| | $k = 1$ | $k \geq 2$ |
|---|---|---|
| $z \leq 2$ | Equivalent notions according to Lemma 4.2 | |
| $z \geq 3$ | and Lemma 4.3 | Counterexamples exist (Proposition 4.4) |

**Table 2: Summary of Section 4.1 on the equivalence between assignments in symmetric PGS and factors in potential graphs. If a symmetric PGS instance admits a $k$-reviewable, $z$-credible assignment, then its potential graph admits a $2k$-factor with no cycles of length smaller than $z$ (Lemma 4.1). The converse is only guaranteed for some values of $k$ and $z$, as indicated above.**

$k + k'$ and an outdegree equals to $k$; and $A$ is $z$-credible if $G'$ has no cycles of length smaller than $z$.

# 4 ALGORITHMIC ANALYSIS

Let us first mention that while our model is faithful to the diversity of PGS situations (see Table 3), the parameter $k'$ has little computational impact in theory. Hence, unless stated otherwise, we assume in this section that there are no consumers in the review process ($C = \emptyset, k' = 0$) and we use the notations PGS($k$, $z$, $\epsilon$, *veto*) and $k$-reviewable.

We start our algorithmic analysis by examining the connections between finding reviewable credible assignments in symmetric PGS instances and the graph-theoretic problem of finding $r$-factors with no short cycles. The correspondence is summarized in Table 2 and will let us transfer some complexity results in graph theory to the symmetric PGS problem.

## 4.1 Correspondence between PGS peer review selection and $r$-factors

PGS peer review selection is closely related to the graph-theoretic problem of finding $r$-factors, when consumers do not participate in the review process ($C = \emptyset$, $k' = 0$), when no skills are required ($\epsilon = 0$), and when $V$ is symmetric. Lemmas 4.1, 4.2, and 4.3 present the condition under which the two problems are equivalent.

Lemma 4.1. *For any $k$ and $z$, an instance of PGS($k$, $z$, 0, symmetric) has a solution* if *the potential review graph has a $2k$-factor that does not include cycles of length smaller than $z$.*

Proof. Assume that the potential review graph admits a $2k$-factor, denoted $(P, E_0)$, which does not include cycles of length smaller than $z$. Algorithm 1 computes an assignment $A$ for PGS($k$, $z$, 0, *symmetric*). Intuitively, Algorithm 1 turns cycles from $E_0$ into directed circuits and adds them to $A$. Notice that it satisfies the following loop invariants.

- $E_0 = E \cup \{\{x, y\} \mid (x, y) \in A\}$.
- Each producer $p$ provides as many reviews as it receives in $A$: $|\{x \mid (x, p) \in A\}| = |\{y \mid (p, y) \in A\}|$.
- For each producer $p$, edges are conserved going from $E$ to $A$, that is

$$2k = \deg_{(P,E)}(p) + |\{x \mid (x, p) \in A\}| + |\{y \mid (p, y) \in A\}|.$$

**Algorithm 1:** $2k$-factor to Assignment ($E_0$)

```
E ← E_0
A ← ∅
while E ≠ ∅ do
    select an arbitrary cycle C from E
    orientate it to obtain a directed circuit C′
    E ← E \ C
    A ← A ∪ C′
return A
```

At the end of the loop, since $E$ is empty, we derive from the loop invariants that $A$ is $V$-respecting, $k$-reviewable, and $z$-credible. □

Lemma 4.2. *For any $k$, if an instance of PGS($k$, 2, 0, symmetric) admits a solution then its potential review graph has a $2k$-factor.*

Proof. Assume that $G$ admits a solution for PGS($k$, 2, 0, symmetric), described as a directed subgraph $G' = (P, A)$. In $G'$, since each producer provides and receives $k$ reviews, the degree of each node is $2k$. Hence, by removing the orientation of the arrows in $A$, graph $G'$ is a $2k$-factor for $G$. □

Lemma 4.3. *For any $z$, if an instance of PGS(1, $z$, 0, symmetric) admits a solution then its potential review graph has a 2-factor that does not include cycles of length smaller than $z$.*

Proof. Assume that $G$ admits a solution for PGS(1, $z$, 0, symmetric), described as a directed subgraph $G' = (P, A)$. It implies that $A$ is a partition of $P$ into disjoint oriented cycles that does not contain any cycle of length smaller than $z$. Hence, by removing the orientation of arrows in $A$, graph $G'$ is a $2k$-factor for $G$ that does not contain any cycle of length smaller than $z$. □

