

Snarky Signatures: Minimal Signatures of Knowledge from Simulation-Extractable SNARKs

Jens Groth – University College London

Mary Maller – University College London

How can a sender of a message prove themselves trustworthy without revealing who they are?



Example: Bitcoin



Bob

Pay Alice a coin
Signed: *Bob*



Bitcoin uses digital signatures:

- Trustworthy?
- Anonymous?

Example: Bitcoin



Bitcoin uses digital signatures:



- Trustworthy?
 - Bob can always convince the verifiers.
 - An adversary cannot forge Bob's signature.
 - An adversary cannot use Bob's signature on a different message.
- Anonymous?

Example: Bitcoin



Bob

Pay Alice a coin
Signed: *Bob*



Bob's wallet

A Fistful of Bitcoins: Characterizing Payments Among Men with No Names

Sarah Meiklejohn Marjori Pomarole Grant Jordan
Kirill Levchenko Damon McCoy Geoffrey M. Voelker Stefan Savage
University of California, San Diego George Mason University²

ABSTRACT

Bitcoin is a purely virtual currency, unbacked by either physical commodities or sovereign obligation. Instead, it relies on a combination of cryptographic protection and a peer-to-peer protocol for witnessing transactions. Consequently, Bitcoin has the intuitive property that while the ownership of money is implicitly anonymous, its flow is globally visible. In this paper we explore this unique characteristic further, using heuristic clustering to group Bitcoin wallets based on evidence of shared authority, and then using re-identification attacks (i.e., empirical purchasing of goods and services) to classify the operations of these clusters. From this analysis, we characterize longitudinal changes in the Bitcoin market, the various flows changes are placing on the system, and the challenges for those seeking to use Bitcoin for criminal or fraudulent purposes at scale.

Categories and Subject Descriptors

K.4.4 (Electronic Commerce): Payment schemes

Keywords
Bitcoin; Measurement; Anonymity

1. INTRODUCTION

Demanded for the virtual currency of various kinds has driven a proliferation in online payment systems over the last decade. Thus, in addition to established payment card networks (e.g., Visa and Mastercard) a broad range of so-called “alternative payment” has emerged including eWallets (e.g., PayPal, Google Checkout, and Bitcoin), direct debit systems (typically via ACH), such as eCheck, money transfer systems (e.g., Moneygram) and so on. However, virtually all of these systems have the property that they are denominated in existing fiat currencies (e.g., dollars), explicitly identify the payer in transactions, and are centrally or quasi-centrally administered.¹

In particular, there is a central controlling authority who has the technical and legal capacity to tie a transaction back to a pair of individuals.

¹Permission to make digital or hard copies of all or part of this work for personal or classroom use is granted without fee provided the copies are not made or distributed for profit or commercial advantage and the copies bear the name and the full citation on the first page. Copyrights for components of this work owned by others than ACM must be honored. Notwithstanding, no part of this publication may be reproduced, stored in a retrieval system, or transmitted, in any form or by any means, electronic, mechanical, photocopying, recording, or by any information storage and retrieval system, without permission from ACM Press. Copyright 2013 ACM 978-1-4503-2015-1/13/0001...\$15.00.
http://dx.doi.org/10.1145/2455000.2455001

By far the most intriguing exception to this rule is Bitcoin. First deployed in 2009, Bitcoin is an independent online monetary system that combines some of the features of cash and existing online payment methods. Like cash, Bitcoin transactions do not explicitly identify the payer or the payee: a transaction is a cryptographically signed transfer of funds from one public key to another. Moreover, like cash, Bitcoin transactions are irreversible (in particular, there is no chargeback risk to with credit cards). However, unlike cash, Bitcoin requires third party mediation: a global peer-to-peer network of participants validates and certifies all transactions; such decentralized accounting requires each network participant to maintain the entire transaction history of the system, currently amounting to over 100 GB of compressed data. Bitcoin identities are thus pseudonymous: while not explicitly tied to real-world individuals or organizations, all transactions are completely transparent.²

This unusual combination of features has given rise to considerable confusion about the nature and consequences of the anonymity that Bitcoin provides. In particular, there is concern that the combination of scalable, irreversible, anonymous payments would prove highly attractive for criminals engaged in fraud or money laundering. In a widely leaked 2012 Intelligence Assessment, FBI analysts made just this case and conclude that a key “defining” of Bitcoin for criminals is that “law enforcement faces difficulties detecting suspicious activity, identifying users and obtaining transaction records” [7]. Similarly, in a late 2012 report on Virtual Currency Schemes, the European Central Bank opines that the lack of regulation and the diligence might enable “criminals, terrorists, fraudsters and money launderers” and that “the extent to which any money flows can be traced back to a particular user is unknown” [6]. Indeed, there is at least some anecdotal evidence that this statement is true, with the widely publicized “Silk Road” service using Bitcoin to trade in a range of illegal goods (e.g., recreational drugs and firearms). Finally, adding to the mystery is Bitcoin’s considerable growth, both quantitatively — a merchant service, Bitpay, announced that it had signed up over 1,000 merchants in 2012 to accept the currency, and in April 2013 the exchange rate soared to 235 USD per Bitcoin before settling to a more modest 100 USD per Bitcoin — and qualitatively via integration with existing payment mechanisms (e.g., BitcoinATM offering to its users’ Bitcoin wallets to Mastercard accounts [5] and Bitcoin Central’s recent partnership with the French bank Cédex Mutual Arkia to gateway Bitcoin into the banking system [14]) and the increasing attention of world financial institutions (e.g., Canada’s recent decision to tax Bitcoin transactions [3] and PayPal’s recent decision

²Note that this statement is not strictly true since private exchanges of Bitcoin between consumers of a single third party exchange, such as Mt. Gox, need not (and do not) engage the global Bitcoin protocol and are therefore not transparent.

