An attempt to give the most abstract possible definition of a blockchain.

1 BlockChain

1.1 Lists and their validation

Given a set S, let $\textsc{Set}(S)$ be the set of sets of elements of S and $\textsc{FList}(S)$ be the set of all finite lists of elements of S. Given $L \in \textsc{FList}(S)$, we use notation $\text{length}(L)$ to refer to the number of elements in L, notation $L[i]$ to refer to the i-th element in L, where $i \in \{1, \ldots, \text{length}(L)\}$, and notation $L[i, j]$ to refer to the sublist $[L[i], \ldots, L[j]]$ of L, where $i, j \in \{1, \ldots, \text{length}(L)\}$ and $i \leq j$. Notice that $\text{length}(L) = 0$ if and only if L is the empty list $[\,]$. Finally, we say that a list L_1 is a prefix of a list L_2 if L_1 is the empty list, or $1 \leq \text{length}(L_1) \leq \text{length}(L_2)$ and $L_1 = L_2[1, \text{length}(L_1)]$.

From now on, assume that Σ is a finite alphabet, and that $\mathbf{B} \subseteq \Sigma^*$ is the set of all possible blocks.

Definition 1. *A validation rule is a function* $V : \textsc{FList}(\mathbf{B}) \rightarrow \textsc{Set}(\mathbf{B})$

Intuitively, V is a function taking a finite list L of blocks as input, and returning the set of blocks that could be added to L to produce a valid blockchain.

A list $G \in \textsc{FList}(\mathbf{B})$ is said to be a genesis list of V if $\text{length}(G) \geq 1$, $G[1] \in V([\,])$ and $G[i+1] \in V(G[1, i])$, for every $i \in \{1, \ldots, \text{length}(G) - 1\}$. That is, G is a genesis list if G is a non-empty valid blockchain.

Definition 2. *Let V be a validation rule and G be a genesis list of V. Then a list $L \in \textsc{FList}(\mathbf{B})$ is valid with respect to (G, V) if:*

1. $\text{length}(G) \leq \text{length}(L)$ *and* $G = L[1, \text{length}(G)]$.
2. $L[i+1] \in V(L[1, i])$, *for every* $i \in \{\text{length}(G), \ldots, \text{length}(L) - 1\}$.

The role of G in this definition is to provide the blocks to startup the system. Let $\textsc{Log}(G, V)$ be the set of valid lists with respect to (G, V).

Two lists $L_1, L_2 \in \textsc{FList}(\mathbf{B})$ are said to disagree in the last element if one of the following conditions holds: (1) $\text{length}(L_1) = 0$ and $\text{length}(L_2) > 0$, (2) $\text{length}(L_1) > 0$ and $\text{length}(L_2) = 0$, or (3) $\text{length}(L_1) > 0$, $\text{length}(L_2) > 0$ and $L_1[\text{length}(L_1)] \neq L_2[\text{length}(L_2)]$.

Definition 3. *Let V be a validation rule and G be a genesis list of V. Then $\textsc{Log}(G, V)$ is safe if for every pair $L_1, L_2 \in \textsc{FList}(\mathbf{B})$ that disagree in the last element, it holds that $V(L_1) \cap V(L_2) = \emptyset$.*

1.2 Body of knowledge

In this document, we will give a game-theoretic characterization of the notion of blockchain where it plays a key role the knowledge of each participant. More precisely, given a validation rule V and a genesis list G of V, a body of knowledge of (G, V) is a non-empty and finite subset K of $\textsc{Log}(G, V)$ satisfying the following closure property:

- if $L_1 \in \textsc{Log}(G, V)$, L_1 is a prefix of L_2 and $L_2 \in K$, then $L_1 \in K$.

Intuitively, if at some iteration a participant considers a list $L \in \textsc{Log}(G, V)$ as valid, then she should also consider as valid every prefix of L including the genesis list, that is, every prefix of L belonging to $\textsc{Log}(G, V)$. The set of bodies of knowledge of (G, V) is denoted by $\text{BK}(G, V)$.

There is a natural way to visualize a body of knowledge K as a graph $\mathcal{G}(K)$. The set of nodes of $\mathcal{G}(K)$ is the set of blocks occurring in the lists in K, and there is an edge from a block b_1 to a block b_2 if there exists a list $L \in K$ such that $b_1 = L[i]$ and $b_2 = L[i+1]$, where $i \in \{1, \ldots, \text{length}(L) - 1\}$.

