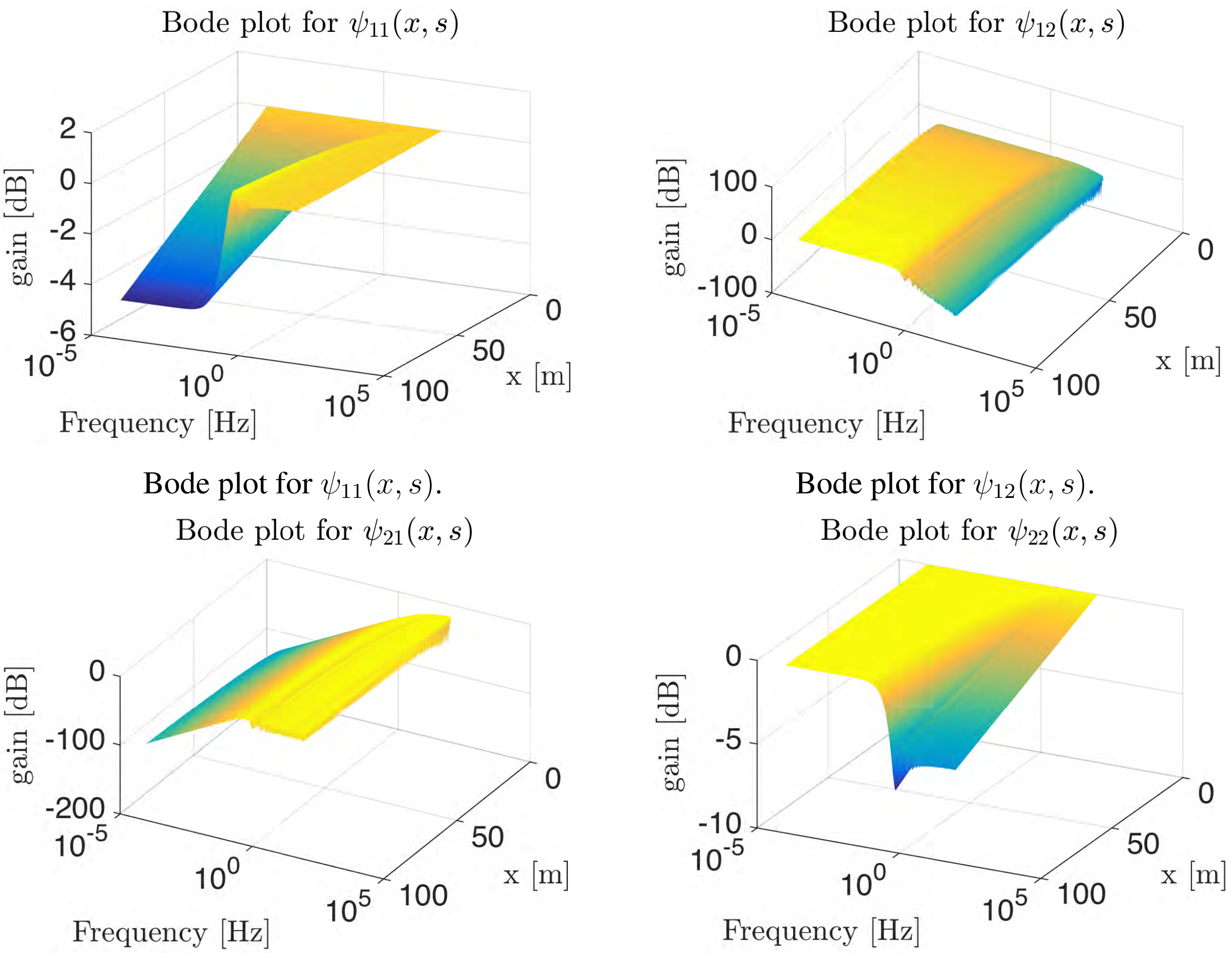


Figure 3.2: Spatial magnitude Bode plots for Riemann invariants in free-flow regime ($|\alpha|$ =0.53Hz)



Bode plot for $\psi_{11}(x,s)$.

Bode plot for $\psi_{12}(x,s)$.

Bode plot for $\psi_{21}(x,s)$.

Bode plot for $\psi_{22}(x,s)$.

Figure 3.3: Spatial magnitude Bode plots for physical variables in free-flow regime ($|\alpha| = 0.53$ Hz)

from the spectral responses. Letting $H_1(t,x) = H\left(t - \frac{x}{\lambda_1}\right)$ and $H_2(t,x) = H\left(t - \frac{x}{\lambda_2}\right)$:

$$\begin{aligned}\tilde{v}(x,t) &= \bar{v}e^{-\frac{x}{\lambda_1\tau}}H_1(t,x) \\ &\quad + \bar{v}e^{-\alpha\left(t-\frac{x}{\lambda_2}\right)}(H_2 - H_1)(t,x) \\ &\quad - \frac{\bar{q}}{\rho^*\tau}\left(e^{-\frac{x}{\lambda_1\tau}}H_1(t,x) - H_2(t,x)\right) \\ &\quad - \frac{\bar{q}}{\rho^*\tau}e^{-\alpha\left(t-\frac{x}{\lambda_2}\right)}(H_2 - H_1)(t,x) \end{aligned} \tag{3.27}$$

$$\begin{aligned}\tilde{q}(x,t) &= \bar{v}\rho^*\tau\alpha e^{-\alpha\left(t-\frac{x}{\lambda_2}\right)}(H_1 - H_2)(t,x) \\ &\quad + \bar{q}H_2(t,x) \\ &\quad + \bar{q}e^{-\alpha\left(t-\frac{x}{\lambda_2}\right)}(H_1 - H_2)(t,x) \end{aligned} \tag{3.28}$$

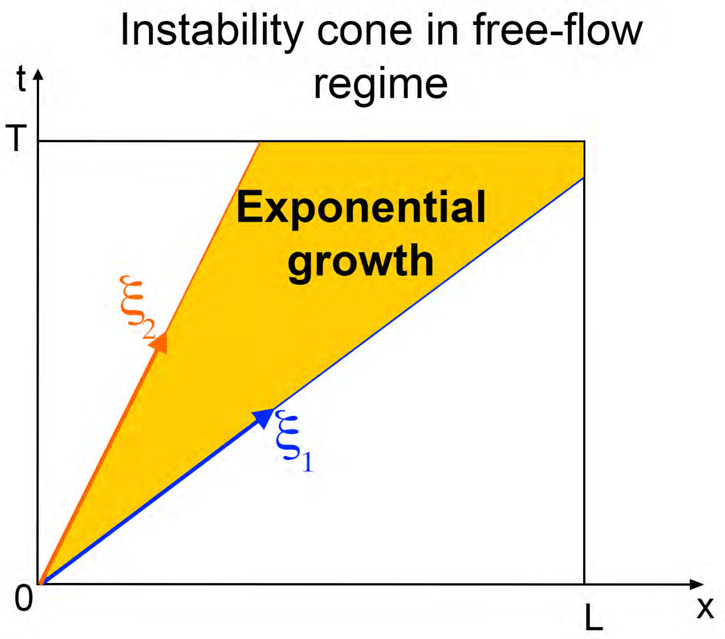


Figure 3.4: Illustration of the exponential growth cone appearing in the free-flowing regime for the time domain expressions of v and q.

With this set of time domain expressions, we can see that a cone of exponentially growing speed and flow linearization errors generally appears between the characteristic lines corresponding to λ_1 and λ_2. This is caused by α being negative in the free flow regime and means that, in this region of the domain $[0, T] \times [0, L]$, the (v, q) state of the linearized system can diverge exponentially fast from the linearization point. This is consistent with the observations in [120] where small local perturbations occurring in free-flow regime can cause traffic to transition durably to the congested regime.

Congested regime ($F > 1$)

We now consider the system in the congested regime.
Using (3.19) we can write

$$\begin{pmatrix} \hat{\xi}_1(x,s) \\ \hat{\xi}_2(x,s) \end{pmatrix} = \underbrace{\Phi(x,s)\begin{pmatrix} 1 & 0 \\ -\frac{\phi_{21}(L,s)}{\phi_{22}(L,s)} & \frac{1}{\phi_{22}(L,s)} \end{pmatrix}}_{\Gamma(x,s)} \begin{pmatrix} \hat{\xi}_1\,(0,s) \\ \hat{\xi}_2\,(L,s) \end{pmatrix}. \tag{3.29}$$

with

$$\gamma_{11}(x,s) = e^{-\frac{x}{\lambda_1}\left(s+\frac{1}{\tau}\right)}, \tag{3.30a}$$

$$\gamma_{12}(x,s) = 0, \tag{3.30b}$$

$$\gamma_{21}(x,s) = \frac{\lambda_1}{\lambda_2}\frac{\alpha}{s+\alpha}e^{-\frac{x}{\lambda_1}\left(s+\frac{1}{\tau}\right)}\left(1-e^{-\frac{(L-x)}{\lambda_1\tau\alpha}(s+\alpha)}\right), \tag{3.30c}$$

$$\gamma_{22}(x,s) = e^{\frac{s(L-x)}{\lambda_2}}. \tag{3.30d}$$

Note that equation (3.29) corresponds to a closed form solution of our initial system, written in spectral form.

For low frequencies ($|s| \ll |\alpha|$), $\gamma_{21}(x,s) \simeq \frac{\lambda_1}{\lambda_2}e^{-\frac{x}{\lambda_1}\left(s+\frac{1}{\tau}\right)}\left(1-e^{-\frac{L-x}{\lambda_1\tau}}\right)$ is the combination of a gain, a distributed delay with propagation speed λ_1, and two distributed gains with characteristic distance $\lambda_1\tau$ that cancel out for $x = L$. The resulting bode plot is presented below in Figure 3.5.

Transfer functions for physical variables in congested regime

In congested regime, the boundary conditions used to control the system are $\hat{\xi}_1(0,\cdot)$ and $\hat{\xi}_2(L,\cdot)$. By linearity of the Laplace transform $\hat{\xi}_1(0,s) = \frac{\rho^*\lambda_2}{\lambda_1-\lambda_2}\hat{v}(0,s) + \hat{q}(0,s)$. Therefore, as $\hat{\xi}_2(0,s) = \gamma_{21}(0,s)\hat{\xi}_1(0,s) + \gamma_{22}(0,s)\hat{\xi}_2(L,s)$, we get $\hat{\xi}_1(0,s) = \frac{1}{d(s)}\hat{q}(0,s) + \frac{n(s)}{d(s)}\hat{v}(L,s)$ where $d(s) = 1 - \frac{\lambda_2}{\lambda_1}\gamma_{21}(0,s)$ and $n(s) = \frac{\rho^*\lambda_2}{\lambda_1-\lambda_2}\gamma_{22}(0,s)$. The (v,q) system has only two degrees of freedom. Therefore we consider that the only inputs to the system are $q(0,\cdot)$ and $v(L,\cdot)$. $v(0,\cdot)$ is then completely determined and can be interpreted as an output of the system. The corresponding transfer equation is

$$\begin{pmatrix}\hat{v}(x,s)\\ \hat{q}(x,s)\end{pmatrix} = \underbrace{R^{-1}\Gamma(x,s)\begin{pmatrix}\frac{n(s)}{d(s)} & \frac{1}{d(s)}\\ \frac{\rho^*\lambda_1}{\lambda_1-\lambda_2} & 0\end{pmatrix}}_{\Theta(x,s)}\begin{pmatrix}\hat{v}(L,s)\\ \hat{q}(0,s)\end{pmatrix} \tag{3.31}$$

where

$$\theta_{11}(x,s) = \frac{\alpha e^{-\frac{x}{\tau\lambda_1}}e^{-\frac{s}{\lambda_1}\left(x-L\frac{\lambda_1}{\lambda_2}\right)} + se^{-\frac{s}{\lambda_2}(x-L)}}{s+\alpha e^{-\frac{L}{\tau\lambda_1}}e^{-\frac{sL}{\lambda_1}\left(1-\frac{\lambda_1}{\lambda_2}\right)}}, \tag{3.32a}$$

$$\theta_{12}(x,s) = \frac{e^{-\frac{L}{\tau\lambda_1}}e^{-\frac{s}{\lambda_2}\left(x-L\left(1-\frac{\lambda_2}{\lambda_1}\right)\right)} - e^{-\frac{x}{\tau\lambda_1}}e^{-\frac{sx}{\lambda_1}}}{\rho^*\tau\left(s+\alpha e^{-\frac{L}{\tau\lambda_1}}e^{-\frac{sL}{\lambda_1}\left(1-\frac{\lambda_1}{\lambda_2}\right)}\right)}, \tag{3.32b}$$

$$\theta_{21}(x,s) = \rho^*\tau\alpha s\frac{e^{-\frac{s(x-L)}{\lambda_2}} - e^{-\frac{x}{\tau\lambda_1}}e^{-\frac{s}{\lambda_1}\left(x-L\frac{\lambda_1}{\lambda_2}\right)}}{s+\alpha e^{-\frac{L}{\tau\lambda_1}}e^{-\frac{sL}{\lambda_1}\left(1-\frac{\lambda_1}{\lambda_2}\right)}}, \tag{3.32c}$$

$$\theta_{22}(x,s) = \frac{\alpha e^{-\frac{L}{\tau\lambda_1}}e^{-\frac{s}{\lambda_2}\left(x-L\left(1-\frac{\lambda_2}{\lambda_1}\right)\right)} + se^{-\frac{x}{\tau\lambda_1}}e^{-\frac{sx}{\lambda_1}}}{s+\alpha e^{-\frac{L}{\tau\lambda_1}}e^{-\frac{sL}{\lambda_1}\left(1-\frac{\lambda_1}{\lambda_2}\right)}}. \tag{3.32d}$$

Low frequency approximation for physical variables in congested regime

We derive approximate expressions in the frequency domain for the transfer functions above when $|s| \ll |\alpha|$:

$$\theta_{11}(x,s) \simeq e^{\frac{s(L-x)}{\lambda_2}} e^{\frac{L-x}{\tau\lambda_1}}, \tag{3.33a}$$

$$\theta_{12}(x,s) \simeq \frac{1}{\rho^*\tau\alpha} e^{-\frac{sx}{\lambda_1}} \left(1 - e^{\frac{L-x}{\tau\lambda_1}}\right), \tag{3.33b}$$

$$\theta_{21}(x,s) \simeq s\rho^*\tau e^{\frac{s(L-x)}{\lambda_2}} e^{\frac{L}{\tau\lambda_1}} \left(1 - e^{-\frac{x}{\tau\lambda_1}}\right), \tag{3.33c}$$

$$\theta_{22}(x,s) \simeq e^{-\frac{sx}{\lambda_1}}. \tag{3.33d}$$

With such expressions, interpreting the approximate transfer functions in low frequencies becomes fairly easy. The transfer function θ_{11} (resp. θ_{12}) appears as the combination of a distributed delay with propagation speed $-\lambda_2$ (resp. λ_1) and a distributed gain (resp. attenuation) with characteristic distance $\lambda_1\tau$. The structure of θ_{21} is similar to that of θ_{11} although it features a derivator component. Once simplified, θ_{22} corresponds to a distributed delay with propagation speed λ_1. Hypothetical poles are not active in the range of low frequencies that will be considered in our traffic flow modeling applications. Once more we can appreciate how these approximate expression help better understand the simple elements that account for the physical dynamics of the ARZ model.

Bode plots for congested regime

We use the same fundamental diagram as in the free-flow case. However the linearization point, $\rho^* = 0.08$ veh/m, corresponds to the congested region of the Greeshields diagram. We show the distributed Bode plots for the Riemann invariants in Figure 3.5 and for the physical variables in Figure . In that case, $\alpha = 0.05$ Hz, which does correspond to a reasonable characteristic frequency for traffic modeling applications.

Similarly to the free-flow case, for high frequencies ($w \gg 2\pi\frac{\lambda_1\tau\alpha}{L} = 0.13$ Hz) near zero values appearing with spatial periodicity $\frac{2\pi}{w}\lambda_1\tau\alpha$ almost cancel out γ_{21}, θ_{12}, and θ_{21}. Such points only appear as irregularities in the Bode plots because the gain is computed on a logarithmic scale.

Poles and BIBO stability of the system

In order to practically assess the presence of poles, numerical search for roots of the denominator of the transfer functions has been conducted thanks to standard equation solvers. Once more $-\alpha$ is a solution and another one was found at $s = -0.0018$. They are both negative reals and therefore cannot make the system unstable. Although the solvers could have detected poles with a non zero imaginary part, none has been found. Holistic search for other poles should be conducted but is out of the scope of this chapter.

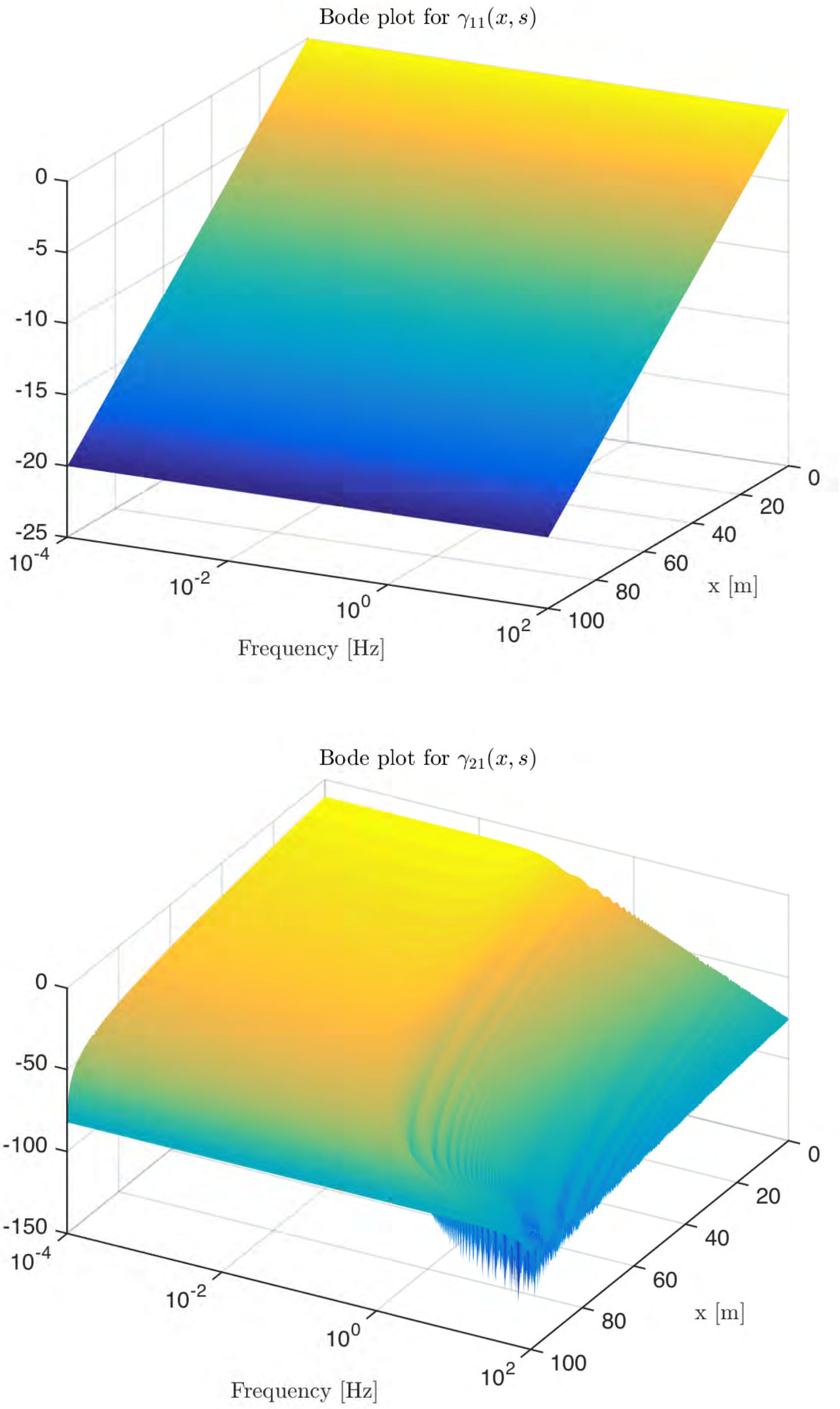
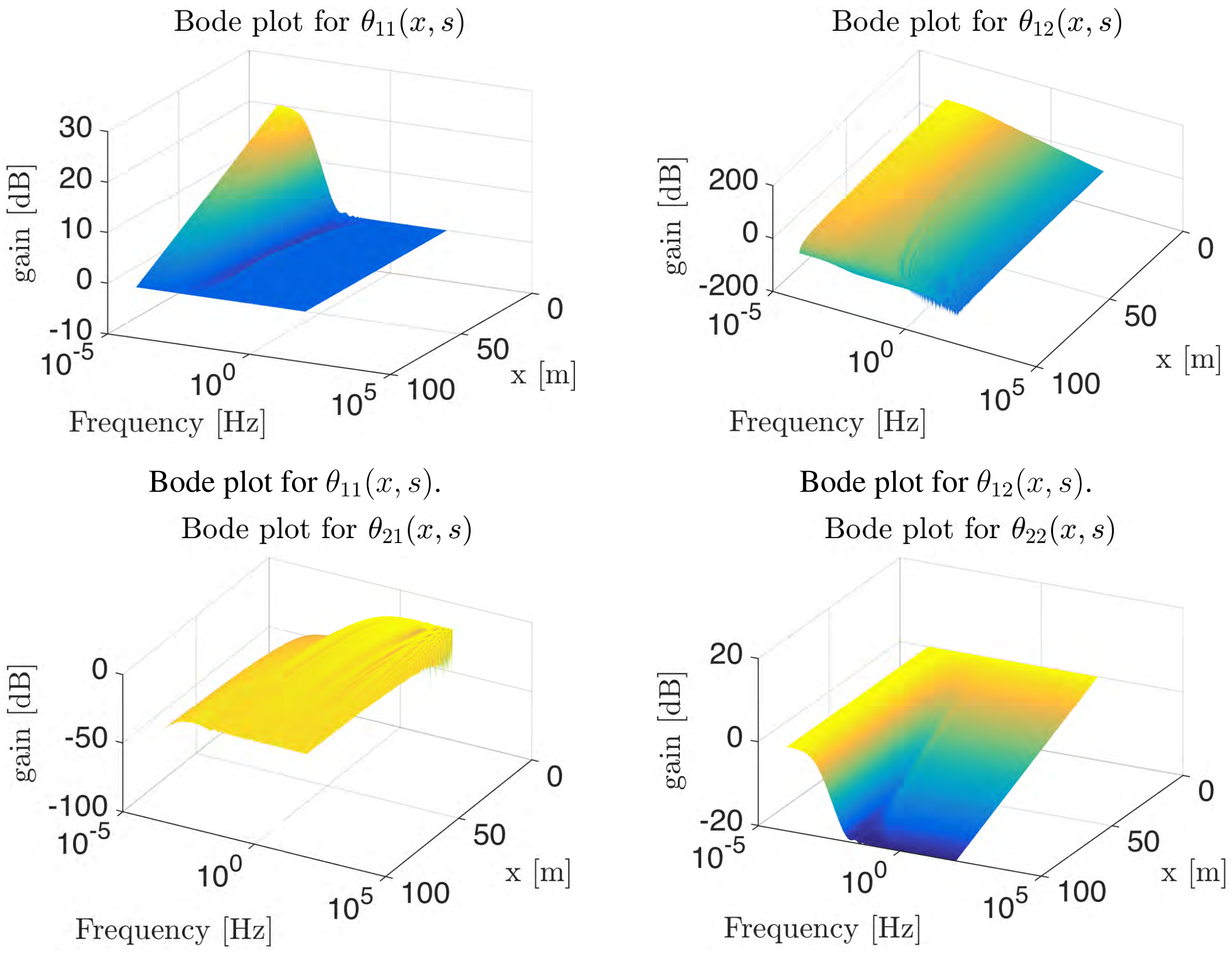


Figure 3.5: Spatial magnitude Bode plots for Riemann invariants in congested regime ($|\alpha|$ =0.05Hz)



Bode plot for $\theta_{11}(x,s)$.

Bode plot for $\theta_{12}(x,s)$.

Bode plot for $\theta_{21}(x,s)$.

Bode plot for $\theta_{22}(x,s)$.

Figure 3.6: Spatial magnitude Bode plots for physical variables in congested regime ($|\alpha| = 0.05$ Hz)

The eigenvalues of matrix $\Theta(x, s)$ are identical to that of $\Gamma(x, s)$. The latter is lower triangular and its eigen values also have a module bounded by 1. The congested regime system is therefore also BIBO stable as the H_∞ norm of matrix $\Theta(x, s)$ is bounded by 1.

Findings and conclusion from the theoretical study

The numerical experiments above have validated the accuracy of the linearized model and highlighted several of its core properties.

The TFN delineates two regimes: congested for $F > 1$ and free-flowing for $F < 1$. This classification, and the resulting stability result legitimize the use of linearization about a nominal point in the stable region.

The assessment of convective instability in the free flow regime is of course applicable to this specific model (other models such as [207] might lead to other conclusions, and all need to be checked against experimental data). Here, exponential growth of the linearization error only occurs in a conic region of the $[0, T] \times [0, L]$ domain where convective instability travels along the characteristics.

The absolute value of the term $\alpha = -\frac{\lambda_2}{\tau(\lambda_1 - \lambda_2)}$ is a characteristic frequency of the system. It delineates the low frequency domain in which approximate expressions help decompose the transfer functions in simple gain and delay components. In the spectral domain, λ_1 and λ_2 appear as information propagation speeds in distributed delay elements while $\tau\lambda_1$ acts as the characteristic distance of distributed gain components.

3.3 Numerical validation

In this section we demonstrate the ability of the linearized ARZ equations to model the various nonlinear dynamics around a nominal operation point and compare the prediction of the model with actual flow and velocity data gathered from the well-known NGSIM data set.

Data source: NGSIM trajectories

We use the NSGIM trajectory data set for a section of the US-101 highway. The set gathers trajectories of vehicles sampled with a 10 Hz frequency thanks to high precision cameras. The data is pre-processed so as to take only cars into account; 45 minutes are recorded on a 650-meter long section with five lanes. The lanes are taken into account when computing the lineic density of vehicles ρ. A map of the time evolution of speed along the section is given in Figure 3.7. Only a subset of the spatial domain is used due to the presence of ramps, which breaks the homogeneity of the freeway. The viable domain is 200 meters long.

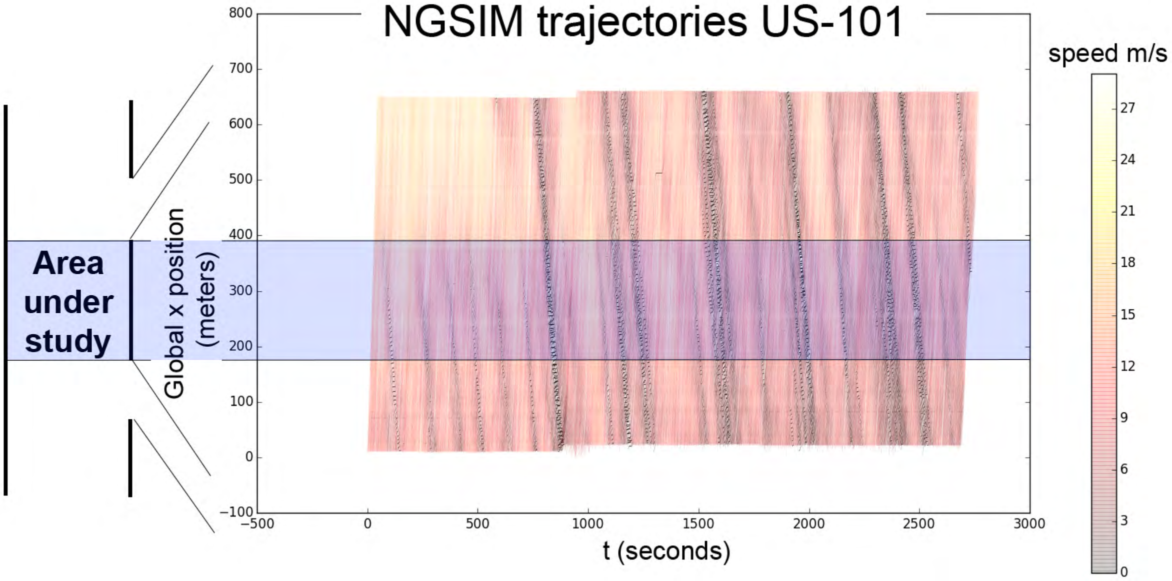


Figure 3.7: NGSIM trajectories. Color represents the measured speed of each vehicle in m/s.

Reconstructing (v, q) maps from NGSIM trajectories

The NGSIM data set does not directly provide the values $v(t, x)$ and $q(t, x)$ in the resolution domain $[0, T] \times [0, L]$. To obtain macroscopic quantities out of the microscopic measurements, we follow the approach devised in [64] and divide the space-time grid into cells $([i\Delta t,\ (i+1)\,\Delta t] \times [j\Delta x,\ (j+1)\Delta x])_{i\in\{1\ldots n_t\}, j\in\{1\ldots n_x\}}$, where n_t and n_x are the number of cells in time and space, respectively. We denote each cell as $\mathrm{bin}_{i,j}$. This operation consists of gathering corresponding data points into cells, then estimating the quantities of interest in each cell, it was for example used in [183].

Within each cell, a specific number of traces, or footprints of a vehicle along its trajectory, are available, and ρ, v, and q are assumed to be constant. We present several formulae to map a set of traces to speed, flow, and density over the space-time grid.

Binning formula for v: Since the speed is assumed to be constant in each cell, a straightforward estimate for the speed is the empirical average. The estimator for v in $\mathrm{bin}_{i,j}$ is

$$\widehat{v}_{i,j} = \mathrm{mean}_{\mathrm{tr}\in\mathrm{bin}_{i,j}}(v(\mathrm{tr})). \tag{3.34}$$

Binning formula for ρ: By definition, the density of $\mathrm{bin}_{i,j}$ is

$$\rho_{i,j} = \frac{\iint_{(t,x)\in[i\Delta t,\,(i+1)\Delta t]\times[j\Delta x,\,(j+1)\Delta x]} \rho(x, t)\mathrm{d}x\mathrm{d}t}{n_{\mathrm{lanes}}\Delta x\Delta t}. \tag{3.35}$$

The position of each vehicle is recorded every 0.1 second. For each cell we count the number of traces and normalize it by the sampling rate. The contribution of a given vehicle to the density of a cell is proportional to the number of traces it has left in the cell. If the speed is assumed to be locally constant, this contribution is proportional to the time this vehicle spends in the cell and is consistent with the conservation of the total number of vehicles across all cells. Then we have the density estimator

$$\widehat{\rho}_{i,j} = \frac{\mathrm{card}(\{\mathrm{tr} \mid \mathrm{tr} \in \mathrm{bin}\})}{n_{\mathrm{lanes}} \Delta x \Delta t \text{ sampling rate}}, \tag{3.36}$$

where $\mathrm{card}(\cdot)$ gives the number of elements in a set, i.e., its cardinal.

Binning formula for q: By definition, $q = \rho v$, so a logical first estimate for q in $\mathrm{bin}_{i,j}$ is

$$\widehat{q}_{i,j} = \widehat{v}_{i,j} \widehat{\rho}_{i,j}. \tag{3.37}$$

We can also approximate the flux through $\mathrm{bin}_{i,j}$ with a simple counting method. If a vehicle crosses spatial coordinate $(j+1)\,\Delta x$ between times $i\Delta t$ and $(i+1)\,\Delta t$, then it leaves a trace in both $\mathrm{bin}_{i,j}$ and $\mathrm{bin}_{i,j+1}$. Counting these vehicles and normalizing by the duration Δt gives the estimator

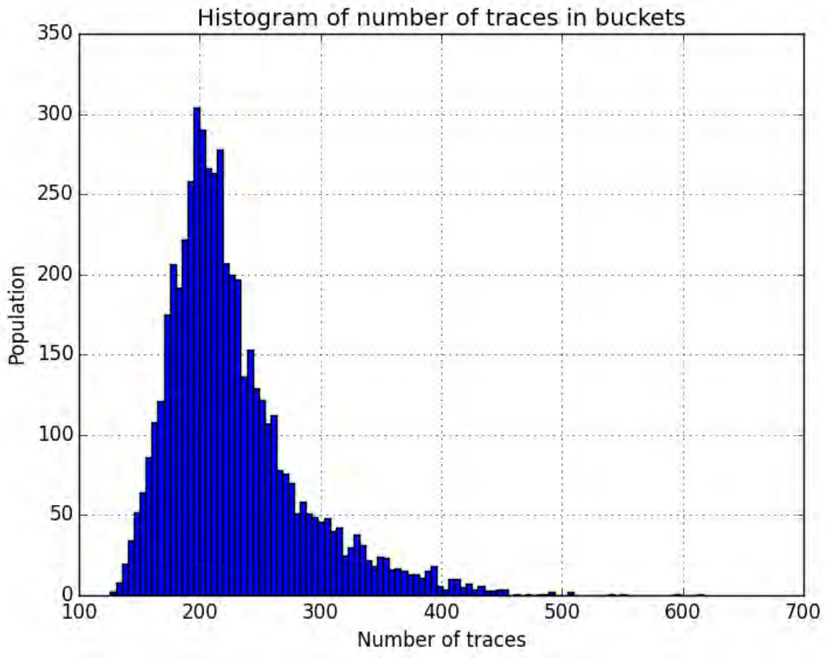
$$\widehat{q}_{i,j}^{\mathrm{count}} = \frac{\text{card}\left(\{\text{id}\,(\text{tr}) \mid \text{tr} \in \mathrm{bin}_{i,j}\} \cap \{\text{id}\,(\text{tr}) \mid \text{tr} \in \mathrm{bin}_{i,j+1}\}\right)}{n_{\mathrm{lanes}} \Delta t}, \tag{3.38}$$

where id$(\cdot)$ gives the identification number of a vehicle.

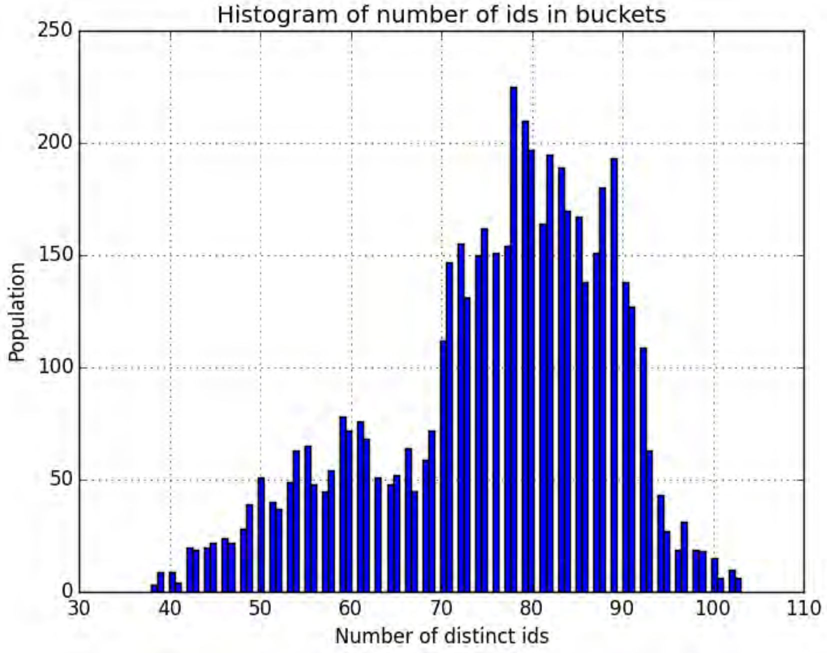
Choosing the number of bins

As the estimation formulae above rely on averaging, having a comfortable number of points in each bin provides more stable estimates. It is worth mentioning that usual central limit theorem based reasoning for convergence of such estimates is flawed as several samples may correspond to the same vehicle or interacting vehicles, violating the independence assumption of the theorem. Proving the convergence of the estimates above lies beyond the scope of this chapter. As a rule of thumb we choose a discretization that guarantees that most bins will host more than 100 traces. This is achieved with a 80×80 grid where the 10^{th} percentile of the number of traces in a given bin is 170. Such a grid also yields a 10^{th} percentile of 56 distinct vehicles per bin. The histograms of number of traces and vehicle per cell are given in Figure 3.8.

While our goal here is not to present theoretical proofs of the convergence of the binned estimators for (v, ρ, q), it is nonetheless possible to check that the procedure is coherent. Two estimators are provided for q that use radically different techniques: the first relies on the average measured speed and the number of traces in a bin, while the other relies on counting vehicles transiting from a cell to another. Figure 3.9 show that the scatter plot of $\widehat{q}_{i,j}^{\mathrm{count}}$ plotted against $\widehat{q}_{i,j}$ coincides nicely with the line $y = x$, validating the overall binning and estimation procedure above.



Histogram of number of traces per cell.



Histogram of number of distinct vehicles per cell.

Figure 3.8: Experimental justification for a 80×80 cell discretization grid for NGSIM data.

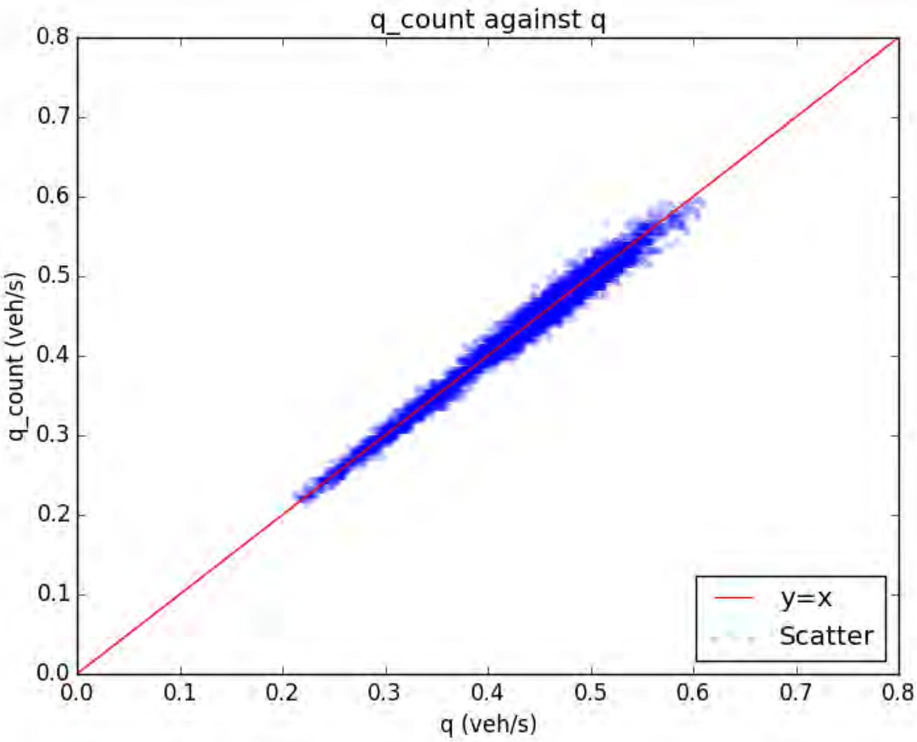


Figure 3.9: Sanity check for the estimation procedure. $\widehat{q}_{i,j}^{\text{count}}$ is plotted against $\widehat{q}_{i,j}$ across the grid of bins.

Estimated values for (v, q)

To check how well the linearized ARZ model fits an actual dataset, we chose a bounded domain and compare the theoretical solution given by the second-order model and the observed data. Again we focus on the variables v and q. Using the estimation procedure above, we compute fundamental diagrams from which we estimate the eigenvalues λ_1 and λ_2. To calibrate the relaxation time τ, we analyze the errors of predicted values of v and q for various τ. The resulting maps of both the predicted and observed values highlight phenomena that the linearized model can and cannot account for.

Maps. The estimates $\widehat{v}_{i,j}$, $\widehat{\rho}_{i,j}$, $\widehat{q}_{i,j}$, and $\widehat{q}_{i,j}^{\text{count}}$ are plotted on the discretized grid in Figure 3.11. Note that $\widehat{q}$ and $\widehat{q}^{\text{count}}$ give extremely similar results, so we may use $\widehat{q}^{\text{count}}$ from this point on. Damped oscillations and smoothly decaying values along characteristic lines are the main characteristic the practical implementation of the model should feature.

Fundamental diagrams. From the estimated values we can easily compute the fundamental diagram. We use the fundamental diagrams to calibrate the model parameters. Though the dataset used is dense, it covers only a small region of time and space. Thus, its small size is a potential flaw in our model parameter calibration as it is certain that our measurements are highly correlated.

