A VOTING POWER MEASURE FOR LIQUID DEMOCRACY WITH MULTIPLE DELEGATION

FRANCISCO M. BERSETCHE

ABSTRACT. We generalize the classical model of liquid democracy by proposing a voting power measure that allows each agent to split and delegate their vote to multiple agents. We prove that this measure is well defined and inherits the most important properties of the classical model. Among these properties we prove the so-called delegation property, which guarantees us that delegating power to an agent is equivalent to copying her delegation profile. Secondly we study the existence of equilibrium states in a delegation game using the proposed measure, for which we prove the existence of pure strategy Nash equilibria.

1. INTRODUCTION

Liquid democracy (LD) is a collective decision making method that can be understood as a midpoint between representative democracy and direct democracy. Under this paradigm, an agent can choose to either vote directly, or to select another agent acting as her proxy. Unlike classical proxy voting, the delegation is transitive, which means that the proxy may choose to delegate her voting power further, generating delegation paths along which the voting power flows and is accumulated.

To illustrate the mechanics of LD, let us assume that we have a set of five agents \{a, b, c, d, e\}, and agent a delegates her voting power to agent b, b delegates to c, c delegates to d, and agents d and e decided to not delegate their vote to anyone. With these delegation setting, the distribution of voting power in the set would be as follow: agent d has a voting power of 4 (the votes of agents c, b, and a, and her own vote), agent e has a voting power of 1, and the rest of the agents has 0 voting power. On the other hand, if agent b decides to delegate to agent e, then the voting power distribution would be: 2 for agent d, 3 for agent e, and 0 for the rest of the agents. Now, a natural question that arises is what happens if agent e decides to delegate to agent a. In such a case a delegation cycle is produced, since a delegates to b, b delegates to e, and e delegates to a. In most articles and practical implementations of LD, delegation cycles are usually ignored, considering the votes trapped in them as null.

During the last decade, several practical experiences of LD have been carried out [21]. We can mention, among others, the platform Zupa [20], used by the Student Union of the Faculty of Information Studies in Novo mesto, Slovenia, LiquidFeedback [2], used by the German Pirate Party, and Sovereign [22][Section 2], a blockchain based platform developed and used by Democracy Earth Foundation in Argentina. In any case, the employ of LiquidFeedback by the German Pirate Party

The author was partially supported by ANID through FONDECYT Project 3220254. Part of this work was performed in Centro Atómico Bariloche under a postdoctoral fellowship funded by CONICET.
is the best documented experience available in literature, and throughout this work we will consider it as a benchmark in LD practical applications.

Proposals for proxy voting models date back to Carrolls’ work \cite{12} in 1884. In the last decade these models have been object of study both in political science \cite{17} and by the artificial intelligence community \cite{8}. Since then, research has focused on several aspects of these models, from possible improvements and modifications, to possible weaknesses \cite{5,10,11,19,18}, among others. We refer to \cite{21}, and the references therein, for a comprehensive review.

**Contribution.** In most implementations of liquid democracy, and in much of the literature, each agent can either delegate all their voting power to a single agent, or cast their vote directly\footnote{The platform Sovereign \cite{22} provides each user with a finite number of votes that can be delegated independently, which can be seen as a multi-agent delegation. However, each received vote must also be dispensed independently by each proxy.}. We will refer to this paradigm as simple delegation or classical delegation. In this paper we generalize the simple delegation model in such a way that agents are able to arbitrarily divide their voting power and delegate it to multiple agents. This is carried out by means of a new voting power measure, which inherits the main properties of classical delegation. Here, as in most platforms and available works, we consider the voting power trapped in delegation cycles as null votes (see \cite{1}).

The proposed measure has the following properties:

(P1) It is a generalization of simple delegation. That is, the classical delegation is recovered when the agents decide to delegate their voting power without dividing it.

(P2) Delegation makes sense. If agent $a$ delegates her voting power to agent $b$, the distribution of voting power is equivalent to the setting where agent $a$ copies the delegation profile of agent $b$. That is, delegating to an agent is equivalent to copying her actions. Although we are being advisedly vague here, this property is formalized and proven in Theorem 3.4.

(P3) Conservation of voting power. The summation of the voting power of the agents is equal to the number of agents, for all delegation setting.

(P4) If an agent decides to delegate all the voting power, then her voting power is zero.

We note that property (P2) tells us that, for an agent, to delegate is equivalent to letting her proxy decide for her, which is intuitively what is expected when delegating. To our knowledge, this key property is not achieved by any multi-agent delegation model available in literature. We call (P2) delegation property.

On the other hand, the time variable is a key aspect that we will take into account in our analysis. As mentioned in \cite{3}, the experience in the use of LiquidFeedback shows that the behavioral change of agents based on their observations of the actions of other agents over time is relevant in the evolution of the decision-making process. Based on this, we consider the delegation profile of an agent as a function that evolves in time and depends on the feedback that the agent receives in real time. From this, it is worth asking if this process is able to reaching an equilibrium state. In this work we address this issue within the framework of game theory.

**Related work.** The model proposed by Degrave \cite{14} is possibly the most similar in spirit to the one we propose in this work, since it allows arbitrarily dividing and delegating voting power to multiple agents. Spectral type centrality measures have been proposed for measuring voting power. In this regard, we can refer to \cite{23} where the authors explore the use of PageRank as a voting power measure, and \cite{6,7}, where the authors also propose a voting power measure based
on a PageRank-type centrality measure, seeking to penalize long delegation paths. Although here the authors use simple delegation to carry out the theoretical analysis, these techniques can be easily generalized to multi-agent delegation. On the other hand, with a different approach, in [13], the authors provide a ranking over multiple delegations, and this information is used as a backup to solve possible delegation cycles. With a similar approach, we can mention [9, 10], where the delegation of power to multiple agents is studied through a ranking of preferences. Finally, we mention [16], where a technique based on fluid dynamics is proposed in order to choose a delegation graph based on the preferences of the agents. This graph attempts to minimize the concentration of power in hands of a few agents.

In Section 4 we define a game where we consider the delegation profile of each agent as a strategy, and an utility function based on the preferences of each agent over the others. This kind of games has been studied in the literature in the context of simple delegation. In this regard we can mention [15], where the authors analyze this kind of games with three different utility functions. It is possible to see that the problem treated in Section 4 is a generalization of the MINDIS problem defined in [15] (Section 3.2). Among other works that deal with similar topics we can mention [4, 24].

Organization of the paper. The rest of the paper is organized as follows. Section 2 gives an intuitive description of the proposed voting power measure, starting from its probabilistic interpretation. These ideas are formalized latter in Section 3, where we define the proposed measure and prove its main properties. Additionally we give an efficient algorithm for its computation. Finally, in Section 4 we deal, by means of game theory, with the question of existence of equilibrium states when agents are able to change their delegation profile based on the outcome of the system.

2. Preliminaries and main ideas

2.1. Preliminaries. Let \( N = \{1, ..., n\} \) be the set of agents. Each agent \( i \) expresses to whom they want to delegate their voting power, and the fraction they want to delegate, through the delegation profile \( x_i \in X \), with \( X := \{x \in \mathbb{R}_{\geq 0}^n : \sum_k x_k = 1\} \). Here \( x_{ij} \) represents the fraction of voting power that the agent \( i \) wants to delegate to agent \( j \). We define the matrix \( P := (x_1 | ... | x_n) \in \mathbb{R}^{n \times n} \) with the delegation profiles as columns, and we call it delegation matrix. Clearly \( P \) is a (left)stochastic matrix (since \( \sum_j P_{ij} = 1 \)), with the delegation profile of agent \( j \) stored in its \( j \)-th column. Given \( i \) and \( j \in N \), we say that there is a delegation path between \( i \) and \( j \) if there is a strictly positive sequence \((x_{ik_1}, x_{k_1k_2}, ..., x_{k_{q-1}k_q}, x_{k_qj})\) with \( k_1, ..., k_q \in N \). Finally, given a vector \( v \in \mathbb{R}^n \), we define the matrix \( \text{diag}(v) \in \mathbb{R}^{n \times n} \) as \( \text{diag}_{ii}(v) := v_i \) for all \( 1 \leq i \leq n \), and \( \text{diag}_{ij}(v) := 0 \) for all \( i \neq j \). Unless specified otherwise, throughout this paper vectors will always be regarded as column vectors.

