Divertible and subliminal-free zero-knowledge proofs for languages[1]

Mike Burmester
Information Security Group, Department of Mathematics
Royal Holloway – University of London, Egham, Surrey TW20 OEX, U.K.

Yvo G. Desmedt
Department of EE & CS, University of Wisconsin – Milwaukee
Milwaukee, WI 53201, U.S.A.

Toshiya Itoh
Department of Information Processing
Interdisciplinary Graduate School of Science and Engineering
Tokyo Institute of Technology, Midori-ku, Yokohama 226, Japan

Kouichi Sakurai
Department of Computer Science and Communication Engineering
Kyushu University, 6-10-1 Hakozaki, Higashi-ku, Fukuoka 812-81, Japan

Hiroki Shizuya
Education Centre for Information Processing
Tohoku University, Kawauchi, Aoba-ku, Sendai 980-77, Japan

Abstract. Divertible proofs are extensions of interactive proofs in which an active eavesdropper, the warden, makes the prover and the verifier untraceable. The warden is transparent to both the prover and the verifier. With subliminal-free proofs the warden controls subliminal messages. In this paper we present divertible and subliminal-free zero-knowledge proofs for various languages. We consider both graph isomorphism and graph non-isomorphism. We show that under a cryptographic assumption, any language in NP has a divertible and a subliminal-free zero-knowledge proof, and then extend this result to IP for subliminal-free proofs. Finally we discuss various applications of divertible and subliminal-free zero-knowledge proofs.

Key Words. Zero-knowledge, untraceability, divertibility, subliminal-free, cryptographic protocols, proof systems, identification.

1 Introduction

Interactive proofs systems (of membership) were introduced by Goldwasser, Micali and Rackoff [17]. Earlier Babai [2] considered Arthur-Merlin games, a somewhat similar type of proof system. Informally, an *interactive proof* (P, V) for a language L is an interactive protocol between a computationally unbounded probabilistic Turing machine P, the prover, and a probabilistic polynomial time Turing machine V, the verifier, which accepts $x \in L$ almost always, but rejects $x \notin L$ almost certainly. Zero-knowledge [17] was introduced to restrict the amount of knowledge revealed by the prover during the

[1]Earlier versions of this paper were presented at Eurocrypt '90 and Asiacrypt '91. Parts of this research were done while Mike Burmester and Yvo Desmedt visited EISS, University of Karlsruhe, Germany, and while visiting each other. Mike Burmester was partially supported by SERC Grant GR/F 5700, and Yvo Desmedt was supported by NSF Grant NCR-9004879 and NSF Grant INT-9123464. This work was done while Kouichi Sakurai was working for Mitsubishi Electric Corporation.

execution of the proof. Informally an interactive proof (P, V) is *zero-knowledge* if, when $x \in L$ the prover reveals no more than the assertion that x belongs to L.

The formal setting for interactive zero-knowledge proofs prevents a dishonest party (who uses a different program from the one specified by the proof) from cheating an honest party. It does not deal with the case when *both* parties are dishonest. Indeed there seems to be little justification for designing such systems. However one can envisage a scenario in which a dishonest prover may use an interactive proof to send secret subliminal information, even if this means that the honest verifier will not accept. A well-designed system should prevent this. In this paper we address such issues, and propose interactive proofs in which dishonest parties are prevented from using the proof system for a different purpose than intended.

Simmons has shown that it is possible to hide a subliminal message inside an authenticator [23]. In a similar way subliminal channels can be introduced in zero-knowledge interactive proofs [13]. These channels are closely related to covert channels, an important topic in computer security [10]. Desmedt, Goutier and Bengio first introduced subliminal-free proofs for quadratic residuocity [13]. The aspect of subliminal-freeness has also been discussed in [12], in particular in the context of authentication. Okamoto and Ohta [21] considered a setting in which an active eavesdropper, the warden W, diverts an interactive proof (P, W) to a proof (W, V) in such a way that any relationship between the proofs is concealed. So if provers P_1 and P_2 prove to a verifier V that $x \in L$, then V cannot trace back the proof to either P_1 or P_2. Similarly a prover P cannot trace the verifier V. Furthermore W is transparent, that is if W is removed then (P, V) is an interactive proof. Such proof systems (P, W, V) are called *divertible* proofs.[2] Okamoto and Ohta proved that there exist divertible zero-knowledge proofs for any commutative random self-reducible language [21].

The main similarity between divertible and subliminal-free proofs is that for both a warden W tries to enforce the honest distribution, when the prover or verifier are possibly dishonest. The essential differences are that with divertible proofs W must enforce the honest distribution and must be transparent, while with subliminal-free proofs W can halt if (he thinks) there is a subliminal message and W is not necessarily transparent.

In this paper we consider both subliminal-free and divertible zero-knowledge proofs. We show that any language in NP has a *statistically*[3] divertible proof (Definition 7) which is *computationally* zero-knowledge (Theorem 3) under the assumption that secure homomorphic commitments exist (Definition 12). A similar result applies to subliminal-free proofs (Theorem 4). We also show that graph isomorphism has a *statistically* divertible proof which is *perfectly* zero-knowledge (Theorem 1) with no cryptographic assumptions, and a subliminal-free proof which is *perfectly* zero-knowledge (Theorem 2). We consider graph non-isomorphism, and show that it has a subliminal-free proof which is *perfectly* zero-knowledge (Theorem 5). Finally we show that any language in IP (the class of languages which have interactive proofs) has a subliminal-free proof.

This paper is organized as follows. In Section 2 we give our definitions. In Section 3 we discuss our basic technique for obtaining divertible and subliminal-free zero-knowledge proofs and describe such a protocol for graph isomorphism. Then in Section 4 we show that any language in NP has a divertible and subliminal-free zero-knowledge proof. In Section 5 we extend our result for subliminal-free zero-knowledge proofs to graph non-isomorphism, and in Section 6 to any language in IP. In Section 7 we consider applications. We conclude in Section 8 with general remarks and discuss open problems.

[2] The first divertible zero-knowledge proof was presented in [13, pp. 37–38] in the context of subliminal-free proofs.

[3] By using a weaker definition of divertibility we could obtain *perfect* divertibility.

2 Model and Definitions

2.1 Background

We use the Goldwasser–Micali–Rackoff [17] model for interactive proofs. Let (P, V) be an interactive protocol, with P, V probabilistic Turing machines which share the same input tape and have communication tapes and private worktapes. P is the prover which has unlimited computational power, whereas V, the verifier, is computationally bounded by a polynomial in $|x|$, the length of the input x. A probabilistic Turing machine which replaces P in (P, V) is called *dishonest* if it has a different program from P. We denote by P' a possibly dishonest prover. Similarly V' is a possibly dishonest verifier.

We assume that all machines have a *history* tape on which a string is written. This is in contrast to most current applications in which only dishonest machines use such tapes. However our setting is more general (see example Section 2.4.2).

Definition 1. (P, V) is an *interactive proof* [17] for a language $L \subseteq \{0,1\}^*$ if, (Completeness) for any constant k, for any sufficiently long $x \in L$ given as input to (P, V): V accepts with probability at least $1 - |x|^{-k}$ (taken over the coin tosses of P and V), and (Soundness) for any constant k, for any sufficiently long $x \notin L$, for any P', on input x to (P', V): V accepts with probability at most $|x|^{-k}$ (taken over the coin tosses of P' and V).

The *view* [17] of V' when interacting with P in (P, V') on input x and history h, consists of the bit strings (messages) that V' receives from P and the portion of the random tape that V' reads. $(P, \underline{V'})(x, h)$ is the random variable whose value is the view of V'. The corresponding ensemble (family of random variables) is $\{(P, \underline{V'})(x, h)\}$.

Let $U(x)$ and $V(x)$ be random variables with parameter x.

Definition 2. $U(x)$ and $V(x)$ are *statistically k-close for x* if,

$$\sum_{\alpha \in \{0,1\}^*} |\operatorname{Prob}(U(x) = \alpha) - \operatorname{Prob}(V(x) = \alpha)| < |x|^{-k}.$$

$U(x)$ and $V(x)$ are *statistically indistinguishable* [17] on $L \subseteq \{0,1\}^*$ if they are statistically k-close, for all constants k and sufficiently long $x \in L$.

For the computational equivalent we consider Boolean circuits. Let C_x be a Boolean circuit with one Boolean output and let $F(x)$ be a random variable. We denote by $\operatorname{Prob}(F(x), C_x, x)$ the probability that C_x outputs 1 on input a random string distributed according to $F(x)$ [17]. A family of Boolean circuits $C = \{C_x\}$ with one Boolean output is a *poly-size family of circuits* if for some constant $c > 0$, all $C_x \in C$ have at most $|x|^c$ gates.

Definition 3. $U(x)$ and $V(x)$ are *computationally k-close for x with respect to C_x* if, $|\operatorname{Prob}(U(x), C_x, x) - \operatorname{Prob}(V(x), C_x, x)| < |x|^{-k}$. $U(x)$ and $V(x)$ are *computationally indistinguishable* [17] on $L \subseteq \{0,1\}^*$ if, for all poly-size families of circuits $C = \{C_x\}$, for all constants k and sufficiently long $x \in L$: $U(x)$ and $V(x)$ are computationally k-close with respect to the circuits C_x.

Definition 4. An interactive proof (P, V) is *perfectly (statistically) (computationally) zero-knowledge* on L [17] if, for any polynomial-time machine V', there is an expected polynomial-time machine $M_{V'}$, called the *simulator*, such that: the ensembles $\{M_{V'}(x, h)\}$ and $\{(P, \underline{V'})(x, h)\}$ are perfectly[4] (statistically) (computationally) indistinguishable on $L' = \{(x, h) \mid x \in L \text{ and } |h| = |x|^c\}$, $c > 0$ constant.

[4] Perfectly indistinguishable ensembles are equal.

2.2 Proofs with warden

We next consider three party proofs. Let (P, W, V) be an interactive protocol in which a prover P and a verifier V communicate with each other only through an active eavesdropper W. P has unlimited computational power whereas W and V are polynomial-time. We call W the *warden*. All three parties P, W, V are interactive probabilistic Turing machines. $W^{\leftrightarrow V}$ means W with oracle V, where V is consulted through the communication tapes that it shares with W. So $(P, W^{\leftrightarrow V})$ is the interactive protocol (P, W) for which V is an oracle for W. Similarly $({}^{P\leftrightarrow}W, V)$ is the interactive protocol (W, V) for which W consults P as an oracle. Let $\mathbf{h} = (h_{P'}, h_{V'})$ be the histories for P', V'. The history of P' can be used to share information (with entropy greater than 0) with V'. As with two party protocols, $({}^{P'\leftrightarrow}W, \underline{V'})(x, \mathbf{h})$ is the random variable whose value is the view of V' when interacting with W, which consists of the messages that V' receives from W, while interacting with P', and the portion of the random tape that V' reads. Similarly $(\underline{P'}, W^{\leftrightarrow V'})(x, \mathbf{h})$ is the random variable whose value is the view of P' when interacting with W. $(P, \underline{W', V'})(x, h_{W'V'})$ is the random variable whose value is the *joint view* of W' and V' when interacting with P, which consists of the messages that P sends and the portion of the random tapes that W', V' read. Here $h_{W'V'}$ is the joint history of W' and V'. The corresponding ensembles are $\{({}^{P'\leftrightarrow}W, \underline{V'})(x, \mathbf{h})\}$, $\{(\underline{P'}, W^{\leftrightarrow V'})(x, \mathbf{h})\}$ and $\{(P, \underline{W', V'})(x, h_{W'V'})\}$ respectively. The honest warden W has an empty history tape.

Definition 5. (P, W, V) is an *interactive proof with warden* for L if,

1. *Completeness for* V. For any k, for any sufficiently long $x \in L$ given as input to (P, W, V): V accepts with probability at least $1 - |x|^{-k}$ (taken over the coin tosses of P, W, V).
2. *Soundness for* V. For any k, for any sufficiently long $x \notin L$, for any P' and W', on input x to (P', W', V): V accepts with probability at most $|x|^{-k}$ (taken over the coin tosses of P', W', V).

