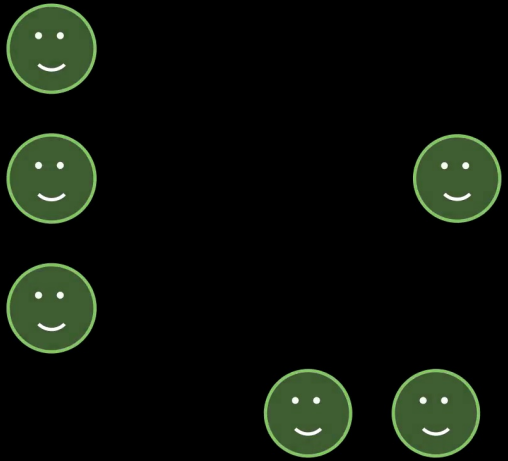
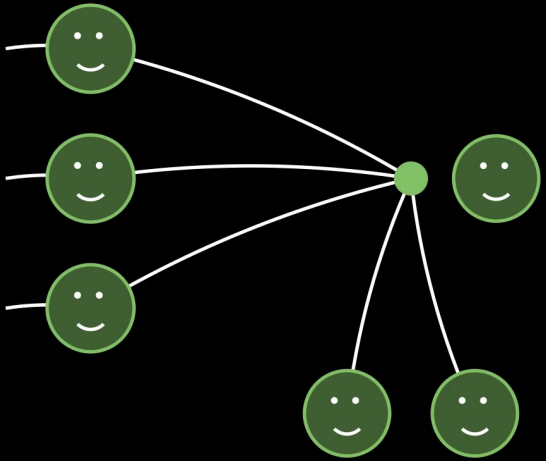
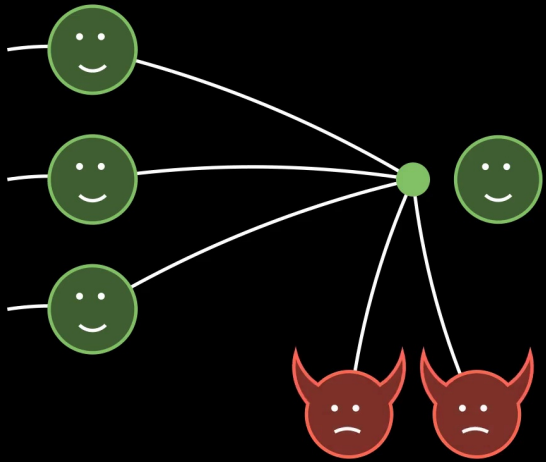
All graph state verification protocols are composably secure

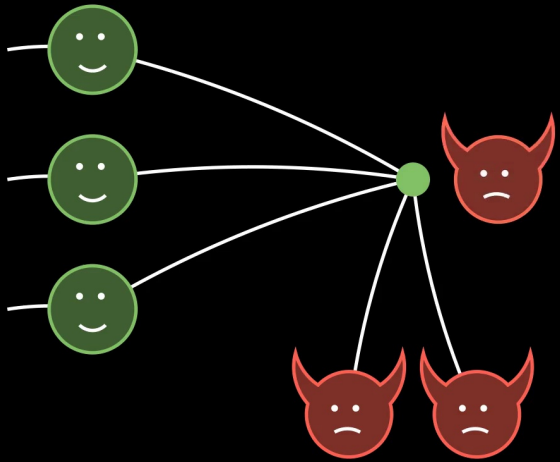
Léo Colisson Palais, Damian Markham, Raja Yehia