2.2. The voting power measure. Denoting by \( D \subset \mathbb{R}^{n \times n} \) the set of all delegation matrices, our goal is to find a function \( V : D \to \mathbb{R}^n \) in such a way that \( V_i(P) \) represents the voting power of agent \( i \), fulfilling the conditions (P1)-(P4) enumerated in the previous section. The function \( V \) that we propose is formally defined below in Section 3, however, to give an intuitive idea, in this section we construct \( V \) starting from its probabilistic interpretation. To this end, we are going to study a particle system, dependent on the agents and their delegation preferences, and governed by the following rules:
Given a delegation matrix $P \in D$, within a time interval of length $\delta t > 0$:

- Each agent $j$ receives 1 particle with probability $\delta t$.
- A particle in position $j$ has a probability $\delta t P_{ij}$ to jump to position $i$, with $j \neq i$.
- A particle in position $j$ leaves the system with probability $\delta t P_{jj}$. In such a case we shall say that the particle was consumed by agent $j$.

We take $\delta t \to 0$, and then define $V_i(P)$ as the average amount of particles consumed by agent $i$ per unit time, once the equilibrium state is reached.

To determine the value of $V_i(P)$, we define $u_i$ as the average amount of particles in position $i$ at the equilibrium state. Using $u_i$ we can easily estimate $V_i(P)$, since particles consumed per unit time can be computed as $P_{ii} u_i$. In order to calculate $u_i$, we pose the balance equations of the system.

Following the rules above described, the amount of particles in position $i$ evolves according to the law:

$$u_i(t + \delta t) = u_i(t) + \delta t \left( \sum_{j \neq i} u_j(t) P_{ij} + 1 - u_i(t) \right),$$

$$\frac{u_i(t + \delta t) - u_i(t)}{\delta t} = \sum_{j \neq i} u_j(t) P_{ij} + 1 - u_i(t).$$

Since we look for $u_i$ at the equilibrium state, we have

$$(2.1) \quad 0 = \sum_{j \neq i} u_j P_{ij} + 1 - u_i.$$

The above expression can be written in matrix form:

$$(2.2) \quad 0 = \tilde{P} u - u + 1,$$

with $\tilde{P} = P - \text{diag}(\{P_{11}, \ldots, P_{nn}\})$.

It can be seen, at this point, that $V(P)$ satisfies most of the properties listed in the previous section. The property (P1) can be easily deduced from the equation (2.1). Indeed, assuming that no agent splits their voting power, we have $P_{ij} \in \{0, 1\}$. Considering an agent $i$ such that $P_{ii} = 1$ we can deduce

$$(2.3) \quad u_i = 1 + \sum_{k \in D} u_k,$$

where $D := \{k \in N \text{ such that } k \text{ delegates to } i\}$. Defining

$$D^* := \{k \in N \text{ such that exist a delegation path from } k \text{ to } i\},$$

and recursively applying (2.3), it can be easily seen that $u_i = 1 + \#D^*$. Since $P_{ii} = 1$, we have $V_i(P) = P_{ii} u_i = 1 + \#D^*$, which coincides with simple delegation.

Regarding (P3) we can observe that, since we define $V_i(P)$ as the average amount of particles that leave the system through the agent $i$ at the equilibrium state, $\sum_i V_i(P)$ represents the total amount of particles leaving the system per unit time. This should be equal to the number of particles entering the system per unit time, otherwise the system would not be at the equilibrium state. This amount is exactly 1 particle per agent, that is $\sum_i V_i(P) = n$. This observation is formally proven in Theorem 3.3.
On the other hand, the property (P4) can be easily verified, since if an agent $i$ delegates all her voting power we have $P_{ii} = 0$, and therefore $V_i(P) = P_{ii}u_i = 0$. It is worth mentioning that the voting models proposed in \cite{14}, as well as the spectral type centrality measures \cite{6, 7, 23}, do not fulfill this property. Finally, we can observe that to calculate which agents consume part of the voting power of a certain agent $i$, we can use the same particle system defined above, but restricting the entry of particles only through the agent $i$. That is, we compute $V(P)$ from $u = (I - \hat{P})^{-1}\delta$, where $\delta_i \in \mathbb{R}^n$, $\delta_i = 1$ and $\delta_{ij} = 0$ if $i \neq j$. As a result, $V_j(P)$ will quantify the amount of voting power consumed by agent $j$, coming from agent $i$.

2.3. Solving delegation cycles. It is clear that equation (2.2) is not always solvable. This is due to the possible existence of delegation cycles. We formally define a delegation cycle as follows:

**Definition 2.1 (Delegation cycle).** Given a set of agents $N = \{1, ..., n\}$ and their delegation profiles $x_1, ..., x_n$, we say that $C \subseteq N$ is a delegation cycle if $x_{ii} = 0$ for all $i \in C$, and given $i \in C$ and $j \in N \setminus C$, there is no delegation path from $i$ to $j$.

From this definition we can see that, being $C$ a cycle, every particle that arrives at an agent $i \in C$ cannot leave the system. To solve this problem, we propose the addition of an artificial agent to the system, the agent $n + 1$, to which every particle has a probability $\delta \varepsilon$ of jumping from any position, within a time interval $\delta t$, with $0 < \varepsilon < 1$ and $P_{n+1,n+1} = 1$. More precisely, we add the agent $n + 1$ with delegation profile $x_{n+1} = (0, ... 0, 1) \in \mathbb{R}^{n+1}$, and redefine the profiles of the rest of the agents as $x_i := ((1 - \varepsilon)x_i, \varepsilon)$, with $i = 1, ..., n$. Note that every particle that reaches agent $n + 1$ is consumed. We denote by $V^\varepsilon$ the voting power obtained in this way.

It is clear that, after this modification, delegation cycles are no longer a problem, since every particle trapped in a cycle is consumed by agent $n + 1$. However, also as a result of this modification, long delegation paths are penalized. To see this we can think of a simple delegation chain (i.e. using classical delegations) of length $k$, between agent $i$ and agent $j$. For a particle that starts from position $i$, the probability $p^k_\varepsilon$ of leaving the system through the agent $n + 1$ before reaching $j$ is $p^k_\varepsilon = 1 - (1 - \varepsilon)^k > 0$. This results in the dissipation of the voting power of agent $i$ through $n + 1$, thus penalizing long delegation paths. Likewise, $V^\varepsilon_{n+1}$ contains the sum of all the voting power dissipated both in cycles and long delegation paths. We observe that $p^k_\varepsilon \to 0$ when $\varepsilon \to 0$, however the voting power dissipated due to cycles remains invariant. To illustrate that point we can think of classical delegation cycles as infinite length delegation chains, and computing $p^\infty_\varepsilon := \lim_{k \to \infty} p^k_\varepsilon = 1$ the independence of $\varepsilon$ can be checked. Since penalizing long delegation chains is not our goal, we redefine

\begin{equation}
V(P) := \lim_{\varepsilon \to 0} V^\varepsilon(P).
\end{equation}

Thus $V_{n+1}(P)$ contains the sum of all voting power wasted in delegation cycles.  

Of course, as mentioned before, this section attempt to give an intuitive idea of $V(P)$ without dealing with formal details. Among these questions we can mention the existence of the limit in (2.4), or the preservation of properties (P1)-(P4) when $\varepsilon \to 0$. In the next section we deal with these problems and give a formal treatment of the ideas presented above.