Definition 6. An interactive proof (P, W, V) with warden is *perfectly* (*statistically*) (*computationally*) *zero-knowledge*[5] on L if, for any polynomial-time machines W', V' there is an expected polynomial-time machine $M_{W'V'}$ such that: the ensemble $\{M_{W'V'}(x, h_{W'}, h_{V'})\}$ is *perfectly* (*statistically*) (*computationally*) indistinguishable from $\{(P, \underline{W', V'})(x, h_{W'}, h_{V'})\}$ on $L' = \{(x, h_{W'}, h_{V'}) \mid x \in L \text{ and } |(h_{W'}, h_{V'})| = |x|^c\}$, $c > 0$ constant. That is, $M_{W'V'}$ simulates the joint view of W' and V' on L'.

From this definition it follows that the view $\{({}^{P\leftrightarrow}W, \underline{V'})(x, h_{V'})\}$ of V' while interacting with (an honest) W in an interactive zero-knowledge proof with warden can be simulated.

2.3 Divertible proofs

We now consider interactive proofs (P, W, V) in which the warden is transparent and the prover and verifier are untraceable.

Transparency requires that (P, W, V) remains a proof for L even when the warden W is inactive (or 'removed'): that is when W simply relays the messages of P and V. For untraceability we consider a prover P' that will be accepted by V when $x \in L$. Such a P' should not see any difference between the honest verifier V and any other possibly dishonest verifier V' when interacting through W. Also a possibly dishonest verifier V' should not see any difference between the honest prover P and a prover P' when interacting through W, even if P' tries to be traceable.

Definition 7. An interactive proof with warden (P, W, V) for L is *perfectly* (*statistically*) (*computationally*) *divertible*[6] if,

[5] This definition is based on [17] and [21].

[6] This definition is based on [17] and [21]. We have made some changes to allow for proofs of membership.

1. *Transparency.* (P, V) is an interactive proof for L in which P and V interact, each running as the respective Turing machines in (P, W, V).

2. $(P, W^{\leftrightarrow V})$ – *proof.*
(a) *Completeness for W.* For any k, for any sufficiently long $x \in L$ given as input to (P, W, V): W accepts with probability at least $1 - |x|^{-k}$ (taken over the coin tosses of P, W, V).
(b) *Weak soundness for W.* For any k, for any sufficiently long $x \notin L$, for any P', on input x to (P', W, V): W accepts with probability at most $|x|^{-k}$ (taken over the coin tosses of P', W, V).

3. *Untraceability.* Let $L' = \{(x, \mathbf{h}) \mid x \in L \text{ and } |\mathbf{h}| = |x|^c, \mathbf{h} = (h_{P'}, h_{V'})\}$, $c > 0$. For any prover P' for which (P', V) is an interactive proof for L, for any verifier V',

   (a) the ensembles $\{(\underline{P'}, W^{\leftrightarrow V'})(x, \mathbf{h})\}$ and $\{(\underline{P'}, V)(x, \mathbf{h})\}$ are perfectly (statistically) (statistically[7]) indistinguishable on L', and

   (b) the ensembles $\{({}^{P'\leftrightarrow}W, \underline{V'})(x, \mathbf{h})\}$ and $\{(P, \underline{V'})(x, \mathbf{h})\}$ are perfectly (statistically) (computationally) indistinguishable on L'.

Remark 1. A weaker form of untraceability could restrict P' to provers who are accepted by V with the same probability as V accepts the honest prover P. We feel that this restriction is too severe and does not capture the essence of untraceability.

Remark 2. If the dishonest prover P' tries 'too hard' to be traceable then the honest verifier V will reject $x \in L$, in which case V' may see a difference. However then, we do not have an interactive proof any more. With divertible proofs we *restrict* ourselves to provers P' for which (P', V) is an interactive proof. A stronger condition would allow for *any* dishonest prover P'. In the following section we consider such a scenario.

2.4 Subliminal-free proofs

With a subliminal-free proof the warden will detect any attempt by the prover (or verifier) to exchange subliminal messages. There are two ways in which a subliminal channel can be established. The first one is by abusing the system. In this case the prover (or verifier) tries to hide secret messages in the strings it exchanges while the protocol is executed. This can be prevented by requiring that the system is abuse-free. The second way is to use a non-minimal protocol which is specifically designed to allow for subliminal channels. We shall discuss this in Section 2.4.2.

2.4.1 Abuse-free proofs

A proof is abused if it is used for a different purpose than intended. In this case we *cannot* restrict ourselves to dishonest provers P' which the honest verifier V will accept. Indeed a dishonest verifier V' may accept a P' which V would have rejected, if V' can somehow benefit from this, *e.g.,* establish a subliminal channel. With abuse-free protocols the warden W should detect abuses. For this purpose W returns a bit w_d as special output which it sets to 1 if it detects an abuse.

We demand that two conditions be satisfied for a proof to be abuse-free. The first is fairness. That is, when P and V are honest and $x \in L$, the warden should not detect an abuse. The second is detectability. That is, if the warden fails to detect an abuse, then: (i) a possibly dishonest prover P' cannot see any difference between the bit strings it gets from a conspiring V' and those it gets from the honest verifier V, and (ii) a possibly dishonest verifier V' cannot see any difference between the bit strings it gets from P' and those it gets from P. The formal definition follows.

[7] P has unlimited computational power, so it makes no sense for these ensembles to be computationally indistinguishable.

Let W set $w_d = 0$ if it fails to detect an abuse and let $({}^{P'\leftrightarrow}W, \underline{V'})_{|w_d=0}(x, \mathbf{h})$ be $({}^{P'\leftrightarrow}W, \underline{V'})(x, \mathbf{h})$ restricted to $w_d = 0$, and $(\underline{P'}, W^{\leftrightarrow V'})_{|w_d=0}(x, \mathbf{h})$ be $(\underline{P'}, W^{\leftrightarrow V'})(x, \mathbf{h})$ restricted to $w_d = 0$.

Definition 8. Let $\mathbf{A} = (P, W, V)$ be an interactive proof with warden for L, and let $L' = \{(x, \mathbf{h}) \mid x \in L \text{ and } |\mathbf{h}| = |x|^c, \mathbf{h} = (h_{P'}, h_{V'})\}$, $c > 0$ constant. $\mathbf{A}$ is *abuse-free*[8] if,

1. *Fairness.* For any k, for any sufficiently long $x \in L$ given as input to $\mathbf{A}$: $\mathrm{Prob}_{\mathbf{A}}(w_d = 0) \geq 1 - |x|^{-k}$, where $\mathrm{Prob}_{\mathbf{A}}(w_d = 0)$ is the probability that $w_d = 0$ during the execution of the proof $\mathbf{A}$.

2. *Detectability.* For any k, for any sufficiently long $(x, \mathbf{h}) \in L'$, for any P' and V' which communicate through W, on input x and history $\mathbf{h}$ to $\mathbf{A}' = (P', W, V')$ we have, either $\mathrm{Prob}_{\mathbf{A}'}(w_d = 0) < |x|^{-k}$, or
(a) *Conditionally perfect.* $(\underline{P'}, W^{\leftrightarrow V'})_{|w_d=0}(x, \mathbf{h}) = (\underline{P'}, W^{\leftrightarrow V})_{|w_d=0}(x, \mathbf{h})$ and $({}^{P'\leftrightarrow}W, \underline{V'})_{|w_d=0}(x, \mathbf{h}) = ({}^{P\leftrightarrow}W, \underline{V'})_{|w_d=0}(x, \mathbf{h})$.
(b) *Conditionally statistical.* The ensembles $\{(\underline{P'}, W^{\leftrightarrow V'})_{|w_d=0}(x, \mathbf{h})\}$ and $\{(\underline{P'}, W^{\leftrightarrow V})_{|w_d=0}(x, \mathbf{h})\}$ are statistically k-close for $(x, \mathbf{h})$ in L', and the ensembles $\{({}^{P'\leftrightarrow}W, \underline{V'})_{|w_d=0}(x, \mathbf{h})\}$ and $\{({}^{P\leftrightarrow}W, \underline{V'})_{|w_d=0}(x, \mathbf{h})\}$ are statistically k-close for $(x, \mathbf{h})$ in L'.
(c) *Conditionally computational.* The ensembles $\{(\underline{P'}, W^{\leftrightarrow V'})_{|w_d=0}(x, \mathbf{h})\}$ and $\{(\underline{P'}, W^{\leftrightarrow V})_{|w_d=0}(x, \mathbf{h})\}$ are statistically[7] k-close for $(x, \mathbf{h})$ in L', and the ensembles $\{({}^{P'\leftrightarrow}W, \underline{V'})_{|w_d=0}(x, \mathbf{h})\}$ and $\{({}^{P\leftrightarrow}W, \underline{V'})_{|w_d=0}(x, \mathbf{h})\}$ are computationally k-close for $(x, \mathbf{h})$ in L' with respect to the Boolean circuits $C_{(x,\mathbf{h})}$, provided V' is polynomial-time.

2.4.2 Minimal proofs

While subliminal channels of the first type deal with dishonest provers or verifiers who abuse a properly designed protocol, subliminal channels of the second type deal with dishonestly designed, or faulty, protocols. Let us describe one such protocol based on the Goldwasser–Micali–Rackoff proof for Quadratic Residuocity [17]. This uses an "atomic" subroutine with three steps in which, first the prover 'commits' to a particular string, then the verifier asks a randomly selected 'query' bit, and finally the prover sends her 'answer' which the verifier checks. The subroutine is repeated m times, where m is polynomial in the length of the input. We now modify this protocol by having the (honest) verifier take as its last query bit (the m-th bit) an appropriate bit of its history tape, instead of a random bit: in this case we assume that the honest verifier V uses its history tape. Clearly the modified protocol is still a perfect zero-knowledge proof of Quadratic Residuocity. Indeed the last bit of the verifier has no significant effect on the completeness or soundness of the proof, and the view of the verifier can be perfectly simulated. However the modified proof allows the verifier to send to the prover, additionally, one subliminal bit with each execution of the protocol.

Minimal proofs will prevent this. Informally, a proof is minimal if it does no more than what is strictly required [11]. With such proofs the verifier learns only one bit of knowledge (that $x \in L$ or otherwise), and the prover learns *nothing*, in an information theoretic sense. From our previous example we see that zero-knowledge proofs are not necessarily minimal. This is because zero-knowledge addresses only the knowledge that the verifier may get, not the information theoretic knowledge that the prover may get.

Definition 9. Let (P, W, V) be an interactive zero-knowledge proof for L with warden in which P and V may have been specified to use their history tape. (P, W, V) is *minimal* if there is an expected polynomial-time machine M_P such that: the ensemble $\{M_P(x, h_P)\}$ is statistically indistinguishable

[8] This definition is a particular case of the general definition in [11].

from $\{(\underline{P}, W^{\leftrightarrow V})(x, (h_P, h_V))\}$ on $L' = \{(x, (h_P, h_V)) \mid x \in L \text{ and } |h_P| = |h_V| = |x|^c\}$, $c > 0$.[9] That is, M_P simulates the view of P when interacting with $W^{\leftrightarrow V}$ on L'.

Observe that with minimal proofs we are only concerned with the view of honest parties, and that the simulator M_P only receives the history h_P. We now combine abuse-free proofs and minimal proofs to get subliminal-free proofs.

2.4.3 Subliminal-free proofs, fair wardens

Definition 10. An interactive zero-knowledge proof (P, W, V) for L is *subliminal-free* if it is abuse-free and minimal. We say that the *warden is (unrestricted) fair* if the fairness condition extends to all strings in $\{0, 1\}^*$.

A warden who is (unrestricted) fair will not detect any abuse when $x \notin L$ if the prover and verifier are honest.

2.5 Other types of proofs

Definition 11. A divertible (subliminal-free) interactive proof (P, W, V) for L is,

- *Sound for W* if, for any constant k, for sufficiently long $x \notin L$, for any P' and V', for any $\mathbf{h} = (h_{P'}, h_{V'})$, on input x and history $\mathbf{h}$ to (P', W, V'): W accepts with probability at most $|x|^{-k}$ (taken over the coin tosses of P', W, V').
- *Sabotage-free (for W)* if, for any constant k, for sufficiently long $x \in L$, for any V', for any $h_{V'}$, on input x and history $h_{V'}$ to (P, W, V'): W accepts with probability at least $1 - |x|^{-k}$ (taken over the coin tosses of P, W, V').

Remark 3. With a divertible proof (P, W, V) which is not sound for W, only V is a verifier in the Goldwasser–Micali–Rackoff [17] sense. That is, only V will reject $x \notin L$ with overwhelming probability whatever programs P' and W' use. Due to the sequential nature of the setting for divertible (subliminal-free) proofs, it is not possible to have sabotage-free proofs for V, *i.e.,* proofs in which the verifier accepts in (P, W', V) for any W' and $x \in L$, when L is not in BPP. For this reason we have not defined it.

