mathematics

MDPI

Article

A Formal Approach to Coercion Resistance and Its Application to E-Voting

Stanislas Riou [1], Oksana Kulyk [2] and David Yeregui Marcos del Blanco [3,*]

[1] ENS Rennes, 35170 Rennes, France; stanislas.riou@ens-rennes.fr
[2] Center for Information Security and Trust, IT University of Copenhagen, Rued Langgaards Vej 7 DK, 2300 Copenhagen, Denmark; okku@itu.dk
[3] Department of Mechanic Engineering, Computer and Aerospacial Sciences, University of Leon, 24071 Leon, Spain
* Correspondence: dmard@unileon.es

Abstract: The outbreak of the COVID-19 pandemic brought renewed attention to electronic voting—this time as a potential option to contain the spread during elections. One of the long unresolved topics with remote voting is the risk of voter's coercion due to the uncontrolled environment in which it takes place, indicating the importance of the coercion resistance property. In the present article, the authors conduct a database analysis of over 350 articles to present different formal definitions of coercion resistance based on three frameworks (game-based definitions, applied pi-calculus, and logic). Finally, the different security properties of each one are studied and compared in order to facilitate the development of electronic voting schemes.

Keywords: electronic voting; coercion resistance; applied pi-calculus; temporal logics

Citation: Riou, S.; Kulyk, O.; Marcos del Blanco, D.Y. A Formal Approach to Coercion Resistance and Its Application to E-Voting. *Mathematics* **2022**, *10*, 781. https://doi.org/10.3390/math10050781

Academic Editors: Luis Hernández Encinas and Víctor Gayoso Martínez

Received: 24 December 2021
Accepted: 23 February 2022
Published: 28 February 2022

1. Introduction

Since the start of the COVID-19 pandemic in 2020, discussions about online voting have become prominent due to the increased health risks of traditional, presential voting. However, despite its advantages, online voting inherently brings certain security risks. Amongst them, the fact that the voting occurs in an uncontrolled environment (e.g., their own living space), as opposed to polling booths, has traditionally been one of the major roadblocks preventing a larger-scale internet voting introduction.

In particular, the risk of voter coercion, ranging from an adversary asking for proof of a voter's choice to a coercer monitoring the voting process has been at the core within the pending challenges.

Any further deployment of e-voting schemes should address the coercion resistance property, which was first defined in [1]. Informally, a coercion-resistant scheme guarantees the voter's privacy even if an attacker can monitor the process and communicate with the voter. A number of such schemes have been proposed, relying on different assumptions about the attackers' capabilities, as well as security assurances against coercion defined through formal specifications (e.g., cryptographic definitions).

However, since no standardized definition or metric of coercion resistance exist, comparing the schemes and deciding which one of them is the most appropriate for a particular practical election scenario is rather troublesome. Therefore, a thorough understanding of the various definitions of coercion resistance is needed as a previous step before reaching meaningful conclusions.

This paper introduces the identification of different formal definitions of coercion resistance available in e-voting literature. We describe the three main groups of such definitions—*game-based*, *applied pi-calculus*, and *logic*—and discuss the differences in security properties amongst them.

2. Methods and Frameworks

In order to build our database, a search using the keywords "voting", "coercion resistance" and "formal" has been conducted in Springer, IEEE, and ACM proceedings databases (for ACM, the keyword "coercion" was used instead of "coercion resistance"). This search resulted in 372 articles. Subsequently, the preview-only content for Springer was excluded, leading to a provisional database of 258 articles. Next, only the articles providing a formal definition of coercion resistance in an e-voting context was considered, excluding those focusing only on the analysis of a scheme or on more generic properties such as security. This process resulted in a final database of 11 articles.

We organized the articles in our database according to the formal concepts used to define coercion resistance: applied pi-calculus, game-based definitions, and logic.

2.1. Applied Pi-Calculus

Applied pi-calculus [2] is a formal language used to describe concurrent processes and their interactions. The calculus consists in names (often data or channels), variables, and a signature of function symbols Σ that typically includes cryptographic primitives (encryption, decryption, hashing, etc.). Terms can be built using any valid combination of names, variables, functions symbols, and other terms. Equations are defined according to an equational theory E. The equality in this theory is denoted $=_E$. A classical example is $dec(enc(message, key), key) =_E message$, which models the correctness of symmetrical encryption.

Applied pi-calculus contains two types of processes: plain and extended. The grammar used to construct plain processes is described below: M and N are terms, n is a name, x is a variable, and u stands either for a name or a variable.

P,Q,R:= plain processes
- 0 null process
- P|Q parallel composition
- !P replication
- νn.P name restriction
- if $M = N$ then P else Q conditional
- in(u, x).P message input
- out(u, N).P message output

Extended processes are obtained by adding active substitutions and variable restrictions.

A,B,C:= extended processes
- P plain process
- A|B parallel composition
- νn.A name restriction
- νx.A variable restriction
- $\{^M/_x\}$ active substitution

The substitution $\{^M/_x\}$ replaces the variable x with the term M. We define $fv(A)$ $fn(A)$ $bv(A)$ $bn(A)$ to be respectively the free variables, free names, bounded variables, and bounded names in A. The frame $\phi(A)$ of an extended process A is obtained by replacing all plain processes in A by the null process. Frames ϕ have a domain $dom(\phi)$ defined as the set of variables for which ϕ defines a substitution. Finally, an evaluation context $C[\_]$ is an extended process with a hole for an extended process that is: not under replication, a conditional, an input, or an output.
We can state that two frames are statically equivalent as per below:

Definition 1 (Static equivalence ($\approx_s$))**.** *Two terms M and N are equal in the frame ϕ, written $(M =_E N)\phi$, if, and only if there exists $\tilde{n}$ and a substitution σ such that $\phi \equiv \nu\tilde{n}.\sigma$, $M\sigma =_E N\sigma$, and $\tilde{n} \cap (fn(M) \cup fn(N)) = \emptyset$.*

Two frames ϕ_1 and ϕ_2 are statically equivalent, $\phi_1 \approx_s \phi_2$, when $dom(\phi_1) = dom(\phi_2)$, and for all terms M,N we have $(M =_E N)\phi_1$ if and only if $(M =_E N)\phi_2$. Two extended processes are statically equivalent, denoted by $A \approx_s B$, if their frames are statically equivalent.

