ETH zürich

Demand-Side Management Problem

Consider a Demand-Side Management (DSM) game in which N energy consumers (players), labelled by $i \in \mathcal{I} := \{1, 2, \dots, N\}$, aim at minimizing their local energy consumption costs. Each player shall choose its energy consumption profile over the following 24 hours, i.e., $x_i = (x_i^1, \dots, x_i^{24}) \in \mathbb{R}^{24}_{\geq 0}$. The energy consumption at each hour h has an upper limit $\bar{x}_i^h \in \mathbb{R}_{\geq 0}$, imposed, for example, by the network operator and/or by the player's consumption preferences, i.e.,

$$x_i^h \in \left[0,\ \bar{x}_i^h\right], \quad \forall h \in \{1, \dots, 24\}.$$

Moreover, the cumulative consumption over the 24 hours must satisfy the player's (cumulative) nominal energy need:

$$\sum_{h=1}^{24} x_i^h = \sum_{h=1}^{24} e_i^h,$$

where the parameter $e_i^h \in \mathbb{R}_{>0}$ represents the nominal energy consumption at time h, namely, the amount of energy player i would normally consume at time h if it was not participating in the DSM game. The goal of each player $i \in \mathcal{I}$ is to choose a feasible energy consumption profile x_i that minimizes the cumulative (over the 24 hours) energy bill, i.e.,

$$J_i(x) = \varphi_i \left(\sum_{j \in \mathcal{I}} x_j \right)^\top x_i,$$

where $x = (x_1, \dots, x_N)$ is the stacked vector of all player's consumption profiles, and $\varphi_i : \mathbb{R}^{24} \to \mathbb{R}^{24}$ is a personalized energy price function defined as

$$\varphi_i(y) = p_{i,1} y + p_{i,2} \mathbf{1}_{24},$$

with $p_{i,1} \in \mathbb{R}$ pricing the aggregate energy consumption, $p_{i,2} \in \mathbb{R}$ representing a baseline energy price, and $\mathbf{1}_{24}$ denoting a column vector of length 24 with all entries 1.

Graziano Davide

DSM Problem

Task A

Find the ranges of the parameters $p_{i,1}$ and $p_{i,2}$ for which the DSM game is a convex game. Is the Nash equilibrium unique for all the price parameters in such ranges? If yes, prove it formally. If not, construct a counter-example.

Solution

Let us verify, one by one, the conditions for which the problem can be rigorously formulated as a convex game:

Formulation of Players

- there are N players, who are the energy consumers $P_i \ \mid \ i \in I = \{1, \ldots, N\}$;

Formulation of Strategy Profiles

- each player can choose a strategy $x_i = (x_i^1, \ldots, x_i^{24}) \in \mathbb{R}^{24}$, that is their energy consumption profile over the following 24 hours;

Non-Emptiness, Compactness and Convexity of feasible strategy spaces

- $K_i \subset \mathbb{R}^{24}$ is the *nominal* **strategy space** of player P_i. For each player P_i, each element x_i^h of a strategy profile $x_i \in K_i$ is:
  - lower bounded, because it must be nonnegative: $x_i^h \geq 0$;
  - upper bounded by the energy consumption limit at hour h: $x_i^h \leq \bar{x}_i{}^h$.

  In mathematical terms, it means that each dimension x_i^h of the nominal space K_i is closed and bounded:

$$x_i^h \in [0, \bar{x}_i^h]$$

  Now, since the above holds for each dimension, the nominal space K_i can be enclosed in a finite region of $\mathbb{R}^{24}$:

$$K_i = [0, \bar{x}_i^1] \times [0, \bar{x}_i^2] \times \ \ldots \ \times [0, \bar{x}_i^{24}]$$

  In other words, K_i is a hyper-parallelepiped, closed along each dimension. Therefore, for each player, their nominal strategy space is non-empty, closed and bounded, i.e. **compact**. Moreover, K_i is also **convex**. In fact, for any two strategies in the nominal strategy space K_i of player P_i, the segment connecting the two strategies completely lies within K_i:

$$\forall \ {}_1x_i\,,\ {}_2x_i \in K_i \quad \& \quad \forall\ t \in [0,1] \quad \rightarrow \quad t \cdot {}_1x_i + (1-t) \cdot {}_2x_i \ \in \ K_i$$

  Proof:

  Let ${}_3x_i(t) = t \cdot {}_1x_i + (1-t) \cdot {}_2x_i$.
  Now, ${}_1x_i$ (and analogously ${}_2x_i$) can be expressed as:

$$_1x_i \;=\; _1x_i^1\cdot\mathbf{e}_1 \;+\; \dots \;+\; _1x_i^{24}\cdot\mathbf{e}_{24} \;=\sum_{k=1}^{24} {_1x_i^k}\cdot\mathbf{e}_k$$

where $\mathbf{e}_k$ is the k-th element of the cartesian basis of $\mathbb{R}^{24}$. It follows that:

$$_3x_i(t) = \sum_{k=1}^{24} \left[t\cdot{_1x_i^k} + (1-t)\cdot{_2x_i^k}\right]\mathbf{e}_k$$

Now: $_3x_i(t) \in K_i \iff 0 \le t\cdot{_1x_i^k} + (1-t)\cdot{_2x_i^k} \le \bar{x}_i^k \quad \forall k \in \{1 , \dots , 24\}$.

From the hypotheses:

- $_1x_i \in K_i \iff 0 \le {_1x_i^k} \le \bar{x}_i^k \quad \forall k \in \{1 , \dots , 24\}$;
- $_2x_i \in K_i \iff 0 \le {_2x_i^k} \le \bar{x}_i^k \quad \forall k \in \{1 , \dots , 24\}$;
- $t \in [0,1]$.

As a result, the following chain of consequences can be inferred:

- $_3x_i^k(t) = t\cdot{_1x_i^k} + (1-t)\cdot{_2x_i^k} \;\mid\; t \in [0,1]$ is the segment connecting the extremes $_1x_i^k$ and $_2x_i^k$, along the k-th dimension of K_i.
- since the extremes of the segment belong to the nominal space K_i, then $_3x_i^k(t) \in K_i \;\; \forall\, t \in [0,1]$, because, along the k-th dimension, K_i is simply connected.
- the above hold for each dimension k.

Therefore, the condition for which $_3x_i(t) \in K_i$ is verified. Since $_1x_i^k$ and $_2x_i^k$ are generic strategies in K_i , the latter is convex.

However, the nominal strategy space of player i $\left(K_i\right)$ is subject to the following constraint:

$$\sum_{k=1}^{24} x_i^k = \sum_{k=1}^{24} e_i^k$$

Translating the above, among the nominal strategies in K_i , the *feasible* ones must must lie on the hyperplane defined by the above equation. Now, let us call the set of strategies x_i that lie on the hyperplane as follows:

$$H_i = \left\{ x_i \in \mathbb{R}^{24} \;\middle|\; \sum_{k=1}^{24} x_i^k = \sum_{k=1}^{24} e_i^k = d_i \right\}$$

A hyperplane is a convex set.

Proof:

Let $a, b \in H_i$ be two generic points on the hyperplane, that is, that they satisfy the equation of the hyperplane: $\sum_{k=1}^{24} a^k = d_i$ and so for b. Consider now the segment:

$$c(t) = t\cdot a + (1-t)b \quad \forall t \in [0,1]$$

$$c(t) = t \cdot a + (1-t)b = t \cdot \sum_{k=1}^{24} a^k \ + \ (1-t)\sum_{k=1}^{24} b^k \ = t \cdot d_i + (1-t)\, d_i = d_i$$

The segment satisfies the equation of the hyperplane. Since a and b are generic points on the hyperplane, H_i is a convex set.

Finally, the **feasible strategy space** $\tilde{K}_i$ of player i is

$$\tilde{K}_i = \left(K_i \cap H_i\right) \subset \mathbb{R}^{24}$$

Assuming that the intersection is non-empty, $\tilde{K}_i$ is also a compact and convex set.

Quick Proof:

- $\tilde{K}_i$ is bounded because it is the intersection of the hyperplane H with the bounded set K_i : the intersection of a set which is enclosed in a finite region of space with any other set can always be enclosed in a finite region of space, which is the definition of boundedness.
- $\tilde{K}_i$ is also closed, by topology of the problem. In fact, a compact hyper-parallelepiped $\left(K_i\right)$ is a finite region of its hyper-space $\left(\mathbb{R}^{24}\right)$ enclosed by compact intersections of planes. Now, the intersection a hyperplane and a compact region of another hyperplane can be:
  - the empty set, if the hyperplanes are parallel and non-coincident. However, this is not the case, since we have assumed a non-empty intersection: $\tilde{K}_i = \left(K_i \cap H_i\right) \neq \{\}$
  - the hyperplanes themselves, when coincident: if this is the case, then $\tilde{K}_i$ is compact, and thus closed, because the compact hyper-parallelepiped is, again, a region enclosed by compact intersections of hyper-planes;
  - a simply connected segment: if this is the case, then $\tilde{K}_i$ is compact, because it is a plane region of the hyper-space delimited by simply connected segments, including the segments.
- $\tilde{K}_i$ is convex because it is the intersection of two convex sets. In fact, every vector

$$x_i \in \tilde{K}_i \quad \Longleftrightarrow \quad x_i \in K_i \ \wedge \ x_i \in H$$

  and the proof of the convexity of $\tilde{K}_i$ becomes trivial. In fact, given any two points in the feasible strategy space $\tilde{K}_i$, they also belong to K_i and H. Since the latter two are convex, the segment connecting the two points belongs to both and, thus, to $\tilde{K}_i$, satisfying the definition of convexity.

REMARK: for easier notation, from now on, I will simply call K_i the feasible action space of player i (without the tilde).

Focus: non-emptiness of feasible strategy spaces

Topologically speaking, if K_i is an empty set for some player i, then it is not possible to define its compactness: in fact, the empty set is both closed and open. By the point of view of the game, if the feasible strategy space K_i of some player i is empty, then the characterisation of the game must be reviewed, because the player has no available strategy. Now, as seen above:

$$K_i = \left\{ \left[0,\, \bar{x}_i^1\right] \times \left[0,\, \bar{x}_i^2\right] \times\ \dots\ \times \left[0,\, \bar{x}_i^{24}\right] \right\} \cap \left\{ x_i \in \mathbb{R}^{24} \;\middle|\; \sum_{k=1}^{24} x_i^k = \sum_{k=1}^{24} e_i^k = d_i \right\}$$

By definition, K_i is non-empty if it contains at least one element. Moreover, the elements of K_i must satisfy the equations of both the hyper-parallelepiped and the hyperplane. One possible way to check the non emptiness of K_i is the following:

- consider the equation of the hyper-plane:

$$\sum_{k=1}^{24} x_i^k = \sum_{k=1}^{24} e_i^k = d_i \qquad \Longleftrightarrow \qquad \mathbf{1}^T x_i = d_i$$

- along each dimension (hour), the vector (strategy profile) x_i must have nonnegative entries. Therefore, we must check that:

$$0 \le x_i^k \quad \forall k \qquad \underset{(\not\Longleftarrow)}{\Longrightarrow} \qquad 0 \le \mathbf{1}^T x_i = d_i \tag{1}$$

- along each dimension (hour), the vector (strategy profile) x_i is upper bounded by $\bar{x}_i^k$ (hourly consumption limit):

$$x_i^k \le \bar{x}_i^k \quad \forall k \qquad \underset{(\not\Longleftarrow)}{\Longrightarrow} \qquad d_i = \mathbf{1}^T x_i \le \mathbf{1}^T \bar{x}_i \tag{2}$$

  where $\bar{x}_i$ is the vector of the stacked upper bounds.