Lemma 1. *Assume that $\textsc{Log}(G, V)$ is safe. Then for every $K \in \text{BK}(G, V)$, it holds that $\mathcal{G}(K)$ is a tree rooted at $G[1]$.*

1.3 Protocols and blockchain

Definition 4. *A relation $\preceq$ on $\textsc{Log}(G, V)$ is said to be a knowledge order over (G, V) if $\preceq$ is a total preorder on $\textsc{Log}(G, V)$, that is, $\preceq$ is reflexive, transitive and total.*

Moreover, a sequence $\{\preceq_i\}_{i\in\mathbb{N}}$ is said to be a blockchain protocol over (G, V) if every $\preceq_i$ $(i \in \mathbb{N})$ is a knowledge order over (G, V).

Definition 5. *Let $\{\preceq_i\}_{i\in\mathbb{N}}$ be a blockchain protocol over (G, V), $K \in \mathrm{BK}(G, V)$ and $t \in \mathbb{N}$. Then a maximal element of K with respect to $\preceq_t$ is said to be a blockchain of K at iteration t with respect to the protocol $\{\preceq_i\}_{i\in\mathbb{N}}$.*

1.4 Definition of the game

Fix a validation rule V and a genesis list G of V. From now on, we assume that $\mathcal{P} = \{1, \ldots, n\}$ is a finite set of players, and we say that a state $\mathbf{q}$ is a tuple $(q_1, \ldots, q_n) \in \mathrm{BK}(G, V)^n$. Intuitively, each component q_i of $\mathbf{q}$ represents the knowledge of player i, so $\mathbf{q}$ contains the knowledge of all the players. Moreover, we denote by $\mathcal{Q}$ the set of all possible states, that is, $\mathcal{Q} = \mathrm{BK}(G, V)^n$.

Definition 6. *Given a player $p \in \mathcal{P}$, a function $a : \mathcal{Q} \to \mathcal{Q}$ is an action for p if*

- *for every $\mathbf{q} \in \mathcal{Q}$ and $p' \in \mathcal{P}$, if $\mathbf{q} = (q_1, \ldots, q_n)$ and $a(\mathbf{q}) = (q'_1, \ldots, q'_n)$, then it holds that:*

$$q_{p'} \subseteq q'_{p'} \subseteq q_{p'} \cup q'_p.$$

Moreover, $\mathcal{A}_p$ is the set of all actions for player p.

An action of a player p is represented by a modification of the knowledge of p and a round of communication between players.

If we need to restrict the number of blocks that can be added when an action is executed (like in the case of Bitcoin), then we need to include in Definition 6 a condition like the following:

- for every $\mathbf{q} \in \mathcal{Q}$, if $\mathbf{q} = (q_1, \ldots, q_n)$ and $a(\mathbf{q}) = (q'_1, \ldots, q'_n)$, then it holds that $|q'_p| \leq |q_p| + 1$.

In this case, at most one block can be added as the result of executing action a by player p.

From now on, assume that $\mathcal{A} = \mathcal{A}_1 \times \mathcal{A}_2 \times \cdots \times \mathcal{A}_n$. Thus, every element of $\mathbf{a} \in \mathcal{A}$ is a tuple containing exactly one action for each player. Moreover, given a player $p \in \mathcal{P}$, a function $r_p : \mathcal{Q} \times \mathcal{A} \to \mathbb{R}$ is called a pay-off function for p. Intuitively, given $(\mathbf{q}, \mathbf{a}) \in \mathcal{Q} \times \mathcal{A}$, we have that $r_p(\mathbf{q}, \mathbf{a})$ is the pay-off of player p when the set of actions to be executed is $\mathbf{a}$ and the knowledge of each player is encoded in $\mathbf{q}$. Finally, assuming that there is a function r_p for each player $p \in \mathcal{P}$, define $\mathcal{R} = (r_1, \ldots, r_n)$ as the pay-off function of the game.