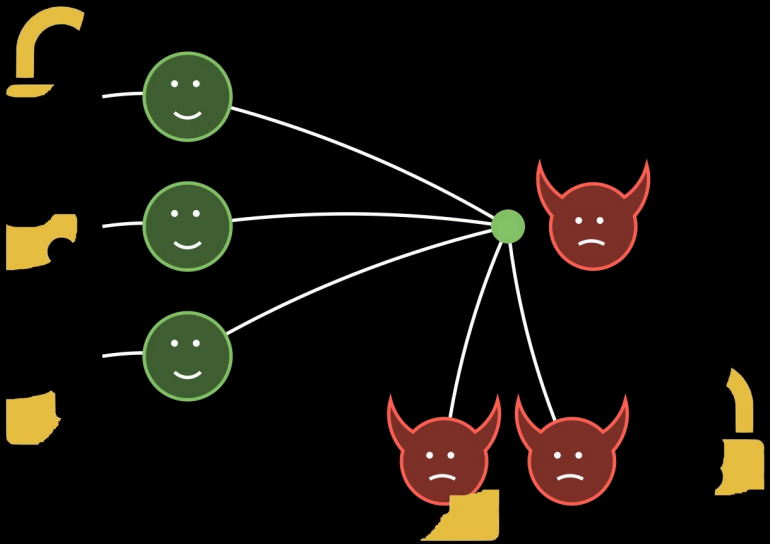
Quantum Cybersecurity Workshop 2025 (Edinburgh)

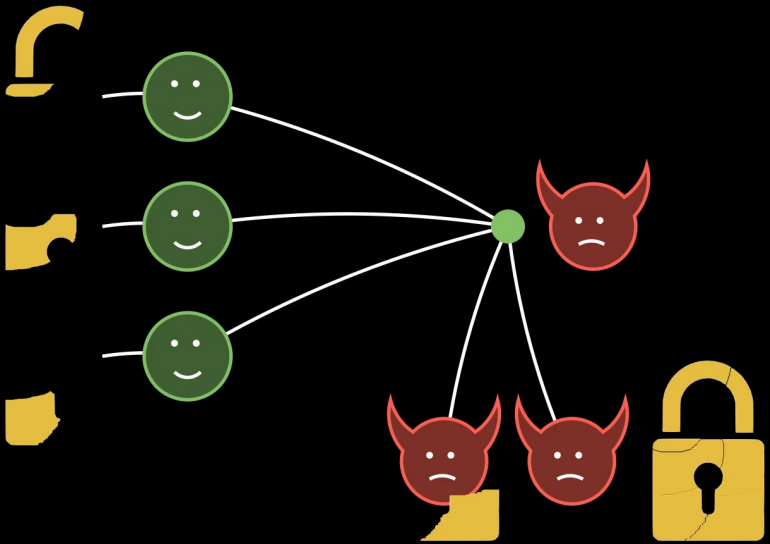










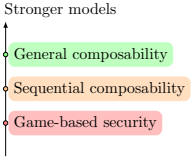


Main question

Are the protocols still secure if we use a state verification protocol instead of a trusted prepared state ?

i.e.

Are state verification protocols composable ?



Previous work

Conjecture from [YDK12]

It is impossible to prove the composable security of state verification protocols.

In this work: we **disprove** this conjecture.

Our results

Theorem 1 (informal)

All graph state verification protocols are composably secure.

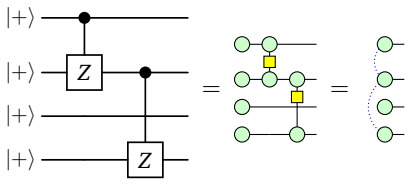
Our results

Theorem 1 (informal)

All graph state verification protocols are composably secure.

Graph state:

Class of quantum states made from $|+\rangle$ and CZ:



E.g.: Bell pairs, GHZ states, etc., are (locally equivalent) to graph states

Our results

Theorem 1 (informal)

All graph state verification protocols are composably secure.

Graph state:

Graph states = starting point of **many protocols**:

- quantum teleportation,
- anonymous transmission,
- Universal Blind Quantum Computing
- …

Our results

Theorem 1 (informal)

All graph state verification protocols can be turned into a composably secure protocol.

Our results

Theorem 1 (informal)

All graph state verification protocols can be turned into a composably secure protocol.

Theorem 2 (informal)

This is optimal for black-box constructions (we either need to change the protocol or the functionality).

Our results

Theorem 1 (informal)

All graph state verification protocols can be turned into a composably secure protocol.

Theorem 2 (informal)

This is optimal for black-box constructions (we either need to change the protocol or the functionality).

Theorem 3 (informal)

Any **unchanged** verification protocol is composably secure with respect to a less natural functionality.

