Review of Economic Design (2022) 26:605–633
https://doi.org/10.1007/s10058-021-00282-3

ORIGINAL PAPER

Computational implementation

Mehmet Barlo[1] · Nuh Aygün Dalkıran[2]

Received: 31 March 2021 / Accepted: 3 December 2021 / Published online: 8 January 2022

Abstract
Following a theoretical analysis of the scope of Nash implementation for a given mechanism, we study the formal framework for computational identification of Nash implementability. We provide computational tools for Nash implementation in finite environments. In particular, we supply Python codes that identify (i) the domain of preferences that allows Nash implementation by a given mechanism, (ii) the maximal domain of preferences that a given mechanism Nash implements Pareto efficiency, (iii) all consistent collections of sets of a given social choice correspondence (SCC), the existence of which is a necessary condition for Nash implementation of this SCC, and (iv) check whether some of the well-known sufficient conditions for Nash implementation hold for a given SCC. Our results exhibit that the computational identification of all collections consistent with an SCC enables the planner to design appealing mechanisms.

Keywords Nash implementation · Computation · Maskin monotonicity · Consistent collections · Maximal domain · Behavioral implementation

JEL Classification C72 · D71 · D78 · D82 · D90

1 Introduction

Implementation theory deals with the problem of designing a mechanism such that the optimal alternatives prescribed by the designer coincide with the equilibrium outcomes

We would like to thank Muhammed Dönmez for the excellent research assistance. We gratefully acknowledge the financial support through TUBITAK Grant No. 119K957. Any remaining errors are ours.

✉ Nuh Aygün Dalkıran
dalkiran@bilkent.edu.tr

Mehmet Barlo
barlo@sabanciuniv.edu

1 Faculty of Arts and Social Sciences, Sabancı University, Istanbul, Turkey

2 Department of Economics, Bilkent University, Ankara, Turkey

of this mechanism. There has been a vast literature on implementation following (Maskin 1999, circulated since 1977), which is the first paper that identifies necessary as well as sufficient conditions for implementation when the equilibrium notion under consideration is Nash Equilibrium. Despite the vast literature on implementation and advances in computational tools, to the best of our knowledge, computational tools have not been employed in the implementation literature.[1] In this paper, we aim to fill the gap by providing computational tools for Nash implementation.

The classical approach in Nash implementation (based on the seminal works Maskin 1999; Moore and Repullo 1990, and Dutta and Sen 1991) seeks to identify social choice correspondences (SCCs) defined on unrestricted domains of preferences that are attainable as Nash equilibrium outcomes of mechanisms.[2] To implement such an SCC, the planner no longer needs to acquire information about the true preference profile of the society as the mechanism the planner employs indirectly provides her with the relevant information: The SCC coincides with the set of Nash equilibrium outcomes of the mechanism at every realized state of the world. If Nash implementation of an SCC is not achievable on unrestricted domains of preferences, it might still be possible to Nash implement this SCC on a restricted domain of preferences.

In a nutshell, there are three essential components of Nash implementation: (i) a domain of feasible preferences; (ii) the optimal outcomes described by an SCC, i.e., the desired goal; (iii) a mechanism (game form) the Nash equilibria of which equal the optimal outcomes at every preference profile in the domain of feasible preferences. The main goal of the mechanism designer then can be thought of as identifying (iii) given (i) and (ii), i.e., identifying a mechanism that Nash implements a given SCC on the feasible domain of preferences.

Our results in this paper are divided into two parts: In the first part, we ask what can be achieved in terms of Nash implementation by a given mechanism. As opposed to the standard approach, we identify (i) and (ii) given (iii). That is, we characterize the SCCs along with the domain of preferences they are defined on that are implementable in Nash equilibrium via a given mechanism. Our results, therefore, describe the scope of Nash implementation by a given mechanism in detail. In the second part of the paper, we turn back to the standard approach: By revisiting the standard necessity and sufficiency results, we provide computational tools that describe the scope of Nash implementation for a given SCC on a given domain of feasible preferences.

In the first part of the paper, for a given mechanism, we establish that the set of attainable Nash equilibrium outcomes of this mechanism partitions the domain of preferences under which there is a Nash equilibrium of this mechanism (Theorem 1). This partition identifies the boundaries of Nash implementation under the mechanism at hand: Given a mechanism, the set of SCCs and the corresponding domain of preferences under which Nash implementation is viable are precisely those that *respect* the intrinsic relation described by the partition structure induced by this mechanism (Theorem 2).

[1] There is a computational mechanism design literature that focuses on Myersonian mechanism design (partial implementation) by restricting attention to direct mechanisms thanks to the revelation principle (Dash et al. 2003). Yet, this literature neglects implementation theory.

[2] For more on the standard approach to Nash implementation, please see Maskin and Sjöström (2002), Palfrey (2002), and Serrano (2004).

Our findings unfold the precise incompatibilities between the desired set of alternatives and the domain of preferences for any SCC that fails to be Nash implementable by a given mechanism. Identifying these incompatibilities empowers us to characterize the maximal domain of preferences where Nash implementation of a given SCC is attainable by the given mechanism. We demonstrate this using the *Pareto efficient SCC*: Given any mechanism, we delineate the maximal domain of preferences under which efficiency restricted to this domain is Nash implementable by this mechanism.

In the second part of the paper, we first focus on the prominent Maskin-monotonicity condition as a necessary Nash implementation condition. Generalizing Maskin (1999)'s results on Nash implementation to behavioral domains, de Clippel (2014) defines the concept of a consistent collection of sets for a given SCC, the existence of which is equivalent to Maskin-monotonicity under rationality. The set of all consistent collections of sets of a given SCC on a given domain of preferences defines the boundaries of Nash implementation of this SCC. We exemplify that by identifying the set of all consistent collections of sets for an SCC, the designer might construct eligible mechanisms to implement the given SCC. Then, we turn to the well-known sufficiency conditions for Nash implementation and provide codes that check whether a given SCC satisfies the sufficiency conditions.

In particular, when there are at least three individuals in the society, the existence of a consistent collection of sets is both necessary and sufficient for Nash implementation in an economic environment. We provide codes that check whether the domain of preferences of an SCC satisfies the economic environment assumption. Furthermore, when there are at least three individuals in the society, no-veto-power (NVP) property is sufficient for Nash implementation when the SCC under consideration has a consistent collection. We also provide codes that check for the NVP property of an SCC.

The results in the second part of our paper can be generalized to behavioral domains as in de Clippel (2014). Instead of the domain of rational preferences, our results also accommodate the domain of individual choices that do not necessarily satisfy the weak axiom of revealed preferences (WARP).

The organization of the rest of the paper is as follows. We present the preliminaries in Sect. 2. The first part of our results where we analyze the scope of Nash implementation by a given mechanism is in Sect. 3. The second part of our results where we provide computational tools for Nash implementation of a given SCC is in Sect. 4. Section 5 provides a brief literature review. Meanwhile, Sect. 6 concludes. Unless stated otherwise, the proofs are presented in the "Appendix". Our Python codes are available online at http://dalkiran.bilkent.edu.tr/Python_Computational_Implementation.zip

2 Preliminaries

Let $N = \{1, \ldots, n\}$ denote a *society* with at least two individuals, X a set of *alternatives*, 2^X the set of all subsets of X, and $\mathcal{X}$ the set of all non-empty subsets of X.

We denote by Ω the set of all *possible states* of the world capturing all the payoff-relevant characteristics of the environment. The *preferences* of individual $i \in N$ at state $\omega \in \Omega$ is captured by a complete and transitive binary relation, a ranking,

$R_i^\omega \subseteq X \times X$.[3] The *ranking profile of the society*, $\mathbf{R} = (R_i^\omega)_{i\in N,\ \omega\in\Omega}$, is in one-to-one correspondence with Ω. Given $i \in N$, $\omega \in \Omega$, and $x \in X$, $L_i^\omega(x) := \{y \in X \mid x R_i^\omega y\}$ denotes the *lower contour set of individual* i *at state* ω *of alternative* x. For all $i \in N$, all $\omega \in \Omega$, and all $S \in \mathcal{X}$, define $C_i^\omega(S) := \{x \in S \mid x R_i^\omega y, \forall y \in S\}$.

We refer to $\Theta \subset \Omega$ as a *domain*. A *social choice correspondence* (SCC) defined on a domain Θ is $f : \Theta \to \mathcal{X}$, a non-empty valued correspondence mapping Θ into X. Given $\theta \in \Theta$, $f(\theta)$, the set of f-*optimal* alternatives at θ, consists of alternatives that the planner desires to sustain at θ.

A mechanism $\mu = (M, g)$ assigns each individual $i \in N$ a non-empty *message space* M_i and specifies an *outcome function* $g : M \to X$ where $M = \times_{j\in N} M_j$. $\mathcal{M}$ denotes the set of all mechanisms. Given $\mu \in \mathcal{M}$ and $m_{-i} \in M_{-i} := \times_{j\neq i} M_j$, the *opportunity set* of individual i pertaining to others' message profile m_{-i} in mechanism μ is $O_i^\mu(m_{-i}) := g(M_i, m_{-i}) = \{g(m_i, m_{-i}) \mid m_i \in M_i\}$. Consequently, a message profile $m^* \in M$ is a ***Nash equilibrium of mechanism*** μ ***at state*** $\omega \in \Omega$ if $g(m^*) \in \cap_{i\in N} C_i^\omega(O_i^\mu(m_{-i}^*))$. Given $\mu \in \mathcal{M}$, the correspondence $NE^\mu : \Omega \twoheadrightarrow 2^X$ identifies ***Nash equilibrium outcomes of mechanism*** μ ***at state*** $\omega \in \Omega$ and is defined by $NE^\mu(\omega) := \{x \in X \mid \exists m^* \in M \text{ s.t. } g(m^*) \in \cap_{i\in N} C_i^\omega(O_i^\mu(m_{-i}^*)) \text{ and } g(m^*) = x\}$.

3 Implementation with a given mechanism

In what follows, we first show that the set of attainable Nash equilibrium outcomes of a given mechanism *partitions* the domain of preferences under which there is a Nash equilibrium of this mechanism. Using this, we establish that the domain of preferences under which Nash implementation is viable via the given mechanism are precisely those that *respect* the intrinsic partition structure of this mechanism.[4]

Given $\mu \in \mathcal{M}$ and $x \in X$, the set of ***states that sustain*** x ***as a Nash equilibrium outcome in mechanism*** μ is given by $\Pi^\mu(\{x\}) := \{\omega \in \Omega \mid x \in NE^\mu(\omega)\}$. Clearly, for all $x \notin g(M)$, $\Pi^\mu(\{x\}) = \emptyset$. Consequently, for any $S \in \mathcal{X}$, define $\Pi^\mu(S) := \cap_{x\in S} \Pi^\mu(\{x\})$. Indeed, for any $\omega \in \Pi^\mu(S)$, $S \subset NE^\mu(\omega)$. Finally, given $\mu \in \mathcal{M}$ and $S \in \mathcal{X}$, the set of ***states at which Nash equilibrium outcomes of mechanism*** μ ***equal*** S is $\Phi^\mu(S) := \Pi^\mu(S) \setminus \Pi^\mu(X \setminus S)$; alternatively, $\Phi^\mu(S) := \cap_{x\in S} \Pi^\mu(\{x\}) \setminus \cup_{y\notin S} \Pi^\mu(\{y\})$. We define the set of ***states at which the set of Nash equilibrium outcomes of mechanism*** μ ***are non-empty*** by $\Phi^\mu := \{\omega \in \Omega \mid NE^\mu(\omega) \neq \emptyset\}$. It is easy to see that $\Phi^\mu = \cup_{S\in\mathcal{X}} \Phi^\mu(S)$. Moreover, the following lemma offers an equivalent definition of Φ^μ.

Lemma 1 *Given mechanism* $\mu = (M, g)$, $\Phi^\mu = \cup_{x\in X} \Pi^\mu(\{x\})$.

The family of sets of alternatives sustained as a Nash equilibrium outcome of mechanism μ at some state $\omega \in \Omega$ is given by $\mathcal{S}^\mu := \{S \in \mathcal{X} \mid \exists \omega \in \Omega \text{ s.t. } NE^\mu(\omega) = S\}$. It is useful to note that $\mathcal{S}^\mu := \{S \in \mathcal{X} \mid \Phi^\mu(S) \neq \emptyset\}$, and $\Phi^\mu = \cup_{\tilde{S}\in\mathcal{S}^\mu} \Phi^\mu(\tilde{S})$.

[3] A binary relation $R \subseteq X \times X$ is *complete* if for all $x, y \in X$ either xRy or yRx or both; *transitive* if for all $x, y, z \in X$ with xRy and yRz implies xRz.

[4] Our results are related to the literature on realizations, message processes, and communication protocols with verification properties. We refer the interested reader to Hurwicz and Reiter (2006) for more on this subject. Section 5 provides further discussions of the related literature.

Table 1 Outcome function of mechanism μ

		Bob		
		L	M	R
Ann	U	b	a	c
	C	a	b	b
	D	c	b	c

The following result establishes a useful partition property:

Theorem 1 *Given mechanism* $\mu = (M, g)$, $\{\Phi^{\mu}(S) \mid S \in \mathcal{S}^{\mu}\}$ *is a partition of* Φ^{μ}.

Example 1.[5] The following helps exemplify our construction: Consider a situation with two individuals, Ann and Bob, and the set of alternatives $X = \{a, b, c\}$. We restrict attention to all strict ranking profiles of Ann and Bob, and hence, Ω corresponds to all strict ranking profiles of a, b, c. There are 6×6 strict ranking profiles and hence possible states of the world. We adopt the notation where xyz denotes the strict preference order with x is strictly preferred to y, y to z, and x, y, z are distinct elements in $\{a, b, c\}$.

We analyze mechanism $\mu = (M, g)$ with $M_A = \{U, C, D\}$ and $M_B = \{L, M, R\}$ where the outcome function $g : M \to X$ is as given in Table 1.

Now, we determine the set of states at which alternative a is among the Nash equilibrium outcomes, i.e., $\Pi^{\mu}(\{a\}) \subset \Omega$. As there are two message profiles, (C, L) and (U, M), that deliver a as an outcome, we identify the strict ranking profiles at which at least one of these message profiles is a Nash equilibrium of μ.

