ZERO KNOWLEDGE PROOF EVENTS: Ali Baba's cave

Almpani Sofia
Gkantzounis Asterios
Stefaneas Petros

INTRODUCTION

- The concept of mathematical proof has undergone significant changes in the 20th century.

Goguen described proof event as:

- a social event that takes place in specific place and time and
- involves public communication.
- It embraces any proving activity.

(incomplete proofs, attempts to verify a conjecture etc.)

INTRODUCTION

Almpani, Stefaneas and Vandoulakis described proof-events as:

- activity of a multi-agent system incorporating their history,
- forming sequences of proof-events evolving in time,
- based on a logic-based argumentative context.
- Agents: Provers and Interpreters

An area of implementation of this theory is the <u>zero knowledge proofs.</u>

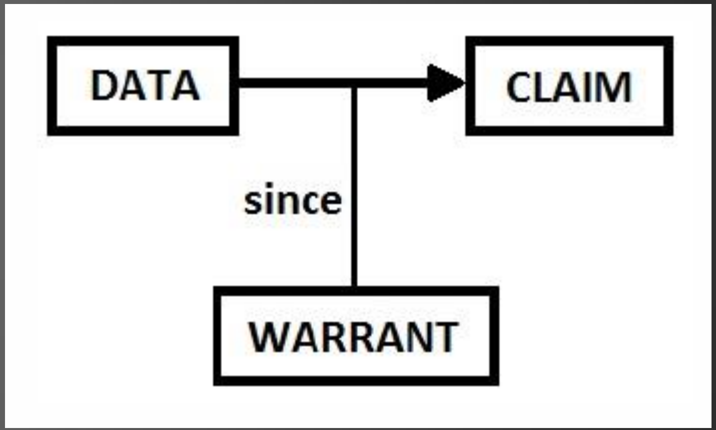
ZERO-KNOWLEDGE PROOFS

- In zero-knowledge proof:
- one party tries to persuade another for the validity of a statement, without revealing any information afar the legitimacy of the proof.
- It is a protocol between (at least) two people:
- The *prover* tries to prove a certain point to the other party, without conveying any information apart from the fact that she knows the proof.
- *Verifier* cannot even prove the statement to anyone else later.

A SYNOPSIS OF PROOF-EVENTS CALCULUS

The standard mathematical notion of a proof has:

- axioms and inference rules - *premises*,
- a *conclusion*, and
- a string of *sentences* that derives the conclusion from the axioms using the inference rules.



In a similar way, an argument has:

- *data* of the argument,
- a *claim* that refers to a fixed problem, and
- the inference rules - *warrant* - which allow data to be connected with the claim.

A SYNOPSIS OF PROOF-EVENTS CALCULUS

- A proof-event *e* can be represented as an argument $\langle \boldsymbol{\Phi}, \boldsymbol{c} \rangle$:

$$e\langle \boldsymbol{\Phi}, \boldsymbol{c} \rangle\text{: } e \cap < communicate < \Phi, c >, w >$$

Φ:Data of the argument,

c:Claim that refers to a fixed problem,

w:are the inference rules (Warrant) which allow Φ to be connected with c.

- Similarly to the *premises, conclusion, and sentences* of a proving.
- The arguments of the verifier are represented by the corresponding pair $\boldsymbol{e}^{*}(\boldsymbol{\Psi}, \boldsymbol{\beta})$.

