Coordinating users of shared facilities via data-driven predictive assistants and game theory

Philipp Geiger[a,b], Michel Besserve[a,c], Justus Winkelmann[d], Claudius Proissl[a], Bernhard Schölkopf[a]
philipp.geiger@tuebingen.mpg.de
[a]Max Planck Institute for Intelligent Systems, [b]Bosch Center for Artificial Intelligence,
[c]Max Planck Institute for Biological Cybernetics, [d]Bonn Graduate School of Economics

Abstract

We study data-driven assistants that provide congestion forecasts to users of shared facilities (roads, cafeterias, etc.), to support coordination between them, and increase efficiency of such collective systems. Key questions are: (1) when and how much can (accurate) predictions help for coordination, and (2) which assistant algorithms reach optimal predictions?

First we lay conceptual ground for this setting where user preferences are a priori unknown and predictions influence outcomes. Addressing (1), we establish conditions under which self-fulfilling prophecies, i.e., "perfect" (probabilistic) predictions of what will happen, solve the coordination problem in the game-theoretic sense of selecting a Bayesian Nash equilibrium (BNE). Next we prove that such prophecies exist even in large-scale settings where only aggregated statistics about users are available. This entails a new (nonatomic) BNE existence result. Addressing (2), we propose two assistant algorithms that sequentially learn from users' reactions, together with optimality/convergence guarantees. We validate one of them in a large real-world experiment.

1 INTRODUCTION

Data-driven interventions on social and economic systems are on the rise, but it remains a challenge to understand when and how they can improve such systems in terms of peoples' actual utilities. Here we consider central *predictive coordination assistants*, that, in the simplest case, work as follows: The assistant provides a congestion forecast A to users of some facility, based on past observations. The users trust A to be a good forecast, and individually optimize their facility use based on it, e.g., their arrival time slot, to coordinate and avoid crowds. Thereby they generate an observable outcome Y, which A is a forecast for. In particular, forecast A *influences* outcome Y. Versions of such assistants exist for roads, trains, swimming pools, etc. [Google, 2019, ASFA, 2019, DB, 2019], or, in our experiment, a cafeteria, see Figure 1.

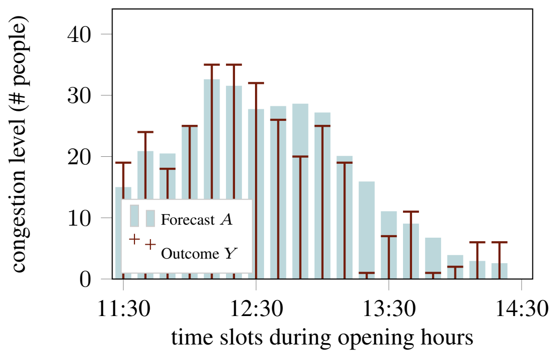

Figure 1: Example of $A \in \mathbb{R}^{18}$, the assistant's (point) forecast, and outcome $Y \in \mathbb{R}^{18}$, in our cafeteria experiment (A updated between but not within days, Section 6).

Main goals and contributions: We aim at (1) understanding to what extent optimally accurate assistant predictions can help coordination between users (Goal 1), and (2) designing sequential assistant algorithms that achieve optimal predictions (Goal 2). Our contributions:

- Introducing new concepts for this setting, we analyze when the assistant achieving a "perfect" (probabilistic) prediction A of Y, i.e., a "self-fulfilling prophecy", is equivalent to "solving" coordination in the sense of selecting a Bayesian Nash equilibrium (BNE) (Theorem 1).
- We establish conditions under which such a prophecy exists even in large-scale settings with only population-level aggregated user data (Theorem 2), using the Leray-Schauder-Tychonoff fixed point theorem. This entails a new nonatomic game BNE existence result (Corollary 3).

• We propose learning assistant Algorithms 1 and 2 (controllers), for large-/small-scale settings, with optimality/convergence guarantees (Propositions 1 and 2).

• We report positive evaluation of Algorithm 1 in a large-scale real-world cafeteria experiment (Section 6).

Overview over closest related research: Within *game theory*, dynamics/equilibria of agents are studied that *learn* about each other by repeatedly interacting, but without central assistant [Shoham and Leyton-Brown, 2008]. Besides this, the following game-theoretic work usually assumes that agents reason fully rationally based on their own a priori given beliefs about other agents, instead of using a predictive assistant informed by past behavioral data: *Congestion games* [Nisan et al., 2007] formalize coordination in certain shared facilities. *(Allocation) mechanisms* are designed [Nisan et al., 2007] that maximize social welfare (which is defined in terms of agent's a priori unknown preferences), in spite of agents being self-interested, by using incentives. Unlike our assistant, these mechanisms often fully control the outcome. And we consider "solving coordination" in game-theoretic (equilibrium selection) rather than in social welfare terms. Beyond game theory, certain *smart cities* research [Mareček et al., 2015] uses a *control-theoretic* approach for congested facilities, but they fix an objective that does not in general account for users' individual, a priori unknown preferences. For further related work, see Section 7, and Section B in [Geiger et al., 2019] (the extended version of this paper).

2 PRELIMINARIES AND SETTING

Notation: For a vector b, b_i or $[b]_i$ is the i-th component, b_{-i} means dropping b_i, and (b_i, b_{-i}) reads b. For a variable Z, range_Z denotes the (implicitly given) range.

2.1 General setting and assistant-based system

Let us first introduce the general users' decision problem. We leave it fairly abstract so that later on we can consider different forms of decision making scenarios based on it.