Bitcoin uses digital signatures:

- Trustworthy = ✓
- Anonymous = ✗
- Pseudonymous – Bob’s real name might be Roberta.
- Can often uncover Bob’s real world identity based on what he spends.

Example: Zerocoin



Owner of an unspent coin

Zerocoin uses signatures of knowledge:

- Trustworthy?



- The owner of an unspent coin can compute a signature.
- A person without an unspent coin cannot compute a signature.
- A signature cannot be adapted for use on a different message.

- Anonymous?

Example: Zerocoin



Owner of an unspent coin

Zerocoin uses signatures of knowledge:

- Trustworthy = ✓
- Anonymous = ✓
 - Signature of knowledge provides no additional information as to who the spender is.

Example: Zerocoin



Owner of an unspent coin

Zerocoin uses signatures of knowledge:

- Trustworthy = ✓
- Anonymous = ✓
 - Signature of knowledge provides no additional information as to who the spender is.

However signatures of knowledge are large and take a long time to verify
=> Zerocoin not efficient.

Example: Zcash



Owner of an unspent coin

Zcash uses zk-SNARKs:

- Trustworthy?
- Anonymous? ✓
 - zk-SNARKs provides no additional information as to who the spender is.

zk-SNARKs are small and take a small time to verify
=> Zcash is efficient.

Example: Zcash



Owner of an unspent coin

Standard zk-SNARKs do not provide this property. Zcash has to take additional steps to prevent transaction malleability.

Zcash uses zk-SNARKs:

- Trustworthy =
 - The owner of an unspent coin can compute a proof.
 - A person without an unspent coin cannot compute a proof.
 - A proof cannot be adapted for use on a different message????

• Anonymous =



Our Contributions

- We construct the first simulation-extractable SNARK (SE-SNARK).
- We exploit a link between signatures of knowledge and SE-SNARKs to also get the first succinct signature of knowledge.



Ingredients:

1. Asymmetric bilinear groups;
2. Square arithmetic programs;
3. External power knowledge of exponent assumption;
4. Computational assumption.

Plan

Definitions

Square
Arithmetic
Programs

Construction

Efficiency

Asymmetric Bilinear Groups

$$bp = (p, \mathbb{G}, \mathbb{H}, \mathbb{T}, e)$$

prime $\rightarrow p$

Groups of order $p \rightarrow \mathbb{G}, \mathbb{H}$

Function $e: \mathbb{G} \times \mathbb{H} \rightarrow \mathbb{T}$

- There are efficient algorithms for deciding group membership and computing group operations;

- No isomorphism between \mathbb{G} and \mathbb{H} is efficiently computable in either direction.

Properties:

Bilinearity: $e(G^a, H^b) = e(G, H)^{ab}$

Non-degeneracy: if $X \neq 1$ and $Y \neq 1$ then $e(X, Y) \neq 1$

Efficient: e is efficiently computable.

SE-SNARK

Simulation-Extractable zero-knowledge
Succinct Non-interactive ARgument of
Knowledge

SE-SNARKs

“A person knows a witness for an instance Φ .”

Properties:

Correct:

A person who knows a witness can always convince the verifier.

Zero Knowledge:

The verifier learns no information from the proof except that the instance is true.

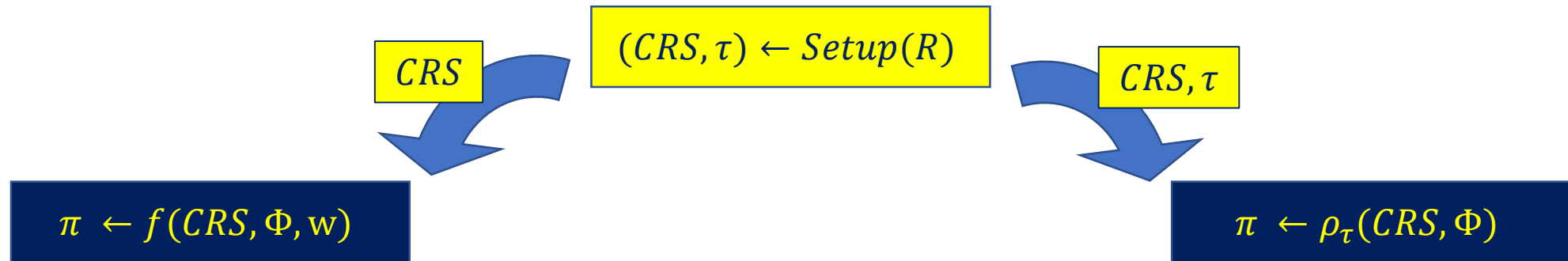
Sound:

A false statement cannot be proven.

Simulation-Extractable: *Old proofs cannot be used to forge new proofs of false statements.*