A divertible proof (P, W, V) which is sound and sabotage-free for W, is a *sequential multi-verifier* proof. In this case both W and V are verifiers and are convinced unconditionally, and we have an interactive proof for both W and V (provided that W is honest) in the Goldwasser–Micali–Rackoff [17] sense.

2.6 Commitment functions and general notation

The following definition is based on the definition of probabilistic encryptions in [16, 15].

Definition 12. A *bit commitment* function[10] is a polynomial-time computable function $f\colon \{0,1\} \times \{0,1\}^* \to \{0,1\}^*$ for which $f(0,t) \neq f(1,t')$ for all $t, t' \in \{0,1\}^*$. Let $f_n(b)$, $b \in \{0,1\}$, be the random variable $f(b,t)$, $t \in_R \{0,1\}^n$. We call n the security parameter of f_n. A commitment f is *secure* [16, 15] if the ensembles $\{f_n(0)\}$ and $\{f_n(1)\}$ are computationally indistinguishable.
Finally, the commitment $f(b,t)$ is *opened* by revealing b and t.

[9] This condition is only sufficient for zero-knowledge proofs. In a more general context the knowledge that each party may get, must be minimal.

[10] Here we only consider unconditionally secure commitments for the verifier. We also have unconditionally secure commitments for the prover (hiding commitments) [5], but these are not used in this paper.

A well-known [16] example of a bit commitment is based on the function,

$$g_{s,m}(b,r) \quad = \quad s^b r^2 \bmod m, \text{ if } b \in \{0,1\} \text{ and } r \in Z_m^*, \tag{1}$$

where the modulus m is a Blum integer [3] and s is an appropriate quadratic non-residue. It is secure if it is hard to decide quadratic residuocity modulo a Blum integer. Here $f(b,t) = (g_{s,m}(b,r), s, m)$, where one part of the bit string t denoted by $\text{part}(t)$ is used to obtain the prime factors of m (for example, p and q) and the non-residue s (the parameters of the commitment), and the other part is used to determine the argument r (so $\text{part}(t) = (p,q,s)$ in the example). The commitment is opened by revealing b, r, the prime factors[11] of m, and s.

We next consider homomorphic commitments and blindings of commitments. Let $a \in_R A$ mean that the element a is selected from the set A uniformly and independently of other selections.

Definition 13. Let $\{F_n\}$ be a partitioning of the commitment f in which, for each n, F_n consists of the restrictions f_{par} of f obtained by limiting the length of its argument t to n, and by using one part of t to determine the parameters $\text{par} = \text{par(t)}$ of f_{par} and the remaining part of length $v(n)$, $v(n) > n^c$, $c > 0$ constant, as the argument r of f_{par}. So $f_{\text{par}}(b,r) = f(b,t)$, where $t = (\text{par}, r)$ with $|t| = n$ and $|r| = v(n)$.

We say that f is *homomorphic* if for each n and $f_{\text{par}} \in F_n$, a binary polynomial-time operation "$\cdot$" is defined on the $f_{\text{par}}(b,r)$'s such that: there is a polynomial-time algorithm which on input $b, b' \in \{0,1\}$ and $r, r' \in \{0,1\}^{v(n)}$, will output an $r'' \in \{0,1\}^{v(n)}$ for which $f_{\text{par}}(b,r) \cdot f_{\text{par}}(b',r') = f_{\text{par}}(b \oplus b', r'')$, where "$\oplus$" be addition mod 2; furthermore, if $r' \in \{0,1\}^{v(n)}$ is uniformly distributed then so is r''. The commitment $f_{\text{par}}(b,r)$ is *opened* by revealing b, r, and the parameters of f_{par} if necessary.

Definition 14. Let $g : \{0,1\}^* \to \{0,1\}^*$ be a polynomial-time computable function and $\{G_n\}$ a partitioning of g which consists of the restrictions g_{par} obtained by limiting the length of its argument t to n, as in Definition 13, and taking one part to determine the parameters par of g_{par} and the other part to determine the arguments u, r'. We say that g is a *blinding* of the commitment f if for each n and $f_{\text{par}} \in F_n$, there is an $n' < n^c$, $c > 0$ constant, and $g_{\text{par}'} \in G_{n'}$ such that: given $u = f_{\text{par}}(b,r)$ and $r' \in_R \{0,1\}^{v'(n')}$, there is an $r'' \in_R \{0,1\}^{v(n)}$ for which $g_{\text{par}'}(f_{\text{par}}(b,r), r') = f_{\text{par}}(b, r'')$.

The function (1) can also be used to define homomorphic commitments. For this purpose we take the length of the argument of g to be $n = 3|m|$, and use the first $2|m|$ bits to determine the prime factors of m and the non-residue s, and the remaining $|m|$ bits to determine r. The function $g_m(u,r) = ur^2 \bmod m$, $u, r \in Z_m^*$, can be used for a blinding of the bit commitment (1).

Below we will use the following notation. Let V be a finite set. Then $\text{Sym}\, V$ is the group of all permutations on V. Furthermore, if $\pi, \pi' \in \text{Sym}\, V$ are permutations then $\pi \circ \pi'$ is their composition (so $\pi \circ \pi'(x) = \pi(\pi'(x))$ for $x \in V$).

3 Basic technique

To illustrate our technique (*cf.*, [6]) we first consider a protocol for Graph Isomorphism (GI) which is obtained by adapting the Goldreich–Micali–Wigderson [15] proof to suit our needs. Then we will extend this to get a divertible proof.

Let $G = (V,E)$ be a graph with vertex set V and edge set E. If $\pi \in \text{Sym}\, V$ then πG is the graph (V,F) with $(u,v) \in E$ if and only if $(\pi(u), \pi(v)) \in F$. We use the notation $\mathbf{A} = (A_0, A_1)$ for ordered pairs. The (external) operator 'swap' is defined by $\text{swap}(e, \mathbf{A}) = (A_e, A_{\overline{e}})$, where $e \in \{0,1\}$ and

[11] So "opening" is total: opened functions cannot be reused.

$\overline{e} = 1 \oplus e$. Let $\mathbf{G} = (G_0, G_1)$ be a pair of graphs on the same vertex set V, and let $\boldsymbol{\pi} = (\pi_0, \pi_1)$ be a pair of permutations of $\operatorname{Sym} V$. We define $\boldsymbol{\pi}\mathbf{G}$ to be the pair of graphs $(\pi_0 G_0, \pi_1 G_1)$. It is easy to check that

$$\operatorname{swap}(e, \boldsymbol{\pi}\mathbf{G}) = \operatorname{swap}(e, \boldsymbol{\pi}) \operatorname{swap}(e, \mathbf{G}). \tag{2}$$

More generally

$$\operatorname{swap}(e, \boldsymbol{\pi}) \operatorname{swap}(f, \mathbf{G}) = \operatorname{swap}(e, \boldsymbol{\pi} \operatorname{swap}(e \oplus f, \mathbf{G})), \tag{3}$$

for any $e, f \in \{0, 1\}$. Finally if $\boldsymbol{\sigma} = (\sigma_0, \sigma_1)$ is a pair of permutations of $\operatorname{Sym} V$, then $\boldsymbol{\pi} \circ \boldsymbol{\sigma} = (\pi_0 \circ \sigma_0, \pi_1 \circ \sigma_1)$, where "$\circ$" is the composition of permutations.

Protocol 1. (An interactive zero-knowledge proof for GI)
Common input: A pair of graphs $\mathbf{G} = (G_0, G_1)$ with vertex set V and m edges.

V rejects if G_0, G_1 are not proper descriptions of graphs which have the same number of vertices and edges. Otherwise the following four steps are repeated m times, independently.

P1. P selects a pair of permutations $\boldsymbol{\pi} \in_R \operatorname{Sym} V \times \operatorname{Sym} V$ and sends V the pair of graphs $\mathbf{H} = \boldsymbol{\pi}\,(G_0, G_0)$.

V1. V sends P a bit $q \in_R \{0, 1\}$.

P2. P sends V the pair of permutations $\boldsymbol{\psi} = \boldsymbol{\pi} \circ (\sigma^q, \sigma^{\overline{q}})$, where $\sigma : G_1 \to G_0$ is an isomorphism ($q \notin \{0, 1\}$ is handled as $q = 0$).

V2. V checks that $\boldsymbol{\psi} \in \operatorname{Sym} V \times \operatorname{Sym} V$ and that $\mathbf{H} = \boldsymbol{\psi} \operatorname{swap}(q, \mathbf{G})$. If this fails, it halts and rejects.

If V has completed successfully m iterations of the above steps then it accepts.

(End of Protocol)

Observe that the main difference between this protocol and the one in [15] is that here the prover P sends a *pair* of permutations and then answers a pair of complementary bit queries.

Lemma 1. *Protocol 1 is an interactive proof for Graph Isomorphism which is perfectly zero-knowledge.*

Proof. This follows directly from the proof in [15, pp.703-706]. ■

We now extend Protocol 1 to get a divertible proof for Graph Isomorphism [6].

Protocol 2. (A divertible zero-knowledge proof for GI)
Common input: A pair of graphs $\mathbf{G} = (G_0, G_1)$ with vertex set V and m edges.

W, V reject if G_0, G_1 are not proper descriptions of graphs which have the same number of vertices and edges. Otherwise the following seven steps are repeated m times, independently.

P1. P selects $\boldsymbol{\pi} \in_R \operatorname{Sym} V \times \operatorname{Sym} V$ and sends W: $\mathbf{H} = \boldsymbol{\pi}\,(G_0, G_0)$.

W1. W selects $\boldsymbol{\pi}' \in_R \operatorname{Sym} V \times \operatorname{Sym} V$, $e \in_R \{0, 1\}$, and sends V: $\mathbf{H}' = \operatorname{swap}(e, \boldsymbol{\pi}'\mathbf{H})$.

V1. V sends W: $q \in_R \{0, 1\}$.

W2. W sends P: $q_1 = q \oplus e$; $q \notin \{0, 1\}$ is handled as $q = 0$.

P2. P sends W the pair of permutations: $\boldsymbol{\psi} = \boldsymbol{\pi} \circ (\sigma^{q_1}, \sigma^{\bar{q}_1})$, where $\sigma : G_1 \to G_0$ is an isomorphism ($q_1 \notin \{0,1\}$ is handled as $q_1 = 0$).

W3. W checks that $\boldsymbol{\psi} \in \mathrm{Sym}\, V \times \mathrm{Sym}\, V$ and that $\mathbf{H} = \boldsymbol{\psi}\, \mathrm{swap}(q_1, \mathbf{G})$. If this fails it rejects. W sends V the pair of permutations: $\boldsymbol{\psi}' = \mathrm{swap}(e, \boldsymbol{\pi}' \circ \boldsymbol{\psi})$.

V2. V checks that $\boldsymbol{\psi}' \in \mathrm{Sym}\, V \times \mathrm{Sym}\, V$ and that $\mathbf{H}' = \boldsymbol{\psi}'\, \mathrm{swap}(q, \mathbf{G})$. If this fails it halts and rejects.

V and W accept if they have completed successfully m iterations of the steps above.

(End of Protocol)

We shall now show that the warden's strategy of swapping graphs and permutations makes the prover and verifier untraceable, whilst the warden is transparent.

Theorem 1. *Protocol 2 is a statistically divertible interactive proof for Graph Isomorphism which is perfectly zero-knowledge and sabotage-free.*

Proof. When $x \in L$ and P and W are honest then for any V',

$$\boldsymbol{\psi}\, \mathrm{swap}(q_1, \mathbf{G}) = \boldsymbol{\pi}(\sigma^{q_1} G_{q_1}, \sigma^{\bar{q}_1} G_{\bar{q}_1}) = \boldsymbol{\pi}(G_0, G_0) = \mathbf{H} \ . \tag{4}$$

So the warden will accept and the protocol is sabotage-free. Furthermore, if P, W and V are honest,

$$\begin{aligned} \boldsymbol{\psi}'\, \mathrm{swap}(q, \mathbf{G}) &= \mathrm{swap}(e, \boldsymbol{\pi}' \circ \boldsymbol{\psi})\, \mathrm{swap}(q, \mathbf{G}) = \mathrm{swap}(e, (\boldsymbol{\pi}' \circ \boldsymbol{\psi})\, \mathrm{swap}(e \oplus q, \mathbf{G})) \\ &= \mathrm{swap}(e, \boldsymbol{\pi}'(\boldsymbol{\psi}\, \mathrm{swap}(e \oplus q, \mathbf{G}))) = \mathrm{swap}(e, \boldsymbol{\pi}'\mathbf{H}) = \mathbf{H}' \ , \end{aligned}$$

by using the swap conditions (3) and (4). So the protocol is complete. Soundness for V and W is reduced to that of the two party protocol $({}^{P'\leftrightarrow}W', V)$, by taking P', W' as one machine. Then we use the soundness proof in [15, p.703]. Zero-knowledge follows from Lemma 1.

