Computational Management Science (2022) 19:627–664
https://doi.org/10.1007/s10287-022-00429-9

ORIGINAL PAPER

Network manipulation algorithm based on inexact alternating minimization

David Müller[1] · **Vladimir Shikhman**[1]

Received: 1 August 2021 / Accepted: 13 June 2022 / Published online: 15 July 2022

Abstract

In this paper, we present a network manipulation algorithm based on an alternating minimization scheme from Nesterov (Soft Comput 1–12, 2020). In our context, the alternative process mimics the natural behavior of agents and organizations operating on a network. By selecting starting distributions, the organizations determine the short-term dynamics of the network. While choosing an organization in accordance with their manipulation goals, agents are prone to errors. This rational inattentive behavior leads to discrete choice probabilities. We extend the analysis of our algorithm to the inexact case, where the corresponding subproblems can only be solved with numerical inaccuracies. The parameters reflecting the imperfect behavior of agents and the credibility of organizations, as well as the condition number of the network transition matrix have a significant impact on the convergence of our algorithm. Namely, they turn out not only to improve the rate of convergence, but also to reduce the accumulated errors. From the mathematical perspective, this is due to the induced strong convexity of an appropriate potential function.

Keywords Network manipulation · Inexact alternating minimization · Discrete choice · Strong convexity

Mathematics Subject Classification 90C25 · 90C35 · 90C90

✉ David Müller
david.mueller@mathematik.tu-chemnitz.de

Vladimir Shikhman
vladimir.shikhman@mathematik.tu-chemnitz.de

[1] Department of Mathematics, Chemnitz University of Technology, Reichenhainer Str. 41, 09126 Chemnitz, Germany

1 Introduction

Networks naturally occur in many areas such as economics, computer science, chemistry or biology. A common way to model scenarios within networks is to use Markov chains. For a finite state space, a transition matrix describes the structure of changes on the chain's states, see e.g. Gagniuc (2017). Usually, the iterative process of repeated transitions over the states provides a stationary distribution. However, if the considered time horizon is short, a question arises on how to efficiently manipulate the distribution of information within the network. As an obvious choice for manipulation, each agent may start with an initial distribution and spread the information by communicating with the neighbors. But, since the own manipulation power is often limited, it is quite reasonable for an agent to engage intermediary organizations instead. This could be due to the restricted access to the parts of the network, revealing the manipulation interests too apparently, or due to the lack of knowledge about the network structure. Possible examples include:

Influencers Companies, who want to credibly advertise their products or services via social media channels, pay influencers on social media platforms. They act as a part of the network and spread information on the products or services.

Search engines Web site owners try to increase the visibility of their web sites. In order to find proper content, most of the Internet users enter a query into a web search engine. Thus, these query results strongly influence the short term behavior of the users.

Conspiracy theory Agents try to spread false information for different interests. In order to spread fake news, agents have to rely on different distribution channels, such as groups in social media networks or Internet blogs.

Note that two of our examples are related to the manipulation of information in a social network. There is a growing literature concerning this topic, see Acemoglu and Ozdaglar (2011) for an overview. Most of the models describe the update of opinions or beliefs, see e. g. DeGroot (1974), which is done according to a convex combination of other network members' opinions. Applying traditional techniques from the analysis of Markov chains, the formation of a consensus is examined. In these approaches, manipulation is modeled by modifying the transition matrix, e. g. introducing randomness Acemoglu and Ozdaglar (2011). Förster et al. (2016) studied manipulation in a model of opinion formation. There, the weights of the transition matrices can be changed by agents, while all starting distributions are fixed. Our model differs from those in the existing literature, since we examine how the information regarding a topic is distributed among network participants through intermediaries. Loosely speaking, we analyze who knows how much and how this

information state can be efficiently manipulated by engaging intermediaries. As mentioned by Acemoglu and Ozdaglar (2011), one central component of opinion formation is how agents update their prior beliefs based on new information. In this paper, we also contribute to the opinion formation, because we investigate a way to manipulate the acquirement of information by employing the network of information sources. Our goal is not to learn the complete structure of the network, for which usually hidden Markov models are applied, see e. g. Yang et al. (1997). Instead, organizations should be able to select a starting distribution aiming to arrive at a certain information state after a number of iterations. Agents are choosing among the intermediary organizations to boost manipulation. This is at the core of our network manipulation algorithm. Note that a similar problem has been analyzed by Lindqvist (1977), where the author applies decision-theoretic techniques to observe a state at a time and obtain information about the initial state.

Let us comment on the mathematics behind the proposed network manipulation algorithm. It is motivated by Nesterov (2020) where a new technique for soft clustering is introduced. For this, voters and political parties alternately solve their subproblems, yielding an alternating minimization scheme. The behavior of voters turns out to be in accordance with the well known multinomial logit model from discrete choice theory. Namely, the voters choose rationally among the parties, but are prone to random errors, see e. g. Anderson et al. (1992). The parties update their political positions depending on how many voters they attract. Overall, the resulting soft clustering is given in terms of probabilities from the multinomial logit model. In this paper, we generalize the idea suggested in Nesterov (2020) to a broader class of discrete choice probabilities. This is done by presenting a network manipulation model based on alternating steps performed by agents and organizations. Agents try to manipulate a network by choosing intermediary organizations for helping in that. In order to select among the organizations, agents observe which of them better manipulate the network in comparison to agents' goals. While doing so, agents are prone to random errors, which lead to choice probabilities following certain discrete choice models examined in our previous paper Müller et al. (2021a). Altogether, we show how the alternating minimization scheme introduced by Nesterov (2020) can be applied for network manipulation. Additionally, we present an inexact version of the alternating minimization scheme. Inexactness is due to the fact that the subproblems of agents and/or organizations may not be solved exactly and may suffer from numerical inaccuracies. Overall, we conclude that the agents' imperfect behavior and organizations' conservatism in profit maximization reduce the accumulated errors.

Notation In this paper, we mainly focus on subspaces of $\mathbb{R}^n$ and $\mathbb{R}^{m\times n}$, where $\mathbb{R}^n$ is the space of n-dimensional column vectors

$$x = \left(x^{(1)}, \dots, x^{(n)}\right)^T,$$

and $\mathbb{R}^{m\times n}$ denotes the linear space of $(m \times n)$-matrices. We denote by $e_j \in \mathbb{R}^n$ the j-th coordinate vector of $\mathbb{R}^n$ and write e for the vector of an appropriate dimension whose components are equal to one. By $\mathbb{R}^n_+$ we denote the set of all vectors with nonnegative components and notation Δ_n is used for the standard simplex

$$\Delta_n = \left\{ x \in \mathbb{R}^n_+ \,:\, \sum_{i=1}^n x^{(i)} = 1 \right\}.$$

We use the norms for $x \in \mathbb{R}^n$:

$$\|x\|_2 = \left[\sum_{i=1}^n x^{(i)2} \right]^{\frac{1}{2}}, \quad \|x\|_1 = \sum_{i=1}^n \left| x^{(i)} \right|, \quad \|x\|_\infty = \max_{i=1,\ldots,n} \left| x^{(i)} \right|.$$

For $x, s \in \mathbb{R}^n$ we use the standard scalar product:

$$\langle x, s \rangle = \sum_{i=1}^n x^{(i)} s^{(i)}.$$

For matrices $A, B \in \mathbb{R}^{m \times n}$ the inner product is defined via the trace:

$$\langle A, B \rangle = \text{ Tr } (A^T \cdot B).$$

A function $F : Q \to \mathbb{R}$ is called β-strongly convex on a convex and closed set $Q \subset \mathbb{R}^n$ w.r.t. a norm $\| \cdot \|$ if for all $x, y \in Q$ and $\alpha \in [0, 1]$ it holds:

$$F(\alpha x + (1 - \alpha) y) \leq \alpha F(x) + (1 - \alpha) F(y) - \alpha(1 - \alpha) \cdot \frac{\beta}{2} \|x - y\|^2.$$

The positive constant β is called the convexity parameter of F. If $\beta = 0$, we call F convex. A function π is β-strongly concave if $-\pi$ is β-strongly convex. For a convex function $F : Q \to \mathbb{R}$ the set $\partial F(x)$ represents its subdifferential at $x \in Q$, i.e.

$$\partial F(x) = \{ g \in \mathbb{R}^n \,:\, F(y) \geq F(x) + \langle g, y - x \rangle \text{ for all } y \in Q \}.$$

Its convex conjugate is

$$F^*(s) = \sup_{x \in \mathbb{R}^n} \{ \langle x, s \rangle - F(x) \},$$

where $s \in R^n$ is a vector of dual variables. We denote by $\nabla F(x)$ the gradient of a differentiable function F at x.

2 Manipulation model

Let us introduce our model in order to later construct a manipulation algorithm based on interaction within a network.

2.1 Interaction network

A central aspect in our model is a network with n nodes. The structure of this network describes how nodes interact among each other, e. g. how persons receive and exchange information. Thereby, a link from node j to i represents a connection. In the context of an information network, such a link would depict that person i acquires information from person j. We summarize the data in a transition matrix $M = \left(M_{ij}\right)_{i,j=1}^{n}$, where M_{ij} denotes the transition probability of node j to node i. Hence, the following holds:

$$\sum_{i=1}^{n} M_{ij} = 1 \quad \text{for} \quad j = 1, \dots, n.$$

M is a column stochastic matrix, i.e. $M \geq 0,\ e^T \cdot M = e^T$. Our model describes the process of information acquirement rather than the formation of opinions as e. g. in Förster et al. (2016). We are interested in a few periods of interactions, thus, we take the transition matrix as fixed. The interaction causes different states of the network, based on the connections of its nodes. The state of a network can be represented as an element of the standard simplex in $\mathbb{R}^n$ dependent on time variable. We call a vector $x(t) \in \Delta_n$ a state of a network at time t. Such a state reflects the value each node possesses after an interaction with other nodes. This could be for example the amount of information a person possesses in relation to the others or the market share of a company.

The dynamics of interaction can be described by an iterative process. Starting with a vector $x(0) \in \Delta_n$, the nodes interact repeatedly with each other. Thus, the iterative process is given by

$$x(t) = M \cdot x(t-1) = \dots = M^t \cdot x(0). \tag{1}$$

Obviously, all $x(t)$'s generated according to this process are elements of Δ_n. Our idea is closely related to the concept of network rankings, such as the famous *PageRank* from Page et al. (1999). However, we focus on a limited, mostly small number, of interaction periods. Within an information network, persons would typically exchange information for a few periods, before they make a decision. This short term behavior endows the starting vector $x(0) \in \Delta_n$ with importance. For the sake of brevity, we drop the time index by writing $x = x(0)$.

2.2 Agents

Let us assume that agents want to manipulate the resulting state of a network in favor of their own interests. Though they aspire certain network states, agents face some challenges by trying to manipulate a network. Often, they do not have knowledge of the network structure. Additionally, there are many situations, where the agents can't participate in the network because they cannot connect to a node without revealing their intentions, e. g. companies cannot credibly advertise their products by

themselves. There might also be networks, where an agent could interact in, but is restricted to start with a fixed vector. In particular, if the information is just spread uniformly. Instead, agents could instruct organizations to manipulate the interaction in order to reach an aspired state of the network. The organizations often have more information or at least experience about the structure of a network. In fact, they could even operate it. The agents choose among K organizations, where each organization provides an observable utility $u^{(k)}$. This discrete choice behavior we describe by means of the so-called additive random utility models. The additive decomposition of utility goes back to psychological experiments accomplished in the 1920's by Thurstone (1927). A formal description of this framework has been first introduced in an economic context by McFadden (1978), where rational decision-makers choose from a finite set of mutually exclusive alternatives $\{1, \dots, K\}$. Although the decision rule follows a rational behavior, agents are prone to random errors. These random errors describe decision-affecting features which cannot be observed. Each alternative $k = 1, \dots, K$ provides the utility

$$u^{(k)} + \epsilon^{(k)},$$

where $u^{(k)} \in \mathbb{R}$ is the deterministic utility part of the k-th alternative and $\epsilon^{(k)}$ is its stochastic error. We use the following notation for the vectors of deterministic utilities and of random utilities, respectively:

$$u = \left(u^{(1)}, \dots, u^{(K)}\right)^T, \quad \epsilon = \left(\epsilon^{(1)}, \dots, \epsilon^{(K)}\right)^T.$$

The probabilistic framework yields choice probabilities for each alternative:

$$p^{(k)} = \mathbb{P}\left(u^{(k)} + \epsilon^{(k)} = \max_{1 \le m \le K} u^{(m)} + \epsilon^{(m)}\right), \quad k = 1, \dots, K. \tag{2}$$

As the consumers behave rationally, their surplus is given by the expected maximum utility of their decision:

$$E(u) = \mathbb{E}_\epsilon\left(\max_{1 \le k \le K} u^{(k)} + \epsilon^{(k)}\right). \tag{3}$$

It is well known that the surplus function is convex, see e.g. Anderson et al. (1992). Additionally we make a standard assumption concerning the distribution of random errors.