Setting 1 (***General (one-stage) setting***)**.** *There is a finite set* $K = \{0, \ldots, |K| - 1\}$ *of* slots[1], *and a set* I*, interpreted as* users *(here and in Section 3.2) or* types of users *(in Section 3.3), respectively. Each user* $i \in I$*:*

- *receives a (private)* signal W_i*,*
- *as (private)* action B_i *chooses a slot in* K*, and*
- *experiences (private)* utility U_i *he wants to maximize.*

Let $W = (W_i)_{i \in I}$*,* $B = (B_i)_{i \in I}$ *and* $U = (U_i)_{i \in I}$*. Besides the private signals, there is a* publicly available signal V*, and some underlying* (latent) state X*. And there is a* publicly observable outcome $Y = \bar{Y}(X, B)$*, for some function* $\bar{Y}$*.*[2] *We assume there is a "true" distribution* $P(X, V, W)$*. If not stated otherwise, we assume that all users* i *are* **inference-assistable***, i.e.,*

$$U_i = \tilde{U}_i(W_i, B_i, h_i(\bar{Y}(X, B))), \quad (1)$$

for (continuous) functions $\tilde{U}_i, h_i$ *such that* $h_i(\bar{Y}(x, (b_i, b_{-i})))$ *does not depend on* b_i*, for all* b, x*.*[3] *And let all users* i *be* **assistant-separable***, i.e.,* $h_i(Y) \perp\!\!\!\perp W_i | V$ *(for any possible mechanism that generates* B *from* V, W*).*[4]

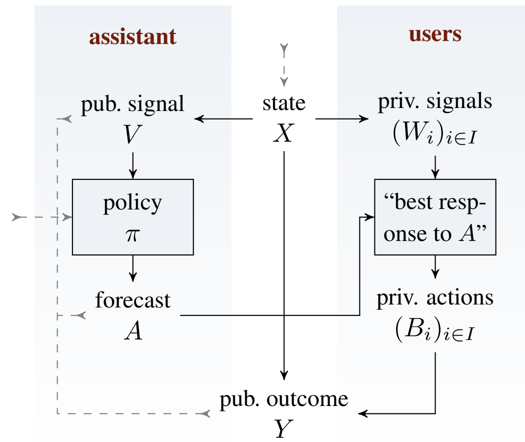

Figure 2: (Causal) diagram of the *assistant-based system* M (without U). The dashed gray arrows indicate the dynamic extension M^{dyn} we will introduce in Section 4.

Setting 1 leaves open how users reason/decide. Our main object of study is a system that enriches this setting: user i chooses B_i that maximizes her expected utility, given Y is distributed according to a central assistant's forecast:

Definition 1 (***Assistant-based system*** M)**.** *Based on Setting 1, or any restricted version, let the* assistant-based (one-stage) system M *be defined by the following objects and assumptions additional to Setting 1, as depicted by the (causal) Bayes net [Pearl, 2000] in Figure 2: There is an assistant that takes public signal* V *as input and outputs* A*, a probabilistic* forecast *for the public outcome* Y*, based on* policy π*, i.e.,* $A = \pi(V)$*. That is,* A *is a distribution over* Y *(later we also consider point forecasts).*

[1] K can be e.g., several facilities, or time slots in one facility.

[2] This models the fact that actions B may no be observed publicly, but just, say, some stochastic aggregation of them.

[3] The intuition behind this constraint on the utility functions is that users' decision making can be discerned into (1) an optimization performed by the users and (2) the task of predicting Y which can be "outsourced" to an assistant.

[4] This means, roughly, that the users do not know more about each other than is contained in the public V.

User $i \in I$ takes forecast A (besides her private signal W_i) as input, and acts **assistant-best-respondingly**, *i.e.,*

$$B_i \in \arg\max_{b_i'} \mathbb{E}_{Y' \sim A}(\tilde{U}_i(W_i, b_i', h_i(Y'))) \quad (2)$$

(breaking ties via K). A joint $P_M(X, V, W, A, B, Y, U)$ is induced by all the above (measurable) equations and $P(X, V, W)$. We may write $P_{M,\pi}$ and $\mathbb{E}_{M,\pi}$ to make the dependence on π explicit. Y is observed, but the specific $P(X, V, W)$ and $\tilde{U}_i$'s are a priori unknown, and X, W, B and utilities U are unobserved by the assistant.

2.2 Game-theoretic tools to characterize efficiency

We want to analyze the degree of efficiency that assistant-based coordination can achieve. For this, we now define what a *"solution"* of the coordination problem would be (a BNE), accounting for users' preferences. This is based on an *idealized*, assistant-free version of Setting 1 (a Bayesian game), where users i have *informed priors and unlimited inference abilities themselves*, using W_i and V as input. Then, for any user behavior that arises in the assistant-based system, we can check if it is (or rather: corresponds to) such a solution. For background on game theory and the Bayesian game definition we use, see Section A in [Geiger et al., 2019].

Definition 2 (***Benchmark (assistant-free) game*** G)**.** *Based on Setting 1, or any restriction of it, let the* benchmark game G *be defined as the Bayesian game canonically associated to this setting: Each user $i \in I$ is a player who has: signal $(W_i, V) \in \text{range}_{(W_i,V)}$, (measurable) utility function $\text{range}_{(X,W_i,B)} \to \mathbb{R}$ given by Eq. 1, and action $B_i \in K$. The utility functions are common knowledge and $P(X, V, W)$ is the common prior.*

As usual, a ***(pure) strategy profile*** for G is a tuple $s = (s_i)_{i \in I}$ of *(measurable, pure)* ***strategies*** s_i : $\text{range}_{(W_i,V)} \to K$[5], $i \in I$. A strategy profile s is a ***Bayesian Nash equilibrium (BNE)*** of G, if

$$s_i(w_i, v) \in \arg\max_{b_i} \mathbb{E}_{G,(b_i,s_{-i})}(U_i | w_i, v), \quad (3)$$

for (almost) all i, w_i, v; with U_i as in Eq. 1, and $\mathbb{E}_{G,(b_i,s_{-i})}$ the expectation under $P_{G,(b_i,s_{-i})}(X, \ldots, U)$ obtained by "plugging" strategy profile (b_i, s_{-i}) into game G (b_i here means the constant strategy).[6] We call the BNE *strict* if the argmax is unique.

To relate M to G, given an assistant policy π of the assistant-based system M, we define the ***corresponding strategy profile*** s_π by the composition of π and users' subsequent (deterministic) "best-response" action, i.e.,

$$[s_\pi]_i(w_i, v) := \mathbb{E}_{M,\pi}(B_i | w_i, v), \quad \text{for all } i, w_i, v. \quad (4)$$

[5]I.e., the strategy s_i maps player i's signal to her action

[6]I.e., each player's strategy is a best response to the others.