A SYNOPSIS OF PROOF-EVENTS CALCULUS

The connection of the abovementioned theories is described with the following relations:

$\boldsymbol{data}: prem(e) = prem(e_1) \cup prem(e_2) \cup prem(e_3) \equiv \Phi_1 \cup \Phi_2 \cup \Phi_3$
$\mathbf{claim}: \mathrm{conc}(e) \equiv \mathrm{c} = \mathrm{c}_1 \cap \mathrm{c}_2 \cap \mathrm{c}_3$
$\mathbf{warrant}: \mathrm{sent}(e) = \mathrm{sent}(e_1) \cup \mathrm{sent}(e_2) \cup \mathrm{sent}(e_2) \equiv w_1 \cup w_2 \cup w_3$

A SYNOPSIS OF PROOF-EVENTS CALCULUS

The temporal predicates are described as below:

Happens(e,t)
Initiates(e, f, t_1): happens$(e, t_1) \rightarrow \neg$attack$(e^*, t_1) \cup$ support(e, t_1)
Terminates(e, f, e_1): $\exists e, e^*, t_1([\mathrm{attack}(e^*, t_1) \rightarrow \neg\mathrm{conc}(e) \cup \neg\mathrm{prem}(e)] \cap [\nexists(\mathrm{Happens}(e_2, t_2) \rightarrow \neg\mathrm{attack}(e^*, t_1))]$,
ActiveAt(e, f, t_{n+1}): Happens$(e_{n+1}, t_{n+1}) \rightarrow \neg$attack$(e_n^*, t_n) \cup$ support(e_n^*, t_n), for every $n \epsilon \mathbb{N}$, $t_{n+1} > t_n$
$\forall i \leq n[\mathrm{ActiveAt}(e, f, t_i) \cap (t_i \leq t_n) \cap \neg\mathrm{Terminates}(e, f, t_i)] \rightarrow$ Valid(e, t_n), at time t_n, $i = 1, .., n, n \epsilon \mathbb{N}$

ZERO KNOWLEDGE PROOFS AS PROOF EVENTS

- Zero Knowledge proofs is a protocol between ***prover*** *and* ***verifier***.
- The two parties play the corresponding roles of **prover** and **interpreter** in proof events.

ZERO KNOWLEDGE PROOFS AS PROOF EVENTS

- The protocol must necessarily require dialectical input from the verifier, usually in the form of a challenges such that the responses from the prover will convince the verifier if and only if the claim is true.

Procedure of justification is a recursion of the same round:

- a commitment message from the prover (data)
- a challenge from the verifier (attack),
- a response to the challenge from the prover (conclusion).

ZERO KNOWLEDGE PROOFS AS PROOF EVENTS

- We need a mechanism, which by recursion can examine the representation of the argumentative dialogue. Kakkas and Moraitis presented three levels of arguments.
- *Object level arguments*: our claim and the initial representations of arguments.
- *First-level priority arguments*: justifications on the object-level arguments in order to resolve possible conflicts. The same pattern continues for n-levels.
- n-level priority arguments: conflicts between priority arguments of the previous level.
- *Higher-order priority arguments*: proof-events sequence either terminates or is proved valid.

ZERO KNOWLEDGE PROOFS AS PROOF EVENTS

- The protocol may repeat for several **rounds,** where each round adds more value for the desirable result.
- Each round is equivalent with the corresponding **levels** of argumentation in proof events.
- Based on the prover's responses in all the rounds, the verifier decides whether **to** accept or reject the proof.

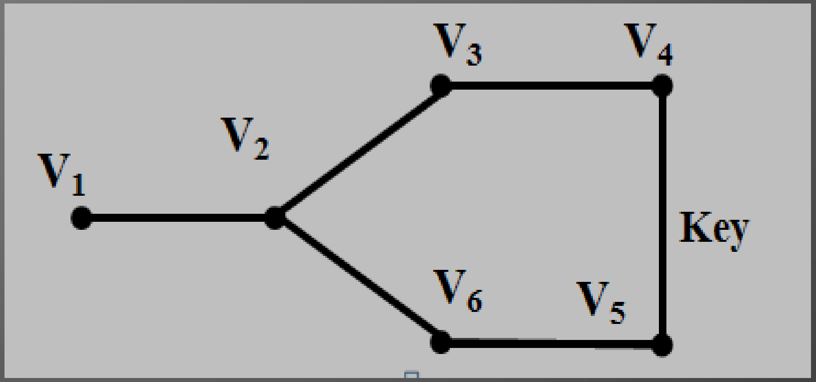
ALI BABA'S CAVE

- In Ali Baba's Cave Paradigm, (as described in [Quisquater et al., 1990]) we have:
- Two parties:

Peggy (Prover)

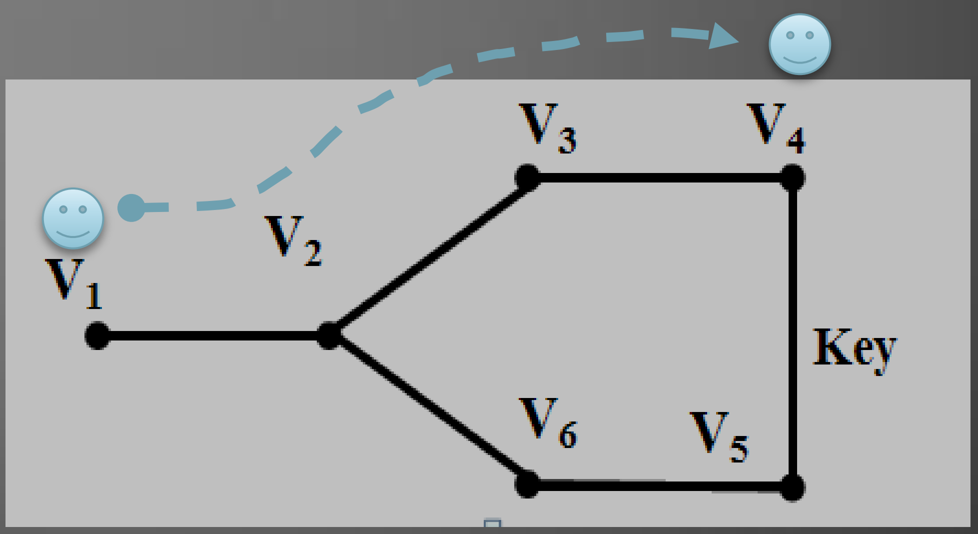
Victor (Verifier)

- A ring shapped cave with entrance on one side and a door blocking at the opposite side.



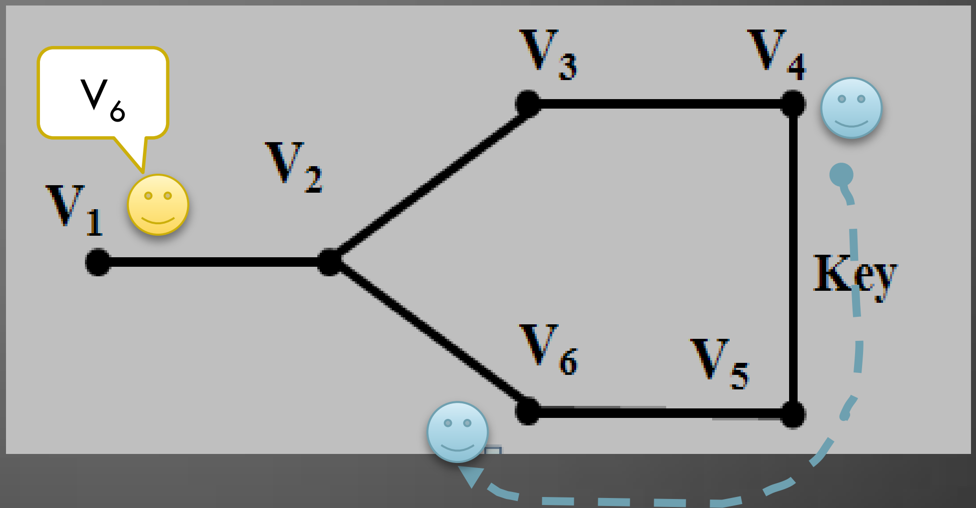
ALI BABA'S CAVE

- Peggy wants to prove to Victor that she knows the magic word (code) that can open the door, without revealing it or any other information to him or anyone else.
- Peggy enters the cave (V_1) and chooses to follow one of the two paths to the blocking door (V_4 or V_5).
  (e.g. V_4)