Intuitively, two processes are statically equivalent if the messages exchanged with the environment cannot be distinguished by an attacker. Labeled bisimilarity extends this notion.

Definition 2 (Labeled bisimilarity ($\approx_l$))**.** *Labeled bisimilarity is the largest symmetric relation $\mathcal{R}$ on closed extended processes such that $A\ \mathcal{R}\ B$ implies*

- $A \approx_s B$;
- *if $A \to A'$, then $B \to^* B'$ and $A'\ \mathcal{R}\ B'$ for some B';*
- *if $A \xrightarrow{\alpha} A'$ and $fv(\alpha) \subseteq dom(\alpha)$ and $bn(\alpha) \cap fn(B) = \emptyset$, then $B \to^* \xrightarrow{\alpha} \to^* B'$ and $A'\ \mathcal{R}\ B'$ for some B'.*

In this case, each process can simulate interactions from the other process, and the two processes are statically equivalent for each step during the execution. Therefore, an attacker cannot distinguish the two processes.

2.2. ATL*

ATL is a generalization of branching-time logic CTL* obtained by replacing path quantifiers with strategic modalities $\langle\langle A\rangle\rangle$. Intuitively, $\langle\langle A\rangle\rangle\gamma$ means that the group of agents A has a collective strategy to enforce the temporal property γ. Subsequently, ATL* [3] extends ATL with the addition of the following operators: X (referring to the next state), G (always from now on), F (eventually), and U (strong until).

Definition 3 (ATL* syntax)**.**

$$\varphi ::= p|\neg\varphi|\varphi \wedge \varphi|\langle\langle A\rangle\rangle\gamma$$
$$\gamma ::= \varphi|\neg\gamma|\gamma \wedge \gamma|X\gamma|\gamma U\gamma$$

where $A \subseteq Agt$ is a set of agents, and $p \in PV$ is a countable set of atomic propositions. The operators G and F and the weak until W are obtained from this definition: $F\gamma \equiv TU\gamma$, $\gamma_1 W\gamma_2 \equiv \neg((\neg\gamma_2)U(\neg\gamma_1 \wedge \neg\gamma_2))$, and $G\gamma \equiv \gamma W\bot$.

The semantic of this language is defined over concurrent game structures with imperfect information (iCGS).

Definition 4 (iCGS)**.** *A concurrent game structure with imperfect information is a tuple $\mathcal{G} = \langle Agt, PV, S, s_0, Act, \{\ _i\}_{i\in Agt}, d, \to, \pi\rangle$ such that*

- *Agt is a non-empty and finite set of agents. Subsets $A \subseteq Agt$ of agents are called coalitions.*
- *PV is a countable set of atomi propositions or atoms.*
- *S is a non-empty set of states and $s_0 \in S$ is the initial state of $\mathcal{G}$.*
- *Act is a finite non-empty set of actions. A tuple $\vec{a} = (a_i)_{i\in Agt} \in Act^{Agt}$ is called a joint action.*
- *For every agent $i \in Agt$, $\sim_i$ is an equivalence relation on S, which is called the indistinguishability relation for i.*
- *$d : Agt \times S \to (2^{Act}\backslash\{\emptyset\})$ is the protocol function, satisfying the property that for all states $s, s' \in S$ and any agent $i, s \sim_i s'$ implies $d(i, s) = d(i, s')$. That is, the same (non-empty) set of actions is available to agent i in uindistinguishable states.*
- *$\to \subseteq S \times Act^{Agt} \times S$ is the translation relation such that for every state $s \in S$ and joint action $\vec{a} \in Act^{Agt}$, $(s, \vec{a}, s') \in \to$ for some state $s' \in S$ if $a_i \in d(i, s)$ for every agent $i \in Agt$. We normally write $s \xrightarrow{\vec{a}} r$ for $(s, \vec{a}, r) \in \to$.*
- *$\pi : S \to 2^{PV}$ is the state-labeling function.*

In an iCGS, a strategy for a is a function $s_a : S \to Act$ such that $s_a(q) \in d(a, q)$. The set of strategies is denoted by Σ_a^{Ir}. When considering a group of agents A, a collective strategy

for the group is a tuple of strategies, with a set denoted by Σ_A^{Ir}. A path λ is an infinite sequence of states such that a transition exists between following states. $\lambda[i]$ denotes the ith state in λ and $\lambda[i, +\infty[$ is the suffix of lambda starting from the ith position. Finally, $out(q, s_A)$ returns the set of paths accessible when the strategy s_A is used from the state q by agents A. We can now provide the semantics of ATL*:

Definition 5 (semantics)**.**

$\mathcal{G}, s \models p$ *iff* $s \in \pi(p)$
$\mathcal{G}, s \models \neg\varphi$ *iff* $\mathcal{G}, s \not\models \varphi$
$\mathcal{G}, s \models \varphi_1 \wedge \varphi_2$ *iff* $\mathcal{G}, s \models \varphi_1$ *and* $\mathcal{G}, s \models \varphi_2$
$\mathcal{G}, s \models \langle\langle A\rangle\rangle\varphi$ *iff there exists* $s_A \in \Sigma_A^{Ir}$ *such that,*
for each path $\lambda \in out(q, s_A)$ *we have* $\mathcal{G}, \lambda \models \varphi$
$\mathcal{G}, \lambda \models \varphi$ *iff* $\mathcal{G}, \lambda[0] \models \varphi$
$\mathcal{G}, \lambda \models \neg\gamma$ *iff* $\mathcal{G}, \lambda \not\models \gamma$
$\mathcal{G}, \lambda \models \gamma_1 \wedge \gamma_2$ *iff* $\mathcal{G}, \lambda \models \gamma_1$ *and* $\mathcal{G}, \lambda \models \gamma_2$
$\mathcal{G}, \lambda \models X\gamma$ *iff* $\mathcal{G}, \lambda[1, +\infty] \models \gamma$
$\mathcal{G}, \lambda \models \gamma_1 U \gamma_2$ *iff there exists* $i \geq 0$ *such that* $\mathcal{G}, \lambda[i, +\infty] \models \gamma_2$ *and*
$\mathcal{G}, \lambda[j, +\infty] \models \gamma_1$ *for all* $0 \leq j \leq i$

Since coercion resistance is related to the knowledge of the adversary about the coerced voter's actions, the knowledge operator K_i has to be added. It is defined by the following semantics:

$$\mathcal{G}, s \models K_i\varphi \text{ iff, for each state } s' \in S, s' \ {}_i s \text{ implies } \mathcal{G}, s' \models \varphi$$

$K_i\varphi$ can also be expressed as $\langle\langle i\rangle\rangle\varphi U \varphi$.