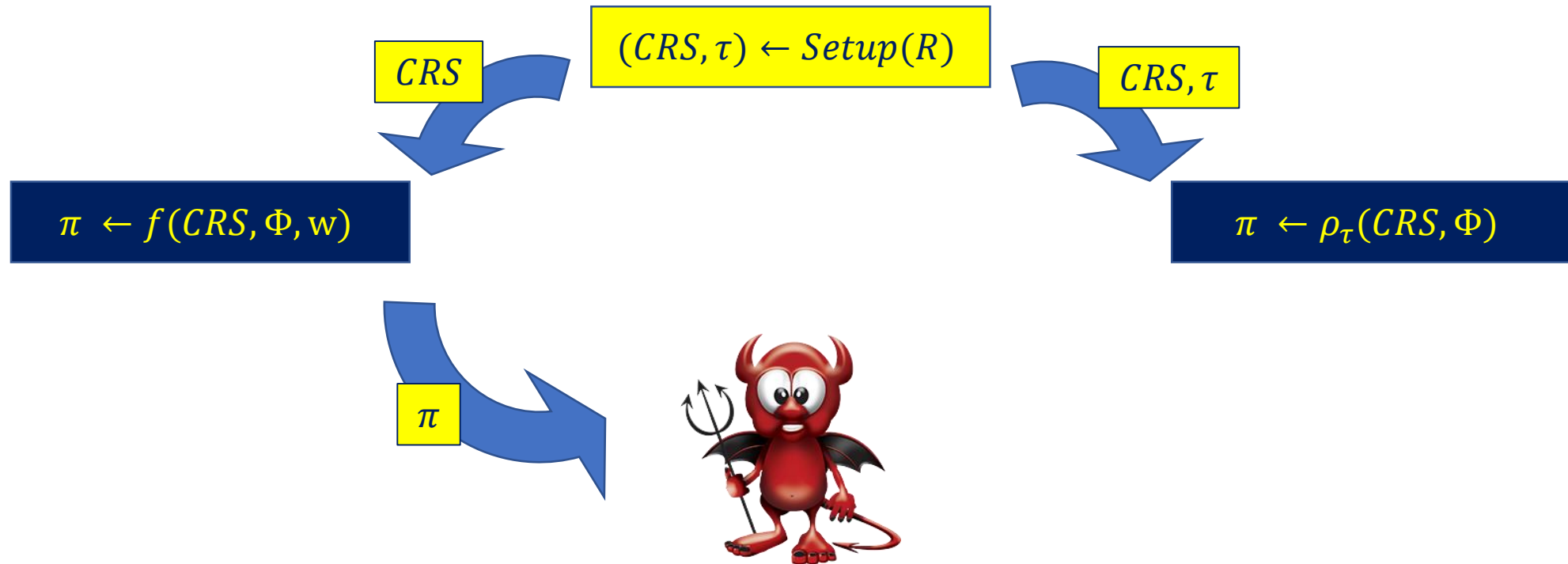
Zero-Knowledge

Zero Knowledge: The verifier learns no information from the proof except that the instance is true.



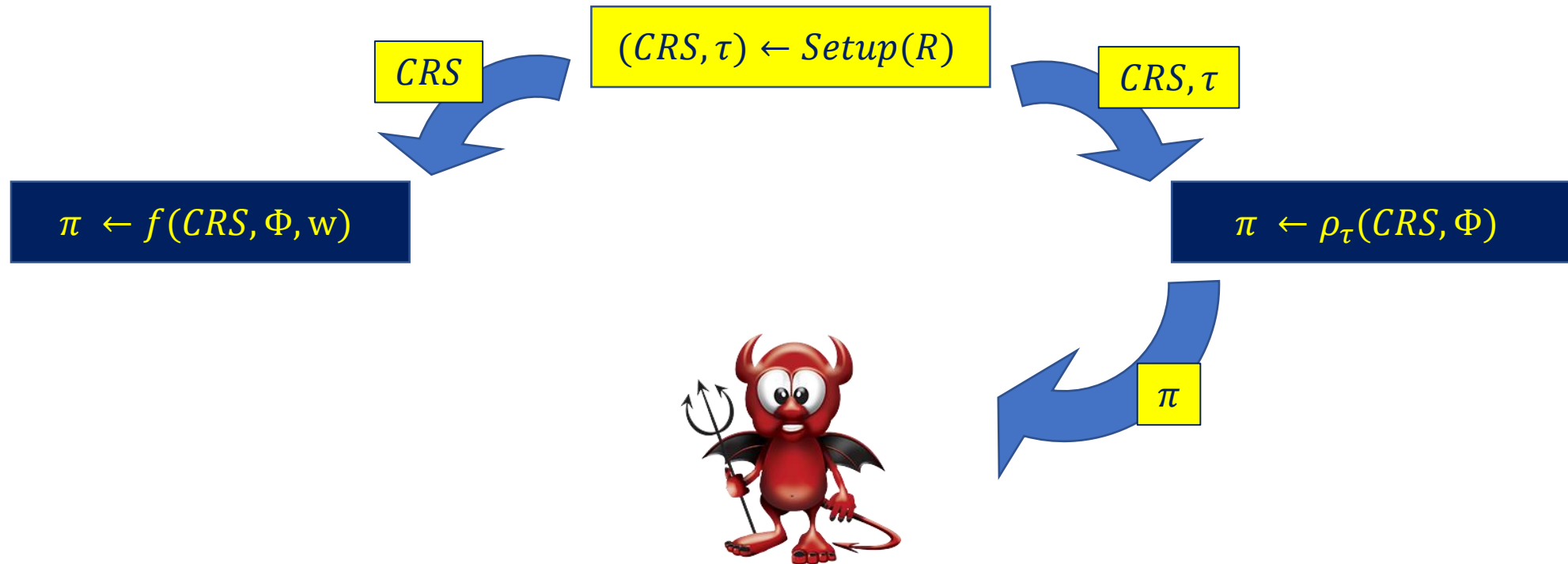
Zero-Knowledge

Zero Knowledge: The verifier learns no information from the proof except that the instance is true.