To summarise, the conditions to be checked are given by: $0 \le d_i = \sum_{k=1}^{24} e_i^k \le \mathbf{1}^T \bar{x}_i$.

***R*emark**: the following is a rather delicate part of the proof.
The above implications (1 and 2) are sufficient, but not necessary. Mathematically, this is due to the shift from a vectorial equation (24 scalar equations) to a scalar equation, which obviously decreases the set of constraints on the variables. It is easy to find a counterexample, for instance to the first implication:

$$\mathbf{1}^T \begin{bmatrix} -1 \\ 1 \\ \vdots \\ 1 \end{bmatrix} = 22 \ge 0 \qquad \not\Longrightarrow \qquad x_i^k \ge 0 \quad \forall k$$

Therefore, if the conditions on d_i are verified, it is <u>not</u> guaranteed that the inequalities defining the hyper-parallelepiped are satisfied as well:

$$0 \le d_i = \sum_{k=1}^{24} e_i^k \le \mathbf{1}^T \bar{x}_i \qquad \not\Longrightarrow \qquad 0 \le x_i^k \le \bar{x}_i^k \quad \forall k$$

However, we are attempting to prove the non-emptiness of the feasible strategy spaces K_i. This means that it is sufficient to find one strategy that satisfies both the inequalities and the hyper-plane equation. Now, assuming that the above conditions on d_i are satisfied, we can always construct a feasible strategy $x_i \in K_i$ as follows:

- all the entries are initialised to zero $\left(x_i^k = 0 \quad \forall k\right)$;
- for $k = \{1, \ldots, 24\}$:

1. compute: $d_i - \sum_{h=1}^{24} x_i^h$
2. if the above difference is: $0 \leq d_i - \sum_{h=1}^{24} x_i^h \leq \bar{x}_i^k \implies x_i^k = d_i - \sum_{h=1}^{24} x_i^h$

   and stop the cycle.
3. otherwise, set: $x_i^k = \bar{x}_i^k$ and go to the next iteration.

- since $0 \leq d_i \leq \mathbf{1}^T \bar{x}_i$ for hypothesis, then the loop will end with a solution x_i that satisfies the nonnegativity constraint and all the upper bounds, therefore it is a feasible strategy $x_i \in K_i$ and we can infer the non-emptiness of K_i.

> **Comments**: let us consider the problem intuitively: player i can satisfy their need, and their strategy space is non-empty, only if the sum of the hourly consumption limits imposed on player i is higher than or equal to their energy consumption need. As a result, the above discussion might seem trivial, even redundant. However, I find that it is not the case: notice, again, that we have a necessary, not a sufficient condition. The reason why the implication still holds (backward) lies in the algorithm illustrated above, which is the key-element that proves the non-emptiness of the strategy space.

To summarise, in this particular problem setup, once verified that $0 \leq d_i \leq \mathbf{1}^T \bar{x}_i$, we can infer that the feasible strategy space of player i $\left(K_i\right)$ is non-empty, compact and convex.

Global Feasible Strategy Space

- let K be the **global feasible strategy space**: $K = K_1 \times K_2 \times \ldots \times K_N \subset \mathbb{R}^{24 \cdot N}$. Since every K_i is compact and convex, their cartesian product is also compact and convex. In fact, K is a hyper-parallelepiped in $\mathbb{R}^{24N}$ subject to the feasibility constraints $\left(H_i\right)$: the proof is precisely the one used for K_i, with the only exception of a higher dimensionality.

Continuity of the Cost Functions

- the outcome of the game for player i is described by the following scalar field:

$$J_i : K \subset \mathbb{R}^{24N} \;\rightarrow\; \mathbb{R} \quad : \quad J_i(x) = \varphi_i \left(\sum_{j=1}^{N} x_j \right)^T x_i$$

where the personalised energy price function φ_i is a vector field defined as follows:

$$\varphi_i : \mathbb{R}^{24} \;\rightarrow\; \mathbb{R}^{24} \quad : \quad \varphi_i \left(\sum_{j=1}^{N} x_j \right) = p_{i,1} \sum_{j=1}^{N} x_j \;+\; p_{i,2}\mathbf{1} \;=\; \begin{bmatrix} \varphi_i^1 \left(\sum_{j=1}^{N} x_j \right) \\ \vdots \\ \varphi_i^{24} \left(\sum_{j=1}^{N} x_j \right) \end{bmatrix}$$

Now, a vector field is continuous if and only if each function is continuous in the argument of the field. Let us verify the latter condition, by considering the k-th function of the field:

$$\varphi_i^k \left(\sum_{j=1}^{N} x_j \right) \;=\; p_{i,1} \sum_{j=1}^{N} x_j^k \;+\; p_{i,2}$$

φ_i^k is linear in its argument, thus it is continuous, and so for each function φ_i^k. Therefore, φ_i is continuous in its argument, for each player i. We can finally infer the continuity of J_i in its argument, because it is a product between two continuous functions: φ_i and x_i. This holds $\forall i$.

Convexity of the Cost Functions for fixed adversarial strategies

- the last condition for the problem to be formulated as a convex game is the convexity of the cost functions J_i in player i's strategy profile (x_i) for fixed adversarial strategies $(\tilde{x}_{-i})$. We can resort to the following theorem:

a twice differentiable-function is convex if and only if its Hessian is positive semi-definite

J_i is quadratic in its variables, therefore it is also twice continuously-differentiable. Let us compute the Hessian of J_i for fixed adversarial strategies $(\tilde{x}_{-i})$:

$$J_i(x_i, \tilde{x}_{-i}) = \varphi_i \left(\sum_{j=1}^{N} x_j \right)^T_{x_{-i}=\tilde{x}_{-i}} x_i \;=\; \left[p_{i,1} \sum_{j=1}^{N} x_j \;+\; p_{i,2}\mathbf{1} \right]^T_{x_{-i}=\tilde{x}_{-i}} x_i$$

$$\nabla_{x_i} J_i = \begin{bmatrix} \frac{\partial J_i}{\partial x_i^1} \\ \vdots \\ \frac{\partial J_i}{\partial x_i^{24}} \end{bmatrix} = \begin{bmatrix} p_{i,1} \sum_{\substack{j=1 \\ j \neq i}}^{N} \tilde{x}_j^1 + p_{i,2} + 2p_{i,1} x_i^1 \\ \vdots \\ p_{i,1} \sum_{\substack{j=1 \\ j \neq i}}^{N} \tilde{x}_j^{24} + p_{i,2} + 2p_{i,1} x_i^{24} \end{bmatrix} = p_{i,1} \sum_{\substack{j=1 \\ j \neq i}}^{N} \tilde{x}_j + p_{i,2}\mathbf{1} + 2p_{i,1} x_i$$

$$H\left(J_i\right) = \nabla_{x_i}\left(\nabla_{x_i} J_i\right) = \begin{bmatrix} \frac{\partial^2 J_i}{\left(\partial x_i^1\right)^2} & \frac{\partial^2 J_i}{\partial x_i^1 \partial x_i^2} & \dots & \frac{\partial^2 J_i}{\partial x_i^1 \partial x_i^{24}} \\ & \vdots & & \\ \frac{\partial^2 J_i}{\partial x_i^{24} \partial x_i^1} & \frac{\partial^2 J_i}{\partial x_i^{24} \partial x_i^2} & \dots & \frac{\partial^2 J_i}{\left(\partial x_i^{24}\right)^2} \end{bmatrix} = \begin{bmatrix} 2p_{i,1} & 0 & \dots & 0 \\ 0 & 2p_{i,1} & \dots & 0 \\ & & \ddots & \\ 0 & \dots & 0 & 2p_{i,1} \end{bmatrix} = 2p_{i,1}\mathbf{1}^{24\times 24}$$

where $\mathbf{1}^{24\times 24}$ is the identity matrix of size 24.
Now, H is semi-positive definite if and only if :

$$v^T H\left(J_i\right) v \geq 0 \quad \forall v \in \mathbb{R}^{24} \setminus \{0\} \quad \Longrightarrow \quad \sum_{j=1}^{24} 2p_{i,1}(v^j)^2 \geq 0$$

Which translates to: $p_{i,1} \geq 0 \quad \forall i$ (in fact, the result holds for all players).

To summarise, *the problem can be formulated as a convex game if and only if* :

$$p_{i,1} \in \mathbb{R}_{\geq 0} \quad \wedge \quad p_{i,2} \in \mathbb{R} \quad \forall i$$

Under the conditions verified so far, a Nash Equilibrium is guaranteed to exist (I will be more specific in task 1.c).

Assessment of the Existence and Uniqueness of the Nash Equilibrium

Let us cover the last part of this exercise, namely let us assess whether the game admits a unique or multiple Nash equilibria.
We can try to resort to the following result:

Consider an N-player convex game and the game map F defined as

$$F : \mathbb{R}^{24N} \to \mathbb{R}^{24N} \quad \Big| \quad F(x) = \begin{bmatrix} \nabla_{x_1} J_1 \\ \vdots \\ \nabla_{x_N} J_N \end{bmatrix}$$

If F is strictly monotone, then the Nash equilibrium exists and it is unique.