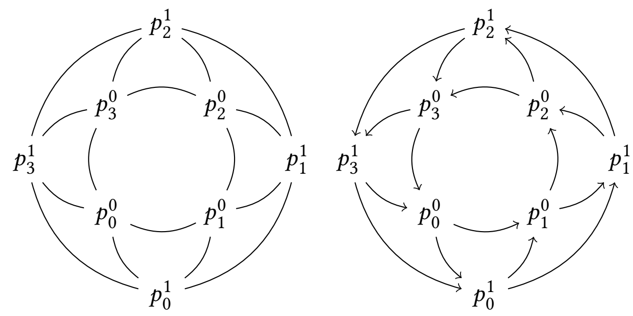
Given an instance of PGS($k$, $z$, 0, *symmetric*), Lemma 4.1 states that a $2k$-factor in its potential review graph provides a solution for this instance. Conversely, Lemma 4.2 and 4.3 show that, when $k = 1$ or $z = 2$, a solution for a PGS($k$, $z$, 0, *symmetric*) instance implies a $2k$-factor without cycle of size lower than $z$ in its potential review graph. Therefore, when either $k = 1$ or $z = 2$, PGS($k$, $z$, 0, *symmetric*) is equivalent to finding a $2k$-factor in the potential review graph. However, Proposition 4.4 shows that this relation does not extend to arbitrary $k$ and $z$.

Proposition 4.4. *For any $k \geq 2$ and $z \geq 3$, there exists a PGS($k$, $z$, 0, symmetric) instance that admits a $k$-reviewable, $z$-credible, and $V$-respecting assignment while any $2k$-factors of the potential review graph includes cycles of length $c$ for every $3 \leq c \leq z$.*

Proof. Given $k \geq 2$ and $z \geq 3$, we define the potential review graph $G = (P, E)$, with $P = \{p_i^j \mid 0 \leq i \leq z, 0 \leq j < k\}$ and

$$\begin{aligned} E =& \{\{p_i^j, p_i^l\} \mid 0 \leq i \leq z, 0 \leq j < k-1, j < l \leq k\} \\ & \cup \{\{p_i^j, p_{i+1}^l\} \mid 0 \leq i < z, 0 \leq j < k, 0 \leq l \leq j\} \\ & \cup \{\{p_z^j, p_0^l\} \mid 0 \leq j < k, 0 \leq l \leq j\}. \end{aligned}$$

First note that $G$ is $2k$-regular, and thus $G$ admits a unique $2k$-factor which is $G$ itself. Moreover, given $c$ such that $3 \leq c \leq z$, the set of edges $\{\{p_i^0, p_{i+1}^0\} \mid 0 \leq i \leq \lceil \frac{c}{2} \rceil - 1\} \cup \{\{p_i^1, p_{i+1}^1\} \mid 0 \leq i \leq$



**(a) Potential review graph $G$** **(b) Assignment graph $G'$**

**Figure 1: Counterexample showing that PGS assignments that are $k$-reviewable and $z$-credible do not always entail $2k$-factors without cycles of length $z$. Prop. 4.4 gives a generic counterexample construction and we display here the $k = 2$, $z = 3$ case. On the one hand, any 4-factor of the potential review graph in Figure 1a has cycles of length 3. On the other hand, the PGS instance admits an assignment that is 3-credible, depicted in Figure 1b.**

$\lfloor \frac{c}{2} \rfloor - 1\} \cup \{\{p_0^0, p_0^1\}, \{p^0_{\lceil \frac{c}{2} \rceil - 1}, p^1_{\lfloor \frac{c}{2} \rfloor - 1}\}\}$ forms a cycle of size $c$ in $G$. Now, consider the assignment graph $G' = (P, A)$ where

$$\begin{aligned} A = &\{(p_i^j, p_i^l) \mid 0 \leq i \leq z, 0 \leq j < k-1, j < l \leq k\} \\ &\cup \{(p_i^j, p_{i+1}^l) \mid 0 \leq i < z, 0 \leq j < k, 0 \leq l \leq j\} \\ &\cup \{(p_z^j, p_0^l) \mid 0 \leq j < k, 0 \leq l \leq j\}. \end{aligned}$$

By definition of $A$, each producer provides $k$ reviews and, since $G$ is a $2k$-factor, each producer also receives $k$. Hence $A$ is $k$-reviewable. Furthermore, assignment $A$ contains only arrows $(p_i^j, p_{i'}^{j'})$ such that $i < i'$ (except for $i = z$) or such that $i = i'$ and $j > j'$. Hence, the smallest cycles in $A$ are of the form $\{(p_i^j, p_{i+1}^j) \mid 0 \leq i < z\} \cup \{(p_z^j, p_0^j)\}$, for $j = 0, \ldots k$, which are all of size $z + 1$. Hence, $A$ is $z$-credible. Finally, $A$ is trivially $V$-respecting. ◻

The proof of Proposition 4.4 involves the construction of a PGS instance generic over the parameters $k$ and $z$. This construction is illustrated by Figure 1a and 1b which respectively depicts the potential review graph and the corresponding assignment, when the values are $k = 2$ and $z = 3$.

## 4.2 Complexity Results

We start our complexity analysis by obtaining a computational upper-bound on any parameterization of PGS($k$, $k'$, $z$, $\epsilon$, *veto*).

Proposition 4.5. *For any fixed $k$, $k'$, $z$, $\epsilon$, and veto, PGS($k$, $k'$, $z$, $\epsilon$, veto) is in* NP.