As a last component of the game, we assume that $\mathbf{Pr} : \mathcal{Q} \times \mathcal{A} \times \mathcal{Q} \to [0, 1]$ is a transition probability function satisfying the following conditions:

1. For every $\mathbf{q} \in \mathcal{Q}$ and $\mathbf{a} \in \mathcal{A}$:

$$\sum_{\mathbf{q}' \in \mathcal{Q}} \mathbf{Pr}(\mathbf{q}, \mathbf{a}, \mathbf{q}') = 1.$$

2. For every $\mathbf{q} \in \mathcal{Q}$, $\mathbf{a} \in \mathcal{A}$ and $\mathbf{q}' \in \mathcal{Q}$, it holds that $\mathbf{Pr}(\mathbf{q}, \mathbf{a}, \mathbf{q}') = 0$ if $\mathbf{a} = (a_1, \ldots, a_n)$ and $a_i(\mathbf{q}) \neq \mathbf{q}'$ for every $i \in \{1, \ldots, n\}$.

Intuitively, $\mathbf{Pr}(\mathbf{q}, \mathbf{a}, \mathbf{q}')$ tell us what the probability of generating $\mathbf{q}'$ from $\mathbf{q}$ is when one of the actions in the tuple $\mathbf{a}$ is executed.

If the knowledge of all the players is given by $\mathbf{q} \in \mathcal{Q}$, and a player p decides to execute an action a_p, its rewards not only depends on $\mathbf{q}$ and a_p, but also on the actions to be executed by the other players. If the tuple of actions to be executed by all the players is $\mathbf{a} = (a_1, \ldots, a_n)$, then the computation of the value $r_p(\mathbf{q}, \mathbf{a})$ should take into consideration the knowledge in $\mathbf{q}$ and the probability that the action executed is a_p. Thus, intuitively, if player p foresees to receive $C(\mathbf{q}, a_p)$ as reward, then we should have that $r_p(\mathbf{q}, \mathbf{a}) = C(\mathbf{q}, a_p) \cdot \mathbf{Pr}(\mathbf{q}, \mathbf{a}, a_p(\mathbf{q}))$.

Summing up, we consider an infinite stochastic game $\Gamma = (\mathcal{P}, \mathcal{A}, \mathcal{Q}, \mathcal{R}, \mathbf{Pr})$ where:

- $\mathcal{P}$ is the set of player.
- $\mathcal{A}$ is the set of available action.
- $\mathcal{Q}$ is the set of states.
- $\mathcal{R}$ is the pay-off function.
- $\mathbf{Pr}$ is the transition probability function.

1.5 Stationary equilibrium

A stationary strategy for a player p a function $s : \mathcal{Q} \to \mathcal{A}_p$. Moreover $\mathcal{S}_p$ is the set of all strategy for player p, and $\mathcal{S} = \mathcal{S}_1 \times \mathcal{S}_2 \times \cdots \times \mathcal{S}_n$ is the set of strategies for the game. Thus, every element of $\mathrm{s} \in \mathcal{S}$ is a tuple containing exactly one strategy for each player. Finally, assuming that $\mathrm{s} = (s_1, \ldots, s_n)$ and $\mathbf{q} \in \mathcal{Q}$, define $\mathrm{s}(\mathbf{q})$ as the tuple of actions $(s_1(\mathbf{q}), \ldots, s_n(\mathbf{q}))$.

Given an initial state $\mathbf{q}_0 \in \mathcal{Q}$ and a strategy $\mathrm{s} \in \mathcal{S}$, the probability of reaching state $\mathbf{q} \in \mathcal{Q}$ in k iterations is recursively defined as follows:

$$\begin{aligned}
\mathbf{Pr}^{\mathbf{s}}_0(\mathbf{q} \mid \mathbf{q}_0) &= \begin{cases} 1 & \text{if } \mathbf{q} = \mathbf{q}_0 \\ 0 & \text{otherwise} \end{cases} \\
\mathbf{Pr}^{\mathbf{s}}_{k+1}(\mathbf{q} \mid \mathbf{q}_0) &= \sum_{\mathbf{q}' \in \mathcal{Q}} \mathbf{Pr}^{\mathbf{s}}_k(\mathbf{q}' \mid \mathbf{q}_0) \cdot \mathbf{Pr}(\mathbf{q}', \mathrm{s}(\mathbf{q}'), \mathbf{q}) \qquad \text{for every } k \in \mathbb{N}
\end{aligned}$$

