

COMPOSITIONAL MODELLING OF NETWORK GAMES

Elena Di Lauro

Tallinn University of Technology

Jules Hedges

University of Strathclyde

Parwet Sobociński

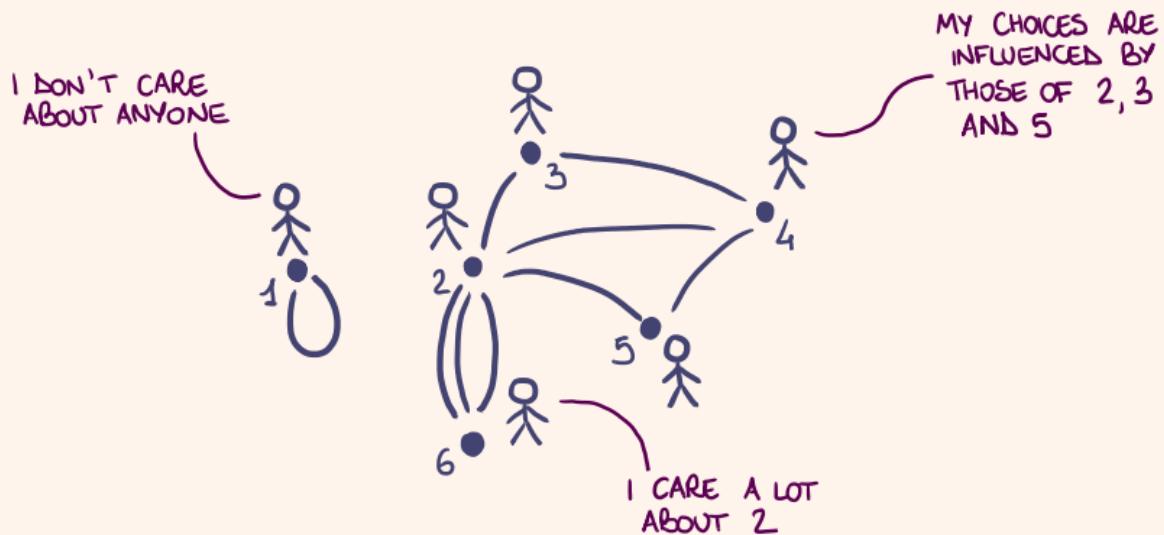
Tallinn University of Technology

[Di Lauro, Hedges, Sobociński, Compositional modelling of network games, 2020]

OUTLINE

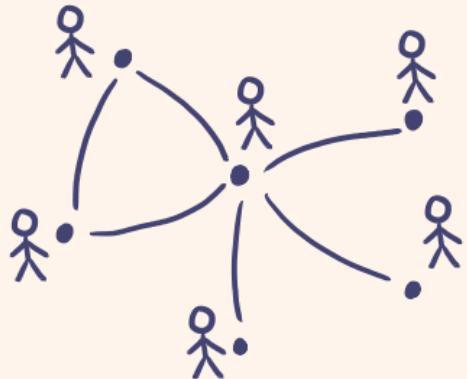
- **dgames on graphs**
- The prop of open graphs
- dgames on graphs as functors

GAMES ON GRAPHS



THE TRAGEDY OF THE COMMONS

- None of my neighbours invests \Rightarrow utility $1 - c + \varepsilon$
- One of my neighbours invests \Rightarrow utility 1
- I invest \Rightarrow utility $1 - c$



MAIN RESULT

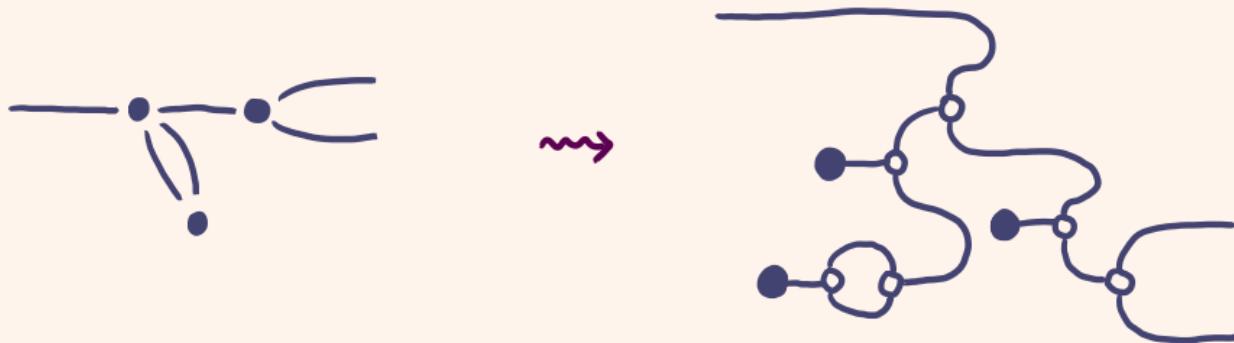
- Network games (a class of them)
correspond to functors