Assumption 1 The random vector ϵ follows a joint distribution with zero mean that is absolutely continuous with respect to the Lebesgue measure and fully supported on $\mathbb{R}^K$.

We stress that the zero mean part of Assumption 1 is not restrictive and could be replaced by a finite mean assumption. By adding constants to the deterministic utilities u, it can be achieved that the random vector ϵ has zero mean, see e. g. Train (2009).

Let $g_{k,m}$ denote the density function of differences $\epsilon^{(m)} - \epsilon^{(k)}$, $k \neq m$ of random errors. Any point $\bar{z}_{k,m} \in \mathbb{R}$ which maximizes the density function $g_{k,m}$ is called a mode of the random variable $\epsilon^{(m)} - \epsilon^{(k)}$.

Assumption 2 The differences $\epsilon^{(k)} - \epsilon^{(m)}$ of random errors have finite modes for all $k \neq m$.

Assumption 1 guarantees that no ties in occur in (3), which provides differentiability of the surplus function. Further, the gradient of E corresponds to the vector of choice probabilities, which is known as the Williams-Daly-Zachary theorem, see e.g. McFadden (1978), i.e.

$$\frac{\partial E}{\partial u^{(k)}} = p^{(k)}, \quad k = 1, \dots, K. \tag{4}$$

Hence, each component of the gradient of E yields the probability that alternative k provides the maximum utility among all alternatives.

Another equivalent representation of choice probabilities can be obtained by means of the convex conjugate of the surplus function. Note that the convex conjugate of E is given by the function $E^* : \mathbb{R}^K \to \mathbb{R} \cup \{\infty\}$, defined by:

$$E^*(p) = \sup_{u \in \mathbb{R}^K} \{\langle p, u \rangle - E(u)\},$$

where $p = \left(p^{(1)}, \dots, p^{(K)}\right)^T$ is the vector of dual variables. In view of conjugate duality, the vector of choice probabilities can be derived from an optimization problem of rational inattention, see e. g. Fosgerau et al. (2020) and Müller et al. (2021a). Indeed, it has been shown that under Assumption 2 the vector of choice probabilities p is the unique solution of

$$\max_{p \in \Delta_K} \{\langle u, p \rangle - E^*(p)\}. \tag{5}$$

Now, we assume that there are N agents trying to manipulate the network. Each agent i has an aspired state of network which we denote by $v_i \in \Delta_n$. In order to reach the aspired state, agents can choose among K organizations. The k-th organization is able to manipulate the interaction dynamics in the network, which yields at time t a state of a network $x_k(t) \in \Delta_n$, $k = 1, \dots, K$. In general agents prefer organizations which provide a network state in line with the states they desire such as an aspired market shares distribution or state of information. In order to assess the outcome of a manipulation, any agent i has to compare K distances, i.e.

$$\|v_i - x_k(t)\|_2, \quad k = 1, \dots, K,$$

respectively

$$\|v_i - M^t \cdot x_k\|_2, \quad k = 1, \dots, K. \tag{6}$$

Note that (6) provides a way for agent i to observe the utility of choosing the k-th organization. The network state at time t is observable, so any agent is able to check,

if an organization has manipulated the network satisfactorily. Let us put all the states in a matrix, which yields a way to summarize all the states of a network at time t in one variable, i.e.

$$X(t) = \big(x_1(t), \dots, x_K(t)\big) \in \Delta_n^K.$$

The matrix above can also be expressed in terms of the starting vectors, by defining

$$X = \big(x_1, \dots, x_K\big) \in \Delta_n^K,$$

which enables us to write

$$X(t) = M^t \cdot X. \tag{7}$$

We define a vector valued function $g_i : \Delta_n^K \to \mathbb{R}_+^K$ for any agent i, which stores all these distances of the i-th agent and, hence, depends on a matrix X as input variable:

$$g_i(X) = \big(\|v_i - M^t \cdot x_1\|_2, \dots, \|v_i - M^t \cdot x_K\|_2\big)^T, \quad i = 1, \dots, N.$$

We write in matrix form:

$$G(X) = \big(g_1(X), \dots, g_N(X)\big) \in \mathbb{R}^{K \times N}.$$

In view of additive random utility models, $g_i(\cdot)$ provides a way to characterize the observable utility u_i by setting

$$u_i = -g_i(X), \quad i = 1, \dots, N.$$

Hence, the vector of the i-th agent choice probabilities has entries

$$p_i^{(k)}(X) = \mathbb{P}\left(-g_i^{(k)}(X) + \epsilon_i^{(k)} = \max_{1 \le m \le K} -g_i^{(m)}(X) + \epsilon_i^{(m)}\right), \quad k = 1, \dots, K.$$

Equivalently, $p_i(X)$ solves the following rational inattention problem:

$$\min_{p \in \Delta_K} \langle g_i(X), p\rangle + E_i^*(p). \tag{8}$$

Let us stack the choice probabilities of all the agents into a matrix and call it the choice matrix:

$$P(X) = \big(p_1(X), \dots, p_N(X)\big) \in \Delta_K^N. \tag{9}$$

Similarly to the choice matrix, we write $P \in \Delta_K^N$ for any matrix of probability vectors, i.e. $P = \big(p_1, \dots, p_N\big)$ with $p_i \in \Delta_K$, $i = 1, \dots, N$.

2.3 Organizations

Let us describe the behavior of advertising organizations. Their goal is to attract agents as clients by providing them with additional manipulation power. This is done by choosing an appropriate starting distribution, thus, the communication

process is initialized by organizations. By strategic decisions, such as substantial alignment, design, product placements or personal relations, or by direct decisions, such as ranking of a website as result of a certain query or advertising products directly on a marketplace, the organizations determine these vectors, which reflect a network state before interaction starts.

In order to attract the i-th agent with aspired state v_i, the k-th organization selects a starting distribution $x_k \in \Delta_n$ such that $\left\|v_i - M^t \cdot x_k\right\|_2$ becomes small. The organization's goal is to acquire as many agents as possible by simultaneously satisfying the corresponding aspired states. However, the agents are not necessarily equally important for the organization. Instead, the organization primary wants to please agents, who already prefer the organization compared to other competitors. Let us state these considerations in a formal way. An organization k observes to which extent the agents choose it, i.e. quantified by choice probabilities $p_i^{(k)}$, $i = 1, \dots, N$. Thus, the k-th organization measures its performance by the following objective:

$$\sum_{i=1}^{N} p_i^{(k)} \cdot \|v_i - M^t \cdot x_k\|_2. \tag{10}$$

Yet, an organization's choice of the manipulation distribution not only depends on the agents' aspired states, but also on its own objectives. This reflects, that an organization might also aspire a certain state of the network in order to gain profits from the network participants. Therefore, we introduce a payoff function for organization k, which depends on its caused state of manipulation:

$$\pi_k\left(M^t \cdot x_k\right). \tag{11}$$

Let us illustrate by examples how a network state could affect the payoff of the k-th organization. Groups in social media platforms might avoid sharing information with persons who have contrary opinions, such that no arguments against their theories or fake news are communicated. Prohibiting or restricting persons' access to information might be a worthwhile purpose in an information network. Particularly, this is interesting in situations, where direct manipulation of opinions is difficult. Since the authors in Acemoglu and Ozdaglar (2011) mention the source of information as a key component of opinion formation, the manipulation of the information acquirement process contributes to the tampering of opinion formation. A social media influencer might loose credibility of her followers, if they find out about an unacceptable advertise. We state an assumption concerning the payoff functions.

Assumption 3 The payoff function π_k is τ_k-strongly concave w.r.t. the norm $\|\cdot\|_2$ for all $k = 1, \dots, K$.

Altogether, the objective function of the k-th organization incorporates the both goals:

$$\sum_{i=1}^{N} p_i^{(k)} \cdot \|v_i - M^t \cdot x_k\|_2 - \frac{1}{\eta_k} \cdot \pi_k\big(M^t \cdot x_k\big), \tag{12}$$

where $\eta_k > 0$ is a regularization parameter, which shows the importance of payoffs generated by the network. Note that small values of η_k indicate a more restrictive behavior of the k-th organization, meaning that it rather focuses on its own interests than to freely adjust the manipulation distribution according to the agents' aspired states. According to Assumption 3, the negative of the payoff function serves as regularization term. Strongly concave regularization is a well-known and widely-used technique in optimization theory, see e.g. Nesterov (2018). From an economic perspective, the payoff function mimics a stable behavior of the organization. Apart from the already mentioned payoffs generated by the network, this function could also reflect that the k-th organization avoids to deviate too much from a certain targeted state $c_k = M^t \cdot s_k$, where $s_k \in \Delta_n$, due to adjustment costs. As a matter of fact, organizations might know from experience which starting distributions cause network states at a neighborhood of the targeted state, but must take more effort to detect starting distributions for states outside this neighborhood and, thus, face larger adjustment costs. Based on these considerations, a typical τ_k-strongly concave payoff function is

$$\pi_k(M^t \cdot x_k) = -\frac{\tau_k}{2} \|M^t \cdot x_k - c_k\|_2^2.$$

We shall discuss the numerical practicability of this choice later on. For a given choice matrix $P \in \Delta_K^N$, network M and time t, the k-th organization chooses its optimal starting distribution $x_k \in \Delta_n$ by solving

$$\min_{x_k \in \Delta_n} \sum_{i=1}^{N} p_i^{(k)} \cdot \|v_i - M^t \cdot x_k\|_2 - \frac{1}{\eta_k} \cdot \pi_k\big(M^t \cdot x_k\big). \tag{13}$$

For now, we assume that the optimization problems given in (13) have unique solutions for any choice matrix P, which we denote by $x_k(P)$, $k = 1, \dots, K$. We keep these optimal manipulation values in a matrix

$$X(P) = \big(x_1(P), \dots, x_K(P)\big), \tag{14}$$

and call it the manipulation matrix.

2.4 Network manipulation algorithm

In the preceding section we described the behavior of agents and organizations when facing the challenge to manipulate a network in favor of agents' desires. The key aspect is that their behavior summarized in (8) and (13) suggests an alternating interaction between both groups. Organizations enter the market and offer their manipulation distributions. Then, agents observe how satisfactory organizations would manipulate the network state in view of the agents aspired states (e. g. by comparing

past results caused by an organization). Based on these observations, agents make their decisions, i.e. they choose organizations with probability according to (8). The choice probabilities provide feedback to the organizations, which then in turn adjust their starting distributions following the behavior given in (13). By using previous notation, we have the following dynamics:

$$P_{\ell+1} = P(X_\ell), \quad X_{\ell+1} = X(P_{\ell+1}),$$

where X_0 is any feasible starting variable, e. g. $X_0 = \frac{1}{n} \cdot ee^T$.

In what follows, we provide an equivalent description of this network manipulation algorithm in order to better study its convergence properties. For that, we define a potential function which incorporates the behavior of all agents and organizations:

$$\Phi(X,P) = \sum_{i=1}^{N} E_i^*(p_i) + \sum_{k=1}^{K} \left(\sum_{i=1}^{N} p_i^{(k)} \cdot \|v_i - M^t \cdot x_k\|_2 - \frac{1}{\eta_k} \cdot \pi_k\big(M^t \cdot x_k\big) \right). \tag{15}$$

Therefore, the choice matrix solves the following minimization problem:

$$P(X) = \arg\min_{P \in \Delta_K^N} \Phi(X,P). \tag{16}$$

Analogously, we have for a manipulation matrix:

$$X(P) = \arg\min_{X \in \Delta_n^K} \Phi(X,P), \tag{17}$$

which means that the network manipulation algorithm can be viewed as an alternating minimization scheme.