The set of strict ranking profiles at which (C, L) is among the Nash equilibrium outcomes of μ is such that Ann ranks a as the first alternative while Bob strictly prefers a to b. Thus, Ann's possible strict rankings are given by abc, acb and Bob's by abc, acb, cab. Therefore, (C, L) is a Nash equilibrium of μ at any state corresponding to $\{\{abc, acb\} \times \{abc, acb, cab\}\}$. Similarly, the strict ranking profiles at which (U, M) is among the Nash equilibrium outcomes of μ are $\{\{abc, acb, cab\} \times \{abc, acb\}\}$. So, $\Pi^{\mu}(\{a\}) = \{\{abc, acb\} \times \{abc, acb, cab\}\} \cup \{\{abc, acb, cab\} \times \{abc, acb\}\}$. Thus,

$$\Pi^{\mu}(\{a\}) = \left\{ \begin{pmatrix} a & a \\ b & b \\ c & c \end{pmatrix}, \begin{pmatrix} a & a \\ b & c \\ c & b \end{pmatrix}, \begin{pmatrix} a & c \\ b & a \\ c & b \end{pmatrix}, \begin{pmatrix} a & a \\ c & b \\ b & c \end{pmatrix}, \begin{pmatrix} a & a \\ c & c \\ b & b \end{pmatrix}, \begin{pmatrix} a & c \\ c & a \\ b & b \end{pmatrix}, \begin{pmatrix} c & a \\ a & b \\ b & c \end{pmatrix}, \begin{pmatrix} c & a \\ a & c \\ b & b \end{pmatrix} \right\}.$$

[5] We refer the interested reader to the "Examples_in_the_Manuscript\Example_1_in_the_Manuscript" in "Python_Codes_Computational_Implementation_Barlo_Dalkiran.zip" for the computational codes/outputs.

By repeating the same arguments, we obtain

$$\begin{aligned}
\Pi^{\mu}(\{b\}) = \Bigg\{ & \begin{pmatrix} b & b \\ a & a \\ c & c \end{pmatrix}, \begin{pmatrix} b & b \\ a & c \\ c & a \end{pmatrix}, \begin{pmatrix} b & b \\ c & a \\ a & c \end{pmatrix}, \begin{pmatrix} b & b \\ c & c \\ a & a \end{pmatrix}, \begin{pmatrix} b & c \\ a & b \\ c & a \end{pmatrix}, \begin{pmatrix} b & c \\ c & b \\ a & a \end{pmatrix}, \begin{pmatrix} c & b \\ b & a \\ a & c \end{pmatrix}, \\
& \begin{pmatrix} c & b \\ b & c \\ a & a \end{pmatrix}, \begin{pmatrix} c & c \\ b & b \\ a & a \end{pmatrix}, \begin{pmatrix} b & a \\ a & b \\ c & c \end{pmatrix}, \begin{pmatrix} b & a \\ c & b \\ a & c \end{pmatrix}, \begin{pmatrix} c & a \\ b & b \\ a & c \end{pmatrix}, \begin{pmatrix} a & b \\ b & a \\ c & c \end{pmatrix}, \begin{pmatrix} a & b \\ b & c \\ c & a \end{pmatrix}, \begin{pmatrix} a & c \\ b & b \\ c & a \end{pmatrix} \Bigg\} \\
\Pi^{\mu}(\{c\}) = \Bigg\{ & \begin{pmatrix} c & a \\ a & c \\ b & b \end{pmatrix}, \begin{pmatrix} c & c \\ a & b \\ b & a \end{pmatrix}, \begin{pmatrix} c & c \\ a & a \\ b & b \end{pmatrix}, \begin{pmatrix} c & a \\ b & c \\ a & b \end{pmatrix}, \begin{pmatrix} c & c \\ b & b \\ a & a \end{pmatrix}, \begin{pmatrix} c & c \\ b & a \\ a & b \end{pmatrix}, \begin{pmatrix} a & c \\ c & a \\ b & b \end{pmatrix}, \begin{pmatrix} a & c \\ c & b \\ b & a \end{pmatrix}, \begin{pmatrix} a & a \\ c & c \\ b & b \end{pmatrix} \Bigg\}
\end{aligned}$$

As for any $S \in \mathcal{X}$, $\Pi^{\mu}(S) := \cap_{x \in S} \Pi^{\mu}(\{x\})$, we see that

$$\Pi^{\mu}(\{a, b\}) = \emptyset, \Pi^{\mu}(\{a, c\}) = \left\{ \begin{pmatrix} a & a \\ c & c \\ b & b \end{pmatrix}, \begin{pmatrix} a & c \\ c & a \\ b & b \end{pmatrix}, \begin{pmatrix} c & a \\ a & c \\ b & b \end{pmatrix} \right\},$$

$$\Pi^{\mu}(\{b, c\}) = \left\{ \begin{pmatrix} c & c \\ b & b \\ a & a \end{pmatrix} \right\}, \Pi^{\mu}(\{a, b, c\}) = \emptyset.$$

We observe that there is no ranking profile at which both a and b appear among the Nash equilibrium outcomes of μ.

Next, for any non-empty subset S of $\{a, b, c\}$, we construct the set of states at which Nash equilibrium outcomes of mechanism μ equals S, namely, $\Phi^{\mu}(S)$. As $\Phi^{\mu}(S) := \cap_{x \in S} \Pi^{\mu}(\{x\}) \setminus \cup_{y \notin S} \Pi^{\mu}(\{y\})$ for any non-empty S that is a subset of $\{a, b, c\}$, we obtain these sets by using $\Pi^{\mu}(\{x\})$ with $x \in \{a, b, c\}$. Noting $\Phi^{\mu}(\{a\}) = \Pi^{\mu}(\{a\}) \setminus (\Pi^{\mu}(\{b\}) \cup \Pi^{\mu}(\{c\}))$, we observe that

$$\Phi^{\mu}(\{a\}) = \left\{ \begin{pmatrix} a & a \\ b & b \\ c & c \end{pmatrix}, \begin{pmatrix} a & a \\ b & c \\ c & b \end{pmatrix}, \begin{pmatrix} a & c \\ b & a \\ c & b \end{pmatrix}, \begin{pmatrix} a & a \\ c & b \\ b & c \end{pmatrix}, \begin{pmatrix} c & a \\ a & b \\ b & c \end{pmatrix} \right\}.$$

Equivalently, as $\Pi^{\mu}(\{a, b\}) = \Pi^{\mu}(\{a, b, c\}) = \emptyset$, we have $\Phi^{\mu}(\{a\}) = \Pi^{\mu}(\{a\}) \setminus \Pi^{\mu}(\{a, c\})$. That is, $\Phi^{\mu}(\{a\})$ consists of all the set of states at which only a (and neither b nor c) is among the Nash equilibrium outcomes of μ. Similar arguments establish the following:

$$\Phi^\mu(\{b\}) = \Pi^\mu(\{b\}) \setminus \Pi^\mu(\{b,c\})$$
$$= \left\{ \begin{pmatrix} b & b \\ a & a \\ c & c \end{pmatrix}, \begin{pmatrix} b & b \\ a & c \\ c & a \end{pmatrix}, \begin{pmatrix} b & b \\ c & a \\ a & c \end{pmatrix}, \begin{pmatrix} b & b \\ c & c \\ a & a \end{pmatrix}, \begin{pmatrix} b & c \\ a & b \\ c & a \end{pmatrix}, \begin{pmatrix} b & c \\ c & b \\ a & a \end{pmatrix}, \begin{pmatrix} c & b \\ b & a \\ a & c \end{pmatrix}, \begin{pmatrix} c & b \\ b & c \\ a & a \end{pmatrix}, \begin{pmatrix} b & a \\ a & b \\ c & c \end{pmatrix}, \begin{pmatrix} b & a \\ c & b \\ a & c \end{pmatrix}, \begin{pmatrix} c & a \\ b & b \\ a & c \end{pmatrix}, \begin{pmatrix} a & b \\ b & a \\ c & c \end{pmatrix}, \begin{pmatrix} a & b \\ b & c \\ c & a \end{pmatrix}, \begin{pmatrix} a & c \\ b & b \\ c & a \end{pmatrix} \right\},$$

$$\Phi^\mu(\{c\}) = \Pi^\mu(\{b\}) \setminus (\Pi^\mu(\{a,c\}) \cup \Pi^\mu(\{b,c\}))$$
$$= \left\{ \begin{pmatrix} c & c \\ a & b \\ b & a \end{pmatrix}, \begin{pmatrix} c & c \\ a & a \\ b & b \end{pmatrix}, \begin{pmatrix} c & a \\ b & c \\ a & b \end{pmatrix}, \begin{pmatrix} c & c \\ b & a \\ a & b \end{pmatrix}, \begin{pmatrix} a & c \\ c & b \\ b & a \end{pmatrix} \right\},$$

$$\Phi^\mu(\{a,b\}) = \emptyset,$$

$$\Phi^\mu(\{a,c\}) = \Pi^\mu(\{a\}) \cap \Pi^\mu(\{c\}) \setminus \Pi^\mu(\{b\}) = \left\{ \begin{pmatrix} a & a \\ c & c \\ b & b \end{pmatrix}, \begin{pmatrix} a & c \\ c & a \\ b & b \end{pmatrix}, \begin{pmatrix} c & a \\ a & c \\ b & b \end{pmatrix} \right\},$$

$$\Phi^\mu(\{b,c\}) = \Pi^\mu(\{b\}) \cap \Pi^\mu(\{c\}) \setminus \Pi^\mu(\{a\}) = \left\{ \begin{pmatrix} c & c \\ b & b \\ a & a \end{pmatrix} \right\},$$

$$\Phi^\mu(\{a,b,c\}) = \emptyset.$$

Therefore, we conclude that $\mathcal{S}^\mu = \{\{a\}, \{b\}, \{c\}, \{a,c\}, \{b,c\}\}$.

Table 2 presents an application of Theorem 1 by demonstrating the partition $\{\Phi^\mu(S) \mid S \in \mathcal{S}^\mu\}$ of Φ^μ in our example.

3.1 Nash implementable SCCs by a given mechanism $\mu = (M, g)$

The definition of Nash implementability of an SCC $f : \Theta \to \mathcal{X}$ by a given mechanism μ can be stated as follows:

Definition 1 Given a mechanism $\mu = (M, g)$, an SCC f defined on a domain $\Theta \subset \Omega$, $f : \Theta \to \mathcal{X}$, is ***Nash implementable*** by μ if for all $\theta \in \Theta$,

$$f(\theta) = S \text{ if and only if } \theta \in \Phi^\mu(S). \tag{1}$$

$\mathcal{F}^\mu$ denotes the Nash implementable SCCs for a given mechanism μ.

This definition signifies that the association proposed by an SCC $f : \Theta \to \mathcal{X}$ between states in its domain Θ and non-empty subsets of alternatives must respect mechanism μ's *inherent* association between the states in Φ^μ and subsets of alternatives in $\mathcal{S}^\mu$ whenever this SCC is Nash implementable by μ. Indeed, from (1), we see that if there is $\theta \in \Theta$ such that $\theta \notin \Phi^\mu(f(\theta))$, then f is not Nash implementable. The existence of $\theta \in \Theta$ such that $\theta \notin \Phi^\mu(f(\theta))$ subsumes the situation when either

Table 2 The partition $\{\Phi^{\mu}(S) \mid S \in \mathcal{S}^{\mu}\}$ of Φ^{μ} in our example

$$\overbrace{\begin{pmatrix} a & a \\ b & b \\ c & c \end{pmatrix}, \begin{pmatrix} a & a \\ b & c \\ c & b \end{pmatrix}, \begin{pmatrix} a & c \\ b & a \\ c & b \end{pmatrix}, \begin{pmatrix} a & a \\ c & b \\ b & c \end{pmatrix}, \begin{pmatrix} c & a \\ a & b \\ b & c \end{pmatrix}}^{\Phi^{\mu}(\{a\})}, \overbrace{\begin{pmatrix} c & c \\ a & b \\ b & a \end{pmatrix}, \begin{pmatrix} c & c \\ a & a \\ b & b \end{pmatrix}, \begin{pmatrix} c & a \\ b & c \\ a & b \end{pmatrix}, \begin{pmatrix} c & c \\ b & a \\ a & b \end{pmatrix}, \begin{pmatrix} a & c \\ c & b \\ b & a \end{pmatrix}}^{\Phi^{\mu}(\{c\})}, \overbrace{\begin{pmatrix} a & a \\ c & c \\ b & b \end{pmatrix}, \begin{pmatrix} a & c \\ c & a \\ b & b \end{pmatrix}, \begin{pmatrix} c & a \\ a & c \\ b & b \end{pmatrix}}^{\Phi^{\mu}(\{a,c\})}, \overbrace{\begin{pmatrix} c & c \\ b & b \\ a & a \end{pmatrix}}^{\Phi^{\mu}(\{b,c\})}$$

$$\underbrace{\begin{pmatrix} b & b \\ a & a \\ c & c \end{pmatrix}, \begin{pmatrix} b & b \\ a & c \\ c & a \end{pmatrix}, \begin{pmatrix} b & b \\ c & a \\ a & c \end{pmatrix}, \begin{pmatrix} b & b \\ c & c \\ a & a \end{pmatrix}, \begin{pmatrix} b & c \\ a & b \\ c & a \end{pmatrix}, \begin{pmatrix} b & c \\ c & b \\ a & a \end{pmatrix}, \begin{pmatrix} c & b \\ b & a \\ a & c \end{pmatrix}, \begin{pmatrix} c & b \\ b & c \\ a & a \end{pmatrix}, \begin{pmatrix} b & a \\ a & b \\ c & c \end{pmatrix}, \begin{pmatrix} b & a \\ c & b \\ a & c \end{pmatrix}, \begin{pmatrix} c & a \\ b & b \\ a & c \end{pmatrix}, \begin{pmatrix} a & b \\ b & a \\ c & c \end{pmatrix}, \begin{pmatrix} a & b \\ b & c \\ c & a \end{pmatrix}, \begin{pmatrix} a & c \\ b & b \\ c & a \end{pmatrix}}_{\Phi^{\mu}(\{b\})}$$

$\theta \notin \Phi^\mu$ or $\theta \in \Phi^\mu$ but $\theta \notin \Phi^\mu(f(\theta))$. In words, f is not Nash implementable by μ whenever either there is a state $\theta \in \Theta$ at which there is no Nash equilibrium of μ or there is no state $\theta \in \Theta$ at which the Nash equilibrium outcomes of μ at θ equals $f(\theta)$.