3. Results

3.1. Game-Based Definitions

3.1.1. Simulation-Based Model

The first formal definition of coercion resistance was introduced by Juels, Catalano, and Jakobsson in [1]. Their definition is centered around a game between the adversary and the voter where the adversary has to guess whether the voter applied a counter-strategy or not. Before introducing this game, the model is briefly explained.

Several sets of entities are present in this model: the registrars denoted by $\mathcal{R} = \{R_1, \ldots, R_{n_R}\}$ who are responsible for issuing credentials to the voters; the talliers denoted by $\mathcal{T} = \{T_1, \ldots, T_{n_T}\}$ in charge of ballot processing, counting, and publishing of the final tally; and the voters, denoted by $\mathcal{V} = \{V_1, \ldots, V_{n_V}\}$. There is also a bulletin board $\mathcal{BB}$ where all players can write, which is used to gather the votes. Finally, the candidate slate is modeled as a list of indexes and is specified by the number of candidates alone. The election system consists of 4 protocols, each represented by a function:

- Registering: $\mathsf{register}(SK_R, i, k_1) \rightarrow (sk_i, pk_i)$, the inputs are the registrar's secret key, a voter's ID, and a security parameter. It returns a pair of keys.
- Voting: $\mathsf{vote}(sk, PK_T, n_C, \beta, k_2 \rightarrow ballot)$, the inputs are: the voter's secret key, the public key of the talliers, the number of candidates, the choice of a voter and a security parameter. It returns the ballot.
- Tallying: $\mathsf{tally}(SK_T, \mathcal{BB}, n_C, \{pk_i\}_{i=1}^{n_V}, k_3) \rightarrow (X, P)$, The inputs are the talliers' secret key, the whole bulletin board, the number of candidates, all public voting keys, and a security parameter. The outputs are the voting tally along with a non-interactive proof that the tally was correctly computed.
- Verifying: $\mathsf{verify}(PK_T, \mathcal{BB}, n_C, X, P) \rightarrow \{0, 1\}$, the inputs are the talliers' public key, the bulletin board, the number of candidates, and the results of the previous function. It returns whether the tally was correct or not.

Several assumptions are made regarding the attacker during each phase:

- Setup phase: only a minority of registrars and talliers can be corrupted by the attacker. Moreover, their secret keys are generated by a trustworthy third party.
- Prior to registration: the attacker may coerce a voter before the registration phase either to obtain a transcript of this phase or to influence the voter's interaction with the registrar.
- Registration phase: one of the following three assumptions is required to prevent simulation attacks from the attacker: either no transcripts of a voter's interaction with the registrar can be made, or the coercer cannot corrupt any registrar, or the voter is aware of the identity of any corrupt registrar.
- Voting, tallying, and verification phases: The attacker can coerce any number of voters in a static, active way. The assumption on corrupted talliers still holds. Moreover, private anonymous channels are required for the cast of ballots. Without them, it is impossible to achieve coercion resistance.

Providing the formal definition of coercion resistance requires one last function used as a counter-strategy for the voter: $\mathsf{fakekey}(PK_T, sk, pk) \rightarrow \tilde{sk}$ which returns a fake voting secret key. Finally, n_A and $n_U = n_V - n_A - 1$ are respectively the number of corrupted voters and the number of uncertain votes (non-corrupted voters other than the coerced voter concerned by the experiment), and D_{n,n_C} is a probability distribution over the possible ballot choices (including abstention and invalid ballots), which is used to represent the view of the attacker.

Once all the definitions, protocols, and assumptions have been presented, the experiment $c - resist$ can be defined as follows.

In this experiment, the attacker targets a voter who flips a coin. According to the result, the voter either uses the counter-strategy to cast her vote and provide the coercer with a fake key or gives in to the coercer and furnishes him her secret key.

$\texttt{Experiment}\ \mathsf{Exp}^{c-resist}_{ES,\mathcal{A},H}(k_1, k_2, k_3, n_V, n_A, n_C)$
$V \leftarrow \mathcal{A}(\texttt{voter names, "control voters")};$
$\{(sk_i, pk_i) \leftarrow \mathsf{register}(SK_{\mathcal{R}}, i, k_2)\}_{i=1}^{n_V};$
$(j, \beta) \leftarrow \mathcal{A}(\{sk_i\}_{i \in V}, \texttt{"set target voter and vote")};$
$\texttt{if}\ |V| \neq n_A\ \texttt{or}\ j \notin \{1, 2, \ldots, n_V\} = V\ \texttt{or}$
$\beta \notin \{1, 2, \ldots, n_C\} \cup \emptyset\ \texttt{then}$
$\texttt{output '0';}$
$b \in_U \{0, 1\};$
$\texttt{if}\ b = 0\ \texttt{then}$
$\tilde{sk} \leftarrow \mathsf{fakekey}(PK_T, sk_j, pk_j);$
$\mathcal{BB} \Longleftarrow \mathsf{vote}(sk_j, PK_T, n_C, \beta, k_2);$
$\texttt{else}$
$\tilde{sk} \leftarrow sk_j;$
$\mathcal{BB} \Longleftarrow vote(\{sk_i\}_{i \neq j, i \notin V}, PK_T, n_C, D_{n_u, n_C}, k_2);$
$\mathcal{BB} \Longleftarrow \mathcal{A}(\tilde{sk}, \mathcal{BB}, \texttt{"cast ballots")};$
$(X, P) \leftarrow \mathsf{tally}(SK_T, \mathcal{BB}, n_C, \{pk_i\}_{i=1}^{n_V}, k_3);$
$b' \leftarrow \mathcal{A}(X, P, \texttt{"guess b")};$
$\texttt{if}\ b' = b\ \texttt{then}$
$\texttt{output '1';}$
$\texttt{else}$
$\texttt{output '0';}$

In order to define coercion resistance, the attacker $\mathcal{A}$ has to be compared with another one $\mathcal{A}'$ within the framework of an ideal voting experiment. The aim of this ideal setting is that the attacker $\mathcal{A}'$ should not be able to learn anything from private keys or to cast ballots. He should only be able to obtain the total number of votes. Due to this ideal setting, the voter always gives her private key to the coercer. To achieve this goal, the tallying function has to be replaced with an ideal function. This ideal function tallies in a normal manner for the honest voters; however, it treats the votes cast by $\mathcal{A}'$ in a special way: votes cast

using a private key that does not belong to a corrupted voter are not counted, with the same happening for double votes. Finally, if the voter escaped coercion, the attacker's vote using her private key is ignored.