From the viewpoint of computational economics, it seems reasonable to assume that agents and organizations are not able to solve their corresponding optimization problems exactly. Rather than that, the solutions can be obtained up to small errors. This can be for example due to observation errors of the input parameters given by choice and/or manipulation matrices. Another reason could be that exact optimization is time-exhaustive or too costly. In order to incorporate this faulty behavior into our manipulation algorithm, we assume that just inexact minimization in (16) and (17) is possible. More precisely, δ_1-inexact solutions for (16) and δ_2-inexact solutions for (17) are available. We recall that evaluated at a δ-inexact solution the function value is at most the minimum value plus δ, see Sect. 3 for details. Thus, we are ready to state a more general network manipulation algorithm, based on an inexact alternating minimization scheme:

Initialize $\tilde{X}_0 \in \Delta_n^K$. For l = 0, 1, 2, … update:

$$\tilde{P}_{\ell+1} = \arg \overset{\delta_1}{\min_{P \in \Delta_K^N}} \Phi\big(\tilde{X}_\ell, P\big)$$

$$\tilde{X}_{\ell+1} = \arg \overset{\delta_2}{\min_{X \in \Delta_n^K}} \Phi\big(X, \tilde{P}_{\ell+1}\big)$$

The inexact algorithm raises the questions, if the corresponding alternating behavior converges to a stable equilibrium. Do agents and organizations reach a state, where their choices do not change anymore no matter what the starting distributions of the organizations look like? In other words, does a unique minimizer of the potential function exist and does the algorithm converge to this minimizer? Moreover, it is interesting to analyze how the faulty behavior in terms of the errors impacts the possible convergence. We shall answer these questions by applying general results on inexact alternating minimization schemes, which we present in Sect. 3. This is possible since the potential function (15) can be suitably decomposed. For that, we define:

$$f(X) = -\sum_{k=1}^{K} \frac{1}{\eta_k} \cdot \pi_k\big(M^t \cdot x_k\big), \quad h(P) = \sum_{i=1}^{N} E_i^*(p_i). \tag{18}$$

Using the standard inner product, the potential function in (15) can be written as follows:

$$\Phi(X, P) = f(X) + \langle G(X), P\rangle + h(P). \tag{19}$$

3 Inexact alternating minimization

In cases, where the analytical solution of an optimization problem cannot be derived, it is necessary to solve the problem numerically. Normally, this numerical solutions are only exact up to a small δ-error. We review some theoretical aspects of inexact optimization, which we need for convergence analysis. Let us consider optimization problems of the form

$$\min_{z \in Q} \Phi(z), \tag{20}$$

where Φ is a strongly convex function and Q a closed and convex set. We denote by z^* the solution of problem (20). Recall that for a β-strongly convex function Φ it holds:

$$\Phi(z) \geq \Phi(z^*) + \frac{\beta}{2}\|z - z^*\|^2 \quad \text{for all } z \in Q. \tag{21}$$

For a δ-inexact solution we use the standard definition, see e. g. Stonyakin et al. (2019):

Definition 1 A point $\tilde{z}$ is a δ-inexact solution with $\delta \geq 0$, i.e.

$$\tilde{z} \in \arg \overset{\delta}{\min_{z \in Q}} \Phi(z),$$

if and only if there exists $g \in \partial\Phi(\tilde{z})$ such that $\langle g, z^* - \tilde{z}\rangle \geq -\delta$.

Due to Definition 1, a point $\tilde{z}$ provides the minimal objective function value of (20) up to the error δ. This can be easily seen, because $\boldsymbol{\Phi}$ is convex and therefore it holds for any $z \in Q$:

$$\boldsymbol{\Phi}(z^*) \geq \boldsymbol{\Phi}(\tilde{z}) + \langle g, z^* - \tilde{z} \rangle \geq \boldsymbol{\Phi}(\tilde{z}) - \delta,$$

which is equivalent to

$$\boldsymbol{\Phi}(\tilde{z}) \leq \boldsymbol{\Phi}(z^*) + \delta \leq \boldsymbol{\Phi}(z) + \delta. \tag{22}$$

In what follows, we shall focus on decision variable z, which can be separated into two blocks, i.e. $z = (x, p)$. For those situations, alternating minimization methods can be applied. The block structure enables to minimize the objective function for each block separately, which is, in particular, a valuable property for big data applications. Over years, many convergence results for alternating minimization methods under different assumptions were shown. For example, Grippof and Sciandrone (1999) show that updating each component in a sequential manner yields a sequence of iterates such that each limit point is a global minimizer of a continuously differentiable and pseudoconvex function. Under the assumption of Lipschitz continuous gradients and coordinate-wise strong convexity of the objective function, Luo and Tseng (1993) prove linear convergence to a stationary point for constrained problems. Convergence of an alternating minimization scheme for objectives functions with non-differentiable parts has been derived by Beck (2015). Pu et al. (2014) show, that under assumptions such as convexity for one and strong convexity for the other objective term, the inexact alternating minimization algorithm applied to the primal coincides with the inexact proximal gradient method to the dual problem. Recently, in Nesterov (2020) an alternating minimization method was used for soft clustering. There, the objective function additionally includes an interaction term linking both blocks of variables. Under certain assumptions, linear convergence was established provided the problem can be solved exactly in each block. In this paper, we are interested in an inexact alternating minimization algorithm for objective functions equipped with the structure introduced by Nesterov (2020). Let Q_1, Q_2 be closed and convex sets in finite dimensional vector spaces $\mathbb{V}_1, \mathbb{V}_2$ and let $\mathbb{V}$ be a finite dimensional vector space. The objective function is given by

$$\boldsymbol{\Phi}(x, p) = f(x) + \langle G_1(x), G_2(p) \rangle + h(p), \quad x \in Q_1, p \in Q_2, \tag{23}$$

where the operators $G_1 : \mathbb{V}_1 \to \mathbb{V}^*$ and $G_2 : \mathbb{V}_2 \to \mathbb{V}$ are Lipschitz-continuous with moduli L_1 and L_2 on the respective sets Q_1, Q_2. Moreover, we assume that the interaction term $\langle G_1(x), G_2(p) \rangle$ is convex and closed in $x \in Q_1$ for any fixed $p \in Q_2$ and vice versa and that the functions f and h are σ_1- and σ_2-strongly convex on Q_1, respectively on Q_2. Further, we assume the following strict inequality to hold

$$L_1^2 \cdot L_2^2 < \sigma_1 \cdot \sigma_2, \tag{24}$$

under which the function $\boldsymbol{\Phi}$ is shown to be strongly convex on $Q = Q_1 \times Q_2$, see Nesterov (2020). Let the optimal solution of (20) be written as $z^* = (x^*, p^*)$. In order

to solve (20), an alternating minimization method has been proposed by Nesterov (2020). This method generates sequences $\{x_\ell\}_{\ell\geq 0}$ and $\{p_\ell\}_{\ell\geq 1}$ as follows:

$$\begin{aligned}&\text{Choose } x_0 \in Q_1. \text{ For } \ell = 0, 1, 2, \dots \text{ update:}\\ &p_{\ell+1} = \arg\min_{p\in Q_2} \boldsymbol{\Phi}(x_\ell, p) = u(x_\ell)\\ &x_{\ell+1} = \arg\min_{x\in Q_1} \boldsymbol{\Phi}(x, p_{\ell+1}) = v(p_{\ell+1})\end{aligned}$$

Convergence analysis in Nesterov (2020) is based on fixed point iteration. For that, the operators $T : Q_1 \mapsto Q_1$ and $S : Q_2 \mapsto Q_2$ are defined as follows:

$$T(x) = v(u(x)), \quad S(p) = u(v(p)). \tag{25}$$

This enables to write the update step of the alternating minimization scheme:

$$x_{\ell+1} = T(x_\ell), \quad p_{\ell+1} = S(p_\ell).$$

Under condition (24), $T(\cdot)$ and $S(\cdot)$ are contraction mappings. Thus, the linear convergence of the generated sequences to the minimizer (x^*, p^*) of $\boldsymbol{\Phi}$ could be shown in Nesterov (2020):

$$\|x_{\ell+1} - x^*\| \leq \lambda^{\ell+1}\|x(0) - x^*\|, \quad \|p_{\ell+1} - p^*\| \leq \lambda^\ell \|p(1) - p^*\|,$$

where

$$\lambda = \frac{L_1^2 \cdot L_2^2}{\sigma_1 \cdot \sigma_2} < 1.$$

We analyze an inexact version of the alternating minimization method applied to objective functions in (23), when subproblems are solved inexactly in the sense of Definition 1. For that, let us adapt the algorithm in the following way:

$$\begin{aligned}&\text{Choose } x_0 \in Q_1. \text{ For } \ell = 0, 1, 2, \dots \text{ update:}\\ &\tilde{p}_{\ell+1} = \arg\min_{p\in Q_2}^{\delta_1^{(\ell)}} \boldsymbol{\Phi}(\tilde{x}_\ell, p) = u^{\delta_1^{(\ell)}}(\tilde{x}_\ell)\\ &\tilde{x}_{\ell+1} = \arg\min_{x\in Q_1}^{\delta 2} \boldsymbol{\Phi}(x, \tilde{p}_{\ell+1}) = v^{\delta_2^{(\ell)}}(\tilde{p}_{\ell+1})\end{aligned}$$

We allow different accuracy for the above subproblems. Moreover, we allow for iteration-specific errors. The equations also suggest that in iteration ℓ a $\delta^{(\ell)}$-error is made twice. This can be seen by looking at the function values evaluated at two consecutive points of the sequences $\{\tilde{x}_\ell\}_{t\geq 0}$ and $\{\tilde{p}_\ell\}_{t\geq 0}$ generated via the δ-inexact solutions of the auxiliary optimization problems:

$$\begin{aligned}\Phi(\tilde{x}_{\ell+1},\tilde{p}_{\ell+1}) &= f(\tilde{x}_{\ell+1}) + \langle G_1(\tilde{x}_{\ell+1}), G_2(\tilde{p}_{\ell+1})\rangle + h(\tilde{p}_{\ell+1})\\ &\le f(\tilde{x}_{\ell}) + \delta_2^{(\ell)} + \langle G_1(\tilde{x}_{\ell}), G_2(\tilde{p}_{\ell+1})\rangle + h(\tilde{p}_{\ell+1})\\ &\le f(\tilde{x}_{\ell}) + \delta_2^{(\ell)} + \langle G_1(\tilde{x}_{\ell}), G_2(\tilde{p}_{\ell})\rangle + h(\tilde{p}_{\ell}) + \delta_1\\ &\le \Phi(\tilde{x}_{\ell},\tilde{p}_{\ell}) + 2\cdot\max\{\delta_1^{(\ell)},\delta_2^{(\ell)}\}.\end{aligned}$$

Next, we estimate the distances between $u^{\delta_1^{(\ell)}}(x)$ and $u(x)$ as well as between $v^{\delta_2^{(\ell)}}(p)$ and $v(p)$.

Lemma 1 *For any $x \in Q_1$ and $\ell = 0, 1, \ldots$, it holds*:

$$\|u^{\delta_1^{(\ell)}}(x) - u(x)\| \le \sqrt{\frac{2\delta_1^{(\ell)}}{\sigma_2}},$$

and for any $p \in Q_2$ $\ell = 0, 1, \ldots$ it holds:

$$\|v^{\delta_2^{(\ell)}}(p) - v(p)\| \le \sqrt{\frac{2\delta_2^{(\ell)}}{\sigma_1}}.$$

Proof Take an arbitrary iteration ℓ. We apply (21) to derive:

$$\begin{aligned}h(u^{\delta_1^{(\ell)}}(x)) + \langle G_1(x), G_2(u^{\delta_1^{(\ell)}}(x))\rangle \;&\ge\; h(u(x)) + \langle G_1(x), G_2(u(x))\rangle\\ &\quad + \frac{\sigma_2}{2}\|u^{\delta_1^{(\ell)}}(x) - u(x)\|^2.\end{aligned}$$

Due to (22) we additionally have:

$$h(u(x)) + \langle G_1(x), G_2(u(x))\rangle \ge h(u^{\delta_1^{(\ell)}}(x)) + \langle G_1(x), G_2(u^{\delta_1}(x))\rangle - \delta_1^{(\ell)}.$$

Altogether, we obtain:

$$\|u^{\delta_1^{(\ell)}}(x) - u(x)\|^2 \le \frac{2\delta_1^{(\ell)}}{\sigma_2}.$$

The proof for $\|v^{\delta_2^{(\ell)}}(p) - v(p)\|$ follows analogously. □

Let us elaborate on the continuity properties for operators $u^{\delta_1^{(\ell)}}(\cdot)$ and $v^{\delta_2^{(\ell)}}(\cdot)$.