Definition 7. *Let $p \in \mathcal{P}$, $\mathbf{q}_0 \in \mathcal{Q}$, $\mathrm{s} \in \mathcal{S}$ and $\beta \in [0, 1]$. Then the β discounted pay-off of the player p for the strategy s from the state $\mathbf{q}_0$, denoted by $u_p(\mathrm{s} \mid \mathbf{q}_0)$, is defined as:*

$$u_p(\mathbf{s} \mid \mathbf{q}_0) = (1 - \beta) \cdot \sum_{i=0}^{\infty} \beta^i \cdot \left(\sum_{\mathbf{q} \in \mathcal{Q}} r_p(\mathbf{q}, \mathbf{s}(\mathbf{q})) \cdot \mathbf{Pr}^{\mathbf{s}}_i(\mathbf{q} \mid \mathbf{q}_0) \right)$$

Given $p \in \mathcal{P}$, $\mathrm{s} \in \mathcal{S}$, with $\mathrm{s} = (s_1, \ldots, s_n)$, and $s \in \mathcal{S}_p$, we denote by $(\mathbf{s}_{-p}, s)$ strategy of the game $(s_1, \ldots s_{p-1}, s, s_{p+1}, \ldots, s_n)$.

Definition 8. *Let $\mathbf{q}_0 \in \mathcal{Q}$, $\mathbf{s} \in \mathcal{S}$ and $\beta \in [0, 1]$. Then $\mathbf{s}$ is a β discounted stationary equilibrium from the state $\mathbf{q}_0$ if for every player $p \in \mathcal{P}$ and every strategy s for player p ($s \in \mathcal{S}_p$), it holds that:*

$$u_p(\mathbf{s} \mid \mathbf{q}_0) \geq u_p((\mathbf{s}_{-p}, s) \mid \mathbf{q}_0).$$

1.6 Properties of a blockchain

Given an initial state $\mathbf{q}_0 \in \mathcal{Q}$ and a strategy $\mathrm{s} \in \mathcal{S}$, the probability of reaching a state of $\mathbf{Q} \in \mathrm{SET}\mathcal{Q}$ without walking by a state of $\mathbf{Q}' \in \mathrm{SET}\mathcal{Q}$ in k iterations is recursively defined as follows:

$$\begin{aligned}
\mathbf{Pr}^{\mathbf{s}}_0(\mathbf{Q}, \mathbf{Q}' \mid \mathbf{q}_0) &= \begin{cases} 1 & \text{if } \mathbf{q}_0 \in \mathbf{Q} \\ 0 & \text{otherwise} \end{cases} \\
\mathbf{Pr}^{\mathbf{s}}_{k+1}(\mathbf{Q}, \mathbf{Q}' \mid \mathbf{q}_0) &= \sum_{\mathbf{q}' \in \mathcal{Q} \setminus \mathbf{Q}'} \mathbf{Pr}^{\mathbf{s}}_k(\{\mathbf{q}'\}, \mathbf{Q}' \mid \mathbf{q}_0) \cdot \sum_{\mathbf{q} \in \mathbf{Q}} \mathbf{Pr}(\mathbf{q}', \mathrm{s}(\mathbf{q}'), \mathbf{q}) \qquad \text{for every } k \in \mathbb{N}
\end{aligned}$$

Lemma 2. *Given an initial state $\mathbf{q}_0 \in \mathcal{Q}$ and a strategy $\mathbf{s} \in \mathcal{S}$, the probability to reach a state of $\mathbf{Q} \in \mathrm{SET}(\mathcal{Q})$ for the first time in k-step is equal to $\mathbf{Pr}^{\mathbf{s}}_k(\mathbf{Q}, \mathbf{Q} \mid \mathbf{q}_0)$*

Proof. immediate. □

Property 1. *Given an initial state $\mathbf{q}_0 \in \mathcal{Q}$ and a strategy $\mathbf{s} \in \mathcal{S}$, the probability to reach a state of $\mathbf{Q} \in \textsc{Set}(\mathcal{Q})$ noted $\mathbf{Pr^s} : \textsc{Set}(\mathcal{V}) \to [0;1]$ is equal to*

$$\mathbf{Pr^s}(\mathbf{Q} \mid \mathbf{q}_0) = \sum_{i=0}^{+\infty} \mathbf{Pr}_i^{\mathbf{s}}(\mathbf{Q}, \mathbf{Q} \mid \mathbf{q}_0)$$

Proof. to do. □

Let P be a property over states, let $\mathbf{q} \in \mathcal{Q}$ we denote $\mathbf{q} \vdash P$ if $\mathbf{q}$ satisfies the property P.