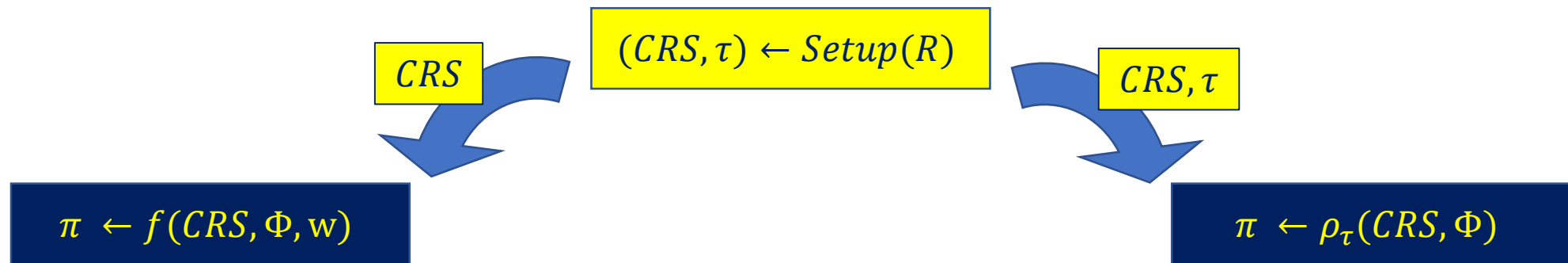
Zero-Knowledge

Zero Knowledge: The verifier learns no information from the proof except that the instance is true.



Zero-Knowledge

Zero Knowledge: The verifier learns no information from the proof except that the instance is true.

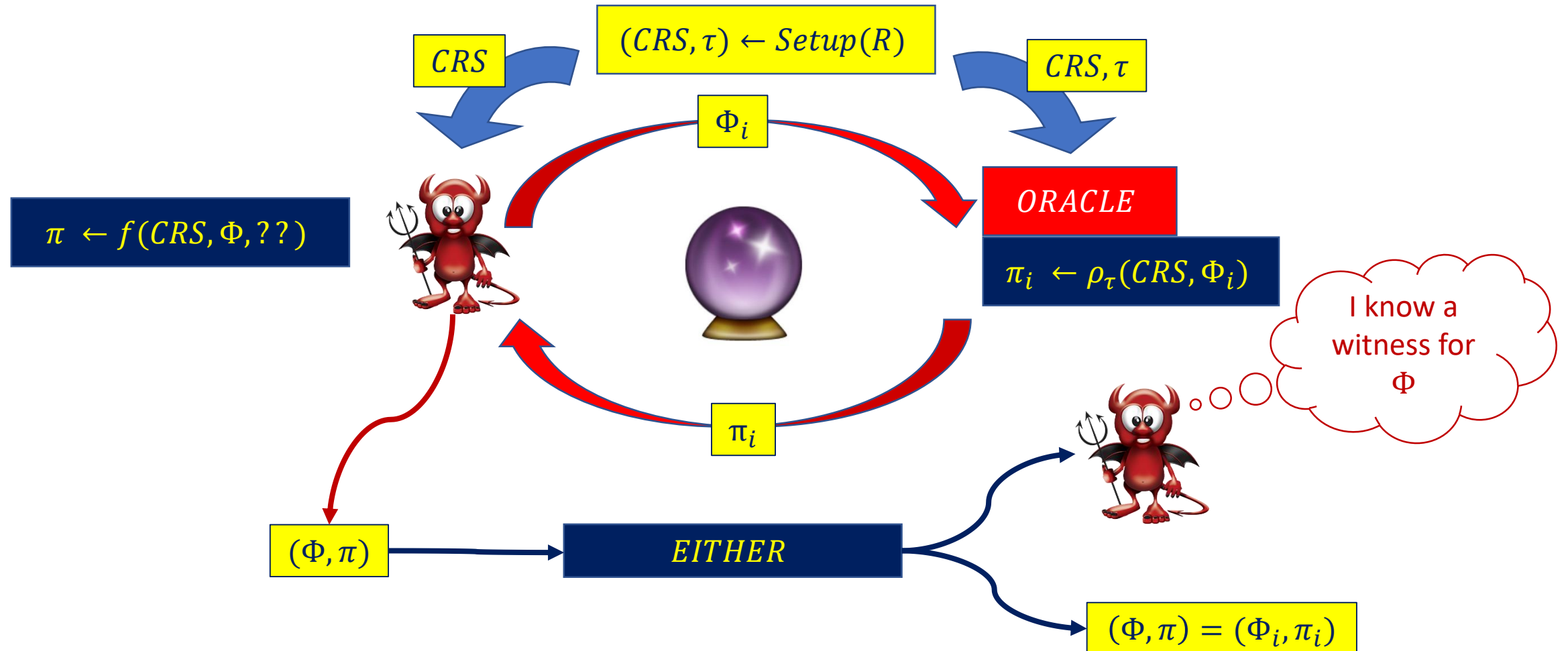


Did the prover use the witness or the trapdoor to compute π ?