$\texttt{Experiment } \mathsf{Exp}^{c-resist-ideal}_{ES,\mathcal{A},H}(k_1,k_2,k_3,n_V,n_A,n_C)$
$V \leftarrow \mathcal{A}'\texttt{(voter names, "control voters");}$
$\{(sk_i,pk_i) \leftarrow \mathsf{register}(SK_{\mathcal{R}},i,k_2)\}_{i=1}^{n_V}\texttt{;}$
$(j,\beta) \leftarrow \mathcal{A}'(\{sk_i\}_{i\in V}\texttt{,"set target voter and vote");}$
$\texttt{if } |V| \neq n_A \texttt{ or } j \notin \{1,2,\ldots,n_V\} = V \texttt{ or}$
$\beta \notin \{1,2,\ldots,n_C\} \cup \emptyset \texttt{ then}$
$\texttt{output '0';}$
$b \in_U \{0,1\}\texttt{;}$
$\texttt{if } b = 0 \texttt{ then}$
$\mathcal{BB} \Longleftarrow \mathsf{vote}(sk_j,PK_T,n_C,\beta,k_2)\texttt{;}$
$\tilde{sk} \leftarrow sk_j\texttt{;}$
$\mathcal{BB} \Longleftarrow vote(\{sk_i\}_{i\neq j,i\notin V},PK_T,n_C,D_{n_u,n_C},k_2)\texttt{;}$
$\mathcal{BB} \Longleftarrow \mathcal{A}'(\tilde{sk},\mathcal{BB}\texttt{,"cast ballots");}$
$(X,P) \leftarrow \mathsf{ideal-tally}(SK_T,\mathcal{BB},n_C,\{pk_i\}_{i=1}^{n_V},k_3);$
$b' \leftarrow \mathcal{A}(X,P\texttt{,"guess b");}$
$\texttt{if } b' = b \texttt{ then}$
$\texttt{output '1';}$
$\texttt{else}$
$\texttt{output '0';}$

Using these two experiments, coercion resistance is achieved when the probability of $\mathcal{A}$ correctly guessing whether the counter-strategy was used or not is only negligibly better than the probability of $\mathcal{A}'$ succeeding. A quantity $f(k)$ is negligible in k if for all $c > 0$ there exists l_c such that $f(k) < k^{-c}$ for $k > l_c$.

This formal definition was used to prove that a protocol provided in the same paper was coercion-resistant under the Decisional Diffie–Hellman Assumption. This definition provides stronger security properties since it covers forced abstention, randomization, and simulation attacks. However, this framework is quite restrictive, since it only considers protocols with a specific structure, (i.e., neither Bingo Voting nor ThreeBallot are in this class of protocols).

A similar approach was used by Unruh and Müller-Quade in [4], extending coercion resistance to a wider range of protocols.

3.1.2. δ-Coercion Resistance

Küsters, Truderung, and Vogt proposed in [5] a new definition covering a wider range of protocols while also allowing to measure the level of coercion resistance. In their model, a probability distribution of the possible choices of votes is associated to the honest voters and is supposed to be known by the coercer (a realistic assumption considering opinion polls). Moreover, it is supposed that at least one step in the protocol cannot be done by the coercer alone (e.g., register, voting, etc.).

The following notions are needed to introduce the definition: Firstly, a function f is overwhelming when $1 - f$ is negligible, and it is *δ-bounded* when f is bounded by δ plus a negligible function. Additionally, an election system S is represented by the number of choices available for the voters, the number of voters, the number of honest voters, and a probability distribution. Finally, a run of S is a run of an instance of S, being a system $(c||v||e_S)$ where $c \in C_S$ is a coercion strategy, $v \in V_S$ a counter-strategy, and e_S a system containing all honest participants. The set of counter-strategies V_S notably contains a *dummy strategy* dum which models the voter giving in to the coercer. Finally, for a security parameter l, a system S (or an instance of S) and a property γ, $Pr[S^{(l)} \mapsto \gamma]$ denotes the probability of γ being satisfied in a run of S.

Definition 6 (Coercion resistance)**.** *Let P be a protocol and $S = P(k, m, n, \vec{p})$ be an election system. Let also $\delta \in [0, 1]$, and γ be properties of S. The system S is δ-coercion-resistant w.r.t. γ, if there exists $\tilde{v} \in V_S$ such that for all $c \in C_S$ we have:*

- $Pr[(c||\tilde{v}||e_S)^{(l)} \mapsto \gamma]$ *is overwhelming as a function of the security parameter.*
- $Pr[(c||\mathsf{dum}||e_S)^{(l)} \mapsto 1] - Pr[(c||\tilde{v}||e_S)^{(l)} \mapsto 1]$ *is δ-bounded as a function of the security parameter.*

The property γ represents the voter's goal. Therefore, the first definition states that the coerced voter will achieve her goal with overwhelming probability, no matter which coercion strategy was used.

The second point states that the coercer cannot distinguish the case where the voter escapes coercion from the case where she gives in and uses the dum strategy. The coercer has almost the same probability of accepting the run in both cases; thus, this imprecision is formalized as δ-bounded.

This difference is required to be δ-bounded instead of negligible because there might be some cases where even a perfect counter-strategy is not enough. For instance, the coercer will be able to distinguish both cases if the candidate he asked the voter to vote for did not receive a single vote. This δ provides a precise measure of the coercion resistance level and can vary according to several parameters, notably the number of honest voters, the number of candidates, or the probability distribution $\vec{p}$.

In order to properly study the coercion resistance level of protocols, the optimal δ designed as δ_{min} such that the ideal voting protocol is δ_{min}-coercion-resistant but is not δ'-coercion-resistant for any $\delta' < \delta_{min}$ has to be determined. The corresponding formula was provided in the article, with its values depending on the number of honest voters, candidates, and on the probability distribution $\vec{p}$.