---

\footnote{This kind of penalty is implemented in \cite{6, 7} by means of a similar damping factor.}

\footnote{We note that this definition is not suitable for the numerical computation of $V$, which is discussed in Section 3.4.}
3. Formal framework

Let $N = \{1, ..., n\}$ be a set of agents with delegation profiles $x_1, ..., x_n$, and $P$ their delegation matrix. We define

$$P^\varepsilon := \begin{pmatrix} (1 - \varepsilon)P & 0 \\ \varepsilon & 1 \end{pmatrix},$$

(3.1)

and

$$u^\varepsilon := (I - \tilde{P}^\varepsilon)^{-1}1_0,$$

where

$$\tilde{P}^\varepsilon = P^\varepsilon - \text{diag}\((P_{11}^\varepsilon, ..., P_{nn}^\varepsilon, P_{n1+1}^\varepsilon)\)$$

and $1_0 = (1, ..., 1, 0) \in \mathbb{R}^{n+1}$. Note that, since $\varepsilon > 0$ we have $\|\tilde{P}^\varepsilon\| < 1$, as a consequence $(I - \tilde{P}^\varepsilon)^{-1}$ exists and $V^\varepsilon(P)$ is well defined.

We define the voting power $V(P)$ associated to a delegation matrix $P$ as

$$V(P) := \lim_{\varepsilon \to 0} V^\varepsilon(P).$$

3.1. Well-definedness. In order to prove the existence of the limit in (3.3), we need the following auxiliary result.

**Lemma 3.1.** Let \(\{a_k\}_{k \in \mathbb{N}_0}\) and \(\{b_k\}_{k \in \mathbb{N}_0} \in \ell^\infty\) be two uniformly bounded sequences, where $a_k$ is periodic with period $q \in \mathbb{N}_0$, and $\sum_k |b_k| < \infty$. Then (i) $\lim_{\varepsilon \to 0} \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k b_k = 0$, and (ii) $\lim_{\varepsilon \to 0} \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k a_k$ exists.

**Proof.** Since $\sum_{k \geq 0} |b_k| = c$ for some constant $c$, we proceed

$$\left| \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k b_k \right| \leq \varepsilon \sum_{k \geq 0} |b_k| = \varepsilon c \to 0$$

when $\varepsilon \to 0$. And then (i) is proved.

On the other hand, since $\{a_k\}$ is periodic with period $q$, there are constants $c_j$, with $j = 0, ..., q - 1$ such that $a_{k+j} = c_j$ for all $k \geq 0$. Then, for every fixed $j$ we have

$$\sum_{k \geq 0} \varepsilon (1 - \varepsilon)^{qk+j} a_{qk+j} = c_j (1 - \varepsilon)^j \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^{qk} = \frac{c_j (1 - \varepsilon)^j \varepsilon}{1 - (1 - \varepsilon)^q}. $$

Note that the denominator in the right hand side is a polynomial $S(\varepsilon)$ of degree $q$. Also, it can be easily verified that $S(0) = 0$ and $S'(0) \neq 0$, from we can conclude that $S(\varepsilon) = \varepsilon R(\varepsilon)$, with $R$ a polynomial of degree $q - 1$, and $R(0) \neq 0$. Then

$$\frac{c_j (1 - \varepsilon)^j \varepsilon}{1 - (1 - \varepsilon)^q} = \frac{c_j (1 - \varepsilon)^j \varepsilon}{R(\varepsilon) R(0)} \to c_j \frac{R(\varepsilon)}{R(0)}$$

when $\varepsilon \to 0$. Then we can conclude that $\lim_{\varepsilon \to 0} \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k a_k = \sum_{j=0}^{q-1} \frac{c_j}{R(0)}$, and therefore (ii) is proved.

\(\square\)

Using the previous lemma, we can prove the main result of this section.
Theorem 3.1. Let \( N = \{1, ..., n\} \) be a set of agents with delegation profiles \( x_1, ..., x_n \), and \( P \) their delegation matrix. Then the limit in (3.3) is well defined.

**Proof.** We have

\[
I - \tilde{P}^\varepsilon = \begin{pmatrix} I - (1 - \varepsilon)\tilde{P} & 0 \\ -\varepsilon & 1 \end{pmatrix},
\]

with \( \tilde{P} = P - \text{diag}(P_{11}, ..., P_{nn}) \). For the sake of simplicity we define \( A_\varepsilon := I - (1 - \varepsilon)\tilde{P} \). Since \( \varepsilon > 0 \) and \( P \) is a stochastic matrix, we have \( \|(1 - \varepsilon)\tilde{P}\| < 1 \). Thus \( A_\varepsilon^{-1} \) exists, and can be computed using the identity

\[
A_\varepsilon^{-1} = \sum_{k \geq 0} (1 - \varepsilon)^k \tilde{P}^k.
\]

Applying the blockwise inversion formula, we obtain

\[
(I - \tilde{P}^\varepsilon)^{-1} = \begin{pmatrix} A_\varepsilon & 0 \\ -\varepsilon & 1 \end{pmatrix}^{-1} = \begin{pmatrix} A_\varepsilon^{-1} & 0 \\ \varepsilon A_\varepsilon^{-1} & 1 \end{pmatrix}.
\]

From we can derive

\[
u_\varepsilon = \begin{pmatrix} A_\varepsilon^{-1} & 0 \\ \varepsilon A_\varepsilon^{-1} & 1 \end{pmatrix} 0 = \begin{pmatrix} A_\varepsilon^{-1} \mathbf{1} \\ \varepsilon A_\varepsilon^{-1} \mathbf{1} \end{pmatrix},
\]

where \( \mathbf{1} = (1, ..., 1) \in \mathbb{R}^n \).

Now we focus on the computation of \( A_\varepsilon^{-1} \mathbf{1} \). Using (3.4) we obtain \( A_\varepsilon^{-1} \mathbf{1} = \sum_{k \geq 0} (1 - \varepsilon)^k \tilde{P}^k \mathbf{1} \).

In order to analyze the powers of \( \tilde{P} \) we define:

\begin{align*}
N_1 &:= \{ i \in N \text{ such that } P_{ii} \neq 0 \}, \\
N_2 &:= \{ i \in N \setminus N_1 \text{ such that exists a delegation path from } i \text{ to } j \in N_1 \}, \\
N_3 &:= N \setminus (N_1 \cup N_2).
\end{align*}

Since \( N = N_1 \cup N_2 \cup N_3 \), w.l.o.g we assume that the agents are indexed in such a way that \( N_1 \) are listed in first, agents in \( N_2 \) second, and agents in \( N_3 \) last. From this, we observe that \( \tilde{P} \) can be written as

\[
\tilde{P} = \begin{pmatrix} P_1 & 0 \\ P_2 & P_3 \end{pmatrix},
\]

with \( P_1 \) being a sub-stochastic matrix, and \( P_3 \) a stochastic matrix. It is well known that the powers of \( \tilde{P} \) can be computed as

\[
\tilde{P}^k = \begin{pmatrix} P_1^k & 0 \\ B_k & P_3^k \end{pmatrix},
\]

with \( k \in \mathbb{N} \), and \( B_k = \sum_{j=0}^{k-1} P_3^j P_2 P_1^{k-j-1} \). Then we have

\[
\tilde{P}^k \mathbf{1} = \begin{pmatrix} P_1^k \\ B_k \end{pmatrix} \mathbf{1} = \begin{pmatrix} P_1^k \mathbf{1} \\ B_k \mathbf{1} + P_3^k \mathbf{1} \end{pmatrix},
\]

as a consequence

\[
A_\varepsilon^{-1} \mathbf{1} = \sum_{k \geq 0} (1 - \varepsilon)^k \tilde{P}^k \mathbf{1} = \left( \sum_{k \geq 0} (1 - \varepsilon)^k P_1^k \mathbf{1} \right) \left( \sum_{k \geq 0} (1 - \varepsilon)^k (B_k \mathbf{1} + P_3^k \mathbf{1}) \right).
\]
Since \( P_1 \) is a sub-stochastic matrix, its spectral radius \( \rho(P_1) < 1 \). Then, each entry of \( P^k_1 \) decays exponentially, and \( \lim_{\varepsilon \to 0} \sum_{k \geq 0} (1 - \varepsilon)^k P^k_1 \) exists. Note that we do not need to compute \( \sum_{k \geq 0} (1 - \varepsilon)^k (B_k \mathbf{1} + P^k_3 \mathbf{1}) \) since these entries correspond to agents \( i \in N_3 \) with \( P_{1i} = 0 \), and then \( V^\varepsilon_i(P) = 0 \) for all \( \varepsilon > 0 \).