The following corollary to Theorem 1 summarizes these findings that are useful when constructing SCCs that are Nash implementable by mechanism μ:

Corollary 1 *An SCC $f : \Theta \to \mathcal{X}$ with domain $\Theta \subset \Omega$ is not in $\mathcal{F}^\mu$, whenever*

(i) either $\theta \notin \Phi^\mu$,
(ii) or $\theta \in \Phi^\mu$ and $\theta \notin \Phi^\mu(f(\theta))$.

In Example 1, $\Phi^\mu \subset \Omega$ does not contain any ω such that $NE^\mu(\omega) = \{a, b\}$ (i.e., $\{a, b\} \notin \mathcal{S}^\mu$). Thus, any SCC $f : \Theta \to \mathcal{X}$ with $f(\theta) = \{a, b\}$ for some $\theta \in \Theta$ cannot be Nash implementable (Corollary 1-(i)). Moroever, from Table 2 we know that at $\theta = (abc, cab)$, $NE^\mu(\theta) = \{a\}$; so, if $f(\theta) = \{a, c\}$ (the efficient alternatives at θ), then f would not be implementable since $f(\theta) = \{a, c\}$ and $\theta \notin \Phi^\mu(\{a, c\})$ but $\theta \in \Phi^\mu(\{a\})$ (Corollary 1-(ii)).

Using the partition structure formalized in Theorem 1, we obtain the following consequence of Nash implementability that also signifies a *robustness* argument:

Corollary 2 *Suppose a pair of SCCs, $f : \Theta \to \mathcal{X}$ and $\tilde{f} : \tilde{\Theta} \to \mathcal{X}$, with domains $\Theta, \tilde{\Theta} \subset \Omega$, are in $\mathcal{F}^\mu$. Then, for all $\theta \in \Theta$ and $\tilde{\theta} \in \tilde{\Theta}$*

$$f(\theta) = \tilde{f}(\tilde{\theta}) \text{ if and only if } \theta \in \Phi^\mu(\tilde{f}(\tilde{\theta})).$$

In words, a pair of SCCs, f and $\tilde{f}$, being Nash implementable by mechanism μ means that the desired alternatives under f at θ coinciding with those under $\tilde{f}$ at $\tilde{\theta}$ is equivalent to θ and $\tilde{\theta}$ being in the same partition associated with μ.

In Example 1, we see that Corollary 2 implies the following: From Table 2, we observe that $\theta = (abc, cab)$ and $\tilde{\theta} = (acb, abc)$ are in $\Phi^\mu(\{a\})$; thus, any pair of Nash implementable SCCs $f, \tilde{f}$ where θ ($\tilde{\theta}$) is in the domain of f ($\tilde{f}$, resp.) must be such that $f(\theta) = \{a\} = \tilde{f}(\tilde{\theta})$.

An equivalent and useful way to define Nash implementability is as follows:

Lemma 2 *Given a mechanism $\mu = (M, g)$, an SCC f defined on a domain $\Theta \subset \Omega$, $f : \Theta \to \mathcal{X}$, is* ***Nash implementable by mechanism*** *μ if and only if*

$$\text{for all } \theta \in \Theta, x \in f(\theta) \text{ if and only if } \theta \in \Pi^\mu(\{x\}). \tag{2}$$

The construction of Nash implementable SCCs and corresponding domains can now be described using these observations. Given mechanism μ, we *first* identify $\mathfrak{P}^\mu := \{\Pi^\mu(\{x\}) \subset \Omega \mid x \in X\}$. Using $\mathfrak{P}^\mu$, we obtain $\{\Phi^\mu(S) \subset \Omega \mid S \in \mathcal{X}\}$. By Lemma 1, Φ^μ defined by $\cup_{S \in \mathcal{X}} \Phi^\mu(S)$ equals $\cup_{x \in X} \Pi^\mu(\{x\})$. By Corollary 1, for any SCC $f : \Theta \to \mathcal{X}$, $\Theta \not\subseteq \Phi^\mu$, we know that $f \notin \mathcal{F}^\mu$. Hence, in what follows we restrict attention to cases with $\Theta \subset \Phi^\mu$. So, for all $\theta \in \Theta \subset \Phi^\mu = \cup_{x \in X} \Pi^\mu(\{x\})$, $NE^\mu(\theta) \neq \emptyset$. Recalling that $\mathcal{S}^\mu = \{S \in \mathcal{X} \mid \exists \omega \in \Omega \text{ s.t. } NE^\mu(\omega) = S\}$, and $\Phi^\mu = \cup_{\tilde{S} \in \mathcal{S}^\mu} \Phi^\mu(\tilde{S})$, by Corollary 1, we observe the following: Any SCC $f : \Theta \to \mathcal{X}$ with $f(\theta) \notin \mathcal{S}^\mu$ for some $\theta \in \Theta$ implies $f \notin \mathcal{F}^\mu$, i.e., f is not Nash implementable by μ. As a result, we restrict attention to situations in which for all $\theta \in \Theta$, $f(\theta) \in \mathcal{S}^\mu$. But then as $\mathfrak{F}^\mu := \{\Phi^\mu(S) \subset \Omega \mid S \in \mathcal{S}^\mu\}$ is a partition of Φ^μ, the characterization of SCCs that are Nash implementable by μ are as follows (as described by Corollary 2): For any $\theta \in \Theta \subset \Phi^\mu$, $f(\theta)$ must equal S such that θ is in the partition $\Phi^\mu(S)$. These deliver the following:

Theorem 2 *Given mechanism μ,*

$$\mathcal{F}^\mu = \left\{ f : \Theta \to \mathcal{S}^\mu \mid f^{-1}(S) \subset \Phi^\mu(S), \forall S \in \mathcal{S}^\mu \right\}. \tag{3}$$

In our example with the mechanism μ defined as in Table 1, $\mathcal{S}^\mu = \{\{a\}, \{b\}, \{c\}, \{a, c\}, \{b, c\}\}$, Φ^μ is a strict subset of Ω, and the partition of Φ^μ, $\{\Phi^\mu(S) \mid S \in \mathcal{S}^\mu\}$, is as given in Table 2. Thus, only SCCs that are defined on a *restricted* domain $\Theta \subset \Phi^\mu$ and resulting in a set of alternatives other than $\{a, b\}$ and $\{a, b, c\}$ are Nash implementable by μ. Indeed, the set of Nash implementable SCCs by our mechanism μ is as follows: If $f(\theta) = S$ with $S \in \{\{a\}, \{b\}, \{c\}, \{a, c\}, \{b, c\}\}$, then $\theta \in \Phi^\mu(S)$ where $\Phi^\mu(S)$ is as given in Table 2.

3.2 Maximal domains

Implementation literature often involves an SCC defined on the full domain of preferences, Ω, given by $f : \Omega \to \mathcal{X}$. When Nash implementation with a fixed mechanism μ is under consideration, the natural question concerns the scope of the domain of preferences, $\Theta \subset \Omega$, under which SCC f restricted to Θ is Nash implementable by μ.

Efficiency provides a natural example towards that regard. Following de Clippel (2014) which extends the notion of efficiency to behavioral environments, the ***efficient SCC*** $f_{\text{eff}} : \Omega \to X$ is defined as follows: For all $\omega \in \Omega$,

$$f_{\text{eff}}(\omega) := \left\{ x \in X \mid \exists (Y_i)_{i \in N} \in \mathcal{X}^N \text{ s.t. } x \in \cap_{i \in N} C_i^\omega(Y_i) \text{ and } \cup_{i \in N} Y_i = X \right\}. \tag{4}$$

When individuals are all rational (i.e., their choice correspondences satisfy the weak axiom of revealed preferences), de Clippel efficiency coincides with Pareto optimality.

Recall that in Example 1 with mechanism μ defined as in Table 1, we have that $\mathcal{S}^\mu = \{\{a\}, \{b\}, \{c\}, \{a, c\}, \{b, c\}\}$, and Φ^μ and its partition $\{\Phi^\mu(S) \mid S \in \mathcal{S}^\mu\}$ are as given in Table 2. Therefore, our mechanism μ cannot implement $f_{\text{eff}} : \Omega \to \mathcal{X}$ in Nash equilibrium due to Corollary 1 as Φ^μ is a strict subset of Ω. In what follows, we characterize the maximal domain of preferences under which f_{eff} is Nash implementable by μ. Towards that regard, we present the association between Nash implementability by μ and efficiency in Table 3, where the corresponding efficient alternatives are depicted with circles.

Table 3 Efficient alternatives associated with mechanism μ

$$\overbrace{\underset{\textcircled{a}}{\begin{pmatrix} a & a \\ b & b \\ c & c \end{pmatrix}}, \underset{\textcircled{a}}{\begin{pmatrix} a & a \\ b & c \\ c & b \end{pmatrix}}, \underset{\textcircled{a}\textcircled{c}}{\begin{pmatrix} a & c \\ b & a \\ c & b \end{pmatrix}}, \underset{\textcircled{a}}{\begin{pmatrix} a & a \\ c & b \\ b & c \end{pmatrix}}, \underset{\textcircled{a}\textcircled{c}}{\begin{pmatrix} c & a \\ a & b \\ b & c \end{pmatrix}}}^{\Phi^{\mu}(\{a\})}, \overbrace{\underset{\textcircled{c}}{\begin{pmatrix} c & c \\ a & b \\ b & a \end{pmatrix}}, \underset{\textcircled{c}}{\begin{pmatrix} c & c \\ a & a \\ b & b \end{pmatrix}}, \underset{\textcircled{a}\textcircled{c}}{\begin{pmatrix} c & a \\ b & c \\ a & b \end{pmatrix}}, \underset{\textcircled{c}}{\begin{pmatrix} c & c \\ b & a \\ a & b \end{pmatrix}}, \underset{\textcircled{a}\textcircled{c}}{\begin{pmatrix} a & c \\ c & b \\ b & a \end{pmatrix}}}^{\Phi^{\mu}(\{c\})}, \overbrace{\underset{\textcircled{a}}{\begin{pmatrix} a & a \\ c & c \\ b & b \end{pmatrix}}, \underset{\textcircled{a}\textcircled{c}}{\begin{pmatrix} a & c \\ c & a \\ b & b \end{pmatrix}}, \underset{\textcircled{a}\textcircled{c}}{\begin{pmatrix} c & a \\ a & c \\ b & b \end{pmatrix}}}^{\Phi^{\mu}(\{a,c\})}, \overbrace{\underset{\textcircled{c}}{\begin{pmatrix} c & c \\ b & b \\ a & a \end{pmatrix}}}^{\Phi^{\mu}(\{b,c\})}$$

$$\overbrace{\underset{\textcircled{b}}{\begin{pmatrix} b & b \\ a & a \\ c & c \end{pmatrix}}, \underset{\textcircled{b}}{\begin{pmatrix} b & b \\ a & c \\ c & a \end{pmatrix}}, \underset{\textcircled{b}}{\begin{pmatrix} b & b \\ c & a \\ a & c \end{pmatrix}}, \underset{\textcircled{b}}{\begin{pmatrix} b & b \\ c & c \\ a & a \end{pmatrix}}, \underset{\textcircled{b}\textcircled{c}}{\begin{pmatrix} b & c \\ a & b \\ c & a \end{pmatrix}}, \underset{\textcircled{b}\textcircled{c}}{\begin{pmatrix} b & c \\ c & b \\ a & a \end{pmatrix}}, \underset{\textcircled{b}\textcircled{c}}{\begin{pmatrix} c & b \\ b & a \\ a & c \end{pmatrix}}, \underset{\textcircled{b}\textcircled{c}}{\begin{pmatrix} c & b \\ b & c \\ a & a \end{pmatrix}}, \underset{\textcircled{a}\textcircled{b}}{\begin{pmatrix} b & a \\ a & b \\ c & c \end{pmatrix}}, \underset{\textcircled{a}\textcircled{b}}{\begin{pmatrix} b & a \\ c & b \\ a & c \end{pmatrix}}, \underset{\textcircled{a}\textcircled{b}\textcircled{c}}{\begin{pmatrix} c & a \\ b & b \\ a & c \end{pmatrix}}, \underset{\textcircled{a}\textcircled{b}}{\begin{pmatrix} a & b \\ b & a \\ c & c \end{pmatrix}}, \underset{\textcircled{a}\textcircled{b}}{\begin{pmatrix} a & b \\ b & c \\ c & a \end{pmatrix}}, \underset{\textcircled{a}\textcircled{b}\textcircled{c}}{\begin{pmatrix} a & c \\ b & b \\ c & a \end{pmatrix}}}^{\Phi^{\mu}(\{b\})}$$

Table 4 Maximal domain of efficiency Nash implementable via mechanism μ

$$\overbrace{\underset{\textcircled{a}}{\begin{pmatrix} a & a \\ b & b \\ c & c \end{pmatrix}}, \underset{\textcircled{a}}{\begin{pmatrix} a & a \\ b & c \\ c & b \end{pmatrix}}, \underset{\textcircled{a}}{\begin{pmatrix} a & a \\ c & b \\ b & c \end{pmatrix}}}^{\Phi^{\mu}(\{a\})}, \overbrace{\underset{\textcircled{c}}{\begin{pmatrix} c & c \\ a & b \\ b & a \end{pmatrix}}, \underset{\textcircled{c}}{\begin{pmatrix} c & c \\ a & a \\ b & b \end{pmatrix}}, \underset{\textcircled{c}}{\begin{pmatrix} c & c \\ b & a \\ a & b \end{pmatrix}}}^{\Phi^{\mu}(\{c\})}, \overbrace{\underset{\textcircled{a}\textcircled{c}}{\begin{pmatrix} a & c \\ c & a \\ b & b \end{pmatrix}}, \underset{\textcircled{a}\textcircled{c}}{\begin{pmatrix} c & a \\ a & c \\ b & b \end{pmatrix}}}^{\Phi^{\mu}(\{a,c\})}, \overbrace{\underset{\textcircled{b}}{\begin{pmatrix} b & b \\ a & a \\ c & c \end{pmatrix}}, \underset{\textcircled{b}}{\begin{pmatrix} b & b \\ a & c \\ c & a \end{pmatrix}}, \underset{\textcircled{b}}{\begin{pmatrix} b & b \\ c & a \\ a & c \end{pmatrix}}, \underset{\textcircled{b}}{\begin{pmatrix} b & b \\ c & c \\ a & a \end{pmatrix}}}^{\Phi^{\mu}(\{b\})}$$

From Table 3, we observe that at $\omega = (abc, cab)$, $NE^{\mu}(\omega) = \{a\}$ but $f_{\text{eff}}(\omega) = \{a, c\}$; at $\tilde{\omega} = (acb, acb)$, $NE^{\mu}(\tilde{\omega}) = \{a, c\}$ but $f_{\text{eff}}(\tilde{\omega}) = \{a\}$. Thus, neither ω nor $\tilde{\omega}$ can be in the maximal domain of preferences under which mechanism μ implements f_{eff} in Nash equilibrium. Hence, the maximal domain of preferences under which mechanism μ implements f_{eff} in Nash equilibrium consists of all states $\hat{\omega}$ such that $NE^{\mu}(\hat{\omega}) = f_{\text{eff}}(\hat{\omega})$. Similarly, $\{b, c\}$ cannot be sustained as a Nash equilibrium outcome of μ when μ Nash implements f_{eff}. This is because $\Phi^{\mu}(\{b, c\}) = (cba, cba)$ while $f_{\text{eff}}(cba, cba) = \{c\}$; i.e., the unique efficient alternative in the unique state that sustains $\{b, c\}$ as a Nash equilibrium outcome of μ equals c. These lead us to conclude that the maximal domain of preferences under which our mechanism μ implements f_{eff} in Nash equilibrium is as given in Table 4.