To prove that the proof is divertible we must show that it is a $(P, W^{\leftrightarrow V})$ – proof, and that we have transparency and untraceability. Completeness and weak soundness for W follow immediately from our earlier discussion since when W and V are honest, W accepts if and only if V accepts. Transparency is obvious. Let us first discuss untraceability informally. Consider the proof (P', W, V') where P' is a prover accepted by the honest verifier with input $x \in L$, *i.e.*, G_0 and G_1 are isomorphic. Then we must have $\mathbf{H} = \boldsymbol{\psi}\, \mathrm{swap}(q_1, \mathbf{G})$ with overwhelming probability, because the completeness condition of interactive proofs allows for a small probability of error. Suppose that this is the case. Then $\mathbf{H}$ is a pair of graphs, not necessarily random, which are both isomorphic to G_0. Because the permutations $\boldsymbol{\pi}'$ are uniformly distributed, the graphs $\mathbf{H}' = \mathrm{swap}(e, \boldsymbol{\pi}'\mathbf{H})$ in Step W1 are uniform (isomorphic to G_0 and G_1), and the permutations $\boldsymbol{\psi}' = \mathrm{swap}(e, \boldsymbol{\pi}' \circ \boldsymbol{\psi})$ in Step W3 are uniform. So q and e are independent and therefore the bits $q_1 = q \oplus e$ in Step W2 are uniform. Let $r_{P'}$ be the portion of the random tape that P' reads and $r_{V'}$ be the portion of the random tape that V' reads. Then, $(r_{P'}, q_1)$ occurs with the same probability as when the verifier is V, and $(r_{V'}, \mathbf{H}', \boldsymbol{\psi}')$ occurs with (almost) the same probability as when the prover is P. By taking into account the fact that there is a small probability of error, we see that the view of P' in $(P', W^{\leftrightarrow V'})$ is statistically indistinguishable from the view of P' in (P', V). Similarly the view of V' in $({}^{P'\leftrightarrow}W, V')$ is statistically indistinguishable from that in (P, V'). We shall prove this formally for the view of V'. The other case is similar.

Let $x \in L$. Define $z' = 0$ to be the event that for all iterations in (P', W, V'), $\mathbf{H} = \boldsymbol{\psi}\, \mathrm{swap}(q_1, \mathbf{G})$, and $z' = 1$ the event that this is not so. Also let $z = 0$ and $z = 1$ be the corresponding events for (P, W, V'). Then clearly $\mathrm{Prob}(z = 0) = 1$ (P is honest) and one can prove that $\mathrm{Prob}(z' = 0) > 1 - |x|^{-k}$, from the completeness of (P', W, V). Let $v' = ({}^{P'\leftrightarrow}W, \underline{V'})(x, \mathbf{h})$ be the view of V' when interacting with P' and $v = (P, \underline{V'})(x, \mathbf{h})$ be the view of V' when interacting with P. We have $\mathrm{Prob}(v' = \alpha) = \mathrm{Prob}(v' = \alpha, z' = 0) + \mathrm{Prob}(v' = \alpha, z' = 1)$, and $\mathrm{Prob}(v = \alpha) = \mathrm{Prob}(v = \alpha, z = 0)$. From our earlier discussion on the uniformity of $\mathbf{H}'$ and $\boldsymbol{\psi}'$, $\text{Prob}(v' = \alpha \mid z' = 0) = \text{Prob}(v = \alpha \mid z = 0)$. Using these facts, and the fact that $\text{Prob}(a, b) = \text{Prob}(a \mid b) \cdot \text{Prob}(b)$ for events a, b, we see that

$$
\begin{aligned}
&\sum_{\alpha \in \{0,1\}^*} |\text{Prob}(v' = \alpha) - \text{Prob}(v = \alpha)| \\
&\quad \leq \sum_{\alpha \in \{0,1\}^*} |\text{Prob}(v' = \alpha, z' = 0) - \text{Prob}(v = \alpha, z = 0)| \; + \sum_{\alpha \in \{0,1\}^*} \text{Prob}(v' = \alpha, z' = 1)\,, \\
&\quad = \sum_{\alpha \in \{0,1\}^*} \text{Prob}(v = \alpha \mid z = 0)\, |\text{Prob}(z' = 0) - \text{Prob}(z = 0)| \; + \; \text{Prob}(z' = 1)\,, \\
&\quad \leq \; |x|^{-k} \sum_{\alpha \in \{0,1\}^*} \text{Prob}(v = \alpha \mid z = 0) \; + \; |x|^{-k} \; \leq \; 2|x|^{-k}\,,
\end{aligned}
$$

and this is true for any $k > 0$. So we have statistical untraceability. ■

Remark 4. The weaker form mentioned in Remark 1 would have given us perfect divertibility in Theorem 1.

We will now show that Protocol 2 can be modified to get a subliminal-free proof for Graph Isomorphism [6].

Protocol 3. (A subliminal-free zero-knowledge proof for GI)

Common input: A pair of graphs $\mathbf{G} = (G_0, G_1)$ with vertex set V and m edges.

W sets $w_d = 0$. If G_0, G_1 are not proper descriptions of graphs which have the same number of vertices and edges, W sets $w_d = 1$. Otherwise the seven steps of Protocol 2 are repeated m times independently, with Steps W2, W3 replaced by:

- W2$'$. W sets $w_d = 1$ if $q \notin \{0,1\}$. W sends P: $q_1 = q \oplus e$ ($q \notin \{0,1\}$ is handled as $q = 0$).
- W3$'$. W checks that $\boldsymbol{\psi} \in \text{Sym}\, V \times \text{Sym}\, V$ and that $\mathbf{H} = \boldsymbol{\psi}\, \text{swap}(q_1, \mathbf{G})$. If this fails it sets $w_d = 1$. W sends V the pair of permutations: $\boldsymbol{\psi}' = \text{swap}(e, \boldsymbol{\pi}' \circ \boldsymbol{\psi})$.

If V has completed successfully m iterations of the steps above then it accepts. W sets $w_d = 1$ if P or V halt prematurely. **(End of Protocol)**

Theorem 2. *Protocol 3 is a subliminal-free interactive proof for Graph Isomorphism which is perfectly zero-knowledge. The detectability is conditionally perfect.*

Proof. Clearly the modified protocol remains perfectly zero-knowledge. Fairness follows trivially from the completeness of (P, W, V). The proof of detectability is similar to that for untraceability in Theorem 1. Consider (P', W, V') when $w_d = 0$ and $x \in L$. From the check in Step W3, $\mathbf{H} = \boldsymbol{\psi}\, \text{swap}(q_1, \mathbf{G})$, so the graphs of $\mathbf{H}$ are isomorphic to G_0. Again, because the permutations $\boldsymbol{\pi}'$ are uniformly distributed, the graphs $\mathbf{H}' = \text{swap}(e, \boldsymbol{\pi}'\mathbf{H})$ are uniform and the permutations $\boldsymbol{\psi}' = \text{swap}(e, \boldsymbol{\pi}' \circ \boldsymbol{\psi})$ are uniform. Furthermore q and e are independent and q_1 is uniform. Thus the conditional ($w_d = 0$) view of P' when interacting with V': $(\cdots (r_{P'}, q_1) \cdots)$, is identical to that when interacting with V. Similarly the conditional view of V' when interacting with P': $(\cdots (r_{V'}, \mathbf{H}', \boldsymbol{\psi}') \cdots)$, is identical to that when interacting with P. So we have detectability. The system is minimal because the view of the honest prover P is simulatable. Therefore the proof is subliminal-free. ■

Remark 5. By extending the argument used in the last part of the proof we see that (P, W, V) is not sound for W. Indeed, suppose that G_0 is not isomorphic to G_1 and that P' chooses $\mathbf{H} = \text{swap}(d, \boldsymbol{\pi}\mathbf{G})$ in Step P1, where d is a random bit and $\boldsymbol{\pi}$ is a random pair of permutations of the vertex set. Then if the verifier V' has unlimited resources it can find $d \oplus e$ by checking which one of the two graphs of $\mathbf{H}' = \text{swap}(e, \boldsymbol{\pi}'\mathbf{H})$ is isomorphic to G_0. Consequently, if V' sends $q = d \oplus e$ in Step V1 and P' takes $\boldsymbol{\psi} = \text{swap}(d, \boldsymbol{\pi})$ in Step P2, we have $q_1 = d$ and the warden will accept since $\mathbf{H} = \boldsymbol{\psi}\, \text{swap}(q_1, \mathbf{G})$ as follows from (2). So, the warden may accept when $x \notin L$.

4 Graph Hamiltonicity

In this section we present a divertible and a subliminal-free zero-knowledge proof for any language in NP. A protocol for SAT was sketched in [6], but here we give a protocol for Hamilton Cycles which is easier to explain [20].

Our protocol employs a homomorphic commitment function f. The commitment is unconditional (lying is not possible) but privacy (hiding) is only conditional. As is typical with such protocols [14], the zero-knowledge simulation involves commitments to illegal values which are hidden and cannot be distinguished from random commitments by a polynomial-time verifier V'. However, if f has a trapdoor (as in the case of the commitment in Section 2.6), then a dishonest prover P' can write this on the history tape of a verifier V'. In this case V' will distinguish its actual view (in which legal values are hidden) from the simulated view, and we lose zero-knowledge. This must be prevented. We do this by having an oracle select independently, and uniformly, the parameters par $=$ par(t) of the commitment $f_{\text{par}} \in F_n$ for each execution of the protocol. The oracle is not needed if there exist homomorphic commitments f with no trapdoor (f is then part of P, W and V as in [14, p.713]). For convenience we represent the commitments in F_n by f (we drop the subscript par) and assume that they require v coin tosses.

Protocol 4. (A divertible zero-knowledge proof for HC)

Common input: A graph $G = (V, E)$ with vertex set V, edge set E, $n = |V|$, $m = |E|$.

An oracle selects the parameters of a homomorphic commitment f randomly in F_n and gives them to P, W, V. Then W sets[12] $w_a = 0$. W, V reject and W sets $w_a = 1$ if G is not a proper description of a graph. Otherwise the following seven steps are executed m times, independently.

P1. P selects $\boldsymbol{\pi} \in_R \operatorname{Sym} V \times \operatorname{Sym} V$ and coin tosses $(r_{ij}^0, r_{ij}^1) \in_R \{0,1\}^v \times \{0,1\}^v$, and commits to the pair of adjacency matrices $\mathbf{A} = (\{a_{ij}^0\}, \{a_{ij}^1\})$ of the graphs $\mathbf{G} = \boldsymbol{\pi}(G, G)$ by using the homomorphic commitment f. Let $(b_{ij}^0, b_{ij}^1) = (f(a_{ij}^0, r_{ij}^0), f(a_{ij}^1, r_{ij}^1))$. P sends W the pair of matrices $\mathbf{B} = (\{b_{ij}^0\}, \{b_{ij}^1\})$.

W1. W selects random coin tosses $(s_{ij}^0, s_{ij}^1) \in_R \{0,1\}^v \times \{0,1\}^v$ and computes the pair of matrices $\mathbf{C} = (\{f(0, s_{ij}^0) \cdot b_{ij}^0\}, \{f(0, s_{ij}^1) \cdot b_{ij}^1\})$. Then W selects a bit $e \in_R \{0,1\}$, the pair of permutations $\boldsymbol{\pi}' \in_R \operatorname{Sym} V \times \operatorname{Sym} V$, and sends V the pair of matrices $\mathbf{D} = (\{d_{ij}^0\}, \{d_{ij}^1\}) = \operatorname{swap}(e, \boldsymbol{\pi}'\mathbf{C})$ ($\boldsymbol{\pi}'$ permutes the corresponding rows and columns of the matrices in $\mathbf{C}$).