This seems to be confirmed by the fact that the fundamental diagrams below correspond only to the congested regime.

Calibration of λ_1 and λ_2, linearization point. In Section 3.1, we found that λ_1 is exactly v^* and λ_2 is the slope of the fundamental diagram at v^*. Thus to calibrate the eigenvalues we must find the linearization point. We estimate the linearization point using the Ordinary Least Squares method. Note the dataset used corresponds only to the congested regime and the fundamental diagram is almost affine. The estimator, $\widehat{\lambda}_1 = \widehat{v}^*$ is chosen as the empirical mean of $\widehat{v}_{i,j}$. To estimate λ_2, we fit a linear model $\widehat{q}^{\text{count}} = b_1 \widehat{\rho} + b_0 + \varepsilon$, where ε represents the noise in the model that would ideally be centered, homoschedastic, and uncorrelated but is not practically. Then $\widehat{\lambda}_2 = \widehat{b}_1$ and we take $\widehat{q}^*$ as the empirical average of $\widehat{q}^{\text{count}}$. The ratio of $\widehat{q}^*$ and $\widehat{v}^*$ gives the estimate $\widehat{\rho}^*$. Provided each estimator is convergent, the continuity of the functional $(x, y) \to \frac{x}{y}$ on its domain guarantees the convergence of $\widehat{\rho}^*$. The empirical results are presented in Figure 3.10. The determination coefficient is poor but can be improved by filtering out outliers and gathering more data. Future work should include improving the quality of the estimation. Significance tests for the coefficients of the linear model are not presented. The assumptions they rely on about the linear dependency between $\widehat{q}$ and $\widehat{v}$ are clearly not respected here as the noise is auto-correlated. Further work should also turn this rather heuristic method for estimating parameters into a fully justified statistical procedure. Note that the goal of the present chapter is to provide a new model and corresponding spectral analysis, which we want to illustrate with state of the art data. Thus, development of statistical methods to handle this data is out of the scope of the present investigation.

Verification of the spectral form

In this section we demonstrate the performance of the spectral form as a prediction tool using the time domain responses derived from the transfer functions and FFT. Since we are working with a linearized system, we can decompose boundary conditions then add predicted values inside the domain $[0, T] \times [0, L]$. Fourier decomposition of boundary conditions is here extremely accurate as the median relative errors for the interpolation of the values of $\xi_1\,(x = 0, \cdot)$ and $\xi_2\,(x = L, \cdot)$ are respectively 2% and 3%.

Simulated maps. Since the spectral form presents information in the diagonalized basis, we need a conversion before we can compare the simulated results to the values estimated from the dataset. To make a comparison in the diagonalized basis, we first compute the estimated deviations from the equilibrium $\widehat{\widetilde{v}}_{i,j} = \widehat{v}_{i,j} - \widehat{v}^*$ and $\widehat{\widetilde{q}}_{i,j} = \widehat{q}_{i,j} - \widehat{q}^*$. Then the estimates for ξ_1 and ξ_2 are given by $\widehat{\xi}_{1_{i,j}} = \frac{\widehat{\rho}^* \widehat{\lambda}_2}{\widehat{\lambda}_1 - \widehat{\lambda}_2} \widehat{\widetilde{v}}_{i,j} + \widehat{\widetilde{q}}_{i,j}$ and $\widehat{\xi}_{2_{i,j}} = \frac{\widehat{\rho}^* \widehat{\lambda}_1}{\widehat{\lambda}_1 - \widehat{\lambda}_2} \widehat{\widetilde{v}}_{i,j}$. To compare the physical variables, we compute the velocity and flow predictions by inverting (3.15): $\widetilde{q} = \xi_1 - \frac{\lambda_1}{\lambda_2} \xi_2$, $\widetilde{v} = \frac{\lambda_1 - \lambda_2}{\rho^* \lambda_1} \xi_2$.

Figure 3.11 shows important qualitative properties of the model. As expected, the model generally predicts with very good accuracy the decay of all quantities along their characteristic lines, a realistic feature that cannot be paralleled by first-order models. The general quality of the fit is rather good with most of the error on v and q in a 20% range of the data's amplitude between minimum and maximum values. Furthermore the linearized second-order model manages to capture oscillations observed on the boundary and account for their decay accurately.

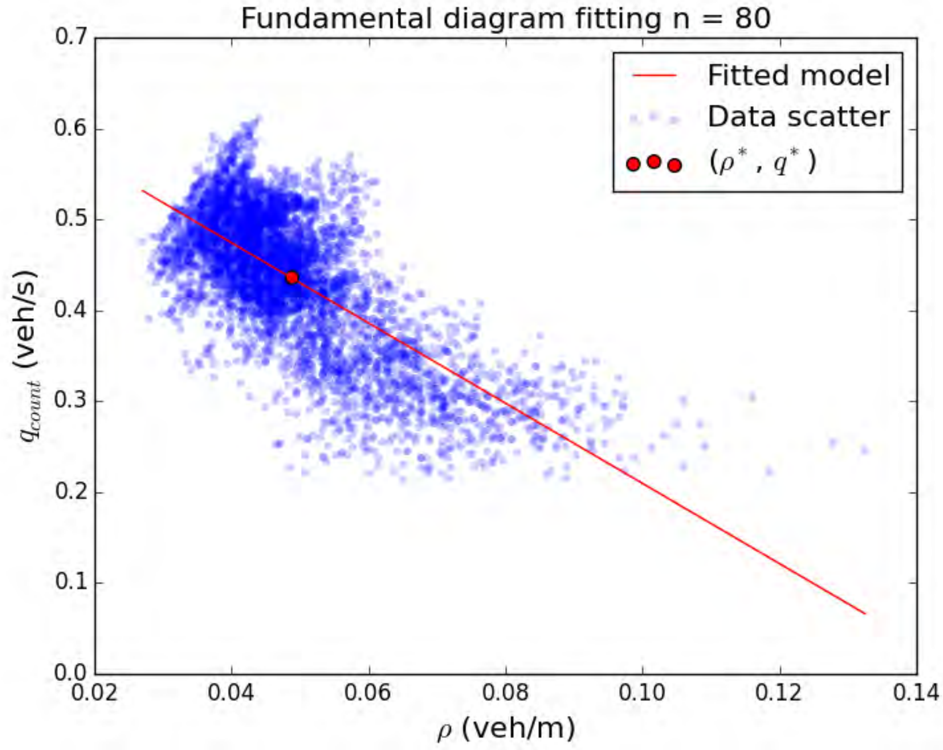


Figure 3.10: Calibration of λ_1 and λ_2. The circle denotes the linearization point. The affine model used to estimate λ_2 and the linearization point is also plotted. The estimates are: $\widehat{\lambda}_1 = 8.96$ m/s, $\widehat{\lambda}_2 = -4.37$ m/s, $\widehat{\rho}^* = 0.049$ veh/m, $\widehat{v}^* = 8.96$ m/s, $\widehat{q}^* = 0.44$ veh/s, with $r^2 = 0.48$. The characteristic frequency of the system is $\widehat{\alpha} = 8.37 \times 10^{-3}$ Hz. Its order of magnitude does correspond to practical traffic flow modeling.

Calibration of τ For each τ we compute the *mean absolute error* (MAE), or the average difference in absolute value between simulated and predicted values for each discretization cell. Since the quantities v and q are not physically homogeneous, it is not sensible to aggregate the errors over these quantities. However, ξ_1 and ξ_2 are both expressed in veh/s. Summing their MAE gives a reliable uni-dimensional index of the quality of the fit with respect to τ. This quantity is computed for different values of τ ranging from 5 to 80 seconds. The value offering the best fit is $\tau^* = 39.18$ s.

Findings and conclusion from the numerical experiments

The numerical experiments above have validated the accuracy of the linearized model and highlighted several of its core properties.

The numerical experiments above show that the linearized ARZ model is capable of reproducing

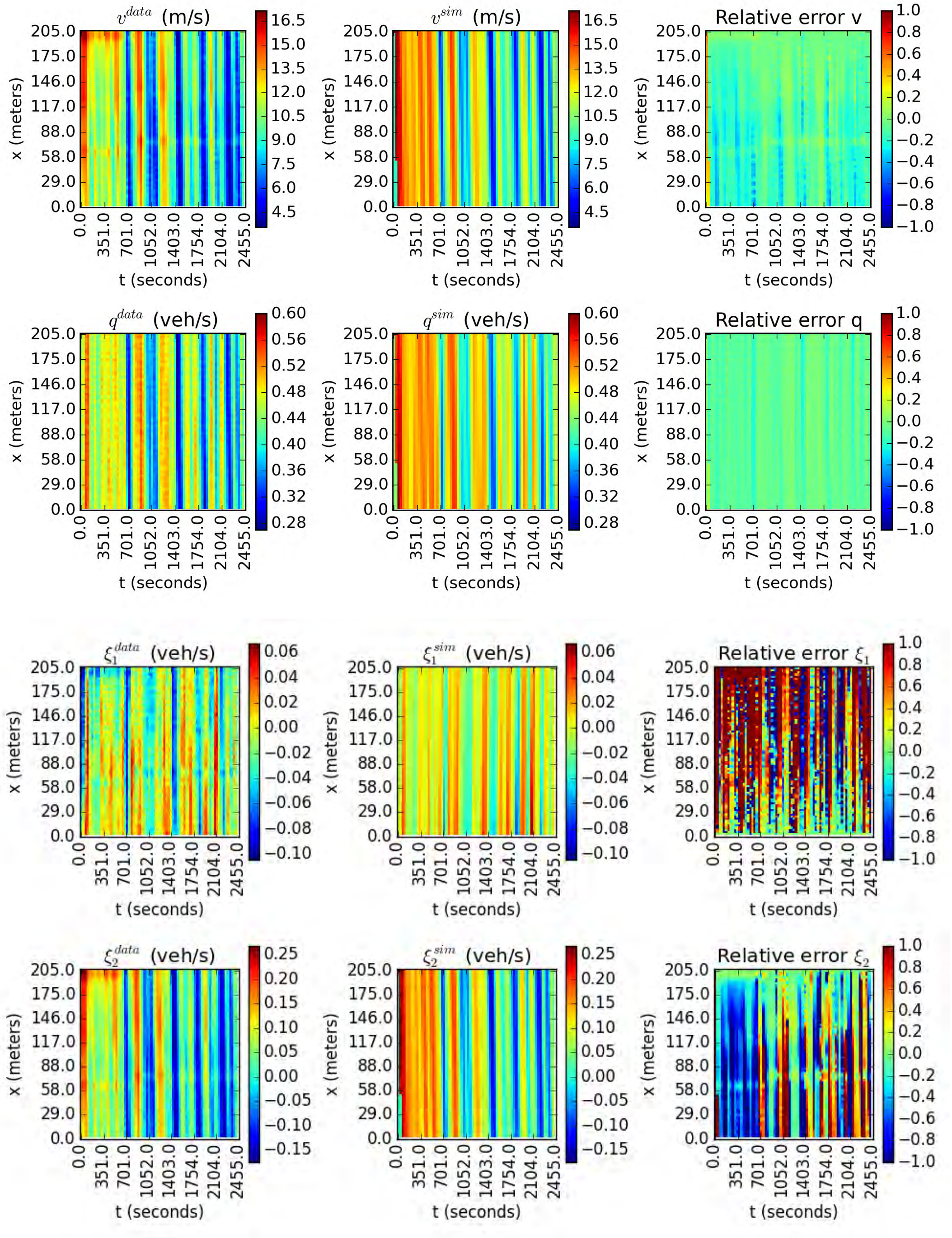
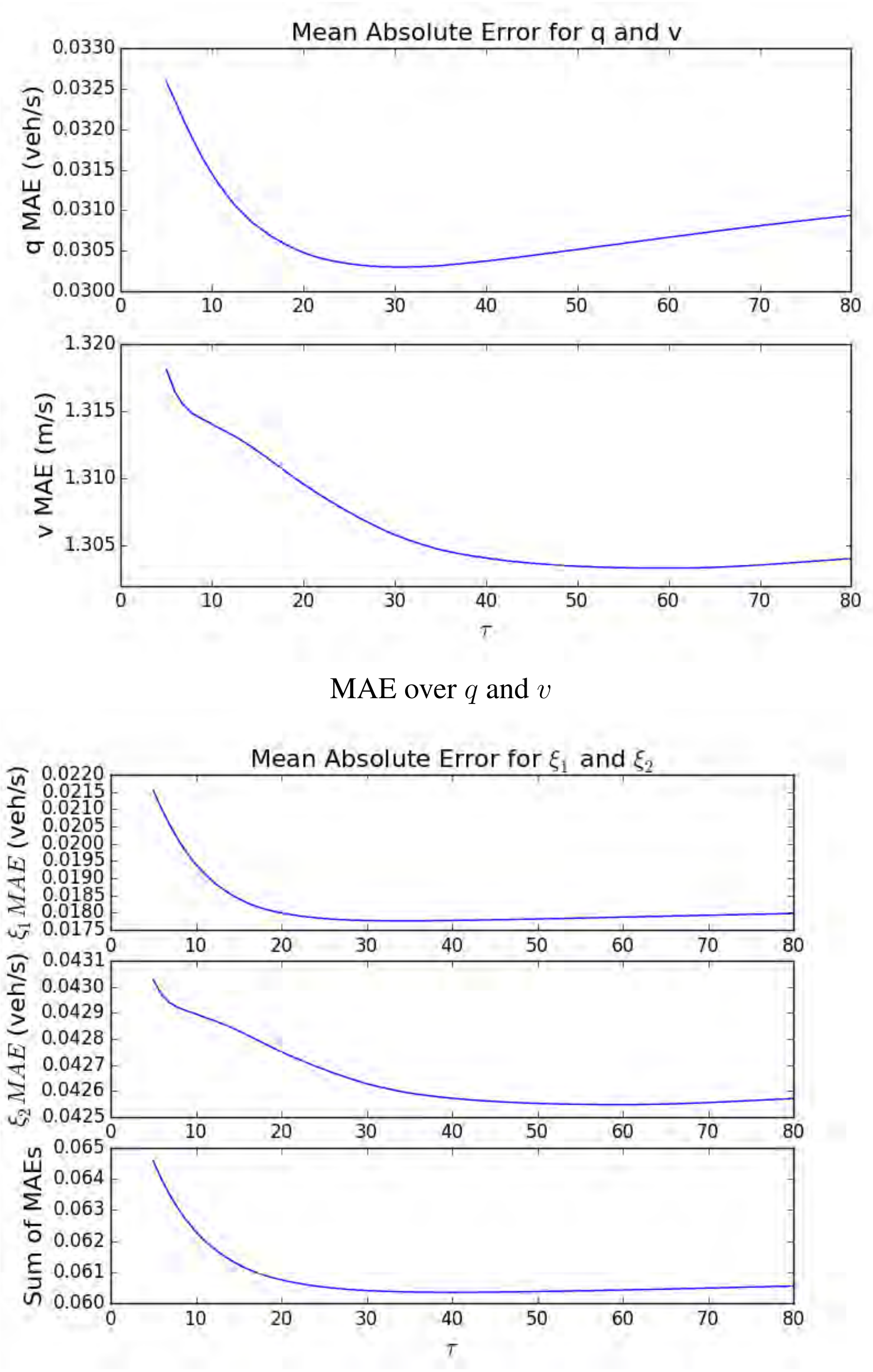


Figure 3.11: Data versus predicted. Top Figure: (v, q) domain, top row is v, bottom row is q. Bottom: (ξ_1, ξ_2) domain, top row is ξ_1, bottom row is ξ_2. First column: data. Middle column: predictions. Third column: error (difference between prediction and data).



MAE over q and v

MAE over ξ_1 and ξ_2 and sum of MAEs.

Figure 3.12: Calibration of τ, one minimizes the sum of MAE over ξ_1 and ξ_2.

NGSIM data accurately for a homogeneous segment of the US-101 freeway. Oscillations are accounted for as well as their damping delay.

The spectral approach provided here supports a solution to the underlying traffic flow model. In other words, the contribution of the work is to show that the model can support oscillatory behavior (through periodic solutions). This is the main difference with purely data driven approaches such as [249] for example.

3.4 Conclusion of the chapter

As the full nonlinear ARZ equations have no known closed form solutions in the general case, they are difficult to analyze. The linearized equations enable the use of spectral methods presented here, allowing for elegantly simple yet powerful analysis tools relying on explicit solutions. These equations are diagonalized, and solved explicitly using a spectral representation (distributed transfer function). Using this approximation, we are able to analyze them around a nominal flow and characterize the oscillatory behavior of the solution. The linearized model is able to capture important features of the flow which first order models cannot.

With the linearized ARZ model, we were also able to define the Traffic Froude Number F. This quantity is computed using the eigenvalues of the system and characterizes the flow regime of the road section under consideration.

We found the eigenvalues of the linearized system to be constant, prompting the definition of the Traffic Froude Number (TFN) which indicates the flow regime of the road section under consideration. Note that the Froude number inequalities delineating the supercritical and subcritical regimes in hydronamics are reversed for our definition of the TFN.

Considering the transfer function of the linearized system of equations delineates the conditions for stability of the approximation about the equilibrium. The time domain responses we derive show that the system is unstable when one of the eigenvalues is negative. In the free-flow regime, $F < 1$, values of flow and speed increase exponentially in a conic region of space and time and the system leaves the linear regime, while in the congested regime, $F > 1$, oscillations decrease. In the latter case, the system remains in the linear regime and oscillations on boundary conditions are damped with an exponential rate along the characteristic lines. Thus, the TFN is also an indicator of convective stability.

The behavior predicted in congested regime for traffic does not present shocks and Fourier spectral analysis cannot account for more nonlinear and non-smooth behavior as well as wavelet transforms. However, our spectral domain study paves the way to applying standard linear system control theory to traffic, with a linearized second model that is empirically reliable in terms of reproducing actual data. Controller design based on the spectral framework presented here becomes immediate as shown in [143] for a similar model involved in the control of water flows in canals.

The present analysis also complements the wavelet-based approach used in [249]. Fourier transform based analysis is a powerful tool in our numerical experiments that does not require any CFL-like condition for stability. The linearity of the system clearly predicts smoother behavior than what is found with wavelet decomposition, and as we have shown, the resulting macroscopic predictions are fairly accurate.

The Laplace domain spectral representation we present here lends itself naturally to control. Such a dual representation of the system has also enabled an elegant and simple analysis although the system of interest was defined in continuous time and had an infinite number of spatial dimensions.

Laplace duality made the properties of interest evident and the analysis of the system simple. Such elegance contrasts with the actual complexity of the dynamics we study. The ARZ set of equations constitutes, even in its linearized form, an infinite-dimensional system defined in continuous time and space.

Along the spatial dimension, we take into account a continuous of interacting elements evolving with different states. In other words, the problem of interest is concerned with a heterogenous continuum of related PDEs defined in continuous time.

Determining conditions of stability for such an infinitely complex system has practical repercussions for millions of drivers who commute everyday on freeways as the formation of traveling waves of braking vehicles can lead to crashes. Such “jamitons” [72] otherwise referred to as convective instability [226] occur at a critical regime when enough vehicles are present on the road to create massive crashes and yet speed is still high. The density of vehicles at the critical regime is still not yet causing an important slow down.

With an insightful choice of representation, the problem was turned into a homogenous set of equations involving Laplace transforms in the time domain parametrized by their spatial location. Laplace transforms were a good fit for this control problem because of their ability to account for causal dynamics and simplify time invariant operators.

As we have seen in the previous chapter, Fourier transforms were well adapted to represent asynchronously sampled streams of data in a parsimonious and statistically sound manner under an assumption of second order stationarity.

The next chapter will introduce another duality: convex duality. The corresponding Fenchel-Legendre transform will help us turn large discrete collections of control problems involving heterogenous constraints into a unifying and data efficient representation. A major advantage will stem from the design of a scalable distributed control scheme that can thoroughly be analyzed and for which strong guarantees can be found.

Chapter 4

Distributed communication avoiding control schemes

The previous chapter used the Aw-Rascle-Zhang Partial Differential equation to show how a change of representation (i.e. using the spectral domain) enables the seamless analysis of an infinite-dimensional system with potential control applications.

We now focus on creating methods that instantiate distributed unifying representations of heterogeneous large scale control problems.

This chapter shows how leveraging the convex geometry of a broad class of collaborative block-constrained control problems yields a scalable solution provided a dual representation is employed. We first present the general constrained optimization problem we consider. The problem is then solved in its dual form. We show how deterministic exact gradient calculations converge to an optimum while only communicating small amounts of data. Furthermore, we prove that even if the distributed optimization algorithm we formulate is subject to random perturbations the resulting stochastic gradient descent converges towards an optimal solution. Beyond communication avoidance, such stochastic perturbations help preserve the privacy of the agents participating in the control scheme. Using convex duality also yields an interpretable Lagrangian price multiplier as a byproduct of the optimization procedure which helps interpret tensions between offer and demand in the systems under study.

For distributed scalable control schemes to be practically usable today, scaling is not the only requirement. The real-world problems we consider involve a Mobility-as-a-Service (MaaS) driver management and smart meters operations, helping smooth out electrical consumption imbalance. Both cases put important information at risk in the form of drivers schedules or household consumption patterns. We prove some privacy preservation properties of the schemes we design in order to encourage individuals to take part in such large collaborative efforts.

The proofs we give compare the rate at which individual data is leaked into the communication medium and the rate at which the cohort of controllers converges to an optimal plan. Such a property shows that the representation we employ as a means to enable scalability is also statistically efficient. Data efficiency appears here in the absence of the need to communicate individual constraints explicitly.

4.1 Challenges with heterogeneous distributed control problems

After formulating a general control method which we will show is relevant to Mobility-as-a-Service and electrical load shaping, we develop an approach to solve distributed control problems subject to heterogenous blockwise-independent constraint sets. The specific aim of this chapter is to enable scalability by preserving communication avoidance in a distributed heterogenous control scheme. We show that a direct consequence of the design choice we make is privacy preservation.

General problem setting

We consider a cyber-physical system consisting of a swarm of agents (i.e. controllers) whose actions are combined to enable collaboration. We assume that the resulting action of the multiple controllers is the linear combination of their individual actions. The combined efforts of the agents are translated into a loss function which we assume is strongly convex. We make further assumptions at the level of the individual agents. We assume that the actions are distributed in a continuous vector space of finite dimensions. The magnitude of an actuation results into energy consumption and/or system wear which leads to the inclusion of strongly convex penalties over the actuations. Finally, the assumption that matters most here lies in the heterogeneity of the system. We assume that the agents' constraints are all unique. Each agent is subject to a particular set of constraints.

In spite of the fact that each agent deals with a particular problem setting because of its own individual constraints, we are looking for a procedure to reformulate the distributed control problem in terms that are commons across agents. The problem of interest and its general structure are presented in Figure 4.1.

Communication avoidance by dual splitting

A salient issue with the convex optimization program that needs to be solved in the chapter is that the number of constraints scales linearly with the number of agents. This aspect makes the complexity of computations cubic in time and memory with respect to the number of controllers for standard convex optimization solvers employing interior point methods [26]. Similar issues hamper any solution based on inverting the matrix of linear constraints that characterizes the problem.

We develop a dual splitting technique which breaks the main control problem into as many subproblems as there are agents involved. We show how this results in a gradient descent based convergence method which requires each controller to solve an individual subproblem at each descent step for the gradient to be collectively computed. We prove that the convergence of the scheme occurs at a linear rate. We also show that the computational burden for each control device is only proportional to the cubed number of constraints of the agent it corresponds to as opposed to that of the whole swarm. A key element is that the only information that needs sharing is a Lagrangian multiplier which mitigates communication needs and thereby enforces scalability.

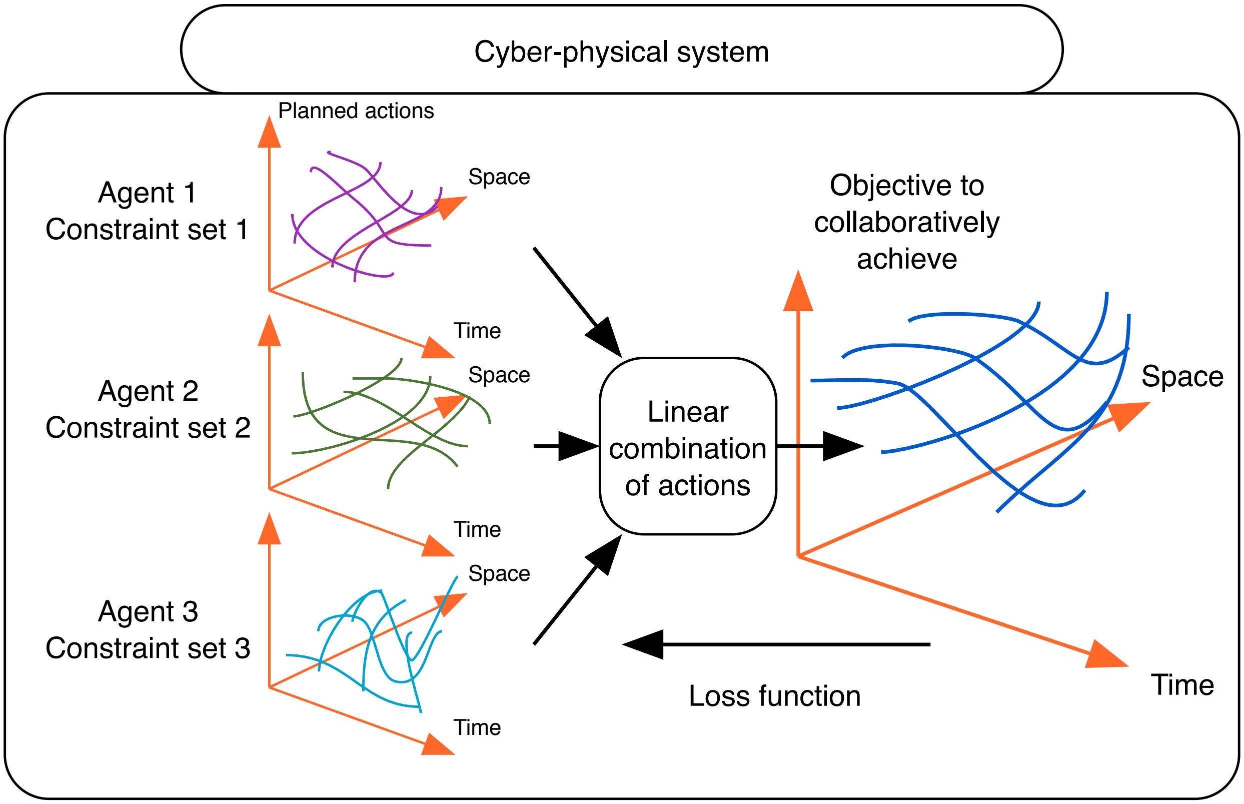


Figure 4.1: General structure of distributed control problem of interest. A swarm of controllers actuates a cyber-physical system. The constraints of each controller are unique and independent. The combined actions of the controllers result in increasing or decreasing a system-wide loss.

The multiplier serves effectively as a common representation entailing information coming from different constraint sets.

Privacy preservation is also therefore a valuable by-product of the method. No agent-level individual constraints need to be shared in our system. We further show how recommended plans of actions can be obfuscated thereby further protecting privacy. The algorithmic contribution relies on techniques different from the famous *Alternating Direction Method of Multipliers* (ADMM) of [27], distributed primal methods [168, 83] or primal-dual methods [33]. Indeed it is tailored to the block-wise independent structure of the constraints and the partitioning of the data across agents.

In the current chapter, we introduce a method that is communication avoiding, scalable, privacy preserving and computationally tractable.

Organization of the chapter

The chapter first delves in the general theory of using convex duality to find a common distributed representation for the heterogenous constraint sets of the agents involved in the system of interest. We design a dual representation that leads naturally to distributed communication avoiding gradient descent. We highlight the modeling assumptions that yield provable convergence rates for the distributed communication algorithm we present which operates on the dual problem. After the first theoretical discussion of the method, we present an application to Mobility-as-a-Service (MaaS), highlight how a convex relaxation corresponds to a realistic assumption which later on allows using a dual splitting technique. We also present an application to the problem of electrical load shaping at the scale of the state of California.

The present chapter is therefore organized as follows. In Section 4.2, a theoretical analysis of the convergence properties of the distributed algorithm we present shows the robustness of the algorithm. In Section 4.3, we first briefly describe the method used to characterize the spatial and temporal structure of demand for mobility as a service rides in New York. While this is not the core focus of the present chapter, it is briefly presented in Section 4.3 since the rest of the section leverages this spatio-temporal structure once it has been characterized. We then formulate a convex optimization problem that matches the probability of presence of drivers with the demand for rides. Numerical experiments confirming these results are shown as a conclusion to Section 4.3. Finally, we showcase in Section 4.4 the generality of the approach by showing that it also directly applies to another high stake distributed control problem: electrical load shaping in California. After having modeled the objective of interest and the impact of actions of fleets of smart meters storing energy in batteries available in electric vehicles, we apply the dual splitting technique we presented. We derive convergence rates in this particular problem setting and confirm the method performs well on numerical experiments.

4.2 Convex optimization for distributed control

Prior to studying two practical instantiations (MaaS and electrical load flattening) we give a general formulation of the collaborative control problem we consider and the assumptions we make about the structure of the heterogeneous constraints controllers are subject to.

The original problem is converted to a homogeneous distributed dual representation which leads to the formulation of gradient based optimization algorithm.

A thorough analysis of the convergence rates of the algorithm we present to solve the convex program is conducted. This analysis is general and takes place for a large class of problems thereby making it useful for practitioners with different objective functions. We show that the scheme is robust to noise being injected in communications. After delineating two privacy preserving scalable algorithms, we prove that as the number of controllers increases, privacy can further be improved. We highlight a crowd obfuscation effect thanks to which an individual attacker cannot learn as fast as the group. We will later confirm the results obtained in this holistic theoretical analysis of the properties of the algorithm with numerical experiments conducted with actual openly accessible data in both the noiseless and noisy settings. Section 4.2 proves that the algorithm we present is robust enough to allow for the obfuscation of the optimal program it converges to. A theoretical comparison of the rate at which individual information is revealed with the rate at which the cohort of controllers converges to optimality highlights that the representation we employ serves a precise purpose. The Lagrangian multiplier updates communicated by the agents provide more information useful to the group than individual information on their constraints. In other words, we show that the homogenous representation we choose to enable scalability is parsimoniously tailored to the purpose of solving the problem at hand. We prove the convergence of this algorithm based on the properties of gradient ascent methods described in [199, 19] which needs no broadcasting of the constraints of the agents as explained later in Figure 4.3.

Background on distributed optimization techniques for control

The approach developed in this chapter is tailored to strongly convex block constrained optimization problems in which the objective is the sum of two terms. The first one is the compound of an averaging function and a strongly convex cost. The second one is a fully decoupled regularization factor. Although constraints are block-wise independent, the averaging term is binding which leads to the use of a dual splitting method in order to solve the problem on a swarm of distributed computing devices. Many techniques have been developed to parallelize convex optimization in the case of separable objectives and binding constraints. The ADMM [26, 83] and graph projection [176] algorithms are two instances of the Douglas-Rachford algorithm [63] which belongs to the larger family of proximal point methods [198]. Other approaches have been developed for splitting similar problems such as Springarn's partial inverse algorithm [213], distributed subgradient descents [168] and the proximal center technique [167]. Closer to the setting under consideration here, the primal-dual approaches developed in [33] and [251] respectively deal with block constrained problems where the objective is known partially to the computation nodes and with optimization over a common constraint set that consists of the intersection of local constraint sets. However efficient and provably fast, these techniques do not leverage the particular structure of the problem under study here. Therefore, we designed a novel dual splitting strategy tailored to the average-based input in the objective. Gradient methods are used to find the optimum of the dual problem. We consider agents should not give out their own information to the rest of the community in order to preserve their privacy and enable scalability through communication avoidance. Convergence rates are calculated for stochastic algorithms that enforce privacy by signal obfuscation. In particular, the proofs below extend previous work on privacy-aware optimization [58, 237, 232]. In this competition to derive estimates of the optimal programs the agents will undertake, we prove that introducing random perturbations in the dual gradient method is much more detrimental to an eavesdropping statistician adversary than it is for the community.

The last observation shows how communication avoidance leads to developing methods featuring strong statistical guarantees.

Primal formulation of a consensus problem

In the current section, we assume that the spatio-temporal structure of demand within the system has been identified (which we conduct later thanks to standard time series analysis methods) as a demand matrix $(d_{t,n})_{t=1\ldots T,n=1\ldots N}$. In the demand matrix t is the temporal indexing of the demand intensity and n its spatial index. Let us assume that the demand matrix has been estimated. Without loss of generality, we can flatten the spatio-temporal demand matrix. The demand intensity is from hereon renumbered as $(d_k)_{k=1\ldots T\times N}$.

The primal optimization program obtained after convex relaxation can now be rewritten as follows:

Generic main primal problem:

$$\min_{\substack{u \\ \text{s.t. } \forall c\in\{1\ldots C\},\, u_c\in\mathcal{C}^c}} \left[\sum_{k=1}^{T\times N} \ell_k\left(d_k - \frac{1}{C}\sum_{c=1}^{C} u_{k,c}\right) + \frac{1}{C}\sum_{c=1}^{C} \mathrm{r}_c\left(u_c\right)\right] \tag{4.1}$$

Penalizations corresponding to failing to match the demand are represented by the family of strongly convex functions $(\ell_k(\cdot))_{k\in\{1\ldots T\times N\}}$ from $\mathbb{R}$ onto $\mathbb{R}$.

Penalizing regularization terms appear as we want the agents to take into account the cost associated to each of their action. Such cost is practically system wear and/or energy consumption. We assume the associated marginal cost is increasing. Therefore, by design, all the elements of the family of regularizing functions $(\mathrm{r}_c(\cdot))_{c\in\{1\ldots N\}}$ from $\mathbb{R}^d$ to $\mathbb{R}$ are strongly convex.

Practical computational issues for naive approaches. Such a problem, with strongly convex objectives and strictly feasible convex constraints can be solved in a generic but non-scalable manner by standard optimization solvers. A key point in the present chapter is to develop specific algorithms which efficiently leverage the structure of problem Eq. (4.1). For general $\ell_k(\cdot)$ and $\mathrm{r}_c(\cdot)$ satisfying the conditions above, standard solver will require too much memory for a single computer to handle as soon as $C \geq 10^3$ and $T \times N \geq 10$. Therefore we want to find a strategy that distributes this problem across several machines. Such scaling properties are of crucial importance in many large cyberphysical systems [109]. Note that to coordinate MaaS dispatch, in numerous cities the driver pool is much larger than 10^3. Similarly, there are many more smart meters in California that can contribute to evening out load imbalance.

Privacy related issues. We also consider that the constraints $u_c \in \mathcal{C}^c$ of each agent c should not be broadcast as would require too much communication. In the distributed setting, such communication costs dominate when considering execution speed [19]. This does not have implications on the formulation (4.1) of the problem but will be essential in the algorithms to solve it. Let u^* be the solution of the minimization problem (4.1). The optimal program $u_n^* \in \mathbb{R}^d$ of each agent also contains sensitive information and has to be difficult for an adversary to estimate (the meaning of this statement will be explicitly defined later in the article). The dual splitting reformulation we propose has the double advantage of offering a scalable method to solve the problem which as a byproduct does not requires the sharing over a network of individual agents' constraints.

Distributed optimization

In this paragraph we introduce additional variables before using a dual reformulation that manages to split the problem with respect to the number of agents. As the number of controllers, C, is the largest scale factor in the problem, it is the most suitable axis for parallelization. We leverage independence between blocks of constraints to render a low-memory, communication avoiding and privacy preserving algorithm.

Introduction of additional variables

We introduce a set of consensus variables entailing the amount of discrepancy there is between the demand $(d_k)_{k=1 \ldots T\times N}$ and the aggregated offer $(\frac{1}{C}\sum_{c=1}^{C} u_{k,c})_{k=1 \ldots T\times N}$

For each $k \in \{1 \ldots d\}$, let $z_k \in \mathbb{R}^d$. The optimization problem now reads

$$\min_{u} \sum_{k=1}^{T\times N} \ell_k\left(z_k\right) + \frac{1}{C}\sum_{c=1}^{C} \mathrm{r}_c\left(u_c\right)$$

$$\text{st}\, \forall c \in \left\{1 \ldots C\right\},\ u_c \in \mathcal{C}_c, \forall k \in \left\{1 \ldots T \times N\right\},\ z_k = d_k - \frac{1}{C}\sum_{c=1}^{C} u_{k,c}$$

We form the Lagrangian with $T \times N$ real dual variables $\left(\lambda_k\right)_{k\in\{1 \ldots T\times N\}}$ corresponding to constraints $\forall k \in \left\{1 \ldots T \times N\right\},\ z_k = d_k - \frac{1}{C}\sum_{c=1}^{C} u_{k,c}$. Slater's conditions hold with the assumptions above (strong convexity, strictly feasible or affine constraints, see [26] for details). Minimization and maximization can therefore be swapped in the Lagrangian. the argument proves that problem (4.1) is equivalent to

$$\max_{\lambda} \min_{u,z} \left[\sum_{k=1}^{T\times N} \ell_k\left(z_k\right) + \lambda_k z_k + \sum_{k=1}^{T\times N} \lambda_k \left(-d_k + \frac{1}{C}\sum_{c=1}^{C} u_{k,c}\right) + \frac{1}{C}\sum_{c=1}^{C} \mathrm{r}_c\left(u_c\right)\right]$$
$$\text{st}\, \lambda \in \mathbb{R}^{T\times N},\ z \in \mathbb{R}^{T\times N},\ \forall c \in \left\{1 \ldots C\right\},\ u_c \in \mathcal{C}^c$$

Block constraints and distribution of min operators

The formulation above solves our need for parallelization across agents as the operator min with respect to each z_k and each u_c. Indeed, z and u are decoupled and the constraints $u_c \in \mathcal{C}^c$ are independent by assumption. The independence of constraints among the agents is consistent with the fact that these are considered heterogenous and private. Now, considering the Fenchel-Legendre transform $\ell_k^*\left(\lambda_k\right) = \sup_{z_k\in\mathbb{R}} l_k\left(z_k\right) + \lambda_k z_k$ of $\ell_k\left(\cdot\right)$, one has

$$\min_{z\in\mathbb{R}^d} \sum_{k=1}^{T\times N} \ell_k\left(z_k\right) + \lambda_k z_k = -\sum_{k=1}^{T\times N} \ell^*\left(-\lambda_k\right).$$

Also, denoting $\Pi_{c=1}^{C}\mathcal{C}^c$ the cartesian product of the constraint sets,

$$\min_{u\in\Pi_{c=1}^{C}\mathcal{C}^c} \frac{1}{C}\sum_{c=1}^{C} \lambda^T u_c + \mathrm{r}_c\left(u_c\right) = \frac{1}{C}\sum_{c=1}^{C} \min_{u_c\in\mathcal{C}^c} \lambda^T u_c + \mathrm{r}\left(u_c\right).$$

The remark above proves that problem (4.1) is equivalent to

$$\max_{\lambda}\left[-\sum_{k=1}^{T\times N}\ell_k^*\left(-\lambda_k\right)-\lambda^T d+\frac{1}{C}\sum_{c=1}^{C}\min_{u_c\in\mathcal{C}_c}\lambda^T u_c+\mathrm{r}_c\left(u_c\right)\right].$$

We can now appreciate how the individual constraint sets can be taken into account independently one from another. In particular, this method will split the memory requirement for a gradient method based numerical resolution in the algorithm presented below (Algorithm 1).

Extended value regularization functions

For each $c\in\{1\dots C\}$, let $\overline{r_c}\left(\cdot\right)$ the extended value function that equals $r_c\left(u_c\right)$ whenever $u_c\in\mathcal{C}_c$ and $+\infty$ otherwise. With the assumptions above, $\overline{r_c(\cdot)}$ is proper, closed and lower semi continuous. It is not differentiable in general but is strongly convex by assumption. Let σ_c its strong convexity constant. Generic convex analysis (see [199] for details) allows us to show that the Fenchel-Legendre transform of $\overline{r_c(\cdot)}$, denoted $\overline{r_c(\cdot)}^*$, is differentiable and has a Lipschitz gradient with constant $\frac{1}{\sigma_c}$. It is also trivially convex. For any $c\in\{1\dots C\}$, the strong convexity assumption on r_c also guarantees uniqueness of $u_c^*\left(\lambda\right)=\operatorname{argmin}_{u_c\in\mathcal{C}_c}\left(\lambda^*\right)^T u_c+\mathrm{r}_c\left(u_c\right)$ where λ^* is the unique solution of problem (4.2) (see [26] for details).