In general, the ***maximal domain of preferences*** under which a given SCC $f : \Omega \to \mathcal{X}$ is Nash implementable by mechanism μ is

$$\Phi_f^{\mu} := \{\omega \in \Omega \mid f(\omega) = S \text{ if and only if } \omega \in \Phi^{\mu}(S)\}. \tag{5}$$

The implementation literature contains interesting work identifying the *maximal domain of preferences,* Θ_f*,under which a given SCC* $f : \Omega \to \mathcal{X}$*restricted to* $\Theta_f \subset \Omega$ *is Maskin monotonic*.[6] To provide a comparison with ours, first we remind the reader of the following well-known definition:

Definition 2 An SCC $f : \Omega \to \mathcal{X}$ restricted to a domain $\Theta \subset \Omega$ is ***Maskin monotonic on domain*** Θ if $x \in f(\theta) \setminus f(\tilde{\theta})$ with $\theta, \tilde{\theta} \in \Theta$ implies there exists $j \in N$ such that $L_j^{\theta}(x) \not\subseteq L_j^{\tilde{\theta}}(x)$.

The following theorem is a reaffirmation of Maskin's necessity result on restricted domains: The maximal domain of preferences under which f is Nash implementable by mechanism μ is a subset of the maximal domain of preferences under which f is Maskin monotonic.

Theorem 3 $\Phi_f^{\mu} \subset \Theta_f$.

To see that this containment relation may be strict, we turn to Example 1: $\Phi_{f_{\text{eff}}}^{\mu}$ is as in Table 4, while we know that $\Theta_f = \Omega$ since SCC f_{eff} is Nash implementable (by the canonical mechanism) on the whole domain Ω (see de Clippel 2014).

4 Computational Nash implementation of an SCC

We now turn to the more standard approach in implementation theory. We revisit the necessity and sufficiency results from the literature and exemplify how we can employ computational tools to analyze the scope of Nash implementation of an SCC.

6 See Sanver (2008) and Sanver (2017) among others.

4.1 Necessity

We start with the necessary conditions for Nash implementation. Before going on, we recall the definition of Nash implementability of an SCC: An SCC $f : \Theta \to \mathcal{X}$ is Nash implementable if there exists a mechanism $\mu \in \mathcal{M}$ such that $f(\theta) = NE^{\mu}(\theta)$ for all $\theta \in \Theta$.

Below is Maskin's necessity result for Nash implementability of an SCC.

Theorem 4 (Maskin 1999) *If $f : \Theta \to \mathcal{X}$ is Nash implementable, then it is Maskin-monotonic.*

de Clippel (2014) generalizes Maskin's result on Nash implementation to behavioral domains, domains where individuals' choices do not necessarily satisfy WARP. The necessary condition that de Clippel (2014) identifies for Nash implementation of an SCC is the existence of a collection of sets that are consistent with this SCC:

Definition 3 For any given SCC $f : \Theta \to \mathcal{X}$, we say that a collection of sets $\mathbb{S} := \{S_i(x, \theta) \mid i \in N, \theta \in \Theta, x \in f(\theta)\}$ is consistent with f if for any $\theta, \theta' \in \Theta$,

(i) if $x \in f(\theta)$, then $x \in \cap_{i \in N} C_i^{\theta}(S_i(x, \theta))$,

(ii) if $x \in f(\theta) \setminus f(\theta')$, then $x \notin \cap_{i \in N} C_i^{\theta'}(S_i(x, \theta))$.

In words, a consistent collection of sets of alternatives, is a family of choice sets indexed for each individual $i \in N$ and each state $\theta \in \Theta$ and each alternative x that is f-optimal at θ such that the following hold: Alternative x is chosen by every individual i at state θ from the corresponding choice set, $S_i(x, \theta)$; and if alternative x is f-optimal at state θ but not at state θ', then there is an individual j who does not choose x at θ' from $S_j(x, \theta)$, j's choice set corresponding to x and θ.

To demonstrate consistency, we revert to Example 1 and consider a domain $\Theta = \{\theta, \theta'\}$ where $\theta = (cab, acb)$ and $\theta' = (cba, abc)$ and the SCC f is such that $f(\theta) = \{a, c\}$ and $f(\theta') = \{b\}$. Then, by (i) of consistency, the collection $\mathbb{S}$ given by $S_A(a, \theta)$, $S_A(c, \theta)$, $S_B(a, \theta)$, $S_B(c, \theta)$, $S_A(b, \theta')$, and $S_B(b, \theta')$ must be such that the following hold: $a \in C_i^{\theta}(S_i(a, \theta))$, $c \in C_i^{\theta}(S_i(c, \theta))$, and $b \in C_i^{\theta'}(S_i(b, \theta)$ for $i = A, B$. Thus, we observe that

$$\begin{aligned} &S_A(a, \theta) \in \Big\{\{a, b\}, \boxed{\{a\}}\Big\}, \ S_A(c, \theta) \in \Big\{\{a, b, c\}, \boxed{\{a, c\}}, \{b, c\}, \{c\}\Big\}, \\ &\quad \text{and } S_A(b, \theta') \in \Big\{\boxed{\{a, b\}}, \{b\}\Big\}; \\ &S_B(a, \theta) \in \Big\{\boxed{\{a, b, c\}}, \{a, b\}, \{a, c\}, \{a\}\Big\}, \ S_B(c, \theta) \in \Big\{\boxed{\{b, c\}}, \{c\}\Big\}, \\ &\quad \text{and } S_B(b, \theta') \in \Big\{\boxed{\{b, c\}}, \{b\}\Big\}. \end{aligned} \tag{6}$$

The collection indicated with rectangles above satisfies (i) of consistency, but not (ii): $a \in f(\theta)$ and $a \notin f(\theta')$, but $a \in C_A^{\theta'}(\{a\}) \cap C_B^{\theta'}(\{a, b, c\})$ where $S_A(a, \theta) = \{a\}$ and $S_B(a, \theta) = \{a, b, c\}$. Therefore, this collection is not consistent with f. That said we observe that

$$\begin{aligned} &S_A(a, \theta) = \{a, b\}, \ S_A(c, \theta) = \{a, b, c\}, \ \text{and } S_A(b, \theta') = \{b\}; \\ &S_B(a, \theta) = \{a\}, \ S_B(c, \theta) = \{b, c\}, \ \text{and } S_B(b, \theta') = \{b, c\}. \end{aligned} \tag{7}$$

is a collection of sets consistent with f: Notice that this collection satisfies (i) of consistency as it is one of the collections captured in (6). Now, for (ii) of consistency consider the following: $a \in f(\theta) \setminus f(\theta')$ and $a \notin C_A^{\theta'}(\{a, b\}) = \{b\}$ (where $S_A(a, \theta) = \{a, b\}$); $c \in f(\theta) \setminus f(\theta')$ and $c \notin C_B^{\theta'}(\{b, c\}) = \{b\}$ (where $S_B(c, \theta) = \{b, c\}$); $b \in f(\theta') \setminus f(\theta)$ and $b \notin C_B^{\theta}(\{b, c\}) = \{c\}$ (where $S_B(b, \theta') = \{b, c\}$). In what follows, we provide another collection of sets consistent with f that follows from our necessity result with consistency, Theorem 5.

In Sect. 4.3, we explain how to identify consistent collections using the computational tools we provide.

When individuals have rational preferences, Maskin-monotonicity of an SCC is equivalent to the existence of a collection of sets that is consistent with this SCC:

Lemma 3 *An SCC $f : \Theta \to \mathcal{X}$ is Maskin-monotonic if and only if there exists a collection of sets that is consistent with f.*

The following theorem is de Clippel's necessity result for Nash implementation on the behavioral domain.

Theorem 5 (de Clippel 2014) *If $f : \Theta \to \mathcal{X}$ is Nash implementable, then there exists a collection of sets consistent with f.*

In the rational domain, Theorem 5 follows directly from Theorem 4 and Lemma 3. Notwithstanding, employing consistency may provide computational advantages for Nash implementation as searching for such collections may be easier or more intuitive than checking whether or not an SCC is Maskin-monotonic. Indeed, the planner's selection from consistent collections may help her design cognitively simpler or more intuitive/appealing mechanisms.

Going back to Example 1, we note that Theorem 2 establishes that the mechanism of Table 1 implements $f : \Theta \to \mathcal{X}$ where $\Theta = \{\theta, \theta'\}$ with $\theta = (cab, acb)$ and $\theta' = (cba, abc)$, and $f(\theta) = \{a, c\}$ and $f(\theta') = \{b\}$. As a result, Theorem 5 empowers us to conclude that the following collection is consistent with f:

$$\begin{array}{l} S_A(a, \theta) = \{a, b\},\ S_A(c, \theta) = \{a, b, c\},\ \text{and } S_A(b, \theta') = \{a, b\}; \\ S_B(a, \theta) = \{a, b, c\},\ S_B(c, \theta) = \{b, c\},\ \text{and } S_B(b, \theta') = \{b, c\}. \end{array} \tag{8}$$

This is due to the following: Given the mechanism of Table 1, one can show that at θ, (U, M) and (D, L) are the only Nash equilibria, while at θ' the unique Nash equilibrium equals (D, M). As $g(U, M) = a$, we obtain $S_A(a, \theta) = O_A^{\mu}(M) = \{a, b\}$ and $S_B(a, \theta) = O_B^{\mu}(U) = \{a, b, c\}$. Similarly, $g(D, L) = c$ implies $S_A(c, \theta) = O_A^{\mu}(L) = \{a, b, c\}$ and $S_B(c, \theta) = O_B^{\mu}(D) = \{b, c\}$; $g(D, M) = b$ implies $S_A(b, \theta') = O_A^{\mu}(M) = \{a, b\}$ and $S_B(b, \theta') = O_B^{\mu}(D) = \{b, c\}$.

In general, if an SCC f is implementable in Nash equilibrium by a mechanism μ, then for every state θ and for every f-optimal alternative at θ, $x \in f(\theta)$, there is a Nash equilibrium of μ at θ that delivers x. In turn, the collection of opportunity sets obtained at this Nash equilibrium satisfies (i) of consistency. Moreover, if x is not f optimal at θ', then there must be an agent who does not choose x from the opportunity set associated with the Nash equilibrium at θ that sustains x; because otherwise, x

would be a Nash equilibrium outcome at θ' as well and hence x would have to be in $f(\theta')$, a contradiction delivering the conclusion that the collection of opportunity sets also satisfies (ii) of consistency.

Nash implementation with two individuals is inherently different than Nash implementation with three or more individuals. A straightforward but important observation is that any message of an individual can be thought of as an opportunity set generated for the other individual in a two-individual mechanism. This helps us sharpen consistency for the case of two individuals as follows:

Definition 4 For any given SCC $f : \Theta \to \mathcal{X}$, we say that a pair of collections of sets $(\mathbb{S}_1, \mathbb{S}_2)$ with $\mathbb{S}_1 := \{S_1(x, \theta) \mid \theta \in \Theta, x \in f(\theta)\}$ and $\mathbb{S}_2 := \{S_2(x, \theta) \mid \theta \in \Theta, x \in f(\theta)\}$ is two-individual consistent with f if for any $\theta, \theta' \in \Theta$, and any $x, y \in X$

(i) If $x \in f(\theta)$, then $x \in C_1^{\theta}(S_1(x, \theta)) \cap C_2^{\theta}(S_2(x, \theta))$,
(ii) If $x \in f(\theta) \setminus f(\theta')$, then there is $j \in \{1, 2\}$ such that $x \notin C_j^{\theta'}(S_j(x, \theta))$.
(iii) If $x \in f(\theta)$ and $y \in f(\theta')$, then $S_1(x, \theta) \cap S_2(y, \theta') \neq \emptyset$.

Going back to Example 1 where the SCC $f : \Theta \to \mathcal{X}$ is such that $\Theta = \{\theta, \theta'\}$ with $\theta = (cab, acb)$ and $\theta' = (cba, abc)$, and $f(\theta) = \{a, c\}$ and $f(\theta') = \{b\}$, we wish to remind that the collection given in (7) is consistent with f. But this collection is not two-individual consistent since (iii) of two-individual consistency does not hold as $S_A(b, \theta') \cap S_B(a, \theta) = \{b\} \cap \{a\} = \emptyset$. Indeed, thanks to our necessity result with two individuals, Theorem 6, presented below, the collection given in (8) is two-individual consistent as the two-individual mechanism of Table 1 Nash implements f.