V1. V sends W a bit $q \in_R \{0,1\}$ as a challenge.

W2. W sends P the bit $q_1 = q \oplus e$ as a challenge; $q \notin \{0,1\}$ is handled as $q = 0$.

P2. P sends W: the permutation π_{q_1}, all the coin tosses $r_{ij}^{q_1}$, a Hamilton cycle $H_{\bar{q}_1}$ in $G_{\bar{q}_1}$, and the n coin tosses $r_{i_1 i_2}^{\bar{q}_1}, \ldots, r_{i_n i_1}^{\bar{q}_1}$ used in its commitment ($q_1 \notin \{0,1\}$ is handled as $q_1 = 0$).

W3. W checks that B_{q_1} is a commitment of the adjacency matrix of $\pi_{q_1} G$ and that $b_{i_1 i_2}^{\bar{q}_1} = f(1, r_{i_1 i_2}^{\bar{q}_1})$, $\ldots, b_{i_n i_1}^{\bar{q}_1} = f(1, r_{i_n i_1}^{\bar{q}_1})$. If either fails then it sets $w_a = 1$ and sends V a string of zeros. Otherwise it computes for all r_{ij}^k received from P, the coin tosses u_{ij}^k such that $f(a_{ij}^k, u_{ij}^k) = f(0, s_{ij}^k) \cdot f(a_{ij}^k, r_{ij}^k)$, and then W sends V: $\pi_q'' = \pi_{q_1}' \circ \pi_{q_1}$, all the coin tosses $t_{ij}^q = u_{\pi_{q_1}'(i)\pi_{q_1}'(j)}^{q_1}$, the cycle $H_{\bar{q}}' = \pi_{\bar{q}_1}' H_{\bar{q}_1}$ and the n coin tosses $t_{i_1 i_2}^{\bar{q}} = u_{\pi_{\bar{q}_1}'(i_1)\pi_{\bar{q}_1}'(i_2)}^{\bar{q}_1}, \ldots, t_{i_n i_1}^{\bar{q}} = u_{\pi_{\bar{q}_1}'(i_n)\pi_{\bar{q}_1}'(i_1)}^{\bar{q}_1}$ used for its commitment.

[12] The Boolean variable w_a is not required for divertibility. It will be needed for the subliminal-free proof.

V2. V checks that D_q is a commitment of the adjacency matrix of $\pi''_q G$ and that $d^{\bar{q}}_{i_1 i_2} = f(1, t^{\bar{q}}_{i_1 i_2}), \ \ldots, d^{\bar{q}}_{i_n i_1} = f(1, t^{\bar{q}}_{i_n i_1})$. If either fails, V halts and rejects.

If V has completed successfully m iterations of the steps above then it accepts. W sets $w_a = 1$ if P halts prematurely. W accepts if $w_a = 0$, otherwise it rejects.

(End of Protocol)

Observe that W can compute u^k_{ij} from a^k_{ij}, r^k_{ij}, and s^k_{ij} in polynomial-time since f is a homomorphic commitment.

Theorem 3. *If there exist secure homomorphic commitments f, then Protocol 4 is a statistically divertible interactive proof for graph Hamiltonicity which is computationally zero-knowledge and sabotage-free, provided that the parameters of f are selected randomly in F_n by an oracle.*

Proof. The proof is an extension of that in [4] and is based on the one in [15]. If P is honest then B_{q_1} is a commitment of the adjacency matrix of $\pi_{q_1} G$ and $H_{\bar{q}_1}$ is a Hamilton cycle in $G_{\bar{q}_1}$, so W accepts and the protocol is sabotage-free for W. If P and W are honest then D_q is the encryption of $\pi''_q G$ and $H'_{\bar{q}}$ is a Hamilton cycle in $\pi_{\bar{q}}{}'' G$, and so V accepts and we get completeness. The proof for soundness is the same as for the two-party protocol (P, V) in [4].

It follows that $(P, W^{\leftrightarrow V})$ is a proof. Clearly we have transparency. For untraceability we only need to consider provers P' who are accepted by the honest verifier V. Then both B_0, B_1 are almost always commitments for adjacency matrices of G. This implies that almost always, (i) the d^k_{ij} are proper commitments with the appropriate distributions and so, (ii) the q and e are uncorrelated and hence $q_1 = q \oplus e$ is uniform, because $e \in_R \{0, 1\}$ [22]. Therefore P' cannot distinguish between the challenges it gets directly from an honest V and those it gets from V' through W, and similarly V' cannot distinguish between the bit strings it would get directly from P and those it gets from P' through W. As in Theorem 1 we have statistical indistinguishability. The proof for zero-knowledge is similar to that for the two party protocol (P, V) [4]. A formal proof for zero-knowledge is obtained by extending the argument used in [15, Theorem 2, pp. 716-721]. ■

We will now show that Protocol 4 can be modified to get a subliminal-free proof for Hamilton Cycles.

Theorem 4. *If there exist secure homomorphic commitments f, then Protocol 4 can be modified to obtain a subliminal-free proof for graph Hamiltonicity, provided that the parameters of f are selected randomly in F_n by an oracle. The proof is computationally zero-knowledge and detectability is conditionally statistical.*

Proof. We modify the protocol by having W initialize with $w_d = 0$, and set $w_d = 1$ if $q \notin \{0, 1\}$ in Step W2. Then at the end of the proof, P proves to W in zero-knowledge the NP statement that all the pairs of matrices $\boldsymbol{B}$ are properly constructed (*e.g.,* by using one of the proofs in [14, 4]). That is, P proves to W using a zero-knowledge proof (P, W) that, pairs of permutations $\boldsymbol{\pi}$ were used to obtain the adjacency matrices of $\boldsymbol{G}$, and that the elements of the matrices $\boldsymbol{B}$ are proper encryptions (*i.e.,* there exist coin tosses r^k_{ij} such that $b^k_{ij} = f(a^k_{ij}, r^k_{ij})$). If P or V halt prematurely, W sets $w_a = 1$. If the (P, W) proof fails, or if $w_a = 1$, then W sets[13] $w_d = 1$.

Clearly the modified protocol remains computationally zero-knowledge. However it is not transparent. To show that it is subliminal-free we must show that it is fair, that we have detectability and that it is minimal knowledge. Fairness follows directly from the modified (P, W, V). We shall now prove that we have detectability for the case when the subliminal receiver is the verifier. The other case is similar.

Let us first sketch the outline of our proof. To begin with, in Part 1, we show that if the probability that the warden W detects an abuse is not overwhelming, then the conditional probability that a

[13] Note that it is not necessary that W accepts or rejects in the modified protocol.

dishonest prover P' uses proper encryptions and that W accepts, given that W fails to detect the abuse, is overwhelming. In Part 2 we use this to show that we have detectability. We now proceed with the proof.

We make three key observations. Let $z = 0$ be the event that $w_a = 0$ and that the matrices $\mathbf{B}$ *are* proper encryptions. Otherwise $z = 1$. The first observation is that when $x \in L$ (*i.e.,* the graph G has a Hamilton Cycle) the conditional probability that the verifier's view has a certain value, given $z = 0$, is the same whether the prover is honest or not. That is,

$$\text{Prob}_{(P',W,V')}(v' = \alpha \mid z = 0) = \text{Prob}_{(P,W,V')}(v = \alpha \mid z = 0) \ , \tag{5}$$

where $v' = ({}^{P'\leftrightarrow}W, \underline{V'})_{|w_d=0}(x, \mathbf{h})$ and $v = ({}^{P\leftrightarrow}W, \underline{V'})_{|w_d=0}(x, \mathbf{h})$. This follows from the untraceability of Protocol 4, because when $z = 0$ the encryptions are proper and $w_a = 0$. The second observation is that for any k, sufficiently long $x \in L$,

$$\text{Prob}_{(P',W,V')}(w_d = 1 \mid z = 1) \geq 1 - |x|^{-3k} \ . \tag{6}$$

This follows from the soundness of the (P, W) proof that all the matrices $\mathbf{B}$ are properly constructed (the reason why we take the exponent to be $3k$ will soon become clear). The third observation is that for any k, sufficiently long $x \in L$,

$$\text{Prob}_{(P',W,V')}(w_d = 0 \mid z = 0) \geq 1 - |x|^{-2k} \ , \quad \text{and} \tag{7}$$

$$\text{Prob}_{(P',W,V')}(w_d = 0 \mid v' = \alpha, z = 0) \geq 1 - |x|^{-2k} \quad \text{(when defined)} \ . \tag{8}$$

This follows from the completeness of the (P, W) proof.

Part 1. Our first goal is to show that for any k, sufficiently long $x \in L$,

$$\max\left(\text{Prob}_{(P',W,V')}\left(w_d = 1\right), \text{Prob}_{(P',W,V')}\left(z = 0 \mid w_d = 0\right)\right) > 1 - |x|^{-k} \ . \tag{9}$$

Indeed, suppose that $\text{Prob}\left(w_d = 1\right) \leq 1 - |x|^{-k}$ so that $\text{Prob}\left(w_d = 0\right) \geq |x|^{-k}$ (to avoid cumbersome notation we drop the subscripts when there is no ambiguity). Let $A = \text{Prob}(w_d = 0 \mid z = 0)$, $B = \text{Prob}(w_d = 0 \mid z = 1)$. Then by (6) we have $B < |x|^{-3k}$, so that by (7), $1 - |x|^{-2k} - |x|^{-3k} < A - B < 1$, and therefore, $1 < (A - B)^{-1} < 1 + 2|x|^{-2k}$. Also, $\text{Prob}(w_d = 0) = A \cdot \text{Prob}(z = 0) + B \cdot (1 - \text{Prob}(z = 0)) = (A - B) \cdot \text{Prob}(z = 0) + B$, so that $\text{Prob}(z = 0)/\,\text{Prob}(w_d = 0) = (A - B)^{-1}(1 - B/\,\text{Prob}(w_d = 0))$, and therefore

$$1 - |x|^{-2k} \ < \ F = \frac{\text{Prob}_{(P',W,V')}(z = 0)}{\text{Prob}_{(P',W,V')}(w_d = 0)} \ < \ 1 + 2|x|^{-2k} \ , \tag{10}$$

since $1 - |x|^{-2k} < 1 - B/\,\text{Prob}(w_d = 0) < 1$. Then by Bayes' law,

$$\begin{aligned} \text{Prob}_{(P',W,V')}(z = 0 \mid w_d = 0) &= \text{Prob}_{(P',W,V')}(w_d = 0 \mid z = 0) \cdot F \\ &> (1 - |x|^{-2k}) \cdot (1 - |x|^{-2k}) > 1 - |x|^{-3k/2} \ , \end{aligned} \tag{11}$$

using (7). This proves (9).