Conversely, given a strategy profile s of the benchmark game G, we define the ***corresponding assistant policy***[7]

$$\pi_s(v) := P_{G,s}(Y | v), \quad \text{for all } v. \quad (5)$$

2.3 Objective functions

We consider the following two objective functions for the assistant's policy π, where, as we will see, the former can be seen as a directly measurable "proxy" to the latter:

- ***(probabilistic) prediction accuracy objective (loss)***:

$$L_\pi^{\text{pred}} := \mathbb{E}\left(d\left(P_{M,\pi}\left(Y|V\right), \pi\left(V\right)\right)\right), \text{ for all } \pi, \quad (6)$$

with $d(\cdot, \cdot)$ some arbitrary but fixed statistical distance which is 0 iff both distributions coincide;

- ***equilibrium selection objective***[8]: π is optimal iff the corresponding strategy profile s_π (Eq. 4) is a BNE of the benchmark game G.

We call a policy π that tries to optimize L_π^{pred} loosely a *(predictive coordination) assistant*, and a π that achieves $L_\pi^{\text{pred}} = 0$ a ***(formally) self-fulfilling prophecy (policy)***.

3 THE UTILITY OF PREDICTIONS FOR COORDINATION – ANALYSIS

In this section, we pursue the following goal, for which the one-stage setting we introduced in Section 2 is sufficient (we will introduce a *repeated* version in Section 4).

Goal 1. *Understand the conditions when, and the degree to which, assistants, that achieve $\pi \in \arg\max_{\pi'} L_{\pi'}^{pred}$, help solve the problem of coordination between users of facilities (here: in terms of equilibrium selection).*

3.1 Characterization step in general setting

Theorem 1 (Self-Fulfilling Prophecy Characterization)**.** *We have, in the general setting (Setting 1, with all users being inference-assistable and assistant-separable):*

- *If the assistant policy π in the assistant-based system M (where all users are assistant-best-responding) is a self-fulfilling prophecy (i.e., $L_\pi^{pred} = 0$), then the corresponding strategy profile s_π is a Bayesian Nash equilibrium (BNE) of the benchmark game G.*
- *Conversely, if the strategy profile s is a strict BNE of the benchmark game G, then the corresponding assistant policy π_s is a self-fulfilling prophecy.*

[7]I.e., the assistant as forecast takes the distribution of outcome $Y = \bar{Y}(X, B)$ given $V = v$, under G, s.

[8]Equilibrium selection is a game-theoretic formulation of solving coordination [Nisan et al., 2007]. Clearly, equilibria can still be inefficient in terms of social welfare, see Section 7.

The proof is in Section D.1 in [Geiger et al., 2019]. Since in Section 2 we were very brief regarding some of the (measurability) assumptions and definitions underlying the theorem, we give a *detailed elaboration* of these assumptions and definitions, and their soundness, in Section C in [Geiger et al., 2019]. For a *justification* of some of the theorem's assumptions see Section 7. Note that Setting 1 is formulated pretty *generally*: the slots K can be any set of options the users have. K can be time slots in one shared facility, like a road section; or K can be several facilities that provide the same service, say civil offices in a city; or it can be a combination, i.e., time slots in several facilities. (Our "slot" is similar to "facility", or, to some extent, "feasible combination of facilities", in congestion games.) The main limitation may be seen in the assumptions of inference-assistability and assistant-best-responding, saying that users can meaningfully evaluate the utility of their choices based on only Y, which A is a forecast for.

3.2 Existence step in small-scale setting

In Theorem 1, we characterized the type of solution (a BNE of game G) that is implemented by the assistant-based system (with significantly lower requirements on users' knowledge/inference capacities than in the game G), *if* the assistant reaches a self-fulfilling prophecy policy ("characterization step"). We established this result for the general setting (Setting 1). As second step towards Goal 1, it remains to understand when such a self-fulfilling prophecy *exists* ("existence step"). This second step we perform separately for two instructive subsettings of the general setting, which are each still reasonably general. As a warm-up exercise, we start in a setting where we can easily build on game-theoretic results – because it corresponds to a classical finite Bayesian game.

3.2.1 Introducing the setting

Setting 2 (***Small-scale setting***)**.** *As a restricted form of Setting 1, consider the following* small-scale setting*:* $I = \{1, \ldots, n\}$ *is finite and we interpret its elements here as* users *(not types), and the individual actions of the users are directly publicly observable, i.e.,* $Y = B$*, and* $h_i(B) = B_{-i}$ *in Eq. 1.* X, V, W *all have finite range.*

We may write M^{small} and G^{small} to denote assistant-based system (Definition 1) and benchmark game (Definition 2), respectively, canonically associated to this particular small-scale setting.

Corollary 1. *Setting 2 is a special case of Setting 1. In particular, it satisfies the conditions of Theorem 1 and hence the theorem's implications hold for* $M = M^{small}$ *and* $G = G^{small}$.

3.2.2 Self-fulfilling prophecy existence

To understand the conditions under which a self-fulfilling prophecy exists, based on the second part of Theorem 1 (or rather: Corollary 1) it is enough to understand when a strict BNE of G exists. But in the current small-scale setting, $G = G^{\text{small}}$ is the classical finite (Bayesian) game, which is well understood. For example, Harsanyi [1973, Theorems 3, 4] showed that when assuming that the players' utilities are contaminated by a small additive noise, then there exists a strict equilibrium with probability one. Furthermore, Bilancini and Boncinelli [2016] establish conditions under which all BNE are (essentially) strict, which entails existence of such strict BNE when combined with general BNE existence results.

3.3 Existence step in large-scale setting

While the above small-scale setting is easy to understand, it has significant limitations: first, the users' actions B have to be fully observable for the (loss L^{pred} of the) assistant, which is often impossible due to data privacy regulations; and second, there has to be a fixed set of unique users, while in practice the set of users may change of course. Therefore we perform the second step towards Goal 1 also for the following large-scale setting (again a subsetting of the general setting of Section 2, different from the small-scale setting). It corresponds to nonatomic games [Schmeidler, 1973], and is mathematically more involved, but abstracts away from individual users and in particular only requires a cross-user aggregate of actions to be publicly observed.