The following presents our necessity result with two individuals:[7]

Theorem 6 *Let $n = 2$. If $f : \Theta \to \mathcal{X}$ is Nash implementable, then there exists a pair of collections of sets, $(\mathbb{S}_1, \mathbb{S}_2)$, that is two-individual consistent with f.*

4.2 Simplicity

There is growing interest in simple mechanisms in the mechanism design literature.[8] However, “[t]he question as to what constitutes a "simple" mechanism is a difficult and controversial one” (Dutta et al. 1995).

As in Barlo and Dalkıran (2020), we consider the total number of message profiles of a mechanism as a measure of its *simplicity*.[9] This measure is similar in spirit with the total size of message spaces used to analyze communication complexity in Nisan and Segal (2006), Segal (2007, 2010) building upon the literature on realization,

[7] The necessary condition analyzed here can be strengthened by replacing (iii) of two-individual consistency with the following requirement: (iii') there exists a function $e : X \times \Theta \times X \times \Theta \to X$ such that for any $\theta, \theta' \in \Theta$, $x \in f(\theta)$, and $x' \in f(\theta')$ such that $(iii'.1)$ $e(x, \theta, x', \theta') \in S_1(x, \theta) \cap S_2(x', \theta')$; and $(iii'.2)$ $e(x, \theta, x', \theta') \in f(\theta^*)$ if $e(x, \theta, x', \theta') \in C_1^{\theta^*}(S_1(x, \theta)) \cap C_2^{\theta^*}(S_2(x', \theta'))$. In the current paper, we have chosen the weaker necessary condition presented in Definition 4 for computational reasons.

[8] See for example, Li (2017), Börgers and Li (2019), and Pycia and Troyan (2019).

[9] Parts of this section overlap with Section 9 of the current version of Barlo and Dalkıran (2020) analyzing ex-post implementation in a behavioral incomplete information setup. Moreover, Section 9 (along with its “Appendix F”) will be discarded in the next draft of that paper due to the associated editorial decisions.

message processes, and communication protocols.[10] These studies aim to describe the "minimal information that must be elicited by the designer in order to achieve the goals" within the framework of *nondeterministic communication protocols* with privacy preservation and verification properties. On the other hand, our analysis seeks to answer the same question restricting attention directly to mechanisms implementing a given goal. As a result, even though our simplicity notions are similar, they do not produce perfectly aligned implications (see "Appendix 1" for the details).

Our necessity results bring about useful insights into the lower bounds on the number of messages required for Nash implementation: As we restrict ourselves to finite setups, there can only be finitely many consistent collections of an SCC. Let $\{\mathbb{S}^k\}_{k=1,\ldots,K}$ be the set of all collections of sets consistent with a given SCC, $f : \Theta \to \mathcal{X}$, represented by $\mathbb{S}^k = \{\mathbb{S}_i^k\}_{i\in N}$ for each $k \in \{1, \ldots, K\}$ with $\mathbb{S}_i^k = \{S_i^k(x, \theta) \mid x \in f(\theta), \theta \in \Theta\}$. Thanks to our necessity result, we know that if f is Nash implementable, then any mechanism that Nash implements f has to induce one of the consistent collections in $\{\mathbb{S}^k\}_{k\in K}$. This observation leads us to the following:

Theorem 7 *Let $f : \Theta \to \mathcal{X}$ be Nash implementable and $\{\mathbb{S}^k\}_{k=1,\ldots,K}$ be the set of all collections of sets consistent with f. Then, in any mechanism that Nash implements f,*

(i) the minimum number of messages of individual i is $\min_{k=1,\ldots,K} \max_{S\in\mathbb{S}_i^k} \#S$,
(ii) the minimum number of message profiles required for the individuals other than i is $\min_{k=1,\ldots,K} \#\mathbb{S}_i^k$, and
(iii) the minimum number of total message profiles is
$$\max\left\{\min_{k=1,\ldots,K} \max_{i\in N} (\#\mathbb{S}_i^k \max_{S\in\mathbb{S}_i^k} \#S), \min_{k=1,\ldots,K} (\textstyle\prod_{i\in N} \max_{S\in\mathbb{S}_i^k} \#S)\right\}.$$

We present the proof of Theorem 7 here as a discussion: Suppose that $\mathbb{S}^k$ is a consistent collection induced by a mechanism that Nash implements f. Then, individual i is able to generate any set in $\mathbb{S}_i^k$, and hence i must have at least as many messages as the cardinality of the maximal set in $\mathbb{S}_i^k$ in this mechanism. This implies that the minimum number of messages of individual i in any mechanism that Nash implements f is $\min_{k\in\{1,\ldots,K\}} \max_{S\in\mathbb{S}_i^k} \#S$. On the other hand, for each different set in $\mathbb{S}_i^k$, there must exist a particular message profile of the individuals other than i that should allow i to generate this particular set, which implies that in any mechanism that Nash implements f the minimum number of message profiles required for the individuals other than i is $\min_{k\in\{1,\ldots,K\}} \#\mathbb{S}_i^k$. Hence, the total number of message profiles in this mechanism must be at least as much as the cardinality of $\mathbb{S}_i^k$ times the cardinality of the maximal set in $\mathbb{S}_i^k$ for each $i \in N$. That is, the number of message profiles in this mechanism must be at least $\max_{i\in N}(\#\mathbb{S}_i^k \max_{S\in\mathbb{S}_i^k} \#S)$. Moreover, the total number of message profiles in this mechanism must also be greater than $\prod_{i\in N} \max_{S\in\mathbb{S}_i^k} \#S$. Therefore, the total number of message profiles in any mechanism that Nash implements f must be greater than or equal to $\max\{\min_{k\in\{1,\ldots,K\}} \max_{i\in N}(\#\mathbb{S}_i^k \max_{S\in\mathbb{S}_i^k} \#S),$ $\min_{k\in\{1,\ldots,K\}}(\prod_{i\in N} \max_{S\in\mathbb{S}_i^k} \#S)\}$.

[10] See Mount and Reiter (1974, 1996, 2002), Hurwicz et al. (1980), Saari (1984), Williams (1986), Reichelstein Reiter (1988), and Hurwicz and Reiter (2006).

Table 5 The feasible domain of preferences, Θ, and the corresponding efficient alternatives

$$\Theta = \left\{ \overset{\theta_1}{\underset{\textcircled{x}}{\begin{pmatrix} x & x \\ y & y \\ z & z \end{pmatrix}}}, \overset{\theta_2}{\underset{\textcircled{x}\textcircled{z}}{\begin{pmatrix} x & z \\ z & x \\ y & y \end{pmatrix}}}, \overset{\theta_3}{\underset{\textcircled{z}}{\begin{pmatrix} z & z \\ x & y \\ y & x \end{pmatrix}}} \right\}.$$

Table 6 A two-individual consistent collection $(\mathbb{S}_A, \mathbb{S}_B)$ of $f_{\text{eff}} : \Theta \to \mathcal{X}$

$\mathbb{S}_A$:	$S_A(x, \theta_1) = \{x, z\}$	$S_A(x, \theta_2) = \{x, z\}$	$S_A(z, \theta_2) = \{y, z\}$	$S_A(z, \theta_3) = \{y, z\}$
$\mathbb{S}_B$:	$S_B(x, \theta_1) = \{x, y\}$	$S_B(x, \theta_2) = \{x, y\}$	$S_B(z, \theta_2) = \{z\}$	$S_B(z, \theta_3) = \{z\}$

Table 7 A 2×2 mechanism that Nash implements $f_{\text{eff}} : \Theta \to \mathcal{X}$

		Bob	
		$\{x, z\}$	$\{y, z\}$
Alice	$\{x, y\}$	x	y
	$\{z\}$	z	z

4.3 Computation of consistent collections: Example 2

For expositional simplicity, we exemplify the computation of consistent collections of an SCC in a two-individual setup:[11] Suppose *Alice* and *Bob* are to collectively choose one of the options in $X = \{x, y, z\}$. The feasible domain of preference is $\Theta = \{\theta_1, \theta_2, \theta_3\}$ with $\theta_1 = (xyz, xyz)$, $\theta_2 = (xzy, zxy)$, and $\theta_3 = (zxy, zyx)$.

We analyze the Nash implementation of the Pareto efficient SCC, $f_{\text{eff}} : \Theta \to \mathcal{X}$, which is given as $f_{\text{eff}}(\theta_1) = \{x\}$, $f_{\text{eff}}(\theta_2) = \{x, z\}$, $f_{\text{eff}}(\theta_3) = \{z\}$. The feasible domain of preferences, Θ, and the Pareto efficient SCC are summarized in Table 5, where the corresponding efficient alternatives are depicted with circles.

Our Python code that computes the two-individual consistent collections reports that there are 1233 two-individual consistent collections for $f_{\text{eff}} : \Theta \to \mathcal{X}$ described above. Let $\{\mathbb{S}^k\}_{k=1}^{1233}$ be the corresponding set of all consistent collections of sets of f_{eff}. We observe that $\min_{k=1,\ldots,K} \max_{S \in \mathbb{S}_i^k} \#S = 2$ for both $i = A, B$. That is, the best one can hope for is a 2×2 mechanism. Table 6 provides a two-individual consistent collection where each individual has two sets in their collections and the sets in these collections with maximal number of elements has two elements.[12]

Indeed, it is straightforward to check that the mechanism given in Table 7 Nash implements $f_{\text{eff}} : \Theta \to \mathcal{X}$.

Looking at the mechanism given in Table 7, we observe the following: Alice has the option to enforce alternative z on Bob. Furthermore, considering state θ_2, $\{x, z\}$ is a weakly dominant action for Bob. That is, even though the Nash equilibrium message

[11] We refer the interested reader to the "Examples_in_the_Manuscript\Example_2_in_the_Manuscript" in "Python_Codes_Computational_Implementation_Barlo_Dalkiran.zip" for the computational codes/outputs.

[12] Two-individual consistent collection #1202 given in "Two-Individual_Consistent_Collections.xslx" in the "Examples_in_the_Manuscript\Example_2_in_the_Manuscript" folder.

Table 8 Another two-individual consistent collection $(\mathbb{S}_A, \mathbb{S}_B)$ of $f_{\text{eff}} : \Theta \to \mathcal{X}$

$\mathbb{S}_A$:	$S_A(x, \theta_1) = \{x, y\}$	$S_A(x, \theta_2) = \{x, y\}$	$S_A(z, \theta_2) = \{y, z\}$	$S_A(z, \theta_3) = \{x, z\}$
$\mathbb{S}_B$:	$S_B(x, \theta_1) = \{x, y\}$	$S_B(x, \theta_2) = \{x, y\}$	$S_B(z, \theta_2) = \{y, z\}$	$S_B(z, \theta_3) = \{x, z\}$

Table 9 A symmetric 3×3 mechanism that Nash implements $f_{\text{eff}} : \Theta \to \mathcal{X}$

		Bob		
		Veto $z \to \{x, y\}$	Veto $y \to \{x, z\}$	Veto $x \to \{y, z\}$
Alice	Veto $z \to \{x, y\}$	x	x	y
	Veto $y \to \{x, z\}$	x	x	z
	Veto $x \to \{y, z\}$	y	z	z

profile $(z, \{y, z\})$ at θ_2 leads to z and z is f_{eff}-optimal at θ_2, Bob might rather prefer the weakly dominant message $\{x, z\}$ at θ_2. Hence, the designer might look for a "better" mechanism where an undominated Nash equilibrium achieves the goal.

Searching through the list of consistent collections of f_{eff} computed by our Python codes, the consistent collection in Table 8 leads to a more appealing mechanism.[13]

Let each individual veto one alternative. If the vetoed alternatives differ the remaining alternative becomes the outcome. On the other hand, if both individuals veto x, then the outcome is z and if they both veto z, then the outcome is x. Finally, if they both veto y the outcome is x.[14] This mechanism is depicted in Table 9.

Now, we show that the mechanism of Table 9 sustains the two-individual consistent collection of sets depicted in Table 8. At θ_1, a Nash equilibrium of this mechanism is (Veto z, Veto z) which delivers x and there are no other Nash equilibrium outcomes and hence $S_i(x, \theta_1) = O_i^{\mu}(\text{Veto } z) = \{x, y\}$ for $i = A, B$. At θ_2, x, z are the only Nash equilibrium outcomes of this mechanism sustained by Nash equilibrium (Veto z, Veto z) for x, and (Veto x, Veto x) for z. Hence, $S_i(x, \theta_2) = O_i^{\mu}(\text{Veto } z) = \{x, y\}$ and $S_i(z, \theta_2) = O_i^{\mu}(\text{Veto } x) = \{y, z\}$ for $i = A, B$. Finally, at θ_3, a Nash equilibrium of this mechanism is (Veto x, Veto x) which delivers z and there are no other Nash equilibrium outcomes and hence $S_i(z, \theta_3) = O_i^{\mu}(\text{Veto } x) = \{y, z\}$ for $i = A, B$.

This example showcases that employing the specific features of consistent collections may help the planner design more appealing mechanisms. Notwithstanding, we wish to note that even though the mechanism of Table 9 is more appealing (on the grounds of implementation via undominated Nash equilibria and its intuitive use of vetoing), it is less simple than the mechanism of Table 7 as the simplicity measure of the former is nine while that of the latter equals four. Therefore, this example also exhibits a potential trade-off between our notion of simplicity and intuitive criteria that the planner may wish to use when designing mechanisms.

13 Two-individual consistent collection #738 given in "Two-Individual_Consistent_Collections.xslx" in the "Examples_in_the_Manuscript\Example_2_in_the_Manuscript" folder.

14 When both individuals veto y, choosing the outcome as z also works.

4.4 Sufficiency

We now turn to the sufficiency results. Below, we first slightly improve the two well-known sufficiency results in the literature and then illustrate how we can computationally check them for a given SCC.