Simulation-Extractability

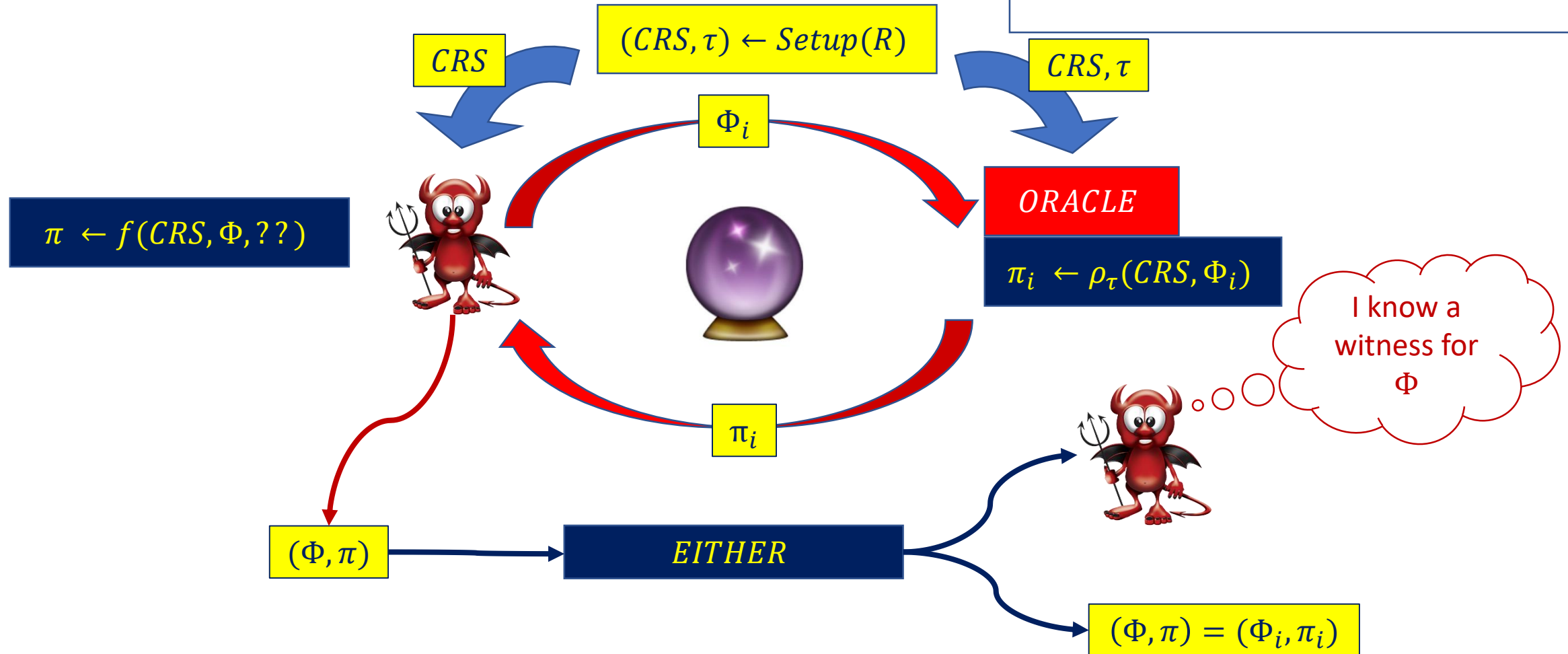
Simulation-Extractable: Old proofs cannot be used to forge new proofs of false statements.



Simulation-Extractability

Simulation-Extractable: Old proofs cannot be used to forge new proofs

Implies Soundness



Succinctness

The size of the proof and the time taken to verify a proof does not depend on the size of the witness.

Signature of Knowledge

“A person who knows a witness for an instance Φ has signed a message.”

Properties:

Correct:

A person who knows a witness can always convince the verifier.

Zero Knowledge:

The verifier learns no information from the signature except that the instance is true.

Sound:

A false statement cannot be signed.

Simulation-Extractable: *Old signatures cannot be used to forge new signatures of false statements.*