As showed above, the game is convex when $p_{i,1} \geq 0 \ \ \forall i$. Let us therefore assess if F is strictly monotone, which, for a vector field, translates to assessing whether the Jacobian of F is strictly positive definite for all $x \in K$.
The Jacobian of F is defined as follows:

$$\mathbf{J}F : \mathbb{R}^{24N} \to \mathbb{R}^{24N \times 24N} \quad \Big| \quad \mathbf{J}F(x) = \begin{bmatrix} \nabla_x \left(\nabla_{x_1} J_1 \right) \\ \vdots \\ \nabla_x \left(\nabla_{x_N} J_N \right) \end{bmatrix}$$

Now, from above:

$$\nabla_{x_i} J_i(x) = \ p_{i,1} \sum_{\substack{j=1 \\ j \neq i}}^{N} x_j \ + \ p_{i,2}\mathbf{1} \ + \ 2p_{i,1}x_i$$

Therefore, the computation of the Jacobian $\mathbf{J}F(x)$ yields:

$$\mathbf{J}F(x) = \begin{bmatrix} 2 \cdot p_{1,1}\mathbf{1}^{24\times24} & p_{1,1}\mathbf{1}^{24\times24} & p_{1,1}\mathbf{1}^{24\times24} & \dots & p_{1,1}\mathbf{1}^{24\times24} \\ p_{2,1}\mathbf{1}^{24\times24} & 2 \cdot p_{2,1}\mathbf{1}^{24\times24} & p_{2,1}\mathbf{1}^{24\times24} & \dots & p_{2,1}\mathbf{1}^{24\times24} \\ & & \vdots & & \\ p_{N,1}\mathbf{1}^{24\times24} & p_{N,1}\mathbf{1}^{24\times24} & p_{N,1}\mathbf{1}^{24\times24} & \dots & 2 \cdot p_{N,1}\mathbf{1}^{24\times24} \end{bmatrix}$$

$$\mathbf{J}F(x) = \begin{bmatrix} p_{1,1} \left[\ 2 \cdot \mathbf{1}^{24\times24} \quad \mathbf{1}^{24\times24} \quad \mathbf{1}^{24\times24} \quad \dots \quad \mathbf{1}^{24\times24} \ \right] \\ p_{2,1} \left[\ \mathbf{1}^{24\times24} \quad 2 \cdot \mathbf{1}^{24\times24} \quad \mathbf{1}^{24\times24} \quad \dots \quad \mathbf{1}^{24\times24} \ \right] \\ \vdots \\ p_{N,1} \left[\ \mathbf{1}^{24\times24} \quad \mathbf{1}^{24\times24} \quad \mathbf{1}^{24\times24} \quad \dots \quad 2 \cdot \mathbf{1}^{24\times24} \ \right] \end{bmatrix}$$

where, again, $\mathbf{1}^{24\times24}$ is the identity matrix of size 24.
By definition, the Jacobian of the game map is strictly positive definite when

$$v^T \mathbf{J}F(x) v > 0 \quad \forall v \in \mathbb{R}^{24N} \setminus \{0\}$$

Performing the calculations:

$$v^T \mathbf{J}F(x) v \ = \ \left[\ x_1^T \ \dots \ x_N^T \ \right] \begin{bmatrix} p_{1,1} \left[\ 2 \cdot \mathbf{1}^{24\times24} \quad \mathbf{1}^{24\times24} \quad \mathbf{1}^{24\times24} \quad \dots \quad \mathbf{1}^{24\times24} \ \right] \\ p_{2,1} \left[\ \mathbf{1}^{24\times24} \quad 2 \cdot \mathbf{1}^{24\times24} \quad \mathbf{1}^{24\times24} \quad \dots \quad \mathbf{1}^{24\times24} \ \right] \\ \vdots \\ p_{N,1} \left[\ \mathbf{1}^{24\times24} \quad \mathbf{1}^{24\times24} \quad \mathbf{1}^{24\times24} \quad \dots \quad 2 \cdot \mathbf{1}^{24\times24} \ \right] \end{bmatrix} \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} =$$

$$= \begin{bmatrix} 2p_{1,1} \cdot x_1 + \sum_{j=2}^{N} p_{j,1} \cdot x_j \\ \vdots \\ 2p_{N,1} \cdot x_N + \sum_{j=1}^{N-1} p_{j,1} \cdot x_j \end{bmatrix}^T \begin{bmatrix} x_1 \\ \vdots \\ x_N \end{bmatrix} =$$

$$= \sum_{i=1}^{N} \left(2p_{i,1} \cdot x_i^T x_i \ + \ \sum_{\substack{j=1 \\ j \neq i}}^{N} p_{j,1} \ x_j^T x_i \right) > 0$$

Remember that the condition that allows the formulation of the problem as a convex game is: $p_{i,1} \geq 0 \ \ \forall i$
Instead, determining the range of parameters $p_{i,1}$ that guarantees $\mathbf{J}F(x)$ to be strictly positive definite is a more difficult task. However, we can observe some facts:

- !! when $p_{i,1} = 0 \ \ \forall i$, then the above strict inequality does not hold, regardless of the strategy profiles. Therefore, we can already infer that not all values of the price parameters are admissible for the uniqueness of the NE, with respect to the ranges that allow to characterise the DSM game as a convex game. !!
- a sufficient condition for the Jacobian to be strictly positive definite is the following: $p_{i,1} > 0 \ \ \forall i$. This is because all strategy profiles are lower-bounded: $x_i^h \geq 0 \quad \Longrightarrow \quad x_j^T x_i \geq 0 \ \ \forall i, j$.

Task B

Find some (non-trivial) ranges of the parameters $p_{i,1}$ and $p_{i,2}$ for which the DSM game is an (exact) potential game. Give the expression of the (exact) potential function.

Preliminary comments: I will give an energetic interpretation of the problem. In fact, intuitively, we can regard the incentive of players to change their strategy as 'forces' driving their decisions towards static equilibria. The reasons behind this modelling choice are essentially two:

- we can rigorously derive the potential of the game, verifying that the latter exists;
- we can give a 'physical' interpretation of the conditions on the parameters $p_{i,1}$ and $p_{i,2}$ that allow to characterise the DSM game as a potential game.

Solution

A game is potential if it admits a potential function $\phi : \mathbb{R}^{24N} \to \mathbb{R}$ such that:

$$J_i(\tilde{x}_i, x_{-i}) - J_i(\hat{x}_i, x_{-i}) = \phi(\tilde{x}_i, x_{-i}) - \phi(\hat{x}_i, x_{-i}) \qquad \forall\, \tilde{x}_i, \hat{x}_i \in K_i \;\wedge\; \forall i \in \{1, \ldots, N\}$$

Now, by a physical point of view, a system admits a potential scalar field if and only if all the forces acting on the system are conservative. One possible definition of conservative force field is the following: the work of the force field along any path is not a function of the path, but it depends on the final and initial conditions only:

$$\int_\Gamma F(\gamma)d\gamma = \phi_{\text{end point}} - \phi_{\text{initial point}}$$

where Γ is any path in the physical domain of the system, in our case the space of the players' strategies $K \subset \mathbb{R}^{24N}$.
At this point, the energetic interpretation of the problem can already be sketched: the force field acting on the system, i.e. on the players, is related to the cost. This is clear by bearing in mind the definition of potential of a game stated above.
Intuitively, the higher the cost, the higher the incentive of a player to change their strategy, that is, the more intense the force acting on the player. As a result, we can measure the intensity of the force field acting on a player as the local gradient of the player's cost, and so for all players. Therefore, the global force field, i.e. the generalised field acting on the whole system, is the game map:

$$F : \mathbb{R}^{24N} \to \mathbb{R}^{24N} \quad \Big| \quad F(x) = \begin{bmatrix} \nabla_{x_1} J_1 \\ \vdots \\ \nabla_{x_N} J_N \end{bmatrix}$$

Now, let us consider the equations of dynamic equilibrium of the system. Specifically, consider Lagrange equations:

$$\frac{d}{dt}\left(\nabla_{\dot{x}} E_k\right) - \nabla_x E_k + \nabla_x V + \nabla_{\dot{x}} D = Q_x = \frac{\partial \delta \mathscr{L}}{\partial \delta x}$$

where $E_k = E_k(x, \dot{x})$ is the kinetic energy of the system, $V = V(x)$ is the potential, $D = D(x, \dot{x})$ is the dissipative function and $Q_x = Q_x(x, \dot{x})$ is the Lagrangian of the system.

However, the system has no dynamics, i.e. we are evaluating it in a static configuration. By a physical point of view, this is due to the fact that players have no inertia and there occurs no dissipation due to their choices. Therefore:

$$\frac{d}{dt}\left(\cancel{\nabla_x E_k}\right) - \cancel{\nabla_x E_k} + \nabla_x V + \cancel{\nabla_x D} = Q_x(x, \dot{x} = 0) = \frac{\partial \delta \mathscr{L}}{\partial \delta x}\Big|_{\dot{x}=0}$$

Now, let us call the potential $V(x) = \phi(x)$ (it is just notation). Moreover, notice that the game map, i.e. the Lagrangian, has a static characterisation, by that meaning that it depends on the players' strategies x but not on their time derivatives $\dot{x}$. In other words, the game map already satisfies the static condition $Q_x(x, \dot{x} = 0)$ (naturally, because this is a static game and thus the forces are static in turn).
The equation of static equilibrium is therefore:

$$\nabla_x \phi(x) = \begin{bmatrix} \nabla_{x_1} \phi(x) \\ \vdots \\ \nabla_{x_N} \phi(x) \end{bmatrix} = F(x) = \begin{bmatrix} \nabla_{x_1} J_1(x) \\ \vdots \\ \nabla_{x_N} J_N(x) \end{bmatrix}$$

In so doing, we have obtained a system of *24N* partial differential equations. By solving them with respect to $\phi(x)$, we will obtain the potential of the game, again assuming that it exists.
Let us begin with the first (vector) partial differential equation (from now on: PDE):

> *Remark*: I will be loose with the notation. By that, I mean that it makes no sense to derive or integrate a function with respect to a vector. Moreover, I will not be rigorous about the differentiability of the integration constant functions: I will simply give for granted that they are differentiable. However, these choices are meant to avoid notational redundancy. I will later check the correctness of the result.

$$\frac{\partial \phi}{\partial x_1} = \frac{\partial J_1}{\partial x_1} \qquad \Longrightarrow \qquad \phi(x) = J_1(x) + g(x_2, \ldots, x_N)$$

Now, let us consider the second PDE and let us use the above expression of the potential:

$$\frac{\partial \phi}{\partial x_2} = \frac{\partial J_2}{\partial x_2} \qquad \Longrightarrow \qquad \frac{\partial J_1}{\partial x_2} + \frac{\partial g}{\partial x_2} = \frac{\partial J_2}{\partial x_2}$$

Now, since we know the analytical expressions of $J_1(x)$ and $J_2(x)$, we can directly compute the partial derivatives:

$$\frac{\partial g}{\partial x_2} = -\frac{\partial J_1}{\partial x_2} + \frac{\partial J_2}{\partial x_2} = -\left(p_{1,1}x_1\right) + \left(p_{2,1}\sum_{\substack{j=1\\ j\neq 2}}^{N} \tilde{x}_j + p_{2,2}\mathbf{1} + 2p_{2,1}x_2\right)$$

$$\int \frac{\partial g}{\partial x_2} dx_2 = \int \left(-\frac{\partial J_1}{\partial x_2} + \frac{\partial J_2}{\partial x_2}\right) dx_2$$

$$g(x_2, \ldots, x_N) = -p_{1,1}x_1^T x_2 + J_2(x) + h(x_3, \ldots, x_N)$$

Now, $g(x_2, \ldots, x_N)$ is not a function of x_1, because it is a constant resulting from integrating the local x_1-gradient of $\phi(x)$ with respect to x_1. Yet, player 1's local strategy appears in the analytical expression of $g(x_2, \ldots, x_N)$. To solve this issue, let us write the expression of $J_2(x)$:

$$g(x_2, \ldots, x_N) = -p_{1,1}x_1^T x_2 + \left(p_{2,1}\sum_{j=1}^{N} x_j^T x_2 + p_{2,2}\mathbf{1}^T x_2\right) + h(x_3, \ldots, x_N)$$