Proof. Given a solution for PGS($k$, $k'$, $z$, $\epsilon$, *veto*), checking whether it is $V$-respecting, $(k, k')$-reviewable, and $E$-compatible is trivial. Furthermore, since finding the cycles of a graph is polynomial, $z$-credibility can also be checked in polynomial time. ◻

Now we identify parameterizations of PGS($k$, $z$, $\epsilon$, *veto*) and PGS($k$, $k'$, $z$, $\epsilon$, *veto*) that lead to tractable problems. We start by showing how PGS($k$, $k'$, $z$, $\epsilon$, *veto*) reduces to PGS($k$, $z$, $\epsilon$, *veto*) when no skill is required, i.e., $\epsilon = 0$.

Lemma 4.6. *Deciding PGS($k$, $k'$, $z$, 0, veto) reduces to deciding PGS($k$, $z$, 0 , veto) in polynomial time.*

Proof. Notice that a producer $p$ can receive $k'$ reviews from consumers if and only if $p$ does not veto more than $|C| - k'$ consumers. Hence, computing an assignment of the consumers such that each producer receives $k'$ reviews is polynomial. Since $\epsilon = 0$, combining this partial assignment with an assignment for PGS($k$, $z$, 0, *veto*) forms a solution for PGS($k$, $k'$, 2, 0, *symmetric*). ◻

The two first tractability results that we present rely on the equivalence between peer review selection in PGS and $r$-factors.

Theorem 4.7. *For any $k$, PGS($k$, 2, 0, symmetric) is in* P.

Proof. Given a PGS instance $P$, $V$ and a producer review target $k$, we can compute in polynomial time whether its potential review graph admits a $2k$-factor [27]. We can then invoke Lemmas 4.1 and 4.2 to conclude. ◻

Theorem 4.8. *PGS(1, 3, 0, symmetric) is in* P.

Proof. Given a PGS instance $P$ and $V$, we can compute in polynomial time whether its potential review graph admits a 2-factor that does not contain cycles of size smaller than 3 [15]. We can then invoke Lemmas 4.1 and 4.3 to conclude. ◻

Both results extend to the case where consumers do participate in the review process, as the following theorem shows.

Theorem 4.9. *For any $k$ and $k'$, PGS($k$, $k'$, 2, 0, symmetric) and PGS(1, $k'$, 3,0, symmetric) are in* P.

Proof. Given a PGS instance $S = (P, C)$, $V$, and review targets $k$ and $k'$, we first reduce PGS($k$, $k'$, 2, 0, *symmetric*) and PGS(1, $k'$, 3, 0, *symmetric*) to PGS($k$,2, 0, *symmetric*) and PGS(1, 3, 0, *symmetric*), respectively, in polynomial time by Lemma 4.6. Then, computing an assignment for PGS($k$, 2, 0, *symmetric*) or PGS(1, 3, 0, *symmetric*) is polynomial by Theorems 4.7 and 4.8. ◻

Theorems 4.7, 4.8, and 4.9 also give rise to a polynomial time algorithms when considering a weighted model of PGS with an utilitarian score (see future works in Section 6). Indeed, Meijer et al. [27]'s algorithm produces a $r$-factor by computing a perfect matching in a modified graph. This construction can be adapted to find a maximum weight perfect matching, which is known to be polynomial.

Next result identifies a tractable case for PGS($k$, $z$, $s$, *symmetric*) when some skills are required at each reviewing visit, which doesn't easily extend to the presence of consummers.

Proposition 4.10. *For any fixed $s$, PGS(1, 2, $s$, symmetric) is in* P.

Proof. When $k = 1$, each producer has to possess all the skills, otherwise he cannot provide a review. Hence, if any producer misses a skill, i.e., if there exists $i \in S$ such that $S_i \subsetneq V$, then we have a NO-instance. Otherwise, PGS(1, 2, $s$, *symmetric*) is equivalent to PGS(1, 2, 0, *symmetric*) which is in P by Theorem 4.7. ◻

Now, we show that parameterizations of PGS($k$, $z$, $\epsilon$, *veto*) lead to computationally hard problems in general. We start with the cases where no skill is required.

Theorem 4.11. *For any $z \geq 5$, PGS(1, z, 0, symmetric) is* NP-*complete.*

Proof. By Lemmas 4.1 and 4.3, PGS(1, $z$, 0, *symmetric*) is equivalent to finding a 2-factor that contains no cycle of length smaller than $z$, which is a NP-complete problem when $z \geq 5$ [15]. □

Notice that the complexity of PGS($k$, 4, 0, *symmetric*) appears difficult to decide since the complexity of finding a 2-factor which does not contain cycles of length smaller than 4 is still under investigation by the graph-theory community.