Formulating an optimal price

The problem (4.1) is now equivalent to the unconstrained minimization below.

Dual split reformulation:

$$\min_{\lambda\in\mathbb{R}^{T\times N}}f\left(\lambda\right)=\min_{\lambda\in\mathbb{R}^{T\times N}}\left[\sum_{k=1}^{T\times N}\ell_k^*\left(-\lambda_k\right)+\lambda^T d+\frac{1}{C}\sum_{c=1}^{C}\overline{\mathrm{r}_c}^*\left(-\lambda\right)\right] \tag{4.2}$$

For each $k\in\{1\dots T\times N\}$ we denote L_k the Lipschitz constant of the gradient of $\ell_k(\cdot)$ and m_k the strong concavity constant of the function. As in [199], $\ell_k^*\left(\cdot\right)$ has a Lipschitz gradient with constant $\frac{1}{m_k}$ and is strongly convex with constant $\frac{1}{L_k}$. Therefore, f is strongly convex with constant $m=\sum_{k=1}^{T\times N}\frac{1}{L_k}$ and has a Lipschitz continuous gradient with constant $L=\sum_{k=1}^{T\times N}\frac{1}{m_k}+\frac{1}{C}\sum_{c=1}^{C}\frac{1}{\sigma_c}$. The strong convexity property shows in particular that there is a unique price vector λ^* that synthesizes the information contained in the common objective and the constraints. Indeed, if agent c is given λ^* it is sufficient for it to individually solve $\min_{u_c\in\mathcal{C}_c}\left(\lambda^*\right)^T u_c+\mathrm{r}_c\left(u_c\right)$ in order to retrieve the optimal action u_c^* that contributes to the overall objective best. In particular, this shows that communication bandwidth requirements are lowered as privacy sensitive constraints do not have to be shared with other participants in the system. Also, this problem is much less memory consuming to solve as only one constraint set corresponding to a single agent needs to be considered for any given time. Moreover, for a given value of λ, the sub-problems can be solved independently in parallel. This shows that this reformulation is privacy preserving, is computationally tractable and scales with respect to the number of agents being dispatched thanks to communication avoidance.

Holistic deterministic gradient descent

Gradient descent and momentum methods are both straightforward ways to minimize f in practice. We have $\nabla f(\lambda) = -\sum_{k=1}^{T\times N} \nabla \ell_k^*(-\lambda_k) + d - \frac{1}{C}\sum_{c=1}^{C} \nabla \overline{\mathrm{r}_c}^*(-\lambda)$.

Usual theorems for differentiating maxima of functions (see [18] for details) give, $\forall c \in \{1 \dots C\}$, $\nabla \overline{\mathrm{r}_c}^*(\lambda) = u_c^*(-\lambda)$, therefore

$$\nabla f(\lambda) = -\sum_{k=1}^{T\times N} \nabla \ell_k^*(-\lambda_k) + d - \frac{1}{C}\sum_{c=1}^{C} u_c^*(\lambda). \tag{4.3}$$

From [199], we know $O\left(\log\left(\frac{L}{m}\frac{1}{\varepsilon}\right)\right)$ iterations are sufficient for the distributed gradient based algorithm below (Algorithm 1) to achieve an ε precision in the value of the function we are trying to minimize.

```
Algorithm 1: Holistic distributed gradient descent
  Data: Constraints $(\mathcal{C}_c)_{c\in\{1\dots C\}}$, target $d \in \mathbb{R}^{T\times N}$
  Result: Optimal dual price $(\mathrm{argmin}_{\lambda\in\mathbb{R}^{T\times N}} f(\lambda))$
decide on initial value $\lambda^{(0)} \in \mathbb{R}^{T\times N}$.
for $i \leftarrow 1$ to maximum number of steps do
    broadcast $\lambda^{(i)}$
    compute optimal response $u_c^*\left(\lambda^{(i)}\right)$
    broadcast $u_c^*\left(\lambda^{(i)}\right)$
    gather and compute concatenated $u^*\left(\lambda^{(i)}\right)$
    compute $\lambda^{(i+1)} = \lambda^{(i)} - s^{(i)}\nabla f\left(\lambda^{(i)}\right)$
end
```

Theorem 4.2.1 *Algorithm 1 solves the dual split problem without requiring the controllers to communicate their constraint sets. At any step i, the only communication needed from agent c is $u^*\left(\lambda^{(i)}\right)$, which does not involve the constraint set $\mathcal{C}_c$. Only the local computation of that optimal response by the agent required knowledge of the agent's constraints. It is possible to solve the dual split problem without requiring the agents to communicate their constraint sets.*

Proof 4.2.1 *With regularity and strong convexity assumptions above, gradient descent in algorithm 1 converges at an linear rate towards λ^* and therefore Theorem 4.2 completes the proof.*

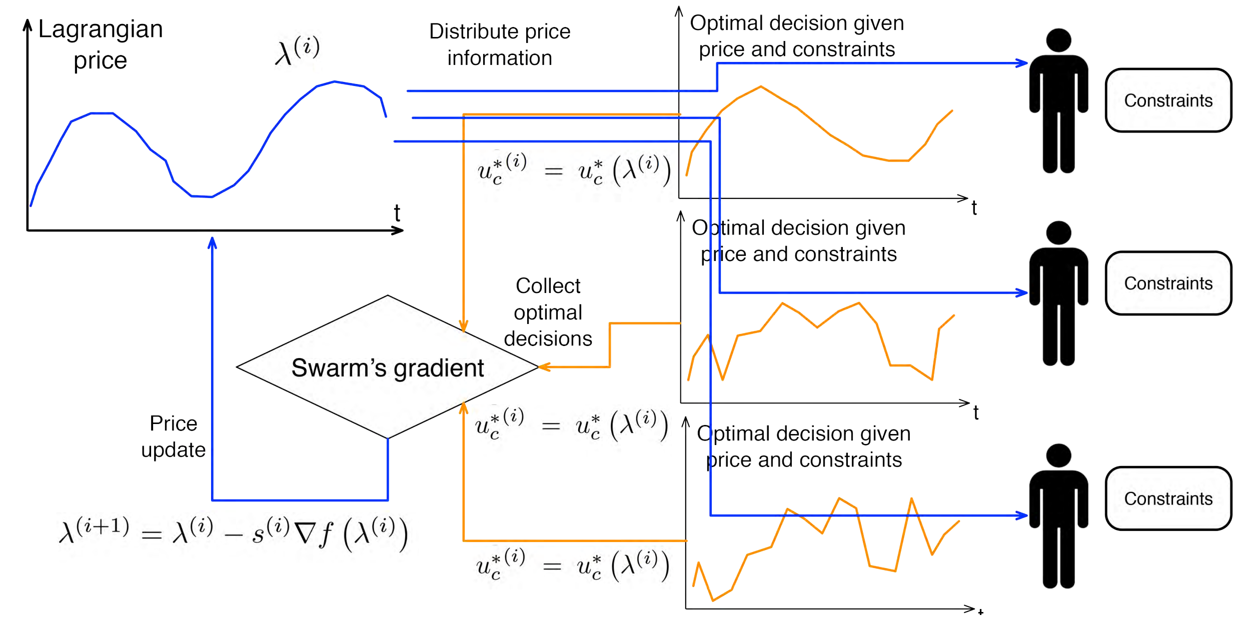

Lagrangian price
$\lambda^{(i)}$
t
Distribute price information
Optimal decision given price and constraints
$u_c^{*(i)} = u_c^*\left(\lambda^{(i)}\right)$
t
Constraints
Optimal decision given price and constraints
Collect optimal decisions
Swarm's gradient
t
Constraints
$u_c^{*(i)} = u_c^*\left(\lambda^{(i)}\right)$
Optimal decision given price and constraints
Price update
Constraints
$\lambda^{(i+1)} = \lambda^{(i)} - s^{(i)}\nabla f\left(\lambda^{(i)}\right)$
$u_c^{*(i)} = u_c^*\left(\lambda^{(i)}\right)$
t

Figure 4.2: Architecture for communication avoiding distributed convex optimization algorithm enforcing privacy for a swarm of controllers. At each step of the dual gradient ascent, a consensus variable is updated without the need for communicating the constraints of agents on the communication medium. The individual contribution to the group's gradient update can also be perturbed so as to obfuscate them. This privacy-preserving algorithm can be modified and track smooth changes dynamically as it is intrinsically iteratively converging to the optimum. In this architecture, each agent locally computes the solution of a small size Linearly Constrained Quadratic Program which provides each controller with an optimal decision while maintaining his personal availability constraints and preferences confidential.

Parallelism with respect to the agents

Let us precisely describe here how computation of the gradients are based on independent calculations by the agents. Indeed, calculating $\nabla\overline{\mathrm{r}_c}^*\left(-\lambda\right) = u_c^*\left(\lambda\right)$ is the only point where agents' constraints are to be taken into account and they are completely decoupled here. In particular, if each agent computes this step locally, it does not have to give any information about its individual constraints to others. This means that the reformulation above yields an intrinsically privacy preserving gradient method which does not require individual constraint to be communicated or centralized at any point. Changing the problem representation is also extremely important for the algorithm to scale. Communication entails the main compute time costs in the present setting. The method enables solving the problem of interest in the distributed setting while only communicating concise summaries of data sufficient to converge to optimality.

Incremental stochastic gradient method

A standard way of dealing with the optimization of the sum of differentiable functions is the incremental stochastic gradient method. Each holistic computation of the gradient is replaced by a small stochastic correction that on average descends towards the optimum. One can write $f(\lambda) = \frac{1}{N}\sum_1^N f_n(\lambda)$ with

$$f_n(\lambda) = \sum_{k=1}^{d} \ell_k^*(-\lambda_k) + \lambda^T o + \overline{\mathrm{r}_n}^*(-\lambda) \tag{4.4}$$

For any $n \in \{1 \dots N\}$, let L_n^* be the Lipschitz constant of the gradient of f_n, $L_n^* \leq \sum_{k=1}^d \frac{1}{m_k} + \frac{1}{\sigma_n}$, let $\lambda^{n,*}$ the minimum of f_n over $\mathbb{R}^d$.

The full gradient update step is replaced by a partial increment that only requires the computation of

$$\nabla^{u_{n_0}} f(\lambda) = -\sum_{k=1}^{d} \nabla\ell_k^*(-\lambda_k) + o - \nabla\overline{\mathrm{r}_{n_0}}^*(-\lambda) \tag{4.5}$$

where n_0 is chosen uniformly at random in $\{1 \dots N\}$. Let us recall that $\nabla\overline{\mathrm{r}_{n_0}}^*(-\lambda) = u_{n_0}^*(-\lambda)$ therefore the constraints of u_{n_0} do not need to be broadcast.

One can show, as in [170], that $\mathbb{E}\left[\|\nabla^{u_{n_0}} f(\lambda)\|_2^2\right] \leq A_{\text{inc}}^2\|\lambda - \lambda^*\|_2^2 + B_{\text{inc}}^2$ where $A_{\text{inc}}^2 = \frac{2}{N}\sum_{n=1}^N {L_n^*}^2$ and $B_{\text{inc}}^2 = \max\left(\frac{2}{N}\sum_{n=1}^N {L_n^*}^2\|\lambda^{n,*} - \lambda^*\|_2^2, \frac{A_{\text{inc}}^2 D_0}{2}\right)$ with $D_0 = \mathbb{E}\left[\|\lambda^{(0)} - \lambda^*\|_2^2\right]$

From [170], we deduce the lemma below.

Lemma 4.2.1 *If one uses decreasing step size $s^{(i)} = 2\left(m\left(2\frac{A_{\mathit{inc}}^2}{m^2} + i\right)\right)^{-1}$ after i iterations of the incremental stochastic gradient method*

$$\mathbb{E}\left[f(\lambda^{(i)}) - f(\lambda^*)\right] \leq \left(\frac{\sum_{n=1}^N L_n^*}{\sum_{n=1}^N {L_n^*}^2}\right)\left(2 + \frac{i}{T_{\mathit{inc}}}\right)^{-1} \tag{4.6}$$

where $T_{\mathit{inc}} = \frac{2}{N}\frac{\sum_{n=1}^N {L_n^*}^2}{m^2}$.

Here the convergence rate of the algorithm presented below, Algorithm 2, is much slower than before. However, the computation burden for each step is considerably lower as compared to the holistic case as only one optimal program needs to be computed and broadcast at each iteration.

Privacy preservation through obfuscation: approach and architecture

In the procedures above, the aim of enforcing the confidentiality of individual's constraints has been successfully achieved. This section will focus on obfuscating information that would help infer those indirectly. It is now a privacy-preserving measure that complements the privacy

Algorithm 2: Incremental stochastic gradient method

```
Data: Constraints $(\mathcal{C}_n)_{n\in\{1\dots N\}}$, target $o \in \mathbb{R}^d$
Result: Optimal dual price (argmin$_{\lambda\in\mathbb{R}^d} f(\lambda)$)
decide on initial value $\lambda^{(0)} \in \mathbb{R}^d$.
for $i \leftarrow 2$ to maximum number of steps do
    select $n_0$ at random in $\{1 \dots N\}$
    send $\lambda^{(i)}$ to node $n_0$
    compute optimal response $u^*_{n_0}\left(\lambda^{(i)}\right)$
    compute $\lambda^{(i+1)} = \lambda^{(i)} - s^{(i)}\nabla^{u_{n_0}} f\left(\lambda^{(i)}\right)$
end
```

preservation of agents' constraints. Our aim here is to show that the communication avoiding distributed representation we designed for the heterogenous sub-problems is a robust summary of their individual constraints.

We show that stochastic gradient ascent can be applied to solve the dual problem which converges to an optimal Lagrangian multiplier which can be interpreted as a of distribution of incentives that should be communicated to the agents for them to coordinate their actions in an optimal manner. In this context, we highlight a *force of the crowd* that preserves the average-based gradient in spite of the presence of random obfuscating perturbations. This guarantees we can reach any arbitrary precision while protecting the controllers with respect to an eavesdropping attacker, provided the fleet of individual agents is large enough.

Objective for privacy preservation through obfuscation: Obfuscation has to protect each agent's optimal program, $u_c^*(\lambda^*)$, from an opponent that is eavesdropping on the broadcast channel. A first obvious solution is to de-identify the vectors $\mathring{u}_n$ that are communicated to other agents so as to compute the new value of λ. Indeed, computation of λ is aggregated with respect to all individuals so as long as each $\mathring{u}_n$ is included once and only once in the process, the result is not changed. In the following we will consider a safer procedure that obfuscates the individuals' programs by adding white noise to communications.

The collective goal is indeed to collaborate in order to answer the overall demand best with an organized aggregated offer. While the contribution of each agent needs to respect to corresponding individual constraint set, the aim is not communicating constraints. Instead, we find here a smaller set of messages that are sufficient to reach optimal coordination and do not involve direct computation of the agent-level constraint blocks.

Stochastic privacy model

The specific approach chosen is to protect against a man-in-the-middle attack [22] which occurs in the system that allows the opponent listens to any broadcast message [5, 218]. We further assume that this adversary has a perfect ability to re-identify message senders. Algorithm 1 is interpreted as a survey in which, at each iteration i, all agents are queried for their optimal action $u_c^*\left(\lambda^{(i)}\right)$.

Agents do not have to send out their personal sets of constraints, $\mathcal{C}^c$, for the dual optimum λ^* to be estimated. However, they send out vectors $u_c^*\left(\lambda\right)$ which correspond to the optimal series of actions to undertake with respect to a given signal vector λ. This information is considered privacy sensitive as it can help infer the agents' trajectories. Therefore, one considers a framework close to that of [236], in which participants in a survey are reluctant to give out personal data. It is possible here to leverage the averaging behavior of the dual gradient in order to compute the common optimum of the whole community without jeopardizing individual's privacy. Consider iteration i of Algorithm 1, the set of optimal programs being broadcast is $u_c^*\left(\lambda^{(i)}\right)$. The attacker can keep these messages in memory and then compare them with $u_c^*\left(\lambda^{(i)}\right)$. Although messages have been de-identified, we consider the attacker can break that protection. In the example below, smartphones or IoT devices are the fundamental computational nodes of the procedure. An adversary can target a given device and listen to its outgoing messages with perfect knowledge of the home address they are being sent from. We assume that the only ability the attacker does not have is accessing the computations inside the nodes that enclose individual constraints. This assumption is standard in a man in the middle attack scheme [22].

We choose a specific attack model that highlights the trade-off solved by the dual representation we employ. Information useful to the group is unveiled at a faster rate than individual information. This shows first that homogeneity is indeed reached as individual-level information does not constitute the data employed by the agents to coordinate their efforts. The individual solutions to dual subproblem however condensate the information needed in the form of a vector with as many element as the demand we are trying to meet. Such a property highlights a certain form of minimalism and data efficiency in the representation design resulting from our search for communication avoidance.

Noise obfuscation of broadcast

Adding artificial noise to communicated data is a powerful tool used in differential privacy for databases in [61, 62, 52, 203] and in filtering [130] for example. Inspired by this work, we design an algorithm where, instead of sending $u_c^{*(i)} = u_c^*\left(\lambda^{(i)}\right)$, agent c broadcasts

$$\widetilde{u_c^*}^{(i)} = u_c^{*(i)} + \nu_c^{(i)} \tag{4.7}$$

in which the d-dimensional white noise sequences $(\nu_c^{(i)})_{i\in\mathbb{C}}$ are all mutually independent and have variance η^2 for each of their d components. This framework where only blurry observations of the gradient are available has also been studied in [58]. The approach presented here diverges in that it intrinsically leverages the effect of having distributed processors taking part in the computation of the gradient. In particular a high value of C is core to obtaining good precision and at the same time privacy enforcement, as in many stochastic approaches to privacy.

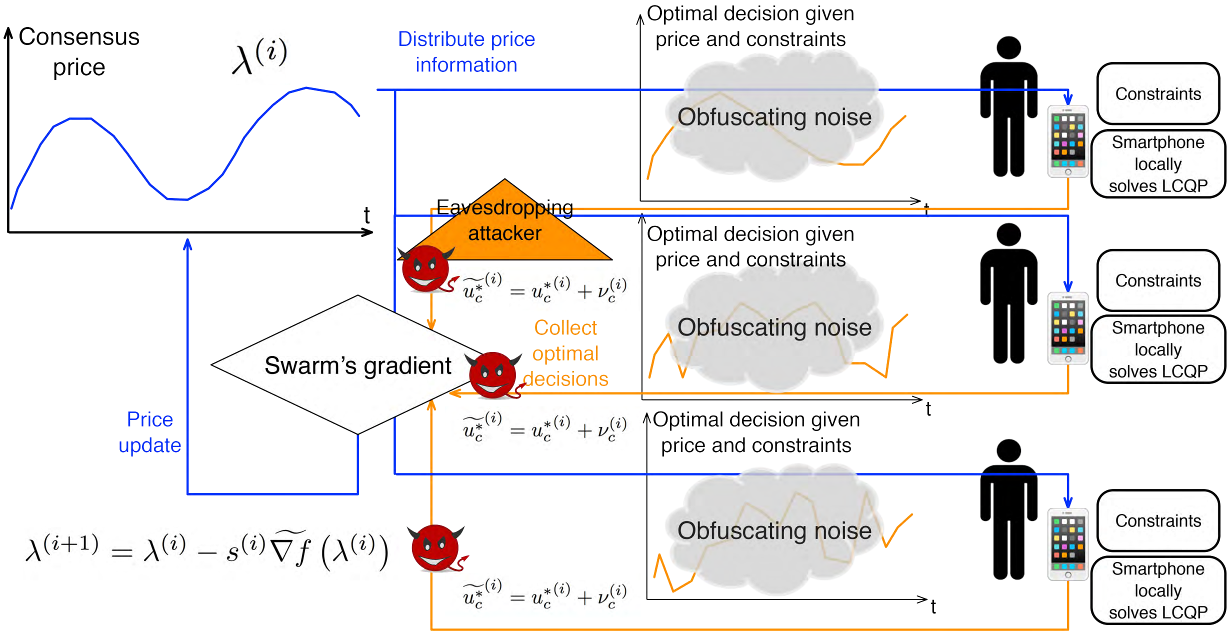

Consensus price
$\lambda^{(i)}$
t
Distribute price information
Optimal decision given price and constraints
Obfuscating noise
t
Constraints
Smartphone locally solves LCQP
Eavesdropping attacker
$\widetilde{u_c^*}^{(i)} = u_c^{*(i)} + \nu_c^{(i)}$
Optimal decision given price and constraints
Obfuscating noise
t
Constraints
Smartphone locally solves LCQP
Collect optimal decisions
Swarm's gradient
Price update
$\widetilde{u_c^*}^{(i)} = u_c^{*(i)} + \nu_c^{(i)}$
Optimal decision given price and constraints
Obfuscating noise
t
Constraints
Smartphone locally solves LCQP
$\lambda^{(i+1)} = \lambda^{(i)} - s^{(i)}\widetilde{\nabla} f\left(\lambda^{(i)}\right)$
$\widetilde{u_c^*}^{(i)} = u_c^{*(i)} + \nu_c^{(i)}$

Figure 4.3: Obfuscated distributed convex optimization algorithm enforcing privacy for a swarm of smartphones. Noise is added to outbound communication from the smartphones in order to obfuscate individuals' information related to the trajectories while enabling an accurate gradient update as the gradient is computed as an average across agents.

Learning rate on personal information

In this setting, the system itself cannot be trusted and there is competition between the speed at which the community discovers λ^* and the rate at which a spying statistician can learn individual information. This attacker is trying to estimate $\widehat{u^*}_c^{(i)}$ based on a series of i observations $\left(\widehat{u^*}_c^{(j)}\right)_{j\in\{1\ldots i\}}$ for a given agent c that is targeted as an individual. Classically, when trying to estimate a vector from a series of linearly perturbed measurements, empirical mean estimators or Kalman filters yield a *Mean Squared Error* (MSE) that will scale proportionally to the variance $T \times N\eta^2$. Therefore we assume the attacker's estimator for $u_c^{*(i)}$ is unbiased and has variance $\mathbb{E}\left[\left\|\widehat{u_c^*}^{(i)} - u_c^{*(i)}\right\|_2^2\right] = \frac{T\times N\eta^2\kappa}{i^\gamma}$ where κ is a constant that depends on the estimation technique adopted by the adversary and γ it's learning rate. The most favorable case for the attacker occurs when the sequence $\left(u_c^{*(j)}\right)_{j=1\ldots i}$ remains constant. The law of large numbers guarantees a convergence rate $\gamma = 1$ for the empirical mean estimator. Thus, from hereon, we will assume $\gamma \leq 1$. The privacy enforcement criterion here is that the MSE of the estimator of the attacker remains above a certain lower bound κ_{min}. This implies the optimization program has an iteration budget

$$i^{\max} = \left(\frac{d\kappa\eta^2}{\kappa_{\min}}\right)^{\frac{1}{\gamma}}. \tag{4.8}$$

Noisy descent

The privacy enforcing strategies below aim at converging towards an optimal scheduling price λ^* faster than the attacker increases its precision in the estimation of u_c^*.

A strategy to preserve privacy in the distributed gradient computation is to run the deterministic holistic descent above with noisy broadcasts from the agents. The update of λ in the descent becomes $\lambda^{(i+1)} = \lambda^{(i)} - s^{(i)}\widetilde{\nabla f}\left(\lambda^{(i)}\right)$ where

$$\widetilde{\nabla f}\left(\lambda^{(i)}\right) = -\sum_{k=1}^{T\times N} \nabla \ell_k^*\left(-\lambda_k^{(i)}\right) + d - \frac{1}{C}\sum_{c=1}^{C} \widetilde{u^*}_c^{(i)}\left(-\lambda^{(i)}\right).$$

Recalling that $\widetilde{u_c^*}^{(i)} = u_c^{*(i)} + \nu_c^{(i)}$, as $\nu_c^{(i)}$ is a white noise whose variance trace is $T \times N\eta^2$, $\mathbb{E}\left[\widetilde{\nabla f}\left(\lambda^{(i)}\right)\right] = \nabla f\left(\lambda^{(i)}\right)$ and

$$\mathbb{E}\left[\left\|\widetilde{\nabla f}\left(\lambda^{(i)}\right)\right\|_2^2\right] \leq A_{\text{hol}}^2\left\|\lambda^{(i)} - \lambda^*\right\|_2^2 + B_{\text{hol}}^2. \tag{4.9}$$

where $A_{\text{hol}} = L = \sum_{k=1}^{T\times N}\frac{1}{m_k} + \frac{1}{C}\sum_{c=1}^{C}\frac{1}{\sigma_c}$ and $B_{\text{hol}}^2 = \frac{T\times N}{C}\eta^2$.

These considerations lead to the following "obfuscation by the crowd" theorem.

Theorem 4.2.2 *The precision that can be reached in terms of MSE for the numerical Lagrangian multiplier while obfuscating the agents' actions is inversely proportional to the size of the fleet.*

Proof 4.2.2 *Using the notation $m = \sum_{k=1}^{\times N}\frac{1}{m_k}$, if one uses step size $s^{(i)} = \left(m\left(2\frac{A_{hol}^2}{m^2} + i\right)\right)^{-1}$, after i iterations of the noisy holistic descent,*

$$\mathbb{E}\left[f(\lambda^{(i)}) - f\left(\lambda^*\right)\right] \leq \frac{2B_{hol}^2}{L\left(2 + \frac{im^2}{A_{hol}^2}\right)} \leq 2\frac{\frac{T\times N}{C}\eta^2 L}{im^2}. \tag{4.10}$$

As we defined L as a function of the average of $\frac{1}{\sigma_c}$ among the agents $c \in \{1\ldots C\}$, L is indeed independent from the size of the fleet we consider.

It is noteworthy that the fact that the constants in this bound are sensitive to T and N does not change the conclusion regarding the role of the number of agents, C, which, as it increases, offers better opportunities for privacy enforcement by obfuscation.

Noisy incremental stochastic descent strategy

In such a case we get a similar bound for the expected value error with constants $A_{\text{inc noisy}} = A_{\text{inc}}$ and $B^2_{\text{inc noisy}} = B^2_{\text{inc}} + \eta^2$.

The convergence rate is unsurprisingly slower. However participating agents send out sensitive data less often as only one individual selected uniformly at random broadcasts its response vector $u_n^{*(i)} = u_n^*\left(\lambda^{(i)}\right)$ at each iteration. Let $\xi^{(i)}$ the number of times the agent selected at the first iteration has sent out its optimal response (including that of the first step). The random variable $\xi^{(i)} - 1$ is the sum of $i-1$ Bernoulli trials with parameter $\frac{1}{N}$. Therefore $\xi^{(i)} - 1$ has a binomial distribution with parameters $\frac{1}{N}$ and $i-1$. Jensen inequality yields

$$\mathbb{E}\left[\left\|\widehat{u^*}_n^{(i)} - u_n^{*(i)}\right\|_2^2\right] = E\left[\frac{d\kappa\eta^2}{\left(\xi^{(i)}\right)^\gamma}\right] \geq \frac{d\kappa\eta^2}{\left(1+\frac{i-1}{N}\right)^\gamma}.$$

Proposition 4.2.1 *After $N\left(\left(\frac{d\kappa\eta^2}{\kappa_{min}}\right)^{\frac{1}{\gamma}} - 1\right)$ iterations of the incremental noisy stochastic descent method the MSE of the adversary, $E\left[\left\|\widehat{u^*}_n^{(i)} - u_n^{*(i)}\right\|_2^2\right]$, is bounded by below by κ_{min} and the privacy preservation constraint is respected.*

Studying a competition of individual and group level learning rates precisely highlights how as the crowd of agents gets larger, getting precise information from each individual is no longer necessary. The dual representation is expressive and robust enough for the controllers to jointly converge to an optimal plan of action.

4.3 Distributed control of Mobility-as-a-Service (MaaS) fleets

On-demand traffic fleet optimization requires operating *Mobility as a Service* (MaaS) companies such as Uber, Lyft to locally match the offer of available vehicles with their expected number of requests referred to as demand (as well as to take into account other constraints such as driver's schedules and preferences). Alongside the demand patterns, it is also preferable to take into account the constraints of the drivers in terms of service start and stop locations as well as their availability schedule. In the present section, we show that this problem can be encoded into a *Constrained Integer Quadratic Program* (CIQP) with block independent constraints that can then be relaxed in the form of a convex optimization program [26]. We prove that this program has a number of heterogenous constraints that scales linearly with the number of vehicles. We leverage this particular structure to design a scalable distributed optimization algorithm corresponding to computing a gradient ascent in a dual space [18] we presented. The goal of the approach is to avoid communication from the drivers, i.e this technique preserves the privacy of the drivers' schedules which no longer need to be made public. This new framework does not require the drivers to share their availabilities with the operating company (as opposed to standard practice in today's mobility as a service companies). The resulting parallel algorithm can run on a distributed smartphone based platform (i.e. can be implemented directly in today's leading companies in the field).

Numerical experiments based on actual data collected on Uber rides in New York including 4×10^5 samples show the algorithm efficiently synthesizes a spatio-temporal price multiplier that enables accurate demand tracking by the probability of presence of vehicles. The algorithm is demonstrated in different settings with up to 10^4 vehicles involved in the fleet of mobility as a service. We prove this method achieves three different goals: privacy preservation, tractability of computations and obfuscation of the optimal solution for the fleet management problem we consider.

Motivation for application to MaaS

Let us first delineate three reasons why MaaS as been considered as a prime application for the approach we developed. Designing a solution to the optimal dispatch problem for MaaS is an opportunity to offer an open-source method to solve a problem of importance for urban areas, provide an approach different from the non-scalable standards in fleet management and offer a method tailored to the constraints of scale, communication avoidance, privacy and latency in MaaS.

Open-sourcing *Mobility as a Service*. A key component in *Mobility as a Service* (MaaS) companies such as Uber and Lyft is market making and market matching. The famous price multiplier of the former incentivizes drivers to go to regions where many rides are requested whereas the latter takes into account the personal schedule of the drivers. To remain competitive, MaaS companies keep their optimization strategies a secret. In the current section, we present a convex optimization formulation and provide an open source algorithm that match drivers with demand in an optimal manner while taking into account the drivers' availability as a set of constraints. In addition, we show that using a dual splitting technique, the convex program can be parallelized with respect to the agents. This has two important consequences. It alleviates the need for a single machine with a large amount of memory. Indeed we leverage the fact that the schedules of the drivers can be considered as independent constraint sets. It also does not require drivers to share their schedules which naturally protects this privacy sensitive piece of information. We focus in the present section on the fleet management dimension of providing MaaS which complements recent work on decentralized trust management for ride sharing in [53, 201] and strategic meeting point choice to mitigate information leakage in [7].

Non scalable methods for fleet management. A sizable body of research focuses on optimizing fleet management, vehicle dispatching and multimodal transportation optimization. In order to assess the best optimization policies for taxi fleets, numerous simulators have been developed. For instance, in [36], an agent-based simulation program was used in conjunction with a large dataset of GPS trajectories to learn and mimic the behavior of taxi drivers. In [148] researchers have modified general purpose agent-based traffic simulators such as MATSim [108] and incorporated taxi ride simulation. Both these approaches aimed at providing taxi companies with tools and methods dedicated to better informing fleet management decisions with a focus on better answering demand modeled as queues and minimizing idling driver time. GPS tracking has been instrumental in calibrating taxi fleet simulators as well as real-time information services for large taxi fleets [11, 140, 139] which aimed at better informing optimization decisions in real time. More recently, the advent of massive electric vehicle fleets has given rise to a new class of problems. These analyze the structure of demand of rides and dispatch vehicles while taking into account the constraints inherent to the use of electric vehicles whose batteries provide a shorter range [78]. Such problems have also been considered for the design of optimal charging station networks for such fleets [30].

Integrating operational constraints in privacy preserving environments. The framework in which we develop the novel approaches presented in this work takes into account constraints that correspond to each agent in the system such as driving preferences, electromobility-induced constraints, etc. These constraints can encode a variety of factors, the most interesting one for the study being driver availability (which we want to protect for privacy reasons). In such a setting, the demand for rides can be studied in a statistical manner leveraging the important amount of data being collected by MaaS systems. Note also, as will appear later, that the method does not require the assumption that the demand for rides is elastic as in [242]. Our focus is that of making MaaS open; this specifically takes the form of a peer-to-peer distributed algorithm that can run on smartphones and achieves an aim similar to that of [112]. As in [147], a major concern is that the algorithm scales which typically becomes problematic when the number of constraints increases linearly with the number of drivers, a problem that we adress as well. For completeness, a large number of approaches for ride-sharing optimization can be found in [4].

In the following, we show how to model the problem of matching demand for rides with the schedule of MaaS drivers and how this can be formulated as an optimization program. We first focus on the identification of the structure of the demand prior to formulating the resulting matching problem.

Characterization of demand as a seasonal time series in the MaaS problem

We show how openly accessible data can, using a coarse yet robust statistical estimation, yield reliable estimates of the locations and times corresponding to demand for mobility. We operate under the assumption that the only shared location by the rider is that of the departure of the trip. This corresponds to the reality of popular applications for MaaS today in which the destination of the rider can be modified until the start of the trip. We show how, once the demand for rides has been characterized using this method, the availability constraints of the drivers can be matched as a mobility offer in the form of an integer program.

System identification

Demand for rides can be characterized as a multivariate time series $(D_t) \in \mathbb{R}^d$ where d is the number of cells considered in the spatial discretization grid (see Figure 4.4 for the case of NYC). These are sometimes referred to as "heatmaps" in MaaS companies. For each element of the grid i and each timestep t (typically 5 minutes), D_t^i is the sum of all ride requests for the service that have been observed in region i. Let $(U_t) \in \mathbb{R}^d$ the sum of vehicles available in a grid cell at time t. A way to encode optimal matching is to penalize the sum of squared distances between the vectors U_t and D_t at each timestamp to which we add a regularizing function over (U_t). This latter component of the objective represents the aversion of drivers to long distances and going to highly congested zones. This will enforce a vehicle allocation schedule matching the spatial density of ride requests with that of available mobility platforms. At the same time, it will penalize the fleet dispatch plan in order to discourage traveling long distances or go through regions with a high number of vehicles, therefore taking into the route preferences of the drivers [24].

Spatio-temporal structure of demand for MaaS

As an instantiation of the method exposed to identify the structure of demand for rides, further exploring the statistical facts exposed in [189] for on-demand mobility in New York city, we show that the demand for rides (D_t) has strong spatial and temporal periodicity (which is illustrated in Figure 4.5 and Figure 4.4). A consequence of this fact is that we can use seasonality analysis [28, 29] as the basis for an optimization scheme taking into account the drivers' availability and schedules. The stochastic input demand profile [8] can of course be updated as new data is collected with a receding horizon approach [159] and the demand forecast obtained by multivariate time series analysis [216, 59, 146] changes. Updates for traffic conditions can be taken into account in real time [15, 157].

Data

Within the data set used consisting of Uber ride requests in 2014, we chose a random day, April 7th, to illustrate the properties of our algorithm with actual demand measured in Manhattan for Uber rides extracted by [71]. Drivers constraints where simulated at random with a start time picked uniformly at random through the day and an end time picked uniformly at random within the reminder of day after the start time. We divided the day into 24 one hour periods and aggregated the demand within these time bins. We also discretized the spatial space of the city with a tessellation grid of size 16×16.

Control problem

Formulation of demand matching with boolean representation of vehicles

The time series $(U_t) \in \mathbb{R}^d$ corresponds to the aggregation of drivers' locations over spatial regions whereas $(D_t) \in \mathbb{R}^d$ is a demand time series summing up collocated requests occurring in the same time interval. The optimization scheme we focus on in this section leverages aggregated demand profiles and enables the optimization of vehicle dispatch at the driver level, taking personal constraints and driver utility functions into account. The discretized spatial grid can easily be modeled as a network graph [243] representing the mobility space (see Figure 4.4). Each edge represents a spatial tesselation region. Neighboring regions are linked one to another by an edge. The position of driver c at time t is modeled by a integer vector over the vertices of $\mathcal{G}$: $(b^c_t) = \left(b^c_{t,i}\right)_{i \in \{1 \ldots d\}} \in \mathbb{N}^d$

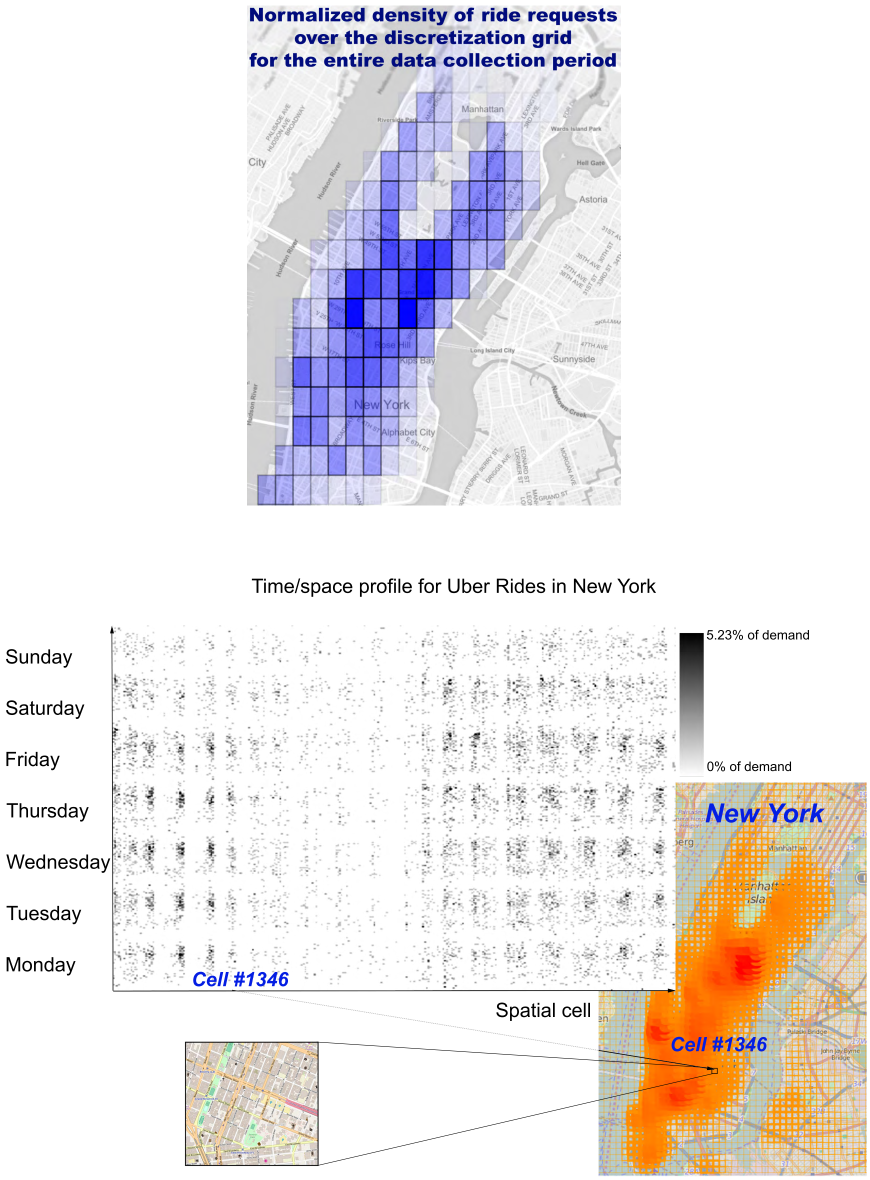


Figure 4.4: **Top:** Spatial quantization of demand for Uber rides in New York. Each element of the grid corresponds to aggregate ride requests in a tessellation region during each time interval (depicted at a given time in this figure). Empty cells correspond to regions with low or zero ride requests. This data is studied as a multivariate time series and its seasonal component studied. The mobility on-demand fleet distribution is then optimized based on this data with the novel algorithm we present. **Bottom:** Spatio-temporal seasonality in Uber demand profile in NYC for April 2014. The strong spatio-temporal seasonality paves the way towards accurate predictions and planning of requests for rides. Forecast can then be dynamically updated by standard linear model iterative fitting to dynamically adapt to randomness in the demand time series.

whose value is 1 for vertex n if and only if the driver's location is vertex n at time t. The value of the integer b_t^c may also be used to encode the number of seats available.

Formulation of demand matching as a discrete optimization program

Up to a scaling factor for the number of shared vehicles located at a given node, alleviating the spread between demand and offer of ride requests can be written as the following objective where C is the total number of drivers taken into account and T the total number of time steps the schedule is planned for.