3.3.1 Introducing the setting

Setting 3 (***Large-scale setting***)**.** *As a restricted form of Setting 1, consider the following (aggregated)* large-scale setting*:*

- *For simplicity, we assume there are only two slots,* $K = \{0, 1\}$*, and that* V, W *are constant.*[9] *We assume* $I = [0, 1]$ *with the Borel sets as* σ*-algebra* $\mathcal{I}$*, and interpret* $i \in I$ *as a* type *of user with a certain form of utility function and private signal (similar as Kim and Yannelis [1997]). Let* range_B *(i.e., the set of possible joint user actions* $(B_i)_{\in I}$*) be the set of* $\{0, 1\}$*-valued Lebesgue-measurable functions on* I*.*
- *Let* $\text{range}_{Y_1} = [0, 1]$ *and* $Y_1 := \int B_i r(i|X) di$*, for* $(r(\cdot|x))_{x \in \text{range}_X}$ *a family of continuous (Lebesgue) densities on* $(I, \mathcal{I})$*, continuous also in* x*. And let* $Y_0 := 1 - Y_1$*. The interpretation is that* Y_1 *is the fraction of users that choose slot* 1*, i.e., a (stochastic) aggregate of*

[9] We will prove the main results, Theorem 2, for an *arbitrary number* $|K|$ *of slots* though. The extension to stochastic V, W is less obvious due to measure-theoretic issues.

B, and Y_0 is the remaining amount of users, that choose slot 0. Since $Y = (Y_0, Y_1)$ is fully parameterized by Y_1, from now on we consider Y to be 1-dimensional and stand for Y_1.

• *Regarding users $i \in I$ and utilities, let h_i (Eq. 1) be the identity, and let $(i, y) \mapsto \tilde{U}_i(k, y)$ be a polynomial in i, y, for all $k \in K$ (we dropped W_i, h_i from the general $\tilde{U}_i$, Eq. 1). This means, in particular, that the utilities only depend on the* amount *of users at the various slots, not on their* identities. *For any $k \neq l \in K$, let $\tilde{U}_i(k, y) - \tilde{U}_i(l, y) = \sum_m i^m q_m(y)$ be such that, for at least one $m \geq 1$, $q_m(y)$ is nonzero and constant in y.*

Note that, while in practice of course the set of (simultaneous) users and thus also (simultaneous) types of users is finite, having $I = [0, 1]$ can be seen as an approximation with nice theoretical properties to real settings with *many* users. We may write M^{large} and G^{large} to denote assistant-based system (Definition 1) and benchmark game (Definition 2), respectively, for this particular large-scale setting.[10] G^{large} can be seen as an incomplete-information *nonatomic game*, related to [Kim and Yannelis, 1997] but different in that our state can have uncountable range, see also Section B in [Geiger et al., 2019].[11] For the sake of completeness, let us formally state a version of Theorem 1 for this setting, proved in Section D.2 in [Geiger et al., 2019].

Corollary 2. *Setting 3 is a special case of Setting 1. In particular, it satisfies the conditions of Theorem 1 and hence the theorem's implications hold for $M = M^{large}$ and $G = G^{large}$.*

3.3.2 Self-fulfilling prophecy existence

In contrast to the small-scale setting, for the large-scale setting and the corresponding benchmark game G^{large} there is less established work that helps to understand existence of a self-fulfilling prophecy policy. Intuitively, a key question in this large-scale setting is: can a forecast that only forecasts an *aggregate* of the users' actions (the Y of Setting 3) actually be a self-fulfilling prophecy and thus help for coordination? For instance, as observed by Mareček et al. [2016], if the population of users is completely homogeneous, they will all respond in the same way upon receiving the same input, making coordination difficult. Here is our answer for this question – the second of our two main theoretical results.

[10] In the assistant-based system M^{large}, let range_A be the set of Borel measures on $[0, 1]$, since A is a probabilistic forecast for Y $(= Y_1)$.

[11] The distribution over the types (which is not to be interpreted as a probability – rather as one actual realization) is *random*, turning it into an incomplete-information setting. The name "nonatomic" comes from the fact that one considers *nonatomic measures* on the type space $I = [0, 1]$.

Theorem 2 (Large-Scale Self-Fulfilling Prophecy Existence)**.** *There exists a self-fulfilling prophecy policy π in the assistant-based system M^{large} (in Setting 3).*

This implies, based on Corollary 2:

Corollary 3 (Large-Scale Bayesian Nash Equilibrium Existence)**.** *The benchmark game G^{large} (for Setting 3) has a Bayesian Nash equilibrium (BNE).*

Proof idea and interpretation: The proof of Theorem 2, which is given in Section D.3 in [Geiger et al., 2019] for an arbitrary number of slots K, is based on the Leray-Schauder-Tychonoff fixed point theorem, harnessing the compactness of the set of Borel measures, range_A, under a weak topology. The most important implication of the theorem is that $\min_\pi L_\pi^{\text{pred}} = 0$. And therefore, together with the first step in the form of Theorem 1, it shows that an assistant that only forecasts an aggregate can nonetheless, when it achieves its optimum, help "solve" the coordination problem – select a BNE. The intuition behind the assumptions is that types and their utility functions have to be *diverse*. Corollary 3 can be seen as standalone, purely game-theoretic result for G^{large}.

3.3.3 An instructive linear special case

Let us consider a simple special case of the large-scale setting (which is not central to understand the rest of the paper and can be skipped). On the one hand, this helps to get an intuition for Theorem 2, on the other hand this will justify assumptions we will make in the analysis of our algorithm in Section 5.1. Assume the utility $\tilde{U}_i(k, y)$ of Setting 3 is *linear* in i, y for all $k \in K$ (making the users "risk-neutral"). So $\tilde{U}_i(1, y) - \tilde{U}_i(0, y) = i + \varphi y + \chi$, for $\varphi, \chi \in \mathbb{R}$. Let $r(i|x) := \frac{1}{2\delta}[x - \delta \leq i \leq x + \delta], i \in I, x \in \text{range}_X$, with $[\cdot]$ the Iverson bracket (i.e., density of the uniform on $[x - \delta, x + \delta]$), and let P_X be the uniform on $[\delta, 1 - \delta]$. Then the value of Y as a function of $A = a, X = x$ is, for H the Heaviside function, given by

$$\int H\left(\int \tilde{U}_i(1, y) - \tilde{U}_i(0, y) da(y)\right) r(i|x) di \quad (7)$$

$$= \int H\left(i + \varphi \mathbb{E}_{Y' \sim a}(Y') + \chi\right) r(i|x) di \quad (8)$$

$$= \frac{\varphi}{2\delta} \mathbb{E}_{Y' \sim a}(Y') + \frac{1}{2\delta} x + \frac{\delta + \chi}{2\delta}, \quad (9)$$

for $x - \delta \leq -\varphi \mathbb{E}_{Y' \sim a}(Y') - \chi \leq x + \delta$, and 0 or 1, respectively, otherwise – a *piece-wise linear function* in $\mathbb{E}_{Y' \sim a}(Y'), x$.