Theorem 4.12. *PGS(1, 2, 0, general) is* NP-*complete.*

Proof. A solution of an instance of PGS(1, 2, 0, *general*) partitions the vertices of the potential review graph into cycles. Hence, PGS(1, 2, 0, *general*) is equivalent to finding a partition of the vertices into hamiltonian subgraphs, which is NP-complete when cycles have to be of size greater than 3 [9]. □

By Lemma 4.6, both results extend to the presence of consumers in the review process.

Theorem 4.13. *For any $k'$ and $z \geq 5$, PGS(2, $k'$, $z$, 0, symmetric) and PGS(1, $k'$, 2, 0, general) are* NP-*complete.*

Our main results, Theorem 4.14 and 4.15, show that the most realistic PGS settings lead to computationally hard problems.

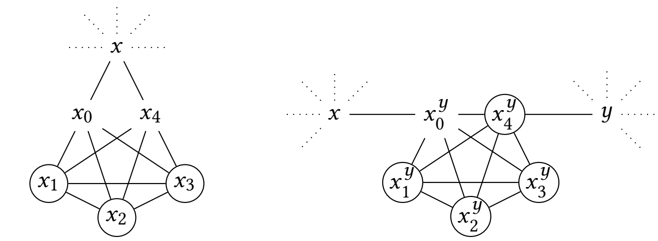
Theorem 4.14. *For any fixed number of reviews $k \geq 2$, any fixed credibility $z \geq 2$, and any fixed number of skills $\epsilon \geq 1$, PGS($k$, $z$, $\epsilon$, symmetric) is* NP-*complete.*

Proof. We give a reduction for the result in the case of $k = 2$, $z = 2$, and $\epsilon = 1$. For other cases, the reduction can be adapted by introducing dummy producers and dummy skills as needed.

We reduce PGS(1, 2, 0, *general*), which is NP-complete by Theorem 4.11, to PGS(2, 2, 1, *symmetric*) as follows. Let $(V, E)$ be the potential review graph of an instance $I$ of PGS(1, 2, 0, *general*). We create an instance $I'$ of to PGS(2, 2, 1, *symmetric*) with potential review graph $(V', E')$ and skilled producer set $S_1$, by using gadgets described in Figure 2.

To simplify notations, we assume in the following that producer indices are considered modulo 5, i.e., $x_{i+5}$ is the same producer $x_i$.

$$\begin{aligned}
V' = \ & V \cup \{x_i \mid x \in V, 0 \leq i < 5\} \\
& \cup \{x_i^y \mid (x,y) \in E, 0 \leq i < 5\} \\
S_1 = & \{x_1, x_2, x_3 \mid x \in V\} \cup \{x_1^y, x_2^y, x_3^y, x_4^y \mid (x,y) \in E\} \\
E' = \ & \{\{x, x_0\}, \{x_4, x\} \mid x \in V\} \\
& \cup \{\{x_i, x_{i+1}\} \mid x \in V, 0 \leq i < 4\} \\
& \cup \{\{x_i, x_{i+2}\} \mid x \in V, 0 \leq i < 5\} \\
& \cup \{\{x, x_0^y\}, \{x_4^y, y\} \mid (x,y) \in E\} \\
& \cup \{\{x_i^y, x_{i+1}^y\}, \{x_i^y, x_{i+2}^y\} \mid (x,y) \in E, 0 \leq i < 5\}
\end{aligned}$$




**(a) Vertex gadget representing $x$.**

**(b) Edge gadget representing the directed edge $(x, y)$.**

**Figure 2: Gadgets. A circle around a producer signifies that they are skilled.**

Intuitively, for each vertex $x \in V$, we create a vertex gadget (see Fig.2 (a)) which ensures that $x$ provides and receives one review to/from $(x_i)_{0 \leq i < 5}$. In addition, for each directed edge $(x, y)$ in $E$, we create an edge gadget (see Fig.2 (b)) which ensures that $(x_i^y)_{0 \leq i < 5}$ can only provide a skilled review to vertex $y$.

Let us first prove that if instance $I$ admits a solution, then the constructed instance $I'$ also admits a solution. Let $A \subseteq E$ be the solution assignment for $I$. Then we construct an assignment for $I'$, by selecting all edges in the vertex gadgets and most edges in the edge gadgets, and orienting them as follows.

$$\begin{aligned}
A' = \ & \{(x, x_0), (x_4, x) \mid x \in V\} \\
& \cup \{(x_i, x_{i+1}) \mid x \in V, 0 \leq i < 4\} \\
& \cup \{(x_i, x_{i+3}) \mid x \in V, 0 \leq i < 5\} \\
& \cup \{(x_i^y, x_{i+1}^y) \mid (x,y) \in E, 0 \leq i < 4\} \\
& \cup \{(x_i^y, x_{i+3}^y) \mid (x,y) \in E, 0 \leq i < 5\} \\
& \cup \{(x, x_0^y), (x_4^y, y) \mid (x,y) \in A\} \\
& \cup \{(x_4^y, x_0^y) \mid (x,y) \in E \setminus A\}
\end{aligned}$$

One can directly check that $A'$ is indeed a solution since every vertex is assigned two reviewers at least one of whom is skilled.