$\text{cgraph} \rightarrow \text{cgame}$

$\text{syntax} \rightarrow \text{semantics}$

MAIN RESULT

- characterise the category of graphs with boundaries as the free category on some generators and equations.



OUTLINE

- cgames on graphs
- The prop of open graphs
- cgames on graphs as functors

PROPS & STRING DIAGRAMS

$$\text{---} \circ \text{---} : 2 \rightarrow 1$$

$$\text{---} \circ \text{---} : 1 \rightarrow 0$$

sequential composition

$$\text{---} \circ \text{---} ; \text{---} \circ \text{---} = \text{---} \circ \text{---} : 2 \rightarrow 0$$

parallel composition

$$\text{---} \circ \text{---} \otimes \text{---} \circ \text{---} = \text{---} \circ \text{---} : 3 \rightarrow 1$$

FREE PROPS & FINSET

- FinSet is the free prop generated by



The diagram illustrates the law of associativity. It shows two ways of grouping three nodes: one where the first two are grouped together, and another where the last two are grouped together. The two configurations are shown as equivalent, separated by an equals sign.

(associativity)

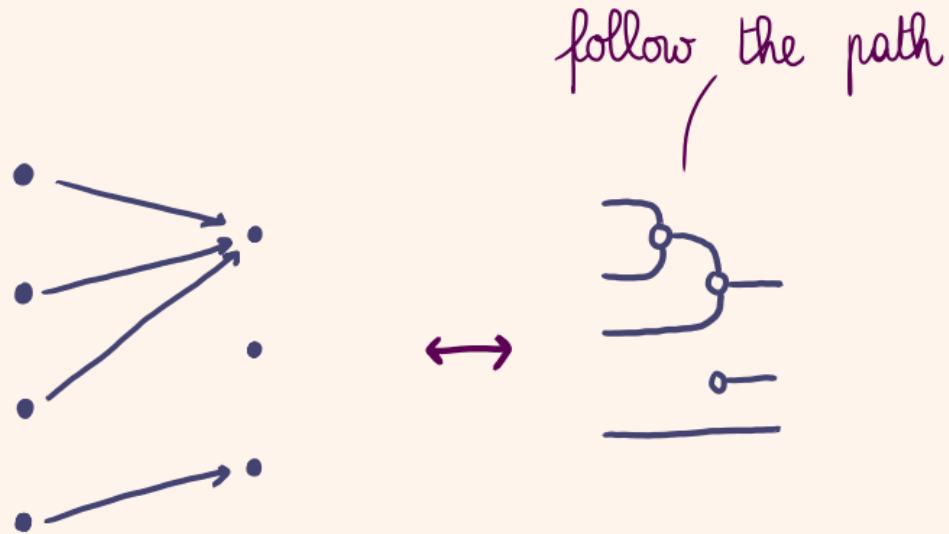
The diagram illustrates the law of unitality. It shows a node connected to a line, followed by an equals sign, then a line segment, followed by another equals sign, and finally a node connected to a line. This indicates that the node acts as a unit or identity element for the operation.

(unitality)

The diagram illustrates the law of commutativity. It shows two nodes connected to a line, followed by an equals sign, and then the same two nodes connected to the line in a different order. This indicates that the order of the nodes does not matter for the result of the operation.