The term $x_1^T x_2$ appears in the sum as well, premultiplied by $p_{2,1}$. In order for $g(x_2, \ldots, x_N)$ not to depend on x_1, we must impose conditions on $p_{1,1}$ and $p_{2,1}$ so that the two terms $x_1^T x_2$ cancel out and $g(x_2, \ldots, x_N)$ can exist. Specifically, the only condition that allows $g(x_2, \ldots, x_N)$ to exist is:

$$p_{1,1} = p_{2,1}$$

At this point, since $g(x_2, \ldots, x_N)$ exists, the potential also exists and its expression is:

$$\phi(x) = J_1(x) + g(x_2, \ldots, x_N) = J_1(x) + J_2(x) - p_{1,1}x_1^T x_2 + h(x_3, \ldots, x_N)$$

We can now consider the third (vector) PDE and resort to the above expression of the potential:

$$\frac{\partial \phi}{\partial x_3} = \frac{\partial J_3}{\partial x_3} \quad \Longrightarrow \quad \frac{\partial J_1}{\partial x_3} + \frac{\partial J_2}{\partial x_3} + \frac{\partial h}{\partial x_3} = \frac{\partial J_3}{\partial x_3}$$

$$\frac{\partial h}{\partial x_3} = -\frac{\partial J_1}{\partial x_3} - \frac{\partial J_2}{\partial x_3} + \frac{\partial J_3}{\partial x_3} = -p_{1,1}x_1 - p_{2,1}x_2 + \nabla_{x_3} J_3(x)$$

$$\Longrightarrow \int \frac{\partial h}{\partial x_3} dx_3 = \int \left(-\frac{\partial J_1}{\partial x_3} - \frac{\partial J_2}{\partial x_3} + \frac{\partial J_3}{\partial x_3}\right) dx_3$$

$$h(x_3, \ldots, x_N) = -p_{1,1}x_1^T x_3 - p_{2,1}x_2^T x_3 + J_3(x) + l(x_4, \ldots, x_N)$$

Again, we have an apparent dependency of $h(x_3, \ldots, x_N)$ on x_1 and x_2. However, by imposing that

$$p_{1,1} = p_{3,1} \quad \underset{\text{by transitivity}}{\Longrightarrow} \quad p_{2,1} = p_{3,1}$$

we can eliminate it and $h(x_3, \ldots, x_N)$ can exist. At this point, since $h(x_3, \ldots, x_N)$ exists, the potential also exists:

$$\phi(x) = J_1(x) + J_2(x) + J_3(x) - p_{1,1}x_1^T x_2 - p_{1,1}x_1^T x_3 - p_{2,1}x_2^T x_3 + l(x_4, \ldots, x_N)$$

Now, we can notice a pattern that repeats each time we consider a new PDE:

- in order for the system of PDEs to have a solution, it is needed to impose that $p_{1,1} = p_{k,1}$ (assuming that we are at the k-th vector PD equation). By transitivity, this also means that $p_{j,1} = p_{k,1} \quad \forall j = 2, \ldots, k-1$;
- the potential function 'increases' by $J_k(x)$ and 'decreases' by $p_{j,1}x_j^T x_k \quad \forall j = 1, \ldots, k-1$. Mathematically speaking, it is not an actual variation of the potential, but the result of imposing the boundary conditions of the system of PDEs: this step eliminates the dependency of the integration function constant on the k-th variable x_k and, thus, virtually modifies the expression of $\phi(x)$.
- at the last step $k = N$, the integration constant will be a scalar.

Drawing inspiration from this pattern, we can build the potential function as follows:

$$\phi(x) = \sum_{i=1}^{N} J_i(x) - \sum_{j=1}^{N-1} \sum_{k=i+1}^{N} p_{j,1}x_j^T x_k + c \qquad \text{subject to} \quad p_{i,1} = p_{j,1} \quad \forall i, j \in \{1, \ldots, N\}$$

Now, the potential is always defined up to a scalar constant, but we are only interested in the change of potential. Therefore, since we have no further boundary conditions available, let $c = 0$.

Now, let us re-arrange the expression of the potential. For notational simplicity, i will call: $p_{i,1} = p_1 \quad \forall i = \{1, \ldots, N\}$:

$$\phi(x) = \sum_{i=1}^{N} J_i(x) - p_1 \sum_{j=1}^{N-1} \sum_{k=i+1}^{N} x_j^T x_k$$

$$\phi(x) = \sum_{i=1}^{N} \left(p_1 \sum_{j=1}^{N} x_j^T x_i + p_{i,2}\mathbf{1}^T x_i \right) - p_1 \sum_{j=1}^{N-1} \sum_{k=i+1}^{N} x_j^T x_k$$

Let us consider the first addendum, namely the sum of the cost functions of all players. As pointed out before, for each player i, the cost function

$$J_i(x) = p_1 x_i^T x_i + p_1 \sum_{\substack{j=1 \\ j \neq i}}^{N} x_j^T x_i + p_{i,2}\mathbf{1}^T x_i$$

- shows a quadratic dependence on player i's strategy profile x_i ;
- includes the 'coupling' of player i's strategy with the other players' strategies ($x_j^T x_i \;\; \forall j \neq i$).

Now, when we sum the cost functions of all players, the coupling terms ($x_j^T x_i \;\; \forall j \neq i$) become twice as much as the last addendum of the potential function:

$$p_1 \sum_{j=1}^{N-1} \sum_{k=i+1}^{N} x_j^T x_k$$

In fact:
- the above addendum expresses all possible combinations, without repetition, of the aforementioned couplings;
- instead, when we sum the cost functions, the coupling $x_j^T x_i$ will be present both in player i and player j's cost functions, that is, twice, and so for all couples of players.

As a result, the potential can be expressed in a much simpler way:

$$\phi(x) = \sum_{i=1}^{N} \left(p_1 x_i^T x_i + p_{i,2}\mathbf{1}^T x_i\right) - p_1 \sum_{j=1}^{N-1} \sum_{k=i+1}^{N} x_j^T x_k$$

As a final step, let us verify that the above expression of the potential satisfies the condition for which the DSM game is a potential game:

$$J_i(\tilde{x}_i, x_{-i}) - J_i(\hat{x}_i, x_{-i}) = \phi(\tilde{x}_i, x_{-i}) - \phi(\hat{x}_i, x_{-i}) \qquad \forall\ \tilde{x}_i, \hat{x}_i \in K_i \ \wedge\ \forall i \in \{1, \dots, N\}$$

Let us begin by evaluating the difference in potential when player i unilaterally changes their strategy from $\hat{x}_i$ to $\tilde{x}_i$:

$$\phi(\tilde{x}_i, x_{-i}) - \phi(\hat{x}_i, x_{-i}) =$$

$$= \left[\sum_{i=1}^{N} \left(p_1 \tilde{x}_i^T \tilde{x}_i + p_{i,2}\mathbf{1}^T \tilde{x}_i\right) - p_1 \sum_{j=1}^{N-1} \sum_{k=i+1}^{N} \tilde{x}_j^T \tilde{x}_k\right] - \left[\sum_{i=1}^{N} \left(p_1 \hat{x}_i^T \hat{x}_i + p_{i,2}\mathbf{1}^T \hat{x}_i\right) - p_1 \sum_{j=1}^{N-1} \sum_{k=i+1}^{N} \hat{x}_j^T \hat{x}_k\right] =$$

$$= \sum_{i=1}^{N} \left(p_1 \tilde{x}_i^T \tilde{x}_i + p_{i,2}\mathbf{1}^T \tilde{x}_i - p_1 \hat{x}_i^T \hat{x}_i - p_{i,2}\mathbf{1}^T \hat{x}_i\right) - p_1 \sum_{j=1}^{N-1} \sum_{k=i+1}^{N} \left(\tilde{x}_j^T \tilde{x}_k - \hat{x}_j^T \hat{x}_k\right)$$

- in the first sum of $\Delta\phi$, since x_{-i} is fixed, the only variation occurs when considering player *i,* while all other terms cancel out.
- analogously, in the second addendum of the potential, a change occurs only when considering player i's strategy.

As a result:

$$\phi(\tilde{x}_i, x_{-i}) - \phi(\hat{x}_i, x_{-i}) =$$

$$= \left(p_1 \tilde{x}_i^T \tilde{x}_i + p_{i,2}\mathbf{1}^T \tilde{x}_i - p_1 \hat{x}_i^T \hat{x}_i - p_{i,2}\mathbf{1}^T \hat{x}_i\right) - p_1 \sum_{\substack{j=1 \\ j \neq i}}^{N} \left(x_j^T \tilde{x}_i - x_j^T \hat{x}_i\right) =$$

$$= \left(p_1 \tilde{x}_i^T \tilde{x}_i + p_{i,2}\mathbf{1}^T \tilde{x}_i - p_1 \hat{x}_i^T \hat{x}_i - p_{i,2}\mathbf{1}^T \hat{x}_i\right) - p_1 \sum_{\substack{j=1 \\ j \neq i}}^{N} x_j^T \left(\tilde{x}_i - \hat{x}_i\right)$$

Now, let us evaluate the change in cost for player i:

$$J_i(\tilde{x}_i, x_{-i}) - J_i(\hat{x}_i, x_{-i}) =$$

$$= \left(p_1 \tilde{x}_i^T \tilde{x}_i + p_1 \sum_{\substack{j=1 \\ j \neq i}}^{N} x_j^T \tilde{x}_i + p_{i,2}\mathbf{1}^T \tilde{x}_i\right) - \left(p_1 \hat{x}_i^T \hat{x}_i + p_1 \sum_{\substack{j=1 \\ j \neq i}}^{N} x_j^T \hat{x}_i + p_{i,2}\mathbf{1}^T \hat{x}_i\right) =$$

$$= \left(p_1 \tilde{x}_i^T \tilde{x}_i + p_{i,2}\mathbf{1}^T \tilde{x}_i - p_1 \hat{x}_i^T \hat{x}_i - p_{i,2}\mathbf{1}^T \hat{x}_i\right) - p_1 \sum_{\substack{j=1 \\ j \neq i}}^{N} x_j^T \left(\tilde{x}_i - \hat{x}_i\right)$$

We have obtained the same result. Therefore, the solution found above is indeed the potential of the game:

$$\phi(x) = \sum_{i=1}^{N} \left(p_1 x_i^T x_i + p_{i,2}\mathbf{1}^T x_i\right) - p_1 \sum_{j=1}^{N-1} \sum_{k=i+1}^{N} x_j^T x_k \quad \text{subject to } p_{i,1} = p_1 \quad \forall i$$

Comments: as stated in the preliminary comments, we can give a 'physical' interpretation of the condition $p_{i,1} = p_1 \quad \forall i$. This condition allows the potential function to exist, and thus the DSM game to be a potential game. Now, again, when a potential function exists, the force fields acting on the system, i.e. the players, are conservative. As a result, when a body of the system (i.e. a player) is driven by the force field (gradient of their cost) along any path (change of strategy), the work of the force depends on the initial and final conditions only. Now, the cost function of each player is composed of two addenda:

- the first contribution derives from the aggregate energy consumption, which we can interpret (in mechanical terms) as the effect of the global motion of the system (overall strategy profile) on the specific player.
- the second contribution, instead, is the personalised baseline price of the player, which can be interpreted as an energetic factor that depends on the specific player and influences only the specific player's motion.