Plan

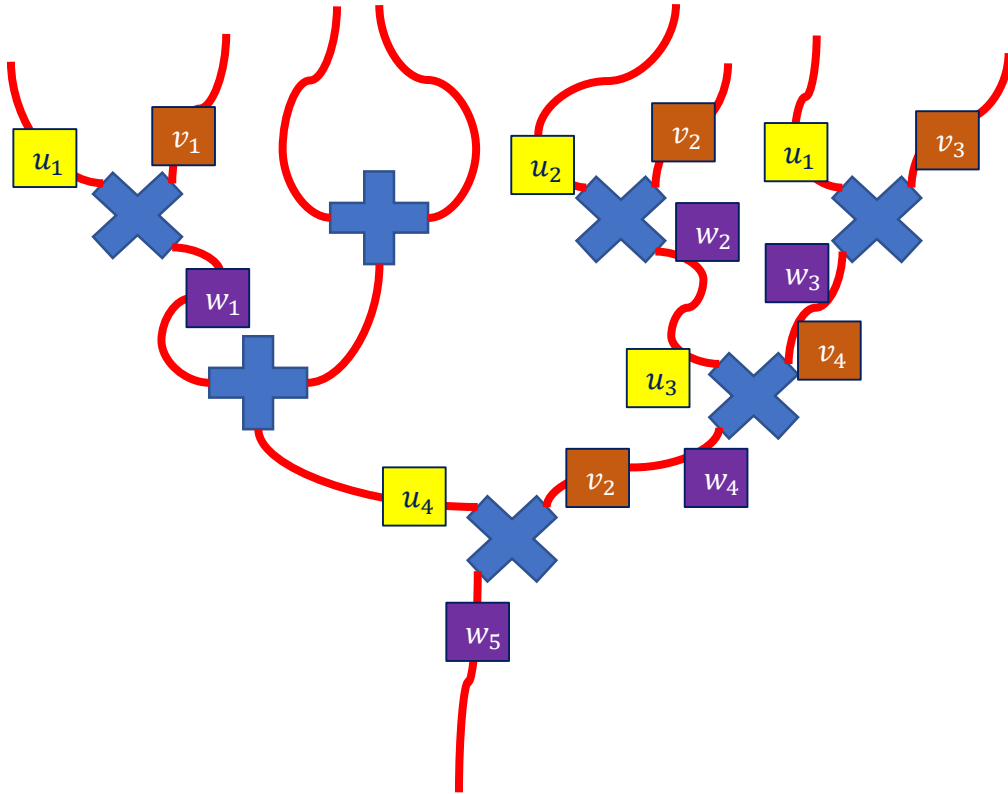
Definitions

Square
Arithmetic
Programs

Construction

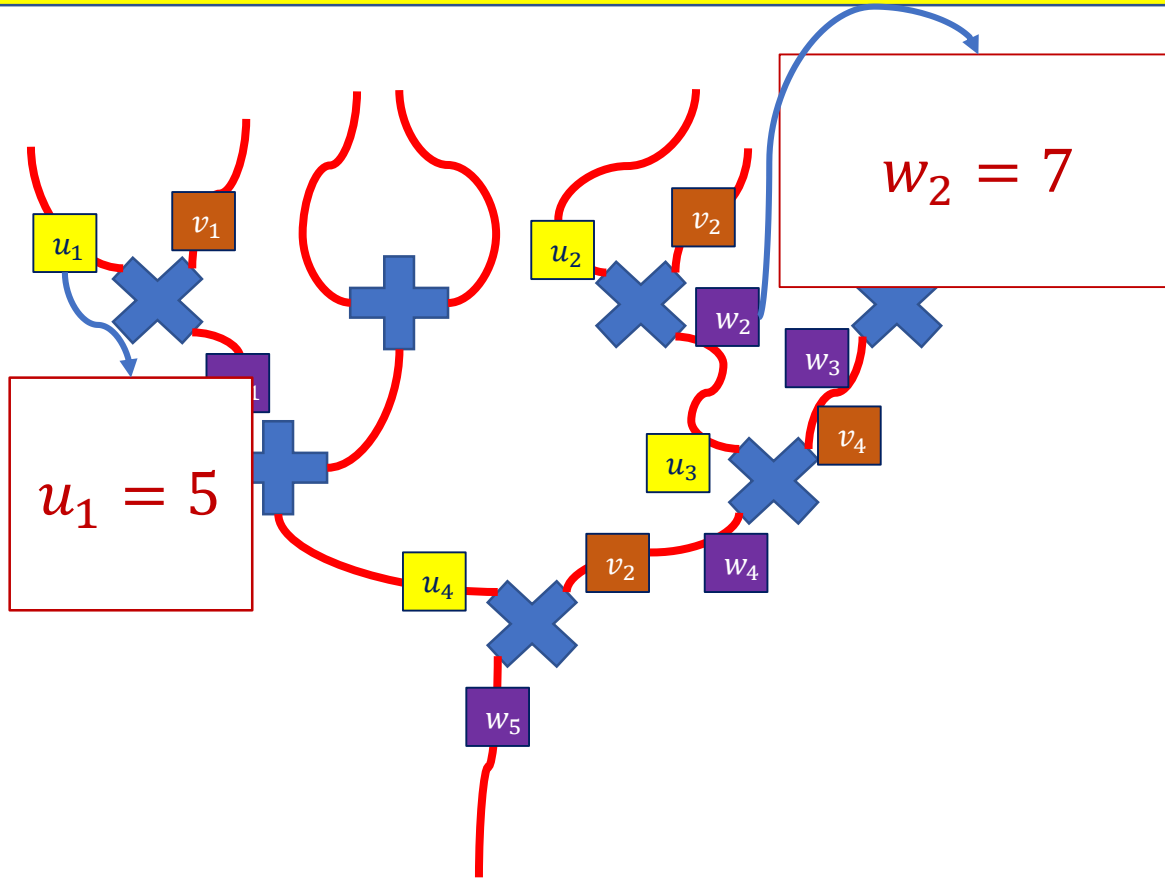
Efficiency

Arithmetic Circuits



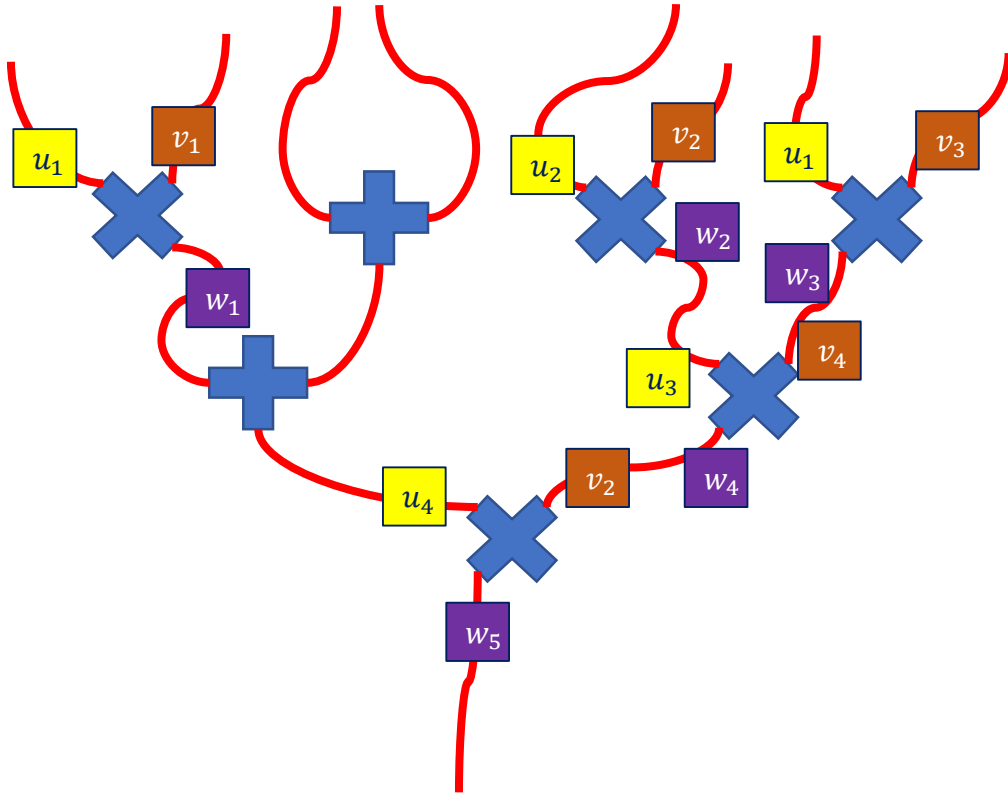
- Encoding of NP languages.
- The instance is some of the wire values that are revealed.
- The witness is the value of the remaining wires.