(commutativity)

EXAMPLE IN FINSET



PROP OF MATRICES

- Natural numbers matrices are the free prop generated by

$$\text{--} \circ \text{---} , \circ \text{---} , \text{---} \circ \text{---} , \text{---}$$

(co)associativity, (co)unitality, (co)commutativity,

$$\circ \text{---} = \text{---} \circ , \text{---} \circ \text{---} = \text{---} \circ \text{---} \circ \text{---} , \quad (\text{bialgebra laws})$$

$$\text{---} \circ \text{---} = \text{---} \circ \text{---} , \quad \circ \text{---} \circ \text{---} =$$

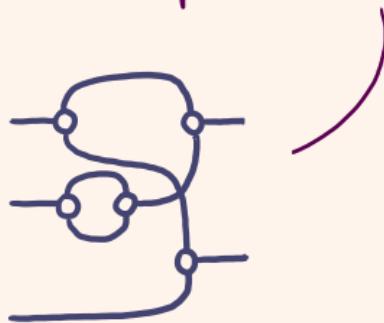
[S. Lack, Composing props, 2004]

[Bonchi, Sobociński, Zanasi, Interacting Hopf algebras, 2014]

EXAMPLE IN N-MATRICES

$$\begin{pmatrix} 1 & 2 & 0 \\ 1 & 0 & 1 \end{pmatrix} \leftrightarrow$$

count the paths between
inputs and outputs



GRAPHS & ADJACENCY MATRICES

$$G \quad \leftrightarrow \quad \begin{matrix} & \begin{matrix} 1 & 2 & 3 \end{matrix} \\ \begin{matrix} 1 \\ 2 \\ 3 \end{matrix} & \begin{pmatrix} 1 & 1 & 1 \\ 0 & 0 & 2 \\ 0 & 0 & 0 \end{pmatrix} \end{matrix} = \begin{pmatrix} 1 & 0 & 1 \\ 1 & 0 & 1 \\ 0 & 1 & 0 \end{pmatrix}$$

$$A \sim B \Leftrightarrow A + A^T = B + B^T$$

ADDING THE CUP

- Adjacency matrices are the $n \rightarrow 0$ morphisms in the free prop generated by



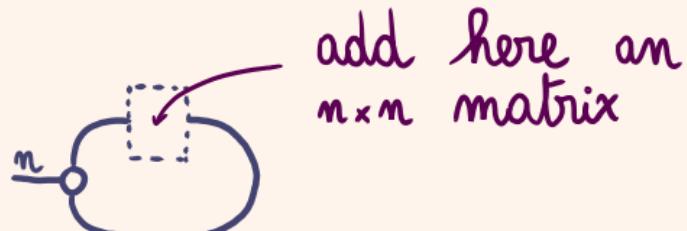
(co)associativity, (co)unitality, (co)commutativity,
bialgebra laws

$$\textcircled{D} = \textcircled{\textcircled{D}} , \textcircled{D} = \textcircled{-} , \textcircled{X} = \textcircled{C}$$

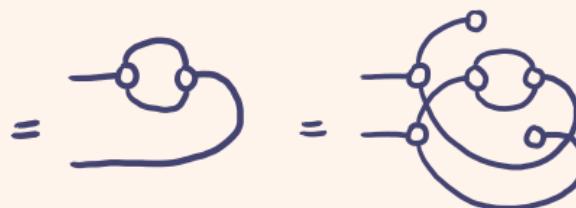
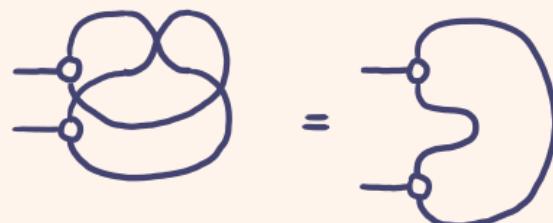
(cup laws)

[Chantawibul, Sobociński, Towards compositional graph theory, 2015]