Fleet management problem:

$$\begin{array}{ll} \min\limits_{u \in \{0,1\}^{d \times C \times T}:} & \sum_{t=1}^{T} \sum_{n=1}^{d} \left(d_{t,n} - \sum_{c=1}^{C} b_{t,n}^{c} \right)^2 + \sum_{c=1}^{C} \sum_{t=1}^{T-1} \sum_{n=1}^{d} \rho_{c,t,n}^2 \left(b_{t+1,n}^{c} - b_{t,n}^{c} \right)^2 \\ & + \sum_{c=1}^{C} \sum_{t=1}^{T} \sum_{n=1}^{d} \sigma_{t,n}^2 \left(b_{t,n}^{c} \right)^2 \\ \forall c \in \{1 \ldots C\}, & \forall n \in \{1 \ldots D\}, \forall t \in \{1 \ldots T\}, b_{t,n}^{c} \geq 0 \text{ if and only if vehicle c in cell } n \\ \forall c \in \{1 \ldots C\}, & b^c \in \mathcal{C}^c \end{array} \tag{4.11}$$

where $\forall c \in \{1 \ldots C\}, b^c \in \mathcal{C}^c$ is equivalent to $\forall n \in \{1 \ldots D\}, \forall t \in \{1 \ldots T\}, b_{t,n}^c = 0$ if vehicle c is not available at time t; $d_{t,n}$ is the demand in cell n at time t. This constraint set will be augmented with mobility constraints as we will show later on; $\left(\rho_{c,t,n}^2\right)$ is a set of travel distance penalization parameters; $\left(\sigma_{t,n}^2\right)$ is a set of penalization parameters over congested cells. In the equations above and the following, t will be a temporal index, n a spatial index, and c will identify vehicles. The specific terms in the program above are explained below:

- Demand matching term: $\sum_{t=1}^{T} \sum_{n=1}^{d} \left(d_{t,n} - \sum_{c=1}^{C} b_{t,n}^{c} \right)^2$ is a square loss penalizing the mismatching between the number of vehicles expected in each cell and the number of rides requested (demand). Note that last section of this section proves that any strongly convex loss can be used here instead of this quadratic.

- Travel time and distance penalty: $\sum_{c=1}^{C} \sum_{t=1}^{T-1} \sum_{n=1}^{d} \rho_{c,t,n}^2 \left(b_{t+1,n}^{c} - b_{t,i}^{c} \right)^2$ is a quadratic penalty term which accounts for the cost in time and energy for the drivers corresponding to moving across the discretized tesselation grid. In other words, penalizing fleet re-balancing is desirable (and encoded in this function). Again, any arbitrary strongly convex loss can be used here, as proved later in the section. This means in particular that instead of a distance an expected travel time can be added, for instance, a quadratic loss on the number of traffic lights present in a cell of the discretized grid.

- Regularization: $\sum_{c=1}^{C} \sum_{t=1}^{T} \sum_{n=1}^{d} \sigma_{t,n}^2 \left(b_{t,n}^{c} \right)^2$ is a quadratic term that discourages drivers from gathering in large groups in the same cells. Once more, any strongly convex loss can replace the quadratic term we chose. Strong convexity implies that there is an increasing marginal cost to concentrating the distribution of a driver and therefore encourages spreading vehicles across the discretization grid. This term may also represent the aversion of drivers to driving in zones that feature problematic driving conditions such as uncertainty about the level of congestion or the absence of stopping locations.

Constraint sets

In this paragraph, we show how the constraints on the possible movements of the agents can be extended and be more detailed. Any fleet management algorithm has to take the drivers' constraints into account as well as their mobility constraints. Letting $avai\left(c, t\right)$ be the Boolean variable that represents the availability of driver c at time t, we have $\forall t \in \left\{1 \dots T\right\}, \forall c \in \left\{1 \dots C\right\}, \sum_{n=1}^{N} b^{c}_{t,n} = avai\left(c, t\right)$. Let $init\left(c\right)$ the entry point of driver c at time t^{s}_{c} (where s stands for "start time", the entry point represents the location at which the agent has parked before starting the MaaS part of their day) when vehicle c start its trip, we add the constraint $b^{c}_{t=t^{s}_{c}} = init\left(c\right)$, where $init\left(c\right)$ is the dirac vector concentrated on the start position of vehicle c. Let $end\left(c\right)$ the exit point of driver c at time t^{e}_{c} when vehicle c ends its trip, $b^{c}_{t=t^{e}_{c}} = end\left(c\right)$ the dirac vector concentrated on the stop position of vehicle c. If we do not assume that the graph is fully connected for any time stamp t, we can add the constraints $\forall t \in \left\{1 \dots T-1\right\},\ b^{c}_{t} = M_{t} b^{c}_{t-1}$ where M_t is the connectivity matrix of the network. In particular M_t is filled with 0, $M_{t,n_1,n_2} = 1$ if anf only if n_1 and n_2 are connected through the network at time t. The final constraint set we obtain, once mobility constraints have been taken into account consists of two elements. *Availability of agent for MaaS:* $\forall c \in \left\{1 \dots C\right\}, b^{c} \in \mathcal{C}^{c}$ is equivalent to $\forall n \in \left\{1 \dots D\right\}, \forall t \in \left\{1 \dots T\right\}, b^{c}_{t,n} = 0$ if vehicle c is not available at time t. *Mobility constraints:* $\forall t \in \left\{1 \dots T-1\right\},\ b^{c}_{t} = M_{t} b^{c}_{t-1}$.

We consider a convex relaxation of the *Constrained Integer Quadratic Programming* (CIQP) problem which differentiates our approach from [150, 132, 197, 208] for vehicles. Our choice of an approximate relaxation convex program as in [158] is motivated by the practical fact that we build a system that encourages drivers to move towards regions with a high number of potential customers. Drivers are indeed free of their re-routing decisions in on-demand mobility, which leads to behavioral changes as compared to what has been previously observed for vehicles [244]. As actors on a two sided market, they are also free to refuse to pickup a costumer and therefore the vehicle occupancy is not taken into account in the optimization objective. We derive a dual reformulation which yields a distributed gradient ascent algorithm converging to the optimal solution of a strongly convex problem.

Our method is therefore tailored to shaping incentives drivers to answer demand in an optimal way while taking their personal constraints into account as they join a massive fleet of vehicles. We leverage this unique aspect of MaaS in order to design a distributed algorithm which is tailored for speed of adaptation to changing conditions and privacy preservation rather than precisely controlling each vehicle individually. Our numerical experiments show indeed that the parallel implementation of our algorithm can run on the equivalent of a group of two thousand smartphones with a limited communication bandwidth and yet find an optimal price multiplier for the drivers in a matter of minutes. These results are moreover rather pessimistic as we did not use a warm-start or a specialized convex optimizer to speed it up. A limitation however is that our algorithm is not adapted to the exact and precise control of a fleet of vehicles.

The computational complexity of the problem prevents efficient solutions to the exact (integer) problem. In addition, solving the problem exactly is not necessary with fleets of drivers that are not directly controlled. This enables us to achieve three main goals which will be at the center of the next two sections.

Hyperparameter tuning

The problem formulation above features several sets of hyperparameters (ρ, σ) which can be calibrated by a cross-validation procedure in order to produce the best outcome for this optimization scheme. This can be done by operating an actual fleet or more realistically in practice using a vehicle fleet simulator such as [36, 148]. In our numerical experiments, we have used nominal parameters that did not result from the use of such simulators different numerical values do not affect the performance of our algorithms significantly. However, the results we present show that the set of hyperparameters we selected successfully enabled the tracking of the demand with the vehicle probability of presence distribution.

Convex relaxation

Assuming vehicles would directly be controlled by the managing company, this leads to a Contrained Quadratic Integer Program (CQIP). As the market making solution being offered by MaaS does not directly control the drivers, we focus on controlling a probability of presence of the drivers which corresponds to a convex relaxation of the integer program we formulate.

As a starter to address the issue of tractability we employ relaxation. In the previous problem, $\forall t \in \{1 \dots T\}$, $\forall c \in \{1 \dots C\}$, $b^c_{t,n} \in \{0,1\}$. We now write a relaxed approximation of the problem with $u^c_{t,n} \in [0,1]$. $\forall c \in \{1 \dots C\}$ let $\mathcal{C}^c$ the set of constraints of vehicle c, the constraints are: $\forall t \in \{1 \dots T\}$, $\forall n \in \{1 \dots N\}$, $u^c_{t,n} \geq 0$;$\forall t \in \{1 \dots T\}$, $\forall n \in \{1 \dots N\}$, $u^c_{t,n} = 0$ if vehicle c is not available at time t; $u^c_{t=0} = init\left(c\right)$ (initial location of vehicle c, e.g. parking location at the driver's home); $u^c_{t=T} = end\left(c\right)$ (final location of vehicle c, e.g. parking location at the driver's home), $\forall t \in \{1 \dots T-1\}$, $u^c_t = M_t u^c_{t-1}$ (movement constraints between traffic regions) if the traffic network is not fully connected at all times. Otherwise one considers the mass conservation constraint $\sum_{n=1}^{N} u^c_{t,n} = 1$. We can see that this problem is equivalent to that of considering the position of a given vehicle at any time t as a probability distribution over the network. Note also that the practice of aggregation of supply often used by MaaS companies such as Uber or Lyft is somewhat aligned with the physical interpretation of relaxation in the present case.

Relaxed problem formulation

The position of driver c at time t is modeled by a presence vector over the vertices of $\mathcal{G}$: $(u^c_t) = \left(u^c_{t,n}\right)_{n \in \{1 \dots N\}} \in \mathbb{R}^N$ whose value is one for node n if and only if the driver's location is n at time t.

The relaxed problem we consider is now convex:

Convex relaxation:

$$\min_{u \in \mathbb{R}^{d \times C \times T}:\, \forall c \in \{1 \ldots C\},\, u^c \in \mathcal{C}^c}$$
$$\sum_{t=1}^{T} \sum_{n=1}^{N} \left(d_{t,n} - \sum_{c=1}^{C} u_{t,n}^{c} \right)^2 + \sum_{c=1}^{C} \sum_{t=1}^{T-1} \sum_{n=1}^{N} \rho_{t,c,n} \left(u_{t+1,n}^{c} - u_{t,n}^{c} \right)^2 + \sum_{c=1}^{C} \sum_{t=1}^{T} \sum_{n=1}^{N} \sigma_{t,n} \left(u_{t,n}^{c} \right)^2 \tag{4.12}$$

The solution to this problem will be denoted $u^* = concat\left((u_c^*)_{c=1\ldots C}\right)$ where u_c^* is the optimal behavior assigned to the agent number c. One major difference appears with the initial boolean problem. New constraints need to be taken into account, for each vehicle. Namely, $u_{t,n}^c \geq 0$ and $\sum_{n=1}^{N} u_{t,n}^c = 1$. This does not change the separability of constraints among vehicles which is the key assumption needed to parallelize solving the convex optimization problem we obtain after relaxation.

Answering demand for rides with a probability distribution of vehicles

We are mainly interested in fleet management for MaaS companies such as Lyft or Uber. In that particular setting it makes sense to consider a probability distribution of presence for the driver and guiding this probability distribution. Indeed, these drivers should not be considered as employees but as contractors and therefore the re-routing scheme is about designing incentives and not control. Not only is this framework useful for large numbers of drivers as observed in urban areas, but it also corresponds to a modeling approach desired by MaaS companies. Also, the present optimization scheme only considers demand is characterized by the location and number of ride requests. The location of the destinations are not taken into account. This models the fact that Uber or Lyft drivers only get to know the destination of the costumer after the pick up and the rider is therefore free to change the destination before pickup time. Considering destination locations is not problematic. However, the size of the state space would increase quadratically therefore requesting more computational time and memory. The corresponding increase in computational complexity can potentially be addressed with the optimization algorithm presented later based on a dual splitting method is tailored for solving the main problem by distributing it over many machines. As illustrated in Figure 4.6, considering the convex relaxation of the problem is key to having a model that although approximate is appropriate for Mobility as Service (MaaS) applications and can be solved in a scalable manner. A thorough analysis has been conducted in [185] and [184] for the recovery of sparse probability measures by convex programming. In these articles, bounds have been proven that could be applied to our setting. We do not need such a thorough analysis in our study as the convex relaxation over the location of the agent characterizes the flexibility offered to the driver by MaaS. Uber drivers are free to go where they want and can only receive an incentive to drive to zones with a higher price multiplier.

Dual splitting method

This section presents a dual perspective of the problem that explicitly separates the constraint sets of the drivers. It leads to the formulation of a Lagrangian quantity encoding both the structure of the demand and the availability constraints of the drivers. The following proposition is a direct consequence of the preliminary study on dual representations of heterogenous distributed control problems.

Proposition 4.3.1 *The primal problem is equivalent to the* ***Dual problem:***

$$\begin{aligned} &\underset{\lambda}{max} - \sum_{t=1}^{T}\sum_{n=1}^{N}\left(\lambda_{t,n}d_{t,n} + \frac{\lambda_{t,n}^2}{4}\right) \\ &+ \sum_{c=1}^{C}\underset{u^c\in\mathcal{C}^c}{min}\left(\sum_{t=1}^{T}\sum_{n=1}^{N}\left(\lambda_{t,n}u_{t,n}^c + \sigma_{t,n}\left(u_{t,n}^c\right)^2\right) + \sum_{t=1}^{T-1}\sum_{n=1}^{N}\rho_{t,c,n}\left(u_{t+1,n}^c - u_{t,n}^c\right)^2\right). \end{aligned} \tag{4.13}$$

Once converged, u^* is equivalent to the primal solution, the solution λ^* to (4.13) is an optimal price multiplier. We will show that at the optimal point (u^*, λ^*) drivers' constraints are respected and yet need not be communicated, privacy is preserved. Indeed, the optimal solution u^* for the entire fleet is the concatenation of the individual optima $(u_c^*)_{c=1\ldots C}$. The calculation of u_c^* for a particular value of c only depends on messages obtained from the rest of the fleet through λ^* as it is the solution to

Proposition 4.3.2 ***Agent level problem:*** *Once the optimal value of λ in* (4.13) *has been found (which will be explained in Algorithm 1), it is sufficient for each agent to solve the sub-problem*

$$\underset{u^c\in\mathcal{C}^c}{min} f_c\left(\lambda^*, u^c\right) = \sum_{t=1}^{T}\sum_{n=1}^{N}\left[\lambda_{t,n}^*u_{t,n}^c + \sigma_{t,n}\left(u_{t,n}^c\right)^2\right] + \sum_{t=1}^{T-1}\sum_{n=1}^{N}\rho_{t,c,n}\left(u_{t+1,n}^c - u_{t,n}^c\right)^2 \tag{4.14}$$

to find its optimal schedule in order to return its additive contribution to the gradient $u^{c*}(\lambda) = \underset{u^c\in\mathcal{C}^c}{argmin} f_n\left(\lambda, u^c\right).$

Dual splitting algorithm The considerations above lead to a straightforward separation of the problem through the dual. In section 4.2 we introduced a gradient ascent based algorithm that achieves convergence. This distributed privacy preserving optimization scheme is provably able to track a demand distribution. In Figures 4.7 and 4.8, we show how the optimal distribution of vehicles is able to track the distribution of demand as it evolves in time. These plots are generated with 10^3 and 10^4 vehicles whose availabilities are generated at random with a uniform law for the beginning and the end of the vehicle shift. In order to check how the scheme performs we compare the demand distribution and the offer distribution with a Sum-Of-Squares metric (Kullback–Leibler divergence cannot be used here as the supports may be disjoints). This result is expected and confirms that the privacy-preserving distributed optimization method we presented is efficient to match demand and offer when constraints on the offer form a cartesian product of independent sets.

Optimal Lagrangian value as a coordination price enabling privacy

One of the most interesting outputs of the optimization scheme we propose more λ^* than u^*. Indeed, while u^* gives access to the fleet management, λ^* is a coordination signal in that it is sufficient to broadcast it to all the vehicles involved in the fleet for them to individually change their planning and update the next price with respect to which the fleet is going to coordinate. This result enables two important extensions to the scheme we propose. Splitting by the dual leads to the formulation of a Lagrangian vector λ^*. For each time step, $(\lambda^*_{t,n})_{n=1\dots N}$ projects a heatmap onto the discretization cells which optimally synthesizes the value that should be attributed to the different regions of the map so that drivers are optimally encouraged to meet the demand. This price takes implicitely into account the availability of the agents through time and the points at which they start and end their service. The optimal value in the dual λ^* is an optimal coordination Lagrangian price that makes taking agents' constraints implicit therefore eliminating the need for communicating them and centralizing them in a single agency.

Section 4.2 already proved, using an analysis of the convergence of the algorithm we propose, that λ^* can be found without any communication of the agents' constraints.

Dynamic tracking. First, if the optimal solution changes slightly because of an unexpected variation of the demand or the addition or deletion of agents from the system, a new optimal price will be devised once the system has once more converged in the dual space that corresponds to the optimum in the primal space. Using a dual splitting reformulation of a convex relaxation approximation enables a re-computation in real time for the swarm of taxis. This allows adaptation to random variations and changes in vehicle availability.

Electric vehicles. Second, another interesting aspect is that constraints that are independent across vehicles can arbitrarily be added to the system without a modification of the scheme. The corresponding individual constraint set $\mathcal{C}^c$ is updated if there is a modification of the properties of the vehicle indexed by c. This naturally enables the optimization program to take into account constraints that entail the charging of electric vehicle batteries. Electric vehicles having a more restricted action radius (sometimes referred to as "range of anxiety"), the operators may want to take the corresponding additional availability constraints into account. These constraints are naturally decoupled across vehicles as they only relate to periods during which a given vehicle is not available for mobility as a service because it needs to be charged. Therefore the procedure we present is naturally able to handle fleets of mobility as a service with electric vehicles.

Enabling coordination while preserving privacy. We therefore showed how separating block independent constraints coupled in the objective by the dual is a technique that entails a natural privacy preservation mechanism. The most significant outcome of the dual splitting method is the formulation of an optimal Lagrangian multiplier λ^* which enables agents to coordinate their response to the demand while respecting their own availability constraints. We proved that convergence towards λ^* is robust and can occur without the need for the agents to communicate their constraints at any iteration of the privacy preserving convergence algorithm we devise.

We show in Figures 4.7 and 4.8 that our strategy manages to create a presence likelihood of vehicles that tracks a given demand. The results below have been computed for 1000 and 10000 vehicles but the algorithm naturally scales [221] on more machine in order to solve the problem with more vehicles and finer time slices. The numerical assessment also focuses on the impact of stochastic obfuscation on the performance of the algorithm. Practically, this corresponds to adding noise to the responses of the devices participating in the distributed gradient ascent in order further enforce privacy. We show that, with enough vehicles, one can obtain an accurate solution while guaranteeing the eavesdropper cannot learn the optimal planned actions drivers will undertake once the gradient method has converged. The efficiency of these algorithmic and theoretical contributions to fleet allocation optimization is confirmed by numerical work conducted with on actual data collected in New York for Uber ride requests. In this section we confirm the theoretical convergence properties proven above by multiple numerical experiments based on actual demand data extracted thanks to the Freedom of Information Act in New York.

Convergence towards a distribution of vehicles tracking the demand

We now demonstrate that our approach is effective with field data and is able to match the spatial and temporal distribution of demand for mobility as a service with the offer. We illustrate this with two different fleet sizes of 10^3 and 10^4 vehicles in Figures 4.7 and 4.8 respectively. The first set of plots in the former show that the offer tracks the demand in the planning that the algorithm has converged to. In the later we show how a simple map of λ^* can be interpreted as a price multiplier and formulates a coordinating scalar map that protects the privacy of drivers. In the figures below, we focus on normalized distributions of offer and demand as our algorithm is used with different fleet sizes which naturally leads to different absolute levels of offer. In particular we check that the distribution of vehicles the algorithm converges to does match that of the demand for rides. This is observed in Figure 4.7 which confirms the theoretical guarantees given above.

Influence of noise

This section focuses on showing the robustness of the scheme with respect to noise added in the broadcasts as in equation (4.7). Adding different Laplace noises in multiple gradient descents as in Figure 4.9 shows that the scheme we present is robust to perturbations. Indeed, the randomness in distributions of L_2 norm between offer for rides and demand does not explode as the noise becomes higher in variance.

Increasing the number of vehicles and obfuscation by the crowd

As demonstrated in the theoretical analysis of privacy through obfuscation, the higher the number of vehicles, the higher the number of iterations the stochastic gradient method can go through without revealing too much of the actions the vehicles will undertake. We test the implications of the analysis in a numerical experiment in which we increase the number of iterations linearly with the number of vehicles in the fleet as prescribed in (4.8) for the worst case scenario of $\gamma = 1$. Figure 4.9 shows how with a higher number of vehicles the supplementary number of iterations does enables quite a substantial improvement. This improvement is not only due to the higher availability of cars as the L_2 distance between offer and demand steadily decreases until the end of each gradient method we present.

Influence of the model parameters

The previous paragraph demonstrated that the method we present is able to reach an optimal schedule in spite of only being allowed to compute a limited number of gradient method steps to preserve privacy. It can be observed on Figure 4.7 that if we only use 1000 vehicles the tracking of the demand by the agent presence probability is only approximate. In this section, Figure 4.10 shows that this matching of offer with demand can be improved as soon as we consider more vehicles (2000 in this paragraph). Indeed, as we increase the number of vehicles, more degrees of freedom are available and demand for rides can better be answered. The remainder of this section therefore focuses on three questions that still need to be answered. First, we need to characterize the influence of the number of spatial cells. After that, we will analyze how the number of discretization time steps influence impacts the convergence towards a solution. A third concern we address is that of being able to compute a solution at a speed adapted to the time discretization resolution. We show that coarser models allow for faster computation of a less accurate solution. We also demonstrate that over-simplification is not necessary as we are able to compute a solution in less seconds than there is time in a time step with a fined grained model. In this model, spatial cells are only the size of a few city blocks and time steps only have a duration of 15 minutes.

Influence of the spatial discretization granularity

In Figures 4.8, 4.7, 4.10, a discretization grid with small cells was used. More precisely, each cell is 0.41 km wide and 0.93 km tall. A finer spatial discretization resolution is problematic in two aspects. As we target the average predicted demand, if fewer observations are available for an element of the state-space of the demand, more noise impacts our estimates. Also, the scale needs to be adapted to the dimensions of the city as we need each discretization cell to feature at least one street for it to be practically used by the drivers. As in Figure 4.4, some cells in central park only feature a single street and most cells only comprise of 10 city blocks. Considering a coarser spatial discretization will have two positive effects on the solution that will be found. The estimates for demand will be more robust because we will leverage more information for each cell. The computational burden will also be lighter for each sub-problem that needs to be solved in the distributed algorithm we design. An obvious negative effect will obviously be the illusion that the problem is easier to solve when it fact this is only a direct consequence of using a less detailed model. The convex problem we solve with a coarser spatial grid at the level of each individual agent is by definition less complex but the schedule we obtain fits the demand profile less accurately. In Figure 4.11, we indeed show that we converge faster with a discretization grid that is ten times as coarse than the initial grid we used.

Influence of the temporal resolution

The influence of the temporal resolution is similar, from an accuracy and computational complexity standpoint, to that of the spatial scale. Figure 4.11 clear demonstrates that it is much faster to converge with a solution with coarser time steps. This solution will only be adapted to a coarser approximation of the demand unfortunately. Based on the empirical observations presented in [54], the mode of the pace in New York is 8 minutes for a single mile. A temporal resolution of 15 minutes is therefore reasonable choice given the characteristic time needed by an agent to traverse a fine spatial cell blocks in the presence of congestion. A given agent will be unlikely to "jump" over a spatial cell in less than a time step.

Computing a solution fast enough to be consistent with the temporal resolution of the model

Another critical aspect of the influence of the temporal resolution is related to an implicit computation speed requirement which is necessary to compute solutions for them to be relevant to the time steps of the optimization we are attempting to solve. Having chosen a time step of 15 minutes, we need to be able to find a solution to our optimization problem in less than 900 seconds for it to be even relevant if unexpected perturbations to the demand require a quick re-computation of the solution. Figure 4.11 shows that the finest model we consider is solved in less than 900 seconds.

This result demonstrates that implementing the scheme on smartphones while providing online adaptation capabilities is feasible. Indeed, as only 400 Amazon EC2 CPUs were used in the experiment featured on Figure 4.11 and a general purpose convex optimization solver was employed to solve the agent-level optimization problems in the multiple steps of the algorithms. The large number agents plays a favorable role here. Our algorithm being scalable with respect to the number of vehicles, if more agents are present, using more CPU cores (practically the drivers' smartphones) will be sufficient. Moreover, more averaging of artificial privacy enforcing noise will occur in the algorithm and the convergence will be more stable. One EC2 Compute Unit offered for rent by this service provider runs on commodity hardware and is the equivalent CPU capacity of a 1.0-1.2 GHz 2007 Opteron or 2007 Xeon processor. Showing that we can allow for each of these to support 5 agents in Figure 4.11 and yet be able to compute a solution in less than 15 minutes therefore indicates that this scheme can be use to have a reactive distributed control scheme supported by the smartphones of the drivers.

Therefore, with the method we propose, drivers can use a simple smartphone and do not need share private information to optimize their actions as a group. The group will however be able to optimize its actions in less than 15 minutes and react to changing conditions. This advantageous property is a consequence of the approximate model we use whose assumptions are tailored to the operational setting of MaaS.

Summary of results on distributed optimization for MaaS

After a statistical characterization of the spatio-temporal structure of demand for MaaS in New York, we devised a CIQP to optimize the response to that demand while taken drivers' availability constraints into account. In order to make solving this problem computationally tractable and more tailored to the fact that MaaS companies such as Uber or Lyft do not directly control their drivers, a convex relaxation version of the problem has been used. Using the dual splitting approach paved the way towards a procedure to optimize the demand response that scales with the number of drivers and does not require them to communicate their availability constraints at any point, thereby enforcing strong privacy standards. The method formulates an optimal price multiplier akin to that of Uber or Lyft (so called “surge” or “primetime”) that has the distribution of the fleet converge to a global optimum. A thorough theoretical analysis of the convergence of the dual gradient ascent algorithm we presented proved that the scheme was reliable, robust to perturbations and thereby enables an obfuscation of the optimal driving program that becomes more efficient as more drivers join the fleet. Indeed, the stochastic version of the gradient ascent [170] obfuscates the advised probability of presence of vehicles in order to protect the privacy of the users of the system in the occurence of an man-in-the-middle attack. Our theoretical contribution shows that this strategy converges to the initial optimum with an arbitrary precision provided enough drivers are involved in the optimization. This analysis of learning rates is different from pre-existing work in [58, 34, 232] and highlights the power of the crowd as a privacy preserving mechanism thanks to a thorough comparison of the convergence rates of the crowd and the attacker. Such strong theoretical guarantees are crucial to provide practitioners with an accurate estimate of their computational time.

4.4 Application to load flattening in California

The convex duality based method also successfully applies to the problem of charging electric devices so as to even out the daily energy demand in California. The solution we provide is a privacy preserving communication avoiding algorithm readily implementable in a network of smart electric vehicle chargers. It can reach any arbitrary precision for the common optimization goal while relying on randomly perturbed information at the agent level. We leverage the fact that provided the community is large enough, an averaging effect at the core of gradient method we designed enables the group to learn its global optimum faster than individual information is leaked. A limited number of messages are sent out in the distributed implementation which prevents adversary statisticians from having low theoretical Mean Square Errors for their estimates.

Characterization of load shaping imbalance

The method we develop is generic and we now show how it applies to another heterogenous large scale control problem: load shaping in California.

Setting

As solar power generation grows in California seasonal patterns will dominate energy generation. Most of the solar electrical power is generated at the middle of the day. Unfortunately, this does not correspond to the the evening peak of electrical consumption and creates an imbalance which is forecast to grow considerably [111]. A key insight to provide a solution to the corresponding “duckcurve” problem consists in storing energy during the production peak and releasing it during consumption hours. The algorithm we developed has been successfully applied to solving the load imbalance problem by using batteries of electric vehicles in order to store energy [128, 129]. The resulting distributed control problem is illustrated in Figure 4.12.

System identification

System identification has been conducted in independent work that focused on two aspects. First, forecasting the electrical imbalance as the penetration of solar power generation increases [111]. Second, while electrical vehicles have high capacity batteries some of the charge needs to be reserved for the actual needs of the car owner. The corresponding availability constraints need to be determined. Also, one needs to take into account when the vehicle is parked and can be charged. Such a study has been conducted in [202]. The schedules of availability of cars prescribe the constraints smart-meters can operate with in order to flatten the load imbalance curve. Each constraint set is unique to a car and therefore a source of heterogeneity in the problem. Constraints in this problem are block-wise independent which prompts the use of the technique we tailored to such systems where scalability is mandatory and sensitive information is handled.

The system of interest here consists of tens of thousands of electric vehicles. The algorithm of interest needs to scale with respect to the number of controllers. Instead of enabling scalability with a general purpose scalable algorithm such as ADMM [27] we rely on a tailored method. While the general analysis of the convergence of the method still applies to this particular problem, we enjoy the fact that the dual splitting effectively leverages block-wise independent constraint sets. We can also appreciate how the formation of a price vector indexed through time which gives interpretability to the Lagrange multiplier being iteratively computed by the crowd of controllers.

The Lagrange multiplier gives guidelines as to which financial incentives could be used through the day to even-out electric consumption.

Problem formulation

Let $(o_k)_{k=1\ldots T}$ be the normalized hourly excess of electrical production in California ($T = 24$). The objective here is to use the electric consumption of N electrical vehicles so as to alleviate the "valley" effect [12]. For a given device n, the daily utilization vector, $u_n = (u_{k,n})_{k=1\ldots T}$ is constrained and belongs to a convex closed set $\mathcal{C}_n \subset \mathbb{R}^T$ representing the periods of the day during which the owner is driving the car. The convex program that should be solved collaboratively by the swarm of smart chargers for electric vehicles is

$$\begin{aligned}&\min_{c} \sum_{k=1}^{T} \frac{1}{2}\left(o_k - \frac{1}{N}\sum_{n=1}^{N} u_{k,n}\right)^2 + \frac{1}{N}\frac{1}{2}\sigma \sum_{n=1}^{N} \|u_n\|_2^2 \\ &\text{st}\, \forall n \in \{1 \ldots N\}\,,\ u_n \in \mathcal{C}_n\end{aligned}$$

This formulation takes into account both evening out the electrical excess in its primary objective and regularizing the devices' utilization vectors so as to preserve their life span [12]. It is similar to (4.1) with $\ell_k(x) = \frac{1}{2}x^2$ for any $k \in \{1 \ldots T\}$, $\mathrm{r}_n(u_n) = \|u_n\|_2^2$. Therefore, $m = 1$ is a strong convexity constant of the dual split objective. Similarly an admissible Lipschitz continuity constant for its gradient is $L = 1 + \frac{1}{\sigma}$.

Distributed control scheme for load shaping

The dual splitting technique we present in this chapter applies directly to the problem of load shaping. The strong convexity of the additively binding objective and block-wise independence of the constraints enable the straightforward use of a distributed control scheme. The scheme operates in dual space and converges to a Lagrange multiplier indexed in the time domain which represents how during hours of higher consumption the price of electricity could be increased in order to mitigate load imbalance.

Iteration budget for the noisy holistic descent

From the derivations above, with the assumptions formulated on the attacker's learning rate, the iteration budget is $i^{\max} = \left(\frac{T\kappa\eta^2}{\kappa_{\min}}\right)^{\frac{1}{\gamma}}$. In this setting, one has $A_{\text{hol}} = L = 1 + \frac{1}{\sigma}$ and $B_{\text{hol}}^2 = T\eta^2$. From (4.10), one deduces the following theorem.

Theorem 4.4.1 *The privacy safe precision of the noisy holistic descent is bounded by*

$$\mathbb{E}\left[f\left(\lambda^{(i^{max})}\right) - f\left(\lambda^*\right)\right] \leq 2\left(\frac{\kappa_{min}}{T\kappa}\right)^{\frac{1}{\gamma}} \frac{T}{N}\left(1 + \frac{1}{\sigma}\right)\eta^{2\left(1-\frac{1}{\gamma}\right)} \tag{4.15}$$

If $\gamma < 1$, increasing the magnitude of the noise, η, increases the precision that can be attained with a privacy safe iteration budget. A particular case occurs when the attacker has an optimal learning rate $\gamma = 1$. In such a setting we leverage the fact that the reachable precision is proportional to $\frac{1}{N}$. An attacker targets individuals whereas the gradient descent computes the average optimal response at each measurements. Therefore, as more computation nodes are involved in the scheme, noise is more problematic to the attacker than it is to the collaborating swarm.

Numerical experiments with noisy holistic method

In the following, actual data of utilization pattern of electric vehicles in California gives a set of $3T$ affine convex constraints for each of the N agents. The prescribed step size is $s^{(i)} = \frac{2}{2\left(1+\frac{1}{\sigma}\right)^2+i}$. The noise magnitude η is multiplied by the standard deviation of the primal optimum u^* in order to be of the same scale as the signal it perturbs. On the figures below, the value of η before scaling is denominated "normalized η". We run 100 instances of the gradient descent for different values of N, with different magnitudes of η for $\sigma = 1$. Noise has a Laplace distribution here as in [61]. Only problems with relatively small crowds of agents are considered as the primal solution is also computed to provide a comparison baseline. Satisfying regularization is a second order problem here. While respecting the constraints, the effort vector $\left(\frac{1}{N}\sum_{n=1}^{N} u_{k,n}\right)_{k\in\{1\ldots T\}}$ needs to replicate $(o_k)_{k\in\{1\ldots T\}}$ which is displayed in Figure 4.14. The effort vector resulting from the dual distributed algorithm is compared to that of the primal solution (calculated by CVX). Filled areas represent the bands between percentiles of empirical distributions. Figure 4.15 highlights that, with the same iteration budget (20), having more agents involved in the optimization yields a better replication precision. We show that the effect of averaging has an impact on the gradient based search as well. Figure 4.16 indicates that the empirical MSE is inversely proportional to N for both the objective and the normalized distance to the optimum $\mathbb{E}\left[\frac{1}{N}\left\|u-u^*\right\|_2^2\right]$ (normalizing by N takes into account that u has $N \times T$ elements). Lower value for the latter shows that the averaging gradient helps reach a better precision in both the objective value and the action vector.

Iteration budget for the noisy incremental descent

The iteration budget here is $i^{\text{max}} = N\left(\left(\frac{T\kappa\eta^2}{\kappa_{\text{min}}}\right)^{\frac{1}{\gamma}} - 1\right)$. With the derivations above, $A^2_{\text{inc noisy}} = 2\left(1+\frac{1}{\sigma}\right)^2$ and $B^2_{\text{inc noisy}} = 2\beta^2\left(1+\frac{1}{\sigma}\right)^2 + T\eta^2$ where $\beta^2 = \max\left(\frac{1}{N}\sum_{n=1}^{N}\left\|\lambda^{n,*}-\lambda^*\right\|_2^2, \frac{D_0}{2}\right)$.

Theorem 4.4.2 *The privacy safe expected precision of the noisy incremental descent is bounded by*

$$\mathbb{E}\left[f\left(\lambda^{(i)}\right) - f\left(\lambda^*\right)\right] \leq 2\frac{\left(1+\frac{1}{\sigma}\right)\left(T\eta^2 + 2\beta^2\left(1+\frac{1}{\sigma}\right)^2\right)}{N\left(\left(\frac{T\kappa\eta^2}{\kappa_{min}}\right)^{\frac{1}{\gamma}} - 1\right)} \tag{4.16}$$

If $\gamma < 1$, any precision can be reached while guaranteeing secrecy enforcement. If $\gamma = 1$ the average privacy secure precision increases linearly with N. Such an observation confirms the intuition that, in the incremental descent case, it becomes more unlikely for the attacker to intercept repeated occurrences of the optimal program of a given agent if N is large.

Numerical experiments with noisy incremental method

The same numerical experiments are conducted as for the holistic method. However, the new increased iteration budget is taken into account. Figure 4.15 shows how the incremental is slightly more sensitive to noise than its holistic counterpart. Figure 4.16 shows that, as in the holistic case, the MSE for the objective value and the agents' actions is inversely proportional to N. The higher number of steps in the descent helps reach a value of u closer to the global optimum u^*.

Summary of results on load shaping

For the class of problems considered above, splitting the binding objective thanks to a dual reformulation enabled the full separation of independent block constraints. This result has strong implications in terms of enforcing privacy. The individual constraints of the agents remain local and noise can be added to communications that hinders information leakage. Convergence rates indicate that this approach becomes more efficient as more agents are involved. Theoretical insight has been confirmed by numerical experiments conducted with actual data on electricity production and consumption in California. We do observe, as predicted, that as more agents are involved in the load balancing scheme, we converge to a better solution for a given amount of information leakage. With enough controllers, we can reach an arbitrary level of convergence around the optimum. This new scheme is readily applicable to smart electric vehicle chargers whose information are highly privacy sensitive.

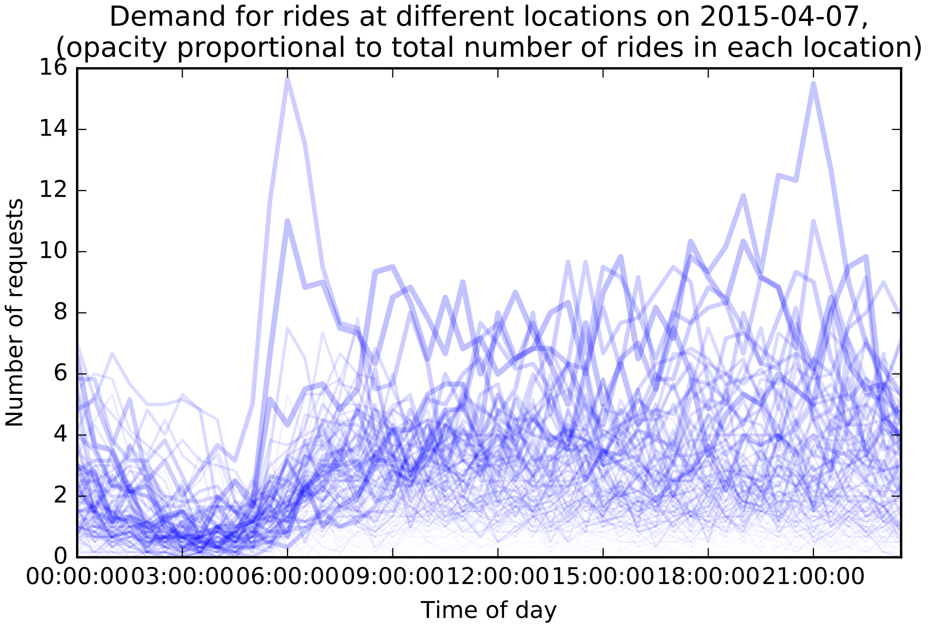
4.5 Conclusion of the chapter

This chapter has shown how heterogenous constraint sets can be unified as part of a communication avoiding scalable distributed control scheme. While each controller had its own unique set of constraints, we employed convex duality to successfully find a common representation in the form of Lagrangian multipliers. We showed that finding a concise set of messages to communicate induced enough robustness in the resulting communications to support noisy broadcast. The procedure lead in particular to a scalable algorithm able to devise a receding horizon MaaS dispatch control scheme for a high temporal and spatial resolution model of New York city although more than a thousand controllers were involved in the distributed control problem. We proved how the system as a whole can learn faster than an attacker can infer individual constraints which turns the concise dual representation into a privacy enforcing summary. The representation we formulate uses convex duality in order to leverage the particular geometry of the control problem of interest in the chapter. In spite of that particularity, it enables statistical efficiency and scalability through communication avoidance.

We also conducted a thorough differential analysis of how much individual level data is revealed in order to make the community converge to optimality. The representation we chose features the interesting property that as the cohort of controllers gets larger, individual messages can get noisier and further preserve privacy while we achieve the same level of performance. The study also shows that the dual representation is tailored to the purpose of collaborative optimization and uses primarily information directly relevant to this purpose.

Just as spectral duality has been used in previous chapter and led to similar conciseness induced robustness, the convex duality we consider achieves both scalability and data efficiency. In the field of distributed control of time series, as with time series analysis, the search for communication avoidance leads to the discovery of a condensed representation of the data on which we can compute the steps necessary to achieve our aim. In this chapter, we considered controllers had separate constraints and additive contributions to the argument of the convex loss characterizing the cyber-physical system.