Arithmetic Circuits



- Encoding of NP languages.
- The instance is some of the wire values that are revealed.
- The witness is the value of the remaining wires.

Arithmetic Circuits



- Prover commits to values of wires.
- Prover shows
 - Output wires consistent with input wires.
 - Multiplication and addition gates calculated correctly.

Quadratic Arithmetic Programs [GGPREurocrypt13]

Relation described by

degree $n - 1$
polynomials

degree n
polynomial

$$R = (p, \ell, \{u_i(X), v_i(X), w_i(X)\}_{i=0}^m, t(X))$$

Instance $\Phi = (s_1, \dots, s_\ell)$ and witness $w = (s_{\ell+1}, \dots, s_m)$ satisfy arithmetic circuit C if and only if

$$(\sum_{i=0}^m s_i u_i(X)) (\sum_{i=0}^m s_i v_i(X)) = (\sum_{i=0}^m s_i w_i(X)) + \text{mod } t(X)$$

Quadratic Arithmetic Programs [GGPREurocrypt13]

Relation described by

degree $n - 1$
polynomials

degree n
polynomial

$$R = (p, \ell, \{u_i(X), v_i(X), w_i(X)\}_{i=0}^m, t(X))$$

Instance $\Phi = (s_1, \dots, s_\ell)$ and witness $w = (s_{\ell+1}, \dots, s_m)$ satisfy arithmetic circuit C if and only if

$$(\sum_{i=0}^m s_i u_i(X)) (\sum_{i=0}^m s_i v_i(X)) = (\sum_{i=0}^m s_i w_i(X)) + \text{mod } t(X)$$

$$s_0 = 1$$

Square Arithmetic Programs

We can ensure that left input wires and right input wires are the same if we double the size of the circuit.

Relation described by

$$R = (p, \ell, \{u_i(X), w_i(X)\}_{i=0}^m, t(X))$$

(s_1, \dots, s_m) satisfy circuit C if and only if

$$(\sum_{i=0}^m s_i u_i(X))^2 = (\sum_{i=0}^m s_i w_i(X)) + \text{mod } t(X)$$

$$s_0 = 1$$

degree $2n - 1$
polynomials

degree $2n$
polynomial

Plan

Definitions

Square
Arithmetic
Programs

Construction

Efficiency

Groth Eurocrypt 2016 Construction

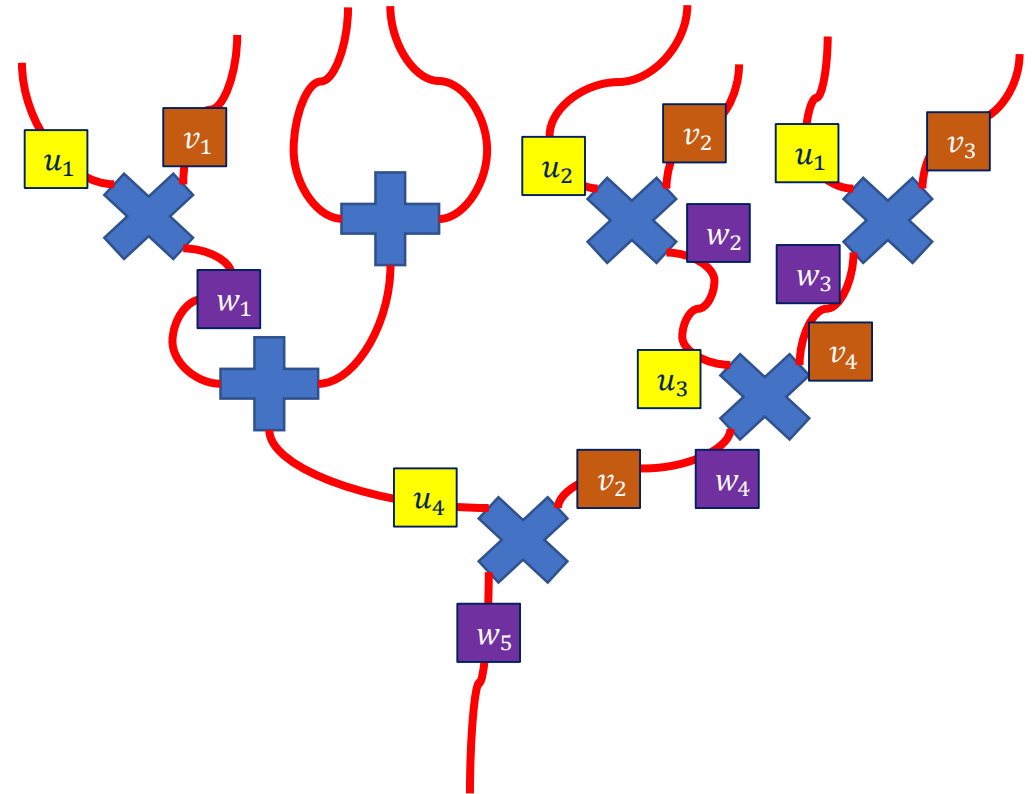
Instance = Φ

Commitment to left input wires

Commitment to right input wires

Proof = (A, B, C) group elements.

Commitment to output wires



Groth Eurocrypt 2016 Construction

Instance = Φ

Proof = (A, B, C) group elements.