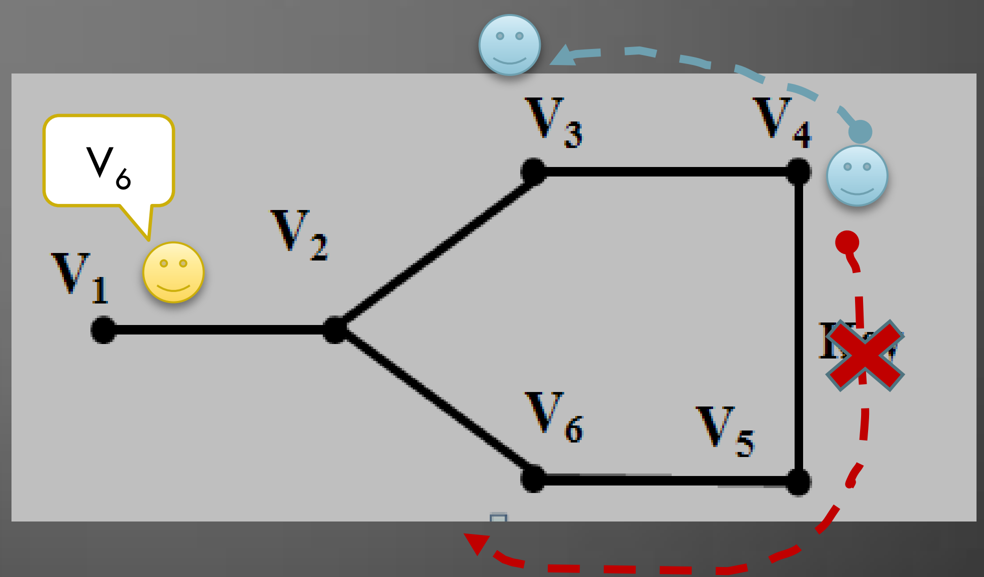
ALI BABA'S CAVE

- Then, Victor enters the cave (V_1) and asks from Peggy to come back to the entrance by following the path of his preference (V_3 or V_6). (e.g. V_6).
- If Peggy knows the secret word she can open the door and follow any path she wants to the entrance.



ALI BABA'S CAVE

- If she doesn't she can only get back by the path she had previously followed.



ALI BABA'S CAVE

- Repeating this procedure and as Peggy always achieves to come back through the requested way, Victor can conclude that she knows the secret word.

We formalize the example of Ali Baba's cave integrating:

- the moves and the temporal predicates of proof events,
- the basic elements and the three levels of argumentation theory, and
- the justification procedure of Zero Knowledge Proofs.

ALI BABA'S CAVE

<u>Object level arguments</u>

- Two agents: a verifier and a prover

$A \in \{A_V, A_P\}$, Verifier=A_V, Prover=A_P

- The basic elements of the statement that we want to prove: Data, Warrant and Claim.

The <u>Data</u> is the Graph G with its Vertices and Edges described as below.

$V(G) = \{V_i | i = 1, \ldots, 6\}$,

$$E(G) = \begin{cases} \{(V_i, V_j) | i + 1 = j \; or \; i = 2, j = 6\} \backslash (V_4, V_5) \; iff \; K((V_4, V_5)) = 0 \\ \{(V_i, V_j) | i + 1 = j \; or \; i = 2, j = 6\} \; iff \; K((V_4, V_5)) = 1 \end{cases}$$

ALI BABA'S CAVE

The warrant is illustrated by Prover's possession of the key, which is the claim to be proved, thus, whether A_P has the Key or not is expressed by:

$$K: (V_4, V_5) \rightarrow \{0,1\}$$

- The possible moves for the agents are below:

$\boldsymbol{StandsOn}$: $\{A_V, A_P\} \rightarrow V(G)$

$\boldsymbol{MovesTo}: StandsOn \rightarrow StandsOn$

$(A_i, V_i) \rightarrow (A_i, V_j)$

iff $(V_i, V_j) \in E(G) \vee (V_i, V_j) = (V_4, V_5)$ and A$=A_P$ and has the Key.