Two case studies were conducted using this result: Initially, the authors showed that the Bingo Voting system has the same level of coercion resistance as the ideal protocol. Subsequently, they showed that the ThreeBallot's level of coercion resistance rapidly decreases as the number of candidates grows. Moreover, the level of coercion resistance is also significantly lower than the one of the ideal protocol, even with only a few candidates. This framework was also reused by the authors to prove that Scantegrity II has an optimal level of coercion resistance, i.e., the same as the ideal protocol [6].

It is important to note that this definition covers multi-voter coercion. Moreover, the flexibility of defining the voter's goal γ allows to obtain a more precise value for the level of coercion resistance since δ might vary depending on the goal (i.e., if the voter wants to vote for a candidate who is unlikely to get many votes, then δ will be bigger than in the case where the voter supports a more popular candidate).

3.2. Applied Pi-Calculus

3.2.1. Swap Coercion Resistance

The first definition of coercion resistance using applied pi-calculus was provided by Delaune, Kremer, and Ryan in [7]. In order to model this security property, the authors previously defined voting processes in applied pi-calculus.

Definition 7 (Voting process)**.** *A voting process is a closed plain process*

$$VP \equiv \nu\tilde{n}.(V\sigma_1|\ldots|V\sigma_n|A_1|\ldots|A_m). \tag{1}$$

The $V\sigma_i$ are the voter processes, the A_j are the election authorities which are required to be honest, and the $\tilde{n}$ are channel names. It is also assumed that $v \in dom(\sigma_i)$ is a variable referring to the value of the vote. An evaluation context S similar to VP is defined, with the difference that it has a hole instead of the $V\sigma_i$.

As per the definition, the processes $A_1 \ldots A_m$ represent honest election authorities. The authorities corrupted by the coercer are not modeled, since all the possible behaviors of the coercer are considered (which includes the corrupted authorities). Moreover, it is assumed that a private anonymous channel exists between the voter and the election administrators.

Finally, two transformations are needed to define coercion resistance: the first one converts a process P into another process P^{ch} which reveals all secret data and inputs on the channel ch. The second one transforms P into P^{c_1,c_2} which, in addition to revealing the secret data on c_1, also takes orders from c_2 before sending a message or branching. The inductive definitions of these transformations can be found in [7].

Initially, one could think about a coercion resistance definition as follows: a protocol is coercion-resistant if there exists V' such that

$$S[V_A\{^{?}/_{v}\}^{c_1,c_2} | V_B\{^{a}/_{v}\}] \approx_l S[V' | V_B\{^{c}/_{v}\}] \tag{2}$$

Intuitively, we have on the left side the coerced voter $V_A\{^{?}/_{v}\}$ who will be forced to vote c no matter what she intended to vote, and on the right the process V' manage to vote a despite the coercion attempt. Nonetheless, if the coercer asks the voter to vote $d \neq c$, then the process $V_B\{^{c}/_{v}\}$ will not be able to counterbalance the result, leading to an obvious difference between the two sides.

This is the reason why a context C is needed in the definition to model the coercer's behavior and ensure that the voter will be coerced to vote c. This leads to the definition of coercion resistance:

Definition 8 (Coercion resistance)**.** *A voting process is coercion-resistant if there exists a closed plain process V' such that for any $C = \nu c_1.\nu c_2(_{|}P)$ satisfying $\tilde{n} \cap fn(C) = \emptyset$ and $S[C[V_A\{^{?}/_{v}\}^{c_1,c_2}] | V_B\{^{a}/_{v}\}] \approx_l S[V_A\{^{c}/_{v}\}^{chc} | V_B\{^{a}/_{v}\}]$, we have*

- $C[V']^{\setminus out(chc,.)} \approx_l V_A\{^{a}/_{v}\}$;
- $S[C[V_A\{^{?}/_{v}\}^{c_1,c_2}] | V_B\{^{a}/_{v}\}] \approx_l S[V' | V_B\{^{c}/_{v}\}]$.

Informally, this definition reads as follows: a voting protocol is coercion-resistant if (i) there exists a voting process V', i.e., a counter-strategy succeeding to vote a despite the coercion attempt, and if (ii) the coercer cannot know whether this counter-strategy was used or not, provided that another voter V_B counterbalances the result.

This article formally proves that coercion resistance implies receipt-freeness. Conducting a formal analysis of voting protocols with this definition is quite difficult according to the authors, since proving coercion resistance requires reasoning about all contexts C.

Three case studies were implemented in this article: two over lacking coercion resistance protocols and one over a coercion-resistant protocol. The last case study is considered to be rather informal by the authors due to the aforementioned reason. This formal framework was also used by Cortier and Wiedling to analyze the Norwegian e-voting protocol [8]. Even though their work focuses on secrecy, it is interesting to note that the present framework allowed for part of their study to be done using the ProVerif tool.

It should also be noted that this framework does not include the randomization attack or the forced abstention attack described in the work of Juels et al. [1]. Since the coercer can count the votes for each candidate in this model, the authors argue that withstanding such attacks would not be possible. Finally, since the definitions relies on the swap of two votes, it is not suitable when weighted votes are used (swapping two votes could lead to different results).

Another definition was proposed by Backes et al. in [9]. In this case, forced abstention is taken into consideration, assuming there is at least one other abstaining voter. They used their framework in order to conduct an analysis of the protocol proposed by Juels et al. in [1] using ProVerif. They were able to automatically prove that this protocol satisfies coercion resistance. However, it still required non-negligible human effort to set up the proof.

3.2.2. Multi-Voter Coercion

An improved definition of coercion resistance based on the work previously described was proposed by Dreier, Lafourcade, and Lakhnech in [10] in order to manage weighted votes. This model also allowed defining situations where multiple voters were coerced. A new definition is needed for this model in order to hide all but one channel. It will be used to reason about the results on the dedicated channel *res*.

Definition 9 ($P|_c$)**.** *Let $P|_c = \nu\tilde{ch}.P$ where $\tilde{ch}$ are all channels except for c, i.e., all the channels are hidden except for c.*

Therefore, the new version of the definition can be introduced.