Verification Equation:

$$e(A, B) = e(G^{\alpha}, H^{\beta}) e(G^{f(\Phi) \frac{1}{\delta_1}}, H^{\delta_1}) e(C, H^{\delta})$$

Known function of Φ

secret

Each of these pairings contribute towards knowledge soundness

Groth Eurocrypt 2016 is Sound

Multiplication and addition gates evaluated correctly

$$e(A, B) = e(G^\alpha, H^\beta) e(G^{f(\Phi) \frac{1}{\delta_1}}, H^{\delta_1}) e(C, H^\delta)$$

α, β ensure internal
wires are consistent

Hard to find $G^{f(\Phi)}$ or $H^{f(\Phi)}$
=
prover must use their witness.

Groth Eurocrypt 2016 not SE

Suppose A, B, C satisfy

$$e(A, B) = e(G^\alpha, H^\beta) e(G^{f(\Phi) \frac{1}{\delta_1}}, H^{\delta_1}) e(C, H^\delta)$$

Then so does
 $A^r, B^{\frac{1}{r}}, C$



Then so does
 $A, B \cdot H^{r\partial}, A^r \cdot C$

Our Techniques: 1

Suppose A, B, C satisfy

$$e(AG^\alpha, BH^\beta) = e(G^\alpha, H^\beta) e(G^{f(\Phi)^{\frac{1}{\delta_1}}, H^{\delta_1}}) e(C, H^\delta)$$

Second
verification
equation

$$e(A, H^\gamma) = e(G^\gamma, B)$$

Then so does
 $A^r, B^{\frac{1}{r}}, C$



Our Techniques: 2

Suppose A, B, C satisfy

CRS
contains
 H, G^γ, H^γ
but not G

$$e(AG^\alpha, BH^\beta) = e(G^\alpha, H^\beta)e(G^{f(\Phi)}, H^\gamma)e(C, H)$$
$$e(A, H^\gamma) = e(G^\gamma, B)$$

Cannot calculate C'
from the CRS

Implies C' contains a factor of γ^2



Implies r depends on γ

Then so does
 $A', B' = BH^r, C'???$

Implies $A' = AG^r$

Need second
verification
equation

Our Techniques: 2

Suppose A, B, C satisfy

CRS
contains
 H, G^γ, H^γ
but not G

$$e(AG^\alpha, BH^\beta) = e(G^\alpha, H^\beta)e(G^{f(\Phi)}, H^\gamma)e(C, H)$$
$$e(A, H^\gamma) = e(G^\gamma, B)$$

Cannot calculate C'
from the CRS

Implies C' contains a factor of γ^2



Implies r depends on γ

Then so does
 $A', B' = BH^r, C'???$

Implies $A' = AG^r$

Need second
verification
equation

Our Techniques: 2

Suppose A, B, C satisfy

CRS
contains
 H, G^γ, H^γ
but not G

$$e(AG^\alpha, BH^\beta) = e(G^\alpha, H^\beta)e(G^{f(\Phi)}, H^\gamma)e(C, H)$$
$$e(A, H^\gamma) = e(G^\gamma, B)$$

Cannot calculate C'
from the CRS

Implies C' contains a factor of γ^2



Implies r depends on γ

Then so does
 $A', B' = BH^r, C'???$

Implies $A' = AG^r$

Need second
verification
equation

Our Techniques: 2

Suppose A, B, C satisfy

CRS
contains
 H, G^γ, H^γ
but not G

$$e(AG^\alpha, BH^\beta) = e(G^\alpha, H^\beta)e(G^{f(\Phi)}, H^\gamma)e(C, H)$$
$$e(A, H^\gamma) = e(G^\gamma, B)$$

Cannot calculate C'
from the CRS

Implies C' contains a factor of γ^2



Implies r depends on γ

Then so does
 $A', B' = BH^r, C'???$

Implies $A' = AG^r$

Need second
verification
equation

Our Techniques: 2

Suppose A, B, C satisfy

CRS
contains
 H, G^γ, H^γ
but not G

$$e(AG^\alpha, BH^\beta) = e(G^\alpha, H^\beta)e(G^{f(\Phi)}, H^\gamma)e(C, H)$$
$$e(A, H^\gamma) = e(G^\gamma, B)$$

Cannot calculate C'
from the CRS

Implies C' contains a factor of γ^2



Implies r depends on γ

The A', B' satisfies $C'???$

Need second
verification
equation

Implies $A' = AG^r$

Plan

Definitions

Square
Arithmetic
Programs

Construction

Efficiency

Efficiency

	Groth	BCTV	This work
CRS size	$m + 2n + 3 \mathbb{G}_1$ $n + 3 \mathbb{G}_2$	$6m + n - \ell \mathbb{G}_1$ $m \mathbb{G}_2$	$m + 5n + 5 \mathbb{G}_1$ $2n + 3 \mathbb{G}_2$
Proof size	$2 \mathbb{G}_1, 1 \mathbb{G}_2$	$7 \mathbb{G}_1, 1 \mathbb{G}_2$	$2 \mathbb{G}_1, 1 \mathbb{G}_2$
Prover computation	$m + 3n - \ell + 3 E_1$ $n + 1 E_2$	$6m + n - \ell E_1$ $m E_2$	$m + 5n - \ell E_1$ $2n E_2$
Verifier computation	$\ell E_1, 3 P$	$\ell E_1, 12 P$	$\ell E_1, 5 P$
Verification equations	1	5	2

- Public parameters and prover computation a bit higher than the others.
- Verifier computation is low
- Verifier equations are minimal for SE-SNARKs
- Proof size is minimal for SE-SNARKs

Proof in full version
eprint.iacr.org/2017/540

Implemented in libsnark by Popovs, Chiesa, and Virza

github.com/scipr-lab/libsnark/tree/master/libsnark/zk_proof_systems/ppzksnark/r1cs_se_ppzksnark



Thank-you for listening