We start with the well-known no-veto-power (NVP) property:

Definition 5 An SCC $f : \Theta \to \mathcal{X}$ satisfies the no-veto-power (NVP) property if for any $j \in N$, $x \in \cap_{i \neq j} C_i^{\theta}(X)$ implies $x \in f(\theta)$.

In words, an SCC, f, satisfies the NVP property if an alternative is f-optimal whenever there are at least $n-1$ individuals who choose this alternative from the set of all possible alternatives. That is, if there are at least $n-1$ individuals who top-rank an alternative, then this alternative must be f-optimal.

We continue with the economic-environment property:

Definition 6 The economic-environment (EE) property holds if for any $\omega \in \Omega$ and any $x \in X$, there are $i, j \in N$ with $i \neq j$ such that $x \notin C_i^{\omega}(X)$ and $x \notin C_j^{\omega}(X)$.

The EE property holds whenever for any alternative, there are two individuals who do not choose this particular alternative from the set of all possible alternatives. We note that the EE property implies that the NVP property holds vacuously as no alternative will be top-ranked by $n-1$ or more individuals in an economic environment.

The well-known sufficiency result of Maskin (1999) is as follows: When there are at least three individuals, any Maskin-monotonic SCC that satisfies the NVP property is Nash Implementable.

Next, we provide a slight generalization of this sufficiency result. To do so, we need to slightly modify the NVP and EE properties so that they accommodate a consistent collection.

Definition 7 A collection of sets $\mathbb{S}$ consistent with $f : \Theta \to \mathcal{X}$ satisfies

(i) the NVP* property if there exists an $\bar{X} \subset X$ such that $\cup_{S \in \mathbb{S}} S \subset \bar{X}$ and for any $j \in N$, $x \in \cap_{i \neq j} C_i^{\theta}(\bar{X})$ implies $x \in f(\theta)$;

(ii) the EE* property if there exists an $\bar{X} \subset X$ such that $\cup_{S \in \mathbb{S}} S \subset \bar{X}$ and for any $\theta \in \Theta$, any $x \in \bar{X}$, there are $i, j \in N$ with $i \neq j$ such that $x \notin C_i^{\theta}(\bar{X})$ and $x \notin C_j^{\theta}(\bar{X})$.

We are ready to present our slight generalization of Maskin (1999)'s sufficiency result:

Theorem 8 *Let $n \geq 3$. If there is a collection of sets consistent with SCC $f : \Theta \to \mathcal{X}$ that satisfies either NVP* property or EE* property, then f is Nash implementable.*

Even though the existence of a consistent collection is necessary for Nash implementation of an SCC, it is not sufficient. Our sufficiency result, on the other hand, highlights the fact that the existence of a consistent collection of sets that satisfies NVP* or EE* guarantees the Nash implementability of this SCC.

Table 10 The feasible domain of preferences for Ann, Bob, and Chris

$$\Theta = \left\{ \overset{\theta_1}{\begin{pmatrix} x & y & z \\ y & x & y \\ z & t & x \\ t & z & t \end{pmatrix}}, \overset{\theta_2}{\begin{pmatrix} y & z & z \\ z & x & y \\ x & y & x \\ t & t & t \end{pmatrix}}, \overset{\theta_3}{\begin{pmatrix} x & x & y \\ z & t & x \\ t & y & t \\ y & z & z \end{pmatrix}} \right\}.$$

We note that if $\mathbb{S}$ is a collection consistent with $f : \Theta \to \mathcal{X}$ and f satisfies the regular NVP property, then $\bar{X} = X$ implies that $\mathbb{S}$ satisfies the NVP* property. Similarly, if $\mathbb{S}$ is a collection consistent with $f : \Theta \to \mathcal{X}$ and the regular EE property holds, then $\bar{X} = X$ implies that $\mathbb{S}$ satisfies the EE* property. The converse directions of these statements do not hold, as we shall see in the following example.

4.5 Computation of consistent collections that satisfy NVP* or EE*: Example 3

We now illustrate the computation of consistent collections that satisfy NVP* or EE* through an example.[15] This example also exhibits that the sufficiency result we provide above is a slight generalization of that of Maskin (1999).

Consider *Ann*, *Bob*, and *Chris* who are to collectively choose an alternative among $X = \{x, y, z, t\}$. Suppose that the set of feasible domain of preferences Θ is as given in Table 10.

Suppose the designer would like to implement SCC $f : \Theta \to \mathcal{X}$ such that $f(\theta_1) = y$, $f(\theta_2) = z$, and $f(\theta_3) = t$.

We note that $f : \Theta \to \mathcal{X}$ does not satisfy the NVP property as x is top-ranked by Ann and Bob at θ_3, but $x \notin f(\theta_3) = \{t\}$. Furthermore, the EE property does not hold as well since z is top-ranked by Bob and Chris at θ_2 and x is top-ranked by both Ann and Bob at θ_3. Therefore, we cannot employ the sufficiency result of Maskin (1999) in this example.

Our Python codes reveal that there are 390120 consistent collections of $f : \Theta \to \mathcal{X}$. 4704 of these consistent collections satisfy the NVP* property with the associated set $\bar{X} = \{y, z, t\}$, and none of them satisfy the EE* property. From this information, we immediately learn that SCC f is Nash implementable.

The designer can search through the consistent collections to design a mechanism with some appealing criteria. To do so, the planner can filter the list of consistent collections (provided by the output of our Python Codes) based on these desirable criteria.

As an example, suppose that the planner considers the following criteria to design a mechanism: (i) no pair of individuals can enforce a specific outcome on the odd-man-out, i.e., none of the opportunity sets induced by this mechanism is a singleton; (ii) each consistent collection induced by the mechanism makes use of only two sets of alternatives; and (iii) this mechanism is one of the simplest (according to our notion

[15] We refer the interested reader to the "Examples_in_the_Manuscript\Example_3_in_the_Manuscript" in "Python_Codes_Computational_Implementation_Barlo_Dalkiran.zip" for the computational codes/outputs.

Table 11 A consistent collection $\mathbb{S} = (\mathbb{S}_A, \mathbb{S}_B, \mathbb{S}_C)$ of $f : \Theta \to \mathcal{X}$

$\mathbb{S}_A$:	$S_A(y, \theta_1) = \{y, t\}$	$S_A(z, \theta_2) = \{z, t\}$	$S_A(t, \theta_3) = \{y, t\}$
$\mathbb{S}_B$:	$S_B(y, \theta_1) = \{y, z\}$	$S_B(z, \theta_2) = \{z, t\}$	$S_B(t, \theta_3) = \{z, t\}$
$\mathbb{S}_C$:	$S_C(y, \theta_1) = \{y, t\}$	$S_C(z, \theta_2) = \{z, t\}$	$S_C(x, \theta_2) = \{z, t\}$

Table 12 A $2 \times 2 \times 2$ mechanism that Nash implements $f : \Theta \to \mathcal{X}$

Chris							
W				E			
Bob				Bob			
		L	R			L	R
Ann	U	y	z	Ann	U	t	y
	D	t	z		D	z	t

of simplicity) among those satisfying requirements (i) and (ii).[16] Filtering 390120 consistent collections of sets, we see that only three consistent collections satisfy these properties. One of these consistent collections is given in Table 11.

It is easy to see that the consistent collection $\mathbb{S}$ given in Table 11 satisfies the NVP* property: y is the best alternative for Ann and Bob in $\bar{X} = \{y, z, t\}$ at θ_1 and $f(\theta_1) = y$; similarly, z is the best alternative for Bob and Chris in $\bar{X} = \{y, z, t\}$ at θ_2 and $f(\theta_2) = z$; and finally, at θ_3, none of the alternatives is the best for two individuals simultaneously.

Therefore, we can construct a mechanism that Nash implements f by using the consistent collection given in Table 11. Indeed, mechanism $\mu = (M, g)$ where $M_A = \{U, D\}$, $M_B = \{L, R\}$ and $M_C = \{W, E\}$ with the outcome function, $g : M \to X$, presented in Table 12 Nash implements f.

This example demonstrates that even when the number of consistent collections is large, filtering them based on desirable criteria helps the planner design eligible mechanisms. Therefore, computational identification of collections consistent with a given SCC that the planner seeks to implement in Nash equilibrium offers practical and theoretical significance.

5 Literature

Following the seminal work of Maskin (1999), there has been a huge literature on Nash implementation, and it is not possible to cite all the interesting work here. Instead, we refer the reader to the following surveys on Nash implementation: Jackson (2001), Maskin and Sjöström (2002), Palfrey (2002), and Serrano (2004).

Our paper is mostly related to de Clippel (2014), which is the first paper highlighting the idea of consistent collections of an SCC. Even though de Clippel obtains his results on a behavioral domain, our results show that they are indeed useful for Nash implementation in the domain of rational preferences as well. Besides de Clippel

[16] Under the requirements of (i) and (iii), the planner seeks a consistent collection such that the number of alternatives in each set in that collection equals two.

(2014), there are other papers on implementation that takes individual choices as primitives such as Korpela (2012), Hayashi et al. (2020), Altun et al. (2021), and Barlo and Dalkıran (2021). Other relatively recent interesting work on implementation includes Doğan and Koray (2015), Koray and Yildiz (2018), Laslier et al. (2021), Núñez and Sanver (2021) among others.

The motivation for our paper bears some similarities with those of a strand of Nash implementation literature that analyzes "the possibility of implementing specific social choice correspondences by means of mechanisms that avoid the complexities of canonical mechanisms." (Dutta et al. 1995) Toward that regard, these papers employ simplicity notions suited for the environment they restrict their attention to, such as pure exchange economies, public good environments. Meanwhile, Saijo (1988) and McKelvey (1989) consider the attainability of Nash implementation with minimal message spaces in general social choice environments. Building upon Hurwicz (1960, 1972), and Mount and Reiter (1974), the literature on realizations, message processes, and communication protocols contains Hurwicz et al. (1980), Saari (1984), Williams (1986), and Reichelstein and Reiter (1988). See Hurwicz and Reiter (2006) for more on this subject. On the other hand, related work analyzing the communication complexity include Mount and Reiter (1996), Mount and Reiter (2002), Nisan and Segal (2006) and Segal (2007, 2010). Moreover, a recent strand in the mechanism design literature focuses on other notions of simplicity of mechanisms, see, e.g., Li (2017), Börgers and Li (2019), and Pycia and Troyan (2019).

6 Concluding remarks

In this paper, we delineate the scope of Nash implementation with the help of computational tools.

In the first part of the paper, we analyze the scope of Nash implementation for a given mechanism. Our findings are of help to a planner who is familiar with a given mechanism (possibly on account of having used it in similar instances) and desires to know its boundaries of Nash implementability. For a given mechanism, we describe all the Nash implementable SCCs along with their domains on which they are Nash implementable. The planner may also utilize our results/codes to diagnose the maximal domain of preferences under which an SCC defined on the unrestricted domain of preferences (e.g., efficiency) is Nash implementable by the mechanism the planner has in mind.

In the second part of the paper, we turn to the computational analysis of Nash implementation of a given SCC. To the best of our knowledge, our paper is the first paper that exhibits the potential benefits of computational identification of collections consistent with an SCC for purposes of Nash implementation. We portray that searching through consistent collections enables the planner to design appealing mechanisms based on some desirable criteria. We hope that our results and codes will pave the way for further computational contributions in the implementation literature.

Appendix

A Proofs

A.1 Proof of Lemma 1

By definition, $\Phi^\mu = \cup_{S\in\mathcal{X}}\Phi^\mu(S) = \cup_{S\in\mathcal{X}}(\Pi^\mu(S)\setminus\Pi^\mu(X\setminus S))$. As for all $S\in\mathcal{X}$, the subset $X\setminus S\in\mathcal{X}$, we see that $\Phi^\mu = \cup_{S\in\mathcal{X}}(\Pi^\mu(S)\setminus\Pi^\mu(X\setminus S)) = \cup_{S\in\mathcal{X}}\Pi^\mu(S)$. As for all $x\in X$, $\{x\}\in\mathcal{X}$, $\cup_{x\in X}\Pi^\mu(\{x\})\subset\cup_{S\in\mathcal{X}}\Pi^\mu(S)$. Moreover, as for any $S\in\mathcal{X}$ and for any $x\in S$, $\cap_{\tilde{x}\in S}\Pi^\mu(\{\tilde{x}\})\subset\Pi(\{x\})$, $\cup_{S\in\mathcal{X}}\Pi^\mu(S) = \cup_{S\in\mathcal{X}}(\cap_{\tilde{x}\in S}\Pi^\mu(\{\tilde{x}\}))\subset\cup_{x\in X}\Pi^\mu(\{x\})$. Thus, $\Phi^\mu = \cup_{S\in\mathcal{X}}\Pi^\mu(S) = \cup_{x\in X}\Pi^\mu(\{x\})$. □

A.2 Proof of Theorem 1

By definition, for all $S\in\mathcal{S}^\mu$, $\Phi^\mu(S)\neq\emptyset$, and $\cup_{S\in\mathcal{S}^\mu}\Phi^\mu(S) = \Phi^\mu$. Thus, all what we need to show is that for all $S,\tilde{S}\in\mathcal{S}^\mu$, $S\neq\tilde{S}$ implies $\Phi^\mu(S)\cap\Phi^\mu(\tilde{S}) = \emptyset$. For a contradiction, suppose that $\omega^*\in\Phi^\mu(S)\cap\Phi^\mu(\tilde{S})$, while $S\neq\tilde{S}$. Without loss of generality, let $x^*\in S\setminus\tilde{S}$. Since $\omega^*\in\Phi^\mu(S) = \cap_{x\in S}\Pi^\mu(\{x\})\setminus\cup_{y\notin S}\Pi^\mu(\{\tilde{y}\})$, we see that $\omega^*\in\Pi^\mu(\{x^*\})$. As $x^*\notin\tilde{S}$ and $\omega^*\in\Pi^\mu(\{x^*\})$, we observe that $\omega^*\in\cup_{\tilde{y}\notin\tilde{S}}\Pi^\mu(\{\tilde{y}\})$ and hence $\omega^*\notin\Phi^\mu(\tilde{S})$, a contradiction. □

A.3 Proof of Lemma 2

For the necessity direction, suppose that $x\in f(\theta)$ and for all $\tilde{\theta}\in\Theta$, $f(\tilde{\theta}) = S$ if and only if $\tilde{\theta}\in\Phi^\mu(S)$. Thus, $x\in f(\theta)$ implies $\theta\in\Phi^\mu(f(\theta)) = \cap_{\tilde{x}\in f(\theta)}\Pi^\mu(\{\tilde{x}\})\setminus\cup_{\tilde{y}\notin f(\theta)}\Pi^\mu(\{\tilde{y}\})\subset\cap_{\tilde{x}\in f(\theta)}\Pi^\mu(\{\tilde{x}\})\subset\Pi^\mu(\{x\})$; establishing the only if direction of (2). To establish the if direction of (2), suppose, for a contradiction, that $\theta\in\Pi^\mu(\{x\})$ and $x\notin f(\theta)$. Then, $\theta\in\cup_{\tilde{y}\notin f(\theta)}\Pi^\mu(\{\tilde{y}\})$. Thus, regardless of whether or not $\theta\in\cap_{\tilde{x}\in f(\theta)}\Pi^\mu(\{\tilde{x}\})$, $\theta\notin\Phi^\mu(f(\theta))$, contradicting with our hypothesis.