Lemma 2 *For any $x_1, x_2 \in Q_1$ and $\ell = 0, 1, \ldots$ it holds*:

$$\|u^{\delta_1^{(\ell)}}(x_1) - u^{\delta_1^{(\ell)}}(x_2)\| \le 2\cdot\sqrt{\frac{2\delta_1^{(\ell)}}{\sigma_2}} + \frac{L_1\cdot L_2}{\sigma_2}\cdot\|x_1 - x_2\|,$$

and for any $p_1, p_2 \in Q_2$ and $\ell = 0, 1, \ldots$ it holds:

$$\|v^{\delta_2^{(\ell)}}(p_1) - v^{\delta_2^{(\ell)}}(p_2)\| \leq 2 \cdot \sqrt{\frac{2\delta_2^{(\ell)}}{\sigma_1}} + \frac{L_1 \cdot L_2}{\sigma_1} \cdot \|p_1 - p_2\|.$$

Proof Fix an arbitrary iteration ℓ. Applying Lemma 1 and the triangle inequality twice yields:

$$\begin{aligned}\|u^{\delta_1^{(\ell)}}(x_1) - u^{\delta_1^{(\ell)}}(x_2)\| &\leq \|u^{\delta_1^{(\ell)}}(x_1) - u(x_1)\| + \|u(x_1) - u^{\delta_1}(x_2)\| \\ &\leq \sqrt{\frac{2\delta_1^{(\ell)}}{\sigma_2}} + \|u(x_1) - u(x_2)\| + \|u(x_2) - u^{\delta_1}(x_2)\| \\ &\leq \sqrt{\frac{2\delta_1^{(\ell)}}{\sigma_2}} + \frac{L_1 \cdot L_2}{\sigma_2} \cdot \|x_1 - x_2\| + \sqrt{\frac{2\delta_1^{(\ell)}}{\sigma_2}}.\end{aligned}$$

The last inequality is due to Nesterov (2020), where it is shown for any $x_1, x_2 \in Q_1$:

$$\|u(x_1) - u(x_2)\| \leq \frac{L_1 \cdot L_2}{\sigma_2}\|x_1 - x_2\|.$$

Similar reflections yield the result for $\|v^{\delta_2^{(\ell)}}(p_1) - v^{\delta_2^{(\ell)}}(p_2)\|$. □

Let us introduce inexact versions of the operators T and S:

$$T^{\delta^{(\ell)}}(x) = v^{\delta_2^{(\ell)}}(u^{\delta_1^{(\ell)}}(x)), \quad S^{\delta^{(\ell)}}(p) = u^{\delta_1^{(\ell)}}(v^{\delta_2^{(\ell)}}(p)),$$

which we use to rewrite the update of the inexact alternating minimization as

$$\tilde{x}_{\ell+1} = T^{\delta^{(\ell)}}(\tilde{x}_\ell), \quad \tilde{p}_{\ell+1} = S^{\delta^{(\ell)}}(\tilde{p}_\ell). \tag{26}$$

The following result provides uniform continuity up to an error of the operators defined in (26).

Proposition 1 *For any $\tilde{x}_1, \tilde{x}_2 \in Q_1$ and $\ell = 0, 1, \ldots$ it holds*:

$$\|T^{\delta^{(\ell)}}(\tilde{x}_1) - T^{\delta^{(\ell)}}(\tilde{x}_2)\| \leq \lambda\|\tilde{x}_1 - \tilde{x}_2\| + 2 \cdot \sqrt{\frac{2\delta_2^{(\ell)}}{\sigma_1}} + 2 \cdot \sqrt{\frac{2\delta_1^{(\ell)}}{\sigma_2}} \cdot \frac{L_1 \cdot L_2}{\sigma_1},$$

and for any $\tilde{p}_1, \tilde{p}_2 \in Q_2$ it holds:

$$\|S^{\delta^{(\ell)}}(\tilde{p}_1) - S^{\delta^{(\ell)}}(\tilde{p}_2)\| \leq +\lambda\|\tilde{p}_1 - \tilde{p}_2\| + 2 \cdot \sqrt{\frac{2\delta_1^{(\ell)}}{\sigma_2}} + 2 \cdot \sqrt{\frac{2\delta_2^{(\ell)}}{\sigma_1}} \cdot \frac{L_1 \cdot L_2}{\sigma_2}.$$

Proof We apply Lemma 2 to derive

$$\begin{aligned}\|T^{\delta^{(\ell)}}(\tilde{x}_1) - T^{\delta^{(\ell)}}(\tilde{x}_2)\| &= \|v^{\delta_2^{(\ell)}}(u^{\delta_1^{(\ell)}}(\tilde{x}_1)) - v^{\delta_2^{(\ell)}}(u^{\delta_1^{(\ell)}}(\tilde{x}_2))\| \\ &\le 2 \cdot \sqrt{\frac{2\delta_2^{(\ell)}}{\sigma_1}} + \frac{L_1 \cdot L_2}{\sigma_1} \cdot \|u^{\delta_1^{(\ell)}}(\tilde{x}_1) - u^{\delta_1^{(\ell)}}(\tilde{x}_2)\| \\ &\le 2 \cdot \sqrt{\frac{2\delta_2^{(\ell)}}{\sigma_1}} + \frac{L_1 \cdot L_2}{\sigma_1} \cdot 2\sqrt{\frac{2\delta_1^{(\ell)}}{\sigma_2}} + \underbrace{\frac{L_1^2 \cdot L_2^2}{\sigma_1 \cdot \sigma_2}}_{=\lambda} \cdot \|\tilde{x}_1 - \tilde{x}_2\|.\end{aligned}$$

Again, the second assertion follows similarly. □

Since we cannot rely on the contraction property of $T^{\delta^{(\ell)}}$ and $S^{\delta^{(\ell)}}$, the convergence analysis of the sequences $\{\tilde{x}_\ell\}_{\ell \ge 0}$ and $\{\tilde{p}_\ell\}_{\ell \ge 1}$ becomes involved. For that, we start with the following auxiliary result.

Lemma 3 *For any $x \in Q_1$ and $\ell = 0, 1, \ldots$ it holds*:

$$\left\|T(x) - T^{\delta^{(\ell)}}(x)\right\| \le \sqrt{\frac{2\delta_2^{(\ell)}}{\sigma_1}} + \frac{L_1 \cdot L_2}{\sigma_1} \cdot \sqrt{\frac{2\delta_1^{(\ell)}}{\sigma_2}},$$

and for any $p \in Q_2$ and $\ell = 0, 1, \ldots$ it holds:

$$\left\|S(p) - S^{\delta^{(\ell)}}(p)\right\| \le \sqrt{\frac{2\delta_1^{(\ell)}}{\sigma_2}} + \frac{L_1 \cdot L_2}{\sigma_2} \cdot \sqrt{\frac{2\delta_2^{(\ell)}}{\sigma_1}}.$$

Proof We show the first part. It follows by means of Lemmata 1 and 2.

$$\begin{aligned}\left\|T(x) - T^{\delta^{(\ell)}}(x)\right\| &= \left\|v(u(x)) - v^{\delta_2^{(\ell)}}(u^{\delta_1^{(\ell)}}(x))\right\| \\ &\le \left\|v(u(x)) - v(u^{\delta_1^{(\ell)}}(x))\right\| + \left\|v(u^{\delta_1^{(\ell)}}(x)) - v^{\delta_2^{(\ell)}}(u^{\delta_1^{(\ell)}}(x))\right\| \\ &\le \frac{L_1 \cdot L_2}{\sigma_1} \cdot \left\|u(x) - u^{\delta_1^{(\ell)}}(x)\right\| + \sqrt{\frac{2\delta_2^{(\ell)}}{\sigma_1}} \\ &\le \frac{L_1 \cdot L_2}{\sigma_1} \cdot \sqrt{\frac{2\delta_1^{(\ell)}}{\sigma_2}} + \sqrt{\frac{2\delta_2^{(\ell)}}{\sigma_1}}.\end{aligned}$$

Clearly, the proof of the second part is similar. □

Now we are ready to state the main result concerning convergence of the inexact alternating minimization scheme.

Theorem 1 *For the inexact alternating minimization scheme holds*:

$$\|\tilde{x}_{\ell+1} - x^*\| \leq \lambda^{\ell+1}\|x_0 - x^*\| + \sum_{k=0}^{\ell} \lambda^k \left[\sqrt{\frac{2\delta_2^{(\ell-k)}}{\sigma_1}} + \frac{L_1 \cdot L_2}{\sigma_1} \cdot \sqrt{\frac{2\delta_1^{(\ell-k)}}{\sigma_2}} \right] \tag{27}$$

and

$$\|\tilde{p}_{\ell+1} - p^*\| \leq \lambda^{\ell} \left[\|p_1 - p^*\| + \sqrt{\frac{2\delta_1^{(0)}}{\sigma_2}} \right] + \sum_{k=1}^{\ell} \lambda^k \left[\sqrt{\frac{2\delta_1^{(\ell-k)}}{\sigma_2}} + \frac{L_1 \cdot L_2}{\sigma_2} \cdot \sqrt{\frac{2\delta_2^{(\ell-k)}}{\sigma_1}} \right]. \tag{28}$$

Proof We apply Lemma 3 to derive:

$$\begin{aligned}
\|\tilde{x}_{\ell+1} - x_{\ell+1}\| &\leq \|T(x_\ell) - T(\tilde{x}_\ell)\| + \left\|T(\tilde{x}_\ell) - T^{\delta^{(\ell)}}(\tilde{x}_\ell)\right\| \\
&\leq \lambda \cdot \|x_\ell - \tilde{x}_\ell\| + \sqrt{\frac{2\delta_2^{(\ell)}}{\sigma_1}} + \frac{L_1 \cdot L_2}{\sigma_1} \cdot \sqrt{\frac{2\delta_1^{(\ell)}}{\sigma_2}} \\
&\leq \ldots \leq \lambda^{\ell+1} \underbrace{\|x_0 - x_0\|}_{=0} + \sum_{k=0}^{\ell} \lambda^k \left[\sqrt{\frac{2\delta_2^{(\ell-k)}}{\sigma_1}} + \frac{L_1 \cdot L_2}{\sigma_1} \cdot \sqrt{\frac{2\delta_1^{(\ell-k)}}{\sigma_2}} \right].
\end{aligned}$$

We are therefore able to estimate the distance of the $(\ell + 1)$-th iterate to the minimizer:

$$\begin{aligned}
\|\tilde{x}_{\ell+1} - x^*\| &\leq \|\tilde{x}_{\ell+1} - x_{\ell+1}\| + \|x_{\ell+1} - x^*\| \\
&\leq \sum_{k=0}^{\ell} \lambda^k \left[\sqrt{\frac{2\delta_2^{(\ell-k)}}{\sigma_1}} + \frac{L_1 \cdot L_2}{\sigma_1} \cdot \sqrt{\frac{2\delta_1^{(\ell-k)}}{\sigma_2}} \right] + \lambda^{\ell+1}\|x_0 - x^*\|.
\end{aligned}$$

For the proof of the inequality (30) note that the first iterate of the algorithm is not chosen freely. Instead it is the solution of the corresponding optimization problem. Hence, the first iterates of the exact and inexact version are in general not equal, i.e. $p_1 \neq \tilde{p}_1$, which provides

$$\|\tilde{p}_{\ell+1} - p_{\ell+1}\| \leq \lambda^{\ell}\|\tilde{p}_1 - p_1\| + \sum_{k=1}^{\ell} \lambda^k \left[\sqrt{\frac{2\delta_1^{(\ell-k)}}{\sigma_2}} + \frac{L_1 \cdot L_2}{\sigma_2} \cdot \sqrt{\frac{2\delta_2^{(\ell-k)}}{\sigma_1}} \right].$$

It remains to recall that $p_1 = u(x_0)$ and $\tilde{p}_1 = u^{\delta_1^{(0)}}(x_0)$ and apply Lemma 1. The result (30) follows then in the same manner as for (29). □

According to Theorem 1, the inexact alternating minimization does not converge in general. Yet, if the sequences of errors $\left\{\delta_1^{(\ell)}\right\}_{\ell \geq 0}$ and $\left\{\delta_2^{(\ell)}\right\}_{\ell \geq 0}$ are not growing, the right hand side of Eqs. (27) and (28) can be controlled by the model's parameter. In order to reach good convergence results, Theorem 1 suggests that at the beginning of the inexact alternating minimization algorithm the

problems can be solved up to bigger errors, whereas at later iterations the errors shall be reduced.