First, this shows that under the mentioned assumptions, Y (and its distribution) only depends on the mean $\mathbb{E}_{Y' \sim a}(Y')$, but no other properties of a. In particular, $L_\pi^{\text{pred}} = 0$ iff $L_{\pi'}^{\text{point}} = 0$, for π an appropriate probabilistic extension of π', and

$$L^{\text{point}}_{\pi'} := \mathbb{E}_{M,\pi'}\left(\|A - \mathbb{E}(Y|V)\|_2^2\right) \quad (10)$$

a ***point prediction version*** of the probabilistic ***prediction accuracy*** loss L^{pred}_{π}.[12] This justifies for the assistant to provide *point forecasts* under the above assumptions. Second, this justifies a (locally) linear model for Y in $(\mathbb{E}_{Y'\sim a}(Y'), x)$ and noise. Note that Theorem 2 restricted to this simple linear case is immediate based on the intuitive fact that a generic linear function has a fixed point.

4 SETTING FOR ALGORITHM PART – CONTROL DYNAMICS

To prepare the algorithm part of the paper, let us extend the general one-stage setting (Setting 1) and the assistant-based one-stage system (Definition 1) to a ***general dynamic setting*** and an ***assistant-based dynamic system*** M^{dyn}, respectively, in the following "natural" way. This directly implies also dynamic extensions of small-scale and large-scale setting (Settings 2 and 3) and the corresponding assistant-based systems (we do not introduce explicit symbols for them though).

The dynamic extensions consists of $\mathbb{N}$ copies of the one-stage versions, called *stages/repetitions*. We denote variables, say A, in the t-th repetition by A^t, $t \in \mathbb{N}$. Furthermore, the dynamic extensions contains the following equations that replace/extend the ones of repetition t – think of it as a form of feedback control model, a partially observable Markov decision process (POMDP) [Sutton and Barto, 1998] (from the perspective of the assistant):

$$X^t = \bar{X}(X^{t-1}, E^t) \quad (11)$$

$$A^t = \pi(V^{0:t}, A^{0:t-1}, Y^{0:t-1}), \quad (12)$$

with E^t independent stochastic error terms, (measurable) function $\bar{X}$, and (measurable) *dynamic assistant policy* π.[13] The gray, dashed arrows of Figure 2 indicates this dynamic extension. Regarding the assistant's objectives, let $L^{t,\text{pred}}, L^{t,\text{point}}$ be defined similarly as $L^{\text{pred}}, L^{\text{point}}$, but additionally conditioning on the observed past:

$$L^{t,\text{pred}}_{\pi} = \mathbb{E}_{M,\pi}\left(d\left(P_{M,\pi}\left(Y^t | A^{0:t-1}, V^{0:t}, Y^{0:t-1}\right), A^t\right)\right), \quad (13)$$

[12] The reason why we take this definition of L^{point}_{π} instead of, say, some form of "$\mathbb{E}((A^t - Y^t)^2|V)$", is because the (distribution of) Y depends on A. And it may happen that the latter quantity, which is some form of "variance" that depends on the distribution of Y, is lower for a non-fixed point A than for a fixed point A, which would hurt the relation to equilibrium selection. A similar reason underlies our definition of L^{pred}_{π}.

[13] $A^{0:t-1}$ means $A^0, \dots, A^{t-1}$; similarly for other variables.

for all π, and similarly $L^{t,\text{point}}_{\pi}$. Remember: *stage t* must not be confused with *(time) slot k within* one stage. To motivate the algorithmic part below, let us give two examples of naive dynamic assistant policies that fail.

Example 1 (Naive assistant yields oscillation)**.** *Consider a toy scenario of two users, $i = 1, 2$, two slots, $K = \{0, 1\}$, W, V, X constant, and $(0, 1)$ and $(1, 0)$ the (pure) Nash equilibria of the induced complete-information benchmark game. For simplicity, let B be directly observed ($Y = B$), let A be a point forecast. As usual, assume each day t both users best-respond to A^t. The assistant starts with, say, $A^0 = (0, 0)$ and then, naively, each day takes yesterday's outcome B^{t-1} as forecast for today, A^t. It is easy to see that this will lead to an* overshooting and oscillating *system $B^0 = (1, 1), B^1 = (0, 0), B^2 = (1, 1), \dots$ (called* flapping *by Mareček et al. [2015]).*

Example 2 ("I.i.d." assistant is sub-optimal)**.** *Classical forecasting applied to the sequence $B^1, B^2, \dots$ from Example 1 would yield the empirical distribution $P(B = b) = \frac{1}{2}(\delta_{(0,0),b} + \delta_{(1,1),b})$, with δ the Dirac delta, as optimal probabilistic forecast A^t – under some stationarity assumption. But the actual best forecast would be a Dirac delta on one of the two Nash equilibria $(0, 1)$ and $(1, 0)$ (Theorem 1; we ignore mixed equilibria here).*

5 PREDICTIVE ASSISTANT ALGORITHMS WITH GUARANTEES

In the first part, we analyzed conditions under which predictive assistants help coordination (in terms of the equilibrium selection objective, Section 2.3), *if* they manage to optimize prediction accuracy, leaving open the *"how"*. Therefore, as second part of the paper, we address:

Goal 2. *Design algorithms for the assistant policy π in the dynamic assistant-based system M^{dyn} that optimize prediction accuracy $L^{t,pred}$ (and asymptotically select an equilibrium, if possible), learning from past interactions.*

We will consider dynamic versions of the two settings for which we established in Section 3 that predictions can help coordination: large-scale setting (in Section 5.1) and small-scale setting (in Section 5.2). For each setting, we propose an assistant algorithm π, and provide a theoretical analysis of its dynamics/convergence. A unifying idea behind both algorithms is that they mitigate certain bad user behavior, e.g., "overshooting" due to too many users jumping to the same purportedly "good" slot, helping convergence to a Nash equilibrium (of the *stage* benchmark game). Recall that *users' utilities (functions) are hidden* from the assistant (Definition 1), so the assistant's inference (about the equilibrium) is mainly based on behavioral data of how users react to forecasts.