In light of this, it is perfectly reasonable to impose the conditions that we have found: in order for the force field to be conservative, the effect of the global motion (aggregate energy consumption) on each player (in terms of cost) must be the same. Instead, the personalised baseline energy price affects only the motion (as in cost) of the specific player and it is linear in the players' strategies x_i, therefore no condition is needed on $p_{i,2}$.

Task C

Assess the existence and uniqueness of Nash equilibria for this game.

Preliminary comments: in task $1.a$, we have verified that, under some assumptions, the DSM game is a convex game and it admits a unique Nash equilibrium. In this section, I will essentially check that those assumptions are verified.

Solution

Non-Emptiness, Compactness and Convexity of feasible strategy spaces

First of all, let us check the non-emptiness of the feasible strategy spaces. As shown in task $1.a$, the condition to verify is the following:

$$0 \leq d_i = \sum_{k=1}^{24} e_i^k \leq \mathbf{1}^T \bar{x}_i :$$

```
%% Exercise 1.c
% non-empty strategy spaces
bool = 1;
empty_spaces = [];
for i = 1:N
    di = sum(energy_need(i,:));
    max_capability = sum(xbar(i,:));
    if (di > max_capability | di < 0)
        bool = 0;
        empty_spaces = [empty_spaces;i];
    end
end
disp(' ');
if bool==1
    disp('All strategy spaces are non-empty!');
else
    disp('The following players have empty strategy spaces:');
    disp(empty_spaces');
end
disp(' ');
```

MATLAB returns: All strategy spaces are non-empty!
In task $1.a$, I have also proved that, when the feasible strategy spaces K_i are non-empty, they are also compact and convex.

Continuity and Convexity of the Cost Functions

In task $1.a$, I have proved that the cost functions $J_i(x)$ are always continuous, for all choices of price parameters $p_{i,1}$ and $p_{i,2}$. Moreover, for fixed adversarial strategy profiles (x_{-i}), they are also convex in the corresponding local variables (x_i) if and only if:

$$p_{i,1} \in \mathbb{R}_{\geq 0} \quad \wedge \quad p_{i,2} \in \mathbb{R}$$

Therefore, let us check the condition on the parameters $p_{i,1}$:

```
% convexity of cost functions for fixed x_{-i}
warning = find(p1<0);
if warning ~= []
    disp('The following players have a personalised aggregate energy consumption parameter p_{i,1} < 0:');
    disp(warning);
else
    disp('All players have a personalised aggregate energy consumption parameter p_{i,1} ≥ 0. ');
end
```

MATLAB returns:

All players have a personalised aggregate energy consumption parameter $p_{i,1} \geq 0$.

Assessment of the Existence and Uniqueness of the Nash Equilibrium

Let us resort to the following result:

Consider an N-player convex game and the game map F defined as

$$F : \mathbb{R}^{24N} \rightarrow \mathbb{R}^{24N} \quad \Bigg| \quad F(x) = \begin{bmatrix} \nabla_{x_1} J_1 \\ \vdots \\ \nabla_{x_N} J_N \end{bmatrix}$$

If F is strictly monotone, then the Nash equilibrium exists and it is unique.

As pointed out in task $1.a$, F is strictly monotone if and only if its its Jacobian is strictly positive definite. Additionally, I have proved that a *sufficient* condition for the Jacobian $\mathbf{J}F(x)$ to be strictly positive definite is: $p_{i,1} > 0 \quad \forall i$. Let us verify it:

```
% Uniqueness of the NE
warning = find(p1<0);
if warning ~= []
    disp('The following players have a personalised aggregate energy consumption parameter p_{i,1} ≤ 0:');
    disp(warning);
else
    disp('All players have a personalised aggregate energy consumption parameter p_{i,1} > 0. ');
end
```

MATLAB returns:

All players have a personalised aggregate energy consumption parameter $p_{i,1} > 0$.

Therefore, the NE exists and it is unique.

Remark: in order for a NE to exist, it is required that $\mathbf{J}F(x)$ is at least positive-semidefinite (not necessarily strictly). However, I have merged the proofs of existence and uniqueness of the NE, because I was required to provide both.

Task D

Design an iterative algorithm that converges to a Nash equilibrium x^* and prove that it converges to a Nash Equilibrium regardless of the initialisation.

Solution

Finding an iterative algorithm that converges to a Nash Equilibrium x^*, regardless of the initialisation, can be translated to finding a map

$$T : K \subset \mathbb{R}^{24N} \to K \subset \mathbb{R}^{24N} \quad \Big| \quad x(t+1) = T\left(x(t)\right)$$

such that, given any initialisation $x(0) = x^\circ \in K$:

- the map is **attractive**: $\lim_{t\to\infty} T\left(x(t)\right) = x^*$;
- the Nash equilibrium of the game is the fixed point of the map: $T\left(x^*\right) = x^*$.

Let us define the map as follows:

$$x(t+1) = T\left(x(t)\right) = \Pi_K\left[\, x(t) - \gamma F\left(x(t)\right)\,\right] = \begin{bmatrix} \Pi_{K_1}\left[\, x_1(t) - \gamma F_1\left(x_1(t)\right)\,\right] \\ \vdots \\ \Pi_{K_N}\left[\, x_N(t) - \gamma F_N\left(x_N(t)\right)\,\right] \end{bmatrix}$$

where:

- F is the game map defined above:

$$F : \mathbb{R}^{24N} \to \mathbb{R}^{24N} \quad \Big| \quad F(x) = \begin{bmatrix} F_1 \\ \vdots \\ F_N \end{bmatrix} = \begin{bmatrix} \nabla_{x_1} J_1 \\ \vdots \\ \nabla_{x_N} J_N \end{bmatrix}$$

- Π_k is the **projection operator**, defined as follows:

$$\Pi_K : \mathbb{R}^{24N} \to K \in \mathbb{R}^{24N} \quad \Big| \quad \Pi_K[z] = \underset{y\in K}{\operatorname{argmin}} \|y - z\|_2^2$$

  The reason why a projection operator is needed lies in the fact that the optimal solution might not lie in the global feasible action set K.
- γ is the **step-length**: it is a scalar coefficient to be chosen in such a way that the convergence of the scalar map towards the fixed point is optimised.

The above $\left(T(x)\right)$ is called **projected game map**. The interpretation is the following: at each iteration, we update the strategy profile of each player x_i in the direction of the steepest improvement of the cost J_i , assuming that the adversarial strategies x_{-i} are fixed.

In the previous tasks, we have already proved that this particular game is a convex game with a unique Nash Equilibrium. Instead, now we will prove that:

1. the equilibrium of the projected game map iteration is the Nash equilibrium of the game;
2. the projected game map iteration converges to its equilibrium

1 - *equilibrium of the projected game map iteration = NE of the game*

Let us resort to the following result (proved in class):

For any step-length $\gamma > 0$:

$$x^* \in \mathrm{SOL}(K, F) \quad \iff \quad x^* = \Pi_K \left[\, x^* - \gamma F(x^*) \, \right]$$

In other words, assuming that the step-length is positive, any fixed point of the projected game map iteration is a solution to the variational inequality of the game map F over the global feasible strategy space K:

$$x^* \in \mathrm{SOL}(K, F) = \left\{ x^* \in K \mid F(x^*)^T (y - x^*) \geq 0 \quad \forall y \in K \right\}$$

Moreover, it is possible to prove the following:

Given a convex, N-player game with $J_i(x)$ continuously differentiable in their local variables (x_i) for all x_{-i}, and the game map:

$$F : \mathbb{R}^{24N} \to \mathbb{R}^{24N} \quad \mid \quad F(x) = \left[\left(\nabla_{x_1} J_1(x) \right)^T \; , \; \ldots \; , \; \left(\nabla_{x_N} J_N(x) \right)^T \right]$$

then x^* is a Nash Equilibrium if and only if $x^* \in \mathrm{SOL}(K, F)$.

Since the hypotheses of this second result are all verified (see previous tasks), by transitivity, we can infer that any fixed point of the projected game map iteration is a Nash equilibrium of the game and conversely.
Finally, in task $1.c$, we have proved that the game admits a unique NE, which is also the unique fixed point of the projected game map iteration.

2 - *the projected game map iteration converges to its fixed point*

We can prove the statement by resorting to Banach fixed point theorem:

1. Let K be a complete metric space with distance $d(\,.\,,\,.\,) : K \times K \to \mathbb{R}$.
2. Let $T : K \to K$ be a **contraction**, that is:

$$\exists \, \tau \in [\,0, 1\,) \quad \Big| \quad d\left(T(x), T(y) \right) \leq \tau \, d(x, y) \quad \forall x, y \in K$$

Then T has a unique fixed point: $x^* \in K \mid x^* = T(x^*)$.
Moreover, starting from any $x(0) = x^\circ \in K$, the iteration

$$x(t+1) = T\left(x(t)\right)$$

converges to x^*.

Let us check, one by one, the hypotheses of the theorem.

Complete Metric Space with Distance (Hypothesis 1)

A metric space is a set equipped with a metric, i.e. a distance function which has some properties. Numerous metrics can be defined. However, I will resort to the most common one, that is, the 2-norm distance:

$$d(x,y) : K \times K \to \mathbb{R} \quad \Big| \quad d(x,y) = \|x-y\|_2$$

The 2-norm distance satisfies the properties for a distance function to be a metric:

- **nonnegativity**, i.e. $d(x,y) \geq 0 \quad \forall x,y \in K \qquad \wedge \qquad d(x,y) = 0 \iff x = y$
    - $x \neq y: \quad d(x,y) = \|x-y\|_2 = \sqrt{(x_1-y_1)^2 + \ \dots \ + (x_{24N}-y_{24N})^2} \geq 0$
    - $x = y: \quad d(x,y) = \|x-y\|_2 = \|x-x\|_2 = \|\mathbf{0}\|_2 = \sqrt{0^2 + \ \dots \ + 0^2} = 0$
- **symmetry**, i.e. $d(x,y) = d(y,x) \quad \forall x,y \in K$

$$d(x,y) = \|x-y\|_2 = \sqrt{(x_1-y_1)^2 + \ \dots \ + (x_{24N}-y_{24N})^2} = \sqrt{(y_1-x_1)^2 + \ \dots \ + (y_{24N}-x_{24N})^2} = \|y-x\|_2 = d(y,x)$$

- **triangle inequality**, i.e. $d(x,z) \leq d(x,y) + d(y,z) \quad \forall x,y,z \in K$

$$d(x,z) = \|x-z\|_2 = \|x-y+y-z\|_2 \leq \|x-y\|_2 + \|y-z\|_2 = d(x,y) + d(y,z)$$

Now, consider the metric space $(K, \|\,.\,\|_2)$. The set $K \subset \mathbb{R}^{24N}$ is compact (see previous tasks) over a complete metric space $\left(\mathbb{R}^{24N}\right)$. Therefore the metric space $(K, \|\,.\,\|_2)$ is also complete. In fact, by definition, a normed space is said to be complete with respect to its norm if and only if every Cauchy sequence in the space converges to a limit that is also in the space. Being $K \subset \mathbb{R}^{24N}$ compact, then every Cauchy sequence in the space converges to a limit in the space itself.
Therefore, the first hypothesis of Banach fixed point theorem is verified.