Definition 10 (Single-Voter Coercion Resistance (SCR))**.** *A voting protocol ensures Single-Voter Coercion Resistance if for any voting processes $VP_A = \nu\tilde{n}.(V\sigma_{id_1}\sigma_{v_1^A}|\ldots|V\sigma_{id_n}\sigma_{v_n^A}|A_1|\ldots|A_l)$, $VP_B = \nu\tilde{n}.(V\sigma_{id_1}\sigma_{v_1^B}|\ldots|V\sigma_{id_n}\sigma_{v_n^B}|A_1|\ldots|A_l)$ and any number $i \in \{1,\ldots,n\}$ there exists a process V_i' such that for any context C_i with $C_i = \nu c_1.c_2.(\_|P_i)$ and $\tilde{n} \cap fn(C) = \emptyset$, $VP'_A[C_i[VC\sigma_{id_i}\sigma_{v_i^A})^{c_1,c_2}]] \approx_l VP'_A[(V\sigma_{id_i}\sigma_{v_i^A})^{chc_i}]$ we have $C_i[V_i']^{\setminus out(chc,.)} \approx_l V\sigma_{id_i}\sigma_{v_i^B}$ and*

$$VP_A|_{res} \approx_l VP_B|_{res} \implies VP'_A[C_i[(C\sigma_{id_i}\sigma_{v_i^A})^{c_1,c_2}]] \approx_l VP'_B[C_i[V_i']],$$

where VP_A and VP'_B are similar to VP_A and VP_B, but with holes for the voter $V\sigma_{id_i}$.

The aforementioned definition does not greatly differ from the previous one: if two instances of a voting protocol give the same result, then they are bi-similar (i.e., the attacker cannot distinguish them) even if a counter-strategy to the coercion was used in one of the instances.

Moreover, this definition of coercion resistance and the previous one are equivalent if swapping votes does not have an impact on the result. This is formalized in the article with a property called Equality of Votes: two instances of the protocol with the same voters give the same result if and only if the votes are a permutation of each other.

The extension to Multi-Voter Coercion is rather simple: a subset of voters is considered instead of one single voter.

Definition 11 (Multi-Voter Coercion Resistance (MCR))**.** *A voting protocol ensures Multi-Voter Coercion Resistance if for any voting processes $VP_A = \nu\tilde{n}.(V\sigma_{id_1}\sigma_{v_1^A}|\ldots|V\sigma_{id_n}\sigma_{v_n^A}|A_1|\ldots|A_l)$, $VP_B = \nu\tilde{n}.(V\sigma_{id_1}\sigma_{v_1^B}|\ldots|V\sigma_{id_n}\sigma_{v_n^B}|A_1|\ldots|A_l)$ and any subset $I \subset \{1,\ldots,n\}$, $I \neq \{1,\ldots,n\}$, if $n > 1$, there exists a process V_i' such that for any context C_i, $i \in I$ with $C_i = \nu c_1.c_2.(\_|P_i)$ and $\tilde{n} \cap fn(C) = \emptyset$, $VP'_A[C_i[VC\sigma_{id_i}\sigma_{v_i^A})^{c_1,c_2}]] \approx_l VP'_A[(V\sigma_{id_i}\sigma_{v_i^A})^{chc_i}]$ we have $\forall i \in I$: $C_i[V_i']^{\setminus out(chc,.)} \approx_l V\sigma_{id_i}\sigma_{v_i^B}$ and*

$$VP_A|_{res} \approx_l VP_B|_{res} \implies VP'_A[|_{i\in I}C_i[(C\sigma_{id_i}\sigma_{v_i^A})^{c_1,c_2}]] \approx_l VP'_B[|_{i\in I}C_i[V_i']],$$

where VP_A and VP'_B are similar to VP_A and VP_B but with holes for all voter $V\sigma_{id_i}$, $i \in I$.

While MCR intuitively implies SCR, the opposite is true only if two properties are verified: (i) Correctness: if in two instances of a voting protocol, the voter's choices are the same, then the results are identical and (ii) Modularity: a voting protocol is modular if (a) for any two voting processes, there exists a bi-similar process to the parallel composition of these two processes and (b) if any voting process can be decomposed in n processes such that it is bi-similar to the parallel composition of these processes. Intuitively, this property allows decomposing an instance with multiple attacked voters into instances where at most one voter is attacked, allowing to use the assumptions about single-voter coercion over these instances.

Three case studies were conducted in related articles using the presented model, proving that Bingo Voting ensures both MCR by combining the result of a previous study that used the DKR model and the relations described previously between Swap coercion resistance, SCR, and MCR.

3.3. Logic

3.3.1. ATL*-Based Definitions

ATL* was used by Tabatabaei, Jamroga, and Ryan in [11] to express the informal definitions from several articles, thus providing a way to highlight the differences in their approaches. The transcriptions of the different definitions are as follows:

- For Delaune, Kremer, and Ryan, two interpretations of the informal definition are proposed: either the coercer cannot know the value of the coerced voter's vote, or he must not be able to find any correlation between the voter and her vote. This leads to two versions of the definition:

$$\bigwedge_{v \in V \setminus \{c\}} \bigwedge_{i \in Bal} \neg \langle\langle c, v \rangle\rangle F(voted_{v,i} \wedge K_c voted_{v,i})$$

$$\bigwedge_{v \in V \setminus \{c\}} \bigwedge_{i \in Bal} \neg \langle\langle c, v \rangle\rangle F(voted_{v,i} \wedge \bigvee_{j \in Bal \setminus i} K_c \neg voted_{v,j})$$

  Despite the voter and the coercer's cooperation, no link can be created between the voter and her vote by the coercer.
- Juels, Catalano, and Jakobsson's definition is translated into three formulas: for basic coercion resistance, randomization attacks, and forced abstention attacks, respectively.

$$\bigwedge_{v \in V \setminus \{c\}} \bigwedge_{i,j \in Bal} \langle\langle v \rangle\rangle F(voted_{v,i} \wedge B_c voted_{v,j})$$

  The voter can successfully deceive the coercer into thinking she followed his instructions.

$$\bigwedge_{v \in V \setminus \{c\}} \neg \langle\langle c, v \rangle\rangle F K_c crossed_{v,1}$$

  Where $crossed_{v,n}$ expresses that voter v has crossed the n^{th} slot on a ballot.

$$\bigwedge_{v \in V \setminus \{c\}} \neg \langle\langle c, v \rangle\rangle G(\bigwedge_{i \in Bal} \neg voted_{v,i} \wedge K_c \bigwedge_{i \in Bal} \neg voted_{v,i})$$

- For Kusters, Truderung, and Vogt's definition, the translation of the case where the coercer instructs the voter to vote for a certain candidate is as follows:

$$\bigwedge_{v \in V \setminus \{c\}} \bigwedge_{i,j \in Bal, i \neq j} \langle\langle v \rangle\rangle F(voted_{v,i} \wedge G \neg K_c \neg voted_{v,j})$$

  The voter has a strategy to reach her goal without the coercer finding out that she disobeyed his instructions.