Let us now prove that if the constructed instance $I'$ has a solution, then instance $I$ also has a solution. Let $A' \subseteq E$ be the solution assignment for $I'$. Then we construct the assignment for $I$ by selecting edges based on which skilled reviewer reviews the original vertices in $V'$.

$$A = \{(x, y) \mid x, y \in V, (x_4^y, y) \in A'\}$$

To prove that $A$ is a satisfying assignment, let us first show that each producer $y \in V$ receives at least one review. Producer $y$ is also a producer in $I'$ and thus receives at least one skilled review in $A'$. Since the only skilled producers adjacent to $y$ in $A'$ are of the form $x_4^y$ for some $x \in V$, there exists an $x$ such that $(x_4^y, y) \in A'$. Thus there is $x$ such that $(x, y) \in A$ and so $y$ receives at least one review in $A$.

We now prove that no producer $x \in V$ participate in more than one review in $A$. Consider the producers $x_0, \ldots, x_4$. There exist only 9 edges linking them in $E'$, which is not enough for all of them to receive two reviews. Therefore, at least one review among $(x, x_0)$ and $(x, x_4)$ belongs to $A'$. As a result, for any $x \in V$, there cannot be more than one $y \in V$ such that $(x, x_4^y) \in A'$. □

Theorem 4.15. *For any fixed $k \geq 1$ and $z \geq 0$, PGS(INPUT, z, INPUT, empty) is* NP*-complete.*

Proof. We give an explicit reduction for the case PGS(INPUT, 0, INPUT, *empty*). The reduction easily adapts to other values of $z$.

We reduce from Set Cover, a classic NP-complete problem defined in Section 2.2.

Let $X, C, t$ be a Set Cover instance. Without loss of generality, we may assume that the subsets in the collection are numbered: $C = \{c_0, \ldots, c_{m-1}\}$. We construct a PGS instance as follows. The set of producers is

$$P = \{v_i, f_i \mid 0 \leq i < m\} \cup \{e_i^j \mid 0 \leq i < m, 0 \leq j < t-2\} \cup \{e\}$$

where $v_i$ are called the *subset producers*, $f_i$ are the *full producers*, and $e_i^j$ and $e$ are the *empty producers*. We create one skill per element of $X$ and define the skill sets such that all full producers have all skills, no empty producer has any skill, and a subset producer has the skills corresponding to its subset. That is, for $x \in X$, we have $S_x = \{v_i \mid x \in c_i\} \cup \{f_i \mid 0 \leq i < m\}$.

We will prove that $(P, (S_x)_{x \in X})$ admits a $(t, 0)$-reviewable and $E$-compatible assignment if and only if $X, C$ admits of cover of size $t$.

To simplify notations, we will assume that the producer indices are cyclical so that $f_{m+i}$ is the same producer as $f_i$, and $e_{m+i}^{t-2+j}$ is the same as $e_i^j$. Let $D \subseteq C$ be a subcollection of subsets, we can create an assignment as follows

$$\begin{aligned} A_D = \ & \{(f_i, f_{i+1}) \mid 0 \leq i < m\} \\ & \cup \{(f_i, v_i), (v_i, f_{i+1}) \mid 0 \leq i < m, \} \\ & \cup \{(f_i, e_i^j), (e_i^j, f_{i+1}) \mid 0 \leq i < m, 0 \leq j < t-2\} \\ & \cup \{(v_i, e_i^j), (e_i^j, v_{i+1}) \mid 0 \leq i < m, 0 \leq j < t-2\} \\ & \cup \{(e_i^j, e_{i+1}^{j+k}) \mid 0 \leq i < m, 0 \leq j, k < t-2\} \\ & \cup \{(v_i, e), (e, v_{i+1}) \mid 0 \leq i < m, c_i \in D\} \\ & \cup \{(v_i, v_{i+1}) \mid 0 \leq i < m, c_i \notin D\} \end{aligned}$$

It is straightforward to observe that if $D$ has size $t$ and covers $X$ then $A_D$ is a $(t, 0)$-reviewable and $E$-compatible assignment. Thus, if the Set Cover instance admits a solution, then so does the PGS instance.

For the other direction, observe first that there are $tm + 1$ producers and $m$ full producers. In any $(t, 0)$-reviewable assignment $A$, each full producer reviews $t$ other people, so at the very least one producer $p$ is not reviewed by a full producer. Choose one such producer $p$ arbitrarily and define $D_A = \{c_i \mid (v_i, p) \in A\}$ to be the subsets corresponding to the subset producers reviewing $p$ in $A$. Since $A$ is $(t, 0)$-reviewable, we have $|D_A| \leq t$, we can easily add other subsets to $D_A$ until it has size $t$. It is then easy to see that if $A$ is $E$-compatible, then $D_A$ is cover of $C$. Thus, if the PGS instance admits a solution, then so does the Set Cover instance. ◻

Notice that Theorem 4.14 and Theorem 4.15 can be extended to the presence of consumers by adding dummy consumers. Furthermore, with a similar reduction as the one for Theorem 4.15, we can show that PGS($k$, INPUT, $z$, INPUT, *empty*) is also NP-complete.