EXAMPLE ADJACENCY MATRICES



$$\begin{pmatrix} 0 & 1 \\ 1 & 0 \end{pmatrix} \leftrightarrow$$



$$\leftrightarrow \begin{pmatrix} 0 & 2 \\ 0 & 0 \end{pmatrix}$$

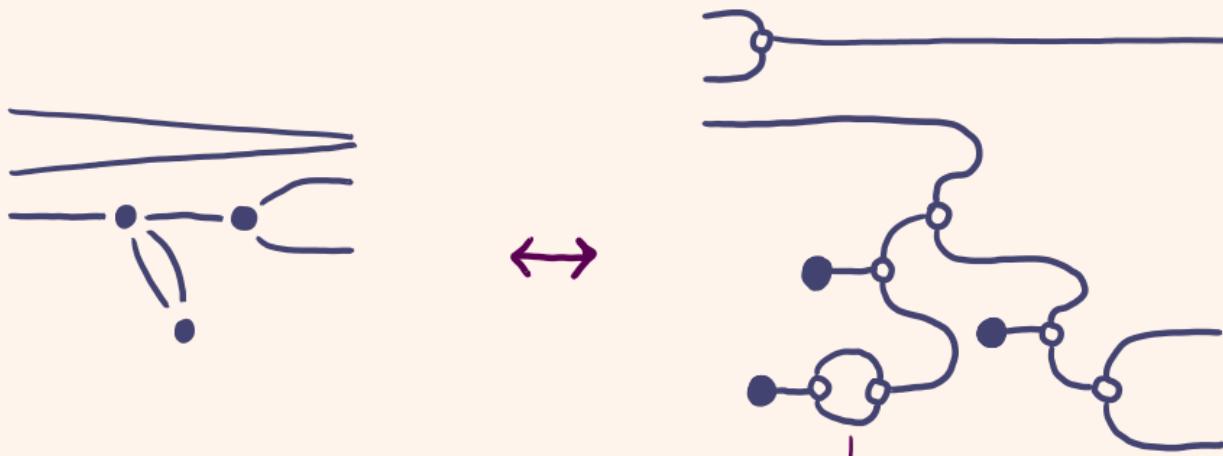
ADDING VERTICES

- graphs with boundaries are the free prop generated by

\exists , \circ , $- \circ -$, $\circ -$, $\circ \exists$

(co)associativity, (co)unitality,
bialgebra laws, cup laws

EXAMPLE OF GRAPH WITH BOUNDARIES



count the number of
paths between vertices
and boundaries

SUMMARY OF THE PROP OF OPEN GRAPHS

$$\text{---}, \circ, , -\langle \rangle, \multimap, \supset, \bullet$$

$$\text{---} = \text{---} \quad \text{---} = \text{---} = \text{---} \quad -\langle \rangle = -\langle \rangle$$

$$\circ\circ = \circ \quad \text{---} \langle \rangle = \text{---} \langle \rangle \quad \text{---}\circ = \circ\text{---}$$

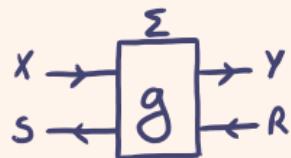
$$-\langle \rangle = -\langle \rangle \quad \circ\circ = \circ\circ \quad \circ\langle \rangle = \langle \rangle\circ$$

$$\circ\circ = \supset \quad \text{---} = \text{---} \quad \text{---} = \multimap$$

OUTLINE

- cgames on graphs
- The prop of open graphs
- cgames on graphs as functors

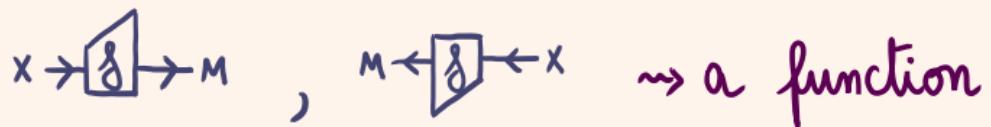
OPEN GAMES



- $P_g : \Sigma \times X \longrightarrow Y$ \rightsquigarrow next move
- $C_g : \Sigma \times X \times R \rightarrow S$ \rightsquigarrow contility
- $B_g : X \times (Y \rightarrow R) \longrightarrow P(\Sigma)$ \rightsquigarrow equilibria

[Lyhann, Hedges, Winskel, Zahn, Compositional game theory, 2018]

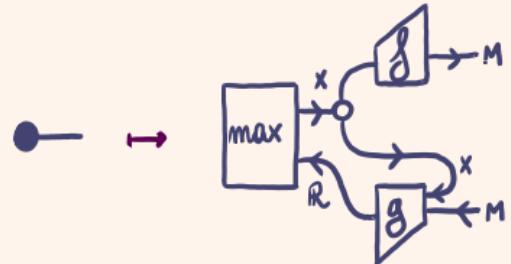
EXAMPLES OF OPEN GAMES



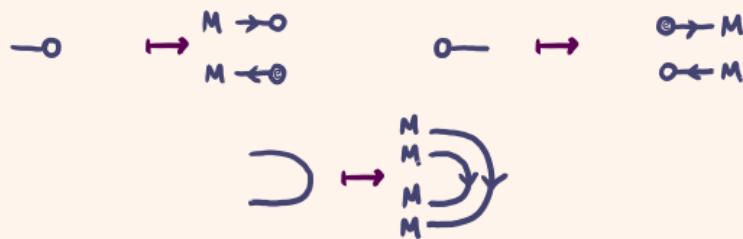
MONOID NETWORK GAMES AS FUNCTORS

$(M, \oplus, e, f: X \rightarrow M, g: X \times M \rightarrow R)$ monoid network game