The next chapter will focus on methods to deal with non-linear interactions between actuations. The tool set we will employ to find a common representation enabling efficient communication between controllers will be radically different. In order to deal with complex non-linear interactions, we will allow our controllers to formulate complex control policies based on expressive neural networks. Neural networks optimized thanks to Policy Gradient methods in Reinforcement Learning lose the convex geometry which enabled the use of the corresponding duality. We will therefore find another strategy to formulate a common representation to the heterogenous control problem under consideration.



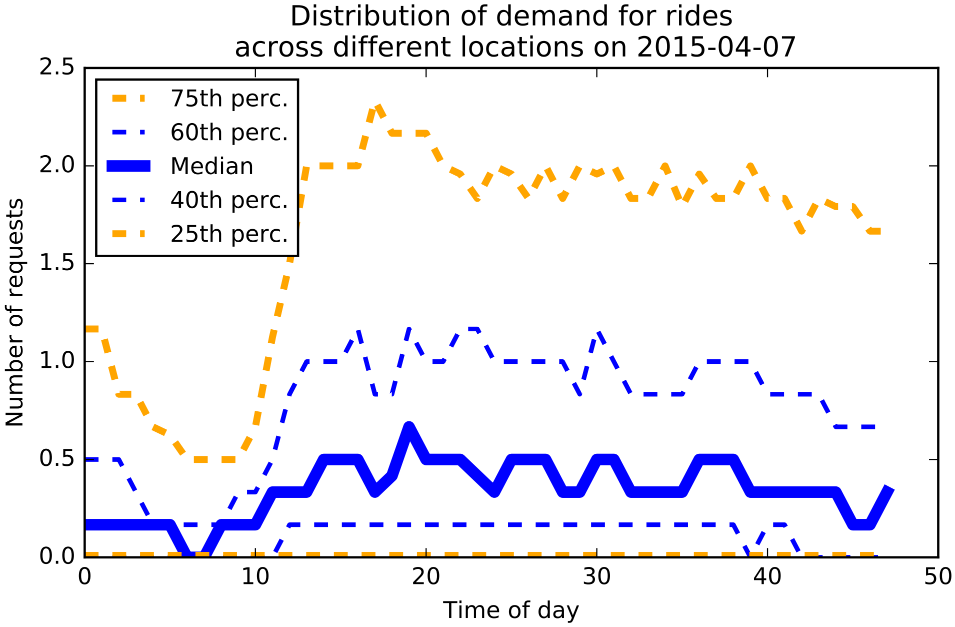


Figure 4.5: **Top:** Demand series for specific locations of NYC on a particular day. Intra-day seasonality appears as a salient feature of the demand of MaaS. For each element of the discretization grid, ride requests are aggregated over 5 minute observation windows and represented with lines whose opacity is proportional to the total number of ride requests observed at the corresponding element of the grid for the entire day. **Bottom:** Average and empirical percentiles computed across all the discretization cells used to represent the state of demand for rides across the city.

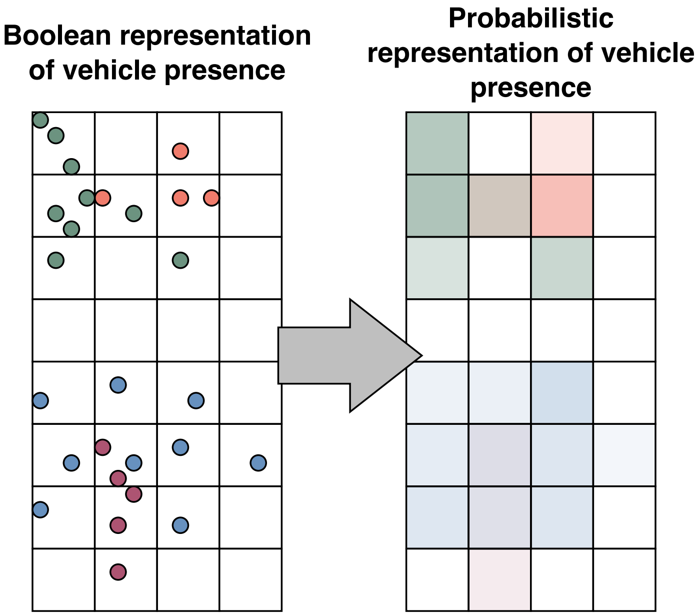

Boolean representation of vehicle presence
Probabilistic representation of vehicle presence

Figure 4.6: Core idea behind the convex relaxation. As there is no direct control of the drivers by MaaS companies, we consider we control a distribution of probability of presence as is done by MaaS companies through the internal use of heatmaps. This corresponds to the intuition that the objective to achieve is finding an optimal incentivization scheme for the drivers. It is key to turn the exact problem into an approximation that can scale with respect to the number of drivers.

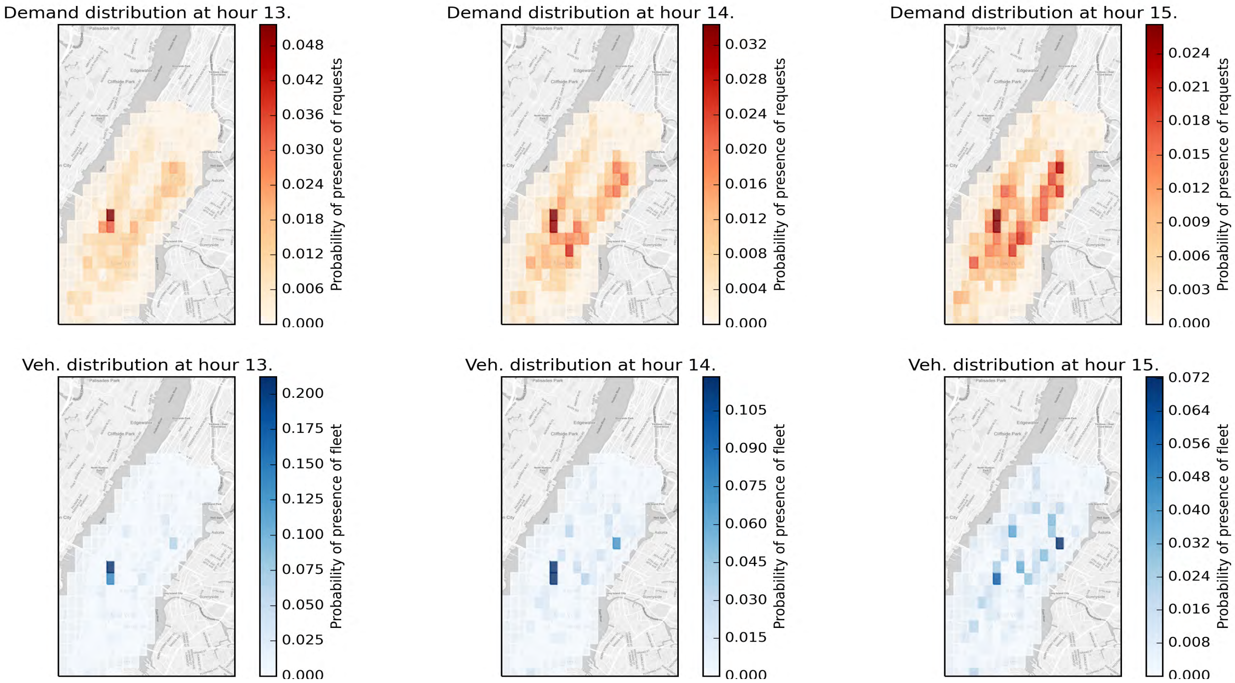

Demand distribution at hour 13.
0.048
0.042
0.036
0.030
0.024
0.018
0.012
0.006
0.000
Probability of presence of requests
Demand distribution at hour 14.
0.032
0.028
0.024
0.020
0.016
0.012
0.008
0.004
0.000
Probability of presence of requests
Demand distribution at hour 15.
0.024
0.021
0.018
0.015
0.012
0.009
0.006
0.003
0.000
Probability of presence of requests
Veh. distribution at hour 13.
0.200
0.175
0.150
0.125
0.100
0.075
0.050
0.025
0.000
Probability of presence of fleet
Veh. distribution at hour 14.
0.105
0.090
0.075
0.060
0.045
0.030
0.015
0.000
Probability of presence of fleet
Veh. distribution at hour 15.
0.072
0.064
0.056
0.048
0.040
0.032
0.024
0.016
0.008
0.000
Probability of presence of fleet

Figure 4.7: **Top:** Quantized density Uber ride requests (1 hour intervals). **Bottom:** Fleet density optimized under driver availability and network connectivity constraints (1 hour intervals). Best viewed in color. Numerical results of our privacy preserving demand tracking algorithm for April 7^{th}, 2014, optimizing for 1000 vehicles. The vehicles belonging to the mobility on-demand pool obey to availability constraints. We also take into account the desire for drivers to minimize the distance they travel. The resulting optimized distribution of vehicles (number of vehicles per cell normalized by the total number of vehicles available) accurately tracks the distribution of demand for ride requests (number of ride requests per cell normalized by the total number of ride requests). Comparing the left and right columns shows the ability of our solution to track the demand for rides with the distribution of vehicles despite hard constraints on the offer in terms of available drivers. The constraints make a perfect match impossible but we find the closest solution given the variable and limited supply of drivers.

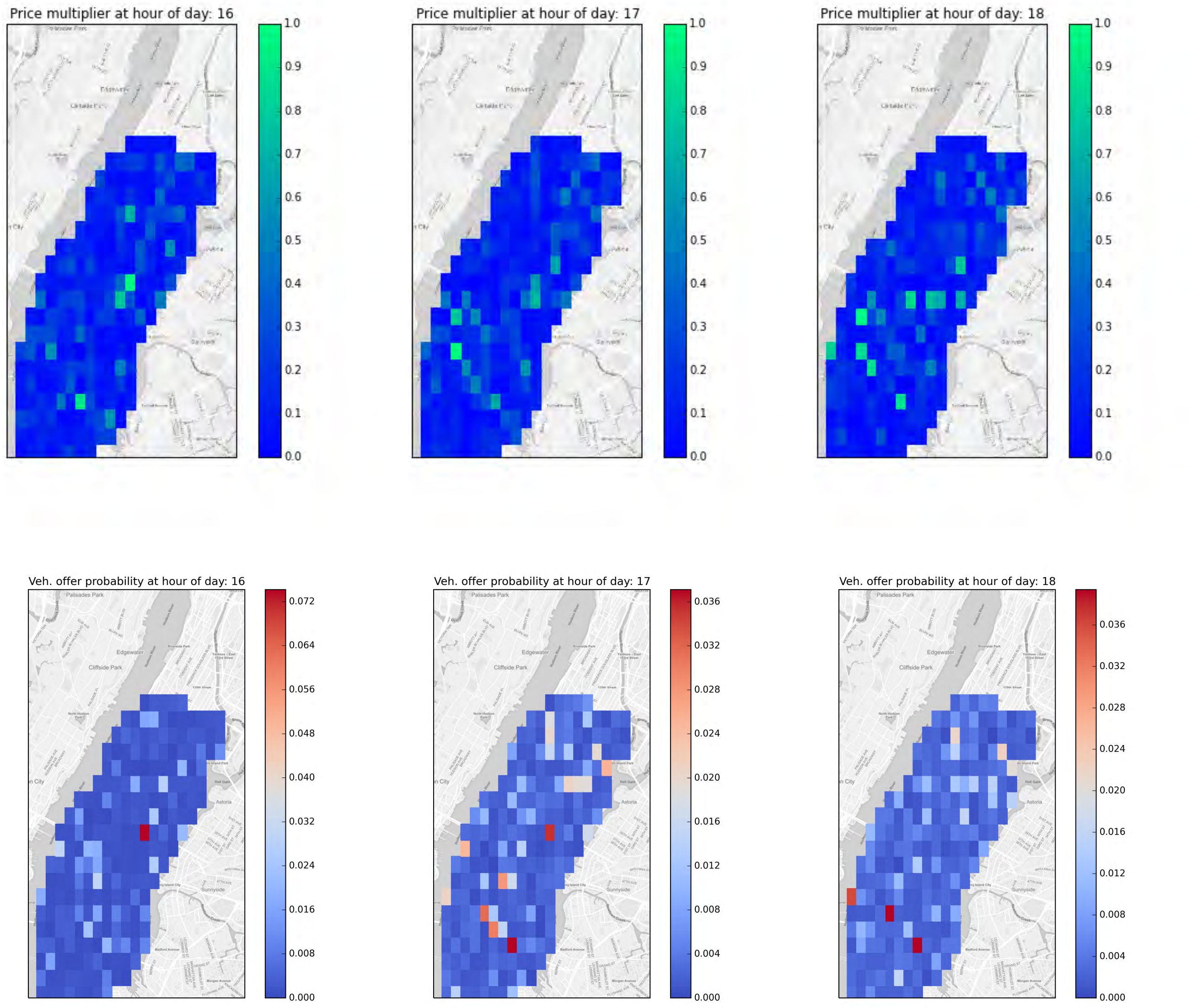


Figure 4.8: **Top:** Lagrangian price signals (1 hour intervals). **Bottom:** Fleet density optimized under driver availability and network connectivity constraints (1 hour intervals). Best viewed in color. Numerical results of our privacy preserving demand tracking algorithm for April 7^{th}, 2014, optimizing for 10000 vehicles. We show here that one of the most compelling outcomes of the dual-splitting approach is the convergence in the dual space towards a Lagrangian variable that can be interpreted, after renormalization, as a price multiplier. On these two series of plots one can appreciate how a single signal broadcast to all the devices leads to a scalable convergence of the fleet towards a distribution tracking the multiplier. Note that the price multiplier drives the fleet towards locations where it is higher.

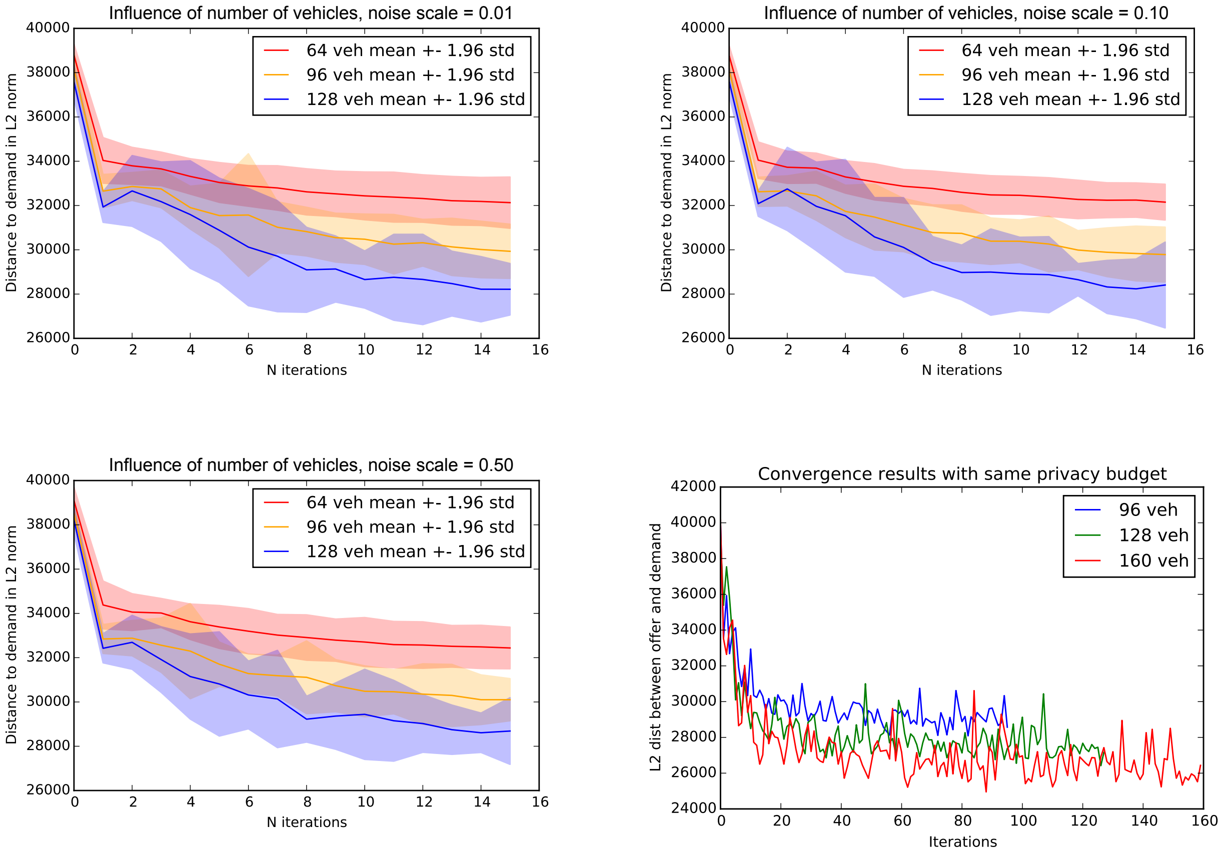
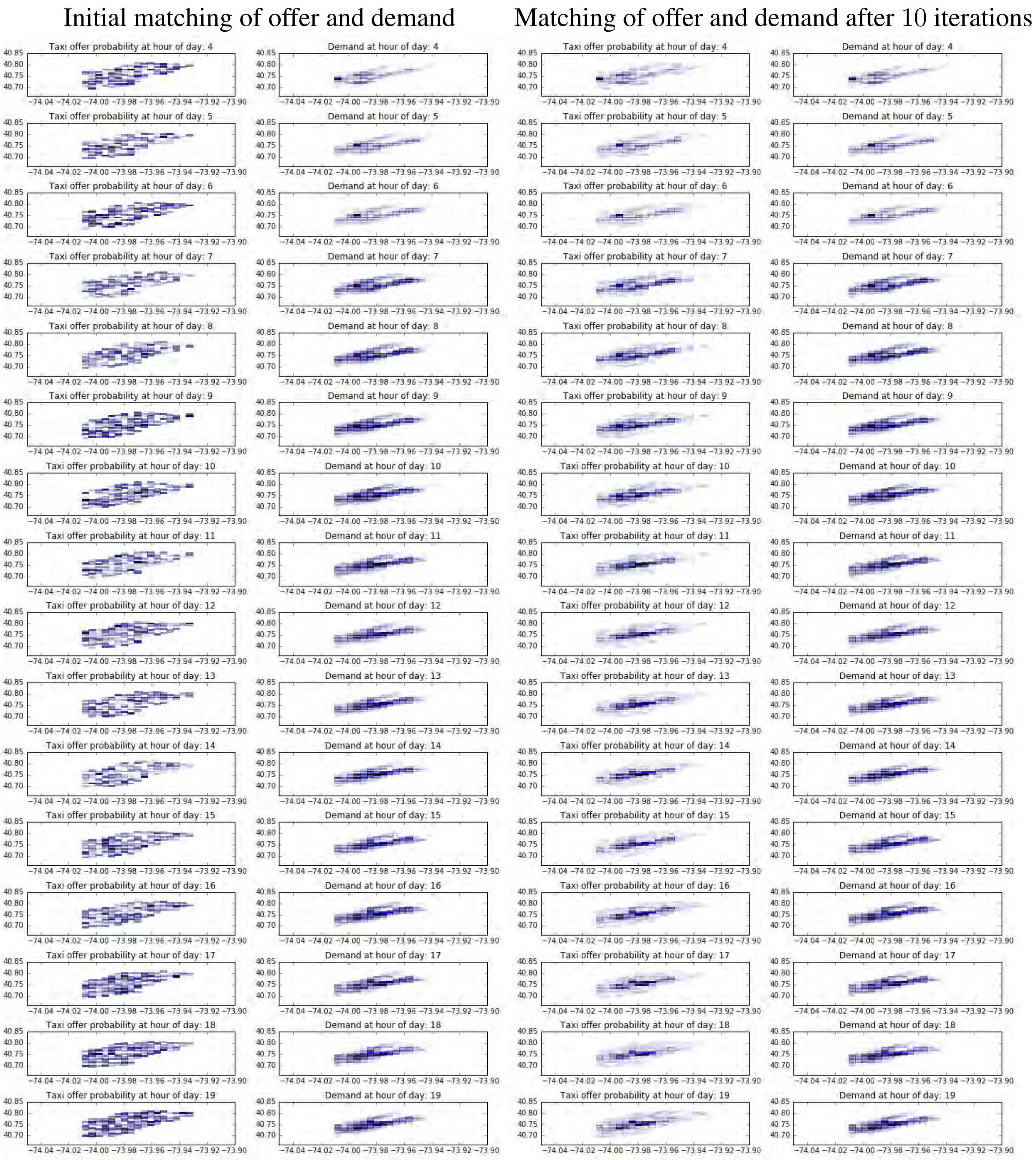


Figure 4.9: Best viewed in color. **First three plots:** We show how adding noise to the gradient descent in the dual, therefore making it stochastic, does not hamper the convergence towards an optimal Lagrangian price and therefore a optimal matching between offer and demand. These curves show how, with different Laplace noise magnitudes and different fleet sizes, convergence to an optimal matching of offer and demand still occurs. **Bottom left right:** We show the convergence results on single runs for a variable fleet size. With a larger fleet size, the obfuscation by the crowd of the information entailed in the messages shared to conduct the stochastic gradient method in the dual space allows for a better matching between offer for rides and demand.



Initial matching of offer and demand

Matching of offer and demand after 10 iterations

Figure 4.10: **Left:** comparison of the offer initially computed with the demand for rides. **Right**: comparison of the offer obtained after 10 steps of gradient method with the demand for rides. After 10 iterations of the gradient method we introduced, the availability of 2000 vehicles considerably improves the quality of the solution we find to match the demand for rides. The discretization is here hourly, and the spatial scale as fine as an area representative of a few streets. The results were computed for April 9th 2013.

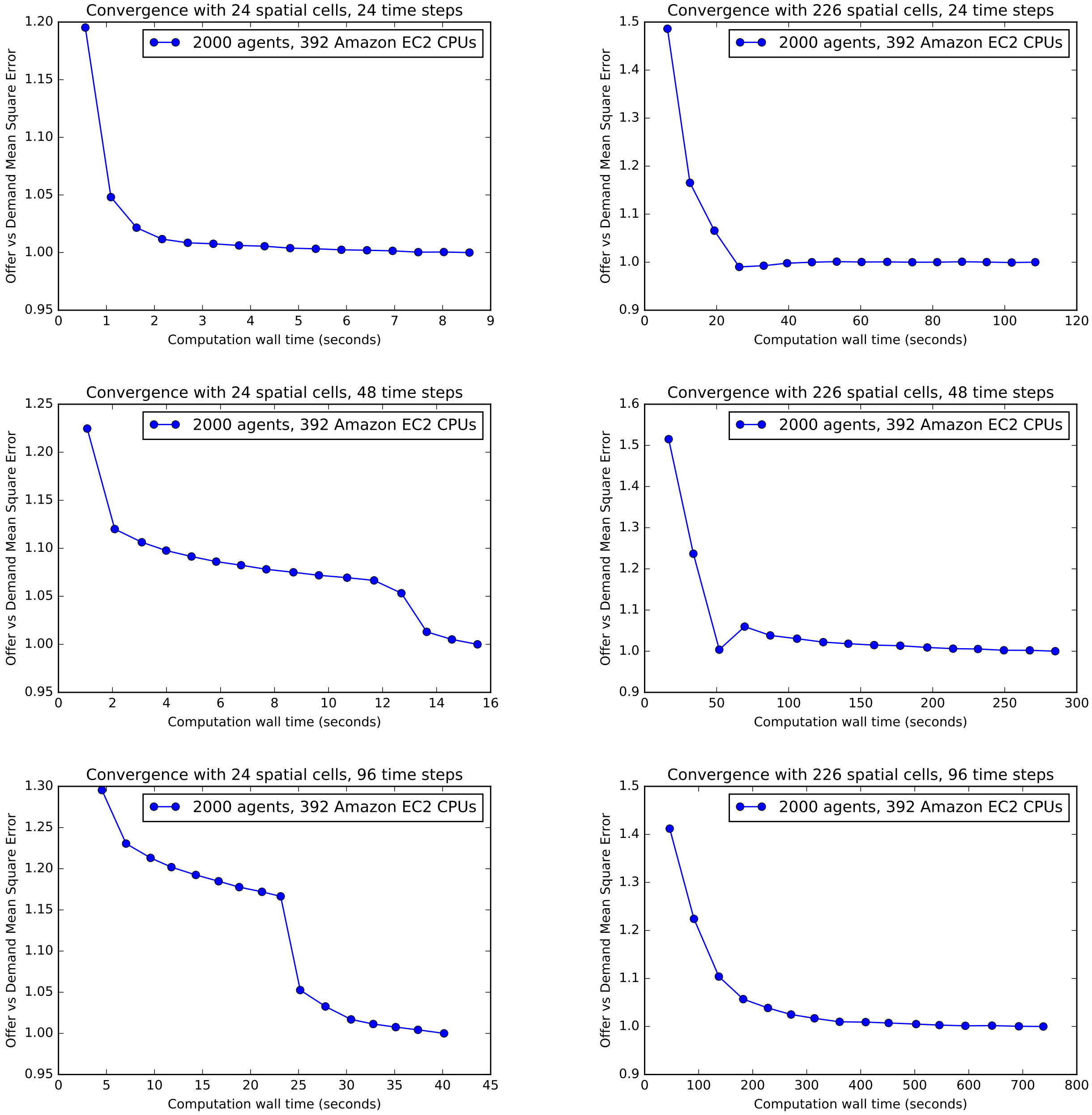


Figure 4.11: Influence of the discretization resolution, in space and time, on the convergence speed of the distributed privacy preserving algorithm. In this experiments, 2000 vehicles are considered and only 400 Amazon EC2 CPUs are used. Here, even with additive noise on communications (privacy preserving Laplace noise with scale 0.1) the schedule can be optimized with a fine spatial resolution and temporal resolution of 15 minutes in less than 5 minutes. This implies that, even with a very detailed model, we can recompute solutions fast enough to adapt to unexpected conditions.

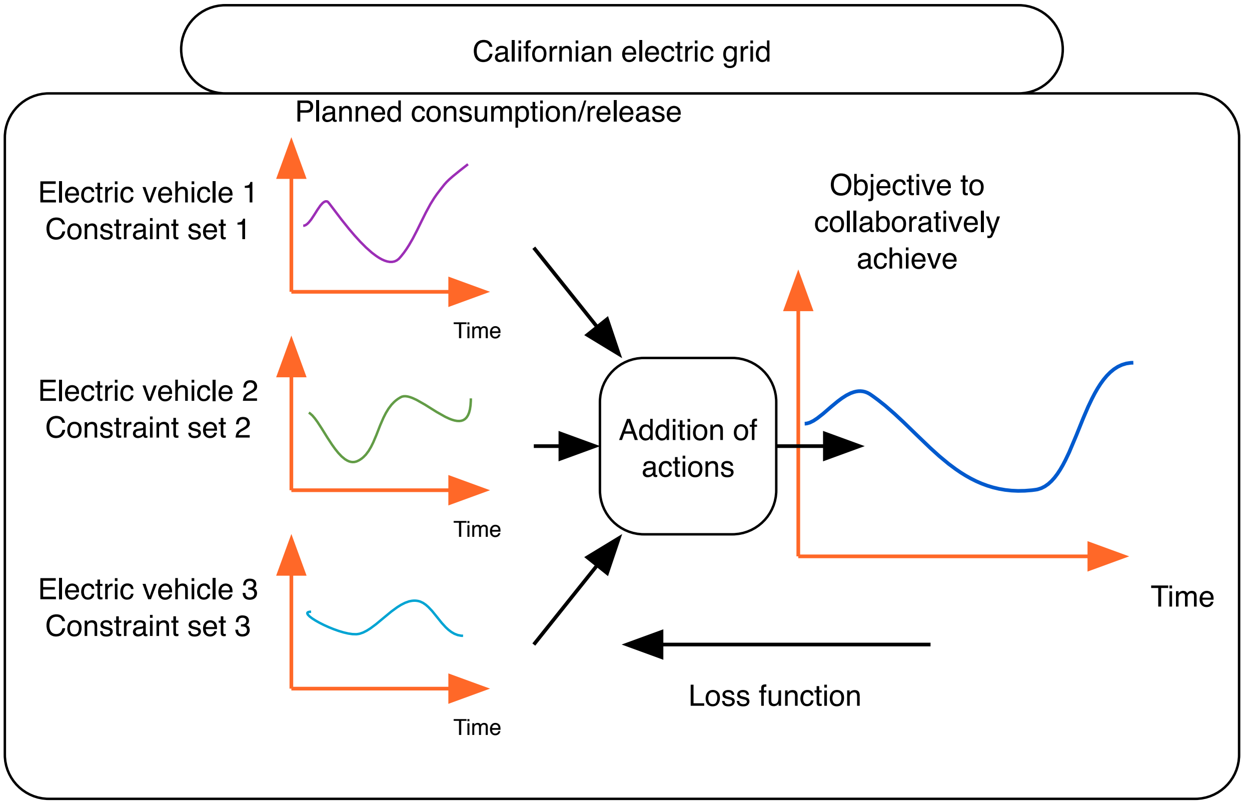


Figure 4.12: Load shaping constrained optimization problem. The charging and utilization schedules of each vehicle as well as the storage capacity and speed of the battery constitute an individual independent constraint set. By choosing to store or release energy the controllers can additively even-out electric load imbalance at the scale of California.

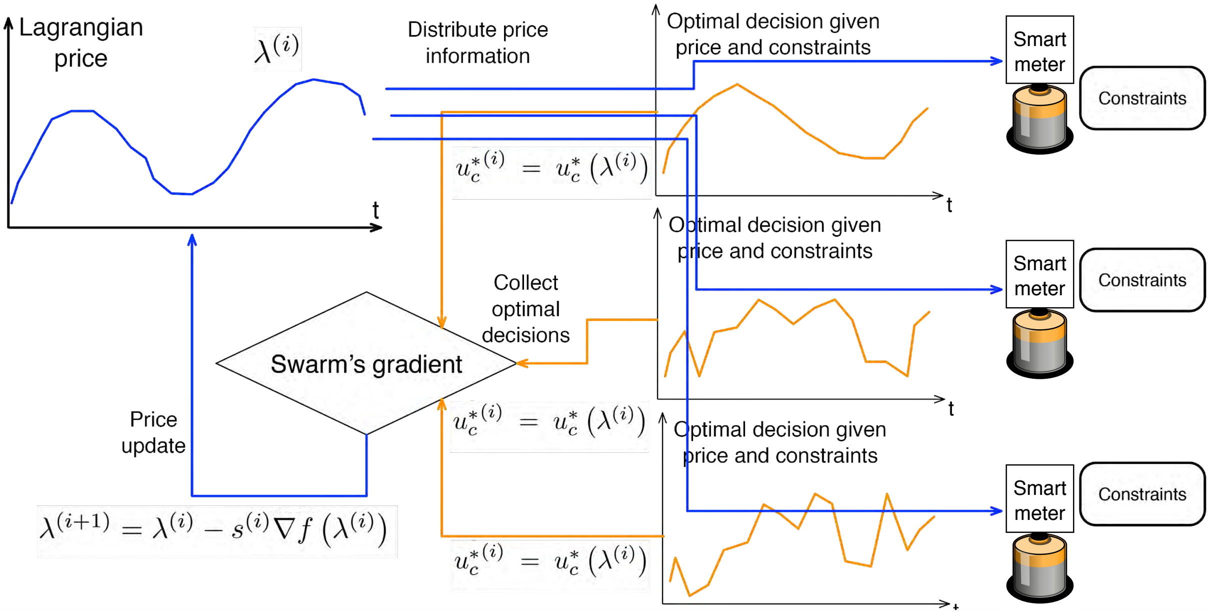

Lagrangian price
$\lambda^{(i)}$
t
Distribute price information
Optimal decision given price and constraints
Smart meter
Constraints
$u_c^{*(i)} = u_c^*\left(\lambda^{(i)}\right)$
t
Optimal decision given price and constraints
Smart meter
Constraints
Collect optimal decisions
Swarm's gradient
$u_c^{*(i)} = u_c^*\left(\lambda^{(i)}\right)$
t
Price update
Optimal decision given price and constraints
Smart meter
Constraints
$\lambda^{(i+1)} = \lambda^{(i)} - s^{(i)}\nabla f\left(\lambda^{(i)}\right)$
$u_c^{*(i)} = u_c^*\left(\lambda^{(i)}\right)$
t

Figure 4.13: Distributed algorithm to solve the issue of electric load imbalance in California.

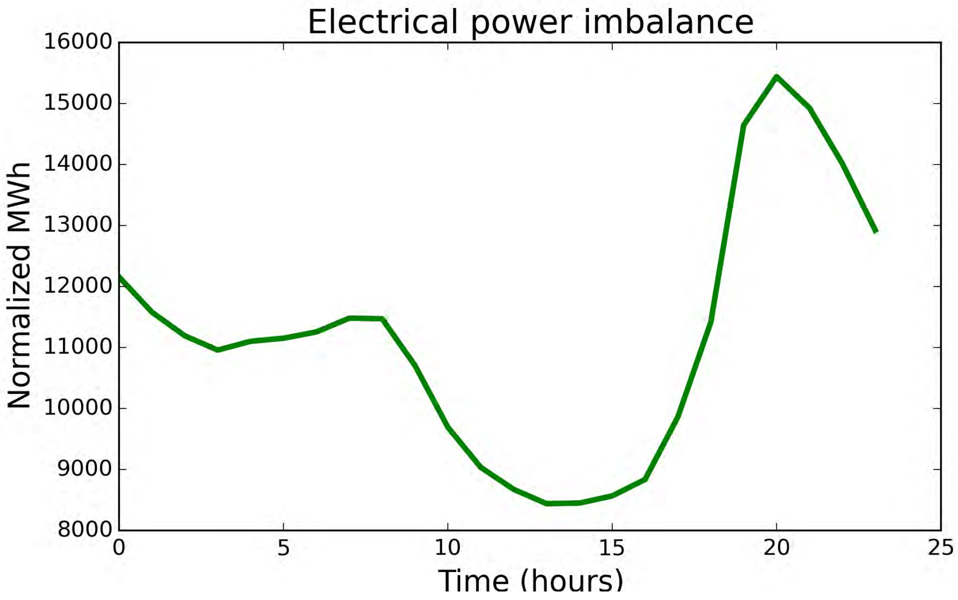


Figure 4.14: Objective to collectively replicate to mitigate electrical power imbalance.

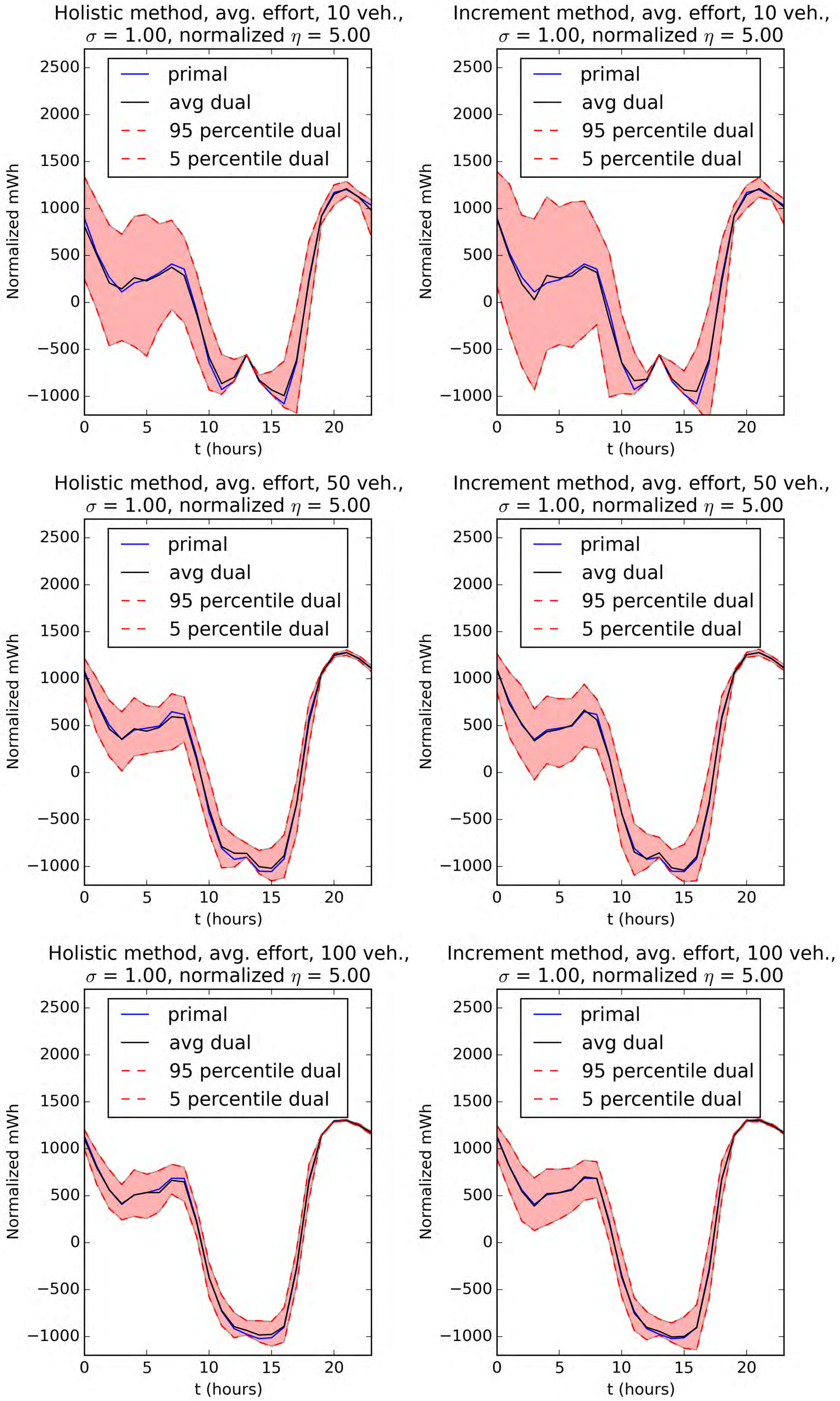


Figure 4.15: Errors in aggregated load balancing effort.

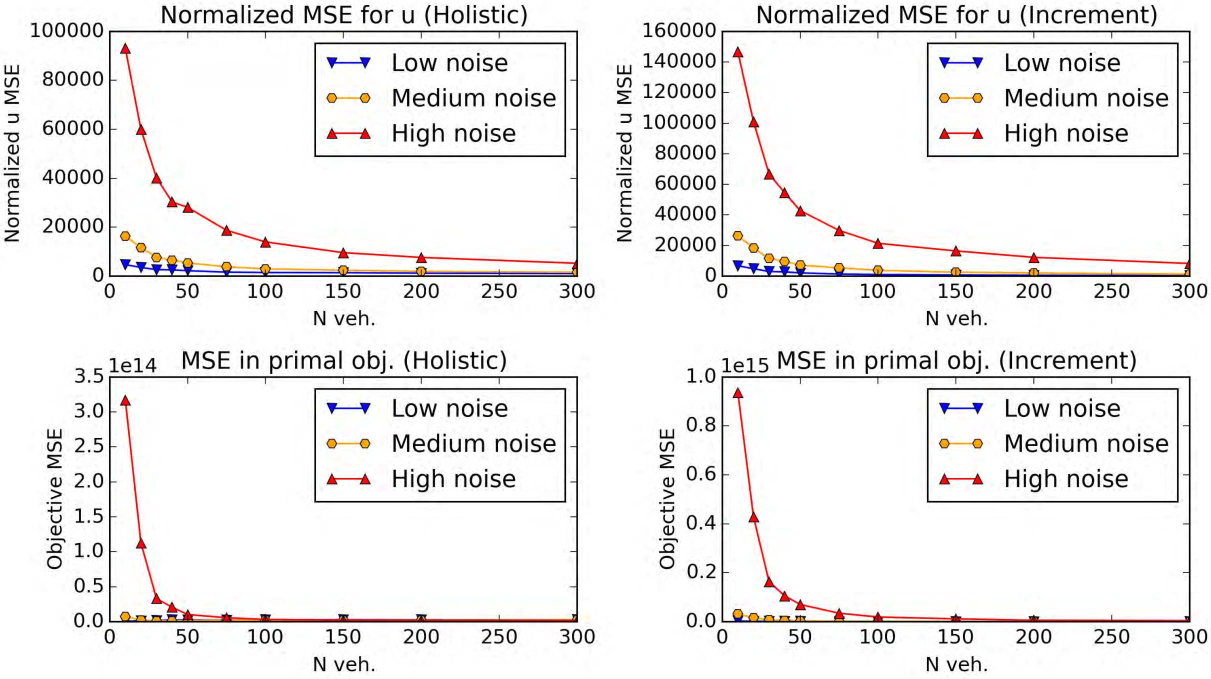


Figure 4.16: Errors in response and objective for electrical load balancing.