Corollary 1 *Let in each iteration the same errors are made, i. e.* $\delta_1^{(\ell)} = \delta_1$ *and* $\delta_2^{(\ell)} = \delta_2$ *for* $\ell = 0, 1, \ldots$. *Then, for the inexact alternating minimization scheme holds*:

$$\|\tilde{x}_{\ell+1} - x^*\| \leq \lambda^{\ell+1}\|x_0 - x^*\| + \frac{1-\lambda^\ell}{1-\lambda}\left[\sqrt{\frac{2\delta_2}{\sigma_1}} + \frac{L_1 \cdot L_2}{\sigma_1} \cdot \sqrt{\frac{2\delta_1}{\sigma_2}}\right] \tag{29}$$

and

$$\|\tilde{p}_{\ell+1} - p^*\| \leq \lambda^\ell\left[\|p_1 - p^*\| + \sqrt{\frac{2\delta_1}{\sigma_2}}\right] + \frac{1-\lambda^{\ell-1}}{1-\lambda}\left[\sqrt{\frac{2\delta_1}{\sigma_2}} + \frac{L_1 \cdot L_2}{\sigma_2} \cdot \sqrt{\frac{2\delta_2}{\sigma_1}}\right]. \tag{30}$$

Proof This directly follows from Theorem 1 and the fact that $\lambda < 1$. □

According to Corollary 1, the distance to the minimizer is bounded by the second term of the right hand side of inequalities (29) and (30). By taking the limits, we obtain:

$$\lim_{\ell\to\infty} \|\tilde{x}_{\ell+1} - x^*\| \leq \frac{1}{1-\lambda}\left[\sqrt{\frac{2\delta_2}{\sigma_1}} + \frac{L_1 \cdot L_2}{\sigma_1} \cdot \sqrt{\frac{2\delta_1}{\sigma_2}}\right]$$

and

$$\lim_{\ell\to\infty} \|\tilde{p}_{\ell+1} - p^*\| \leq \frac{1}{1-\lambda}\left[\sqrt{\frac{2\delta_1}{\sigma_2}} + \frac{L_1 \cdot L_2}{\sigma_2} \cdot \sqrt{\frac{2\delta_2}{\sigma_1}}\right].$$

Obviously, convergence is guaranteed if the subproblems can be solved exactly, i.e. if $\delta_1 = \delta_2 = 0$. This is not surprising as in this case the inexact alternating minimization scheme coincides with the exact method proposed by Nesterov (2020). Therefore, the iterates generated by the inexact alternating minimization scheme (26) coincide with those generated by the exact method. Inequalities (29) and (30) show that the total error of the inexact alternating minimization scheme can be controlled. Furthermore, large convexity parameters not only improve the rate of convergence for the exact version of the algorithm, but also decrease the total accumulated error in the inexact scenario.

4 Convergence analysis

We analyze the convergence of our network manipulation algorithm by applying the general theory of inexact alternating minimization from Sect. 3. First, we estimate the convexity parameter of

$$h(P) = \sum_{i=1}^{N} E_i^*(p_i)$$

w.r.t. the norm

$$\|P\|_{\mathbb{H}} = \left(\sum_{i=1}^{N} \|p_i\|_1^2 \right)^{\frac{1}{2}}, \quad P \in \Delta_K^N.$$

It turns out that the strong convexity of E_i^* holds due to Assumption 2. This has been recently shown in Müller et al. (2021a).

Lemma 4 (Müller et al. (2021a)) *Let the differences $\epsilon_i^{(k)} - \epsilon_i^{(m)}$ of random errors have modes $\bar{z}_i^{k,m} \in \mathbb{R}$, $k \neq m$. Then, the corresponding convex conjugate E_i^* is β_i-strongly convex w.r.t. the norm $\|\cdot\|_1$, where the convexity parameter is given by*

$$\beta_i = \frac{1}{2 \sum_{k=1}^{K} \sum_{m \neq k} g_i^{k,m}\left(\bar{z}_i^{k,m}\right)},$$

and $g_i^{k,m}$ denotes the density function of $\epsilon_i^{(k)} - \epsilon_i^{(m)}$.

Let us review important discrete choice models in accordance with Assumption 2, where convexity parameters can be explicitly estimated.

Remark 1 In the multinomial logit model (MNL), the error terms are IID Gumbel distributed with zero location parameter and variance $\frac{\pi \cdot \mu}{\sqrt{6}}$, where $\mu > 0$, see e.g. Anderson et al. (1992). The choice probabilities are:

$$\mathbb{P}\left(u^{(k)} + \epsilon^{(k)} = \max_{1 \leq m \leq K} u^{(m)} + \epsilon^{(m)} \right) = \frac{e^{\frac{u^{(k)}}{\mu}}}{\sum_{m=1}^{K} e^{\frac{u^{(m)}}{\mu}}}, \quad k = 1, \ldots, K.$$

From the choice probabilities we can conclude that the parameter μ reflects the randomness of the decision. If μ converges to zero, this would lead to the deterministic decision based on the observable utility only. On the other hand, very large values of the parameter provide very random choices, tending towards the uniform distribution in the limit. The convex conjugate of the corresponding surplus function is up to an additive constant:

$$E^*(p) = \mu \sum_{k=1}^{K} p^{(k)} \cdot \ln p^{(k)}.$$

It is well known from Pinsker inequality that this function is μ-strongly convex w.r.t. the norm $\|\cdot\|_1$.

Remark 2 Another famous example is the nested logit model (NL) introduced in McFadden (1978). Compared to the MNL, the NL is more appropriate in situations where some of the alternatives are correlated, i.e. the axiom of irrelevance of independent alternatives is violated, see e. g. Anderson et al. (1992). In the NL, each alternative k belongs to one of L different nests $N_\ell \subset \{1, \dots, K\}$ for $\ell = 1, \dots, L$. The choice probabilities for $k \in N_\ell$, $\ell \in L$ are

$$\mathbb{P}\left(u^{(k)} + \epsilon^{(k)} = \max_{1 \le m \le K} u^{(m)} + \epsilon^{(m)}\right) = \frac{e^{\mu_\ell \ln \sum_{m \in N_\ell} e^{\frac{u^{(m)}}{\mu_\ell}}}}{\sum_{\ell \in L} e^{\mu_\ell \ln \sum_{m \in N_\ell} e^{\frac{u^{(m)}}{\mu_\ell}}}} \cdot \frac{e^{\frac{u^{(k)}}{\mu_\ell}}}{\sum_{m \in N_\ell} e^{\frac{u^{(m)}}{\mu_\ell}}},$$

where the following condition shall be satisfied:

$$0 < \mu_\ell \le 1, \quad \ell = 1, \dots, L.$$

The parameter μ_ℓ determines the randomness of choices within the ℓ-th nest. Further, the correlation of alternatives within the same ℓ-th nest is given by $1 - \mu_\ell^2$. The convex conjugate of the NL surplus function has been derived up to an additive constant in Fosgerau et al. (2020):

$$E^*(p) = \sum_{\ell \in L} \mu_\ell \sum_{i \in N_\ell} p^{(m)} \ln p^{(m)} + \sum_{\ell \in L} \left(1 - \mu_\ell\right) \left(\sum_{m \in N_\ell} p^{(m)}\right) \ln \left(\sum_{m \in N_\ell} p^{(m)}\right).$$

It is $\left(\min_{\ell \in L} \mu_\ell\right)$-strongly convex w.r.t. the norm $\|\cdot\|_1$, see Müller et al. (2021b).

Remark 3 MNL and NL belong to the broader class of generalized nested logit models (GNL) introduced in Wen and Koppelman (2001). GNL surplus functions are determined by the generating function

$$G(x) = \sum_{\ell \in L} \left(\sum_{i=1}^{n} \left(\sigma_{i\ell} \cdot x^{(i)}\right)^{\frac{1}{\mu_\ell}}\right)^{\frac{\mu_\ell}{\mu}}.$$

Here, L is a generic set of nests. The parameters $\sigma_{i\ell} \ge 0$ denote the shares of the i-th alternative with which it is attached to the ℓ-th nest. For any fixed $i \in \{1, \dots, n\}$ they sum up to one:

$$\sum_{\ell \in L} \sigma_{i\ell} = 1,$$

and $\sigma_{i\ell} = 0$ means that the ℓ-th nest does not contain the i-th alternative. Hence, the set of alternatives within the ℓ-th nest is

$$N_\ell = \left\{i \mid \sigma_{i\ell} > 0\right\}.$$

The nest parameters $\mu_\ell > 0$ describe the variance of the random errors while choosing alternatives within the ℓ-th nest. Analogously, $\mu > 0$ describes the variance of

the random errors while choosing among the nests, where the following conditions shall be satisfied

$$\mu_\ell \leq \mu \quad \text{for all } \ell \in L.$$

Apart from MNL and NL, the concrete specification of the surplus' convex conjugate E^* is not known yet. Estimates of the convexity parameter of the convex conjugate are derived in Müller et al. (2021a). However, the choice probabilities are given as a formula.

The choice probability of the i-th alternative according to GNL amounts to

$$\mathbb{P}\left(u^{(i)} + \epsilon^{(i)} = \max_{1\leq i\leq n} u^{(i)} + \epsilon^{(i)}\right) = \mu \frac{\partial G(e^u)}{\partial x^{(i)}} \cdot \frac{e^{u^{(i)}}}{G(e^u)} = \sum_{\ell\in L} q_\ell \cdot p_{i\ell},$$

where we set $e^u = \left(e^{u^{(1)}}, \dots, e^{u^{(n)}}\right)$ for the sake of brevity.

We state Lemma 5 concerning the strong convexity of the function h defined in (18).

Lemma 5 *Let the functions E_i^* be β_i-strongly convex w.r.t. the norm $\|\cdot\|_1$, $i = 1, \dots, N$. Then, the function h is σ_2-strongly convex w.r.t. the norm $\|\cdot\|_{\mathbb{H}}$, where*

$$\sigma_2 = \min_{1\leq i\leq N} \beta_i.$$

Proof Take any $P, Q \in \Delta_K^N$, $\alpha \in [0, 1]$. Then the following holds

$$\begin{aligned}
h(\alpha \cdot P + (1-\alpha) \cdot Q) &= \sum_{i=1}^N E_i^*\big(\alpha \cdot p_i + (1-\alpha) \cdot q_i\big) \\
&\leq \alpha \cdot \sum_{i=1}^N E_i^*(p_i) + (1-\alpha) \cdot \sum_{i=1}^N E_i^*(q_i) \\
&\quad - \alpha \cdot (1-\alpha) \cdot \sum_{i=1}^N \frac{\beta_i}{2} \|p_i - q_i\|_1^2 \\
&\leq \alpha \cdot \sum_{i=1}^N E^*(p_i) + (1-\alpha) \cdot \sum_{i=1}^N E^*(q_i) \\
&\quad - \alpha \cdot (1-\alpha) \cdot \frac{\sigma_2}{2} \cdot \sum_{i=1}^N \|p_i - q_i\|_1^2 \\
&= \alpha \cdot h(P) + (1-\alpha) \cdot h(Q) - \alpha \cdot (1-\alpha) \cdot \frac{\sigma_2}{2} \cdot \|P - Q\|_{\mathbb{H}}^2.
\end{aligned}$$

□

Hence, the worst convexity parameter amongst all agents determines the strong convexity of the function $h(P)$. In order to apply results from Sect. 3, we secondly need to show that

$$f(X) = -\sum_{k=1}^{K} \frac{1}{\eta_k} \cdot \pi_k\big(M^t \cdot x_k\big)$$

is strongly convex w.r.t. the norm

$$\|X\|_{\mathbb{F}} = \left[\sum_{k=1}^{K} \|x_k\|_2^2\right]^{\frac{1}{2}}, \quad X \in \Delta_n^K.$$

For that, we need to assume that the underlying network is regular.

Assumption 4 The smallest singular value of M is positive, i. e. $\sigma_{\min}(M) > 0$ holds.

As a consequence, we are able to estimate the convexity parameter of f w.r.t. the norm $\|\cdot\|_{\mathbb{F}}$.

Lemma 6 *The function f is σ_1-strongly convex w.r.t. the norm $\|\cdot\|_{\mathbb{F}}$, where*

$$\sigma_1 = \min_{1 \le k \le K} \frac{\tau_k}{\eta_k} \cdot \big[\sigma_{\min}(M)\big]^{2t}.$$

Proof First, we recall:

$$\sigma_{\min}(M) = \min_{\|x\|_2 = 1} \|M \cdot x\|_2.$$

Hence, we get:

$$\begin{aligned}\sigma_{\min}\big(M^t\big) = \sigma_{\min}\big(M \cdot M^{t-1}\big) &= \min_{\|x\|_2=1} \Big\|M \cdot M^{t-1} \cdot x\Big\|_2 \\ &\ge \sigma_{\min}(M) \cdot \min_{\|x\|_2=1} \Big\|M^{t-1} \cdot x\Big\|_2 \ge \ldots \ge \big[\sigma_{\min}(M)\big]^t.\end{aligned}$$

For any $\alpha \in [0,1]$ and $X, Z \in \Delta_n^K$ it holds due to the τ_k-strong convexity of $-\pi_k$ w.r.t. the norm $\|\cdot\|_2$:

$$\begin{aligned}&-\pi_k\big(\alpha \cdot M^t \cdot x_k + (1-\alpha) \cdot M^t \cdot z_k\big) \\ &\quad \le -\alpha \cdot \pi_k\big(M^t \cdot x_k\big) - (1-\alpha) \cdot \pi_k\big(M^t \cdot z_k\big) \\ &\quad - \alpha \cdot (1-\alpha) \cdot \frac{\tau_k}{2} \cdot \|M^t x_k - M^t z_k\|_2^2.\end{aligned}$$

Further, we have:

$$\|M^t x_k - M^t z_k\|_2 \ge \sigma_{\min}\big(M^t\big) \cdot \|x_k - z_k\|_2 \ge \big[\sigma_{\min}(M)\big]^t \cdot \|x_k - z_k\|_2.$$

Hence, the convexity parameter of $-\pi_k(M^t x_k)$ is $\tau_k \cdot [\sigma_{\min}(M)]^{2t}$. The assertion follows analogously to the proof of Lemma 5. □

Note that the considerations in the proof of Lemma 6 also guarantee the existence of a unique minimizer $x_k^*(P)$ for each objective function in (13), i.e. the manipulation matrix $X_*(P)$ is indeed well defined.