So P can move through (V_4, V_5) iff P has the Key.

- $\boldsymbol{Sees}: StandsOn \times StandsOn \rightarrow \{0,1\}$
- $Sees((A_V, V_i), (A_P, V_j)) = \begin{cases} 1, if\ (V_i, V_j) \in E(G) \\ 0, if\ (V_i, V_j) \notin E(G) \end{cases}$

ALI BABA'S CAVE

- **First-level priority arguments**

The Verifier A_V and the Prover A_P StandsOn V_1.

$$\text{StandsOn}=\text{Happens}(A_V, V_1),$$
$$\text{StandsOn} = \text{Happens}(A_P, V_1),$$

A_P MovesTo either V_4 or V_5. There is no attack for this move.

$$[\text{Happens}(A_P, V_2)) \rightarrow \text{MovesTo}(A_P, V_i)] \rightarrow \text{Initiates}(A_P, f_0, V_i) \text{ with } i = 4 \text{ or } 5$$

The procedure of proving Initiates and the verifier is testing whether prover has the key-proof by demanding to appear from one of the two possible exits of the cave (V_3 or V_6).

$$\text{Initiates}\left(A_p, f_m, V_i\right), \text{with } i = 4 \text{ or } 5$$
$$\text{MovesTo}(A_V, V_2)] \rightarrow \text{Happens}(A_V, V_2),$$
$$D_V=\text{attacks}\left(A_V, f_m, V_j\right), \text{with } j = 3 \text{ or } 6$$

ALI BABA'S CAVE

A_P MovesTo $(V_3$ or $V_6)$, if $Sees((A_V, V_2), (A_P, D_V)) = 0$ then it Terminates.

$[\text{attack}(A_V, V_j) \wedge \neg\text{Sees}((A_V, V_2), (A_P, V_j))] \rightarrow \neg\text{StandsOn}(A_P, V_J) \wedge \neg\, K: (V_4, V_5\,) \rightarrow \text{Terminates}(A_P, f_m, V_j)]$, with j=3 or 6, m=1,…,n-1

Else,

$\text{ActiveAt}(A_P, f_m, V_i)$ for i=4,5, m=1,…,n-1

It continues to the second-level by repeating the procedure from the beginning.

The same pattern continues for *n*-level priority arguments and for *n* fluents $f_,$ until verifier is convinced that prover has the key-proof.

ALI BABA'S CAVE

- **Higher-order priority arguments**

In the final n-level if at the time $t_{n.}$ we have:

$$\exists \mathrm{j}, \mathrm{j} \epsilon \mathbb{N}: \left[\mathrm{ActiveAt}\left(\mathrm{e_P}, \mathrm{f_n}, \mathrm{V_j}\right) \cap \neg \mathrm{Terminates}\left(\mathrm{e_P}, \mathrm{f_n}, \mathrm{V_j}\right)\right] \rightarrow \mathrm{Valid}(\mathrm{e_P}, \mathrm{t_n}),$$

then our claim is proved valid.

CONCLUSIONS

- We have developed a connection of the argumentative proof-events calculus with zero knowledge proofs.
- Proof-events are not considered as infallible facts before their ultimate validation, thus enabling the connection with the procedure of zero knowledge proofs where a recursive tentative process is required until the final validation of the proof.
- Future work: Application of this model to express further examples of zero knowledge proofs' sequence and properties to create a generalized abstract model of zero knowledge proofs' cases.

Thank you!!!