The same framework was used in an article by Belardinelli et al. in [12]. The main outcome was to define the relation of bi-simulation between iCGS and prove that it preserves the interpretation of ATL* formulas. This was subsequently used in order to analyze two variants of Three Ballot: one where the voter's ballots are sorted by lexicographic order by the voting machine, and another where ballots are not sent to the bulletin board but instead the votes for each candidate are counted. The definition used was as follows:

$$\varphi_i = \neg \langle\langle att \rangle\rangle F((pub \wedge v_i) \rightarrow \bigvee_{1 \leq j < nc} K_{att}(j = ch_i))$$

This formula is a variant of coercion resistance which states that the attacker (who is also a voter) has no strategy allowing him to know how i voted. In order to have the variant of coercion resistance, this formula has to be verified for all agents i who are not the attacker.

Since the aforementioned three models were proven to be bi-similar, they verify the same ATL* formula and hence present the same security properties. Moreover, model-checking algorithms were run on these models to verify the different ATL* formulas. The bi-simulation between these models allowed for a significant gain in time and memory by simply undergoing the model checking on a "smaller" bi-similar reduction of the Three Ballot model.

3.3.2. A Probabilistic Definition

A similar approach was used by Schnoor in [13] using an extension of ATL* called QAPI. The main benefit of this new framework is the inclusion of probabilistic aspects in the definition of coercion resistance.

Since QAPI is an extension of ATL*, this article will only describe the new features of QAPI: Given a set of coalitions $A_1, \ldots, A_n$, ψ a path formula, $S_1, \ldots, S_n$ variables for strategies, and $\blacktriangleleft$ one of $\leq, <, \geq, >$, then $\langle\langle A_1 : S_1 \ldots A_n : S_n \rangle\rangle^{\blacktriangleleft \alpha} \psi$ is a state formula.

This formula is satisfied provided that if the coalitions play their respective strategies, then the resulting path satisfies the formula with probability $\blacktriangleleft\ \alpha$. QAPI also features the operator X^{-1} referring to the previous state. Finally, for a coalition A and a degree of information i, $\mathcal{K}_i^A \psi$ denotes the fact that A knows ψ is true with information degree i.

The definition of coercion resistance introduces the following sub-formulas: Let $\varphi^{A-coerc}$ and $\varphi^{A-counter}$ be formulas expressing that the running strategy signaled coercion or a counter-strategy, and let T be the test principal whose goal is to guess if the counter-strategy was used.

Let φ^{T-suc} be the formula that expresses the success of T and φ^V be the formula expressing that the voter voted according to her strategy S_V. The following formula expresses that the success probability of T is less than δ for both strategies:

$$\varphi^{T<\delta} = \neg((\langle\langle T : S_T, \mathcal{A} : S_{counter} \rangle\rangle^{\geq \delta} \varphi^{T-suc}) \wedge (\langle\langle T : S_T, \mathcal{A} : S_{coerc} \rangle\rangle^{\geq \delta} \varphi^{T-suc})$$

Then, this formula expresses that S_{coerc} signals coercion correctly (the analog for the counter-strategy is defined in the same way):

$$\varphi^{sig-coerc} = \langle\langle \mathcal{A} : S_{coerc} \rangle\rangle^{\geq 1} \Diamond \varphi^{A-coerc}$$

The following formula expresses that the voter manages to vote as she wants to thanks to the counter-strategy:

$$\varphi^{vote} = \langle\langle \mathcal{A} : S_{counter} \rangle\rangle^{\geq 1} \Diamond \varphi^{V}$$

Finally, the definition of coercion resistance can be expressed as:

$$\forall_3 S_{coerc} \exists_3 S_{counter} \forall_3 S_V \forall_3 S_T \varphi^{sig-coerc} \rightarrow (\varphi^{sig-counter} \wedge \varphi^{vote} \wedge \varphi^{T<\delta})$$

Informally, the previous definition implies that for every coercion strategy, the voter has a counter-strategy such that, whatever her vote is, she will be able to vote as she wants and there is no strategy for the test principal to find out with a probability greater than δ regardless of whether she submits it to the coercer or not.

Schnoor proved that the security properties in this model are decidable for convergent sub-term theories, implying that there exists an algorithm able to check whether properties are satisfied or not for a given protocol and corrupted identities.

4. Discussion

Coercion resistance is a security property that can be expressed in a wide variety of ways. One of the main points of divergence between the different definitions is the

presence or absence of probabilistic considerations, namely, whether the definition allows for potentially small but non-negligible probability for the attacker to detect that the voter attempted to avoid the coercion.

The definitions given in [5,13] provide a weaker definition since the coercer succeeds with a non-zero probability. On the positive side, they are more realistic, since coercion is unavoidable in very specific situations, such as the case (e.g., occurring in smaller elections) where all the votes end up being cast for the same candidate. Moreover, they also provide a way to precisely measure the level of coercion resistance, which can be especially useful to analyze a protocol according to certain parameters (number of candidates, honest voters, etc.).

The security properties guaranteed by the researched definitions also vary, with the different attacks presented by Jules et al. in [1] (randomization, forced abstention) not covered in the definitions based on applied pi-calculus [7,8]. This difference is due to the use of a different attacker model. The formal transcriptions of the informal definitions in ATL* by Tabatabaei et al. in [11] also proved that the aim of the counter-strategies differ according to the definitions: Sometimes, the purpose is to make the coercer believe that the voter followed his instructions, while in other cases, the objective is only for the attacker to not know the coerced voter's final ballot. This distinction is relevant, since exercising control (and being able to make sure that the voter follows the instructions or to know when the voter has disobeyed) might be of higher preference to an attacker in certain cases. This topic needs to be further researched, taking into account different aspects of potential expected coercive situations and its expected practical implications.

To sum up, the presented frameworks offer different possibilities for the analysis of a protocol. Applied pi-calculus and temporal logics allowed for more autonomous proofs, using protocol verifiers or model-checkers. Even if these methods are not yet optimal, ProVerif was used on a simplified model in [8] and on Jules, Catalano, and Jakobsson's protocol in [9]. Moreover, model-checking algorithms were able to succeed only on small instances of the protocol in [12]. On the other hand, formal proofs of protocols tend to be increasingly difficult and tedious. Being able to automate this process could greatly simplify it while also removing potential human errors.