It remains to inspect the multiplicative term. We study the Lipschitz-continuity property of the operator

$$G(X) = \big(g_1(X), \dots, g_N(X)\big) \in \mathbb{R}^{K\times N},$$

where

$$g_i(X) = \big(\|v_i - M^t \cdot x_1\|_2, \dots, \|v_i - M^t \cdot x_K\|_2\big)^T, \; i = 1, \dots, N.$$

For that, the dual norm of $\|\cdot\|_{\mathbb{H}}$ is required, see Nesterov (2020):

$$\|Z\|_{\mathbb{H}}^* = \left[\sum_{i=1}^N \|z_i\|_\infty^2\right]^{\frac{1}{2}}, \quad Z \in \mathbb{R}^{K\times N}.$$

Lemma 7 *The operator G is Lipschitz-continuous with modulus*

$$L_1 = N^{\frac{1}{2}} \cdot \big[\sigma_{\max}(M)\big]^t,$$

where $\sigma_{\max}$ denotes the largest singular value of M. This is to say that

$$\|G(X) - G(Y)\|_{\mathbb{H}}^* \le L_1 \cdot \|X - Y\|_{\mathbb{F}}, \quad X, Y \in \Delta_n^K.$$

Proof The Lipschitz-continuity of G follows mainly from (Nesterov 2020). In fact, take any $X, Y \in \Delta_n^K$:

$$\|G(X) - G(Y)\|_{\mathbb{H}}^* = \left[\sum_{i=1}^N \|g_i(X) - g_i(Y)\|_\infty^2\right]^{\frac{1}{2}}.$$

It holds by means of the triangle inequality:

$$\begin{aligned} \big|g_i(X) - g_i(Y)\big| &= \big|\|v_i - M^t \cdot x_k\|_2 - \|v_i - M^t \cdot y_k\|_2\big| \\ &\le \|M^t x_k - M^t y_k\|_2 \le \big[\sigma_{\max}(M)\big]^t \cdot \|x_k - y_k\|_2. \end{aligned}$$

Therefore,

$$\begin{aligned}\|g_i(X) - g_i(Y)\|_\infty^2 &= \left(\max_{1\le k\le K} \left|\|v_i - M^t \cdot x_k\|_2 - \|v_i - M^t \cdot y_k\|_2\right|\right)^2 \\ &\le \left[\sigma_{\max}(M)\right]^{2t} \cdot \max_{1\le k\le K} \|x_k - y_k\|_2^2 \\ &\le \left[\sigma_{\max}(M)\right]^{2t} \cdot \sum_{k=1}^{K} \|x_k - y_k\|_2^2,\end{aligned}$$

and the assertion follows. □

Note that all components of G are convex and nonnegative. Moreover, all entries of the matrices P are nonnegative. Due to Lemmata 5 and 6, $\Phi(\cdot, P)$ is strongly convex for any fixed $P \in \Delta_K^N$ and $\Phi(X, \cdot)$ is strongly convex for any fixed $X \in \Delta_n^K$. We therefore conclude that the alternating update steps of our network manipulation algorithm are well defined.

Let us finally present our main results on the convergence of the network manipulation algorithm. Recall that the derived constants are as follows:

$$\sigma_1 = \min_{1\le k\le K} \frac{\tau_k}{\eta_k} \cdot \left[\sigma_{\min}(M)\right]^{2t}, \quad \sigma_2 = \min_{1\le i\le N} \beta_i, \tag{31}$$

and

$$L_1 = N^{\frac{1}{2}} \cdot \left[\sigma_{\max}(M)\right]^t, \quad L_2 = 1. \tag{32}$$

Moreover, for the rate of convergence we have:

$$\lambda = \frac{L_1^2 \cdot L_2^2}{\sigma_1 \cdot \sigma_2} = \frac{N \cdot [\kappa(M)]^{2t}}{\min\limits_{1\le k\le K} \frac{\tau_k}{\eta_k} \cdot \min\limits_{1\le i\le N} \beta_i}, \tag{33}$$

where $\kappa(M)$ denotes the condition number of the matrix M. In order to establish convergence of the network manipulation algorithm, we need an additional assumption which indicates a certain stability for the model.

Assumption 5 It holds:

$$[\kappa(M)]^t < \left(\frac{\min\limits_{1\le k\le K} \frac{\tau_k}{\eta_k} \cdot \min\limits_{1\le i\le N} \beta_i}{N}\right)^{\frac{1}{2}}. \tag{34}$$

Assumption 5 is a version of condition (24), which enforces $\lambda < 1$ and thereby, guarantees strong convexity of the potential function (19). The straightforward application of Theorem 1 respectively Corollary 1 now provides:

Theorem 2 *Let* $(X^*, P^*) \in \Delta_n^K \times \Delta_K^N$ *be the unique minimizer of the potential function* (19). *Then, for the sequences* $\{\tilde{X}_\ell\}_{\ell\geq 0}$ *and* $\{\tilde{P}_\ell\}_{\ell\geq 1}$ *it holds*:

$$\|\tilde{X}_{\ell+1} - X^*\|_{\mathbb{F}} \leq \lambda^{\ell+1}\|X_0 - X^*\|_{\mathbb{F}} + \frac{1-\lambda^\ell}{1-\lambda}\left[\sqrt{\frac{2\delta_2}{\sigma_1}} + \frac{L_1 \cdot L_2}{\sigma_1} \cdot \sqrt{\frac{2\delta_1}{\sigma_2}}\right] \quad (35)$$

and

$$\|\tilde{P}_{\ell+1} - P^*\|_{\mathbb{H}} \leq \lambda^\ell\left[\|P_1 - P^*\|_{\mathbb{H}} + \sqrt{\frac{2\delta_1}{\sigma_2}}\right] + \frac{1-\lambda^{\ell-1}}{1-\lambda}\left[\sqrt{\frac{2\delta_1}{\sigma_2}} + \frac{L_1 \cdot L_2}{\sigma_2} \cdot \sqrt{\frac{2\delta_2}{\sigma_1}}\right], \quad (36)$$

where $\sigma_1, \sigma_2, L_1, L_2$, *and* λ *are given in* (31)–(33).

Let us comment on Assumption 5 by elaborating how the model parameters enter into the inequality (34):

Interaction network The network structure plays a key role in (34). This is reflected by the condition number $\kappa(M)$. Its large values cause instability of manipulation, since small changes regarding the aspired states could lead to big changes of optimal starting distributions. In other words, a more predictive pattern of network transitions speeds up the convergence. The minimum value of the condition number is attained for permutation matrices. In this case, the network interaction is obviously predictable, i.e. organizations can easily determine how network participants distribute information. For similar reasons, the number of interaction periods t has a negative impact on possibility of manipulation. More periods hamper the influence of the starting distribution on the resulting state. Instead, if time progresses the state is mainly determined just by the network structure independently of the starting distributions.

Agents Clearly, more agents N slow down the rate, as organizations have to pay attention to more aspired states. Moreover, large values of β_i, $i = 1, \ldots, N$, improve the rate of convergence. In order to interpret this fact, we refer to Remark 1. There, it has been shown that β_i's, can be viewed as measures that agents are still pretty uncertain about their decisions. Due to the duality of discrete choice and rational inattention, agents prone to errors have high information processing costs. Thus, these agent pay less attention to the observable utility, i.e. if their aspired states were reached. The fact that imperfect behavior of agents could help to faster stabilize economic systems was recently also described in Müller et al. (2021b).

Organizations The parameters η_k, $k = 1, \dots, K$, reflect to which extent organizations take into account their network payoffs. If η_k's are relatively large, the organizations focus mainly on reaching the agents' aspired states. It seems surprising that this would not improve the convergence rate of the network manipulation algorithm, but actually worsen it. However, if organizations do not properly act on the network by maximizing their profits, their manipulation power diminishes, since they lose their credibility – e.g., their followers may be disappointed by getting biased information and leave them. Organizations thus become worthless for agents in terms of manipulation and, as consequence, the network manipulation algorithm becomes less efficient. Hence, the parameter mirrors a certain credibility of the organizations. Further, the impact of τ_k, $k = 1, \dots, K$, on the convergence rate becomes clear if we interpret the parameters as measures of organizations' reluctance to change their starting distributions. From this point of view, the conservative behavior of organizations towards profit maximization makes the network manipulation more stable.

Additionally, it is worth to mention that the parameters β_i's, η_k's, and τ_k's, which reflect the behavior of agents and organizations, also affect the error bounds in (35) and (36). The corresponding interpretation is similar to that for the convergence rate. Namely, the agents' imperfect behavior and organizations' conservatism in profit maximization reduce the accumulated errors.

5 Computational aspects

We discuss the implementation and benefits of the inexact alternating minimization scheme applied to the objective function (15). We recall that we have to minimize a strongly convex function $\Phi(X, P)$ over a convex and bounded set $\Delta_n^K \times \Delta_K^N$. Alternative ways to find a solution are therefore minimization via direct methods. The efficiency of these methods obviously depends on the properties of the objective function. By applying the alternating minimization scheme instead, it is possible to exploit the properties of the components of $\Phi(X, P)$ separately, as each iteration consists of solving the two subproblems

$$\tilde{P}_{\ell+1} = \arg \min_{P \in \Delta_K^N}^{\delta_1} \Phi\left(\tilde{X}_\ell, P\right), \tag{37}$$

$$\tilde{X}_{\ell+1} = \arg \min_{X \in \Delta_n^K}^{\delta_2} \Phi\left(X, \tilde{P}_{\ell+1}\right). \tag{38}$$

Hence, the performance of the alternating scheme crucially depends on how efficiently these subproblems can be solved. Furthermore, the inexact version provides

an opportunity to accelerate the alternating scheme, since we can accept approximate solutions of the subproblems and thus, stop an optimization method at an earlier iteration. In fact, due to Theorem 2 the impact of these numerical inaccuracies on the convergence of the alternating scheme can be controlled by means of the model's parameters.

5.1 Agent's subproblem

The complexity of solving the agent's subproblem (37) depends on the concrete specification of the underlying discrete choice model. In order to update the choice matrix $\tilde{P}_{\ell+1}$, the following minimization problem has to be inexactly solved:

$$\min_{P \in \Delta_K^N}^{\delta_1} \langle G(\tilde{X}_\ell), P \rangle + h(P).$$

Notice that this minimization problem is separable, which yields:

$$\min_{p_i \in \Delta_K}^{\delta_1} \langle g_i(\tilde{X}_\ell), p_i \rangle + E_i^*(p_i), \quad i = 1, \dots, N. \tag{39}$$

The solution of each of these problems is given by the choice probabilities of the underlying discrete choice model, cf. Fosgerau et al. (2020) and Müller et al. (2021a). The challenge of solving problem (39) lies in the concrete specification of the function E_i^*. In general, the derivation of this convex conjugate of the surplus function can be very involved. In fact, for many discrete choice models the convex conjugate E_i^* is not known yet. This is e.g. the case for the most of the generalized nested logit, the probit or the mixed logit models. However, for a large class of discrete choice models we are able to inexactly solve (39), even without the knowledge of the functions E_i^*. Let us discuss this in more detail:

- For a variety of discrete choice models the choice probabilities are given by a *formula*. This is the case for the generalized nested logit models, where the formula is presented in Remark 1. Note that the formulas for multinomial logit and nested logit are a special case, see Remark 3. Thus, we are able to solve Problem (39) without knowing the concrete specification of the function E_i^*. In this case, the computational costs of solving the subproblem are determined by the costs of evaluating the distance matrix, which can be done in $\mathcal{O}(KnN)$ operations, if we assume that the costs of computing $\|v - M^t \cdot x\|_2$ are $\mathcal{O}(n)$.
- For general discrete choice models the *simulation* of the choice probabilities can be performed, see Train (2009). This is the case e. g. for multinomial probit where the random errors are normally distributed or the mixed logit which can approximate any random utility model, see McFadden and Train (2000). Those two models are very flexible in terms of modeling substitution, however, simulation of the choice probabilities at each iteration could be computationally expensive. For the multinomial probit it is possible to use analytical approximations of the integral. Connors et al. (2014) show that such approximations perform rather

fast in numerical tests compared to simulation approaches such as the GHK-simulator by Börsch-Supan and Hajivassiliou (1993).