Finally, we need to calculate \( \lim_{\varepsilon \to 0} \varepsilon A^{-1} \varepsilon \). Since \( \varepsilon A^{-1} \varepsilon \mathbf{1} = \mathbf{1} \varepsilon A^{-1} \mathbf{1} \), we only need to analyze the behavior of \( \varepsilon A^{-1} \mathbf{1} \). Indeed, we have

\[
\varepsilon A^{-1} \mathbf{1} = \left( \sum_{k \geq 0} \frac{\varepsilon (1 - \varepsilon)^k P^k_1}{\varepsilon (1 - \varepsilon)^k (B_k \mathbf{1} + P^k_3 \mathbf{1})} \right).
\]

As we observed before, \( \lim_{\varepsilon \to 0} \sum_{k \geq 0} (1 - \varepsilon)^k P^k_1 \) is well defined. Then we focus on \( \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k (B_k \mathbf{1} + P^k_3 \mathbf{1}) \). From the definition of \( B_k \), and the fact that \( P_1 \) is a sub-stochastic matrix, it can be easily verified that each entry of \( B_k \mathbf{1} \) is an exponentially convergent sequence. Let us define \( \lim_{k \to \infty} [B_k \mathbf{1}]_i =: \ell_i \), with \( i = 1, \ldots, N + 1 \), and the vector \( L \in \mathbb{R}^N \), \( L_i := \ell_i \). Then each entry of the sequence \( B_k \mathbf{1} - L \) goes to zero exponentially fast, and we have

\[
\sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k B_k \mathbf{1} = \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k (B_k \mathbf{1} - L) + \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k L,
\]

\[
= \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k (B_k \mathbf{1} - L) + L,
\]

where in the last step we computed the geometric summation. Since every entry of \( B_k \mathbf{1} \) is a positive summable sequence, from Lemma 3.1 we conclude that \( \lim_{\varepsilon \to 0} \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k (B_k \mathbf{1} - L) = 0 \), and then

\[
\lim_{\varepsilon \to 0} \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k B_k \mathbf{1} = L.
\]

It remains to compute \( \lim_{\varepsilon \to 0} \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k P^k_3 \mathbf{1} \). For this case, we only know that \( P_3 \) is a stochastic matrix. It is well known that the powers of \( P_3 \) can be written as

\[
P^k_3 = C_1 + C^k_2 + C^k_3,
\]

where \( C_i = X J_i X^{-1} \), being \( J \) the Jordan normal form \( P_3 = X J X^{-1} \), with \( J = J_1 + J_2 + J_3 \). Here \( J_1 \) contains only the Jordan block of the eigenvalue \( \lambda = 1 \), \( J_2 \) contains the Jordan blocks of the eigenvalues \( \lambda \) with \( |\lambda| = 1 \) and \( \lambda \neq 1 \), and \( J_3 \) contains the rest of the blocks. Since \( P_3 \) is a stochastic matrix, the geometric multiplicity of each eigenvalue \( \lambda \) with \( |\lambda| = 1 \) is 1. Thus, the sequence \( C^k_{2} \mathbf{1} \) is periodic. Also, from the definition of \( J_3 \) we have that the sequence \( C^k_{3} \mathbf{1} \to \mathbf{0} \) exponentially fast. From this we have

\[
\sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k P^k_3 \mathbf{1} = \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k C_1 \mathbf{1} + \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k C^k_2 \mathbf{1} + \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k C^k_3 \mathbf{1},
\]

\[
= C_1 \mathbf{1} + \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k C^k_2 \mathbf{1} + \sum_{k \geq 0} \varepsilon (1 - \varepsilon)^k C^k_3 \mathbf{1}.
\]

From Lemma 3.1 we know that the limit when \( \varepsilon \to 0 \) exists for the second and third term. And then the theorem is proved. \( \square \)
3.2. **Generalization.** Up to this point, we are assuming that each agent plays an equal role in the decision-making process. In terms of the particle system defined in Section 2.2 this is reflected in the fact that each agent receives one particle per unit time. This in turn can be seen in the formal definition of $V$, when in (3.1) we use the vector $1_0$ to construct $u^\varepsilon$. Thus, if we want to change the inherent voting power of each agent, we simply have to modify the definition of $u^\varepsilon$ by replacing the vector $1_0$ with another vector $f \in \mathbb{R}^{n+1}$, where $f_i$ will represent the inherent voting weight of agent $i$. Then we can generalize the definitions of $u^\varepsilon$, $V^\varepsilon(P)$, and $V^\varepsilon(P)$ as follows: Given a set of agents $N = \{1, ..., n\}$ with delegation profiles $x_1, ..., x_n$, $P$ their delegation matrix, and $f \in \mathbb{R}^{n+1}$, we define

\begin{align*}
u^\varepsilon &:= (I - \tilde{P}^\varepsilon)^{-1} f, \\
V^\varepsilon(P, f) &:= (P_{11}u_1^\varepsilon, ..., P_{nn}u_n^\varepsilon, u_{n+1}^\varepsilon), \quad \text{and} \\
V(P, f) &:= \lim_{\varepsilon \to 0} V^\varepsilon(P, f).
\end{align*}

We note that the proof of Theorem (3.1) can be trivially modified to extend the result to these definitions. Then we have:

**Theorem 3.2.** Let $N = \{1, ..., n\}$ be a set of agents with delegation profiles $x_1, ..., x_n$, $P$ their delegation matrix, and $f \in \mathbb{R}^{n+1}$. Then the limit in definition (3.12) is well defined.

Although we are considering an arbitrary vector $f \in \mathbb{R}^{n+1}$, not for all $f$ there exist an interpretation within the proposed model. For example, some inputs of $f$ could take negative values, or we could have $f_{n+1} \neq 0$ so that decision power is assigned to the auxiliary agent $n + 1$. However, for the sake of generality, we develop the theory for any $f \in \mathbb{R}^{n+1}$, and the meaning of certain values of $f$ is left for a later analysis.

3.3. **Conservation of voting power.** We prove in the following result that the measure $V$ has the property (P3). This guarantees that the sum of the inherent voting power of all agents is constant under any delegation setting.

**Theorem 3.3.** Let $N = \{1, ..., n\}$ be a set of agents with delegation profiles $x_1, ..., x_n \in X$, $P$ their delegation matrix, and $f \in \mathbb{R}^{n+1}$. Then $\sum_{i=1}^{n+1} V_i(P, f) = \sum_{i=1}^{n+1} f_i$.