Theorem 4.16. *For any fixed $k \geq 2$ and $k' \geq 0$, any fixed $z \geq 2$, and any fixed $\epsilon \geq 1$, PGS(k, k′, z, ϵ, symmetric) is* NP*-complete. For any fixed $k \geq 1$ and $k' \geq 0$, and $z \geq 0$, PGS(INPUT, k′, z, INPUT, empty) and PGS(k, INPUT, z, INPUT, empty) are* NP*-complete.*

# 5 ANSWER SET PROGRAMMING MODELISATION

While some parameterizations of the PGS decision problem are tractable, they require restrictions that are not desirable for end-users. However, the membership in the class NP (Proposition 4.5) means that we can encode the problem in existing solving formalisms (such as SAT, Integer Programming, or Answer Set Programming) and invoke high-performance off-the-shelf software to solve our PGS instances.

We chose to develop an Answer Set Programming approach to solve PGS scenarios which will be made available to the PGS community.[3] In our approach, and in accordance with practice in ASP, we separate the encoding of the problem from the encoding of the instances. Specifically, we have a 3-layered approached: in the first layer, a single file (`spg.constraints.lp`) contains the logic and constraints relevant to PGS in a generic way; in the second layer, a file contains the parameterization corresponding to the rules of a specific PGS organization or country (e.g., `spg.config.india.lp`); the last layer contains the data corresponding to an instance we want to solve, i.e., the name of the stakeholders and which skills they possess as well as the vetoes between them, if any (for instance the file `spg.data.india.2019.lp`).

In Section 5.1, we describe three distinct parameterizations of our PGS model that correspond to real-world PGS organizations. The specific data we used in our experimentation is presented in Section 5.2.

We experimented with six scenarios altogether, and all our simulations can be solved within 1 second on a standard laptop machine with the Clingo Answer Set Programming solver [10]. This demonstrates that although PGS is intractable in theory, real-world instances are small enough to be addressed automatically.

## 5.1 Modeling real-world PGS organizations

The initial motivation for this model comes from real-life PGS stakeholders in three countries: Morocco, France and India. The model, which is built to easily adapt to the diversity of PGS, was tested on several instances of these three cases. In all cases, only a minimal credibility rule is applied: $z = 2$. We describe these cases in this section. See Table 3 for a summary of parameterizations.

Reviewers' assignments differ from one year to the following, both to promote knowledge exchange and to increase the credibility of the system (by decreasing the risk of collusion). Nevertheless, some groups prefer to keep an identical reviewer from one year to the following (*e.g.* a producer or a consumer in Morocco, or the

[3] The ASP encoding and the data files can be found at https://bitbucket.org/Abdallah/participatory-guarantee-systems/. Our experiments can be reproduced using the ASP solver Clingo [10].

**Table 3: Instantiation of three organizations' PGS in our mathematical model.**

| Organization | | Model | | | | |
|---|---|---|---|---|---|---|
| Name | Country | $k$ | $k'$ | $z$ | $\epsilon$ | *veto* |
| SPG Agroécologie | Morocco | 2 | 1 | 2 | 2 | general |
| Nature & Progrès | France | 1 | 1 | 2 | 1 | general |
| PGS India Organic | India | 3–5 | 0 | 2 | 1 | empty |

consumer in France) to maintain knowledge on the history and evolution of the farm.

SPG Agroécologie (Morocco) was created in 2017 and has experienced two years of certification process, in 2018 and 2019.[4] The rounds of reviews imply that each producer is evaluated by 2 producers and one consumer and has to evaluate 2 producers: $k = 2$ and $k' = 1$. The PGS takes into account two skills: $E = \{review, agroecology\}$. Valid reviews require that each skill should be possessed by at least one reviewer. All producers undertake to participate to at least two farm reviews.

Nature & Progrès (France) was created in 1964 and keeps evolving and expanding.[5] It is organized as a network of 23 relatively independent groups. Local groups account from 20 to 60 members producer members plus consumers. All groups organize reviewing in the same fashion. Each production site is reviewed by one producer and one consumer: $k = 1$ and $k' = 1$. All producers undertake to participate to at least one farm review. For now, a single skill is taken into account: $E = \{review\}$. Each producer needs to be reviewed by at least one person that is skilled in reviewing.

PGS India Organic is a governmental system, which was created in 2011.[6] In 2019, the PGS comprised 18 179 local groups organized into 326 regional councils. Local groups include from a minimum of five to several dozen members. Each farm is reviewed by three to five peer reviewers (according to the group). In the case of small groups, members from other PGS groups are invited to perform reviews. Each producer participates to at least one farm review. Reciprocal review between two producers of the same group is not allowed. A single skill is taken into account: at least one reviewer must be literate to be able to fill the evaluation report: $E = \{literate\}$.