We stress that for applying inexact alternating minimization method it is sufficient to have approximate solutions of the subproblems (39). Exact solutions of (39) are desirable, but not needed.

5.2 Organization's subproblem

In order to update the manipulation values, we have to inexactly solve the subproblem (38) or, equivalently,

$$\min_{X \in \Delta_n^K}^{\delta_2} f(X) + \langle G(X), \tilde{P}_{\ell+1} \rangle.$$

Its decomposable structure enables to solve for any $k = 1, \dots K$:

$$\min_{x_k \in \Delta_n}^{\delta_2} \sum_{i=1}^{N} \tilde{p}_i^{(k)} \cdot \|v_i - M^t \cdot x_k\|_2 - \frac{1}{\eta_k} \cdot \pi_k\big(M^t \cdot x_k\big). \tag{40}$$

The computational efficiency of a chosen algorithm applied to (40) depends on the properties of the payoff functions π_k. Note that these functions are allowed to be nonsmooth as well as not necessarily simple. In the most general situation, we might, hence, have to deal with a strongly convex and nonsmooth objective function and have to rely on first-order methods. Under the assumption of Lipschitz-continuity, nonsmooth convex optimization problems can be solved at a rate of $\mathcal{O}\Big(\frac{1}{\sqrt{T}}\Big)$, where T is the iteration counter. For strongly convex problems this rate can be improved to $\mathcal{O}\Big(\frac{1}{T}\Big)$, see e.g. Lan (2020). We point out for the evaluation of a subgradient of the objective function in (40), it is necessary to compute the transpose of M^t. This explains why the manipulation of organizations is required to influence the network. Namely, they likely possess the knowledge of the network's structure in terms of M rather than the agents do.

The total complexity of the subproblems does not only depend on the rate of convergence but also on how efficient each iteration can be computed. Usually, there is a trade-off between achieving a better rate in terms of iterations and the numerical efficiency per iteration. An advantage of the inexact version is to counteract this trade-off, as it enables to stop the algorithm at an earlier stage. By applying the mirror descent with the relative entropy as Bregman divergence, the projection on the probability simplex is avoided, since the update steps are then given by a closed-form expression, see e.g. Beck and Teboulle (2003). For large networks, i. e. with large values of n, this is crucial, since in the Euclidean projection on the simplex Δ_n comes with the cost of $\mathcal{O}(n \log n)$, see Chen and Ye (2011). There is also a variant of the mirror descent for strongly convex functions, which achieves $\mathcal{O}\Big(\frac{1}{T}\Big)$ proposed by

Juditsky and Nemirovski (2011). However, the evaluation of each iteration step becomes more involved.

Remark 4 (Entropic mirror descent) We recall the entropic setup of the mirror descent method for minimizing a convex function $f : \Delta_n \to \mathbb{R}$ on the probability simplex with (sub-)gradients $f'(x)$ an the point x. The update step at iteration ℓ is then given by (Beck 2017):

$$x_{\ell+1}^{(i)} = \frac{x_\ell^{(i)} \cdot \exp\left(-\alpha_\ell \cdot f'(x_\ell)^{(i)}\right)}{\sum_{j=1}^n x_\ell^{(j)} \cdot \exp\left(-\alpha_\ell \cdot f'(x_\ell)^{(j)}\right)}, \quad j = 1, 2, \dots, n, \tag{41}$$

where α_ℓ is a suitable chosen stepsize. In practice, a dynamic adaptive stepsize is often chosen, while the fixed stepsize turns out to be useful for the complexity analysis, see for example Beck (2017). In order to control the level of inexactness in the alternating minimization scheme, we can therefore rely on the complexity analysis of the mirror descent method. More precisely, let the (sub-)gradients be bounded, i. e. $\|f'(x_\ell)\|_\infty \leq M_f$ for all $x \in \Delta_n$ and let the fixed stepsize $\alpha_\ell = \frac{\sqrt{2\log(n)}}{M_f\sqrt{L+1}}$, $\ell = 0, \dots, L$ be selected. Then, after L periods the optimality gap reads as

$$f_L^{\text{best}} - f^* \leq \frac{\sqrt{2\log(n)}M_f}{\sqrt{L+1}},$$

where f_L^{best} is the minimal function value so far (Beck 2017). This enables to determine the maximum number of iterations L for attaining a δ-inexact solution, i. e.

$$L > \frac{2\log(n) \cdot M_f^2}{\delta^2}. \tag{42}$$

The details of applying the mirror descent method to (40) are given in Sect. 6, where π_k is taken as the squared Euclidean distance. □

The first part of the objective function in (40) consists of a finite sum. If N is very large, the efficiency of first-order methods might suffer, since in any iteration N gradients must be evaluated. Therefore, it seems reasonable to apply a stochastic version of the mirror descent algorithm. However,the stochastic mirror descent for composite objectives does not linearize the second part of the objective function, see Duchi et al. (2010). Therefore, if π_k is not simple, the computation of an iteration step might be here too expensive.

Smoothing techniques for nonsmooth functions are commonly used in optimization for achieving the convergence of order $\mathcal{O}\left(\frac{1}{T}\right)$, see Nesterov (2005) and Beck (2017). There exist several approximations of the norm $\|\cdot\|_2$ with Lipschitz smooth gradients. E.g., it is possible to replace each summand of the first term of (40) by the following approximation:

$$\sqrt{\|v_i - M^t \cdot x_k\|_2^2 + \delta_k^2} - \delta_k, \quad k = 1, \ldots, K. \tag{43}$$

The function in (43) has $\frac{[\sigma_{\max}(M)]^{2t}}{\delta_k}$-smooth gradients, see Beck and Teboulle (2012). Note that large values of δ_k yield a very smooth function but provide a worse approximation. Furthermore, we stress that it is possible to examine the effect of the smoothing parameter on the convergence of the inexact alternating minimization method. Incorporating these smoothened versions enables to improve the efficiency even if the functions π_k remain nonsmooth. For simple π_k's the problem can be solved in $\mathcal{O}\left(\log \frac{1}{\epsilon}\right)$ iterations by a version of the accelerated gradient method, see e.g. Lan (2020). However, at each iteration N gradients have to be evaluated yielding a total of $\mathcal{O}\left(N \cdot \log \frac{1}{\epsilon}\right)$ gradient evaluations. Reducing these gradient computations can be achieved by applying the random primal–dual gradient method, which has been introduced by Lan and Zhou (2018). There, only one component of the sum is randomly selected and its gradient is used for the update step. Compared to the accelerated gradient version, this can save up to $\mathcal{O}(\sqrt{N})$ gradient evaluations, see Lan (2020).

Obviously, better properties of the payoff functions increase the possibilities to efficiently solve the subproblems (40). If the payoff functions have Lipschitz-smooth gradients, or can be smoothed, replacing the norm $\|\cdot\|_2$ by (43) enables to apply conditional gradient methods for solving the subproblems. Such algorithms are of the order $\mathcal{O}\left(\frac{1}{T}\right)$ and hence, an ϵ-solution can be found in $\mathcal{O}\left(\frac{1}{\epsilon}\right)$ iterations. From the numerical point of view, it is important to note that the conditional gradient method does not rely on the projection, see Jaggi (2013). This facilitates its application in our setting and lowers the per iteration cost for large networks. In fact, the alternating structure enables to deal with subproblems on the probability simplex. Minimizing a linear function on the simplex is straightforward, thus, a dominant factor of each iteration step is to compute the gradient. Recently, a conditional gradient sliding method has been introduced by Lan and Zhou (2016), where the number of calls of the first-order oracle can be reduced to $\mathcal{O}\left(\log \frac{1}{\epsilon}\right)$, while the number of iterations remains unchanged. We note that stochastic versions of the conditional gradient sliding are available.

The preceding discussions clearly suggest that the alternating minimization algorithm simplifies the computation of update steps to optimize objective functions of the form (15). Moreover, the possibility to compute inexact solutions of the subproblems turn out to be crucial for an efficient implementation. Popular choice models like probit must rely on approximate solutions of the choice probabilities and being able to stop an algorithm earlier automatically saves computational effort, like gradient evaluations.

6 Numerical examples

We examine the theoretical findings in Theorem 2 by means of numerical examples. Due to Theorem 2, we are allowed to solve the subproblems inexactly and still achieve convergence up to an error which can be controlled by the model parameters. Note that the bounds derived in Theorem 2 are based on a worst-case analysis. In practice, we might expect to see improved results. For all numerical tests, we select

$$\pi_k(M^t \cdot x_k) = \frac{1}{2}\left\|M^t \cdot x_k - c_k\right\|_2^2, \quad k = 1, \dots, K,$$

where c_k is the k-th organization's targeted network state. For each organization the subproblem consists of minimizing a nonsmooth function on the probability simplex, which is solved by the entropic mirror descent, see Remark 4.

Let us provide a simple test example of 2 agents, who choose between 4 credible organizations according to the multinomial logit model with parameters

$$\mu_1 = 8.2, \quad \mu_2 = 9.$$

The network is of size $n = 20$ and is block-diagonal. More precisely, the j-th of, say, 8 blocks has structure $\left(e \cdot e^T - I\right) \cdot \frac{1}{(n_j - 1)}$, where e is the vector of ones and I is the identity matrix of dimension n_j and size $n_j \times n_j$, respectively. We choose n_j, $j = 1, \dots 13$, with

$$\sum_{j=1}^{13} n_j = n, \quad 2 \le n_j \le 3.$$

There is only one period of interaction, i.e. $t = 1$, and the credibility values of the organizations are

$$\eta_1 = 0.95, \quad \eta_2 = 0.81, \quad \eta_3 = 1, \quad \eta_4 = 0.79.$$

The aspired states of the agents v_1, v_2, and v_3 are randomly generated. Note that in this example it holds:

$$\sigma_1 = \min_{1 \le k \le K} \frac{\tau_k}{\eta_k} \cdot \left[\sigma_{\min}(M)\right]^{2t} = \frac{1}{1} \cdot 0.5^2, \quad \sigma_2 = \min_{1 \le i \le N} \beta_i = 8.2,$$

and

$$L_1 = N^{\frac{1}{2}} \cdot \left[\sigma_{\max}(M)\right]^t = 2^{\frac{1}{2}} \cdot 1, \quad L_2 = 1.$$

Hence, Assumption 5 is satisfied, since

$$2 < \left(\frac{\frac{1}{1} \cdot 8.2}{2}\right)^{\frac{1}{2}} = 2.02485.$$

First, we produce iterates (P^*, X^*) due to the exact version of the alternating minimization algorithm. This is done by solving organizations' inner subproblems via entropic mirror descent with dynamic adaptive stepsize (Beck 2017), where the algorithm terminates if no sufficient progress is achieved. The latter is controlled by comparing consecutive iterates element-wise and by stopping if this element-wise difference is less than 10^{-10}. Due to the fixed point scheme of the inexact alternating minimization algorithm, an element-wise comparison with precision 10^{-9} is used as stopping criterion for the outer iterations. Second, we produce iterates $\left(\tilde{P}^*, \tilde{X}^*\right)$ due to the inexact version of the alternating minimization algorithm. For computing the inexact solutions we rely on the entropic mirror descent with fixed stepsize discussed in Remark 4. Therefore, a δ-level is chosen and the algorithm stops after a number of iterations determined by Eq. (42). Figure 1 shows that the gaps between exact solutions $P^\star, X^\star$ and inexact solutions denoted by $\tilde{P}^\star, \tilde{X}^\star$ are closing rapidly, at least as fast as proportionally to the square root of δ. This observation confirms our theoretical convergence results in Theorem 2.

However, the inexact algorithm yields a numerical advantage. In Table 1 the computational time of the exact method is compared to the computational time of different inexact versions of the alternating minimization scheme. Clearly, the computational time can be significantly reduced by applying inexact versions.

From now on, we ignore Assumption 5 and test the performance of the inexact in relation to the exact version by running numerical simulations. In order to illustrate the numerical performance, we control the number of inner iterations by choosing to iterate 10, 50, 100, 1000 times. All subproblems are solved by entropic mirror descent with dynamic adaptive stepsize. For that, we focus on agents choosing probabilities according to the nested logit model, implying that their updates can be written in closed form. There are 5 organizations to choose from, where the organizations 1 and 4 are in the first nest, organization 5 is in the third nest, organizations 2 and 3 are in the second nest. The nest parameters are chosen uniformly at random between 0.01 and 0.6. The aspired states as well as the targeted network states of the organizations are also randomly initialized. We set

$$\eta_1 = 1, \quad \eta_2 = 2, \quad \eta_3 = 3, \quad \eta_4 = 0.01, \quad \eta_5 = 2,$$

and the network structure is randomly generated. The results are comprised in Fig. 2. The computational time is significantly reduced, see Table 2. A higher number of agents increases the computational effort, which is summarized in Table 3, while the inexact versions yield close approximations at the latest from 100 maximum inner iterations, see Fig. 3.