The Projected Game Map is a contraction (Hypothesis 2)

We can resort to the following result (proved in class):

1. Let F be μ-strongly monotone
2. Let F be L-Lipschitz
3. Let $\gamma \in (0, 2\mu/L^2)$

Then the projected game map iteration

$$x(t+1) = T\left(x(t)\right) = \Pi_K\left[\, x(t) - \gamma F\left(x(t)\right) \,\right]$$

is a contractive map (composition of a non-expansive map over a contractive map, which is a contractive map).

Now, F is μ-strongly monotone if and only if

$$\big(F(y) - F(x) \big)^T (y - x) \geq \mu \| y - x \|_2^2 \quad \forall x, y \in K$$

Instead, F is L-Lipschitz if and only if

$$\| F(y) - F(x) \|_2 \leq L \| y - x \|_2 \quad \forall x, y \in K$$

Let us begin by finding the strong monotonicity constant μ.
Remember that the game map is linear in all its variables:

$$F(x) = \begin{bmatrix} \nabla_{x_1} J_1 \\ \vdots \\ \nabla_{x_N} J_N \end{bmatrix} = \begin{bmatrix} p_{1,1} \sum_{j=1}^{N} x_j \ + \ p_{1,2}\mathbf{1} \ + \ p_{1,1} x_1 \\ \vdots \\ p_{N,1} \sum_{j=1}^{N} x_j \ + \ p_{N,2}\mathbf{1} \ + \ p_{N,1} x_1 \end{bmatrix}$$

Therefore, its first-order approximation is actually an exact relation:

$$F(x) = F(0) + \left[\mathbf{J}F(0) \right] x = \begin{bmatrix} p_{1,2}\mathbf{1} \\ \vdots \\ p_{N,2}\mathbf{1} \end{bmatrix} + \begin{bmatrix} p_{1,1} \left[\ 2 \cdot \mathbf{1}^{24\times 24} \quad \mathbf{1}^{24\times 24} \quad \mathbf{1}^{24\times 24} \quad \ldots \quad \mathbf{1}^{24\times 24} \ \right] \\ p_{2,1} \left[\ \mathbf{1}^{24\times 24} \quad 2 \cdot \mathbf{1}^{24\times 24} \quad \mathbf{1}^{24\times 24} \quad \ldots \quad \mathbf{1}^{24\times 24} \ \right] \\ \vdots \\ p_{N,1} \left[\ \mathbf{1}^{24\times 24} \quad \mathbf{1}^{24\times 24} \quad \mathbf{1}^{24\times 24} \quad \ldots \quad 2 \cdot \mathbf{1}^{24\times 24} \ \right] \end{bmatrix} x$$

In light of this, <u>**assuming that the Jacobian and its transpose**</u> $\left(\mathbf{J}F \text{ and } \left([\mathbf{J}F]^T \right) \right)$ <u>**have strictly positive eigenvalues**</u>:

$$\big(F(y) - F(x) \big)^T (y - x) = (y - x)^T [\mathbf{J}F]^T (y - x) \geq$$

$$\geq (y - x)^T \lambda_{\min} \left([\mathbf{J}F]^T \right) (y - x) = \lambda_{\min} \left([\mathbf{J}F]^T \right) \|y - x\|_2^2 \quad \forall x, y \in K$$

Therefore: $\mu = \lambda_{\min} \left([\mathbf{J}F]^T \right)$.

Let us now find the Lipschitz constant of the game map L:

$$\| F(y) - F(x) \|_2 = \| \mathbf{J}F \, (y - x) \|_2 \leq \| \lambda_{\max}(\mathbf{J}F) \ (y - x) \|_2 = |\lambda_{\max}(\mathbf{J}F)| \ \|y - x\|_2$$

Therefore, $L = \left| \lambda_{\max}(\mathbf{J}F) \right|$.

To summarise, if $\gamma \in \left(0, 2\dfrac{\mu}{L^2} \right) = \left(0, 2\dfrac{\lambda_{\min}\left([\mathbf{J}F]^T \right)}{\lambda_{\max}(\mathbf{J}F)^2} \right)$, then the projected game map is contractive, which satisfies the hypothesis of Banach fixed point theorem and proves the convergence of the projected game map iteration towards its fixed point, which is also the unique Nash Equilibrium of the game.

Fundamental Remark

1. First of all, in the above discussion, I have made a rather strong assumption, that is, the Jacobian of the game map and its transpose have strictly positive eigenvalues. This assumption is fundamental for the upper bound of γ to be meaningful.
2. Moreover, the above illustrated procedure might be affected by many potential issues.

I will focus on these aspects in the following task.

Task E

In this section I will justify the choices that I have made to improve the efficacy of the projected game map iteration. I will also include all the attempts that I have made.

Solution

As proved in the previous task, the projected game map iteration converges towards its unique fixed point, i.e. the Nash equilibrium of the game, if and only if the step-size

$$\gamma \in \left(0\,,2\frac{\mu}{L^2}\right) = \left(0\,,2\frac{\lambda_{\min}\left([\mathbf{J}F\,]^T\right)}{\lambda_{\max}(\mathbf{J}F\,)^2}\right)$$

However, for the statement to be meaningful, it is crucial to check the conditions on the eigenvalues of $\mathbf{J}F$ and $\left([\mathbf{J}F\,]^T\right)$. In this regard, notice that, though being strictly positive definite, the Jacobian is not symmetric, which means that its eigenvalues and the eigenvalues of its transpose *are not* guaranteed to be strictly positive.

> *Remark*: requiring that the eigenvalues of $\mathbf{J}F$ and $\left([\mathbf{J}F\,]^T\right)$ are all strictly positive is only a sufficient condition for the statement of convergence to be meaningful, but it is not also necessary. In other words, if the above conditions are not both met, the provided interval for the step-size might still be meaningful, but further observations would be required.

Let us check the above condition:

```
% Condition on the eigenvalues of JF and JF^T
eigvals_JFT = eig(JF');
eigvals_JF = eig(JF);
if all(eigvals_JFT > 0)
    disp('The eigenvalues of JF^T are all strictly positve!');
else
    disp('The eigenvalues of JF^T are NOT all strictly positve!');
end
disp("    ");
if all(eigvals_JF > 0)
    disp('The eigenvalues of JF are all strictly positve!');
else
    disp('The eigenvalues of JF are NOT all strictly positve!');
end
disp("    ");
```

MATLAB returns:

> The eigenvalues of $\left([\mathbf{J}F\,]^T\right)$ are all strictly positive!
> The eigenvalues of $\mathbf{J}F$ are all strictly positive!

Therefore, the provided interval for the step-size is meaningful and the algorithm illustrated in task $1.d$ can be used for this task.
We can now compute μ, L and finally the open upper bound for γ:

$$\gamma \in \left(0\,,0.012995075352174\right) \approx (0\,,0.013)$$

Attempts of Efficacy Improvement

Consider again the proposition:

1. Let F be μ-strongly monotone
2. Let F be L-Lipschitz
3. Let $\gamma \in (0, 2\mu/L^2)$

Then the projected game map iteration

$$x(t+1) = T\left(x(t)\right) = \Pi_K\left[\ x(t) - \gamma F\left(x(t)\right)\ \right]$$

is a contractive map (composition of a non-expansive map over a contractive map, which is a contractive map).

In class, we have proved that the proposition holds because

$$\left\| \Pi_K[x - \gamma F(x)] - \Pi_K[y - \gamma F(y)] \right\|_2^2 \leq \ \dots \ \leq \left(1 + \gamma^2 L^2 - 2\gamma\mu\right) \|x - y\|_2^2 \quad \forall x, y \in K$$

which satisfies the hypothesis of contractive map, required for Banach fixed point theorem to hold. Now, let

$$\tau(\gamma) = 1 + \gamma^2 L^2 - 2\gamma\mu \qquad \forall \gamma \in \left(0, 2\frac{\mu}{L^2}\right)$$

$\tau(\gamma)$ is the contraction coefficient of the projected game map, which is a measure of the convergence rate of the projected game map iteration. In order to optimise the convergence of the algorithm, let us minimise the contraction coefficient. Keeping in mind that μ and L are fixed, because they are properties of the game map $F(x)$:

$$\frac{\partial\tau}{\partial\gamma} = 0 \qquad \Longleftrightarrow \qquad 2\gamma L^2 - 2\mu = 0 \qquad \Longleftrightarrow \qquad \gamma_{\text{opt}} = \frac{\mu}{L^2} \in \left(0, 2\frac{\mu}{L^2}\right)$$

the optimal step-size is half the upper bound found above. As a result, the optimal contraction coefficient is

$$\tau(\gamma_{\text{opt}}) = 1 + \gamma_{\text{opt}}^2 L^2 - 2\gamma_{\text{opt}}\,\mu = 1 + \frac{\mu^2}{L^2} - 2\frac{\mu^2}{L^2} = 1 - \frac{\mu^2}{L^2} \approx 0.999931775854401$$

In other words, the contractiveness of the projected game map is extremely low and so is the convergence rate. I will show that this will be a challenging issue, because the problem is computationally quite expensive and running many iterations is very time-consuming.

Before diving into the limits of the projected game map iteration, let us interpret the convergence rate in mathematical terms. Consider again the projected game map

$$x(t+1) = \Pi_K\left[x(t) - \gamma F\left(x(t)\right)\right] = \Pi_K\left[x(t) - \gamma\left(F(0) + \left[\mathbf{J}F(0)\right] x(t)\right)\right] =$$

$$= \Pi_K\left[\left(\mathbf{1} - \gamma\,\mathbf{J}F\right) x(t) - \gamma\, F(0)\right] = \Pi_K\left[A\, x(t) - b\right]$$

In this particular problem setup, the projected game map iteration is nothing more than the implementation of a splitting method, specifically a projected **Richardson static method**. In other words, the equilibrium point can be found by projecting the solution of a linear system.
The contractiveness of the projected game map thus reads as follows:

$$\| \Pi_K [Ax - b] - \Pi_K [Ay - b] \|_2^2 \underset{\Pi_K \text{ non-expansive operator}}{\leq} \| [Ax - b] - [Ay - b] \|_2^2 =$$

$$= \| A(x - y) \|_2^2 \leq \left(\lambda_{\max}(A)\right)^2 \| x - y \|_2^2$$

where $\lambda_{\max}(A)$ is the spectral radius of $A = \mathbf{1} - \gamma \mathbf{J}F$, which, again, is non-symmetric, but (I have verified that) all its eigenvalues are strictly positive.

From this perspective, we can make some comments about the nature of the problem:

- theoretically, it is possible to improve the convergence rate of the algorithm by resorting to a preconditioning matrix P

$$A = \mathbf{1} - \gamma P^{-1} \mathbf{J}F$$

  such that the spectral radius of A decreases. One very interesting choice could be the Gauss-Seidel preconditioner $P = \text{tril}(\mathbf{J}F)$, that is, the lower triangular matrix of $\mathbf{J}F$. The reason lies in the fact that the Gauss-Seidel method allows to maximise the use of information. For instance, once computed the best response of player 1, such a preconditioner allows to compute the best response of player 2 based on the new result for player 1 and so on for all players. In other words, $P = \text{tril}(\mathbf{J}F)$ allows to build an algorithm that simulates a repeated game where, in turn, only one player updates their strategy based on the strategies of all other players.
  However, $\mathbf{J}F$ is neither symmetric nor diagonally strictly dominant, therefore the convergence of the algorithm with a Gauss-Seidel preconditioning matrix is not guaranteed (I have checked that the algorithm does not converge).