## 5.2 Typical data Experimental evaluation of an ASP encoding

In 2018, SPG Agroécologie had 16 producers and 10 consumers, of which 11 (7 producers and 4 consumers) were skilled in reviewing and 12 producers skilled in agroecology. One member was in an unusual position that generated an important number of vetoes. In 2019, including newcomers, the PGS had 27 producers and 19 consumers. In total, taking into account new skills acquired by initial members, 16 were experienced in reviewing and 16 in agroecology. To solve this case, we added the 2018 review assignment as vetoes. All interpersonal vetoes from the previous year were lifted, except one concerning two brothers. Finally, we simulated a review round for 2020 adding the assignment of 2019.

In 2019, in the specific Nature & Progrès group (from Hérault region, South of France) we tested the model with 20 producers (of which 15 skilled in review) and 8 consumers (all skilled in review).

For PGS India Organic, we tested the model by simulating a group from South Andaman Islands of 20 members, 12 of whom are literate. We ran 3 rounds to simulate 2019, 2020, and 2021.

## 6 CONCLUSION AND FUTURE WORKS

This paper focused on peer review selection in PGS. It emerged from real-life PGS stakeholders' demand and was modeled according to authors' knowledge of such systems. We proposed a formal model encompassing the diversity of PGS local situations. While we showed that peer review selection in PGS may lead to computationally hard problems, we identified tractable cases. Finally, our encoding in ASP shows that modern solvers can handle this problem in practice.

The proposed model and its ASP implementation are now being tested with stakeholders in real PGS settings - first in February 2020 in Morocco - to ensure they are adapted to practical use. During these tests, new development needs will certainly emerge. We can already forsee future works that comprise extensions of our formal model and results on parameterized complexity.

**Dynamic model** Regular monitoring reviews, generally on a yearly basis, introduce a dynamic component to PGS. The stakeholders population can change, typically new producers may join an emerging PGS, or a PGS could split when it becomes too large to be managed locally. A dynamic model would allow us to model constraints over multiple years, e.g., ensuring that producers are not reviewed by the same producers each year.

**Weighted model** PGS are considered cheaper and less time consuming than traditional third-party certification systems. Our model can be extended by adding weights between producers which would correspond to the cost of making one producer review the other. An interesting solution would then minimize some functions of the costs that it induces.

**Few vetos** Producers usually express few vetos towards other producers. A study on parameterized complexity with respect to the number of vetos would help us identify cases that are tractable in practice.

[4] http://reseauriam.org/upload/documents/rispgmarocvdef5juin.pdf
[5] https://www.natureetprogres.org/
[6] https://www.pgsindia-ncof.gov.in/

## REFERENCES


[1] Fernando Alarcón, Guillermo Durán, and Mario Guajardo. 2014. Referee assignment in the Chilean football league using integer programming and patterns. *International Transactions in Operational Research* 21, 3 (2014), 415–438.

[2] Laurel Bellante. 2017. Building the local food movement in Chiapas, Mexico: rationales, benefits, and limitations. *Agriculture and human values* 34, 1 (2017), 119–134.

[3] Herve Bouagnimbeck, Roberto Ugas, and Jannet Villanueva. 2014. Preliminary results of the global comparative study on interactions between PGS and social processes. *Building Organic Bridges* 2 (2014), 435–438.

[4] Ioannis Caragiannis, George A Krimpas, and Alexandros A Voudouris. 2015. Aggregating partial rankings with applications to peer grading in massive online open courses. In *AAMAS*. 675–683.

[5] Laurent Charlin, Richard Zemel, and Craig Boutilier. 2011. A framework for optimizing paper matching. In *UAI*. 86–95.

[6] Don Conry, Yehuda Koren, and Naren Ramakrishnan. 2009. Recommender systems for the conference paper assignment problem. In *RecSys*. 357–360.

[7] Nancy Falchikov. 2013. *Improving assessment through student involvement: Practical solutions for aiding learning in higher and further education*. Routledge.

[8] Maria Fernanda Fonseca. 2004. Alternative certification and a network conformity assessment approach. *The Organic Standard* 38, 37 (2004), 1–7.

[9] Michael R Garey and David S Johnson. 2002. *Computers and intractability*. Vol. 29. wh freeman New York.

[10] Martin Gebser, Roland Kaminski, Benjamin Kaufmann, and Torsten Schaub. 2012. Answer set solving in practice. *Synthesis lectures on artificial intelligence and machine learning* 6, 3 (2012), 1–238.

[11] Martin Gebser, Benjamin Kaufmann, André Neumann, and Torsten Schaub. 2007. clasp: A Conflict-Driven Answer Set Solver. In *Logic Programming and Nonmonotonic Reasoning*, Chitta Baral, Gerhard Brewka, and John Schlipf (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 260–265.

[12] Martin Gebser and Torsten Schaub. 2016. Modeling and language extensions. *AI Magazine* 37, 3 (2016), 33–44.