The computation gets noticeably more involved when more periods of interaction take place. Setting $t = 2$ significantly increases the running time in order to compute the exact version (Table 4).

The difference between exact and inexact versions are shown in Fig. 4.

Let us increase the network size to $n = 100$ and randomly initialize a sparse network. As Table 5 shows, the computational effort for an exact version dramatically

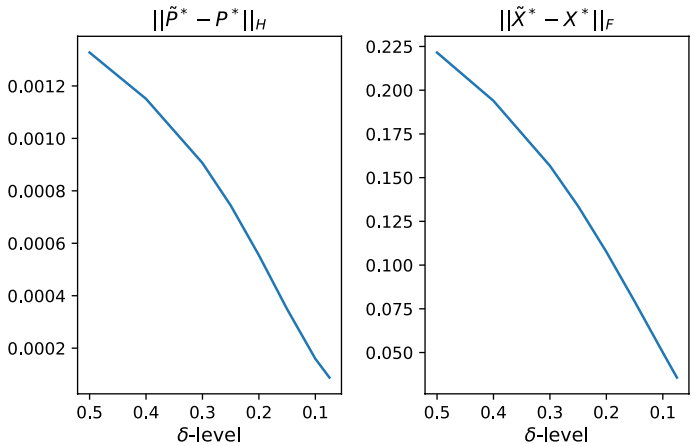

Fig. 1 Difference between the exact solutions $P^\star, X^\star$ and the inexact solutions $\tilde{P}^\star, \tilde{X}^\star$ dependent on the chosen δ-level of the inner optimization problems

Table 1 Computational time in seconds for different inexact versions

δ-level	Computational time (s)
Exact	977.2344
0.075	466.7031
0.1	134.5781
0.15	32.3906
0.3	6.5469
0.5	2.3906

increases. At the same time, the inexact versions can be solved much faster and yield reasonable approximations.

Overall, our numerical experiments suggest that the alternating minimization algorithm is rather robust if inexactly solving inner subproblems. This feature prevails if the number of organizations grows, there are more interactions, the network is ill-conditioned or sparse. Thus, the presented computational results support our theoretical findings in Theorem 2 and motivate the use of the inexact alternating minimization.

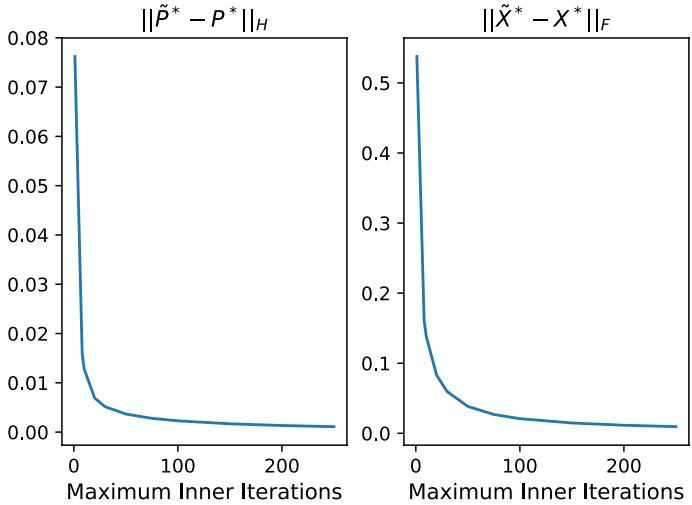

Fig. 2 Difference between exact and inexact solutions for $n = 50, N = 10, t = 1$

Table 2 Computational time for $n = 50, N = 10, t = 1$

Maximum Inner Iterations	Computational time (s)
Exact (1000000)	914.4688
1000	37.8125
100	4.2656
50	1.8438
10	0.4063

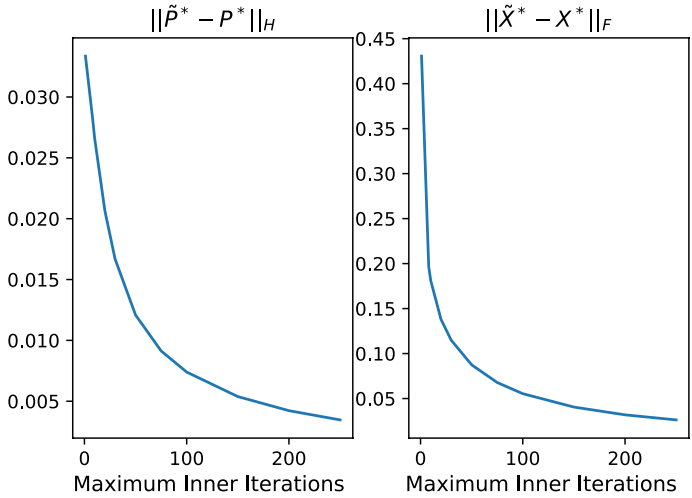

Fig. 3 Difference between exact and inexact solutions for $n = 50, N = 30, t = 1$

Table 3 Computational time for $n = 50, N = 30, t = 1$

Maximum inner iterations	Computational time (s)
Exact (1000000)	2272.2031
1000	80.5313
100	7.7031
50	3.7656
10	0.75

Table 4 Computational time: $n = 50, N = 30, t = 2$

Maximum inner iterations	Computational time (s)
Exact (1000000)	8676.9375
1000	53.4688
100	5.2188
50	2.79688
10	0.5313

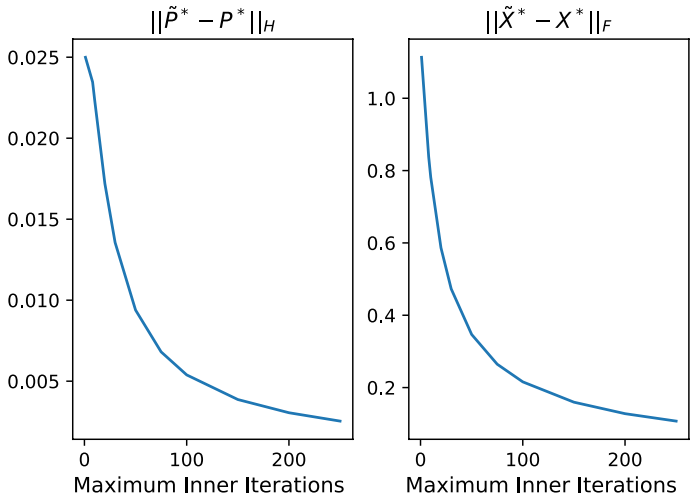

Fig. 4 Difference between exact and inexact solutions for $n = 50, N = 30, t = 2$

Table 5 Computational time and errors for $n = 100, N = 10, t = 1$, sparse network

Maximum inner iterations	Computational time (s)	$\|\tilde{P}^* - P^*\|_H$	$\|\tilde{X}^* - X^*\|_F$
Exact (1000000)	4799.1406	–	–
1000	133.625	0.000237	0.000236
100	13.1563	0.003663	0.026953
50	5.8594	0.006626	0.042717
10	1.4063	0.019262	0.113434

Acknowledgements The authors would like to thank the anonymous referees for their precise and constructive remarks which considerably improved the quality of the manuscript.

Funding Open Access funding enabled and organized by Projekt DEAL. Not applicable.

Availability of data and material Not applicable.

Code availability Not applicable.

Declarations

Conflict of interest Not applicable.

Ethical approval Not applicable.

Consent to participate Not applicable.

Consent for publication Not applicable.

References

Acemoglu D, Ozdaglar A (2011) Opinion dynamics and learning in social networks. Dyn Games Appl 1(1):3–49

Anderson SP, Palma AD, Thisse LF (1992) Discrete choice theory of product differentiation. MIT press, Cambridge

Beck A (2015) On the convergence of alternating minimization for convex programming with applications to iteratively reweighted least squares and decomposition schemes. SIAM J Optim 25(1):185–209

Beck A (2017) First-order methods in optimization. SIAM, Philadelphia, PA

Beck A, Teboulle M (2003) Mirror descent and nonlinear projected subgradient methods for convex optimization. Oper Res Lett 31(3):167–175

Beck A, Teboulle M (2012) Smoothing and first order methods: a unified framework. SIAM J Optim 22(2):557–580

Börsch-Supan A, Hajivassiliou VA (1993) Smooth unbiased multivariate probability simulators for maximum likelihood estimation of limited dependent variable models. J Econ 58(3):347–368

Chen Y, Ye X (2011) Projection onto a simplex. arXiv preprint arXiv:1101.6081

Connors RD, Hess S, Daly A (2014) Analytic approximations for computing probit choice probabilities. Transp A Trans Sci 10(2):119–139

DeGroot MH (1974) Reaching a consensus. J Am Stat Assoc 69(345):118–121

Duchi J, Shalev-Shwartz S, Singer Y, Tewari A (2010) Composite objective mirror descent. In: Proceedings of the 23rd annual conference on learning theory. Omnipress. pp 14–26

Förster M, Mauleon A, Vannetelbosch VJ (2016) Trust and manipulation in social networks. Netw Sci 4(2):216–243

Fosgerau M, Melo E, De Palma A, Shum M (2020) Discrete choice and rational inattention: a general equivalence result. Int Econ Rev 61(4):1569–1589
Gagniuc PA (2017) Markov chains: from theory to implementation and experimentation. Wiley, Hoboken
Grippof L, Sciandrone M (1999) Globally convergent block-coordinate techniques for unconstrained optimization. Optim Methods Softw 10(4):587–637
Jaggi M (2013) Revisiting Frank-Wolfe: projection-free sparse convex optimization. In: International conference on machine learning. PMLR. pp 427–435
Juditsky A, Nemirovski A (2011) First order methods for nonsmooth convex large-scale optimization, I: general purpose methods. Optim Mach Learn 30(9):121–148
Lan G (2020) First-order and stochastic optimization methods for machine learning. Springer Nature, Heidelberg
Lan G, Zhou Y (2016) Conditional gradient sliding for convex optimization. SIAM J Optim 26(2):1379–1409
Lan G, Zhou Y (2018) An optimal randomized incremental gradient method. Math Program 171(1):167–215
Lindqvist B (1977) How fast does a Markov chain forget the initial state? Adecision theoretical approach. Scand J Stat 4(4):145–152.
Luo ZQ, Tseng P (1993) Error bounds and convergence analysis of feasible descent methods: a general approach. Ann Oper Res 46(1):157–178
McFadden D (1978) Modeling the choice of residential location. Transp Res Rec 673:72–77
McFadden D, Train K (2000) Mixed MNL models for discrete response. J Appl Economet 15(5):447–470
Müller D, Nesterov Y, Shikhman V (2021) Discrete choice prox-functions on the simplex. Math Oper Res 47(1):485–507. https://doi.org/10.1287/moor.2021.1136
Müller D, Nesterov Y, Shikhman V (2021) Dynamic pricing under nested logit demand. J Pure Appl Funct Anal 6(6):1435–1451
Nesterov Y (2005) Smooth minimization of non-smooth functions. Math Program 103(1):127–152
Nesterov Y (2018) Lectures on convex optimization, vol 137. Springer, New York
Nesterov Y (2020) Soft clustering by convex electoral model. Soft Comput 24(23):17609–17620
Page L, Brin S, Motwani R, Winograd T (1999) The pagerank citation ranking: bringing order to the web. Tech. rep, Stanford InfoLab
Pu Y, Zeilinger MN, Jones CN (2014) Inexact fast alternating minimization algorithm for distributed model predictive control. In: 53rd IEEE conference on decision and control. pp 5915–5921
Stonyakin FS, Dvinskikh D, Dvurechensky P, Kroshnin A, Kuznetsova O, Agafonov A, Gasnikov A, Tyurin A, Uribe CA, Pasechnyuk D, Artamonov S (2019) Gradient methods for problems with inexact model of the objective. In: International conference on mathematical optimization theory and operations research. Springer. pp 97–114
Thurstone L (1927) A law of comparative judgment. Psychol Rev 34(4):273
Train KE (2009) Discrete choice methods with simulation. Cambridge University Press, Cambridge
Wen CH, Koppelman FS (2001) The generalized nested logit model. Trans Res Part B Methodol 35(7):627–641
Yang J, Xu Y, Chen CS (1997) Human action learning via hidden Markov model. IEEE Trans Syst Man Cybern-Part A Syst Humans 27(1):34–44