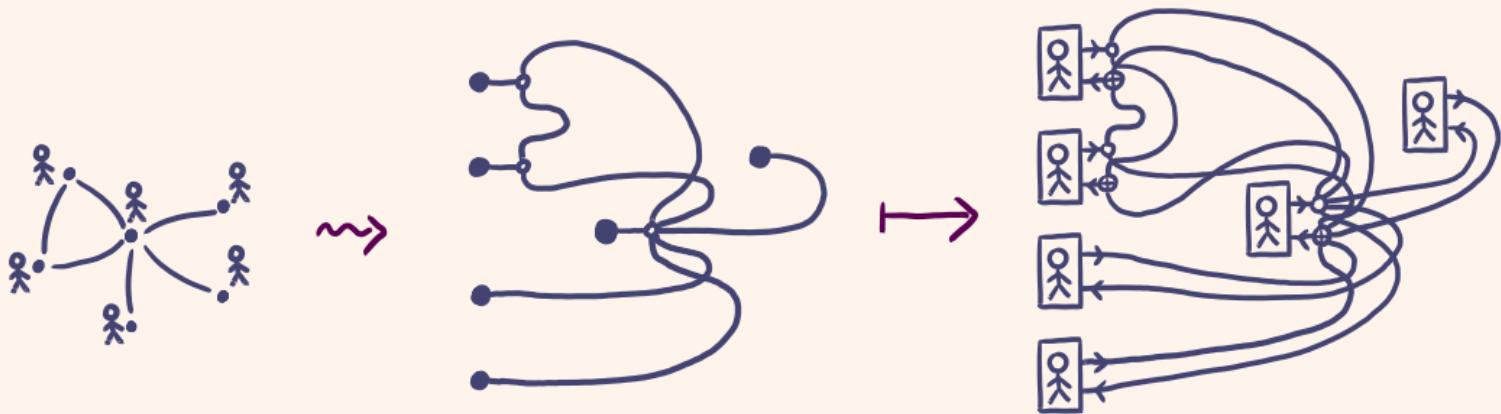
- vertex \mapsto player



- structure \mapsto structure



THE TRAGEDY OF THE COMMONS



$$g(x, y) = \begin{cases} 1 - C + \varepsilon & x, y = 0 \\ 1 & y = 1, x = 0 \\ 1 - C & x = 1 \end{cases}$$

A STANDARD PROCEDURE GRAPHS → GAMES

graph specified with a matrix



graph as a morphism in `cgraph`



monoid network game

game played on the graph as a morphism in `cgame`

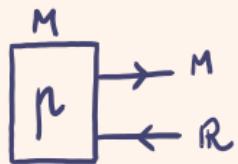
SUMMARY

- The category of graphs with boundaries is the free prop generated by

\exists , \circ , $- \{$, $- \circ$, \exists , \bullet
+ equations

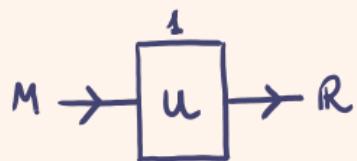
- Monoid network games correspond to functors
 $\text{cgraph} \rightarrow \text{cgame}$

EXAMPLE : A PLAYER



- $P_p(m) = m$
- $C_p(r) = *$
- $B_p(u: M \rightarrow R) = \operatorname{argmax}_{m \in M} u(m)$