Algorithm 1: Expodamp (large-scale setting)

Input: parameter: α
for *each stage* $t \geq 1$ **do**
  Input: A^{t-1}, Y^{t-1}
  Output: $A^t := A^{t-1} + \alpha(Y^{t-1} - A^{t-1})$

5.1 Expodamp for large-scale setting

Consider the dynamic large-scale setting[14] (Section 4) and let A be a *point forecast* for Y, i.e., $\mathrm{range}_A = \mathrm{range}_Y$, and consider $L_\pi^{t,\text{point}}$ as loss (dynamic version of Eq. 10, as described in Section 4). Recall that in Section 3.3.3 we gave conditions that justify this point prediction approach.

We propose *Expodamp* as described in Algorithm 1 as the assistant's dynamic policy π. The intuition behind Expodamp is that this formula can dampen oscillations due to "overshooting" user behavior (Example 1) but it can also accommodate for non-stationarities in user preferences. These intuitions will be made rigorous in the proposition below.[15]

Assumption 1. *Let the following equations hold for the dynamic assistant-based system* M^{dyn}*,* $t \geq 1$*:*

$$X^t = X^{t-1} + E_X^t, \tag{14}$$

$$Y^t = \beta A^t + \gamma X^t + E_Y^t, \tag{15}$$

with E_X^t, E_Y^t *noise terms that are independent of the past and each other. (This is a* state-space model *known from the* Kalman filter *[Lütkepohl, 2006].)*

Recall that in Section 3.3.3 we gave conditions, in the large-scale setting, that justify the linearity in Assumption 1 (note that the X in Assumption 1 would correspond to a parameter of the distribution of X rather than to X itself in Section 3.3.3, but we neglect this detail for simplicity of notation). Also note that Assumption 1 is a linear approximation which facilitates the theoretical analysis but comes at the cost of a mismatch to the actual setting: $(Y^t)_{t\in\mathbb{N}}$ in Assumption 1 can leave $[0,1]$ in the long run, so the model should rather be seen as a local approximation. Note that, due to convexity, Expodamp will always output $A^t \in [0,1]$ upon $Y^{t-j} \in [0,1], j \geq 1$ though. Keep in mind that the fixed point (self-fulfilling prophecy) of the linear function $a \mapsto \beta a + \gamma x$ (ignoring the noise term) is $\gamma(1-\beta)^{-1}x$ (exists whenever $\beta \neq 1$). In particular, if $\beta = (1-\gamma)$, then the fixed point (corresponding to the self-fulfilling prophecy/BNE) is x. We can give the following guarantees, for which we prove a generalization[16] in Section E.1 in [Geiger et al., 2019].

Proposition 1 (Optimality and Convergence Rate of Expodamp)**.** *In the dynamic large-scale setting (Section 4), let Assumption 1 hold true. Let the assistant's policy* π *be Expodamp (Algorithm 1).*

- Stochastic case: *In Expodamp, let* $\alpha := (1-\beta)^{-1}$ *for the true* β *of Eq. 15. Assume* $E_Y^t = 0, t \geq 1$*. Then, at each stage t,* $L^{t,point} = 0$ *and*

$$A^t = \arg\min_{a'} \mathbb{E}\left(\|A^t - Y^t\|_2^2 \mid A^t = a', A^{0:t-1}, Y^{0:t-1}\right). \tag{16}$$

- Deterministic case: *Assume that* $X^t = x$ *is constant, that* $\beta = (1-\gamma)$ *and that* $E_X^t = E_Y^t = 0$*. Then*

$$Y^t = x + (1-\gamma)(A^0 - x)(1-\alpha\gamma)^t, \text{ for all } t \geq 0.$$

That is, Y^t *converges exponentially with rate* $\gamma\alpha$ *towards the "optimum"/fixed point* x *(and thus also* A^t *converges to* x *based on Expodamp's formula) if* $0 < \gamma\alpha < 2$*.*

When applying Algorithm 1 in practice, often one does not know the parameter α a priori and has to infer it. As a first approximation, it may be learned by naively fitting Algorithm 1 to past observational data as if it were a classical (non-influential) forecasting method [Hyndman et al., 2008]. In principle however, without going into detail, α rather has to be learned like a *control policy*, based on how the environment responds to it.

5.2 Partpred for small-scale setting

While Expodamp is the main algorithm of this paper, here we also provide a proof-of-concept algorithm for the repeated *small-scale* setting (Section 4). Assume X^t to be independent of $X^{1:t-1}$, i.e., the special case where the $X^t, t \in N$ are i.i.d. The algorithm, *Partpred*, is sketched – for the case that V is constant – in Algorithm 2, and fully described in Section E.2 in [Geiger et al., 2019].

The basic idea is as follows: as long as there is (significant) uncertainty about where the optimum (self-fulfilling prophecy/BNE) would be, the algorithm tries to make a prediction that is at least partially correct (i.e., makes the correct prediction at least w.r.t. the behavior of one player). The algorithm combines ideas from *best-response dynamics* and *congestion games* [Roughgarden, 2016] with random exploration whenever the best-response dynamics would cycle. Let $\bar{A}$ be the (finite)

[14] In particular, Y is considered 1-dimensional (since Y_1 determines Y_0). The extension to more slots is straight forward.

[15] The formula in Algorithm 1 is a case of a so-called exponential smoothing method [Hyndman et al., 2008]. However, so far (to the best of our knowledge) it has only been applied to classical forecasts that do not influence the outcome. In a sense, we generalize the established method to this new setting.

[16] It is formulated slightly cleaner, using the do-operator.

Algorithm 2: Partpred (small-scale; sketch)

1 **Input:** parameters: $\bar{A}$, r; initialization: $a \in \bar{A}$
2 For r steps, output $A = a$ and sample B. Let $\hat{P}_a^r$ be the resulting empirical distribution of B.
3 Let $a' := \arg\min_{a'' \in \bar{A}} \|a'' - \hat{P}_a^r\|$. Let a'' be obtained by only taking over a subset of best responses from a and a' such that a'' is a correct forecast at least w.r.t. some users
4 **if** $a'' = a$ **then**
5 | Keep outputting a forever
6 **else if** *a'' and all other $a' \in \bar{A}$ have been tried r times* **then**
7 | Set $a'' := \arg\min_{a'} \|a' - \hat{P}_{a'}^r\|$
8 | Keep outputting a'' forever
9 **else if** *a'' has been tried r times* **then**
10 | Pick unused $a'' \in \bar{A}$ at random
11 Set $a = a''$ and jump to line 2

set of all distributions $P_{G,s}(B)$ that arise from (deterministic) strategy profiles s of G^{small}. For simplicity, we assume $\bar{A}$ to be given, but in a next step this could be inferred as well. We give the following guarantee, sketched for V constant, whose general version is proved in Section E.2 in [Geiger et al., 2019].