**Proof.** Consider $V^\varepsilon(P, f) \in \mathbb{R}^{n+1}$ defined as in (3.11) for some $\varepsilon > 0$. First, we are going to prove that $\sum_{i=1}^{n+1} V_i^\varepsilon(P, f) = \sum_{i=1}^{n+1} f_i$ for all $\varepsilon > 0$. Indeed, from definitions (3.11) and (3.10) we have

\begin{equation}
(I - \tilde{P}^\varepsilon)u^\varepsilon = f.
\end{equation}

Let $D \in \mathbb{R}^{n+1 \times n+1}$ be the diagonal matrix $D := \text{diag}((P_{11}^\varepsilon, ..., P_{nn}^\varepsilon, P_{n+1,n+1}^\varepsilon))$. Since $\tilde{P}^\varepsilon = P^\varepsilon - D$, from (3.13) we obtain

\begin{equation}
(I - \tilde{P}^\varepsilon)u^\varepsilon = u^\varepsilon - P^\varepsilon u^\varepsilon + D u^\varepsilon = f.
\end{equation}

Summing all the equations of the previous linear system we obtain

\begin{equation}
\sum_{i=1}^{n+1} u_i^\varepsilon - \sum_{i=1}^{n+1} \sum_{j=1}^{n+1} P_{ij}^\varepsilon u_j^\varepsilon + \sum_{i=1}^{n+1} V_i^\varepsilon(P, f) = \sum_{i=1}^{n+1} f_i.
\end{equation}
Since $P^\varepsilon$ is a stochastic matrix, $\sum_{i=1}^{n+1} P^\varepsilon_{ij} = 1$ for all $j$, and then
\[
\sum_{i=1}^{n+1} \sum_{j=1}^{n+1} P^\varepsilon_{ij} u_j^\varepsilon = \sum_{i=1}^{n+1} u_i^\varepsilon = \sum_{i=1}^{n+1} P^\varepsilon_{ij} = \sum_{j=1}^{n+1} u_j^\varepsilon.
\]
Combining this with the previous equality we have
\[
\sum_{i=1}^{n+1} V_i^\varepsilon(P, f) = \sum_{i=1}^{n+1} f_i.
\]
Now, since $W := \{w \in \mathbb{R}^{n+1} : \sum_{i=1}^{n+1} w_i = \sum_{i=1}^{n+1} f_i\}$ is a closed set, and $V^\varepsilon(P, f) \in W$ for all $\varepsilon > 0$, using the fact that $V(P, f) = \lim_{\varepsilon \to 0} V^\varepsilon(P, f)$ we can conclude $V(P, f) \in W$, and the proof is completed.

We note that, in particular, if $f = 1_0$ we have $\sum_{i=1}^{n+1} V_i(P) = n$, as stated in property (P3) in Section 1.

3.4. Numerical computation. It is clear that the computation of $V^\varepsilon$ requires to calculate the inverse of a bad conditioned matrix when $\varepsilon$ is small. In this regard, we observe that, from the proof of Theorem 3.1, it can be derived a procedure to calculate $V$ avoiding the computation of $V^\varepsilon$. Indeed, from (3.5) and (3.9) we deduce that we only need to compute the inverse of $(I - P_1)$, with $P_1$ being a sub-stochastic matrix. This is due to the fact that the voting power of agents in $N_3$ (and $N_2$) is zero, and then we only need to obtain the voting power of the agents belonging to $N_1$. We can summarize this observation in Algorithm 1.

**Algorithm 1 Computation of $V(P, f)$**

**Input:** A set of agents $N = \{1, ..., n\}$, their delegation profiles $x_1, ..., x_n$, and a vector $f \in \mathbb{R}^{n+1}$; 
1. Define the delegation matrix $P$ using the profiles $x_1, ..., x_n$, and $N_1$ and $N_2$ as in (3.5) and (3.6) respectively. Define $I_i$ as the $i$-th element of $N_1 \cup N_2$, with $1 \leq i \leq \#(N_1 \cup N_2)$; 
2. Define $\tilde{P} := P - \text{diag}((P_{11}, ..., P_{nn}))$, and $\tilde{P}_r$ as the restriction of $\tilde{P}$ to $N_1 \cup N_2$. That is, remove from $\tilde{P}$ the $i$-th row and the $i$-th column, for all $i \notin N_1 \cup N_2$. Analogously, define $f_r$ removing the $i$-th entry of $f$ for all $i \notin N_1 \cup N_2$.
3. If $\tilde{P}_r \neq \emptyset$, compute $u_r = (I - \tilde{P}_r)^{-1} f_r$;
4. Compute $V_i(P, f) = P_{i, I_i} [u_r]_i$ for all $1 \leq i \leq \#(N_1 \cup N_2)$, and $V_j(P, f) = 0$ for all $j \notin N_1 \cup N_2 \cup \{n + 1\}$. Finally, using Theorem 3.3 compute $V_{n+1}(P, f) = \sum_{i=1}^{n+1} f_i - \sum_{i=1}^{n} V_i(P, f)$.

3.5. The delegation property. In this section we formalize and prove the so-called delegation property (P2). This property, roughly speaking, assures us that delegation makes sense. It tells us that if an agent $i$ delegates her voting power to an agent $j$, and this agent in turn delegates all her voting power further, the distribution of power is equivalent to the one obtained when agent $i$ uses the delegation profile $x_j$ instead of $x_i$. Furthermore, if agent $i$, for example, delegates $1/4$ of her voting power to agent $j$ and $3/4$ to agent $k$, and these in turn delegate all their voting power further, the voting power distribution is equivalent to the one that agent $i$ uses as delegation profile $\frac{1}{4} x_j + \frac{3}{4} x_k$ instead of $x_i$. In these examples we are assuming that agents $j$ and $k$ do not delegate a
portion of their voting power back to agent \( i \). In such a case, that fraction of the voting power is distributed proportionally among the proxies. More precisely, we formulate the following result.

**Theorem 3.4** (Delegation property). Let \( N = \{1, \ldots, n\} \) be a set of agents with delegation profiles \( x_1, \ldots, x_n \in X \), \( P \) their delegation matrix, and \( f \in \mathbb{R}^{n+1} \). Let \( D \subset N \), with \( x_{ii} = 0 \) for all agent \( i \in D \), and assume we have agent \( k \in N \), with \( k \notin D \), in such a way that \( x_{kj} > 0 \) for all \( j \in D \) and \( x_{kj} = 0 \) otherwise. Additionally assume that agent \( k \) is not part of a delegation cycle. For all \( i \in D \), we define: \( x_i^* \in \mathbb{R}^n \) as \( x_{ij}^* := x_{ij} \) for all \( j \neq k \) and \( x_{ik}^* := 0 \); \( x_k^* \in X \) as

\[
(3.14) \quad x_k^* := \frac{\sum_{i \in D} x_{ki} x_i^*}{1 - \sum_{i \in D} x_{ki} x_{ik}}.
\]

and \( P^* \in \mathbb{R}^{n \times n} \) such that \( P^* := (x_1|\ldots|x_{k-1}|x_k^*|\ldots|x_n) \) (the delegation matrix obtained using \( x_k^* \) instead of \( x_k \)). Then, we have \( V(P, f) = V(P^*, f) \).

**Proof.** First we check that (3.14) is well defined, that is, \( 1 - \sum_{i \in D} x_{ki} x_{ik} > 0 \). Indeed, by contradiction, suppose we have \( \sum_{i \in D} x_{ki} x_{ik} = 1 \). Since \( 0 \leq x_{ik}, x_{ki} \leq 1 \) for all \( i \in N \), and \( \sum_{i \in D} x_{ki} = 1 \), we can conclude that \( x_{ik} = 1 \) for all \( i \in D \). Now, since \( x_{kj} = 0 \) for all \( j \notin D \), then \( D \cup \{k\} \) is a delegation cycle, which is a contradiction since we assume that agent \( k \) is not part of a cycle.