[13] Martin Gebser, Torsten Schaub, and Sven Thiele. 2007. GrinGo: A New Grounder for Answer Set Programming. In *Logic Programming and Nonmonotonic Reasoning*, Chitta Baral, Gerhard Brewka, and John Schlipf (Eds.). Springer Berlin Heidelberg, Berlin, Heidelberg, 266–271.

[14] Judy Goldsmith and Robert H Sloan. 2007. The AI conference paper assignment problem. In *AAAI Workshop on Preference Handling for Artificial Intelligence*. 53–57.

[15] Pavol Hell, David Kirkpatrick, Jan Kratochvíl, and Igor Kříž. 1988. On restricted two-factors. *SIAM Journal on Discrete Mathematics* 1, 4 (1988), 472–484.

[16] Sonja Kaufmann and Christian R Vogl. 2018. Participatory Guarantee Systems (PGS) in Mexico: a theoretic ideal or everyday practice? *Agriculture and human values* 35, 2 (2018), 457–472.

[17] Chinmay Kulkarni, Koh Pang Wei, Huy Le, Daniel Chia, Kathryn Papadopoulos, Justin Cheng, Daphne Koller, and Scott R Klemmer. 2013. Peer and self assessment in massive online classes. *ACM Transactions on Computer-Human Interaction (TOCHI)* 20, 6 (2013), 33.

[18] David Kurokawa, Omer Lev, Jamie Morgenstern, and Ariel D. Procaccia. 2015. Impartial peer review. In *IJCAI*. 582–588.

[19] Igor Labutov and Christoph Studer. 2017. JAG: a crowdsourcing framework for joint assessment and peer grading. In *AAAI*. 1010–1016.

[20] Amina Lamghari and Jacques A Ferland. 2010. Metaheuristic methods based on Tabu search for assigning judges to competitions. *Annals of Operations Research* 180, 1 (2010), 33–61.

[21] Sylvaine Lemeilleur and Gilles Allaire. 2018. Système participatif de garantie dans les labels du mouvement de l'agriculture biologique. Une réappropriation des communs intellectuels. *Économie rurale* 365, 3 (2018), 7–27. https://doi.org/10.4000/economierurale.5813

[22] Sylvaine Lemeilleur and Gilles Allaire. 2019. *Participatory Guarantee Systems for organic farming: reclaiming the commons.* Technical Report 201902. UMR MOISA. https://ideas.repec.org/p/umr/wpaper/201902.html

[23] Xinlian Li and Toyohide Watanabe. 2013. Automatic paper-to-reviewer assignment, based on the matching degree of the reviewers. *Procedia Computer Science* 22 (2013), 633–642.

[24] Rodrigo Linfati. 2012. *Referee Assignment Problem Case: Italian Volleyball Championships.* Ph.D. Dissertation. Università di Bologna.

[25] Rodrigo Linfati, Gustavo Gatica, and J Escobar. 2019. A flexible mathematical model for the planning and designing of a sporting fixture by considering the assignment of referees. *International Journal of Industrial Engineering Computations* 10, 2 (2019), 281–294.

[26] Ngar-Fun Liu and David Carless. 2006. Peer feedback: the learning element of peer assessment. *Teaching in Higher education* 11, 3 (2006), 279–290.

[27] Henk Meijer, Yurai Núñez-Rodríguez, and David Rappaport. 2009. An algorithm for computing simple k-factors. *Inform. Process. Lett.* 109, 12 (2009), 620–625.

[28] Erin Nelson, Laura Gómez Tovar, Elodie Gueguen, Sally Humphries, Karen Landman, and Rita Schwentesius Rindermann. 2016. Participatory guarantee systems and the re-imagining of Mexico's organic sector. *Agriculture and Human Values* 33, 2 (2016), 373–388.

[29] Erin Nelson, Laura Gómez Tovar, Rita Schwentesius Rindermann, and Manuel Ángel Gómez Cruz. 2010. Participatory organic certification in Mexico: an alternative approach to maintaining the integrity of the organic label. *Agriculture and Human Values* 27, 2 (2010), 227–237.

[30] Michael D Plummer. 2007. Graph factors and factorization: 1985–2003: a survey. *Discrete Mathematics* 307, 7-8 (2007), 791–821.

[31] Simon Price and Peter A Flach. 2017. Computational support for academic peer review: a perspective from artificial intelligence. *Commun. ACM* 60, 3 (2017), 70–79.

[32] Giovanna Sacchi, Vincenzina Caputo, and Rodolfo Nayga. 2015. Alternative labeling programs and purchasing behavior toward organic foods: The case of the participatory guarantee systems in Brazil. *Sustainability* 7, 6 (2015), 7397–7416.

[33] Ivan Stelmakh, Nihar B Shah, and Aarti Singh. 2018. PeerReview4All: Fair and Accurate Reviewer Assignment in Peer Review. *preprint arXiv:1806.06237* (2018). arXiv:1806.06237

[34] Keith J Topping. 2009. Peer assessment. *Theory into practice* 48, 1 (2009), 20–27.