EXAMPLE : A (UTILITY) FUNCTION

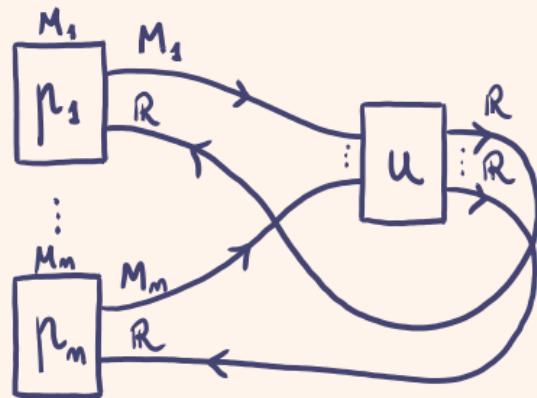


- $P_u(m) = u(m)$

- $C_u(m) = *$

- $B_u(m) = \{ *\}$

NETWORK GAMES IN OPEN GAMES



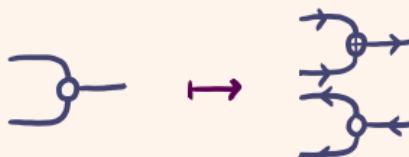
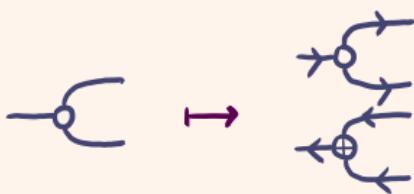
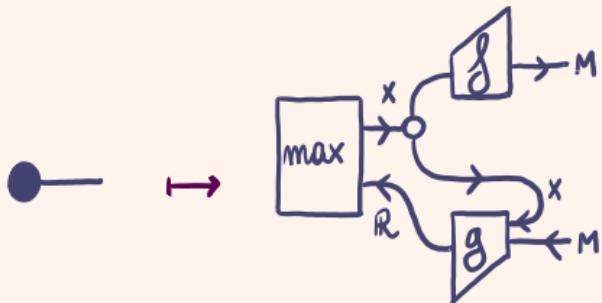
- $p_i : \left(\begin{smallmatrix} 1 \\ 1 \end{smallmatrix}\right) \xrightarrow{M_i} \left(\begin{smallmatrix} M_i \\ R \end{smallmatrix}\right)$ \rightsquigarrow players
- $u : \left(\begin{smallmatrix} M_1 \times \dots \times M_m \\ 1 \end{smallmatrix}\right) \xrightarrow{1} \left(\begin{smallmatrix} R^m \\ 1 \end{smallmatrix}\right)$ \rightsquigarrow utility function

MONOID NETWORK GAMES

- (M, \oplus, e) monoid
- X set
- $f: X \rightarrow M$
- $g: X \times M \rightarrow \mathbb{R}$

$$u_i(G; x_1, \dots, x_n) = g(x_i, \bigoplus_{\substack{j \text{ neighbours} \\ \text{of } i}} f(x_j))$$

MONOID NETWORK GAMES AS FUNCTORS

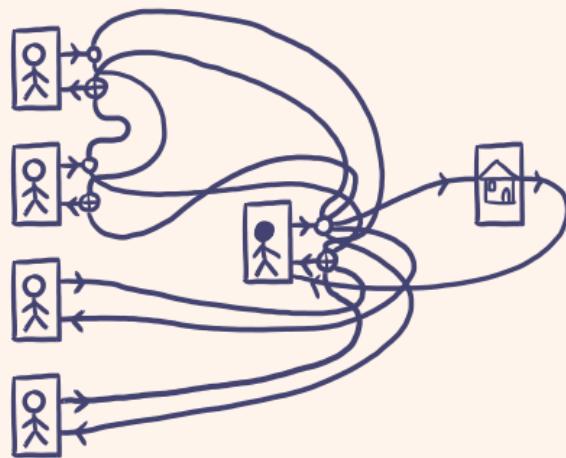
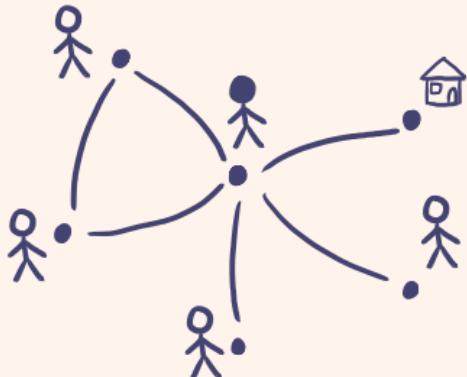


DIRECTED OPEN GRAPHS

Two generators for the objects : \rightarrow , \leftarrow



CHANGING THE INCENTIVES



$$B \rightarrow \boxed{\text{house}} \rightarrow R = \begin{cases} C & x=1 \\ 0 & x=0 \end{cases}$$