Chapter 5

Sharing knowledge across models and controllers

The present chapter shows how the recent breakthroughs in Reinforcement Learning (RL) that have enabled robots to learn to play arcade video games [160], walk or assemble colored bricks [136], can be used to perform other tasks that are currently at the core of engineering cyber-physical systems. We present the first use of RL for the control of systems modeled by discretized non-linear Partial Differential Equations (PDEs) and devise a novel algorithm to use non-parametric control techniques for large multi-agent systems.

We employ a regularization scheme tailored to the collaborative control setting that formulates a representation enabling distributed controllers to exchange the experience they get as they learn by trial and error with RL.

The previous chapters of the thesis all leveraged a specific dual representation to their primal problem of interest in order to great a homogenous representation for a distributed set of heterogenous problems.

In the present setting, we consider controllers maximizing the level of service on a system where congestion occurs. While each controller is dealing with a specific queue that needs to go through a section of freeway and is located at a specific location with particular dynamics, they all share the same environment and fundamental physical behavior.

There exists a common denominator to the dynamics of the sub-part of the system each controller actuates as freeway dynamics follow general patterns that are accurately represented by PDEs such as the Aw-Rascle-Zhang system of the second chapter. Therefore, an opportunity to better control the system with RL consists in enabling the learning control agents to share knowledge while at the same time giving them enough liberty to adapt to the idiosyncrasy related to being spatially present at a certain location in the system.

We show in the present chapter that the parameters learned by a controlling neural network represent, as they are being shared across agents, an opportunity to achieve better control performance. The parameters being shared have a small size that should be compared the amount of experience accumulated before updating model parameters with a Stochastic Gradient Descent method. Also, although controllers may have different inputs, long as they share a layer with common size and purpose, the sharing can occur.

Based on these elements, we argue that we can use the direct values of a controlling model to formulate a homogenous representation enabling agents with heterogenous control problems to share knowledge.

We do not obtain analytical guarantees in the present chapter. However, we demonstrate that our model-free controllers can compete with state of the art methods calibrated with perfect knowledge of the system we simulate.

The absence of modeling assumptions makes the use of Fourier, Laplace or convex duality difficult. However, as we use model parameters to unify and condensate distributed heterogenous control experiences, we leverage the weights of the neural network we consider as an advantageous representation. The case for a model-free approach is made clearer in the chapter as we show that we can learn to identify and control a system at once while keeping the ability to operate on complex non-linear and changing dynamics.

Our choice of representation leads to a method that is simple to implement, communication avoiding and universally applicable as no modeling assumptions are made. The main burden is now on the computational side as we train complex neural policies that need many different trials to learn. Fortunately, such a computational burden is no longer a challenge thanks to the emergence of faster hardware.

Cyberphysical systems (e.g., hydraulic channels, transportation systems, the energy grid, electromagnetic systems) are commonly modeled by PDEs which historically have been a reliable way to enable engineering applications in these domains. However, it is known that the control of these PDE models is notoriously difficult. We show how neural network based RL enables the control of discretized PDEs whose parameters are unknown, random, and time-varying. We introduce an algorithm of Mutual Weight Regularization (MWR) which alleviates the curse of dimensionality of multi-agent control schemes by sharing experience between agents while giving each agent the opportunity to specialize its action policy so as to tailor it to the local parameters of the part of the system it is located in.

A discretized PDE such as the scalar Lighthill—Whitham-–Richards (LWR) PDE [141] – briefly mentioned in the second chapter – can indeed be considered as a macroscopic freeway traffic simulator and which presents the most salient challenges for learning to control large cyberphysical system with multiple agents. We illustrate the incredible potential of neuromorphic RL in non-parametric control of cyberphysical system through the specific example of transportation using the common Lighthill–Whitham–Richards (LWR) [141] scalar hyperbolic PDE. We consider two different discretization procedures and show the opportunities offered by applying deep reinforcement for continuous control on both. This illustrative example was chosen because it is complex and expressive enough to be representative of a much broader class of PDEs (Navier-Stokes, Maxwell, Saint-Venant, etc.).

Using a neural RL PDE controller on a traffic flow simulation based on a Godunov discretization [82] of the of the San Francisco Bay Bridge we are able to achieve precise adaptive metering without model calibration thereby beating the state of the art in traffic metering. Furthermore, with the more accurate BeATS [17] simulator we manage to achieve a control performance on par with ALINEA [175], a state of the art parametric control scheme, and show how using MWR improves the learning procedure. The method is demonstrated with a traffic flow algorithm implemented for I80 on the San Francisco Bay Bridge, to achieve adaptive metering without model calibration (hence beating the state of the art in traffic metering).

Background on the work

In the United States alone, the cost of direct and indirect consequences of traffic congestion was estimated to 124 billions USD in 2013, this cost taking the form of time spent by commuters in traffic jam, air pollution, accidents, etc. It represents almost 1% of the country's GDP, and is expected to grow by 50% within the next 15 years. Dealing with this issue is becoming a priority of government agencies, as the U.S. Department of Transportation Budget rose to almost 100 billions USD in 2016. In this context, any improvement to travel times on highways can lead to tremendous nationwide and worldwide improvements for the economy and the environment.

Maintaining and building road infrastructure, as well as urban planning are the most obvious ways to adapt the traffic network to the ever growing demand for mobility. However, changing the network architecture can only occur seldom and at an expensive cost.

Control of traffic flow is an alternate approach to addressing this issue as it aims at using existing infrastructure more efficiently and adapt it dynamically to the demand. In this chapter, we introduce new techniques for traffic control based on advances in Reinforcement Learning (RL) and Neural Networks. As opposed to most commonly used approaches in traffic control, we want to achieve control in a model-free fashion, meaning that we do not assume any prior knowledge of a model or the parameters of its dynamics, and thus do not need to rely on the expensive and time consuming model calibration procedures.

Recent developments in Reinforcement Learning (RL) have enabled machines to learn how to play video games with no other information than the screen display [160], remarkably beat champion human players at Go with the AlphaGo program [210], or complete various tasks including locomotion and simple problem solving [136, 69, 57]. The advent of policy gradient-based optimization algorithms enabling RL for highly-dimensional continuous control systems as can be found in [135, 205], has generalized model-free control to systems that were characteristically challenging for Q-learning. Q-learning approaches, although successful in [160] suffer from a curse of dimensionality in continuous control if they rely on discretizing the action space.

In this chapter, RL trains a traffic management policy able to control the metering of highway on-ramps. The current state of the art Ramp-Metering policy ALINEA [175] controls incoming flow in order to reach an optimal density locally. This optimal density depends on the model used and has to be manually specified to have an optimal control. Recently, non-parametric approaches based on Reinforcement Learning such as [67] or [195] have been proposed to achieve ramp-metering, but face two main limitations. These methods are not scalable beyond a few on-ramps, and limit traffic management to on-ramp control.

We introduce a way to learn an optimal control policy with numerous agents, and demonstrate the flexibility of our approach by applying it to different scenarios. The following contributions of our work are presented in this chapter:

- We introduce a framework to use RL as a generic optimization scheme that enables the control of discretized Partial Differential Equations (PDEs) in a robust and non-parametric manner. To our knowledge, this is the first use of RL for control of PDEs discretized by finite differencing. Discretized non-linear PDEs are notoriously difficult to control if the difference scheme used is non-smooth and non-continuous, which is usually required to capture nonlinear and non-smooth features of the solution (as is the case here).

- In the case of PDEs used to model traffic, we demonstrate on different examples an extensive control over boundary conditions as well as inner domain for the first time in a non-parametric way. We showcase the robustness of the approach and its ability to learn from real-world examples by assessing its performance after an extremely noisy training phase with stochastic variations in the underlying PDE parameters of the model used

- We introduce an algorithm to train Neural Networks that we denote Mutual Weights Regularization (MWR) which enables the sharing of experience between agents and specialization of each agent thanks to Multi Task Learning [32]. MWR is a Neural Network training approach that allows Reinforcement Learning to train a policy in a multi-agent environment without being hampered by a curse of dimensionality with respect to the number of agents. Applied to the actual traffic control problem of “Ramp metering”, our model-free approach achieves a control of a comparable level to the currently used and model-dependent implementation of ALINEA which constitutes the state art and was in our case calibrated by an world-wide renowned traffic engineer.

We first present the PDEs used to simulate traffic and introduce the generic PDE discretization scheme. A first simulator based on a Godunov scheme [82] is used to demonstrate the efficiency of our approach on multiples situations. The Berkeley Advanced Transportation Simulator (BeATS), a state of the art macroscopic simulator implementing a particular instantiation of the Godunov scheme sometimes referred to as the Cell Transmission Model [164] and used in traffic research [233] is also introduce, as we use it for our final benchmark. Traffic control is presented in the form of a Reinforcement Learning problem and we present the MWR algorithm to mitigate the issues arising from the high dimensionality of the problem. We eventually present the results we achieve, and compare our algorithm to the pre-existing state of the art techniques. This state of practice reference, an ALINEA control scheme calibrated by traffic engineers at California PATH can be considered representative of state of the art expert performance. To the best of our knowledge, we are the first to present a non-parametric scalable method calibrated with RL that performs as well as the pre-existing parametric solutions provided by traffic engineering experts.

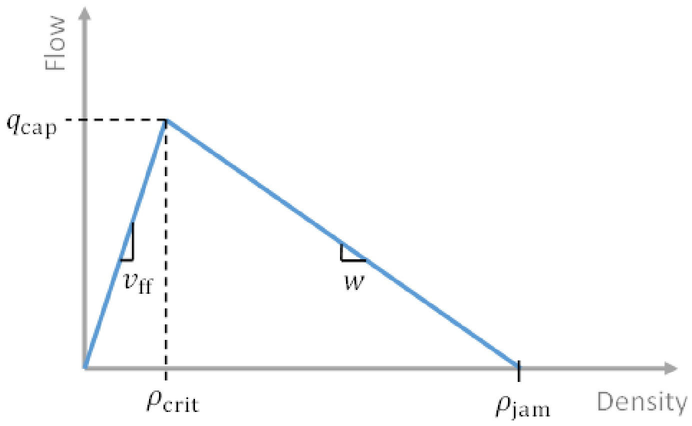


Figure 5.1: Example triangle fundamental diagram.

5.1 Models and methodology

Highway traffic PDE

A highway vehicle density may be modeled by a Partial Differential Equation (PDE) of the following form:

$$\partial_t \rho + \partial_x F(\rho) = 0. \qquad \text{(LWR PDE)}$$

For a given uniform vehicle density ρ, $F(\rho)$ is the flow of the vehicles (in vehicles per time unit); F is the fundamental diagram, its maximum is called the critical density, and corresponds to the optimal density to maximize the flow.

F usually has the following typical shape:

- When the density ρ is lower than the critical density ρ_{crit}, the vehicle flow increases with the density
- When the density exceeds this value, congestion appears, and adding more vehicles actually reduces the flow

Control of PDEs with reinforcement learning

Non-parametric control of PDEs takes the form in the present chapter of a Markov Decision Process (MDP) which is formally introduced below in 5.3. The PDE in its discretized form devises a transition probability P between two different states of the system. Solving a MDP is a known procedure if the transition probability P is known beforehand, using techniques such as Dynamic or Linear Programming. If P is unknown, it is more challenging. However, it is more appealing a procedure as devising P for discretized PDEs is generally intractable and requires estimates of the parameters of the system. Also, as we operate in a continuous action space we will not consider Q-learning based approaches which are typically challenged by high-dimensionality in that setting. Therefore, policy gradient algorithms present a compelling opportunity as model dynamics are sampled from the simulation trials in algorithms such as [116, 181, 230, 103] and no prior knowledge of the model is necessary to train the policy. This creates a model-independent paradigm which abstracts out the model and makes the approach generic.

Simulation

The experimental method we followed consists of two steps.

A first one uses a coarser less realistic discretization scheme for the LWR PDE named after its inventor: Godunov [82]. We provide more details about this scheme in Appendix 5.2. This step serves the purpose of a proof of concept that discretized PDEs can be controlled by a neural net trained using a RL scheme in completely different situations and with different objectives. Our first contribution is to show that policy gradient algorithms achieve that aim.

In our second step we use a more accurate cell transition model [164] scheme entailed in the state of the art BeATs simulator in order to show that the procedure we present is still valid in a more realistic setting. We also change the tasks we assigned to the control scheme in order to precisely account for the actual needs of traffic management systems used in production. We show that training our neural net based policy by policy gradient methods achieves comparable performance with the state of the art ALINEA control scheme [175] although the former is non-parametric when the latter requires a calibration of traffic related parameters. In both cases, the neural net manages to implicitly learn the intrinsic properties of the road segment under consideration and provide a good control policy.

5.2 Godunov discretization scheme

Because of the presence of discontinuities in their solutions, the benchmark PDE we consider is formulated in the weak sense. We consider an open set Ω in $\mathbb{R}$, and a measurable function $\rho = \rho(t, x)$ is a distributional solution to the system of conservation laws

$$\partial_t \rho + \partial_x F\left(\rho\right) = 0. \tag{5.1}$$

if, for every C^1 function ϕ defined over Ω with compact support, one has

$$\int_{t \in \mathbb{R}, x \in \Omega} \left[\rho \partial_t \phi + F\left(\rho\right) \partial_x \phi\right] dx dt = 0. \tag{5.2}$$

The operator F in Eq. (5.1) will be referred to as flux function, also called the “Fundamental diagram” in transportation engineering. The operator F defines entirely the dynamics at stake and therefore is often domain specific. Given an initial condition

$$\rho(0,x) = \bar{\rho}(x) \tag{5.3}$$

where $\bar{\rho}(x)$ is locally integrable, $y : [0,\, T] \times \mathbb{R} \to \mathbb{R}$ is a distributional solution to the Cauchy problem defined by Eq. (5.2) and Eq. (5.3) if

$$\int_{t\in[0,\,T],x\in\Omega} \left[\rho\partial_t\phi + F\left(\rho\right)\partial_x\phi\right] dxdt + \int_{x\in\Omega} \left[\bar{\rho}(x)\phi(0,x)\right] dx = 0. \tag{5.4}$$

for every C^1 function ϕ with compact support contained in $]-\infty,\, T] \times \mathbb{R}$. If y is a continuous function from $[0,\, T]$ into the set of locally integrable on Ω, solves the Cauchy problem above in the distribution sense, ρ is referred to as a weak solution to the Cauchy problem.

The Dirichlet problem corresponding to a boundary condition (as done later in the chapter) can be formulated in a similar manner and is left out of the chapter for brevity. Such a definition of weak solutions is not sufficient to guarantee their being admissible solutions.The entropy condition guarantees the uniqueness to the problem and continuous dependence with respect to the initial data (derivation also left out of the chapter for brevity).

The Godunov's scheme computes an approximate weak solution $\tilde{\rho}$ to the Dirichlet problem Eq. (5.2), Eq. (5.3) with the following recursive equation and Godunov flux G:

$$\begin{aligned} &\tilde{\rho}(n\Delta t + \Delta t, i\Delta x) = \\ \tilde{\rho}(n\Delta t, i\Delta x) - &\frac{\Delta t}{\Delta x} G\left(\tilde{\rho}(n\Delta t, i\Delta x - \Delta_x), \tilde{\rho}(n\Delta t, i\Delta x + \Delta_x)\right) \end{aligned}$$

$$\begin{aligned} G\left(\tilde{\rho}_l, \tilde{\rho}_r\right) =& F(\tilde{\rho}_l) \text{ if } \tilde{\rho}_l > \tilde{\rho}_r \text{ and } (F(\tilde{\rho}_r) - F(\tilde{\rho}_l))/(\tilde{\rho}_r - \tilde{\rho}_l) > 0, \\ & F(\tilde{\rho}_r) \text{ if } \tilde{\rho}_l > \tilde{\rho}_r \text{ and } (F(\tilde{\rho}_r) - F(\tilde{\rho}_l))/(\tilde{\rho}_r - \tilde{\rho}_l) < 0, \\ & F(\tilde{\rho}_l) \text{ if } \tilde{\rho}_l < \tilde{\rho}_r \text{ and } DF(\tilde{\rho}_l) > 0, \\ & F(\tilde{\rho}_l) \text{ if } \tilde{\rho}_l < \tilde{\rho}_r \text{ and } DF(\tilde{\rho}_r) < 0, DF^{-1}(0) \text{ otherwise }. \end{aligned}$$

The Godunov scheme is second order accurate in space. Unfortunately, like most numerical schemes, it is non-differentiable because of the presence of the "if-then-else" statements in its explicit form. Another problematic aspect related with computing numerical weak entropy solutions with most numerical schemes (incl. Godunov) is their relying on a numerical evaluation of F which often takes the form of a parametrized function. The estimation of these parameters is often difficult and it is practically intractable to assess the impact of the parameter uncertainty on the solutions because of the non-linearity, non-smoothness and non-differentiability of the schemes.

5.3 Controlling cyberphysical systems with Neural Networks trained by Reinforcement Learning

The control policy presented is a neural network trained through Reinforcement Learning. In this section, we introduce these notions, precise the implementation, and describe how our approach differs from previously existing ones. We eventually benchmark our results by comparing our non-parametric algorithm to the state of the art parametric equivalent. In order to be able to control a complex cyberphysical system a non-parametric manner, we adopt a Reinforcement Learning formulation.

Reinforcement Learning formulation

RL is concerned with solving a finite-horizon discounted Markov Decision Process (MDP). A MDP is defined by a tuple $(S, A, P, R, P_0, \gamma, T)$. The set of states is denoted S and will typically be $\mathbb{R}^d$ in our instance where d is the number of finite differences cells as in [82]. The action space A will correspond to a vector in $\mathbb{R}^d$ which represents the vehicular flow that the actuator lets enter the freeway which corresponds in the present case to the weak boundary condition implemented in the form of a ghost cell. The transition probability $P : S \times A \times S \to \mathbb{R}_+$ is determined by the freeway traffic simulator we use i.e. the Godunov discretization scheme and the stochastic queue arrival model devised, discussed below. Random events such as perturbations to the input flow of vehicles or accidents affect the otherwise deterministic dynamics of the discretized system. The transition probability P is affected by these events and there likelihood but does not need to be known analytically for the system to operate nor be estimated. This is one of the key advantages offered by Reinforcement Learning over other approaches is classically degenerate (dirac function) in the classical deterministic numerical schemes such as Godunov [13, 194] however this is practically a very unrealistic assumption as the systems under consideration are very often randomly perturbed. The real valued reward function R is for the practitioner to define which implies that the same training algorithm can be used to achieve different objectives. The initial state distribution is denoted P_0, the discount factor γ and the horizon T. If the state is only partially observed, the resulting Partially Observed Markov Decision Process (POMDP) is defined with two additional components. We denote the set of observations Ω and the observation posterior probability distribution on the states $O : S \times \Omega \to \mathbb{R}_+$. Generically, RL consists in finding the policy $\pi_\theta : S \times A \to \mathbb{R}_+$ that maximizes the expected discounted reward.

The Reinforcement Learning problem

To train our control policy, we formalize the situation as a Reinforcement Learning problem. This approach gives a training procedure despite having no optimal control available. It is also flexible as it can be used in many different situations and with different objectives by changing the reward function only. Formally, a Reinforcement Learning problem is defined with:

- A set of states S and a set of possible actions A
- A (stochastic) transition rule

$$(s_t, a_t) \to (s_{t+1}, r_t)$$

  - For any state $s_t \in S$

  - For any possible action $a_t \in A$
  - Determines a new environment state $s_{t+1} \in S$
  - And awards an immediate reward $r_t \in \mathbb{R}$

- A (stochastic) observation rule describing what the agent observes to take its decision

A policy is a rule to determine which action should be taken for a given observation. A policy may be stochastic or deterministic. The goal of Reinforcement Learning algorithms is to find a policy which maximizes a function of the collected reward over time. In practice we want to maximize the expected discounted reward over time, when following the learned policy π.

$$\mathbb{E}\left(\eta(\pi)\right) \text{ where } \eta(\pi) := \sum_{t=0}^{T_{max}} \gamma^t r_t$$

$\eta\left(\pi_\theta\right) = E\left[\sum_{t=0}^{T} \gamma^t R(s_t, a_t)\right]$. We denote $\tau = (s_0, a_0, \ldots)$ the representation of a trajectory defined by the probability distributions $s_0 \sim P_0$, $a_t \sim \pi(a_t|s_t)$, the state transition probability $s_{t+1} \sim P\left(s_{t+1}|s_t, a_t\right)$ and the reward distribution $r_t \sim P\left(r_t|s_t, a_t\right)$.

We will consider a stochastic policy which defines a probability distribution of a_t conditional to s_t (or the observation of the state at time t) parametrized by θ. This creates a stochastic regularization of the objective to maximize and enables to computation of the gradient of a policy with respect to its parameters in spite of the dynamics of the system under consideration not being differentiable, continuous, or even known.

Reinforcement Learning based control of discretized PDEs

The recent developments in RL featured in [134, 205, 206] guarantee convergence properties similar to those of standard control methods and therefore strongly motivates their usage for the control of PDEs. On the contrary, since they are being model-independent, they are intrinsically robust to varying parameters and are able to track parameter slippage. This leads us to consider them as the new generation of generic control schemes. We show how the use of RL on discretized PDEs enables the extension of schemes to systems featuring random dynamics, unknown parameters and regime changes, hence surpassing parametric control schemes.

Controlling PDEs as a Reinforcement learning problem

The MPC approach in [13] and the adjoint method based technique of [194] both rely on the definition of a cost function which needs to be minimized. For PDEs such as the LWR PDE, one typically maximizes throughput, minimizes delay, or a functional of the state (for example encompassing energy emissions). A RL approach will therefore focus on maximizing a decreasing function of that cost which will be our accumulated reward. This is standard practice to encode an operational objective.

Stochastic dynamics and policies. Standard control techniques developed in [13] and [194] are dedicated to deterministic dynamics. This assumption is somewhat unrealistic as cyberphysical systems do not feature fully deterministic dynamics. For instance, traffic is naturally subject to random boundary conditions are the arrival of new vehicles in the region of interest is a random process. Using stochastic policies for such system seems less intuitive but has strong implications in terms of enforcing the robustness of the control scheme in the presence of intelligent agents.

State and action space.

Consider a discretized approximation of the solution y to Eq. (5.1) (see appendix) by the Godunov scheme described in appendix. The solution domain is $[0,\ T] \times [0,\ L]$, the discretization resolutions Δt and Δx satisfying $\Delta t \leq c\Delta x$ are chosen to meet well posedness requirements (Courant-–Friedrichs—Lewy condition where c is the maximum characteristic speed of Eq. (5.1)). The solution to the equation is approximated by a piecewise constant solution computed at the discrete time-space points $\{0, \Delta t, \dots, T - \Delta t, T\} \times \{0, \Delta x, \dots, L - \Delta x, L\}$. The action space for this system consists of incoming flow at the discretized elements $\{0, \Delta t, \dots, T - \Delta t, T\} \times \{0\}$, and generally belongs to a bounded domain $[0 \dots u^{\max_{i,j}}]$. The policy will control this vector of incoming flows at each time step.

Neural Networks for Parametrized stochastic policies.

In this paragraph we show how to construct an actuator based on a Neural Network.

Parametrized stochastic policies.

A vast family of stochastic policies are available for us to use so as to choose an action conditionally to an observation of the state. A common paradigm is to create a regression operator, typically a Neural Network, which is going to determine the values of the parameters of a probability distribution over the actions based on the space observation. We practically use a Multilayer Perceptron that determines the mean and covariance of a Gaussian distribution over the action space. The action the policy undertakes is sampled from this parametrized distribution and will manage to maximize its expected rewards provided a reliable training algorithm is used.

Neural Networks

The policy we train are implemented as Artificial Neural Networks, containing Artificial Neural wired together.

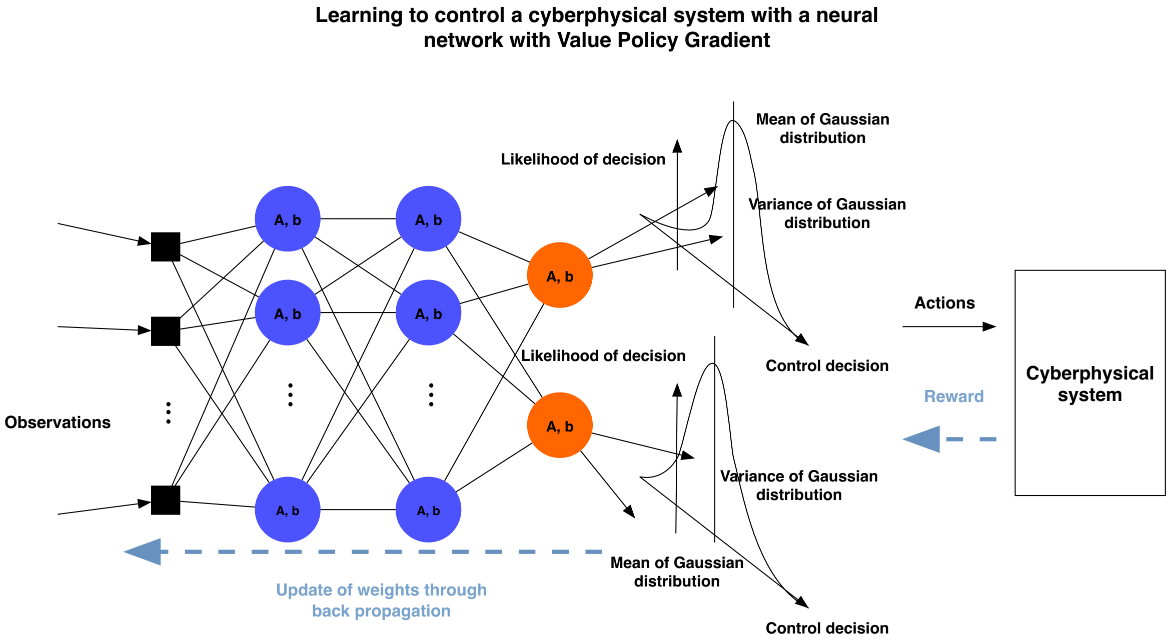

Learning to control a cyberphysical system with a neural network with Value Policy Gradient
Observations
A, b
Likelihood of decision
Mean of Gaussian distribution
Variance of Gaussian distribution
Control decision
Actions
Reward
Cyberphysical system
Update of weights through back propagation

Figure 5.2: Non parametric control learned by experience. A neural network decides the parameter of probability distributions the actions will be sampled from based on observations of the state.

Artificial Neural Model

For $p \in \mathbb{N}$, an Artificial Neural computes an output $y \in \mathbb{R}$ from an input vector $X \in \mathbb{R}^p$ with the following formula:

$$y = \phi(WX + b)$$

Where $\phi : \mathbb{R} \to \mathbb{R}$ is called the activation function which responds to the outcome of an affine transformation of its input space parametrized by $W \in \mathbb{R}^p$ and $b \in \mathbb{R}$.

Networks

A group of q artificial neurons can be linked to a single input $X \in \mathbb{R}^p$ to create an output vector $Y \in \mathbb{R}^q$. This forms a neural layer. When several layers are stacked, with the output of one being the input of the next one, one can speak of an Artificial Neural Network, whose input is the input of the first layer, and output is the output of the last layer.

The general organization and architecture of such a network may vary depending on usages, the type of input data to process. In the setting of computer vision, convolutional neural networks famously achieved human level image classification thanks to the invariance by translation and rotation of the convolution masks they progressively learnt [123].

Back propagation In order to train Neural Networks back-propagation [127] is a key algorithm that performs a stochastic gradient descent on a non-convex function [20]. Approaches to train such a Neural Network for control in the Q-learning framework has been adopted in [160] and were successful in a discrete control setting. With continuous control, a different family of methods is generally used that encourage the policy entailed in the network parameters to take actions that are on average advantageous and discourage actions that have an expected negative reward.

Training algorithms in a RL context

Modern training algorithms for continuous control stochastic policies can be divided in policy gradient-based approaches and non-gradient-based approaches. The former family encompasses first order methods such as REINFORCE [205] which we will denote Vanilla Policy Gradient (VPG), approximated second order methods based on the use of natural gradient descent [116], local line search methods such as Trust Region Policy Optimization (TRPO) [206], LBGFs inspired methods such as Penalized Policy Optimization (PPO) [57] and gradient free approaches such as the cross-entropy method [219]. The performance of these algorithms have been thoroughly compared in [57] where the natural gradient based method Truncated Natural Policy Gradient (TNPG) and TRPO generally outperformed other approaches. In our numerical experiments we find that when the statistical patterns at the stochastic boundary conditions are stationary enough, all approaches perform conveniently. However, TNPG and TRPO outperform other methods when regime changes occur.

REINFORCE

The Reinforce algorithm [205] has been used to train our policy to maximize $\mathbb{E}\left(\eta(\pi)\right)$.

We consider a parametric policy and we denote by θ its parameters. In our case, the policy is a neural network parametrized by its weights. The input layer is filled with the environment observation, and the output layer contains the action probability distribution.

For $a \in A$, and $s \in S$, we note $\pi_\theta(a|s)$ the probability to take action a while being under state s, and following the policy parametrized by θ.

In practice, we use the following equality to compute a gradient average across multiple trajectories:

$$\partial_\theta \mathbb{E}\left(\eta(\pi_\theta)\right) = \mathbb{E}\left(\left(\sum_{t=0}^{T_{max}} \partial_\theta \log\left(\pi_\theta\left(a_t|s_t\right)\right)\right)\sum_{t=0}^{T_{max}} \gamma^t r_t\right)$$

The right side term can be approximated by simply running enough simulations with the given policy according to the law of large numbers.

Once we obtain the gradient $\partial_\theta \mathbb{E}\left(\eta(\pi_\theta)\right)$ we can perform a gradient ascent on the parameters to incrementally improve our policy.

Architecture and choices

Beside these theoretical considerations and algorithms, the network architecture choice has a crucial impact on the training of the policy. If the layers are not adapted to the input data, the training algorithm will converge to a bad local minimum, or may not converge at all.

In our settings, the observation consists of a $n \times 3$ array, where n is the number of discretization cells of the highway in the simulator, and every cell contains 3 information:

- The vehicle density scaled to have a median value of 0, and a standard deviation of 1 in average
- A Boolean value indicating the presence of an off-ramp
- A logarithmic scaled value indicating the number of vehicle waiting in the on-ramp queue if there is any, 0 otherwise.

As this data is spatially structured, we chose to process it with convolutionnal neural layers the core idea being to handle data in a spatially invariant way. Local features are created as a function of these local values independently of the highway location. A pipeline of 3 convolutionnal neural network layers are stacked to create local features. A last layer on the top of these convolution takes the action which consists in deciding how many vehicle can enter at the highway respective on-ramp. This practically achieve by ramp metering traffic lights. This algorithm is illustrated in Figure 5.3.

Sharing information while allowing specialization among agents: the Mutual Weight Regularization algorithm

The features used for decision making on the low level layers of the network are created with a convolutionnal neural network, which exploits the spatial invariance of the problem.

There are two possible situations for the last layer:

- Parameters sharing for all on-ramp policies. This results into having the exact same policy for all agents. This should not be the case because of local specificity of the highway, such as a reduced number of lanes, or a different speed limit for instance.
- Every on-ramp has its dedicated set of parameters. It allows more flexibility and different control for every on-ramp, but dramatically increases the number of parameters, does not share learning between agents, and finally does not converge to a good policy

The novel approach we introduce and call Mutual Weight Regularization (MWR) is between these 2 extremes. It acknowledges the fact that experiences and feedback should be shared between agents, to mitigate the combinatorial explosion when the number of agents scales, and still allows some agent specific modifications to adapt to local variations.

Let us consider:

- $m \in \mathbb{N}$ the number of features computed per cell.
- $i \in R$ where R is the set of cells linked to a controllable on-ramp.
- $X \in \mathcal{M}_{n,m}(\mathbb{R})$ the output of the convolutionnal layers and we note $X[i]$ the features of the agent i.

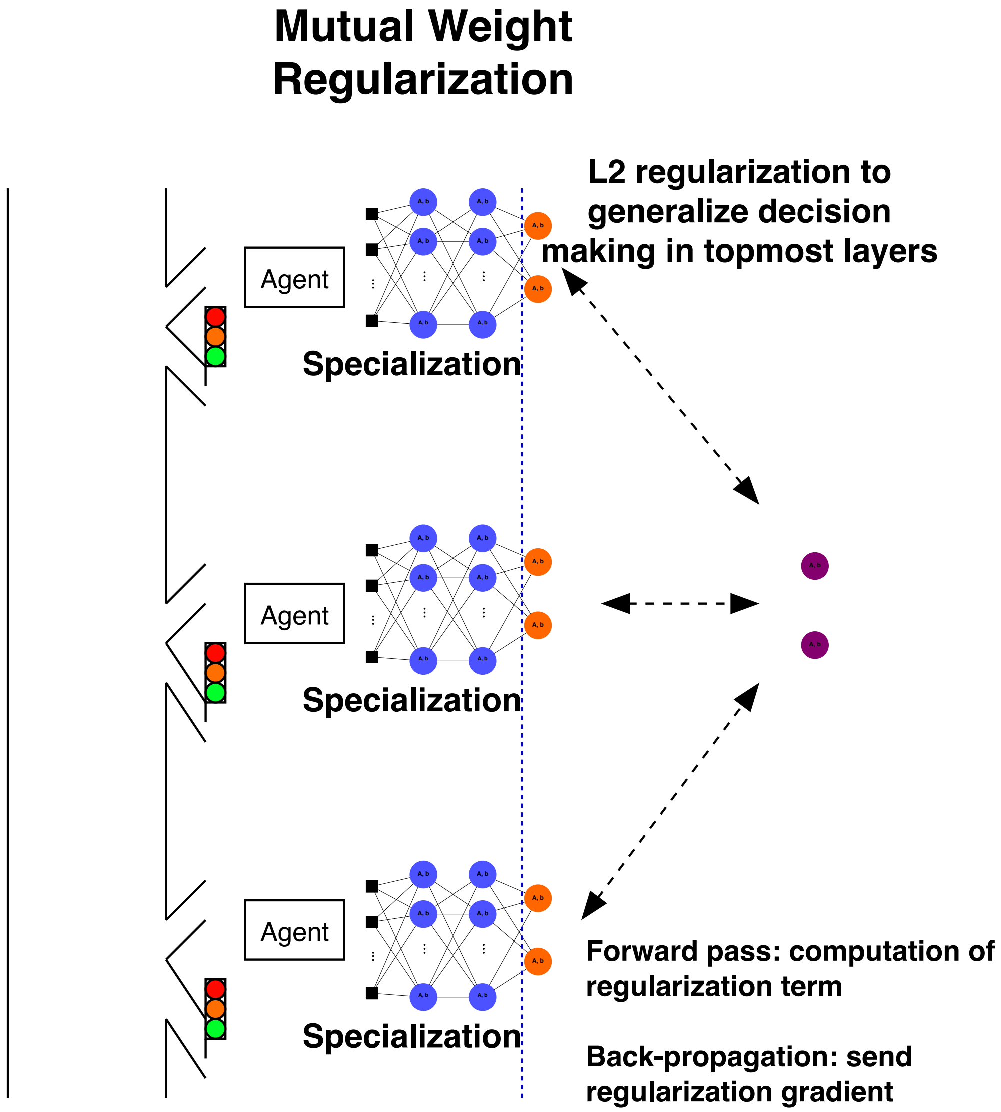

Mutual Weight Regularization
L2 regularization to generalize decision making in topmost layers
Agent
Specialization
Agent
Specialization
Agent
Specialization
Forward pass: computation of regularization term
Back-propagation: send regularization gradient

Figure 5.3: Sharing experience across agents while allowing for specialization. The only messages that are being transferred are parameter values of small size that effectively summarize the learnings of the controllers down to the very elements they should share to learn collaboratively.

We introduce $W_0, W_1, \ldots, W_n \in \mathbb{R}^m$ the distinct parameters of every agent, and define, for $i \in R$, y_i as

$$y_i = \langle W_i \, , X[i] \rangle$$

The decision for the average of the distribution of actions of a given agent will be determined by a non-linear transforms of y_i. Similarly for the variance of this Gaussian stochastic policy distribution [240].

The MWR methods consist in adding a regularization term to the global gradient used for the gradient ascent:

$$\partial_\theta \mathbb{E}\left(\eta(\pi_\theta) - \frac{\alpha}{2}\sum_{j=1}^{n}(W_j - W_0)^2\right) = \tag{5.5}$$

$$\partial_\theta \mathbb{E}\left(\eta(\pi_\theta)\right) - \alpha\sum_{j=1}^{n}(W_j - W_0)\,\partial_\theta \mathbb{E}\left(W_j - W_0\right) \tag{5.6}$$

Where the hyperparameter α defines the strength of the regularization and therefore how much mutual information is shared between agents in the gradient descent. Note that:

- $\alpha = 0$ is equivalent to having independant policies for every on-ramp.
- $\alpha = \infty$ is equivalent to having a shared policy making algorithm for every on-ramp (shares weights).
- W_0 is not actually used for control computation, but is rather a reference weight.

5.4 Experimental results

Proof of concept on different scenario cases

In this first experimentation set, we demonstrate that RL can be used to control PDEs in a robust and generic way. The same training procedure converges to a successful policy for two very different tasks. In this section, simulations are run using a simple Godunov discretization scheme for the LWR PDE (Eq. (5.1)), see appendix.

Highway outflow control

Traffic management scenario. We consider a 5 mile section of I80W starting from the metering toll plaza and ending within San Francisco (see Figure 5.5. The flow is metered at the toll plaza at a rate shown in Figure 5.5 (iii) (i.e. a vehicle rate). The Godunov scheme is implemented with 20 cells and simulated over a time span enabling several bridge crossings. The inflow integrates random arrival rates for inbound traffic, which consists of a sinusoid perturbed by random noise. The state is vehicular density (i.e. $\rho(t, x)$ as defined by the LWR PDE, vehicles per unit length) from which flow $F(\rho(t, x))$ (numbers of vehicles per unit time at x) can be computed.

Action space and environment We consider the following operational scenario. The number of vehicles upstream from the meter (see Figure 5.5 (iii)) is randomized for each simulation in order to reproduce the diversity of freeway flow dynamics as they actually occur. At every time step, the observation forward propagated in the policy Neural Network is the value of current time step in time unit. An action undertaken is also a positive scalar representing the number of cars permitted to enter the highway per time unit. We train the policy to reproduce a pattern $z(t)$ on the outflow with the following reward

$$R\left(s_t, a_t\right) = -\left(\rho(t, L) - z(t)\right)^2 .$$

The choice of this reward function encodes our intention to replicate the function $z(t)$ on the downstream boundary condition (located at spatial offset L). In particular, what is remarkable here is that this is the only way the environment provides information to the policy about the state of the freeway. Indeed, we only provide the current time step t as state observation to our policy along with the reward associated with the result obtained after a simulation roll out.

Learning in the presence of disruptive perturbations The discretized model we use is well-suited for representing traffic accidents. The state update mechanism may be randomly altered by accidents which drastically change the maximum flow the freeway can carry at local points in space. Accidents can be simulated by locally decreasing the maximum flow speed in a given discretization cell for a given time interval. A key goal of our work is alleviating the impact of accidents while simultaneously tracking operational objectives (e.g. desired outflow of the bridge into the city), and achieving it with a robust and generic method is a tremendous breakthrough for urban planning. See the appendix for a presentation of the accident scenario.

Learning algorithms and convergence to an effective policy In Figure 5.5, we analyze the results of the control scheme learnt by a given policy consisting of two fully connected hidden layers of size 32. The policy controls the inflow of cars (boundary control). We choose an arrival rate sufficiently high on average to provide the controller with sufficient numbers of vehicles to match the prescribed downstream conditions (note obviously in congestion this is always the case). The results prove that in spite of the problem being non-differentiable, non-smooth, non-linear and perturbed drastically by unpredictable accidents and random input queues, the policy converges to a control scheme that manages to replicate the objective density. The learning phase uses different policy update methods such as [116, 135, 206] and benchmarked in [57]. In this benchmark, among gradient based methods, TNPG, TRPO and PPO seem to outperform the simpler REINFORCE method which only leverages first order gradient information. In Figure 5.6 we show how PPO, TRPO and REINFORCE are all reliable in this instance and converge to more effective policies than TNPG. It is also noteworthy that PPO converges faster to a plateau of rewards.