In order to apply the procedure of Section 3.4, we define the sets \( N_1, N_2, \) and \( N_3 \) as in (3.3), (3.6), and (3.7) respectively. Analogously, we define \( N_1^*, N_2^*, \) and \( N_3^* \) using the delegation profile \( x_k^* \) instead of \( x_k \). For the sake of simplicity, we assume that the agents in \( N \) are ordered in such a way that the ones in \( N_1 \) are listed in first, agents in \( N_2 \) second, agents in \( N_3 \) last, and \( k = \#(N_1 \cup N_2) \). Observe that every agent \( i \in N \) is part of a cycle if and only if \( i > k \). We define \( D' := D \cap (N_1 \cup N_2) \), and we observe that \( D' \neq \emptyset \), otherwise \( D \cup \{k\} \) would be a cycle.

Now, we assert that \( N_3^* = N_3 \). Indeed, at one hand, since agent \( k \) is not part of a cycle, changing her delegation profile by \( x_k^* \) cannot resolve any cycle, and then \( N_3 \subseteq N_3^* \). On the other hand, since \( D' \neq \emptyset \), there is an agent \( q \in D' \) in such a way that \( x_{qs} > 0 \), with \( s \in (N_1 \cup N_2) \). From this, and definition (3.14), it can be seen that \( x_{ks}^* > 0 \). Then agent \( k \notin N_3^* \), and we can conclude that no new delegation cycle is created using \( x_k^* \) instead of \( x_k \). Therefore \( N_3^* \subseteq N_3 \).

From the fact that \( N_3 = N_3^* \) it can be easily verified that \( N_1 = N_1^* \) and \( N_2 = N_2^* \). Then, if we apply the Algorithm 1 on \( P \) and \( P^* \), we observe that their restricted matrices, \( \tilde{P}_r \) and \( \tilde{P}^*_r \) respectively, have the same size. That is, \( \tilde{P}_r, \tilde{P}^*_r \in \mathbb{R}^{k \times k} \). In order to compute \( V(P, f) \) and \( V(P^*, f) \), we must solve the linear systems:

\[
(1 - \tilde{P}_r)u = f_r,
\]

and

\[
(1 - \tilde{P}^*_r)u^* = f_r
\]

respectively, where \( f_r \) is the restricted source term defined in Algorithm 1. We observe that we only need to verify that \( u_i = u_i^* \) for all \( i \in N_1 \), since \( V_i(P, f) = V_i(P^*, f) = 0 \) for all \( i \in N_2 \cup N_3 \).

In order to do that, we define \( A := (1 - \tilde{P}_r) \), and we consider the following change of variables:

\[
(3.15) \quad \begin{cases} u_i' = u_i - x_{ki} u_k & \text{for all } i \in D', \\ u_k' = u_k (1 - \sum_{i \in D'} x_{ki} x_{ik}), \\ u_i' = u_i & \text{for all } i \notin D' \cup \{k\}. \end{cases}
\]
From this, it can be seen that \( u' \in \mathbb{R}^k \) solves the linear system

\[
\begin{pmatrix} A_1 & \cdots & A_k \end{pmatrix} \begin{pmatrix} A_k + \sum_{i \in D'} x_{ki} A_i k \left(I - \sum_{i \in D} x_{ki} x_{ik} \right) u' = f_r,
\end{pmatrix}
\]

where \( A_i \) denotes the \( i \)-th column of \( A \). Now, we observe that \( A_i = I_i - x_i \), then

\[
A_{ki} - x_{ki} A_{ii} = x_{ki} - x_{ki} - x_{ki} x_{ii} = -x_{ki} x_{ii},
\]

and

\[
A_{kk} - x_{ki} A_{kk} = 1 - x_{ki} x_{ik}.
\]

From this, we can see that

\[
A_k + \sum_{i \in D'} x_{ki} A_i = (1 - \sum_{i \in D'} x_{ki} x_{ik}) I_k - \sum_{i \in D} x_{ki} x_i^*,
\]

From the fact that the construction of \( \tilde{P}_r \) only depends on the values \( x_i^* \) with \( i \leq k \), which in turn depend on \( x_i \) with \( i \in D' \cup \{ k \} \), we can deduce

\[
(I - \tilde{P}_r) = \begin{pmatrix} A_1 & \cdots & A_k \end{pmatrix} \begin{pmatrix} A_k + \sum_{i \in D'} x_{ki} A_i k \left(I - \sum_{i \in D} x_{ki} x_{ik} \right) \end{pmatrix}.
\]

Finally, we can conclude that \( u' = u^* \), and using (3.15), we have \( u_i = u_i' = u_i^* \) for all \( i \in N_1 \), and the theorem is proved. \( \square \)

In order to clarify the statement of Theorem 3.4 and its scope, let us consider first the example of Figure 3.1. In this example we have a delegation setting and its equivalent according to Theorem 3.4, using \( N = \{1,...,4\} \), \( k = 3 \), \( D = \{2\} \). The equivalent delegation setting is obtained replacing \( x_3 \) by \( x_3^* \) according to (3.14), and it tells us that to delegate all the voting power to agent 2 is the same as copying her delegation profile.

The example in Figure 3.2, on the other hand, shows us what results when an agent delegates to another agent that, at the same time, is delegating a fraction of her voting power back to the first agent. As in the example of Figure 3.1, in terms of Theorem 3.4, we have \( N = \{1,...,4\} \), \( k = 3 \), \( D = \{2\} \). In this case, (3.14) tells us that if agent 3 delegates all her voting power to agent 2, is the same as copying the delegation profile of agent 2, nullifying the fraction that 2 delegates to 3, and normalizing the obtained vector in order to yield a delegation profile. More precisely, since \( x_2 = (1/2,0,1/4,1/4) \) and \( x_3 = (0,1,0,0) \), from (3.14) we have \( x_3^* = (1/2,0,0,1/4)/(1-1/4) = (2/3,0,0,1/3) \).

Of course, the equivalence in both examples can be easily verified applying Algorithm 1. Also, we observe that Theorem 3.4 tells us that, if an agent delegates all her voting power among several agents, and these agents in turn delegate all their voting power further, to delegate to these agents is the same as change the delegation profile by a convex combination of their profiles. In this convex combination we use either the modified profiles as in Figure 3.2 or the unmodified profiles as in Figure 3.1, as required by equation (3.14).

Finally, we observe that using the examples in figures 3.1 and 3.2 it can be easily verified that the measures proposed in [14], as well as the spectral centrality measures [6, 7, 23], do not satisfy the delegation property.
According to Theorem 3.4, the delegation settings shown above are both equivalent. It can be seen that to delegate all the voting power to agent 2 is the same as copying her delegation profile.

Theorem 3.4 tells us that the delegation setting at the left is equivalent to the one at the right. That is, for agent 3, to delegate the voting power to agent 2 has the same effect as copying her the delegation profile, and distributing the voting power that agent 2 delegates back to 3, proportionally among her proxies.

4. Delegation Game

As mentioned above, in this paper we consider the decision-making process not as a single event, but as a continuous dynamic over time. That is, the agents delegate power or make use of it, impacting these actions on the rest of the agents, which in turn use this feedback to modify their delegation profiles and their actions. This leads us to the question of whether equilibrium states exist for this kind of mechanics.