- the problem is extremely well conditioned, which means that the algorithm is stable with respect to perturbations in the initial conditions:

$$K_2 = \sqrt{\frac{\lambda_{\max}(A^T A)}{\lambda_{\min}(A^T A)}} \approx 1$$

  However, the convergence rate remains a huge issue.

Now, in light of the above considerations, let us implement the algorithm.

- **initialisation**:
  - as a first step, I have defined a random vector. I have chosen the values of the random vector to be in a range $x_0(i) \in (0, 0.01)$.
  - then, I have projected the initial guess x_0 in the global feasible strategy space: $x_0 = \Pi_K(x_0)$. The reason lies in Banach fixed point theorem, which guarantees the convergence of the projected game map iteration for any $x_0 \in K$.

```
% Random Initialisation: x0 ∈ K
x0 = 0.01*rand(N*h,1);
xProj = sdpvar(N*h,1);
sum_constraint = [];
upper_bound = [];
for i = 1:N
    sum_constraint = [sum_constraint, sum(xProj(((i-1)*h+1):i*h)) == sum(energy_need(i,:))];
    for j = 1:h
        upper_bound = [upper_bound , xProj((i-1)*h+j) <= xbar(i,j) ];
   end
end
constraints = [0 <= xProj; upper_bound ; sum_constraint];
objective = (xProj-x0)'*(xProj-x0);
optimize(constraints,objective,sdpsettings('verbose',0));
x_it = value(xProj);
```

- projected game map iteration (the number of iterations is max_it = 100):

```
% Best-response algorithm
for it = 1:max_it
    % Update
    x_it = A*x_it-b;
    % Projection
    xProj = sdpvar(N*h,1);
    sum_constraint = [];
    upper_bound = [];
    for i = 1:N
        sum_constraint = [sum_constraint, sum(xProj(((i-1)*h+1):i*h)) == sum(energy_need(i,:))];
        for j = 1:h
            upper_bound = [upper_bound , xProj((i-1)*h+j) <= xbar(i,j) ];
        end
    end
    constraints = [0 <= xProj; upper_bound ; sum_constraint];
    objective = (xProj-x_it)'*(xProj-x_it);
    optimize(constraints,objective,sdpsettings('verbose',0));
    x_it = value(xProj);
end
```

I have run the above code multiple times and I have observed the following facts:

1. with a number of iterations equal to 100, the time required by MATLAB to run the code is approximately 90 seconds.
2. the solutions of different simulations differ at the same order of magnitude of the amplitude of the interval of the initial guess. For instance, calling x_{NE1} and x_{NE2} two different solutions:

$$x_0(i) \in (0, 0.5) \qquad \text{results in} \qquad \max_{i \in \{1, \dots 2400\}} |x_{NE1}(i) - x_{NE2}(i)| \approx 0.58$$

3. from the above, it follows that, the shorter the interval for the initial guess x_0, the 'better' the convergence, apparently.

A very reasonable explanation to the last statement is that, the lower the variability in the initial guess, "the less random it is". In other words, when running the code multiple times, we are not actually considering different initial strategies, but quite close ones. Furthermore, I have run the algorithm allowing 1000 iterations and the result does not improve much.
We can therefore draw a conclusion: as expected when computing the contraction rate of the projected map (≈ 1), the convergence of the algorithm is too slow !!!

Graziano Davide

It is possible to implement a different algorithm, specifically one that simulates a repeated game where one player at a time updates their strategy by playing their best response:

$$x_i(t+1)\Big|_{x_{-i}=\tilde{x}_{-i}} = \Pi_{K_i}\left[x_i(t) - \gamma F_i\left(x_i(t)\,,\tilde{x}_{-i}(t)\right)\right]$$

However, we can already expect an even worse convergence, because, for one update of the global strategy profile x, 100 projections are needed. The algorithm is the following:

```
% Best-response algorithm
residual = 1;
norm_b = norm(b);
for it = 1:max_it
    x_prev = x_it;
    % Update
    x_it = A*x_it-b;
    % Projection
    residual = norm(x_prev-x_it)/norm_b;
    xProj = sdpvar(N*h,1);
    sum_constraint = [];
    upper_bound = [];
    for i = 1:N
        sum_constraint = [sum_constraint, sum(xProj(((i-1)*h+1):i*h)) == sum(energy_need(i,:))];
        for j = 1:h
             upper_bound = [upper_bound , xProj((i-1)*h+j) <= xbar(i,j) ];
        end
    end
    constraints = [0 <= xProj; upper_bound ; sum_constraint];
    objective = (xProj-x_it)'*(xProj-x_it);
    optimize(constraints,objective,sdpsettings('verbose',0));
    x_it = value(xProj);
    if residual < 10^(-7)
       break;
    end
end
```

As expected, the convergence is even slower.

With the specification of a maximum elapsing time of 2 minutes, the projected game map algorithm needs to be reviewed, if possible.
Consider again the result obtained above:

$$\|\,\Pi_K[Ax-b]-\Pi_K[Ay-b]\,\|_2^2 \underset{\Pi_K \text{ non-expansive operator}}{\leq} \|\,[Ax-b]-[Ay-b]\,\|_2^2 =$$

$$= \|\,A(x-y)\,\|_2^2 \;\leq\; \left(\lambda_{\max}(A)\right)^2 \|\,x-y\,\|_2^2$$

Here is a key-idea: we can boost the convergence by modifying the spectral radius of A. In this regard, notice that

$$Av = \lambda v \qquad \Longleftrightarrow \qquad \alpha A v = \alpha\lambda v \quad \forall \alpha \neq 0$$

modifying the spectral radius of A is extremely simple: it is sufficient to multiply the matrix by some coefficient $\alpha \neq 0$. Now, remembering that the contraction coefficient of the map is

$$\tau = \left(\lambda_{\max}(A)\right)^2$$

we can identify an interval where α should be placed: $\alpha \in (0,1)$. In so doing, the contraction ratio is still $\tau \in [0,1)$, as required by Banach fixed point theorem.
However, modifying the matrix $A = \mathbf{1} - \gamma \mathbf{J}F$ implies a change in the structure of the projected game map as well:

$$x(t+1) = \Pi_K\left[\alpha A x - b\right] = \Pi_K\left[\alpha x - \gamma\left(F(0) + \alpha \mathbf{J}Fx\right)\right] = \Pi_K\left[\alpha x - \gamma F(\alpha x)\right]$$

1. Banach fixed point theorem still holds. Therefore, the new projected game map iteration still converges towards its unique fixed point for any initial guess in K.
2. However, it is needed to verify that the fixed point of the map still corresponds to the unique Nash equilibrium of the game:

   For any step-length $\gamma > 0$:

$$x^* = \Pi_K\left[\ \alpha x^* - \gamma F(\alpha x^*)\ \right] \quad \Longleftrightarrow \quad x^* \in \mathrm{SOL}(K, F)$$

It is easy to verify that the above theorem does not hold. In fact:

$$x = \alpha x^* - \gamma F(\alpha x^*) \quad \longrightarrow \quad \bar{x} = \Pi_K[x] \neq x^*$$

Therefore, multiplying the matrix by some coefficient can boost the convergence of the map, but the fixed point of the latter is not anymore the Nash Equilibrium of the game. However, the intuition is still of use.

Improved Algorithm

Consider one last time the following:

1. Let F be μ-strongly monotone
2. Let F be L-Lipschitz
3. Let $\gamma \in (0, 2\mu/L^2)$

Then the projected game map iteration

$$x(t+1) = T\left(x(t)\right) = \Pi_K\left[\ x(t) - \gamma F\left(x(t)\right)\ \right]$$

is a contractive map (composition of a non-expansive map over a contractive map, which is a contractive map).

It provides a sufficient condition!!! The optimal step-size might not be $\gamma \in (0, 2\mu/L^2)$ and, in general, it does not have to.
Hence, I considered again the intuition I had above. The contraction rate of the projected game map is given by

$$\tau = \left(\lambda_{\max}(A)\right)^2$$

Thus, we can decrease the spectral radius of A by minimising over $\gamma > 0$, so that Banach fixed point theorem still holds. To clarify, I will formalise my intention, based on what I have observed so far:

Conjecture

Let the game map, defined as

$$F : \mathbb{R}^{24N} \to \mathbb{R}^{24N} \quad \Big| \quad F(x) = \left[\left(\nabla_{x_1} J_1(x) \right)^T \quad \dots \quad \left(\nabla_{x_N} J_N(x) \right)^T \right]$$

be linear in its argument. Let $\gamma > 0$. Let K be a non-empty complete metric space. Let T be the projected game map iteration

$$T : K \to K \quad \Big| \quad x(t+1) = T\left(x(t)\right) = \Pi_K\left[x(t) - \gamma F\left(x(t)\right) \right]$$

If the spectral radius

$$\rho\left(\mathbf{1} - \gamma \mathbf{JF}\right) = \max_{\lambda} \left| \text{Spectrum}\left(\mathbf{1} - \gamma \mathbf{JF}\right) \right| < 1$$

then, T converges towards its unique fixed point, for any initial guess $x^\circ \in K$. Moreover, an (at least locally) optimal step-size is

$$\gamma_{opt} = \underset{\gamma \in \mathbb{R}_{>0}}{\text{argmin}}\, \rho\left(\mathbf{1} - \gamma \mathbf{JF}\right)$$

Sketch of Proof

First of all, since F is linear, its first order approximation is actually an exact relation:

$$F(x) = F(x_0) + \mathbf{J}F(x_0)(x - x_0) = F(x_0) + \mathbf{J}F(x - x_0)$$

Then, the projected game map reads as follows:

$$x(t+1) = \Pi_K\left[x(t) - \gamma F\left(x(t)\right)\right] = \Pi_K\left[x(t) - \gamma\left(F(x_0) + \left[\mathbf{J}F(0)\right]\left(x(t) - x_0\right)\right)\right] =$$

$$= \Pi_K\left[\left(\mathbf{1} - \gamma\mathbf{J}F\right)x(t) - \gamma\left(F(x_0) - \mathbf{J}F\, x_0\right)\right] = \Pi_K\left[A\,x(t) - b\right]$$

Now, being K a complete non-empty metric space for hypothesis, Banach fixed point theorem guarantees the convergence of the projected game map $\forall\, x^\circ \in K$ if the map is a contraction. More accurately, if

$$\exists\, \tau \in [0, 1) \quad \Big| \quad d\left(T(x), T(y)\right) \leq \tau\, d(x, y) \quad \forall x, y \in K$$

then T has a unique fixed point $x^* \in K \mid x^* = T(x^*)$ and the iteration converges to x^* as $t \to \infty$. Then, let us find the condition for which T is a contraction:

$$\| \Pi_K[Ax - b] - \Pi_K[Ay - b] \|_2^2 \underset{\Pi_K \text{ non-expansive operator}}{\leq} \| [Ax - b] - [Ay - b] \|_2^2 =$$

$$= \| A(x-y) \|_2^2 \leq \left(\max_{\lambda} |\lambda(A)| \right)^2 \| x-y \|_2^2$$

To guarantee contractiveness:

$$0 \leq \max_{\lambda} |\lambda(A)| < 1$$

which proves the first argument.