Part 2. For conditionally statistical detectability we must show that for any k, sufficiently long $x \in L$, if $\text{Prob}_{(P',W,V')}(w_d = 0) \geq |x|^{-k}$, then

$$\sum_{\alpha \in \{0,1\}^*} |\,\text{Prob}_{(P',W,V')}(v' = \alpha \mid w_d = 0) - \text{Prob}_{(P,W,V')}(v = \alpha \mid w_d = 0)| \ < \ |x|^{-k} \ . \tag{12}$$

Let us expand the left hand side of this expression by splitting the views into those for which $z = 0$ and those for which $z = 1$. Since $z = 0$ when (P, W, V') is executed, we get,

$$\sum_{\alpha \in \{0,1\}^*} |\text{Prob}(v' = \alpha, z = 0 \mid w_d = 0) + \text{Prob}(v' = \alpha, z = 1 \mid w_d = 0) - \text{Prob}(v = \alpha, z = 0 \mid w_d = 0)|$$

$$\leq \sum_{\alpha \in \{0,1\}^*} |\mathrm{Prob}(v' = \alpha, z = 0 \mid w_d = 0) - \mathrm{Prob}(v = \alpha, z = 0 \mid w_d = 0)| \tag{13}$$

$$+ \sum_{\alpha \in \{0,1\}^*} \mathrm{Prob}(v' = \alpha, z = 1 \mid w_d = 0)\,. \tag{14}$$

Now the sum in (14) is $\mathrm{Prob}(z = 1 \mid w_d = 0)$. By (11) this is less or equal to $|x|^{-3k/2}$, if $\mathrm{Prob}_{(P',W,V')}(w_d = 0) \geq |x|^{-k}$. So we only need to focus on the rest of the sum, that is (13). We shall show that the probabilities in this sum are statistically close. First observe that

$$\mathrm{Prob}\,(v' = \alpha, z = 0 \mid w_d = 0) \quad =$$
$$\mathrm{Prob}(w_d = 0 \mid v' = \alpha, z = 0) \cdot \mathrm{Prob}(v' = \alpha \mid z = 0) \cdot \frac{\mathrm{Prob}_{(P',W,V')}(z = 0)}{\mathrm{Prob}_{(P',W,V')}(w_d = 0)}\ , \tag{15}$$

and that $\mathrm{Prob}(v' = \alpha \mid z = 0) = \mathrm{Prob}(v = \alpha \mid z = 0)$ by (5). Then by (10) and (8),

$$1 - |x|^{-3k/2} \quad < \quad \frac{\mathrm{Prob}_{(P',W,V')}(v' = \alpha, z = 0 \mid w_d = 0)}{\mathrm{Prob}_{(P,W,V')}(v = \alpha \mid z = 0)} \quad < \quad 1 + 2|x|^{-2k}\,. \tag{16}$$

Next we consider $\mathrm{Prob}_{(P,W,V')}(v = \alpha, z = 0 \mid w_d = 0)$. For this we get a similar expansion to (15), but with v', P' replaced by v, P. That is,

$$1 - |x|^{-3k/2} \quad < \quad \frac{\mathrm{Prob}_{(P,W,V')}(v = \alpha, z = 0 \mid w_d = 0)}{\mathrm{Prob}_{(P,W,V')}(v = \alpha \mid z = 0)} \quad < \quad 1 + 2|x|^{-2k}\,. \tag{17}$$

Combining (17) and (16), we see that the sum in (13) is less than

$$(2|x|^{-2k} + |x|^{-3k/2}) \cdot \sum_{\alpha \in \{0,1\}^*} \mathrm{Prob}(v = \alpha \mid z = 0) \quad < \quad 2|x|^{-3k/2}\,.$$

Since we have already bounded (14) by $|x|^{-3k/2}$, we get (12). This completes the proof for detectability. The proof is minimal because the view of the honest prover can be simulated. We have shown that the modified protocol is fair, detectable and minimal. It follows that it is subliminal-free. ∎

Remark 6. We can use bit commitments with blindings instead of homomorphic bit commitments in Protocol 4. Theorem 3 and Theorem 4 will still hold, provided the parameters of the commitment scheme and its blinding are selected randomly by the oracle. The oracle is not needed if there exist commitments f with blindings for which there is no trapdoor.

5 Graph Non-Isomorphism

We show that Graph Non-Isomorphism has a subliminal-free zero-knowledge proof with no unproven assumption. It should be noted that Graph Non-Isomorphism is known to be in AM but is conjectured not to be in NP. Our protocol is based on the proof in [15, 20] and uses our swapping technique for Graph Isomorphism. In this protocol we will use expressions such as γ_i^k, τ_{ij}^k, etc.: in these the i, j, k are all indices.

Protocol 5. (A subliminal-free zero-knowledge proof for GNI)

Common input: A pair of graphs $\mathbf{G} = (G_0, G_1)$ with vertex set V, n vertices and m edges.

W sets $w_d = 0$. If $\mathbf{G}$ is not a proper description of graphs then W sets $w_d = 1$ and V rejects. If the number of vertices, or edges, are distinct, then V accepts. Otherwise, the following ten steps are executed m times, independently.

V1. V selects $\alpha \in_R \{0,1\}$, $\boldsymbol{\pi} \in_R \mathrm{Sym}\, V \times \mathrm{Sym}\, V$ and constructs the pair of graphs $\mathbf{H} = \boldsymbol{\pi}\, \mathrm{swap}(\alpha, \mathbf{G})$. Then V selects $\boldsymbol{\tau}_i^k \in_R \mathrm{Sym}\, V \times \mathrm{Sym}\, V$ and $\gamma_i^k \in_R \{0,1\}$, and constructs the graphs $\mathbf{T}_i^k = \boldsymbol{\tau}_i^k\, \mathrm{swap}(\gamma_i^k, \mathbf{G})$, $i = 1, \ldots, n^2$, $k = 0, 1$, and sends W the pairs: $\mathbf{H}$ and $\{(\mathbf{T}_i^0, \mathbf{T}_i^1)\}$.

W1. W proceeds similarly. It selects $a \in_R \{0,1\}$, $\boldsymbol{\phi} \in_R \mathrm{Sym}\, V \times \mathrm{Sym}\, V$ and constructs the pair of graphs $\mathbf{I} = \boldsymbol{\phi}\, \mathrm{swap}(a, \mathbf{H})$. Then W selects $\boldsymbol{\sigma}_i^k \in_R \mathrm{Sym}\, V \times \mathrm{Sym}\, V$ and $c_i^k \in_R \{0,1\}$. In addition, it selects $e_i \in_R \{0,1\}$ and swaps to obtain the pairs of graphs $\mathbf{S}_i^k = \boldsymbol{\sigma}_i^{k \oplus e_i}\, \mathrm{swap}(c_i^{k \oplus e_i}, \mathbf{T}_i^{k \oplus e_i})$, $i = 1, \ldots, n^2$, $k = 0, 1$, and sends P the pairs: $\mathbf{I}$ and $\{(\mathbf{S}_i^0, \mathbf{S}_i^1)\}$.

P1. P selects bits $q_i \in_R \{0,1\}$, $i = 1, \ldots, n^2$, and sends W the string $\{q_i\}$.

W2. W sends V the bits q_i', where $q_i' = q_i \oplus e_i$, $i = 1, \ldots, n^2$. If $q_i \notin \{0,1\}$ then W sets $w_d = 1$.

V2. V sends W: $\{(\gamma_i^{q_i'}, \boldsymbol{\tau}_i^{q_i'})\}$ and $\{(s_i^{\bar{q}_i'}, \boldsymbol{\mu}_i^{\bar{q}_i'})\}$, where $s_i^{\bar{q}_i'} = \alpha \oplus \gamma_i^{\bar{q}_i'}$ and $\boldsymbol{\mu}_i^{\bar{q}_i'} = \boldsymbol{\tau}_i^{\bar{q}_i'} \circ \mathrm{swap}(\alpha \oplus \gamma_i^{\bar{q}_i'}, \boldsymbol{\pi}^{-1})$.[14]

W3. W checks for each $i = 1, \ldots, n^2$, that the $\boldsymbol{\tau}_i^{q_i'}$ are isomorphisms from $\mathrm{swap}(\gamma_i^{q_i'}, \mathbf{G})$ to $\mathbf{T}_i^{q_i'}$, and that the $\boldsymbol{\mu}_i^{\bar{q}_i'}$ are isomorphisms from $\mathrm{swap}(s_i^{\bar{q}_i'}, \mathbf{H})$ to $\mathbf{T}_i^{\bar{q}_i'}$. If either fails it sets $w_d = 1$. Otherwise it sends P: $\{(t_i^{q_i}, \boldsymbol{\nu}_i^{q_i})\}$, where $t_i^{q_i} = \gamma_i^{q_i'} \oplus c_i^{q_i'}$, $\boldsymbol{\nu}_i^{q_i} = \boldsymbol{\sigma}_i^{q_i'} \circ \mathrm{swap}(c_i^{q_i'}, \boldsymbol{\tau}_i^{q_i'})$, and $\{(t_i^{\bar{q}_i}, \boldsymbol{\nu}_i^{\bar{q}_i})\}$, where $t_i^{\bar{q}_i} = s_i^{\bar{q}_i'} \oplus a \oplus c_i^{\bar{q}_i'}$, $\boldsymbol{\nu}_i^{\bar{q}_i} = \boldsymbol{\sigma}_i^{\bar{q}_i'} \circ \mathrm{swap}(c_i^{\bar{q}_i'}, \boldsymbol{\mu}_i^{\bar{q}_i'}) \circ \mathrm{swap}(s_i^{\bar{q}_i'} \oplus a \oplus c_i^{\bar{q}_i'}, \boldsymbol{\phi}^{-1})$.

P2. P checks that for each $i = 1, \ldots, n^2$, that the $\boldsymbol{\nu}_i^{q_i}$ are isomorphisms from $\mathrm{swap}(t_i^{q_i}, \mathbf{G})$ to $\mathbf{S}_i^{q_i}$ and that the $\boldsymbol{\nu}_i^{\bar{q}_i}$ are isomorphisms from $\mathrm{swap}(t_i^{\bar{q}_i}, \mathbf{I})$ to $\mathbf{S}_i^{\bar{q}_i}$. If either fails, P sets $w_d = 1$. Otherwise it computes $\beta \in \{0,1\}$ such that the graphs $\boldsymbol{I}$ and $\mathrm{swap}(\beta, \boldsymbol{G})$ are isomorphic, and sends this to W. If no such β exists, it halts.

W4. If $\beta \notin \{0,1\}$ then W sets $w_d = 1$. Otherwise it sends V: $\beta' = a \oplus \beta$.

V3. V checks that $\alpha = \beta'$. If this fails it halts and rejects. Otherwise V sends W the permutations $\boldsymbol{\pi}$.

W5. W checks that $\mathbf{H} = \boldsymbol{\pi}\, \mathrm{swap}(\beta', \mathbf{G})$. If this fails it sets $w_d = 1$.

If V has completed successfully m rounds it accepts. W sets $w_d = 1$ if P or V halt prematurely. **(End of Protocol)**

Theorem 5. *Protocol 5 is a subliminal-free proof for Graph Non-Isomorphism which is perfectly zero-knowledge. The detectability is conditionally perfect.*

Proof. For fairness we use the swap conditions (2) and (3). The checks in Step W3 are valid when P, W, V are honest since,

$$\begin{aligned} \boldsymbol{\mu}_i^{\bar{q}_i'}\, \mathrm{swap}(s_i^{\bar{q}_i'}, \mathbf{H}) &= (\boldsymbol{\tau}_i^{\bar{q}_i'} \circ \mathrm{swap}(s_i^{\bar{q}_i'}, \boldsymbol{\pi}^{-1}))\, \mathrm{swap}(s_i^{\bar{q}_i'}, \mathbf{H}) = \boldsymbol{\tau}_i^{\bar{q}_i'}\, \mathrm{swap}(s_i^{\bar{q}_i'}, \boldsymbol{\pi}^{-1}\mathbf{H}) \\ &= \boldsymbol{\tau}_i^{\bar{q}_i'}\, \mathrm{swap}(s_i^{\bar{q}_i'}, \mathrm{swap}(\alpha, \mathbf{G})) = \boldsymbol{\tau}_i^{\bar{q}_i'}\, \mathrm{swap}(s_i^{\bar{q}_i'} \oplus \alpha, \mathbf{G}) \\ &= \boldsymbol{\tau}_i^{\bar{q}_i'}\, \mathrm{swap}(\gamma_i^{\bar{q}_i'}, \mathbf{G}) = \mathbf{T}_i^{\bar{q}_i'}. \end{aligned} \tag{18}$$

[14] In this protocol for each i the verifier sends four permutations, whereas in [15] either one or two permutations are sent. This is made possible by the 'doubling' of the original protocol.