Definition 9. *Given an initial state $\mathbf{q}_0 \in \mathcal{Q}$ and a property P We say that P is verified by (G, V), $\{\preceq_i\}_{i \in \mathbb{N}}$ regarding Γ with a probability α if and only if:*

- *Exists a β discounted stationary equilibrium of Γ*
- *Forall β discounted stationary equilibrium of Γ $\mathbf{s} \in \mathcal{S}$ we have :*

$$\forall \mathbf{Q} \in \textsc{Set}\mathcal{Q}, (\mathbf{Prs}(\mathbf{Q} \mid \mathbf{q}_0) \geq (1 - \alpha) \implies \exists \mathbf{q} \in \mathbf{Q}, \mathbf{q} \vdash P)$$

The previous definition is a bit strong, we can reduce it by considering only the majority of knowledge in $\mathbf{q}$

2 Block Equivalence

2.1 Equivalent Body of knowledge

Definition 10. *Given a validation rule V, a genesis list G of V and an equivalence relationship $\equiv$ over $\mathbf{B}$. We say that $K_1 \in \mathrm{BK}(G, V)$ and $K_2 \in \mathrm{BK}(G, V)$ are $\equiv$ equivalent if and only if:*

$$\forall L_1 \in K_1, \exists L_2 \in K_2, \forall i \in [\![1, |L_1|]\!], L_1[i] \equiv L_2[i]$$
$$\forall L_2 \in K_2, \exists L_1 \in K, \forall i \in [\![1, |L_2|]\!], L_1[i] \equiv L_2[i]$$

By extension we denote $K_1 \equiv K_2$ resp. $L_1 \equiv L_2$ when two body knowledge resp. list are $\equiv$ equivalent.

We denote $\mathrm{BK}^{\equiv}(G, V)$ the set of equivalence classes of $\mathrm{BK}(G, V)$

Definition 11. *Given a validation rule V, a genesis list G of V and $\{\preceq_i\}_{i \in \mathbb{N}}$ a blockchain protocol over (G, V) we say that an equivalence relationship $\equiv$ over $\mathbf{B}$ is $\{\preceq_i\}_{i \in \mathbb{N}}$ compatible if and only if:*

$$\forall K_1, K_2 \in \mathrm{BK}(G, V)$$

$$K_1 \equiv K_2 \implies \forall i \in \mathbb{N}, \forall L_1 \in \{L | L \in K_1, \forall L' \in K_1, L' \preceq_i L\}, \exists L_2 \in \{L | L \in K_2, \forall L' \in K_2, L' \preceq_i L\}, L_1 \equiv L_2$$

2.2 Game with equivalence

For now on we consider a game $\Gamma = (\mathcal{P}, \mathcal{A}, \mathcal{V}, \mathcal{R}, \mathbf{Pr})$ associated to a validation rule V a genesis list G and a blockchain protocol $\{\preceq_i\}_{i \in \mathbb{N}}$.

We say that two view $\mathbf{v}_1, \mathbf{v}_2 \in \mathcal{V}$ are equivalent regarding $\equiv$ a equivalent relationship over $\mathbf{B}$ noted $\mathbf{v}_1 \equiv \mathbf{v}_2$ if

$$\forall p \in \mathcal{P}, v_{1p} \equiv v_{2p}$$

We denote $\mathcal{V}^{\equiv}$ the set of equivalence classes of $\mathcal{V}$

Definition 12. *Let $\equiv$ a equivalence relationship over $\mathbf{B}$ we say that $\equiv$ is $\mathcal{A}$ compatible if $\forall p \in \mathcal{P}$ and $\forall a \in \mathcal{A}_p$ we have*

$$\forall \mathbf{v}_1, \mathbf{v}_2 \in \mathcal{V}, \mathbf{v}_1 \equiv \mathbf{v}_2 \implies a(\mathbf{v}_1) \equiv a(\mathbf{v}_2)$$

Definition 13. *Let $p \in \mathcal{P}$,considering $a_1, a_2 \in \mathcal{A}_p$ and $\equiv$ a equivalence relationship $\mathcal{A}$ compatible we say that a_1 and a_2 are equivalent noted $a_1 \equiv a_2$ if and if:*

$$\forall \mathbf{v}_1, \mathbf{v}_2 \in \mathcal{V}, \mathbf{v}_1 \equiv \mathbf{v}_2 \implies a_1(\mathbf{v}_1) \equiv a_2(\mathbf{v}_2)$$

We denote $\mathcal{A}_p^{\equiv}$ the set of equivalence classes of $\mathcal{A}$ then a element of $\mathcal{A}_p^{\equiv}$ is a function

$$a^{\equiv} : \mathcal{V}^{\equiv} \to \mathcal{V}^{\equiv}$$

.