Inner domain control

Reward shaping, in the form of assigning a target density and penalizing the L_2 distance between the observed density and the objective enables us to reproduce an arbitrary image with the solution density in the solution domain only by controlling it on the boundaries. The results in Figure 5.7 demonstrate the ability of the method we present to train a policy to extensively control the values of a solution to the discretized PDE we study in its solution domain. The action space here is much higher dimensional as ramps are present all along the freeway that can let cars in at a sequence spatial offsets each separated by 3 cells. At the same time, in our simulation, half of the flow leaves the freeway so that the density of vehicles on the road can also decrease. An off ramp split model handles the vehicle leaving the freeways. In spite of the increased dimensionality, a neural net with tree fully connected hidden layers of size 64 trained by the TRPO method converges to a policy

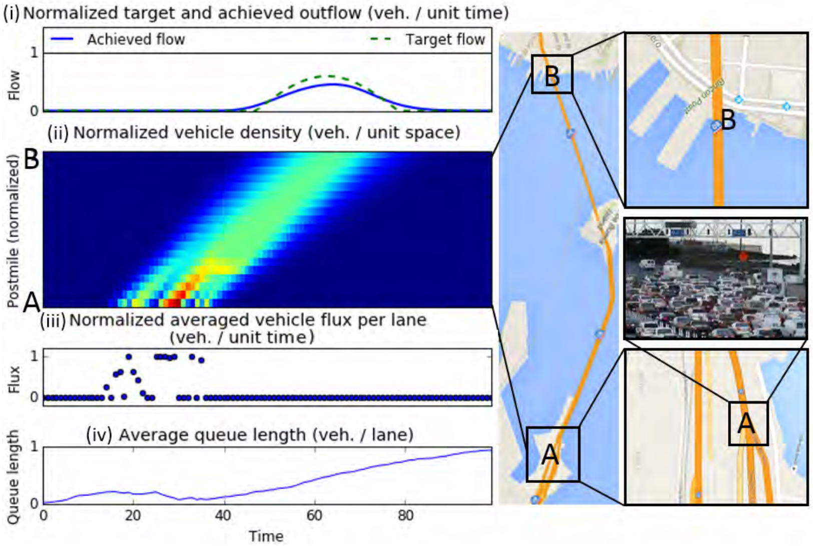


Figure 5.4: I80 Bay Bridge metering control. The control (iii) generated by neural RL leads to a solution (ii) of the PDE, which achieves an outflow shown in (i), close to the desired profile in (i). In the process, the queue, shown in (iv) grows as vehicles into the toll plaza continue to arrive at a higher rate than allowed by the control scheme. The time space diagram i.e. (ii) is a common tool used in transportation engineering to show the solution of the PDE in space (x) and time (y). The tracking policy learned through neural RL.

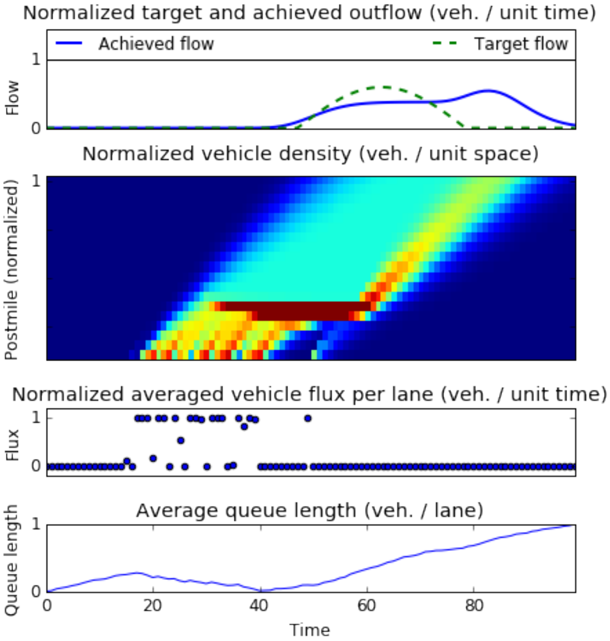


Figure 5.5: During the training period, major variations in the traffic model are generated to illustrate the robustness of the trained policy. Despite a highly perturbed training phase, the learnt control policy manages to converge to the prescribed objective of outbound downstream flow.

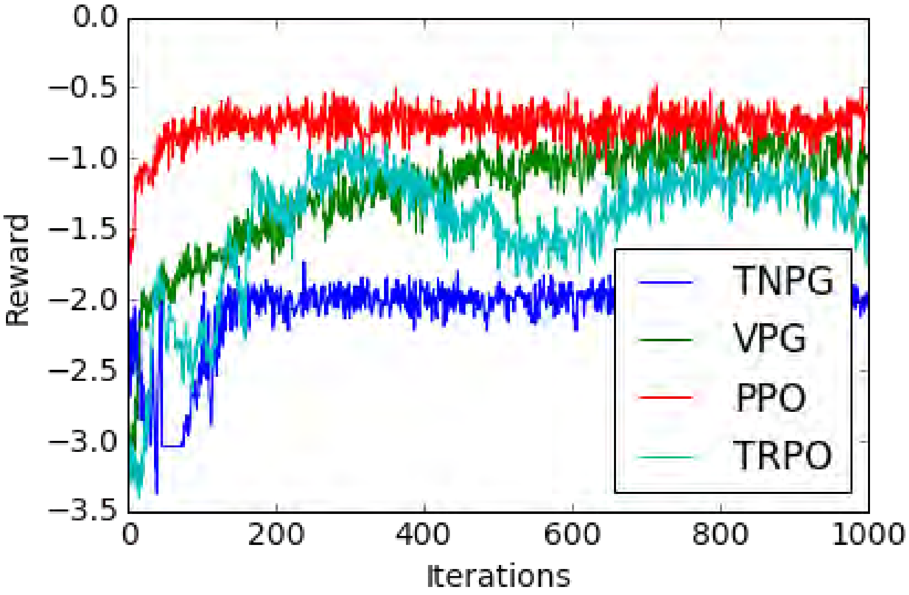


Figure 5.6: Comparison of different policy training algorithms for the control of the downstream boundary condition by the upstream boundary condition in the presence of accidents. We notice that TNPG performs poorly, VPG and TRPO offer mediocre performance and PPO outperforms all the methods.

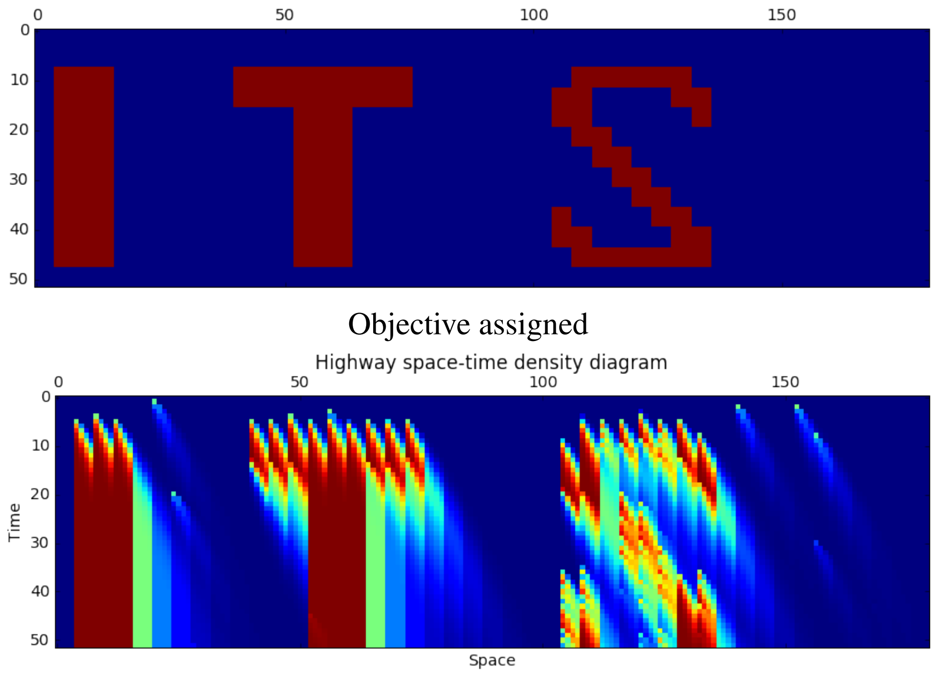
capable of reproducing a target solution in the interior domain as shown in Figure 5.7, whereas TNPG failed in this instance to converge to an efficient policy. From a practitioner's perspective, this example is very powerful, it shows the ability to generate arbitrary congestion patterns based on metering along the freeway. From a PDE control standpoint, this is even more powerful, as direct state actuation is a very hard problem in manufacturing and has a lot of applications with PDEs such as the (nonlinear) heat equation, Hamilton-Jacobi equations, and several others.

Optimal ramp metering control

In order to demonstrate the applicability of this novel method to real world cases, we consider the Ramp-Metering problem in a 20 miles (33 km) long section of the 210 Eastbound freeway in southern California as illustrated in Figure 5.8. For this simulation, we use the BeATS simulator calibrated by traffic experts based on real-world data. Every simulation run lasts for 4 hours, after a 30 minutes warmup period to initialize the freeway. The simulation starts at 12pm and the traffic peak happens between 3pm and 4pm, as the demand curve reaches a maximum (Fig. 5.14).

Two reinforcement learning algorithms and the ALINEA control scheme are benchmarked against the baseline scenario in which no control occurs at all:

- **NoRM, baseline**: The baseline without any ramp metering. Cars instantly enter the freeway when they reach an onramp, and if the freeway has enough capacity



Objective assigned

Objective achieved after 2000 iterations

Figure 5.7: Control of multiple entry points on the freeway enables the training of a policy capable of replicated a prescribed density.

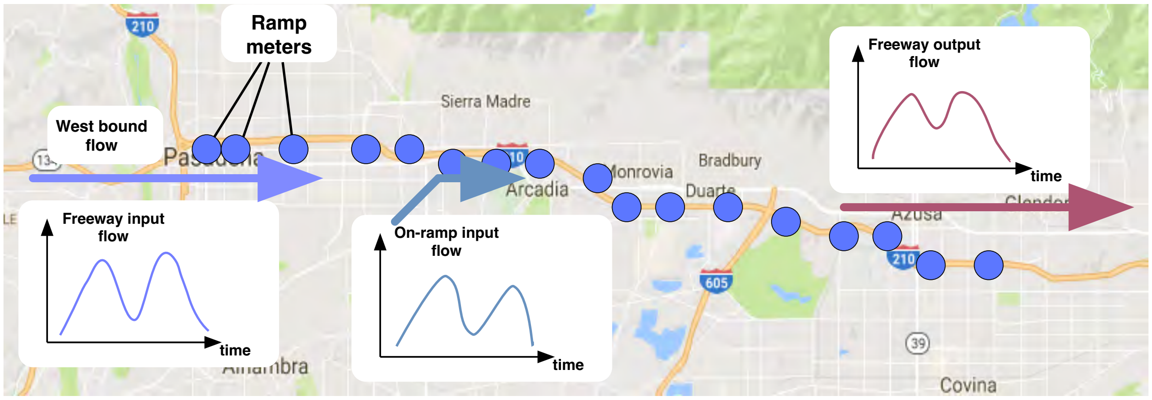


Figure 5.8: Setting simulated by BeATS in our experiment

- **NoMWR, standard deep reinforcement learning**: The Reinforcement Learning based policy we introduce, trained with shared weights for the last layer
- **MWR, novel approach to training**: The same policy as NoMWR, but trained with MWR.
- **ALINEA, parametric control**: The state of the art reference algorithm, using with model and parameters the exact same values used in the simulator it is being benchmarked on.

Reinforcement Learning problem

In this scenario, the agent takes an action every 32 seconds. An action is a $\mathbb{R}^{29}$ vector with the ramp-metering rates for the 29 on-ramps on the highway section, in vehicle per time unit. The reward we collect at every time step is the total outflow in the last 32 seconds (in number of vehicle per time unit).

The highway is discretized in 167 cells of 200 meters each to generate an observation vector in $\mathbb{R}^{167\times 3}$. For every cell, the following 3 data values are provided to the network:

- Density in vehicle per space unit, normalized
- A boolean value indicating the presence of an off-ramp on this cell
- The number of cars waiting in this cell's onramp queue (logarithmically scaled), or 0 if there is no onramp

It is worth mentioning that the Reinforcement Learning policies are trained in a stochastic way to ensure that a generic policy is learned. This is done by introducing some noise in the actions taken by the Neural Network: the ramp metering actually applied is sampled from a Gaussian distribution centered on the network output. This strategy, along with the use of shared learning techniques over the 29 onramps, globally prevents overfitting issues.

Numerical results

After the training, we compared the results of our approach to existing algorithm on several criteria. The average speed is globally increased as expected (Fig. 5.9). We also report the Total Vehicle Miles in Fig. 5.13, along with the Total Vehicle Hour (Fig. 5.12) that assess the performance of our approach with a single score. In both cases, the MWR training approach provides a significative performance increase regular parameters sharing (NoMWR), and almost reaches the performance of the reference and state of the art parametric method ALINEA.

5.5 Conclusion of the chapter

We have demonstrated how neural RL substantially improves upon the state-of-the-art in the field of control of discretized PDE. It enables reliable non-parametric control while offering theoretical guarantees similar to that of classic parametric control techniques. it offers a novel non-parametric

Approach	Score in veh.hr (lower is better)	Score in veh.mile (higher is better)
ALINEA	10514	644522
MWR	10575	644334
NoMWR	10617	643605
NoRM	11085	639709

Table 5.1: Aggregated scores of the different control strategies over the congested period. Reinforcement learning enables a non parametric control scheme whose performance is similar to that of the parametric ALINEA scheme and MWR, the new learning algorithm we introduce, for improves this performance.

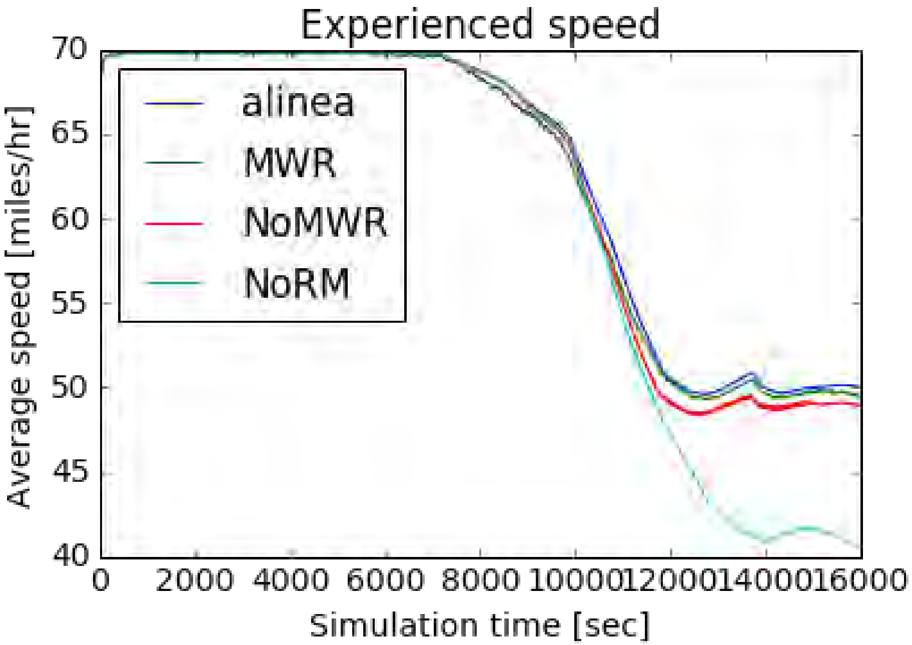


Figure 5.9: ALINEA maintains speed above the optimal level defined during calibration by experts. Reinforcement Learning methods and the MWR especially, manage to implicitly deduce this optimal speed during training.

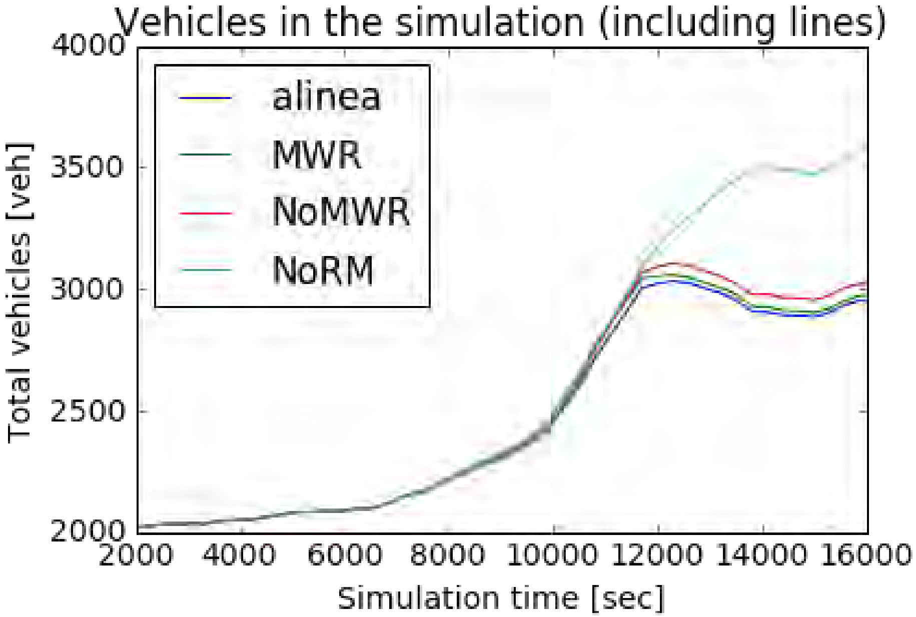


Figure 5.10: Ramp Metering methods reduce by 20% the number of vehicles in the freeway during congestion peak times. Our RL learning trained with MWR maintains the number of vehicles in the simulation to a level similar to what ALINEA performs.

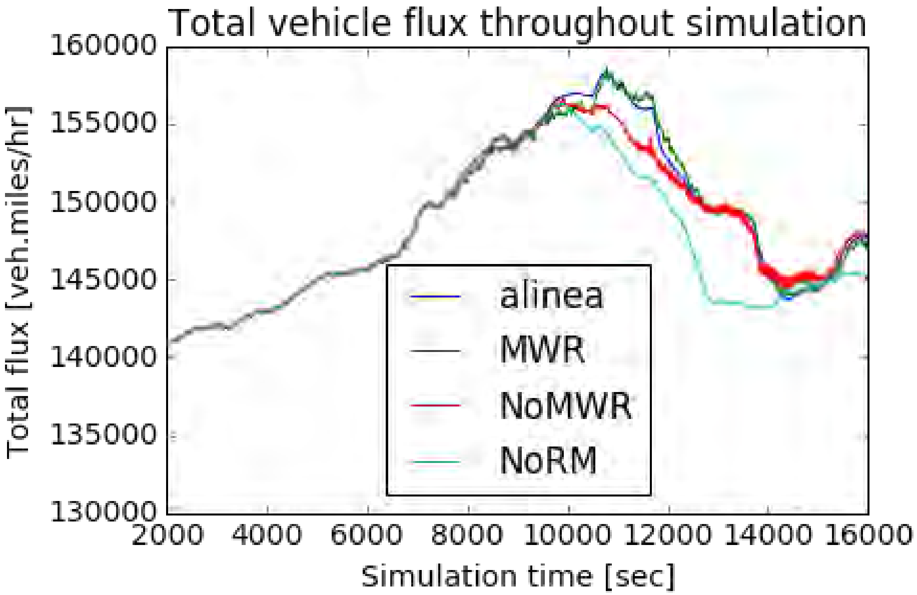


Figure 5.11: Vehicle flux

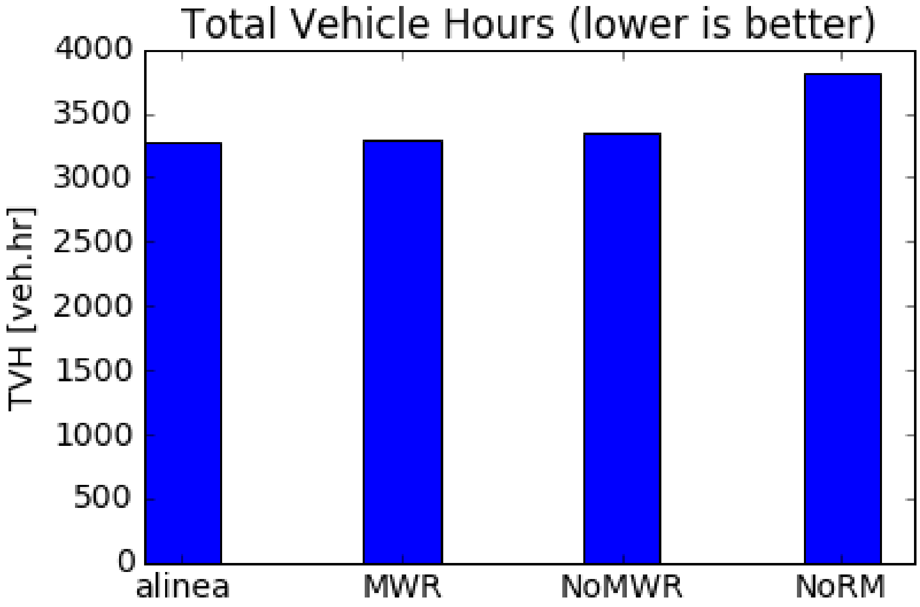


Figure 5.12: Total Vehicle Hour during congestion time

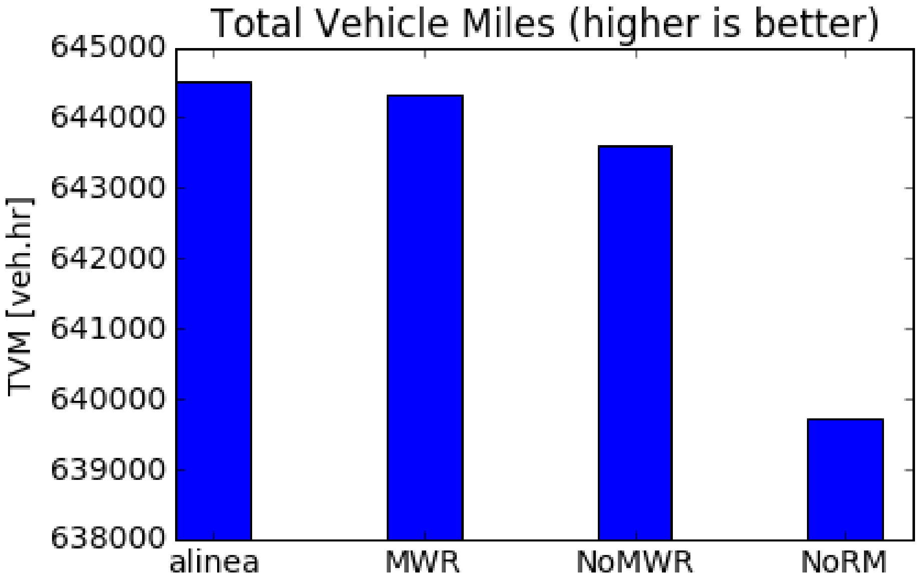


Figure 5.13: MWR provides a significant performance improvement and almost reaches ALINEA performance. Compared to no ramp-metering, MWR reproduces 96% of ALINEA performance improvement, while the regular RL training only reaches 80% of ALINEA performance.

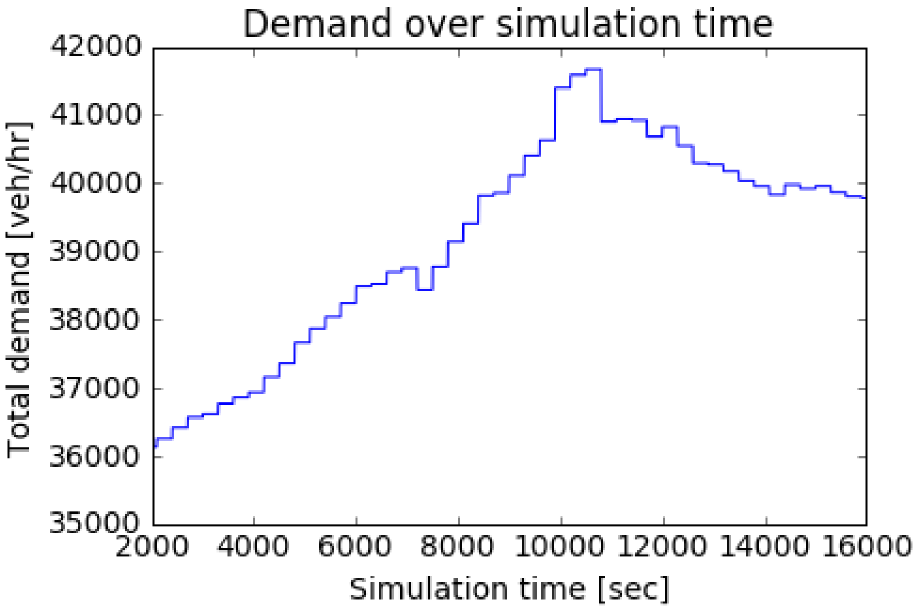


Figure 5.14: Total demand curve. Congestion occurs after a few hours of simulation.

method to control these models. In particular, neural RL can be applied without an explicit model of system dynamics, and instead only requires the ability to simulate the system under consideration. In addition, neural RL, unlike traditional PDE boundary control techniques, is robust to regime change and nonlinear dynamics enabling the control of complex and highly variable real-world systems. Through our experimental evaluation, we demonstrated that neural RL approach can be used to control discretized macroscopic vehicular traffic equations by their boundary conditions in spite of accidents drastically perturbing the system. Achieving such robustness is a significant breakthrough in the field of control of cyberphysical systems. Specific to the practice of transportation, the results are a major disruption as they enable us to beat current controllers by performing adaptive control, without the need for model calibration. By eliminating the need for calibration, our method addresses one of the critical challenges and dominant causes of controller failure making our approach particularly promising in the field of traffic management. We also introduced a novel algorithm, MWR, to achieve multi-agent control and leverage trial and error experiences across different agents while at the same time allowing each agent to learn how to tailor its behavior to its localization in the large cyberphysical system under study.

This last chapter on model-free free control policies reveals multiple elements of patterns that unify heterogenous distributed representations for control. First, there is no systematic need for a preliminary system identification in an analytical form. However, as the learn process takes a lot of trial, there is need for pre-training on a simulator and therefore a certain degree of understanding of the dynamics of the system under consideration is required. Second, in the absence of Slater's conditions, we know that solving the dual of a non-convex problem is always at most as good as solving the primal problem of interest. Such are the theoretical concessions that needed to be made for instance in [194]. Such are two major advantages of the approach we propose to control a complex non-linear system in the last chapter.

We have demonstrated that the need to use a dual of the problem is not necessary to create a meaningful, data efficient and homogenous representation for a heterogeneous distributed control problem.

While training a model with policy gradient methods, the parameters of the stochastic policy the model learns to optimize progressively summarize a large amount of experience acquired along many batches of attempts to control the system. Such are the messages passed from an agent to another to enable better performance. What we lost in theoretical guarantees we gained in generality and adaptiveness. Also there is no trade-off in terms of the scalability of the scheme we designed. The communication avoiding sharing of parameter updates leverages the summaries that are effectively synthesized in the form of model parameters.

Conclusion

The alternate representations we presented have provably enabled scalability and statistical efficiency for time series analysis and control in the four use cases we considered. Our search for small yet expressive summaries for heterogenous problems – which was partly motivated by mitigating communications whose cost dominate compute time in the distributed setting – has successfully led to the design of parsimonious and robust unifying representers of collections of heterogenous data sets and control problems.

Summary of contributions

The second chapter of the thesis stated key statistical results that give practitioners the confidence that kernel smoothed estimation of continuous time randomly observed time series is reliable in the setting of event-driven second-order stationary dynamics. We also showed how a Fourier transform based representer provides communication avoiding algorithms to compute the kernel smoothed cross-correlation estimator we proposed. We therefore have designed a communication avoiding computation pattern in the distributed setting for cross-correlation estimation.

The statistical guarantees we gave leveraged an analysis of Long Range Dependent second-order estimation with observation timestamps that are not generated independently from the process of interest. The most important intuition was to move back and forth between time domain and frequency domain. In the first chapter, it was easier to assess the properties of kernel smoothed second-order estimators in the continuous time domain while the Fourier domain provided tremendous computational advantages.

The third chapter of the thesis stepped away from computational issues to focus on the analysis of an infinite dimensional continuous time system corresponding to a system of Partial Differential Equations: the Aw-Rascle-Zhang model. In such a setting, we employed distributed Laplace transforms to provide an expressive yet simple representation of the linearized Partial Differential Equations under study.

The use of Laplace transform was motivated by the design of a study which are a representation of complex non-linear system such as the Aw-Rascle-Zhang model that can easily enable control. A byproduct of the approach was the discovery of an equivalent of the Froude number for macroscopic freeway traffic dynamics that delineates conditions under which fast expanding shockwaves appear in the system. Such shocks are concerning as they can lead to brutal decelerations and crashes. Our numerical study leveraged the spectral decomposition in order to show that in spite of the linear approximation, the predictions given by the simplified model were still reliable.

The fourth chapter of the thesis focused on a discrete collection of small control problems subject to different and independent constraint sets. In its most general form, the approach we developed can be used in any collaborative control scheme where a swarm of controllers with independent constraints are bound by a common objective. Assumptions of strong convexity led us to consider a convex dual of the primal problem we considered.

Under such assumptions, Slater's conditions were met and we could rely on the equivalence between the primal and dual problems to solve a dual-split representation of the problem of interest. The dual split approach provided many advantages. We were able to break the binding objective into smaller sub-problems and therefore distribute computations across controllers. Only small messages needed to be communicated and constraint sets could remain private. Furthermore, we proved that the representing messages could be obfuscated by additive noise. A comparative study of the cohort's convergence speed and the amount of individual information leakage showed that our representer is data efficient. The information broadcasted on the communication method enables convergence to an optimal Lagrangian multiplier faster than it reveals the individual level optimal control schedules.

The fifth chapter made use of Reinforcement Learning to optimally control discretized systems of PDEs accounting for traffic dynamics. A Godunov discretization scheme was employed to build a simulator of the Lighthill-Whitham-Richards model (a simpler model than Aw-Rascle-Zhang although still realistic) which enabled the training of a control policy to optimize traffic flow. Learning occurred by trial and error which made the procedure adaptive and it was able to learn how to deal with perturbations such as crashes.

Such properties clearly differ from pre-existing techniques such as ALINEA or adjoint-based optimization which make strong assumptions about the dynamics of the stretch of freeway they control. We showed that the adaptiveness of the method does not hinder any ability to extensively control a discretized PDE. We managed to write precise symbols with the density of vehicles. We also optimized throughput with comparable performance to an ALINEA control scheme. A major difference though was that our method learned the dynamics of the system on the fly whereas ALINEA knew the very parameters we used for simulation.

As we considered distributed control schemes involving multiple ramp meters affecting neighboring locations on a freeway, we created a method for the controllers to share their learnings while still being able to adapt to the idiosyncracies of the ramp they are located on. In this model-free setting, we were not able to leverage Fourier, Laplace or convex duality. However, by using the very parameters of the policy being learned as a representation of heterogenous problems, we were able to share common features of the control sub-problems with a unifying parsimonious representation. The neural network weights being learned were indeed a way of summarizing learnings gathered along thousands of simulation runs.

Open problems

For estimation under irregular sampling, a representer consisting of the coordinates of a signal on a small subset of a Fourier basis is unfortunately poorly adapted to the more general setting of non-stationary time series. The focus of our current research is to employ Wavelet based representations [152] in order to more robustly and accurately capture changes of regimes. We find that, as fast computations of dyadic wavelet decompositions and orthogonal wavelet transforms rely on cascading convolutions [152, 151, 180], interesting interactions arise with the fast developing field of Deep Learning [84]. It is noteworthy that the typical access patterns to the data are then very different as Deep Neural Networks are generally non-convex and therefore training with a method of the family of Stochastic Gradient Descent [170] is recommended. The noisy stochastic descent presents the advantage of being able to leave low quality local minima.

Our analysis of the ARZ coupled PDEs has a direct extension to enable control. The distributed Laplace representation we obtained can be leveraged to control the system. Such is the approach that was employed in a similar system of PDEs – the Saint Venant Equations – in [143]. The study led to a ground breaking control scheme for shallow water canals enabling H_∞ control among many developments. Our aim is now to leverage a similar analysis in the setting of macroscopic traffic dynamics in order to provide H_∞ control of infinite dimensional models that accurately account for the dynamics affecting the commute of millions each day. A clear motivation is indeed to better control traffic flows in order to mitigate congestion. The ultimate goal is to shorten travel times as well as decreasing harmful gas emissions. Less congestion on freeways leads to higher economic productivity and lower environmental impact in the ever growing dense urban areas of the 21th century [204].

In the fourth chapter of the thesis, we demonstrated that the Lagrangian multiplier we obtain in the dual formulation of the problem can easily be interpreted as a price multiplier in two key applications: Mobility-as-a-Service and electric load flattening. Such price can be used as a differentiator of locations in time and/or space where there is more imbalance between the offer controllers can achieve and the demand that needs to be matched.

Although the multiplier was obtained as the byproduct of our dual splitting formulation, we now plan to consider other ways of having a cohort of agents converge to a price: by negotiation. The distributed computation pattern of the algorithm we developed alternates between distributed communication of an updated price between agents and individual level adjustment of the control scheme based on the current price. Such a scheme can be used outside the framework of strong duality. Deep Reinforcement Learning [135] can indeed be employed to let the cohort of agents learn how to negotiate a price that leads to optimal collaboration without communicating individual level constraints.

Such techniques could also be used to extend the findings of the final chapter. The singularity of the chapter concerned with distributed Deep Reinforcement Learning is that perception and action are learned as part of the same neural network. In other words, we implicitly use the same single model for system identification and control. Using black box adaptive models presents many advantages such as expressiveness, simplicity of deployment and effective mitigation of model staleness by online adaptation. However, when such a strategy is applied to a cyber-physical system whose sensors are easily accessible to all and in particular malicious entities, it is primordial to assess its level of exposure to adversarial inputs. Using adversarial networks [85] to give more robustness to the control scheme is the subject of current research. The idea correspond to training an attacker of the cyber-physical system in order to force the learned control policy to immunize itself against adversarial inputs [91]. In particular, such techniques would warrant a more confident use of neural controllers in city centers. Micro-simulators that consider dynamics at the level of individual vehicles could then be employed at the cost of longer simulations in order to realistically train controllers of distributed groups of traffic lights in arterial networks.

General directions for further investigations

Stepping back to a higher level of analysis of the present work, it has been proven recently that forcing fewer parameters to represent large data sets is key enabler of generalization in Deep Learning [224]. Such an observation corroborates our general finding that seeking a parsimonious representer for a large heterogeneous collection of data sets or control problems enables data efficiency as a by-product. One of the aims of future work corresponds to leveraging the information theoretic considerations developed in [224] in order to better characterize the relationship between the size of the representer and its statistical efficiency.

The flexibility of information theory [41] where the data processing inequality or the invariance under homeomorphic re-parametrization are true under very general assumptions is very appealing. It can help us to rely less on modeling assumption and focus more of the very nature of the communication pattern occurring in the distributed setting.

We indeed seek theory that works in more general settings as neural networks provide better modeling and control ability although they generally lead to problems for which general assumptions are hard to find. Such deep models will play a key role in time series analysis and control thanks to their ability to dynamically identify temporal scales at which analysis or control should occur.

The thesis focused on heterogenous problems where sensing was asynchronous across sensing channels, an infinity of different states were considered, a collection of sub-problem were affected by different constraints and finally a cohort of agents located at unique locations in a cyberphysical system needed to learn to collaborate.

However, the characteristic time scale under consideration was always homogenous. Such a feature is rather restrictive and a novel aim consists in being free to detect the best temporal scale at which the data should be examined. The temporal structure of the control scheme being developed should also be adaptive in order to better take into account the characteristics of the system being controlled and the objective of the controller in the long run.

Dilated convolution stacks that were first used to compute fast wavelet transform [152] are now being employed in computer vision to enable multi-scale scenic parsing [245], leading to ground breaking results.

Time series can also be considered as scenes that can be parsed at different characteristic scales. Such a paradigm provides key statistical advantages [180] and enable the capture of non-stationary dynamics [151].

The general direction of the following work will therefore point towards black box multi-scale control policies relying on a multi-scale perception architecture. The interaction between the ability for wavelet transforms to compress data [151] with novel neural architectures relying on orthogonal non-redundant representations [37, 68] may lead to expressive architectures relying on few parameters. Based on the information theoretical analysis of Deep Neural Networks in [224] we mentioned, such models would enjoy expressiveness and good generalization abilities thanks to their parsimonious use of free parameters.

Bibliography

[1] Martın Abadi et al. "Tensorflow: Large-scale machine learning on heterogeneous distributed systems". In: *arXiv preprint arXiv:1603.04467* (2016).

[2] Pieter Abbeel, Adam Coates, and Andrew Y Ng. "Autonomous helicopter aerobatics through apprenticeship learning". In: *The International Journal of Robotics Research* 29.13 (2010), pp. 1608–1639.

[3] Frédéric Abergel et al. *Market microstructure: confronting many viewpoints*. John Wiley & Sons, 2012.

[4] Niels Agatz et al. "Optimization for dynamic ride-sharing: A review". In: *European Journal of Operational Research* 223.2 (2012), pp. 295–303.

[5] Rakesh Agrawal and Ramakrishnan Srikant. "Privacy-preserving data mining". In: *ACM Sigmod Record*. Vol. 29. 2. ACM. 2000, pp. 439–450.

[6] Soyoung Ahn and Michael J. Cassidy. "Freeway Traffic Oscillations and Vehicle Lane-Change Maneuvers". In: *17Th Int. Symp. on Transportation and Traffic Theory*. Elsevier, 2007, pp. 691–710.

[7] Ulrich Matchi Aıvodji et al. "Meeting points in ridesharing: A privacy-preserving approach". In: *Transportation Research Part C: Emerging Technologies* 72 (2016), pp. 239–253.

[8] Aykagan Ak and Alan L Erera. "A paired-vehicle recourse strategy for the vehicle-routing problem with stochastic demands". In: *Transportation Science* 41.2 (2007), pp. 222–237.

[9] Karl J Åström. *Introduction to stochastic control theory*. Courier Corporation, 2012.

[10] A. Aw and M. Rascle. "Resurrection of second order models of traffic flow". In: *SIAM Journal of Applied Mathematics* 60.3 (2000), pp. 916–938.

[11] Rajesh Krishna Balan, Khoa Xuan Nguyen, and Lingxiao Jiang. "Real-time trip information service for a large taxi fleet". In: *Proceedings of the 9th international conference on Mobile systems, applications, and services*. ACM. 2011, pp. 99–112.

[12] Saeid Bashash et al. "Plug-in hybrid electric vehicle charge pattern optimization for energy cost and battery longevity". In: *Journal of power sources* 196.1 (2011), pp. 541–549.

[13] Tom Bellemans, B De Schutter, and Bart De Moor. "Model predictive control for ramp metering of motorway traffic: A case study". In: *Control Engineering Practice* 14.7 (2006), pp. 757–767.

[14] Abdelghani Bellouquid, Elena De Angelis, and L Fermo. “Towards the modeling of vehicular traffic as a complex system: A kinetic theory approach”. In: *Mathematical models and methods in applied sciences* 22.supp01 (2012).

[15] Moshe Ben-Akiva et al. “Network state estimation and prediction for real-time traffic management”. In: *Networks and Spatial Economics* 1.3-4 (2001), pp. 293–318.

[16] Aharon Ben-Tal, Laurent El Ghaoui, and Arkadi Nemirovski. *Robust optimization*. Princeton University Press, 2009.

[17] *Berkely Advanced Traffic Simulator*. `https://github.com/calpath/beats`. Accessed: 2016-10-17.

[18] Dimitri P Bertsekas. “Nonlinear programming”. In: (1999).

[19] Dimitri P Bertsekas and John N Tsitsiklis. *Parallel and distributed computation: numerical methods*. Vol. 23. Prentice hall Englewood Cliffs, NJ, 1989.

[20] Christopher M Bishop. *Neural networks for pattern recognition*. Oxford university press, 1995.

[21] Christopher M Bishop. *Pattern recognition and machine learning*. springer, 2006.

[22] Matt Bishop. *Introduction to Computer Security*. Addison-Wesley Professional, 2004. ISBN: 0321247442.

[23] Sebastien Blandin et al. “A general phase transition model for vehicular traffic”. In: *SIAM journal on Applied Mathematics* 71.1 (2011), pp. 107–127.

[24] Enide Bogers, Michel Bierlaire, and Serge Hoogendoorn. “Modeling learning in route choice”. In: *Transportation Research Record: Journal of the Transportation Research Board* (2015).