For completeness we use the swap condition (3). The checks in Step P2 are valid since,

$$\boldsymbol{\nu}_i^{q_i}\,\mathrm{swap}(t_i^{q_i},\mathbf{G}) = (\boldsymbol{\sigma}_i^{q_i'}\circ\mathrm{swap}(c_i^{q_i'},\boldsymbol{\tau}_i^{q_i'}))\,\mathrm{swap}(t_i^{q_i},\mathbf{G}) = \boldsymbol{\sigma}_i^{q_i'}\,\mathrm{swap}(c_i^{q_i'},\mathbf{T}_i^{q_i'}) = \mathbf{S}_i^{q_i'\oplus e_i} = \mathbf{S}_i^{q_i},$$

and

$$\begin{aligned}
\boldsymbol{\nu}_i^{\bar{q}_i}\,\mathrm{swap}(t_i^{\bar{q}_i},\mathbf{I}) &= (\boldsymbol{\sigma}_i^{\bar{q}_i'}\circ\mathrm{swap}(c_i^{\bar{q}_i'},\boldsymbol{\mu}_i^{\bar{q}_i'})\circ\mathrm{swap}(t_i^{\bar{q}_i},\boldsymbol{\phi}^{-1}))\,\mathrm{swap}(t_i^{\bar{q}_i},\mathbf{I})\\
&= (\boldsymbol{\sigma}_i^{\bar{q}_i'}\circ\mathrm{swap}(c_i^{\bar{q}_i'},\boldsymbol{\mu}_i^{\bar{q}_i'}))\,\mathrm{swap}(t_i^{\bar{q}_i},\mathrm{swap}(a,\mathbf{H}))\\
&= (\boldsymbol{\sigma}_i^{\bar{q}_i'}\circ\mathrm{swap}(c_i^{\bar{q}_i'},\boldsymbol{\mu}_i^{\bar{q}_i'}))\,\mathrm{swap}(c_i^{\bar{q}_i'}\oplus s_i^{\bar{q}_i'},\mathbf{H})\\
&= \boldsymbol{\sigma}_i^{\bar{q}_i'}\,\mathrm{swap}(c_i^{\bar{q}_i'},\boldsymbol{\mu}_i^{\bar{q}_i'}\,\mathrm{swap}(s_i^{\bar{q}_i'},\mathbf{H})) = \mathbf{S}_i^{\bar{q}_i'\oplus e_i} = \mathbf{S}_i^{\bar{q}_i},
\end{aligned}$$

using (18) in the last line. The check in Step V3 is obviously valid. So the protocol is complete for V. Soundness for V is reduced to that of the two party protocol $({}^{P'\leftrightarrow}W', V)$ by taking P', W' as one machine, using the proof in [15, p.708]. We get zero-knowledge by extending the argument in [15, pp.709-711].

We will now show that the protocol is subliminal-free. The proof of detectability is as in Theorem 1. Consider (P', W, V') when $w_d = 0$ and $x \in L$. Then from Step W5, $\mathbf{H} = \boldsymbol{\pi}\,\mathrm{swap}(\alpha, \mathbf{G})$. So the graphs which W sends to P' in Step W1 are uniform random swaps of pairs of uniform random graphs isomorphic to $\mathbf{G}$, and the bits that W sends in Step W2 occur with the same probability as when V' is V. Also, the bits, permutations and graphs which W sends in Step W3 and Step W4 occur with the same probability as in the case (P, W, V). So the conditional $(w_d = 0)$ view of P' when interacting with V' is identical to that when interacting with V. Similarly for the conditional view of V'. Clearly the proof is minimal. So it is subliminal-free. ■

Remark 7. This proof is not divertible[15] because the warden W cannot produce the appropriate distribution $(\{(\underline{P'}, V)(x, \mathbf{h})\})$ when a dishonest verifier V' halts in $(P', W^{\leftrightarrow V'})$. Observe that this proof is not sabotage-free. Furthermore it cannot be sound for W.

Remark 8. It is easy to see that Protocol 5 can be applied to quadratic non-residuocity (QNR), and to non-membership of a language $L = \langle a \rangle_p$, where p is a prime and $a \in Z_p^*$ (however both languages are in NP).

6 A subliminal-free zero-knowledge proof for languages in IP

In this section we consider languages in IP, the class of languages which have interactive proofs. We will show that if there exist secure homomorphic bit commitments then any language $L \in$ IP has a subliminal-free interactive zero-knowledge proof. Our proof is based on a system proposed in [19] which uses as a building block an Arthur-Merlin proof [2] for L (not necessarily zero-knowledge). Arthur-Merlin proofs, denoted by A-M, are interactive proofs in which the verifier, Arthur, is allowed only to send his coin tosses to the prover, Merlin. So if $q_1, q_2, \ldots, q_\ell$ are the ℓ strings which Arthur sends to Merlin during the execution of the protocol, then the concatenation $q_1 q_2 \cdots q_\ell = q$ must be the string that Arthur reads from his random tape.

Lemma 2. (Impagliazzo-Yung [19]) *If there exist secure bit commitments then any language in* IP *has an interactive proof which is computationally zero-knowledge.*

Proof. Goldwasser–Sipser [18] have shown that any language $L \in$ IP has an Arthur-Merlin proof A-M, which is not necessarily zero-knowledge. Suppose that in this proof, on input $x \in L$, Arthur sends

[15] A variant of this protocol was presented in [20]. However a different definition of divertibility was used there.

$q_1, q_2, \ldots, q_\ell$ and Merlin sends $y_1, y_2, \ldots, y_\ell$. Let s_i be the concatenation of $q_1, y_1, q_2, y_2, \ldots, q_i, y_i$, with $s_0 = \varepsilon$ the empty string, and let $M(s_i, q_{i+1}) = y_{i+1}$ mean that Merlin, on input (s_i, q_{i+1}), produces the next message y_{i+1}. Furthermore in the last round, let $A(x, s_\ell) = 1$ or 0, mean that Arthur accepts or rejects Merlin's proof.

To prove the lemma we will describe the Impagliazzo-Yung protocol (P, V) for an interactive zero-knowledge proof for L in which the prover P and verifier V emulate the protocol A-M. P and V will jointly compute the coin tosses $q = q_1 q_2 \cdots q_\ell$ of Arthur, and then P will prove to V, in zero-knowledge, that Arthur would have accepted Merlin's proof in A-M, had Arthur's messages been $q_1, q_2, \ldots, q_\ell$.

Let us explain this in more detail. For convenience we assume that the length of the messages of Arthur and Merlin are v (which must be polynomially bounded in the length of the input), and that f is a bit commitment function which requires v coin tosses.

To determine bit $q_{1,j}$, $1 \leq j \leq v$, of Arthur's first message q_1, P sends V a commitment (for which lying is unconditionally impossible) $f(q_{P,1,j}, r_{P,1,j})$, $r_{P,1,j} \in_R \{0,1\}^v$, for a bit $q_{P,1,j}$, and V sends P a bit $q_{V,1,j}$, $1 \leq j \leq v$. Then P opens his commitment to V. The bit $q_{P,1,j} \oplus q_{V,1,j}$ is taken as the joint bit $q_{1,j}$, that is $q_{1,j} = q_{P,1,j} \oplus q_{V,1,j}$, and q_1 is the bit string $q_{1,1} \ldots q_{1,v}$. Then P commits to V the message y_1 of Merlin. That is, if $y_{1,1}, \ldots, y_{1,v}$ are the bits of y_1, P sends V the commitments $d_{1,j} = f(y_{1,j}, t_{1,j})$, $t_{1,j} \in_R \{0,1\}^v$, $1 \leq j \leq v$.

The procedure is repeated for q_2 and y_2. Then for q_3 and y_3, and so on, until q_ℓ and y_ℓ. Then P proves to V the NP statement:
$\exists\, y_1, \ldots, y_\ell, t_{1,1}, \ldots, t_{1,v}, t_{2,1}, \ldots, t_{\ell,v}$:

$$A(x, s_\ell) \quad \wedge \quad \left(\bigwedge_{\substack{1 \leq i \leq \ell \\ 1 \leq j \leq v}} (d_{i,j} = f(y_{i,j}, t_{i,j})) \right) \quad = \quad 1 , \tag{19}$$

where $s_\ell = q_1 y_1 \cdots q_\ell y_\ell$, $y_i = y_{i,1} \cdots y_{i,v}$, by using one of the zero-knowledge proofs in [14, 4].

Completeness follows immediately. For soundness observe that if a prover P' succeeds in convincing V that the predicate (19) is satisfied with a probability which is not negligible then, from the soundness of the zero-knowledge proof of the NP statement, it follows that Merlin would also succeed in convincing Arthur with a probability which is not negligible (by using the same protocol as P, but this time sending the y_i's instead of just committing to them). Since the Arthur-Merlin proof is sound we must have $x \in L$. Finally for zero-knowledge the simulator selects for each $1 \leq i \leq \ell$, $1 \leq j \leq v$, random bits $q_{P,i,j}$, for the prover and sends commitments of these to a blackbox simulation of the verifier V'. From this it gets bits $q_{V,i,j}$ and thus computes the bits $q_{i,j}$. By concatenating these it gets all the q_i. For each i, $1 \leq i \leq \ell$, the simulator selects $y'_{i,j} \in_R \{0,1\}$, $t'_{i,j} \in_R \{0,1\}^v$, and computes commitments $d'_{i,j} = f(y'_{i,j}, t'_{i,j})$, $1 \leq j \leq v$ (which are indistinguishable from the "proper" commitments $d_{i,j}$ for the bits of y_i). Then the simulator runs the simulation of the zero-knowledge proof for the predicate (19).[16] ■

We now consider a subliminal-free proof for IP. Our protocol uses homomorphic commitments. We remind the reader that, for convenience, the commitments in F_n are represented by f (we drop the subscript par).

Theorem 6. *If there exist secure homomorphic commitments f, then any language in* IP *has a subliminal-free proof which is sabotage-free, provided that the parameters of f are selected randomly in F_n by an oracle. The proof is computationally zero-knowledge and detectability is conditionally computational.*

Proof. We extend the argument used in the Impagliazzo–Yung proof to get a subliminal-free zero-knowledge proof (P, V). For simplicity we assume as in Lemma 2 that the messages of Arthur and Merlin and the coin tosses have length v.

[16] Although the predicate might not be satisfiable, this does not affect the total simulation.

The outline of our proof is as follows. To start with, in Subroutine 1, the prover P, the warden W, and the verifier V construct jointly the messages q_i of Arthur in the proof A-M. This must be done in such a way that P and V are prevented from sending each other subliminal messages either directly, or indirectly by halting prematurely when the message stream does not have a particular pattern. In the latter case, non-halting would leak information (halting can have irreparable consequences with cryptographic protocols [9]). For this purpose the warden blinds the commitments of the prover, and uncovers the blinding *only* at the very end of the entire protocol, when the protocol halts in any case. After each message q_i is computed, P commits to Merlin's reply y_i. Again the warden must blind the commitment. In Subroutine 2, P proves, by using a subliminal-free zero-knowledge proof that Arthur would have accepted Merlin's proof if the message stream were the one determined in Subroutine 1. Finally Subroutine 3, the warden unblinds his commitments and the verifier does all the necessary checks. The protocol is as follows.

Protocol 6. (A subliminal-free zero-knowledge proof for IP)
Common input: $x \in L$

An oracle selects randomly the parameters of a homomorphic commitment $f \in F_n$ and gives them to P, W, V. W sets $w'_d = 0$. Then the following subroutines are executed sequentially.

Subroutine 1. (Simulating the proof A-M: P, W, V commit to Arthur's messages q_i and then P commits to Merlin's replies y_i)

Set $i = 1$. The following steps are executed ℓ times, incrementing i by one each time.

P1. P selects bits $q_{P,i,j} \in_R \{0,1\}$ and coin tosses $r_{P,i,j} \in_R \{0,1\}^v$ and sends W the commitments $c_{P,i,j} = f(q_{P,i,j}, r_{P,i,j})$, $1 \leq j \leq v$.

W1. W checks that the $c_{P,i,j}$ are bit commitments, using[17] P if necessary, and sets $w'_d = 1$ if this is not the case. Then W selects bits $q_{W,i,j} \in_R \{0,1\}$ and coin tosses $r_{W,i,j} \in_R \{0,1\}^v$ to flip and blind the commitments of P. W sends V the resulting commitments $c_{PW,i,j} = c_{P,i,j} \cdot f(q_{W,i,j}, r_{W,i,j})$, $1 \leq j \leq v$.

V1. V selects bits $q_{V,i,j} \in_R \{0,1\}$ and flips the commitments of W. Let $c_{PWV,i,j} = c_{PW,i,j} \cdot f(q_{V,i,j}, 0^v)$, $1 \leq j \leq v$. (These commit P, W, V to the joint bits $q_{i,j} = q_{P,i,j} \oplus q_{W,i,j} \oplus q_{V,i,j}$.) V sends all the bits $q_{V,i,j}$ to W.

W2. W computes the commitments $c_{PWV,i,j}$. Then it selects coin tosses $r'_{W,i,j} \in_R \{0,1\}^v$ to blind the $c_{PWV,i,j}$. W sends P the resulting commitments $c_{i,j} = c_{PWV,i,j} \cdot f(0, r'_{W,i,j})$, $1 \leq j \leq v$.

P2. P now can compute the jointly committed bits $q_{i,j}$ of Arthur's message q_i from the $c_{i,j}$ (since P has unlimited resources). Then P computes Merlin's reply y_i, and selects coin tosses $t_{P,i,j} \in_R \{0,1\}^v$ and sends W the commitments $d_{P,i,j} = f(y_{i,j}, t_{P,i,j})$, $1 \leq j \leq v$.