Proposition 2 (Convergence of Algorithm 2 (Sketch))**.** *In the setting described above, assume G^{small} has a strict BNE. Let the assistant's policies $\pi_r, r \in \mathbb{N}$ be given by Algorithm 2, with parameter $\bar{A}$ as defined above. Then, for any $\varepsilon > 0$, there exists R, T such that for all $r > R, t > T$, it holds that $P(L_{\pi_r}^{t,pred} = 0) > 1 - \varepsilon$ and $P(s_{\pi_r}$ is a BNE of $G^{small}) > 1 - \varepsilon$.*

6 EXPERIMENT

Here we empirically evaluate Expodamp (Algorithm 1 for the large-scale setting) and a baseline.

Experimental setup: We conducted our experiment in a real-world congested campus cafeteria with around 400 users per day. Here, observation $[Y^t]_k$ is (a proxy to) the number of people in the queue at time k of day t.[17] The coordination assistant in this experiment is a web app which provides the daily forecast (i.e., the forecast is updated once per day, in the morning – more dynamic versions are future work) to the cafeteria users, to inform their decisions in terms of when to go to the cafeteria. The web app is used by between 15 and 45 users per day but may influence more (slightly deviating from our model). Besides Expodamp (with parameter α tuned based on a previous observational sample), we evaluate the baseline method *Average* defined by

$$a^{t+1} := \frac{1}{t} \sum_{s=1}^{t} y^s, t \geq 2 \qquad (17)$$

(i.e., treating $y^{1:t}$ as purely observational i.i.d. sample). Expodamp and Average are run as the policy that generates the forecast (which is then provided via the web app to the users of the cafeteria), each for a period of $T = 35$ days. See Figure 3 for an illustration of the experimental protocol. As metric, we use the mean squared error[18]

$$\tilde{L}^{t,\text{point}} := \frac{1}{T} \sum_{t=1}^{T} \|a^t - y^t\|_2^2. \qquad (18)$$

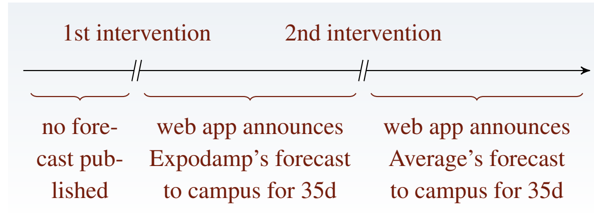

Figure 3: Protocol of our real-world interventional experiment in a large campus cafeteria; steps along Y-axis.

Method	$\tilde{L}^{t,\text{point}}$ (MSE; Eq. 18)
Expodamp (Algorithm 1)	69.56
Average (baseline; Eq. 17)	74.25

Table 1: Evaluation shows that Expodamp has higher prediction accuracy.

The outcome is in Table 1, showing that Expodamp outperforms Average in this experiment. For illustration, we also show a sample of Expodamp's output A^t and actual outcome Y^t, for one day t, in Figure 1.

7 REMARKS AND FURTHER RELATED WORK

This section discusses additional aspects of the main results and further related work.

Why prediction accuracy / equilibrium selection as objective. Alternative to our approach in this paper, one

[17] While our general considerations allow Y to be queue length, in our large-scale setting the components of Y are the slots, of which queue length is rather something like an integral.

[18] We use $\tilde{L}^{t,\text{point}}$ as a sample-level proxy for the population-level loss L^{point} of Eq. 10. We conjecture that, under appropriate assumptions related to Assumption 1, it can be shown that the policy that is optimal under the former loss converges (say, in probability) to a policy which is also optimal under the latter loss. The argument may build on the equivalence between $L^{t,\text{point}} = 0$ and Eq. 16 in Proposition 1.

could start from some (somehow legitimized) *social welfare* [Nisan et al., 2007] as a function of users' preferences, and design assistants that try to optimize it. This would be somewhat more in line with the economic notion of optimizing efficiency. Here, we rather follow a heuristic approach of starting with the "natural" prediction accuracy objective, because it compares well to the benchmark of equilibrium selection (Theorem 1), and for the following reasons: First, prediction accuracy can be directly measured, while social welfare seems hard to infer/identify from the incomplete information contained in the behavioral data available in our setting. Second, it is easy to interpret for users and leads to a form of "incentive compatibility" of users' assistant-best-response (see remark below). Third, we feel that in our coordinative setting, equilibrium outcomes can be quite efficient in terms of social welfare. Generally, social welfare functions of course are hard to pick and impose in the first place. Nonetheless, equilibrium outcomes can of course be significantly inefficient, which has extensively been studied under the name of *price of anarchy* [Roughgarden, 2005, Nisan et al., 2007]. But even in this regard, Theorem 1 can be helpful in that it makes predictive assistant-based settings amenable to such studies.

Remarks on our model assumptions: To justify our assumption of users "blindly" best-responding to the assistant's forecast (Definition 1) observe that it can be seen as consistent with (instrumental) rationality[19] in the following sense: if only considering the *asymptotic* utility (once the assistant converged), then deviating from this behavior means deviating from a BNE, based on Theorem 1.[20] Furthermore, all users best-responding simultaneously can sometimes be a too strong assumption, but we feel that it is *a* situation that can happen (more or less) at least *sometimes*, and therefore is worth analyzing. This being said, the assistant-best-responding assumption should be seen as a pragmatic *first step* that can be refined in future work. Generally, Theorem 1 shows that assistant-based systems can achieve coordination comparable to the benchmark game (additionally, it serves as a mechanism for equilibrium *selection* if there are several) – *but at a significantly lower cost*, since the inference task is *centrally* done by the assistant. (Obviously, it is only cheaper when inference comes at a cost – otherwise raw data V could simply be provided to users directly.)