[25] George EP Box et al. *Time series analysis: forecasting and control*. John Wiley & Sons, 2015.

[26] Stephen Boyd and Lieven Vandenberghe. *Convex optimization*. Cambridge university press, 2004.

[27] Stephen Boyd et al. “Distributed optimization and statistical learning via the alternating direction method of multipliers”. In: *Foundations and Trends® in Machine Learning* 3.1 (2011), pp. 1–122.

[28] David R Brillinger. *Time series: data analysis and theory*. Vol. 36. Siam, 1981.

[29] Peter J Brockwell and Richard A Davis. *Time series: theory and methods*. Springer Science & Business Media, 2013.

[30] Hua Cai et al. “Siting public electric vehicle charging stations in Beijing using big-data informed travel patterns of the taxi fleet”. In: *Transportation Research Part D: Transport and Environment* 33 (2014), pp. 39–46.

[31] Frank M Callier and Charles A Desoer. *Linear system theory*. Springer Science & Business Media, 2012.

[32] Rich Caruana. “Multitask learning”. In: *Learning to learn*. Springer, 1998, pp. 95–133.

[33] TH Chang, Angelia Nedic, and Anna Scaglione. “Distributed constrained optimization by consensus-based primal-dual perturbation method”. In: (2014).

[34] P. Chathuranga Weeraddana et al. “On the Privacy of Optimization Approaches”. In: *ArXiv e-prints* (). arXiv: `1210.3283`.

[35] Danjue Chen et al. “A behavioral car-following model that captures traffic oscillations”. In: *Transp. Res. B* 46.6 (2012), pp. 744–761.

[36] Shih-Fen Cheng and Thi Duong Nguyen. “Taxisim: A multiagent simulation platform for evaluating taxi fleet operations”. In: *Proceedings of the 2011 IEEE/WIC/ACM International Conferences on Web Intelligence and Intelligent Agent Technology-Volume 02*. IEEE Computer Society. 2011, pp. 14–21.

[37] Yu Cheng et al. “An exploration of parameter redundancy in deep networks with circulant projections”. In: *Proceedings of the IEEE International Conference on Computer Vision*. 2015, pp. 2857–2865.

[38] B. Coifman, S. Krishnamurthy, and X. Wang. “Lane-Change Maneuvers Consuming Freeway Capacity”. English. In: *Traffic and Granular Flow 03*. Springer Berlin Heidelberg, 2005, pp. 3–14. ISBN: 978-3-540-25814-8.

[39] Rinaldo M Colombo. “Hyperbolic phase transitions in traffic flow”. In: *SIAM Journal on Applied Mathematics* 63.2 (2003), pp. 708–721.

[40] George F Coulouris, Jean Dollimore, and Tim Kindberg. *Distributed systems: concepts and design*. pearson education, 2005.

[41] Thomas M Cover and Joy A Thomas. *Elements of information theory*. John Wiley & Sons, 2012.

[42] DR Cox and PAW Lewis. “Multivariate point processes”. In: *Selected Statistical Papers of Sir David Cox: Volume 1, Design of Investigations, Statistical Methods and Applications* 1 (2005), p. 159.

[43] Nello Cristianini and John Shawe-Taylor. *An introduction to support vector machines and other kernel-based learning methods*. Cambridge university press, 2000.

[44] C. F. Daganzo. “Requiem for second-order fluid approximations of traffic flow”. In: *Transp. Res. B* 29.4 (1995), pp. 277–286.

[45] C. F. Daganzo. “The cell transmission model, part II: network traffic”. In: *Transp. Res. B* 29.2 (1995), pp. 79–93.

[46] Carlos F. Daganzo. “The cell transmission model: A dynamic representation of highway traffic consistent with the hydrodynamic theory”. In: *Transp. Res. B* 28.4 (1994), pp. 269–287.

[47] Daryl J Daley and David Vere-Jones. *An introduction to the theory of point processes: volume II: general theory and structure*. Springer Science & Business Media, 2007.

[48] Jeffrey Dean and Sanjay Ghemawat. "MapReduce: simplified data processing on large clusters". In: *Communications of the ACM* 51.1 (2008), pp. 107–113.

[49] Jeffrey Dean et al. "Large scale distributed deep networks". In: *Advances in neural information processing systems*. 2012, pp. 1223–1231.

[50] Pierre Degond and Marcello Delitala. "Modelling and simulation of vehicular traffic jam formation". In: *Kinet. Relat. Models* 1.2 (2008), pp. 279–293.

[51] A.I. Delis, I.K. Nikolos, and M. Papageorgiou. "High-resolution numerical relaxation approximations to second-order macroscopic traffic flow models". In: *Transp. Res. C* 44 (2014), pp. 318–349.

[52] Josep Domingo-Ferrer, Francesc Sebé, and Jordi Castella-Roca. "On the security of noise addition for privacy in statistical databases". In: *International Workshop on Privacy in Statistical Databases*. Springer. 2004, pp. 149–161.

[53] Josep Domingo-Ferrer et al. "Self-enforcing protocols via co-utile reputation management". In: *Information Sciences* 367 (2016), pp. 159–175.

[54] Brian Donovan and Daniel B Work. "Using coarse gps data to quantify city-scale transportation system resilience to extreme events". In: *arXiv preprint arXiv:1507.06011* (2015).

[55] Joseph Leo Doob. "Stochastic processes". In: (1990).

[56] Paul Doukhan, George Oppenheim, and Murad S Taqqu. *Theory and applications of long-range dependence*. Springer Science & Business Media, 2003.

[57] Yan Duan et al. "Benchmarking Deep Reinforcement Learning for Continuous Control". In: *arXiv preprint arXiv:1604.06778* (2016).

[58] John C Duchi, Michael I Jordan, and Martin J Wainwright. "Local privacy, data processing inequalities, and minimax rates". In: *arXiv preprint arXiv:1302.3203* (2013).

[59] Pablo L Durango-Cohen. "A time series analysis framework for transportation infrastructure management". In: *Transportation Research Part B: Methodological* 41.5 (2007), pp. 493–505.

[60] James Durbin and Siem Jan Koopman. *Time series analysis by state space methods*. Vol. 38. OUP Oxford, 2012.

[61] Cynthia Dwork et al. "Calibrating Noise to Sensitivity in Private Data Analysis". In: *Proceedings of the Third Conference on Theory of Cryptography*. TCC'06. New York, NY, 2006, pp. 265–284. ISBN: 3-540-32731-2, 978-3-540-32731-8.

[62] Cynthia Dwork et al. "Our data, ourselves: Privacy via distributed noise generation". In: *Annual International Conference on the Theory and Applications of Cryptographic Techniques*. Springer. 2006, pp. 486–503.

[63] Jonathan Eckstein and Dimitri P Bertsekas. "On the Douglas—Rachford splitting method and the proximal point algorithm for maximal monotone operators". In: *Mathematical Programming* 55.1 (1992), pp. 293–318.

[64] L. C. Edie. “Discussion of Traffic Stream Measurements and Definitions”. In: *Proc. 2nd Int. Symp. on the Theory of Traffic Flow*. 1963, pp. 139–154.

[65] Charles A Eldering, Mouhamadou Lamine Sylla, and Jeffrey A Eisenach. “Is there a Moore’s law for bandwidth?” In: *IEEE Communications magazine* 37.10 (1999), pp. 117–121.

[66] S. Fan, M. Herty, and B. Seibold. “Comparative model accuracy of a data-fitted generalized Aw–Rascle–Zhang model”. In: *Networks and Heterogeneous Media* 9.2 (2014), pp. 239–268.

[67] Ahmed Fares and Walid Gomaa. “Freeway ramp-metering control based on Reinforcement learning”. In: *11th IEEE International Conference on Control & Automation (ICCA)*. IEEE. 2014, pp. 1226–1231.

[68] X Yu Felix et al. “Orthogonal random features”. In: *Advances in Neural Information Processing Systems*. 2016, pp. 1975–1983.

[69] Chelsea Finn et al. “Deep Spatial Autoencoders for Visuomotor Learning”. In: *arXiv preprint arXiv:1509.06113* (2015).

[70] Patrick Flandrin. “On the spectrum of fractional Brownian motions”. In: *Information Theory, IEEE Transactions on* 35.1 (1989), pp. 197–199.

[71] Andrew Flowers. *Uber FOIL request*. 2016. URL: `https://github.com/fivethirtyeight/uber-tlc-foil-response`.

[72] M. R. Flynn et al. “On ”jamitons,” self-sustained nonlinear traffic waves”. In: *ArXiv e-prints* (2008). arXiv: `0809.2828 [math.AP]`.

[73] M. R. Flynn et al. “Self-sustained nonlinear waves in traffic flow”. In: *Physical Review E* 79 (2009), p. 056113.

[74] M Franklin et al. “Mllib: A distributed machine learning library”. In: *NIPS Machine Learning Open Source Software* (2013).

[75] Bernard Friedland. *Control system design: an introduction to state-space methods*. Courier Corporation, 2012.

[76] Jerome Friedman, Trevor Hastie, and Robert Tibshirani. *The elements of statistical learning*. Vol. 1. Springer series in statistics Springer, Berlin, 2001.

[77] Milton Friedman. “The interpolation of time series by related series”. In: *Journal of the American Statistical Association* 57.300 (1962), pp. 729–757.

[78] Bernat Gacias and Frédéric Meunier. “Design and operation for an electric taxi fleet”. In: *Or Spectrum* 37.1 (2015), pp. 171–194.

[79] M. Garavello and B. Piccoli. *Traffic Flow on Networks: Conservation Laws Models*. AIMS series on applied mathematics. American Institute of Mathematical Sciences, 2006. ISBN: 9781601330000.

[80] Mauro Garavello and Benedetto Piccoli. “Traffic flow on a road network using the Aw–Rascle model”. In: *Communications in Partial Differential Equations* 31.2 (2006), pp. 243–275.

[81] Hong-xia Ge et al. “An improved car-following model considering influence of other factors on traffic jam”. In: *Physics Letters A* 377.1–2 (2012), pp. 9–12. ISSN: 0375-9601.

[82] S. K. Godunov. “A Difference Scheme for Numerical Solution of Discontinuous Solution of Hydrodynamic Equations”. In: *Math. Sbornik* 47 (1969), pp. 271–306.

[83] Donald Goldfarb and Shiqian Ma. “Fast multiple-splitting algorithms for convex optimization”. In: *SIAM Journal on Optimization* 22.2 (2012), pp. 533–556.

[84] Ian Goodfellow, Yoshua Bengio, and Aaron Courville. *Deep learning*. MIT press, 2016.

[85] Ian Goodfellow et al. “Generative adversarial nets”. In: *Advances in neural information processing systems*. 2014, pp. 2672–2680.

[86] Clive WJ Granger. “Causality, cointegration, and control”. In: *Journal of Economic Dynamics and Control* 12.2 (1988), pp. 551–559.

[87] Clive WJ Granger. “Investigating causal relations by econometric models and cross-spectral methods”. In: *Econometrica: Journal of the Econometric Society* (1969), pp. 424–438.

[88] J. M. Greenberg. “Congestion redux”. In: *SIAM J. Appl. Math* 64 (2004), pp. 1175–1185.

[89] Leslie Greengard and June-Yub Lee. “Accelerating the nonuniform fast Fourier transform”. In: *SIAM review* 46.3 (2004), pp. 443–454.

[90] B. D. Greenshields. “A study of traffic capacity”. In: *Highway Research Board Proceedings*. 1934, pp. 448–474.

[91] Shixiang Gu and Luca Rigazio. “Towards deep neural network architectures robust to adversarial examples”. In: *arXiv preprint arXiv:1412.5068* (2014).

[92] A K Gupta and V K Katiyar. “Analyses of shock waves and jams in traffic flow”. In: *Journal of Physics A: Mathematical and General* 38.19 (2005), p. 4069.

[93] Arvind Kumar Gupta and Isha Dhiman. “Analyses of a continuum traffic flow model for a nonlane-based system”. In: *International Journal of Modern Physics C* 25.10 (2014), p. 1450045.

[94] Arvind Kumar Gupta and Isha Dhiman. “Analyses of a continuum traffic flow model for a nonlane-based system”. In: *International Journal of Modern Physics C* 25.10 (2014), p. 1450045.

[95] Arvind Kumar Gupta and VK Katiyar. “A new anisotropic continuum model for traffic flow”. In: *Physica A: Statistical Mechanics and its Applications* 368.2 (2006), pp. 551–559.

[96] Arvind Kumar Gupta and Sapna Sharma. “Analysis of the wave properties of a new two-lane continuum model with the coupling effect”. In: *Chinese Physics B* 21.1 (2012), p. 015201.

[97] Arvind Kumar Gupta and Sapna Sharma. “Nonlinear analysis of traffic jams in an anisotropic continuum model”. In: *Chinese Physics B* 19.11 (2010), p. 110503.

[98] ArvindKumar Gupta and Isha Dhiman. “Phase diagram of a continuum traffic flow model with a static bottleneck”. In: *Nonlinear Dynamics* 79.1 (2015), pp. 663–671.

[99] James Douglas Hamilton. *Time series analysis*. Vol. 2. Princeton university press Princeton, 1994.

[100] Andrew C Harvey and AC Harvey. *Time series models*. Vol. 2. Harvester Wheatsheaf New York, 1993.

[101] Alan G Hawkes. “Point spectra of some mutually exciting point processes”. In: *Journal of the Royal Statistical Society. Series B (Methodological)* (1971), pp. 438–443.

[102] Takaki Hayashi, Nakahiro Yoshida, et al. “On covariance estimation of non-synchronously observed diffusion processes”. In: *Bernoulli* 11.2 (2005), pp. 359–379.

[103] Nicolas Heess et al. “Learning continuous control policies by stochastic value gradients”. In: *Advances in Neural Information Processing Systems*. 2015, pp. 2926–2934.

[104] Dirk Helbing. “Traffic and related self-driven many-particle systems”. In: *Reviews of modern physics* 73.4 (2001), pp. 1067–1141.

[105] Ryan Herring et al. “Mobile millennium-participatory traffic estimation using mobile phones”. In: *CPS Forum, Cyber-Physical Systems Week 2009*. 2009.

[106] Marc Hoffmann, M Rosenbaum, Nakahiro Yoshida, et al. “Estimation of the lead-lag parameter from non-synchronous data”. In: *Bernoulli* 19.2 (2013), pp. 426–461.

[107] A. Hofleitner, C. Claudel, and A. Bayen. “Reconstruction of boundary conditions from internal conditions using viability theory”. In: *American Control Conference*. IEEE. June 2012, pp. 640–645.

[108] Andreas Horni, Kai Nagel, and Kay W Axhausen. “The multi-agent transport simulation MATSim”. In: (2016).

[109] Timothy Hunter et al. “Scaling the mobile millennium system in the cloud”. In: *Proceedings of the 2nd ACM Symposium on Cloud Computing*. ACM. 2011, p. 28.

[110] Nicolas Huth and Frédéric Abergel. “High frequency lead/lag relationships - Empirical facts”. In: *Journal of Empirical Finance* 26 (2014), pp. 41–58.

[111] C ISO. “What the duck curve tells us about managing a green grid”. In: *Calif. ISO, Shap. a Renewed Futur* (), pp. 1–4.

[112] R Jayakrishnan. “A Real-Time Algorithm to Solve the Peer-to-Peer Ride-Matching Problem in a Flexible Ridesharing System”. In: (2015).

[113] I Ge Jin and Gabor Orosz. “Dynamics of connected vehicle systems with delayed acceleration feedback”. In: *Transportation Research Part C: Emerging Technologies* 46 (2014), pp. 46–64.

[114] Wen-long Jin. “On the equivalence between continuum and car-following models of traffic flow”. In: *In review* ().

[115] Jer-Nan Juang. “Applied system identification”. In: (1994).

[116] Sham Kakade. “A Natural Policy Gradient.” In: *NIPS*. Vol. 14. 2001, pp. 1531–1538.

[117] Holger Kantz, Jürgen Kurths, and Gottfried Mayer-Kress. *Nonlinear analysis of physiological data*. Springer Science & Business Media, 2012.

[118] Ioannis Karatzas and Steven Shreve. *Brownian motion and stochastic calculus*. Vol. 113. Springer Science & Business Media, 2012.

[119] Kari Karhunen. “Über die Struktur stationärer zufälliger Funktionen”. In: *Arkiv för Matematik* 1.2 (1950), pp. 141–160.

[120] B. S. Kerner and H. Rehborn. “Experimental Properties of Phase Transitions in Traffic Flow”. In: *Phys. Rev. Lett.* 79 (20 Nov. 1997), pp. 4030–4033.

[121] Brucek K Khailany et al. “A programmable 512 GOPS stream processor for signal, image, and video processing”. In: *IEEE Journal of Solid-State Circuits* 43.1 (2008), pp. 202–213.

[122] Hassan K Khalil. “Noninear Systems”. In: *Prentice-Hall, New Jersey* 2.5 (1996), pp. 5–1.

[123] Alex Krizhevsky, Ilya Sutskever, and Geoffrey E Hinton. “Imagenet classification with deep convolutional neural networks”. In: *Advances in neural information processing systems*. 2012, pp. 1097–1105.

[124] Vipin Kumar et al. *Introduction to parallel computing: design and analysis of algorithms*. Vol. 400. Benjamin/Cummings Redwood City, 1994.

[125] Pierre-Olivier Lamare and Nikolaos Bekiaris-Liberis. “Control of 2 2 Linear Hyperbolic Systems: Backstepping-Based Trajectory Generation and PI-Based Tracking”. In: *In review* ().

[126] Leslie Lamport et al. “Paxos made simple”. In: *ACM Sigact News* 32.4 (2001), pp. 18–25.

[127] B Boser Le Cun et al. “Handwritten digit recognition with a back-propagation network”. In: *Advances in neural information processing systems*. Citeseer. 1990.

[128] Caroline Le Floch, Francois Belletti, and Scott Moura. “Optimal charging of electric vehicles for load shaping: A dual-splitting framework with explicit convergence bounds”. In: *IEEE Transactions on Transportation Electrification* 2.2 (2016), pp. 190–199.

[129] Caroline Le Floch et al. “Distributed optimal charging of electric vehicles for demand response and load shaping”. In: *Decision and Control (CDC), 2015 IEEE 54th Annual Conference on*. IEEE. 2015, pp. 6570–6576.

[130] Jerome Le Ny and George J Pappas. “Differentially private filtering”. In: *Automatic Control, IEEE Transactions on* 59.2 (2014), pp. 341–354.

[131] Jean-Patrick Lebacque, Salim Mammar, and Habib Haj-Salem. “The Aw–Rascle and Zhang’s model: Vacuum problems, existence and regularity of the solutions of the Riemann problem”. In: *Transp. Res. Part B* 41.7 (2007), pp. 710–721.

[132] Soonhui Lee et al. “Improving fleet utilization for carriers by interval scheduling”. In: *European Journal of Operational Research* 218.1 (2012), pp. 261–269.

[133] Charles-Albert Lehalle and Sophie Laruelle. *Market microstructure in practice*. World Scientific, 2013.

[134] Sergey Levine and Pieter Abbeel. "Learning neural network policies with guided policy search under unknown dynamics". In: *Advances in Neural Information Processing Systems*. 2014, pp. 1071–1079.

[135] Sergey Levine and Vladlen Koltun. "Guided Policy Search". In: *International Conference on Machine Learning (ICML 2013)*. 2013.

[136] Sergey Levine et al. "End-to-end training of deep visuomotor policies". In: *arXiv preprint arXiv:1504.00702* (2015).

[137] Philip Levis et al. "TinyOS: An operating system for sensor networks". In: *Ambient intelligence* 35 (2005), pp. 115–148.

[138] Wentian Li and Kunihiko Kaneko. "Long-range correlation and partial 1/fα spectrum in a noncoding DNA sequence". In: *EPL (Europhysics Letters)* 17.7 (1992), p. 655.

[139] Ziqi Liao. "Real-time taxi dispatching using global positioning systems". In: *Communications of the ACM* 46.5 (2003), pp. 81–83.

[140] Ziqi Liao. "Taxi dispatching via global positioning systems". In: *IEEE Transactions on Engineering Management* 48.3 (2001), pp. 342–347.

[141] M. J. Lighthill and J. B Whitham. "On kinematic waves. II: A Theory of traffic flow on long crowded roads". In: *Proc. Royal. Soc.* (1955), pp. 317–345.

[142] Xavier Litrico and Vincent Fromion. "H-infinity control of an irrigation canal pool with a mixed control politics". In: *IEEE Transactions on Control Systems Technology* 14.1 (2006), pp. 99–111.

[143] Xavier Litrico and Vincent Fromion. *Modeling and control of hydrosystems*. Springer, 2009.

[144] Lennart Ljung. "System identification". In: *Signal analysis and prediction*. Springer, 1998, pp. 163–173.

[145] Nicholas R Lomb. "Least-squares frequency analysis of unequally spaced data". In: *Astrophysics and space science* 39.2 (1976), pp. 447–462.

[146] Helmut Lütkepohl. *New introduction to multiple time series analysis*. Springer Science & Business Media, 2005.

[147] Shuo Ma, Yu Zheng, and Ouri Wolfson. "T-share: A large-scale dynamic taxi ridesharing service". In: *Data Engineering (ICDE), 2013 IEEE 29th International Conference on*. IEEE. 2013, pp. 410–421.

[148] M Maciejewski and Kai Nagel. *Simulation and dynamic optimization of taxi services in MATSim*. Tech. rep. VSP Working Paper 13-05, TU Berlin, Transport Systems Planning and Transport Telematics. www. vsp. tu-berlin. de/publications, 2013.

[149] John M Maheu and Thomas H McCurdy. “News arrival, jump dynamics, and volatility components for individual stock returns”. In: *The Journal of Finance* 59.2 (2004), pp. 755–793.

[150] Hani Mahmassani, Yongjin Kim, and Patrick Jaillet. “Local optimization approaches to solve dynamic commercial fleet management problems”. In: *Transportation Research Record: Journal of the Transportation Research Board* 1733 (2000), pp. 71–79.

[151] Stephane Mallat. *A wavelet tour of signal processing: the sparse way*. Academic press, 2008.

[152] Stephane G Mallat. “A theory for multiresolution signal decomposition: the wavelet representation”. In: *IEEE transactions on pattern analysis and machine intelligence* 11.7 (1989), pp. 674–693.

[153] Salim Mammar, Jean-Patrick Lebacque, and Habib Haj Salem. “Riemann Problem Resolution and Godunov Scheme for the Aw-Rascle-Zhang Model”. In: *Transportation Science* 43.4 (2009), pp. 531–545.

[154] Benoit B Mandelbrot. *Fractals and Scaling in Finance: Discontinuity, Concentration, Risk. Selecta Volume E*. Springer Science & Business Media, 2013.

[155] Horia Mania et al. “Perturbed iterate analysis for asynchronous stochastic optimization”. In: *arXiv preprint arXiv:1507.06970* (2015).

[156] Michael Mauch and Michael J. Cassidy. “Freeway traffic oscillations: Observations and predictions”. In: *15th Int. Symp. on Transportation and Traffic Theory*. Elsevier, 2002, pp. 653–674.

[157] Adolf D May et al. “Loop detector data collection and travel time measurement in the Berkeley highway laboratory”. In: *California Partners for Advanced Transit and Highways (PATH)* (2003).

[158] Fei Miao et al. “Robust Taxi Dispatch under Model Uncertainties”. In: *Conference for Decision and Control* (2015).

[159] Fei Miao et al. “Taxi dispatch with real-time sensing data in metropolitan areas: a receding horizon control approach”. In: *Proceedings of the ACM/IEEE Sixth International Conference on Cyber-Physical Systems*. ACM. 2015, pp. 100–109.

[160] Volodymyr Mnih et al. “Playing atari with deep reinforcement learning”. In: *arXiv preprint arXiv:1312.5602* (2013).

[161] Gordon E Moore. “Cramming more components onto integrated circuits, Reprinted from Electronics, volume 38, number 8, April 19, 1965, pp. 114 ff.” In: *IEEE Solid-State Circuits Society Newsletter* 20.3 (2006), pp. 33–35.

[162] S. Moutari and M. Rascle. “A Hybrid Lagrangian Model Based on the Aw–Rascle Traffic Flow Model”. In: *SIAM Journal on Applied Mathematics* 68.2 (2007), pp. 413–436.

[163] Manfred Mudelsee. “Climate time series analysis”. In: *Atmospheric and* 397 (2010).

[164] Ajith Muralidharan, Gunes Dervisoglu, and Roberto Horowitz. "Freeway traffic flow simulation using the link node cell transmission model". In: *2009 American Control Conference*. IEEE. 2009, pp. 2916–2921.

[165] Kevin P Murphy. *Machine learning: a probabilistic perspective*. MIT press, 2012.

[166] IJ Nagrath. *Control systems engineering*. New Age International, 2006.

[167] Ion Necoara and Johan AK Suykens. "Application of a smoothing technique to decomposition in convex optimization". In: *IEEE Transactions on Automatic control* 53.11 (2008), pp. 2674–2679.

[168] Angelia Nedic and Asuman Ozdaglar. "Distributed subgradient methods for multi-agent optimization". In: *Automatic Control, IEEE Transactions on* 54.1 (2009), pp. 48–61.

[169] Daniel Neil, Michael Pfeiffer, and Shih-Chii Liu. "Phased LSTM: Accelerating Recurrent Network Training for Long or Event-based Sequences". In: *Advances in Neural Information Processing Systems*. 2016, pp. 3882–3890.

[170] Arkadi Nemirovski et al. "Robust stochastic approximation approach to stochastic programming". In: *SIAM Journal on Optimization* 19.4 (2009), pp. 1574–1609.

[171] G. F. Newell. "A simplified theory of kinematic waves in highway traffic, part II: queueing at freeway bottlenecks". In: *Transp. Res. B* 27.7 (1993), pp. 289–303.

[172] G. F. Newell. "Nonlinear Effects in the Dynamics of Car Following". In: *Operations Research* 9.2 (1961), pp. 209–229.

[173] S. Osher. "Riemann Solvers, the Entropy Condition, and Difference". In: *SIAM Journal on Numerical Analysis* 21.2 (1984), pp. 217–235.

[174] Markos Papageorgiou, Jean-Marc Blosseville, and Habib Hadj-Salem. "Macroscopic modelling of traffic flow on the Boulevard Peripherique in Paris". In: *Transp. Res. B* 23.1 (1989), pp. 29–47. ISSN: 0191-2615.

[175] Markos Papageorgiou, Habib Hadj-Salem, and F Middelham. "ALINEA local ramp metering: Summary of field results". In: *Transportation Research Record: Journal of the Transportation Research Board* 1603 (1997), pp. 90–98.

[176] Neal Parikh and Stephen Boyd. "Block splitting for distributed optimization". In: *Mathematical Programming Computation* 6.1 (2014), pp. 77–102.

[177] Emanuel Parzen. *Time Series Analysis of Irregularly Observed Data: Proceedings of a Symposium Held at Texas A & M University, College Station, Texas February 10–13, 1983*. Vol. 25. Springer Science & Business Media, 2012.

[178] H. J. Payne. *Models of Freeway Traffic and Control*. Simulation Councils, Incorporated, 1971.

[179] David Peleg. "Distributed computing". In: *SIAM Monographs on discrete mathematics and applications* 5 (2000).

[180] Donald B Percival and Andrew T Walden. *Wavelet methods for time series analysis*. Vol. 4. Cambridge university press, 2006.

[181] Jan Peters and Stefan Schaal. “Policy gradient methods for robotics”. In: *Intelligent Robots and Systems, 2006 IEEE/RSJ International Conference on*. IEEE. 2006, pp. 2219–2225.

[182] “Phase transition of traffic states with on-ramp”. In: *Physica A: Statistical Mechanics and its Applications* 371.2 (2006), pp. 674–682.

[183] Benedetto Piccoli et al. “Second-order models and traffic data from mobile sensors”. In: *Transp. Res. C* 52 (2015), pp. 32–56.

[184] Mert Pilanci, Laurent E Ghaoui, and Venkat Chandrasekaran. “Recovery of sparse probability measures via convex programming”. In: *Advances in Neural Information Processing Systems*. 2012, pp. 2420–2428.

[185] Mert Pilanci, Martin J Wainwright, and Laurent El Ghaoui. “Sparse learning via Boolean relaxations”. In: *Mathematical Programming* 151.1 (2015), pp. 63–87.

[186] Vladas Pipiras and Murad S Taqqu. *Long-range dependence and self-similarity*. Vol. 45. Cambridge University Press, 2017.

[187] Sreeram Potluri et al. “Efficient inter-node MPI communication using GPUDirect RDMA for InfiniBand clusters with NVIDIA GPUs”. In: *Parallel Processing (ICPP), 2013 42nd International Conference on*. IEEE. 2013, pp. 80–89.

[188] Tobias Preis et al. “GPU accelerated Monte Carlo simulation of the 2D and 3D Ising model”. In: *Journal of Computational Physics* 228.12 (2009), pp. 4468–4477.

[189] Xinwu Qian, Xianyuan Zhan, and Satish V Ukkusuri. “Characterizing urban dynamics using large scale taxicab data”. In: *Transportation Research Record* 35.36 (2013), p. 37.

[190] M. Rascle. “An improved macroscopic model of traffic flow: derivation and links with the Lighthill-Whitham model”. In: *Mathematical and computer modelling* 35 (2002), pp. 581–590.

[191] Carl Edward Rasmussen. “Gaussian processes for machine learning”. In: (2006).

[192] Benjamin Recht et al. “Hogwild: A lock-free approach to parallelizing stochastic gradient descent”. In: *Advances in neural information processing systems*. 2011, pp. 693–701.

[193] Fabrice Reclus and Kristen Drouard. “Geofencing for fleet & freight management”. In: *Intelligent Transport Systems Telecommunications,(ITST), 2009 9th International Conference on*. IEEE. 2009, pp. 353–356.

[194] Jack Reilly et al. “Adjoint-based optimization on a network of discretized scalar conservation laws with applications to coordinated ramp metering”. In: *Journal of optimization theory and applications* 167.2 (2015), pp. 733–760.

[195] Kasra Rezaee. “Decentralized Coordinated Optimal Ramp Metering using Multi-agent Reinforcement Learning”. PhD thesis. University of Toronto, 2014.

[196] Paul I. Richards. “Shock Waves on the Highway”. In: *Operations Research* 4.1 (1956), pp. 42–51.

[197] Francesc Robust, Carlos F Daganzo, and Reginald R Souleyrette. “Implementing vehicle routing models”. In: *Transportation Research Part B: Methodological* 24.4 (1990), pp. 263–286.

[198] R Tyrrell Rockafellar. “Monotone operators and the proximal point algorithm”. In: *SIAM journal on control and optimization* 14.5 (1976), pp. 877–898.

[199] R.T. Rockafellar. *Convex Analysis*. Convex Analysis. Princeton University Press, 1997. ISBN: 9780691015866.

[200] Stuart Russell, Peter Norvig, and Artificial Intelligence. “A modern approach”. In: *Artificial Intelligence. Prentice-Hall, Egnlewood Cliffs* 25 (1995), p. 27.

[201] David Sánchez, Sergio Martınez, and Josep Domingo-Ferrer. “Co-utile P2P ridesharing via decentralization and reputation management”. In: *Transportation Research Part C: Emerging Technologies* 73 (2016), pp. 147–166.

[202] Adella Santos et al. *Summary of travel trends: 2009 national household travel survey*. Tech. rep. 2011.

[203] Rathindra Sarathy and Krishnamurty Muralidhar. “Evaluating Laplace Noise Addition to Satisfy Differential Privacy for Numeric Data.” In: *Transactions on Data Privacy* 4.1 (2011), pp. 1–17.

[204] David Schrank, Bill Eisele, and Tim Lomax. “TTI 2012 urban mobility report”. In: *Texas A&M Transportation Institute. The Texas A&M University System* (2012).

[205] John Schulman et al. “High-dimensional continuous control using generalized advantage estimation”. In: *arXiv preprint arXiv:1506.02438* (2015).

[206] John Schulman et al. “Trust region policy optimization”. In: *arXiv preprint arXiv:1502.05477* (2015).

[207] B. Seibold et al. “Constructing set-valued fundamental diagrams from jamiton solutions in second order traffic models”. In: *ArXiv e-prints* (Apr. 2012). arXiv: `1204.5510 [math.AP]`.

[208] Kiam Tian Seow, Nam Hai Dang, and Der-Horng Lee. “A collaborative multiagent taxi-dispatch system”. In: *Automation Science and Engineering, IEEE Transactions on* 7.3 (2010), pp. 607–616.

[209] Konstantin Shvachko et al. “The hadoop distributed file system”. In: *Mass Storage Systems and Technologies (MSST), 2010 IEEE 26th Symposium on*. IEEE. 2010, pp. 1–10.

[210] David Silver et al. “Mastering the game of Go with deep neural networks and tree search”. In: *Nature* 529.7587 (2016), pp. 484–489.

[211] David H Smith et al. “The impact of prescribing safety alerts for elderly persons in an electronic medical record: an interrupted time series evaluation”. In: *Archives of Internal Medicine* 166.10 (2006), pp. 1098–1104.

[212] Torsten Söderström and Petre Stoica. *System identification*. Prentice-Hall, Inc., 1988.

[213] Jonathan E Spingarn. “Applications of the method of partial inverses to convex programming: decomposition”. In: *Mathematical Programming* 32.2 (1985), pp. 199–223.

[214] Terry W. Sturm. *Open channel hydraulics*. McGraw-Hill, 2001.

[215] Yuki Sugiyama et al. “Traffic jams without bottlenecks, experimental evidence for the physical mechanism of the formation of a jam”. In: *New J. Phys.* 10 (2008), p. 033001.

[216] Hongyu Sun et al. “Use of local linear regression model for short-term traffic forecasting”. In: *Transportation Research Record: Journal of the Transportation Research Board* 1836 (2003), pp. 143–150.

[217] Richard S Sutton et al. “Policy Gradient Methods for Reinforcement Learning with Function Approximation.” In: *NIPS*. Vol. 99. 1999, pp. 1057–1063.

[218] Latanya Sweeney. “k-anonymity: A model for protecting privacy”. In: *International Journal of Uncertainty, Fuzziness and Knowledge-Based Systems* 10.05 (2002), pp. 557–570.

[219] István Szita and András Lörincz. “Learning Tetris using the noisy cross-entropy method”. In: *Neural computation* 18.12 (2006), pp. 2936–2941.

[220] Benjamin M Tabak and Daniel O Cajueiro. “Are the crude oil markets becoming weakly efficient over time? A test for time-varying long-range dependence in prices and volatility”. In: *Energy Economics* 29.1 (2007), pp. 28–36.

[221] Andrew S Tanenbaum and Maarten Van Steen. *Distributed systems: principles and paradigms*. Prentice-Hall, 2007.

[222] T.T. Tchrakian and S. Zhuk. “A Macroscopic Traffic Data-Assimilation Framework Based on the Fourier-Galerkin Method and Minimax Estimation”. In: *IEEE Transactions on Intelligent Transportation Systems* 16.1 (2014), pp. 452–464.

[223] Ashish Thusoo et al. “Hive-a petabyte scale data warehouse using hadoop”. In: *Data Engineering (ICDE), 2010 IEEE 26th International Conference on*. IEEE. 2010, pp. 996–1005.

[224] Naftali Tishby and Noga Zaslavsky. “Deep learning and the information bottleneck principle”. In: *Information Theory Workshop (ITW), 2015 IEEE*. IEEE. 2015, pp. 1–5.

[225] Martin Treiber, Ansgar Hennecke, and Dirk Helbing. “Congested traffic states in empirical observations and microscopic simulations”. In: *Physical Review E* 62.2 (2000), pp. 1805–1824.

[226] Martin Treiber and Arne Kesting. “Evidence of convective instability in congested traffic flow: A systematic empirical and theoretical investigation”. In: *Transportation Research Part B: Methodological* 45.9 (2011), pp. 1362–1377.

[227] Martin Treiber and Arne Kesting. *Traffic Flow Dynamics: Data, Models and Simulation*. Springer, 2013. ISBN: 9783642324598.

[228] Martin Treiber, Arne Kesting, and Dirk Helbing. "Three-phase traffic theory and two-phase models with a fundamental diagram in the light of empirical stylized facts". In: *Transp. Res. B* 44.8 (2010), pp. 983–1000.

[229] Ruey S Tsay. *Analysis of financial time series*. Vol. 543. John Wiley & Sons, 2005.

[230] Herke Van Hoof, Jan Peters, and Gerhard Neumann. "Learning of Non-Parametric Control Policies with High-Dimensional State Features." In: *AISTATS*. 2015.

[231] Mathukuma Vidyasagar. *Nonlinear systems analysis*. Prentice Hall, 1978.

[232] Martin J Wainwright, Michael I Jordan, and John C Duchi. "Privacy aware learning". In: *Advances in Neural Information Processing Systems*. 2012, pp. 1430–1438.

[233] Nianfeng Wan et al. "Prediction on Travel-Time Distribution for Freeways Using Online Expectation Maximization Algorithm". In: *Transportation Research Board 93rd Annual Meeting*. 2013, pp. 14–3221.

[234] Yibing Wang and Markos Papageorgiou. "Real-time freeway traffic state estimation based on extended Kalman filter: a general approach". In: *Transp. Res. B* 39.2 (2005), pp. 141–167. ISSN: 0191-2615.

[235] Jonathan A. Ward and R. Eddie Wilson. "Criteria for convective versus absolute string instability in car-following models". In: *Proceedings of the Royal Society of London A: Mathematical, Physical and Engineering Sciences* (2011). DOI: 10.1098/rspa.2010.0437.

[236] Stanley L Warner. "Randomized response: A survey technique for eliminating evasive answer bias". In: *Journal of the American Statistical Association* 60.309 (1965), pp. 63–69.

[237] Pradeep Chathuranga Weeraddana et al. "On the privacy of optimization approaches". In: *arXiv preprint arXiv:1210.3283* (2012).

[238] G. B. Whitham. *Linear and Nonlinear Waves*. Wiley, 1974. ISBN: 9780471940906.

[239] Norbert Wiener. *Extrapolation, interpolation, and smoothing of stationary time series*. Vol. 2. MIT press Cambridge, MA, 1949.

[240] Ronald J Williams. "Simple statistical gradient-following algorithms for connectionist reinforcement learning". In: *Machine learning* 8.3-4 (1992), pp. 229–256.

[241] Wayne Wolf, Burak Ozer, and Tiehan Lv. "Smart cameras as embedded systems". In: *Computer* 35.9 (2002), pp. 48–53.

[242] KI Wong, SC Wong, and Hai Yang. "Modeling urban taxi services in congested road networks with elastic demand". In: *Transportation Research Part B: Methodological* 35.9 (2001), pp. 819–842.

[243] Jianjun Wu et al. "Urban transit system as a scale-free network". In: *Modern Physics Letters B* 18.19n20 (2004), pp. 1043–1049.

[244] Hai Yang et al. "Equilibria of bilateral taxi–customer searching and meeting on networks". In: *Transportation Research Part B: Methodological* 44.8 (2010), pp. 1067–1083.

[245] Fisher Yu and Vladlen Koltun. “Multi-scale context aggregation by dilated convolutions”. In: *International Conference on Learning Representations (ICLR)*. 2016.

[246] Matei Zaharia et al. “Resilient distributed datasets: A fault-tolerant abstraction for in-memory cluster computing”. In: *Proceedings of the 9th USENIX conference on Networked Systems Design and Implementation*. USENIX Association. 2012, pp. 2–2.

[247] H. M. Zhang. “A non-equilibrium traffic model devoid of gas-like behavior”. In: *Transp. Res. B* 36 (2002), pp. 275–290.

[248] H. M. Zhang. “A theory of nonequilibrium traffic flow”. In: *Transp. Res. B* 32.7 (1998), pp. 485–498.

[249] Zuduo Zheng et al. “Freeway Traffic Oscillations: Microscopic Analysis of Formations and Propagations using Wavelet Transform”. In: *Trans. Res. Part B* 45.9 (2011), pp. 1378–1388.

[250] Kemin Zhou, John Comstock Doyle, Keith Glover, et al. *Robust and optimal control*. Vol. 40. Prentice hall New Jersey, 1996.

[251] Minghui Zhu and Sonia Martınez. “On distributed convex optimization under inequality and equality constraints”. In: *IEEE Transactions on Automatic Control* 57.1 (2012), pp. 151–164.

[252] Barret Zoph and Quoc V Le. “Neural architecture search with reinforcement learning”. In: *arXiv preprint arXiv:1611.01578* (2016).