W3. W checks that these are bit commitments, using[17] P if necessary, and sets $w'_d = 1$ if this is not the case. Then it selects coin tosses $t_{W,i,j}, t'_{W,i,j} \in_R \{0,1\}^v$ to blind the commitments $d_{P,i,j}$. W sends P the commitments $d_{PW,i,j} = d_{P,i,j} \cdot f(0, t_{W,i,j})$, $1 \leq j \leq v$, and V the commitments $d_{PW^2,i,j} = d_{PW,i,j} \cdot f(0, t'_{W,i,j})$, $1 \leq j \leq v$.

[17] If W cannot check this, then P proves to W in zero-knowledge that there exist bits $q_{P,i,j}$ and coin tosses $r_{P,i,j} \in \{0,1\}^v$ such that $c_{P,i,j} = f(q_{P,i,j}, r_{P,i,j})$. With the commitment scheme in [16], W just has to check that the Jacobi symbol of the commitment is $+1$.

Subroutine 2. (The (P, W, V) proof: P proves that Arthur would accept Merlin's proof)
The prover P proves the NP statement:
$\exists q_{1,1}, \ldots, q_{1,v}, \ldots, q_{\ell,v}, \; y_{1,1}, \ldots, y_{1,v}, \ldots, y_{\ell,v}, \; z_{1,1}, \ldots, z_{1,v}, \ldots, z_{\ell,v}, \; t_{1,1}, \ldots, t_{1,v}, \ldots, t_{\ell,v}$:

$$A(x, s_\ell) \;\wedge\; \left(\bigwedge_{\substack{1 \le i \le \ell \\ 1 \le j \le v}} (c_{i,j} = f(q_{i,j}, z_{i,j})) \right) \;\wedge\; \left(\bigwedge_{\substack{1 \le i \le \ell \\ 1 \le j \le v}} (d_{PW,i,j} = f(y_{i,j}, t_{i,j})) \right) \;=\; 1 \;,$$

where $s_\ell = q_1 y_1 \cdots q_\ell y_\ell$, the bits of q_i are $q_{i,j}$, and the bits of y_i are $y_{i,j}$, by using Protocol 4. (Observe that only W can verify this proof. Indeed V, while asking its queries, does not know the commitments $c_{i,j}$ and $d_{PW,i,j}$, only the $c_{PWV,i,j}$ and $d_{PW^2,i,j}$. So it keeps a record of all the received messages, and will verify these in the following subroutine.) If in this proof the warden outputs the local variable $w_d = 1$, then W sets $w'_d = 1$.

Subroutine 3. (V checks the proof in Subroutine 2)
W sends V all the coin tosses $r'_{W,i,j}$ and $t'_{W,i,j}$ used to blind the commitments $c_{PWV,i,j}$ and $d_{PW,i,j}$. V uses these to compute the $c_{i,j}$ and $d_{PW^2,i,j}$ and then verifies the proof in Subroutine 2. V rejects if the verification fails.

If in the subroutines above, or if $w'_d = 1$, or if P, V, or W halt before the end of the protocol, then W outputs $w_d = 1$. **(End of Protocol)**

Proof, continued: Completeness follows directly. For soundness we use the argument in Lemma 2. From the same Lemma we get zero-knowledge. For subliminal-freeness observe that fairness is obvious. Detectability follows from Theorem 4 and from the fact that the commitment scheme is homomorphic. Observe that a dishonest verifier V' with unlimited resources could compute the committed bits of Arthur's message q_i, and halt prematurely if a particular pattern is not present. This is why we only get conditionally computational detectability. The proof is clearly minimal. ∎

Remark 9. This protocol is not divertible because it is not transparent.

7 Applications

Okamoto and Ohta have described various applications of divertible zero-knowledge proofs, such as untraceability, blind signatures, and elections [21, p.143]. A problem with the subliminal-free proofs presented in the previous sections is that when $x \notin L$ the honest prover will halt and the warden will set $w_d = 1$ (observe that the common input x is not generated by the prover or verifier). Since the warden cannot distinguish between the cases $x \in L$ and $x \notin L$, in a real life situation it will 'apprehend' the honest prover, or verifier. Such a warden is not 'fair' and will not be very popular! In this section we consider a scenario in which the warden is fair (Definition 10), and describe protocols which implement it.

Theorem 7. *Protocol 3 can be modified to obtain a subliminal-free zero-knowledge proof for Graph Isomorphism for which the detectability is conditionally statistical, and for which the warden is (unrestricted) fair. Similarly Protocols 4, 5, 6 can be modified to obtain subliminal-free zero-knowledge proofs for HC, GNI and IP for which the warden is (unrestricted) fair. For HC and GNI the detectability is conditionally statistical, for IP it is computational. Zero-knowledge is as in Theorems 2, 4, 5 and 6, respectively.*

Proof. If G_0, G_1 are not isomorphic then P sends W the bit $b_P = 1$ and proves to W that G_0, G_1 are not isomorphic. If this proof fails, then W sets $w_d = 1$. If G_0, G_1 are isomorphic then P sends

W the bit $b_P = 0$ and Protocol 3 is executed. Obviously the warden is (unrestricted) fair. Since the case ($b_P = 1, x \in L, w_d = 0$) happens with negligible probability, we only have conditionally statistical detectability. The proof for Protocols 4, 5, 6 is similar. ∎

A nice application of divertible proofs would allow a prover to convince *simultaneously* two (or more) verifiers W, V ($W_1, W_2, \ldots, V$). We call such proofs, sequential multi-verifier proofs. This would save the prover having to give two (or more) independent proofs. Such proofs give the warden W *in real-time* some power. An example of a multi-verifier proof is given in [8, Appendix].

8 Conclusions

It is known that there exist statistically[18] divertible perfectly zero-knowledge proofs for any commutative random self-reducible language [21]. In this paper we have shown that (*i*) Graph Isomorphism (which is not commutative random self-reducible [1, 21]) has a statistically divertible proof which is perfectly zero-knowledge, and (*ii*) any language in NP has a statistically divertible proof which is computationally zero-knowledge, provided secure encryption homomorphisms exist. We have also shown that (*iii*) Graph Isomorphism and Graph Non-Isomorphism (which is seemingly not in NP but in AM) have subliminal-free zero-knowledge proofs, and that (*iv*) any language in NP has a subliminal-free zero-knowledge proof, provided secure encryption homomorphisms exist, and then (*v*) extended this last result to IP.

We have also discussed applications in the context of untraceability, and subliminal-free channels. The following are open problems:

- Which classes of languages have statistically (computationally) divertible proofs which are perfectly (statistically) (computationally) zero-knowledge?
- Which classes of languages have subliminal-free proofs with conditionally perfect (statistical) (computational) detectability, fair warden, and perfect (statistical) (computational) zero-knowledge?

For a survey on subliminal-free channels the reader is refered to [7].

Acknowledgments:

The authors would like to thank Tatsuaki Okamoto, Kaoru Kurosawa, and John Leo for their several valuable comments on an earlier version of this work. The authors are grateful to Moti Yung for helpful discussions about [19] and the definition of subliminal-free proofs, and also to one of the referees for several corrections and important remarks, and to René Peralta for discussions related to this paper.

References

[1] D. Angluin and D. Lichtenstein. Provable security of cryptosystems: a survey. Technical Report TR288, Yale University, October 1983.

[2] L. Babai. Trading group theory for randomness. In *Proceedings of the seventeenth annual ACM Symp. Theory of Computing, STOC*, pp. 421–429, 1985.

[3] M. Blum. Coin flipping by telephone — a protocol for solving impossible problems. In *digest of papers COMPCON82*, IEEE Computer Society, pp. 133–137, 1982.

[4] M. Blum. How to prove a theorem so no one else can claim it. In *Proceedings of the International Congress of Mathematicians*, pp. 1444–1451, 1987.

[18] Perfectly divertible using the weaker definition (Remark 1).

[5] G. Brassard, D. Chaum, and C. Crépeau. Minimum disclosure proofs of knowledge. *Journal of Computer and System Sciences*, Vol 37(2), pp. 156–189, 1988.

[6] M. V. D. Burmester and Y. Desmedt. All languages in NP have divertible zero-knowledge proofs and arguments under cryptographic assumptions. In I. Damgård, editor, *Advances in Cryptology – Eurocrypt '90*, Lecture Notes in Computer Science Vol 473, Springer-Verlag, Berlin, pp. 1–10, 1991.

[7] M. Burmester, Y. G. Desmedt, T. Itoh, K. Sakurai, H. Shizuya, and M. Yung. A progress report on subliminal-free channels. In R. Anderson, editor, *Information Hiding, First International Workshop, Proceedings*, Lecture Notes in Computer Science Vol 1174, Springer-Verlag, Berlin, pages 159–168, 1996.

[8] L. Chen, I. Damgård, and T. P. Pedersen. Parallel divertibility of proofs of knowledge. In A. De Santis, editor, *Advances in Cryptology – Eurocrypt '94*, Lecture Notes in Computer Science Vol 950, Springer-Verlag, Berlin, pp. 140-155, 1995.

[9] R. Cleve. Limits on the security of coin flips when half the processors are faulty. In *Proceedings of the eighteenth annual ACM Symp. Theory of Computing, STOC*, pp. 364–369, 1986.

[10] D. E. R. Denning. *Cryptography and Data Security.* Addison-Wesley, Reading, MA, 1982.

[11] Y. Desmedt and M. Yung. Minimal cryptosystems and defining subliminal-freeness. In *Proceedings 1994 IEEE International Symposium on Information Theory*, page 347, Trondheim, Norway, June 27–July 1, 1994. Final paper in preparation.

[12] Y. G. Desmedt. Subliminal-free cryptosystems. Submitted to the Journal of Cryptology April 1989, revised version submitted May 3, 1994.

[13] Y. Desmedt, C. Goutier, and S. Bengio. Special uses and abuses of the Fiat-Shamir passport protocol. In C. Pomerance, editor, *Advances in Cryptology – Crypto '87*, Lecture Notes in Computer Science Vol 293, Springer-Verlag, Berlin, pp. 21–39, 1988.

[14] O. Goldreich, S. Micali, and A. Wigderson. Proofs that yield nothing but their validity and a methodology of cryptographic protocol design. In *27th Annual Symp. on Foundations of Computer Science (FOCS)*, IEEE Computer Society Press, pp. 174–187, 1986.

[15] O. Goldreich, S. Micali, and A. Wigderson. Proofs that yield nothing but their validity or all languages in NP have zero-knowledge proof systems. *Journal of the ACM*, 38(1), pp. 691–729, 1991.

[16] S. Goldwasser and S. Micali. Probabilistic encryption. *Journal of Computer and System Sciences*, 28(2), pp. 270–299, 1984.

[17] S. Goldwasser, S. Micali, and C. Rackoff. The knowledge complexity of interactive proof systems. *Siam J. Comput.*, 18(1), pp. 186–208, 1989.

[18] S. Goldwasser and M. Sipser. Private coins versus public coins in interactive proof systems. In *Proceedings of the eighteenth annual ACM Symp. Theory of Computing, STOC*, pp. 59–68, 1986.

[19] R. Impagliazzo and M. Yung, 1987. Personal Communication, 1992.

[20] T. Itoh, K. Sakurai, and H. Shizuya. Any language in IP has a divertible ZKIP. In H. Imai, R. L. Rivest, and T. Matsumoto, editors, *Advances in Cryptology – Asiacrypt '91*, Lecture Notes in Computer Science Vol 739, Springer-Verlag, Berlin, pp. 382–396, 1993.

[21] T. Okamoto and K. Ohta. Divertible zero knowledge interactive proofs and commutative random self-reducibility. In J.-J. Quisquater and J. Vandewalle, editors, *Advances in Cryptology – Eurocrypt '89*, Lecture Notes in Computer Science Vol 434, Springer-Verlag, Berlin, pp. 134–149, 1990.

[22] C. E. Shannon. Communication theory of secrecy systems. *Bell System Techn. Jour.*, Vol 28, pp. 656–715, 1949.

[23] G. J. Simmons. The prisoners' problem and the subliminal channel. In D. Chaum, editor, *Advances in Cryptology, Proceedings of Crypto 83*, Plenum Press N.Y., pp. 51–67, 1984.