Further general related work: Let us mention that for the various versions of assistants we mentioned in Section 1 that are publicly available [Google, 2019, DB, 2019, ASFA, 2019], we could not find out what algorithms or theory they rely on.[21] Research-wise, in *mechanism design*[22], a related direction has been emerging that studies how to design the *information structure* [Taneva, 2015, Bergemann and Morris, 2017] instead of the allocation/payment structure. Furthermore, *data-driven approaches to mechanism design* have gained momentum [Balcan et al., 2016, Duetting et al., 2019, Tang, 2017, Kearns et al., 2014]. But these lines of research differ from ours – often additionally to what we already mentioned in Section 1 (bounded rationality of our users and limited power of our mechanism) as follows: either they assume that agents input their (true, if "incentive compatible") preferences explicitly (instead of behavioral data), or they neglect, to some extent, agent's actual preferences (which can be appropriate for revenue maximization of course).

8 CONCLUSIONS

In this work, we studied when and how parts of the coordination process of users of shared facilities can be "outsourced" to a central data-driven predictive assistant. Our theoretical analysis showed that such assistants can help solve the coordination problem in a game-theoretic sense, but non-trivial conditions have to be met: in terms of the information and preference structure of users, and stochasticity of their preferences in case only large-scale aggregated information is available to the assistant. Based on this analysis, we proposed two machine learning coordination assistant algorithms on behavioral data. We used linear dynamical systems models to prove their optimality/convergence, accounting for the fact that there is a feedback loop from predictions to outcomes. And we conducted a large-scale interventional experiment in a real campus cafeteria that provided empirical hints for the validity of our main algorithm.

Generally, the mentioned related work and our work indicate that there is a plethora of possible computational mechanisms for collective decision making, in terms of inputs (high-level information, behavioral data, and beyond) and influences (full control over the outcome, money incentives, pure information/predictions, and beyond), many of which may still be unexplored.

Acknowledgments. The authors thank Carl-Johann Simon-Gabriel, Jonathan Williams and Sebastian Stark for insightful discussions and engineering support.

[19] In this work, we adopt the game-theoretic view of humans in social situations as "selfish" agents maximizing exogenously given individual utility functions. We feel this is appropriate for our simple setting of facility use. But overall, decision making in social systems has many more aspects of course.

[20] Nisan et al. [2011] studied rationality of best-response dyn.

[21] Also note that some of them do not explicitly call the service an "assistant" or a "forecast".

[22] The analogy between our assistant and a mechanism is that they are both "institutions" added to the set of agents to solve some collective decision making problem.

References

ASFA. France motorways: traffic forecast on french motorways - ASFA. https://www.autoroutes.fr/en/traffic-forecast.htm, 2019. Accessed: 2019-01-25.

M.-F. Balcan, T. Sandholm, and E. Vitercik. Sample complexity of automated mechanism design. In *Advances in Neural Information Processing Systems*, pages 2083–2091, 2016.

D. Bergemann and S. Morris. Information design: A unified perspective. 2017.

E. Bilancini and L. Boncinelli. Strict Nash equilibria in non-atomic games with strict single crossing in players (or types) and actions. *Economic Theory Bulletin*, 4(1):95–109, 2016.

DB. DB BAHN – TravelService – Query page. https://reiseauskunft.bahn.de/bin/query.exe/en, 2019. Specifically, we refer to the "Demand" symbol that occurs for some train connections (which may be informed by bookings, besides past behavioral data). Accessed: 2019-06-04.

P. Duetting, Z. Feng, H. Narasimhan, D. C. Parkes, and S. S. Ravindranath. Optimal auctions through deep learning. In *36th International Conference on Machine Learning (ICML)*, 2019.

P. Geiger, M. Besserve, J. Winkelmann, C. Proissl, and B. Schölkopf. Coordinating users of shared facilities via data-driven predictive assistants and game theory. *ArXiv e-prints*, 2019. URL https://arxiv.org/abs/1803.06247. arXiv:1803.06247 [cs.GT]. This is the extended version of the present paper, including supplement.

Google. Google Maps. https://www.google.com/maps, 2019. Specifically, we refer to the "Popular times" bar diagram that is often displayed for facilities like swimming pools, train stations, etc. Accessed: 2019-06-04.

J. C. Harsanyi. Games with randomly disturbed payoffs: A new rationale for mixed-strategy equilibrium points. *International journal of game theory*, 2(1):1–23, 1973.

R. Hyndman, A. B. Koehler, J. K. Ord, and R. D. Snyder. *Forecasting with exponential smoothing: the state space approach*. Springer Science & Business Media, 2008.

M. Kearns, M. Pai, A. Roth, and J. Ullman. Mechanism design in large games: Incentives and privacy. In *Proceedings of the 5th conference on Innovations in theoretical computer science*, pages 403–410. ACM, 2014.

T. Kim and N. C. Yannelis. Existence of equilibrium in Bayesian games with infinitely many players. *Journal of economic theory*, 77(2):330–353, 1997.

H. Lütkepohl. *New Introduction to Multiple Time Series Analysis*. Springer, Berlin, Heidelberg, New York, oxford statistical science series edition, 2006.

J. Mareček, R. Shorten, and J. Y. Yu. Signalling and obfuscation for congestion control. *International Journal of Control*, 88(10):2086–2096, 2015.

J. Mareček, R. Shorten, and J. Y. Yu. r-extreme signalling for congestion control. *International Journal of Control*, 89(10):1972–1984, 2016.

N. Nisan, T. Roughgarden, E. Tardos, and V. V. Vazirani. *Algorithmic game theory*, volume 1. Cambridge University Press Cambridge, 2007.

N. Nisan, M. Schapira, G. Valiant, and A. Zohar. Best-Response Mechanisms. In *ICS*, pages 155–165, 2011.

J. Pearl. *Causality*. Cambridge University Press, 2000.

T. Roughgarden. *Selfish routing and the price of anarchy*, volume 174. MIT press Cambridge, 2005.

T. Roughgarden. *Twenty lectures on algorithmic game theory*. Cambridge University Press, 2016.

D. Schmeidler. Equilibrium points of nonatomic games. *Journal of statistical Physics*, 7(4):295–300, 1973.

Y. Shoham and K. Leyton-Brown. *Multiagent systems: Algorithmic, game-theoretic, and logical foundations*. Cambridge University Press, 2008.

R. S. Sutton and A. G. Barto. *Reinforcement learning: An introduction*. MIT press, 1998.

I. A. Taneva. Information design. 2015.

P. Tang. Reinforcement mechanism design. In *IJCAI*, volume 17, pages 26–30, 2017.