Property 2. *Let $p \in \mathcal{P}$, $\equiv$ a equivalence relationship $\mathcal{A}$ compatible and $a^{\equiv} \in \mathcal{A}_p^{\equiv}$ then*

- *for every $\mathbf{v}^{\equiv} \in \mathcal{V}^{\equiv}$ and $q \in \mathcal{P}$, if $\mathbf{v}^{\equiv} = (v_1^{\equiv}, \dots, v_n^{\equiv})$ and $a^{\equiv}(\mathbf{v}^{\equiv}) = (w_1^{\equiv}, \dots, w_n^{\equiv})$, then it holds that:*

$$\forall v_q \in v_q^{\equiv}, \forall w_q \in w_q^{\equiv}, \forall w_p \in w_p^{\equiv}, v_q \subseteq w_q \subseteq v_q \cup w_p.$$

Proof. to do. □

Property 3. *Let $p \in \mathcal{P}$, and $\equiv$ a $\mathcal{A}$ compatible equivalence relationship over* **B** *then the function $\mathbf{Pr}^{\equiv} : \mathcal{V}^{\equiv} \times \mathcal{A}^{\equiv} \times \mathcal{V}^{\equiv} \to [0,1]$ such that:*

$$\mathbf{Pr}^{\equiv}(\mathbf{v}^{\equiv}, \mathbf{a}^{\equiv}, \mathbf{w}) = \sum_{\mathbf{a} \in \mathbf{a}^{\equiv}} \mathbf{Pr}(\mathbf{v}, \mathbf{a}, \mathbf{w}) \textit{ where : } \mathbf{v} \in \mathbf{v}^{\equiv} \textit{ and } \mathbf{w} \in \mathbf{w}^{\equiv}$$

is well defined and

$$\forall \mathbf{v}^{\equiv} \in \mathcal{V}^{\equiv}, \forall \mathbf{a} \in \mathcal{A}^{\equiv}, \sum_{\mathbf{w}^{\equiv} \in \mathcal{V}^{\equiv}} \mathbf{Pr}^{\equiv}(\mathbf{v}^{\equiv}, \mathbf{a}^{\equiv}, \mathbf{w}^{\equiv}) = 1$$

Proof. to do. □

Definition 14. *Let $\equiv$ a equivalence relationship over* **B** *we say that $\equiv$ is $\mathcal{R}$ compatible if its $\mathcal{A}$ compatible and $\forall p \in \mathcal{P}$ and $\forall \mathbf{v} \in \mathcal{V}$ we have*

$$\forall \mathbf{a}_1, \mathbf{a}_2 \in \mathcal{A}, \mathbf{a}_1 \equiv \mathbf{a}_2 \implies r_p(\mathbf{v}, \mathbf{a}_1) = r_p(\mathbf{v}, \mathbf{a}_1)$$

Property 4. *Let $p \in \mathcal{P}$, and $\equiv$ a $\mathcal{R}$ compatible equivalence relationship over* **B** *then the function $r_p^{\equiv} : \mathcal{V}^{\equiv} \times \mathcal{A}^{\equiv} \to \mathbb{R}$ such that :*

$$r_p^{\equiv}(\mathbf{v}^{\equiv}, \mathbf{a}^{\equiv},) = r_p(\mathbf{v}, \mathbf{a}) \textit{ where : } \mathbf{v} \in \mathbf{v}^{\equiv} \textit{ and } \mathbf{a} \in \mathbf{a}^{\equiv}$$

is well defined.

Proof. to do. □

Property 5. *Let $\equiv$ a $\mathcal{R}$ compatible equivalence relationship over* **B** *then $\Gamma^{\equiv} = (\mathcal{P}, \mathcal{A}^{\equiv}, \mathcal{V}^{\equiv}, \mathcal{R}^{\equiv}, \mathbf{Pr}^{\equiv})$ is a well defined infinite stochastic game.*

Proof. immediate. □

3 Etienne ′s modification space