For the sufficiency direction, suppose that (2) of the lemma holds and we need to show that for all $\theta\in\Theta$, $f(\theta) = S$ if and only if $\theta\in\Phi^\mu(S)$. Now, suppose $\theta\in\Theta$ is such that $f(\theta) = S$ and $x\in f(\theta)$. Then, by the only if direction of (2), $\theta\in\Pi^\mu(\{x\})$, thus $\theta\in\cap_{\tilde{x}\in f(\theta)}\Pi^\mu(\{\tilde{x}\})$. On the other hand, if $\theta\in\cup_{\tilde{y}\notin f(\theta)}\Pi^\mu(\{\tilde{y}\})$, then there is $y\in X\setminus f(\theta)$ such that $\theta\in\Pi^\mu(\{y\})$. So by the if direction of (2), $y\in f(\theta)$, which is a contradiction. Thus, $\theta\notin\cup_{\tilde{y}\notin f(\theta)}\Pi^\mu(\{\tilde{y}\})$. Therefore, $\theta\in\Phi^\mu(f(\theta)) = \Phi^\mu(S)$. Now, if $\theta\in\Theta$ is such that, $\theta\in\Phi^\mu(S) = \cap_{\tilde{x}\in S}\Pi^\mu(\{\tilde{x}\})\setminus\cup_{\tilde{y}\notin S}\Pi^\mu(\{\tilde{y}\})$, we observe that for all $x\in S$, $\theta\in\Pi^\mu(\{x\})$; so, by the if direction of (2), $x\in f(\theta)$. Hence, $S\subset f(\theta)$. Finally, to see that $f(\theta)\subset S$, observe that for any $y\notin S$, as $\theta\in\Phi^\mu(S) = \cap_{\tilde{x}\in S}\Pi^\mu(\{\tilde{x}\})\setminus\cup_{\tilde{y}\notin S}\Pi^\mu(\{\tilde{y}\})$ implies $\theta\notin\cup_{\tilde{y}\notin S}\Pi^\mu(\{\tilde{y}\})$ and in turn $\theta\notin\Pi^\mu(\{y\})$. By the contrapositive of the only if direction of (2), $y\notin f(\theta)$. □

A.4 Proof of Theorem 3

We prove that if SCC $f : \Omega \to \mathcal{X}$ restricted to a domain $\Theta \subset \Omega$ is Nash implementable by mechanism μ, then f is Maskin monotonic on domain Θ. Suppose $\theta, \tilde{\theta} \in \Theta$ are such that $x \in f(\theta) \setminus f(\tilde{\theta})$. Then, by the only if direction of (2) of Lemma 2, $x \in f(\theta)$ implies $\theta \in \Pi^{\mu}(\{x\})$. Hence, there is m^x with $g(m^x) = x$ and $x \in \cap_{i \in N} C_i^{\theta}(O_i^{\mu}(m_{-i}^x))$; ergo, $O_i^{\mu}(m_{-i}^x) \subset L_i^{\theta}(x)$ for all $i \in N$. On the other hand, by the contrapositive of the if direction of (2), $x \notin f(\tilde{\theta})$ implies $\tilde{\theta} \notin \Pi^{\mu}(\{x\})$. Thus, $x \notin \cap_{i \in N} C_i^{\tilde{\theta}}(O_i^{\mu}(m_{-i}^x))$; so, there is $j \in N$ such that $O_j^{\mu}(m_{-j}^x) \not\subset L_j^{\tilde{\theta}}(x)$. Therefore, $O_j^{\mu}(m_{-j}^x) \subset L_j^{\theta}(x)$ and $O_j^{\mu}(m_{-j}^x) \not\subset L_j^{\tilde{\theta}}(x)$ implies $L_j^{\theta}(x) \not\subset L_j^{\tilde{\theta}}(x)$. □

A.5 Proof of Theorem 4

Let $\mu = (M, g)$ be a mechanism such that $f(\theta) = NE^{\mu}(\theta)$ for all $\theta \in \Theta$. Then, for any $x \in f(\theta)$, there is $m^x \in M$ such that $g(m^x) = x$ and $g(m^x) \in \cap_{i \in N} C_i(O_i^{\mu}(m_{-i}^x))$. Suppose, for contradiction, $x \in f(\theta)$ and [for all $i \in N$ and $y \in X$, $x R_i^{\theta} y \implies x R_i^{\theta'} y$] but $x \notin f(\theta')$. Then, as $x \notin f(\theta')$, m^x cannot be a Nash equilibrium at θ'. This means there is $j \in N$ such that $x \notin C_j^{\theta'}(O_j^{\mu}(m_{-j}))$. Therefore, there is $y \in C_j^{\theta'}(O_j^{\mu}(m_{-j}))$ such that $y P_j^{\theta'} x$. On the other hand, $x \in C_j^{\theta}(O_j^{\mu}(m_{-j}))$ and hence $x R_j^{\theta} y$. But, this is a contradiction since $x R_j^{\theta} y$ but $\neg x R_j^{\theta'} y$. □

A.6 Proof of Theorem 5

Let $\mu = (M, g)$ be a mechanism such that $f(\theta) = NE^{\mu}(\theta)$ for all $\theta \in \Theta$. Then, for any $x \in f(\theta)$, there is $m^x \in M$ such that $g(m^x) = x$ and $g(m^x) \in \cap_{i \in N} C_i(O_i^{\mu}(m_{-i}^x))$. If $x \in f(\theta) \setminus f(\theta')$, then $g(m^x)$ is not a Nash equilibrium outcome at θ'. Then, there is $j \in N$ such that $g(m^x) \notin C_j(O_j^{\mu}(m_{-j}^x))$. For all $i \in N$, $\theta \in \Theta$, $x \in f(\theta)$, setting $S_i(x, \theta) := O_i^{\mu}(m_{-i}^x)$, we obtain (i) and (ii) of Definition 3. Therefore, $\mathbb{S} := \{S_i(x, \theta) \mid i \in N, \theta \in \Theta, x \in f(\theta)\}$ is a consistent collection with f.

□

A.7 Proof of Lemma 3

($\Rightarrow$) Let $f : \Theta \to \mathcal{X}$ be Maskin-monotonic. Construct the collection of sets $\mathbb{S} := \{S_i(x, \theta) \mid i \in N, \theta \in \Theta, x \in f(\theta)\}$ such that $S_i(x, \theta) := L_i^{\theta}(x) = \{y \in X \mid x R_i^{\theta} y\}$, i.e., $S_i(x, \theta)$ is the lower contour set of agent i at state θ of alternative x. Observe that (i) of consistency (Definition 3) immediately follows as for any $x \in f(\theta), x \in C_i^{\theta}(L_i^{\theta}(x))$ for all $i \in N$. If $x \in f(\theta) \setminus f(\theta')$, then it follows from the definition of Maskin-monotonicity that there exists $j \in N$ and $y \in X$ such that $x R_i^{\theta} y$ but $y P_i^{\theta'} x$. This means $x \notin C_j^{\theta'}(L_i^{\theta}(x))$, i.e., $x \notin C_j^{\theta'}(S_j(x, \theta))$ implying (ii) of consistency.

($\Leftarrow$) Let $\mathbb{S} := \{S_i(x, \theta) \mid i \in N, \theta \in \Theta, x \in f(\theta)\}$ be a collection of sets that is consistent with $f : \Theta \to \mathcal{X}$. Let $x \in X$ and $\theta, \theta' \in \Theta$, suppose, for contradiction,

$x \in f(\theta)$ and [for all $i \in N$ and $y \in X$, $xR_i^\theta y \implies xR_i^{\theta'} y$] but $x \notin f(\theta')$. Then, as $x \in f(\theta) \setminus f(\theta')$, by (ii) of consistency, there is $j \in N$ such that $x \notin C_j^{\theta'}(S_j(x,\theta))$. Therefore, there is $y \in S_j(x,\theta)$ such that $yP_j^{\theta'} x$. On the other hand, by (i) of consistency, $x \in C_j^\theta(S_j(x,\theta))$ and hence $xR_j^\theta y$. But, this is a contradiction since $xR_j^\theta y$ but $\neg xR_j^{\theta'} y$. □

A.8 Proof of Theorem 6

Let $\mu = (M, g)$ be a mechanism with $f(\theta) = NE^\mu(\theta)$ for all $\theta \in \Theta$. For all $i \in N$, $\theta \in \Theta$, $x \in f(\theta)$, set $S_i(x,\theta) := O_i^\mu(m_{-i}^x)$. Then, (i) and (ii) of Definition 4 follow directly as in the proof of Theorem 5. Let $x \in f(\theta)$ and $y \in f(\theta')$. Then, there exist $m^x \in M$ and $m^y \in M$ such that $g(m^x) = x \in C_1^\theta(S_1(x,\theta)) \cap C_2^\theta(S_2(x,\theta))$ and $g(m^y) = y \in C_1^{\theta'}(S_1(y,\theta')) \cap C_2^{\theta'}(S_2(y,\theta'))$. Since $S_1(x,\theta) = O_1^\mu(m_2^x) = \{g(m_1, m_2^x) \mid m_1 \in M_1\}$ and $S_2(y,\theta') = O_2^\mu(m_1^y) = \{g(m_1^y, m_2) \mid m_2 \in M_2\}$, we have $g(m_1^y, m_2^x) \in O_1^\mu(m_2^x) \cap O_2^\mu(m_1^y) = S_1(x,\theta) \cap S_2(y,\theta')$, i.e., $S_1(x,\theta) \cap S_2(y,\theta') \neq \emptyset$. □

A.9 Proof of Theorem 7

The proof is presented as a discussion in the text right after Theorem 7. □

A.10 Proof of Theorem 8

Let $\mathbb{S} := \{S_i(x,\theta) \mid i \in N, \theta \in \Theta, x \in f(\theta)\}$ be a consistent collection of sets associated with $f : \Theta \to \mathcal{X}$ that satisfies either NVP* or EE* and $\bar{X} \subset X$ be the corresponding set with $\cup_{S \in \mathbb{S}} \subset \bar{X}$. The proof employs the *canonical mechanism* $\mu = (M, g)$ constructed employing the consistent collection $\mathbb{S}$: Let $M_i := \Theta \times \bar{X} \times \mathbb{N}$ with $m_i = (\theta^{(i)}, x^{(i)}, k^{(i)}) \in M_i$. The outcome function $g : M \to X$ is given by

$$
\begin{aligned}
&\textbf{Rule 1}: g(m) = x && \text{if } m_i = (\theta, x, \cdot) \text{ for all } i \in N \\
& && \text{with } x \in f(\theta), \\
&\textbf{Rule 2}: g(m) = \begin{cases} x' \text{ if } x' \in S_j(x,\theta) \\ x \ \text{ otherwise.} \end{cases} && \text{if } m_i = (\theta, x, \cdot) \text{ for all } i \in N \setminus \{j\} \\
& && \text{with } x \in f(\theta), \text{ and} \\
& && m_j = (\theta', x', \cdot) \neq (\theta, x, \cdot), \\
&\textbf{Rule 3}: g(m) = x^{(i^*)} \text{ where} && \text{otherwise.} \\
&\qquad i^* = \min\{j \in N : k^{(j)} \geq \max_{i' \in N} k^{(i')}\}
\end{aligned}
$$

For any $\theta \in \Theta$ and $x \in f(\theta)$, consider $m^* \in M$ with $m_i^* = (\theta, x, 1)$ for all $i \in N$. Then, by Rule 1, $g(m^*) = x$. Given Rule 1 and Rule 2, the corresponding opportunity set of each $i \in N$ is $S_i(x,\theta)$, i.e., $O_i^\mu(m_{-i}^*) = S_i(x,\theta)$. Then, by (i) of consistency (Definition 3), $g(m^*) \in \cap_{i \in N} C_i^\theta(O_i^\mu(m_{-i}^*))$. That is, $g(m^*) = x$ is a Nash equilibrium outcome at θ. Therefore, $f(\theta) \subset NE^\mu(\theta)$ for each $\theta \in \Theta$.