Now, let $\rho = \max_{\lambda} |\lambda(A)|$. The convergence rate of the map can be found by induction, resorting to a Cauchy sequence:

$$d\left(T(x_m), T(x_n) \right) \leq \rho\, d(x_m, y_n) \leq \rho \left[d(x_m, y_{m-1}) + d(x_{m-1}, y_{m-2}) + \ \dots \ + d(x_{n+1}, y_n) \right]$$

$$\leq \rho \left[\rho^{m-1} d(x_1, x_0) + \rho^{m-2} d(x_1, x_0) + \ \dots \ + \rho^n d(x_1, x_0) \right] =$$

$$= \rho^{n+1} d(x_1, x_0) \sum_{k=0}^{m-n-1} \rho^k \leq$$

$$\leq \rho^{n+1} d(x_1, x_0) \sum_{k=0}^{\infty} \rho^k =$$

$$= \rho^{n+1} d(x_1, x_0) \left(\frac{1}{1-\rho} \right)$$

The distance between the n-th and the m-th projections on K via T decreases as a power of ρ. Thus

$$\gamma_{opt} = \underset{\gamma \in \mathbb{R}_{>0}}{\operatorname{argmin}} \max_{\lambda} |\lambda\left(A(\gamma) \right)|$$

is an (at least locally) optimal step-size for the projected game map iteration and the second argument is proved.

By solving the optimisation, I have found $\gamma_{opt} \approx 1.56$, which yields:

$$\rho = \max_{\lambda} |\lambda(A)| = 0.9836$$

The convergence rate is still low, but its value is much smaller than before. Moreover, the stability of the algorithm is basically not affected by this change:

$$K_2 = \sqrt{\frac{\lambda_{\max}(A^T A)}{\lambda_{\min}(A^T A)}} \approx 1.2$$

Therefore, as a final step, let us implement the code:

- initialisation: let us try to wildly increase the randomness in the initial guess, by setting the amplitude of the random interval to 20

```
% Random Initialisation: x0 ∈ K
x0 = 20*rand(N*h,1);
xProj = sdpvar(N*h,1);
sum_constraint = [];
upper_bound = [];
for i = 1:N
    sum_constraint = [sum_constraint, sum(xProj(((i-1)*h+1):i*h)) == sum(energy_need(i,:))];
    for j = 1:h
        upper_bound = [upper_bound , xProj((i-1)*h+j) <= xbar(i,j) ];
   end
end
constraints = [0 <= xProj; upper_bound ; sum_constraint];
objective = (xProj-x0)'*(xProj-x0);
optimize(constraints,objective,sdpsettings('verbose',0));
x_it = value(xProj);
```

- algorithm:

```
% Best-response algorithm
% residual = 1;
% norm_b = norm(b);
for it = 1:max_it
    x_prev = x_it;
    % Update
    x_it = A*x_it-b;
    % Projection
    xProj = sdpvar(N*h,1);
    sum_constraint = [];
    upper_bound = [];
    for i = 1:N
        sum_constraint = [sum_constraint, sum(xProj(((i-1)*h+1):i*h)) == sum(energy_need(i,:))];
        for j = 1:h
             upper_bound = [upper_bound , xProj((i-1)*h+j) <= xbar(i,j) ];
        end
    end
    constraints = [0 <= xProj; upper_bound ; sum_constraint];
    objective = (xProj-x_it)'*(xProj-x_it);
    optimize(constraints,objective,sdpsettings('verbose',0));
    x_it = value(xProj);
    % residual = norm(x_prev-x_it)/norm_b;
    % if residual < 10^(-6)
    %    break;
    % end
end
```

By allowing a number of iterations equal to 100:
- MATLAB still requires 90 seconds to run the code;
- the difference in results between different runs has an order of magnitude of 10^{-3}
- the results are essentially equal to the ones that I have obtained by letting the original iteration run for 1000 iterations.

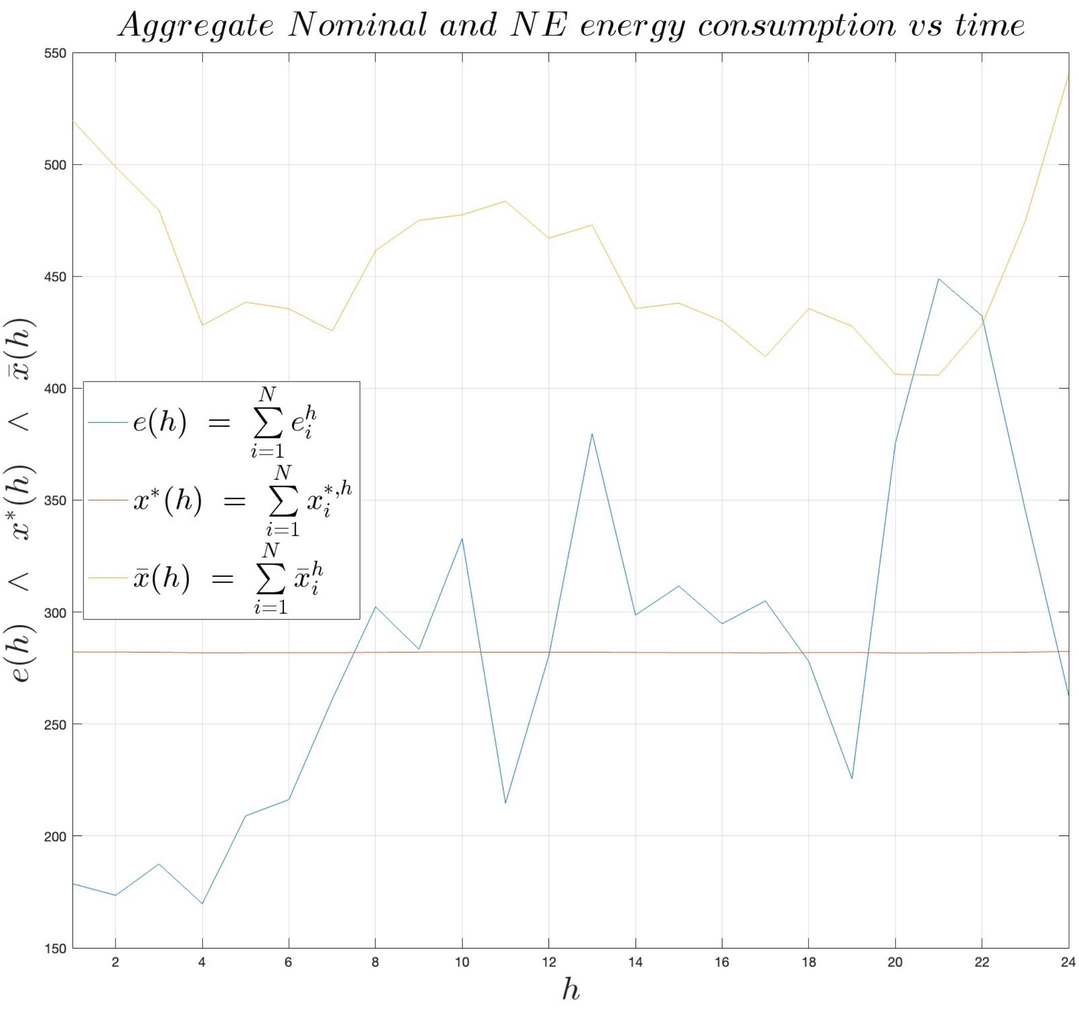
Hence, we can draw a conclusion: despite a much increased randomness in the initial guess, the convergence rate has improved of a factor 10!
I have also tried to further boost the convergence with a preconditioning matrix, but again the margin is too low and the iteration does not converge.

Task F

Plot the aggregate nominal energy consumption and the aggregate energy consumption at the Nash equilibrium in one figure over the 24 hours. Compare the two plots and comment on their differences.

Solution

The requested plot is the following:



- in blue, I have plotted the nominal aggregate energy consumption;
- in red, I have plotted the aggregate energy consumption at the Nash equilibrium;
- in yellow, I have plotted the hourly aggregate energy consumption limit.

We can observe that the nominal aggregate energy consumption is poorly smoothed over the 24 hours. On average, we can somehow identify an upward trend. However, the only conclusion that we can draw is that the energy need of the players ranges in a rather wide interval. This scenario is extremely inefficient for the players, in terms of cost. In fact, consider the definition of the cost function:

$$J_i(x) = \left(p_{i,1} \sum_{j=1}^{N} x_j \ + \ p_{i,2}\mathbf{1} \right)^T x_i = p_{i,1} \sum_{j=1}^{N} x_j^T x_i \ + \ p_{i,2}\mathbf{1}^T x_i$$

When the aggregate energy consumption peaks, on average, the first addendum peaks as well among the players. There might still be some player who consumes low energy at peak hours, thus damping the effect of the peaks. However, the aggregate energy consumption surges when, on average, the majority of the players consumes high level of energy. For those players, the cost at peak hours can result in a very poor outcome.
Having said that, it is more than rational to infer that the nominal aggregate energy consumption is not a Nash equilibrium. In fact, for fixed adversarial strategies, a player who is consuming high energy at peak hours can unilaterally decrease their cost by reducing their consumption at peak hours and increasing it where the global consumption plummets.

I have also plotted the hourly aggregate energy consumption limit. The reason behind this choice is to investigate which is the degree of freedom of the global system in the simulated DSM game. In other words, this plot clarifies how much the whole set of players can vary their global strategy without falling outside the global feasible strategy space. Of course, this plot is not instructive of the single players, yet I find it helpful to understand the dynamics of the game as a whole. We can observe that the rangeability of the global strategy is very high, which is a positive factor for the convergence of the simulated DSM game towards a NE.

Finally, the red line is the unique Nash equilibrium of the DSM game. Now, this particular instance of the game cannot be associated with a potential function, since the price parameters $p_{i,1}$ are different for all players. However, the energetic interpretation that I have sketched in task $1.b$ still holds. Specifically, the cost functions can be interpreted as force fields acting on the players, driving them towards equilibria. As pointed out, the aggregate energy consumption affects all cost functions, i.e. it is a force that drives all players towards the NE. Therefore, it is in all players' best interest to flatten as much as possible the global energy consumption over the 24 hours.
Naturally, the above is a global interpretation, which is a more straightforward one. As regards the internal dynamics, i.e. the strategy profiles of the single players, the interpretation is much more complicated. In fact, single players:

- try to flatten the global energy consumption, because it is in their best interest;
- must choose a strategy which respects their inequality and equality constraints;
- play, at each iteration, their best response strategy based on the adversarial strategies observed at the previous iteration of the algorithm.

As a result, the Nash equilibrium strategy profiles of the single players do not uniformly distribute the energy consumption over the 24 hours.