In order to partially model this dynamic, we define the game $G(N, W, \varepsilon)$, where $N = \{1, \ldots, n\}$ is the set of agents and $W = \{w_1, \ldots, w_n\}$, with $w_i \in \mathbb{R}^{n+1}$, the set of preferences. Here $w_i \in W$ gives to each agent a weight, denoting the preferences of agent $i$, that is, the higher $w_{ij}$, the more satisfied is agent $i$ with the fact that $j$ consumes her voting power (note that $w_i$ could have negative entries). The strategy of each agent $i$ is given by its delegation profile $x_i \in X$. Each agent seeks to maximize its associated utility function $U_i : X \rightarrow \mathbb{R}$,

$$U_i(x, x_{-i}) := \sum_{j=1}^{n+1} w_{ij} V^\varepsilon((x|x_{-i}), \delta_i) = w_i \cdot V^\varepsilon((x|x_{-i}), \delta_i).$$

Here $\delta_i \in \mathbb{R}^{n+1}$ is such that $\delta_{ii} = 1$ and $\delta_{ik} = 0$ if $i \neq k$, and $(x|x_{-i})$ denotes the matrix $(x_1, \ldots, x_i, \ldots, x_n)$ with column $x$ in the $i$-th place, where $x_k$ is the delegation profile of agent $k$ with $k \neq i$. We take $0 < \varepsilon < 1$. As we have seen in Section 2.2, $V^\varepsilon(P, \delta_i) \in \mathbb{R}$ tells us how much voting
Consider two strategies guarantees the existence of a fixed point if the following conditions are satisfied: want to prove that continuous function for all if the following conditions are satisfied: Let us consider and Let Theorem 4.1. For the sake of convenience we consider the elements of Proof. From the definition of we have: Since \( f_i \geq 0 \) for all \( i \), it can be easily verified that \( u^\varepsilon \geq 0 \). From (4.1) we have \[ u_j^\varepsilon - (1 - \varepsilon) \sum_{i \neq j} P_{ji} u_i^\varepsilon = f_j > 0, \quad \text{if } j < n + 1. \] From we can conclude that \( u_j^\varepsilon > 0 \) if \( j < n + 1 \). The case \( j = n + 1 \) can be treated analogously. □

Finally, we state the main result of this section.

**Theorem 4.1.** Let \( N = \{1, \ldots, n\} \) be a set of agents with delegation profiles \( x_1, \ldots, x_n \in X \), \( P \) their delegation matrix, and \( f \in \mathbb{R}^{n+1} \) in such a way that \( f_i \geq 0 \) for all \( 1 \leq i \leq n + 1 \) and \( f_j > 0 \) for some \( 1 \leq j \leq n + 1 \). Then \( u_j^\varepsilon > 0 \) for all \( \varepsilon > 0 \).

**Proof.** For the sake of convenience we consider the elements of \( W \) as row vectors, that is \( W \subset \mathbb{R}^{1 \times n+1} \). Let \( r_i(x_{-i}) \) be the best response of agent \( i \) to the strategies of all other agents:

\[ r_i(x_{-i}) := \arg \max_{x_i} U_i(x_i, x_{-i}). \]

Let us consider \( x \in X := X^n = X \times \ldots \times X \), and define the set valued function \( r : X \to 2^X \), \( r(x) := r_1(x_{-1}) \times \ldots \times r_n(x_{-n}) \). Note that every fixed point of \( r \) is a Nash equilibrium. Then we want to prove that \( r \) has a fixed point. To this end, we use Kakutani’s fixed point theorem, which guarantees the existence of a fixed point if the following conditions are satisfied:

1. \( X \) is compact, convex, and nonempty,
2. \( r(x) \) is nonempty,
3. \( r(x) \) is upper hemicontinuous,
4. \( r(x) \) is convex.

Condition [1] is trivially satisfied. Conditions [2] and [3] can be verified using the fact that \( U_i \) is a continuous function for all \( i \) and for all \( 0 < \varepsilon < 1 \), combined with Berge’s maximum theorem.

To prove [1] from the definition of \( r(x) \), we only need to see that \( r_i(x_{-i}) \) is convex for all \( i \). Consider two strategies \( x \) and \( x' \in r_i(x_{-i}) \). Defining \( \eta := x'-x \) and \( x(t) := x + t\eta \), we want to see
that \( x(t) \in r_i(x_{-i}) \) for all \( t \in [0, 1] \). For the sake of simplicity and w.l.o.g, we assume \( i = 1 \). From the definition \((3.2)\), if we define \( P(t) := (x(t)|x_{-1}) \), the utility function \( U_1 \) can be written as
\[
U_1(x(t), x_{-1}) = w_1D_tu(t),
\]
where \( D_t := \text{diag}((P_{11}^x(t), ..., P_{nn}^x(t), P_{n+1n+1}^x(t))) \), and \( u(t) := (I - \tilde{P}^x(t))^{-1}\delta_1 \). We observe that \( u(t) \) satisfies the equation
\[
\begin{pmatrix}
1 - (1 - \varepsilon)\tilde{P}(t) & 0 \\
\varepsilon & 1
\end{pmatrix} u(t) = \delta_1,
\]
with \( \tilde{P}(t) = P(t) - \text{diag}((P_{11}(t), ..., P_{nn}(t))) \). Using the definition of \( x(t) \), we can rewrite the former expression as
\[
\left[ \begin{pmatrix}
1 - (1 - \varepsilon)\tilde{P}(0) & 0 \\
\varepsilon & 1
\end{pmatrix} - \begin{pmatrix}
(1 - \varepsilon)t\tilde{\eta} & 0 \\
0 & 0
\end{pmatrix} \right] u(t) = \delta_1,
\]
where \( \tilde{\eta} \in \mathbb{R}^n \), with \( \tilde{\eta}_i := \eta_i \) for all \( i > 1 \) and \( \tilde{\eta}_1 := 0 \). Defining
\[
A_0 := \begin{pmatrix}
1 - (1 - \varepsilon)\tilde{P}(0) & 0 \\
\varepsilon & 1
\end{pmatrix} \quad \text{and} \quad \Delta := \begin{pmatrix}
(1 - \varepsilon)\tilde{\eta} & 0 \\
0 & 0
\end{pmatrix},
\]
we have \((A_0 - t\Delta)u(t) = \delta_1\), and then
\[
u(t) = A_0^{-1}\delta_1 + tA_0^{-1}\Delta u(t).
\]
Also, we observe that \( D_t = D_0 + tD_\Delta \), with
\[
D_\Delta := \begin{pmatrix}
x_1^t & x_1 & 0 \\
x_0^t & 0 & 0
\end{pmatrix}.
\]

On the other hand, consider a constant vector \( c \in \mathbb{R}^{1 \times n+1} \), with \( c_j = C \in \mathbb{R} \) for all \( j \). Defining \( w^* = w_1 + c \) we observe that maximize (or minimize) the utility function \( w_1D_tu(t) \) is equivalent to maximize (or minimize) the utility function \( U^*(x(t)|x_{-1}) := w^*D_tu(t) \). Indeed, we have \( w^*D_tu(t) = w_1D_tu(t) + cD_tu(t) \), and
\[
cD_tu(t) = c \cdot V^*((x(t)|x_{-1}), \delta_1) = C \sum_{j=1}^{n+1} V_j^*((x(t)|x_{-1}), \delta_1) = C,
\]
where in the last step we use Theorem \((3.3)\) Then \( U^*(x(t)|x_{-1}) = U(x(t)|x_{-1}) + C \), therefore
\[
\arg \max_x U_1(x, x_{-1}) = \arg \max_x U^*(x, x_{-1}).
\]

Now, we choose \( c \in \mathbb{R}^{1 \times n+1} \) with \( c_j = -w_{11} \) for all \( j \), so that \( w_1^* = 0 \). As a consequence, from the definition of \( w^* \) and \((4.4)\), we have \( w^*D_\Delta = 0 \), thus
\[
U^*(x(t)|x_{-1}) = w^*D_tu(t) = w^*(D_0 + D_\Delta)u(t) = w^*D_0u(t),
\]
and using \((4.3)\) we obtain
\[
U^*(x(t)|x_{-1}) = w^*D_0A_0^{-1}\delta_1 + tw^*D_0A_0^{-1}\Delta u(t).
\]
From the definition of \( \Delta \) \((4.2)\) we have
\[
\Delta u(t) = u_1(t) \begin{pmatrix}
\tilde{\eta} \\
0
\end{pmatrix},
\]
and we can rewrite (4.6) as follow

\[(4.7)\]
\[
U^*(x(t)|x_{-1}) = w^*D_0A_0^{-1}\delta_1 + tu_1(t)w^*D_0A_0^{-1}\begin{pmatrix} \hat{\eta} \\ 0 \end{pmatrix}.
\]