To see that $NE^\mu(\theta) \subset f(\theta)$ for each $\theta \in \Theta$, observe that (i) when $\mathbb{S}$ satisfies EE*, all Nash equilibria arise under Rule 1; (ii) if $\mathbb{S}$ satisfies NVP*, then whenever there is a Nash equilibrium at some $\theta \in \Theta$ under Rule 2 or Rule 3, the corresponding Nash equilibrium outcome must be f-optimal at θ since the opportunity set of at least $n-1$ individuals then is $\bar{X}$. Therefore, it is enough to check whether Nash equilibrium outcomes that arise under Rule 1 are f-optimal. Let $\tilde{m} \in M$ be a Nash equilibrium of μ at some θ such that Rule 1 holds. Then, for all $i \in N, \tilde{m}_i = (\tilde{\theta}, \tilde{x}, \cdot)$ for some $\tilde{\theta} \in \Theta$ and $\tilde{x} \in f(\tilde{\theta})$. Then, by Rule 1, $g(\tilde{m}) = \tilde{x}$. Suppose, for contradiction, $\tilde{x} \notin f(\theta)$ so that we have $\tilde{x} \in f(\tilde{\theta}) \setminus f(\theta)$. Then, by (ii) of consistency (Definition 3), there is $j \in N$ such that $\tilde{x} \notin C_j^{\theta}(S_j(\tilde{x}, \tilde{\theta}))$. Since, by construction, $O_j^{\mu}(\tilde{m}_{-j}) = S_j(\tilde{x}, \tilde{\theta})$, this implies $g(\tilde{m}) \notin C_j^{\theta}(O_j^{\mu}(\tilde{m}_{-j}))$, a contradiction to $\tilde{m}$ being a Nash equilibirum at θ. □

B Simplicity versus communication complexity

Nisan and Segal (2006), Segal (2007, 2010) address the problem of communication in economic mechanisms and analyze nondeterministic communication protocols *realizing* a social choice rule:

A *nondeterministic communication protocol* Segal (2010) is a triple $\Gamma = \langle L, \nu, h \rangle$ where L is the *(joint) message space*, $\nu : \Theta \twoheadrightarrow L$ is the *message correspondence* satisfying *Privacy Preservation*: $\nu(\theta) = \cap_{i \in N} \nu_i(\theta_i)$ for all $\theta \in \Theta$, where $\nu_i : \Theta_i \twoheadrightarrow L$ for all $i \in N$, and $h : L \to X$ is the *outcome function of the communication protocol*. We say that a communication protocol Γ *realizes the SCC* $f : \Theta \to \mathcal{X}$ if $\emptyset \neq h(\nu(\theta)) \subset f(\theta)$ for all $\theta \in \Theta$; *fully realizes the SCC* $f : \Theta \to \mathcal{X}$ if $h(\nu(\theta)) = f(\theta)$ for all $\theta \in \Theta$. Moreover, a (joint) message $\ell \in L$ in protocol Γ *verifies alternative* $x \in X$ in the SCC f if $\nu^{-1}(\ell) \subset f^{-1}(x)$, i.e., $x \notin f(\theta)$ implies there is $j \in N$ such that $\ell \notin \nu_j(\theta_j)$.

The communication complexity of Segal (2007, 2010) is based on the following: A (joint) message $\tilde{\ell}$ in protocol $\tilde{\Gamma} = \langle \tilde{L}, \tilde{\nu}, \tilde{h} \rangle$ is *less informative* than message ℓ in protocol $\Gamma = \langle L, \nu, h \rangle$ if $\nu^{-1}(\ell) \subset \tilde{\nu}^{-1}(\tilde{\ell})$; messages ℓ and $\tilde{\ell}$ are *equivalent* if they are equally informative, i.e., $\nu^{-1}(\ell) = \tilde{\nu}^{-1}(\tilde{\ell})$. Moreover, message ℓ is a *minimally informative message verifying alternative* $x \in X$ if it verifies x, and any less informative message verifying x is equivalent to ℓ. We extend the notion of informative messages to communication protocols by: For any pair Γ and $\tilde{\Gamma}$ fully realizing a given SCC f, we say that Γ is a *more informative communication protocol* than $\tilde{\Gamma}$ if for all $\ell \in L$ and $\tilde{\ell} \in \tilde{L}$ both minimally verifying an alternative $x \in X$, $\tilde{\ell}$ is less informative than ℓ.

Next, we provide an example that showcases the following: The implications of our simplicity on mechanisms that implement an SCF is not in line with the implications of using the communication complexity of Segal (2007, 2010) to evaluate the communication protocols (associated with these mechanisms that also satisfy the needed verification properties) concerning the same SCF.

Let $N = \{1, 2\}$; $X = \{a, b, c, d, e\}$; $\Theta_1 = \Theta_2 = \{\theta_H, \theta_L\}$; $\Theta = \Theta_1 \times \Theta_2$. The SCF $f : \Theta \to X$ is given by $f(\theta_H, \theta_H) = a$, $f(\theta_H, \theta_L) = b$, $f(\theta_L, \theta_H) = c$, and $f(\theta_L, \theta_L) = d$. We consider two communication protocols $\Gamma = \langle L, \nu, h \rangle$ and $\tilde{\Gamma} = \langle \tilde{L}, \tilde{\nu}, \tilde{h} \rangle$. Γ is such that $L = \{I, II, III, IV\}^2$; $\nu_1(\theta_H) = \{(I, I), (II, II)\}$, $\nu_1(\theta_L) = \{(III, III), (IV, IV)\}$, $\nu_2(\theta_H) = \{(I, I), (III, III)\}$,

Table 13 Mechanisms associated with the communication protocols

Ind. 2					
		I	II	III	IV
Ind. 1	I	a	e	e	e
	II	e	b	e	e
	III	e	e	c	e
	IV	e	e	e	d

Mechanism μ associated with Γ

Ind. 2			
		α	β
Ind. 1	α	a	b
	β	c	d

Mechanism $\tilde{\mu}$ associated with $\tilde{\Gamma}$

and $\nu_2(\theta_L) = \{(II, II), (IV, IV)\}$. Then, $\nu(\theta_H, \theta_H) = \nu_1(\theta_H) \cap \nu_2(\theta_H) = \{(I, I), (II, II)\} \cap \{(I, I), (III, III)\} = \{(I, I)\}$, (and by repeating this argument, we observe that) $\nu(\theta_H, \theta_L) = \{(II, II)\}$, $\nu(\theta_L, \theta_H) = \{(III, III)\}$, and $\nu(\theta_L, \theta_L) = \{(IV, IV)\}$. The outcome function h is such that $h(I, I) = a$, $h(II, II) = b$, $h(III, III) = c$, $h(IV, IV) = d$, and $h(\ell) = e$ for all other $\ell \in L$. Meanwhile, $\tilde{\Gamma}$ is defined by $\tilde{L} = \{\alpha, \beta\}^2$; $\tilde{\nu}_1(\theta_H) = \{(\alpha, \alpha), (\alpha, \beta)\}$, $\tilde{\nu}_1(\theta_L) = \{(\beta, \alpha), (\beta, \beta)\}$, $\tilde{\nu}_2(\theta_H) = \{(\alpha, \alpha), (\beta, \alpha)\}$, and $\tilde{\nu}_2(\theta_L) = \{(\alpha, \beta), (\beta, \beta)\}$. Now, $\tilde{\nu}(\theta_H, \theta_H) = \{(\alpha, \alpha)\}$, $\tilde{\nu}(\theta_H, \theta_L) = \{(\alpha, \beta)\}$, $\tilde{\nu}(\theta_L, \theta_H) = \{(\beta, \alpha)\}$, and $\tilde{\nu}(\theta_L, \theta_L) = \{(\beta, \beta)\}$. The outcome function $\tilde{h}$ is such that $\tilde{h}(\alpha, \alpha) = a$, $\tilde{h}(\alpha, \beta) = b$, $\tilde{h}(\beta, \alpha) = c$, $\tilde{h}(\beta, \beta) = d$.

Notice that $\nu^{-1}(I, I) = (\theta_H, \theta_H)$, $\nu^{-1}(II, II) = (\theta_H, \theta_L)$, $\nu^{-1}(III, III) = (\theta_L, \theta_H)$, and $\nu^{-1}(IV, IV) = (\theta_L, \theta_L)$, while $\nu^{-1}(\ell) = \emptyset$ for all other $\ell \in L$; $\tilde{\nu}^{-1}(\alpha, \alpha) = (\theta_H, \theta_H)$, $\tilde{\nu}^{-1}(\alpha, \beta) = (\theta_H, \theta_L)$, $\tilde{\nu}^{-1}(\beta, \alpha) = (\theta_L, \theta_H)$, and $\tilde{\nu}^{-1}(\beta, \beta) = (\theta_L, \theta_L)$. So, $\nu^{-1}(I, I) = \tilde{\nu}^{-1}(\alpha, \alpha)$, $\nu^{-1}(II, II) = \tilde{\nu}^{-1}(\alpha, \beta)$, $\nu^{-1}(III, III) = \tilde{\nu}^{-1}(\beta, \alpha)$, $\nu^{-1}(IV, IV) = \tilde{\nu}^{-1}(\beta, \beta)$ while $\nu^{-1}(\ell) = \emptyset$ for all other $\ell \in L$. We also observe that Γ and $\tilde{\Gamma}$ both fully realizes the SCC f; (I, I) in Γ and (α, α) in $\tilde{\Gamma}$ are the minimally informative messages verifying alternative $f(\theta_H, \theta_H) = a$, (II, II) in Γ and (α, β) in $\tilde{\Gamma}$ are the minimally informative messages verifying $f(\theta_H, \theta_L) = b$, (III, III) in Γ and (β, α) in $\tilde{\Gamma}$ are the minimally informative messages verifying $f(\theta_L, \theta_H) = c$, and (IV, IV) in Γ and (β, β) in $\tilde{\Gamma}$ are the minimally informative messages verifying $f(\theta_L, \theta_L) = d$.

Therefore, $\Gamma = \langle L, \nu, h\rangle$ and $\tilde{\Gamma} = \langle \tilde{L}, \tilde{\nu}, \tilde{h}\rangle$ are equally informative communication protocols as Γ is a more informative communication protocol than $\tilde{\Gamma}$ and $\tilde{\Gamma}$ is a more informative communication protocol than Γ.

The mechanisms in Table 13 can be associated with the communication protocols Γ and $\tilde{\Gamma}$.

On the other hand, according to our definition of simplicity of mechanisms, $\tilde{\mu}$, the mechanism associated with $\tilde{\Gamma}$, is simpler than μ, the mechanism associated with Γ. Ergo, even though our simplicity notion is inherently similar in spirit with the communication complexity of Segal (2007, 2010), they do not produce perfectly aligned implications.

References

Altun OA, Barlo M, Dalkiran NA (2021) Implementation with a sympathizer. Sabanci University, Istanbul

Barlo M, Dalkiran NA (2020) Behavioral implementation under incomplete information. Sabanci University, Istanbul

Barlo M, Dalkiran NA (2021) Implementation with missing data. Sabanci University, Istanbul

Börgers T, Li J (2019) Strategically simple mechanisms. Econometrica 87(6):2003–2035

Dash RK, Jennings NR, Parkes DC (2003) Computational-mechanism design: a call to arms. IEEE Intell Syst 18(6):40–47

de Clippel G (2014) Behavioral implementation. Am Econ Rev 104(10):2975–3002

Doğan B, Koray S (2015) Maskin-monotonic scoring rules. Soc Choice Welf 44(2):423–432

Dutta B, Sen A (1991) A necessary and sufficient condition for two-person nash implementation. Rev Econ Stud 58(1):121–128

Dutta B, Sen A, Vohra R (1995) Nash implementation through elementary mechanisms in economic environments. Econ Des 1(1):173–203

Hayashi T, Jain R, Korpela V, Lombardi M (2020) Behavioral strong implementation. Available at SSRN 3657095

Hurwicz L (1960) Optimality and informational effciency in resource allocation processes. In: Arrow JK, Karlin S, Suppes P (eds) Mathematical methods in the social Sciences, Stanford University Press

Hurwicz L (1972) On informationally decentralized systems. In: Radner R, McGuire CB (eds) Decision and organization, North-Holland

Hurwicz L, Reiter S (2006) Designing economic mechanisms. Cambridge University Press, Cambridge

Hurwicz L, Reiter S, Saari D (1980) On constructing an informationally efficient decentralized process implementing a given performance function (Technical report). Discussion paper

Jackson MO (2001) A crash course in implementation theory. Soc Choice Welf 18(4):655–708

Koray S, Yildiz K (2018) Implementation via rights structures. J Econ Theory 176:479–502

Korpela V (2012) Implementation without rationality assumptions. Theor Decis 72(2):189–203

Laslier J-F, Nunez M, Sanver MR (2021) A solution to the two-person implementation problem. J Econ Theory 194:105261

Li S (2017) Obviously strategy-proof mechanisms. Am Econ Rev 107(11):3257–87

Maskin E (1999) Nash equilibrium and welfare optimality. Rev Econ Stud 66(1):23–38

Maskin E, Sjöström T (2002) Implementation theory. Handb Soc Choice Welf 1:237–288

McKelvey RD (1989) Game forms for nash implementation of general social choice correspondences. Soc Choice Welf 6(2):139–156

Moore J, Repullo R (1990) Nash implementation: a full characterization. Econometrica 58:1083–1099

Mount K, Reiter S (1974) The informational size of message spaces. J Econ Theory 8(2):161–192

Mount K, Reiter S (1996) A lower bound on computational complexity given by revelation mechanisms. Econ Theory 7(2):237–266

Mount K, Reiter S (2002) Computation and complexity in economic behavior and organization. Cambridge University Press, Cambridge

Nisan N, Segal I (2006) The communication requirements of efficient allocations and supporting prices. J Econ Theory 129(1):192–224

Núñez M, Sanver MR (2021) On the subgame perfect implementability of voting rules. Soc Choice Welf 56(2):421–441

Palfrey TR (2002) Implementation theory. Handbook of game theory with economic applications, vol 3. Elsevier, Amsterdam, pp 2271–2326

Pycia M, Troyan P (2019) A theory of simplicity in games and mechanism design

Reichelstein S, Reiter S (1988) Game forms with minimal message spaces. Econometrica 661–692

Saari DG (1984) A method for constructing message systems for smooth performance functions. J Econ Theory 33(2):249–274

Saijo T (1988) Strategy space reduction in Maskin's theorem: sufficient conditions for Nash implementation. Econometrica 56:693–700

Sanver MR (2008) Nash implementability of the plurality rule over restricted domains. Econ Lett 99(2):298–300

Sanver MR (2017) Nash implementing social choice rules with restricted ranges. Rev Econ Des 21(1):65–72

Segal I (2007) The communication requirements of social choice rules and supporting budget sets. J Econ Theory 136(1):341–378

Segal I (2010) Nash implementation with little communication. Theor Econ 5(1):51–71
Serrano R (2004) The theory of implementation of social choice rules. SIAM Rev 46(3):377–414
Williams SR (1986) Realization and nash implementation: two aspects of mechanism design. Econometrica 54:139–151