Table 1 summarizes our findings in comparing the frameworks. There is no framework clearly outweighing the rest. Therefore, it is to be decided on a case-to-case basis which properties should be prioritized for each specific election: protection against specific attacks (forced abstention and randomization), tolerance toward non-negligible probabilities of the coercer succeeding or the comparatively higher level of assurance to be obtained from automatable proofs. For such decisions to be effective, further research needs to be done on developing and communicating election requirements involving a variety of stakeholders.

Table 1. Summary table.

Framework	Forced Randomization	Forced Abstention	Partially Automatable	Probabilistic
JCJ	●	●	○	○
KTV	●	●	○	●
DKR	○	○	●	○
Backes	○	◓ [1]	●	○
DLL	○	○	●	○
Belardinelli	○	○	●	○
Schnoor	○	○	●	●

● = Holds ○ = Does not hold/not studied ◓ = Requires an assumption. [1] Requires at least one other abstaining voter.

5. Conclusions

In order to conduct a formal security analysis of a voting protocol, the different properties must be precisely defined first. The literature review in the present article focused on the different definitions of coercion resistance proposed in the academic literature. We have categorized and discussed their potential applications and limitations.

Our results show that while a number of definitions have been proposed, they differ according to the security guarantees they ensure, i.e., whether a non-negligible probability of being unable to protect against coercion is acceptable, or whether randomization and forced abstention attacks should be protected against.

Therefore, our findings stress the need to communicate to the election officials not just the fact that the system provides coercion resistance (which has been formally proven) but also the relevant details of the concrete guarantees it provides. By doing so, an informed, individualized decision can be made for each particular system in a specific election scenario, including which non-technical measures (e.g., voter education or implementing safe and accessible ways to report coercion to authorities) should be applied to reinforce the technical security measures. Investigating the most effective ways for such a communication is an important topic for future works.

One final important note is that this article focused solely on coercion resistance. Other key security properties for e-voting, such as receipt-freeness (which is a weaker form of coercion resistance), or verifiability were not contemplated. They are also of great importance in the research literature, and therefore, we believe that there is a need to perform a full analysis on them over an e-voting protocol, thus constituting an interesting topic for further research.

Author Contributions: This work was carried out as part of an internship by S.R. and supervised by O.K. and D.Y.M.d.B. All authors have read and agreed to the published version of the manuscript.

Funding: This research received no external funding.

Institutional Review Board Statement: Not applicable.

Informed Consent Statement: Not applicable.

Data Availability Statement: Not applicable.

Conflicts of Interest: The authors declare no conflict of interest.

References

1. Juels, A.; Catalano, D.; Jakobsson, M. Coercion-resistant electronic elections. In Proceedings of the 2005 ACM workshop on Privacy in the Electronic Society, Alexandria, VA, USA, 7 November 2005; pp. 61–70.
2. Abadi, M.; Fournet, C. Mobile values, new names, and secure communcation. In Proceedings of the 28th ACM Symposium on Principles of Programming Languages (POPL'01), London, UK, 17–19 January 2001; pp. 104–115.
3. Alur, R.; Henzinger, T.A.; Kupferman, O. Alternating-time temporal logic. *J. ACM* **2002**, *49*, 672–713. [CrossRef]
4. Unruh, D.; Mûller-Quade, J. Universally Composable Incoercibility. Available Online: https://eprint.iacr.org/2009/520.pdf (accessed on 1 December 2021).
5. Küsters, R.; Truderung, T.; Vogt, A. A game-based definition of coercion resistance and its applications. *J. Comput. Secur.* **2012**, *20*, 709–764. [CrossRef]
6. Küsters, R.; Truderung, T.; Vogt, A. Proving Coercion-Resistance of Scantegrity II. In *Information and Communications Security. ICICS 2010. Lecture Notes in Computer Science*; Soriano, M., Qing, S., López, J., Eds.; Springer: Berlin/Heidelberg, Germany, 2010; Volume 6476. [CrossRef]
7. Delaune, S.; Kremer, S.; Ryan, M. Verifying privacy-type properties of electronic voting protocols. *J. Comput. Secur.* **2009**, *17*, 435–487. [CrossRef]
8. Cortier, V.; Wiedling, C. A Formal Analysis of the Norwegian E-voting Protocol. In *Principles of Security and Trust. POST 2012. Lecture Notes in Computer Science*; Degano, P., Guttman, J.D., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; Volume 7215. [CrossRef]
9. Backes, M.; Hritcu, C.; Maffei, M. Automated Verification of Remote Electronic Voting Protocols in the Applied Pi-Calculus. In Proceedings of the 2008 21st IEEE Computer Security Foundations Symposium, Pittsburgh, PA, USA, 23–25 June 2008; pp. 195–209. [CrossRef]

10. Dreier, J.; Lafourcade, P.; Lakhnech, Y. Defining Privacy for Weighted Votes, Single and Multi-voter Coercion. In *Computer Security—ESORICS 2012. ESORICS 2012. Lecture Notes in Computer Science*; Foresti, S., Yung, M., Martinelli, F., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; Volume 7459. [CrossRef]
11. Tabatabaei, M.; Jamroga, W.; Ryan, P.Y. Expressing receipt-freeness and coercion-resistance in logics of strategic ability: Preliminary attempt. In Proceedings of the 1st International Workshop on AI for Privacy and Security, PrAISe@ECAI 2016, The Hague, The Netherlands, 29–30 August 2016; pp. 1:1–1:8. [CrossRef]
12. Belardinelli, F.; Condurache, R.; Dima, C.; Jamroga, W.; Knapik, M. Bisimulations for verifying strategic abilities with an application to the ThreeBallot voting protocol. *Inf. Comput.* **2021**, *276*, 104552. [CrossRef]
13. Schnoor, H. Deciding Epistemic and Strategic Properties of Cryptographic Protocols. In *Computer Security—ESORICS 2012. ESORICS 2012. Lecture Notes in Computer Science*; Foresti, S., Yung, M., Martinelli, F., Eds.; Springer: Berlin/Heidelberg, Germany, 2012; Volume 7459. [CrossRef]