Since \(x\) and \(x'\) are maximizers of \(U_1\), from (4.5) we know that they are also maximizers of \(U^*\), then

\[
U^*(x(0)|x_{-1}) = U^*(x(1)|x_{-1}),
\]
\[
w^*D_0A_0^{-1}\delta_1 = w^*D_0A_0^{-1}\delta_1 + u_1(1)w^*D_0A_0^{-1}\begin{pmatrix} \hat{\eta} \\ 0 \end{pmatrix},
\]
\[
0 = u_1(1)w^*D_0A_0^{-1}\begin{pmatrix} \hat{\eta} \\ 0 \end{pmatrix}.
\]

From Lemma 4.1 we have \(u_1(1) > 0\), then

\[
w^*D_0A_0^{-1}\begin{pmatrix} \hat{\eta} \\ 0 \end{pmatrix} = 0,
\]

and therefore, using (4.7), \(U^*(x(t)|x_{-1}) = U^*(x(0)|x_{-1})\) for all \(t \in [0,1]\). Thus \(x(t)\) is a maximizer of \(U^*\) and, from (4.6), we can conclude that \(x(t)\) is a maximizer of \(U_1\) for all \(t \in [0,1]\). As a consequence \(r^1(x_{-1})\) is a convex set. Analogously, \(r_i(x_{-i})\) is a convex set for all \(1 \leq i \leq n+1\), and the theorem is proved. \(\Box\)

Remark 2. We note that if we restrict agents’ strategies to convex, compact, and non-empty spaces, Theorem 4.1 can be trivially reproduced. For example, we can consider a social network in which \(N_i \subseteq N\) denotes the neighbors of the agent \(i\), and restrict the strategies of each agent to its neighborhood, that is \(x_i \in X_i := \{x \in \mathbb{R}_{\geq 0}^n : x_k = 0 \forall k \notin N_i, \text{ and } \sum_k x_k = 1\}\). For this case the proof of the Theorem 4.1 can be easily replicated using 

\[
X := X_1 \times \ldots \times X_n.
\]

This setting can be seen as a generalization of problem MINDIS, form [14] [Section 3.2].

5. Conclusions

We proposed a way to measure voting power in a liquid democracy model where agents are able to arbitarily divide and delegate their vote to multiple agents. This measure of voting power, formally defined in Section 3, generalizes the classical LD model along with its main properties. We proved that this measure is well defined and satisfies the properties (P1)-(P4). Particularly we focused on the so-called delegation property. This property, formally defined in Theorem 3.4, roughly tells us that delegating voting power to an agent is equivalent to copying her delegation profile. We observed that the voting measure proposed in [14], as well as the spectral centrality measures proposed in [6] [7] and [23], do not satisfy this property.

On the other hand, in Section 4 we considered the delegation profile of each agent as a function that evolves in time, which is modified by means of the feedback that each agent receives from the system. We posd the problem of the existence of an equilibrium state and we addressed this problem by defining a delegation game, where the strategy of each agent consists of choosing a delegation profile. Here, the utility function of each agent is given by the voting power distribution \(V^\varepsilon\) weighted by the agents’ preferences. We observed that both \(V\) and \(V^\varepsilon\) are not equivalent to the measure given by mixed strategies from simple delegations. Finally we proved the existence of pure strategy Nash equilibria for the proposed game.
References

[1] J. Behrens. Circular delegations – myth or disaster? The Liquid Democracy Journal, 3, 2015.
[2] J. Behrens, A. Kistner, A. Nitsche, and B. Swierczek. The principles of LiquidFeedback. Interaktive Demokratie, 2014.
[3] J. Behrens, A. Kistner, A. Nitsche, and B. Swierczek. The temporal dimension in the analysis of liquid democracy delegation graphs. Interaktive Demokratie, 2021.
[4] D. Bloembergen, D. Grossi, and M. Lackner. On rational delegations in liquid democracy. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 33, pages 1796–1803, 2019.
[5] C. Blum and C. I. Zuber. Liquid democracy: Potentials, problems, and perspectives. Journal of Political Philosophy, 24(2):162–182, 2016.
[6] P. Boldi, F. Bonchi, C. Castillo, and S. Vigna. Voting in social networks. In Proceedings of the 18th ACM conference on Information and knowledge management, pages 777–786, 2009.
[7] P. Boldi, F. Bonchi, C. Castillo, and S. Vigna. Viscous democracy for social networks. Communications of the ACM, 54(6):129–137, 2011.
[8] F. Brandt, V. Conitzer, U. Endriss, J. Lang, and A. D. Procaccia. Handbook of computational social choice. Cambridge University Press, 2016.
[9] M. Brill, T. Delemazure, A.-M. George, M. Lackner, and U. Schmidt-Kraepelin. Liquid democracy with ranked delegations. In Proceedings of the AAAI Conference on Artificial Intelligence, volume 36, pages 4884–4891, 2022.
[10] M. Brill and N. Talmon. Pairwise liquid democracy. In IJCAI, volume 18, pages 137–143, 2018.
[11] I. Caragiannis and E. Micha. A contribution to the critique of liquid democracy. In IJCAI, pages 116–122, 2019.
[12] L. Carroll. The principles of parliamentary representation. Harrison and Sons, 1884.
[13] R. Colley, U. Grandi, and A. Novaro. Unravelling multi-agent ranked delegations. Autonomous Agents and Multi-Agent Systems, 36(1):1–35, 2022.
[14] J. Degrave. Resolving multi-proxy transitive vote delegation. arXiv preprint arXiv:1412.4039, 2014.
[15] B. Escoffier, H. Gilbert, and A. Pass-Lanneau. The convergence of iterative delegations in liquid democracy in a social network. In International Symposium on Algorithmic Game Theory, pages 284–297. Springer, 2019.
[16] P. Gölz, A. Kahng, S. Mackenzie, and A. D. Procaccia. The fluid mechanics of liquid democracy. ACM Transactions on Economics and Computation, 9(4):1–39, 2021.
[17] J. Green-Armytage. Direct voting and proxy voting. Constitutional Political Economy, 26(2):190–220, 2015.
[18] D. Halpern, J. Y. Halpern, A. Jadbabaie, E. Mossel, A. D. Procaccia, and M. Revel. In defense of liquid democracy. arXiv preprint arXiv:2107.11868, 2021.
[19] A. Kahng, S. Mackenzie, and A. Procaccia. Liquid democracy: An algorithmic perspective. Journal of Artificial Intelligence Research, 70:1223–1252, 2021.
[20] A. Paulin. Župa-grassroots e-democracy revolution on the web. In International Conference on E-Democracy, pages 113–123, 2010.
[21] A. Paulin. An overview of ten years of liquid democracy research. In The 21st Annual International Conference on Digital Government Research, pages 116–121, 2020.
[22] S. Siri, V. Deville, P. Berman, E. Medina, H. Stephens, S. Miller, A. Bacelar, D. Wilson, M. Williams, L. Margot-Duclot, et al. The social smart contract. The Democracy Earth Foundation, 2018.
[23] H. Yamakawa, M. Yoshida, and M. Tsuchiya. Toward delegated democracy: Vote by yourself, or trust your network. International Journal of Human and Social Sciences, 1(2):146–150, 2007.
[24] Y. Zhang and D. Grossi. Power in liquid democracy. In Proceedings of the AAAI conference on Artificial Intelligence, volume 35, pages 5822–5830, 2021.

(F.M. Bersetche) DEPARTAMENTO DE MATEMÁTICA, UNIVERSIDAD TÉCNICA FEDERICO SANTA MARÍA, AV. ESPAÑA 1680, VALPARAÍSO, CHILE.

Email address: bersetche@gmail.com