Our results

Theorem 1 (informal)

All graph state verification protocols can be turned into a composably secure protocol.

Theorem 2 (informal)

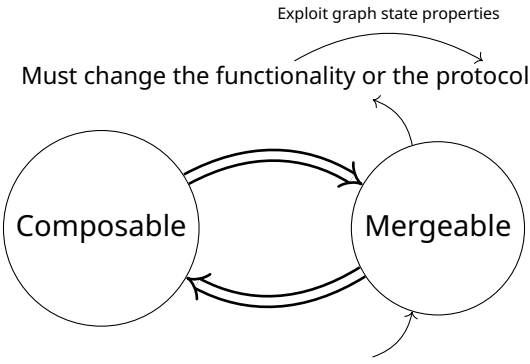
This is optimal for black-box constructions (we either need to change the protocol or the functionality).

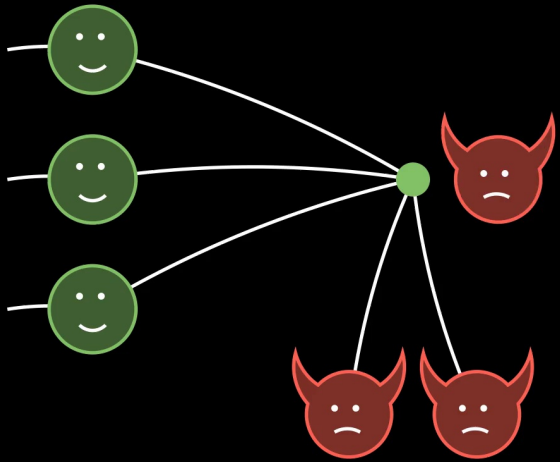
Theorem 3 (informal)

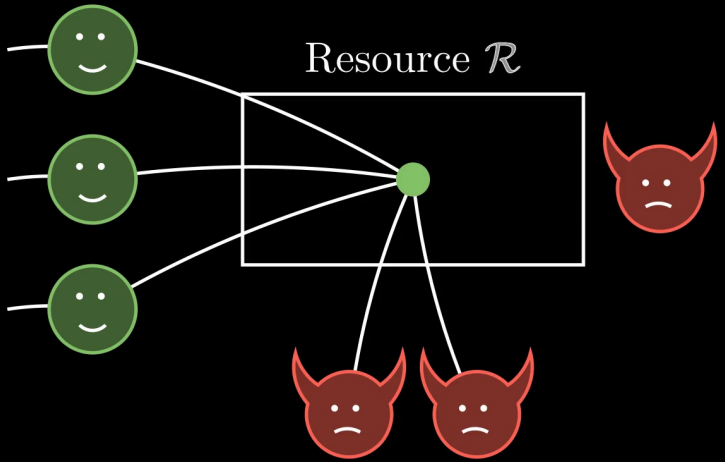
Any **unchanged** verification protocol is composably secure with respect to a less natural functionality.

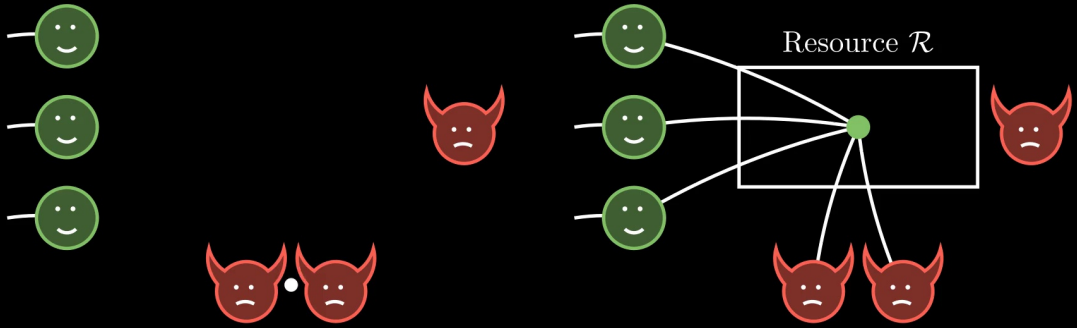
Bonus: we generalize entanglement swapping & we use the scalable ZX-calculus

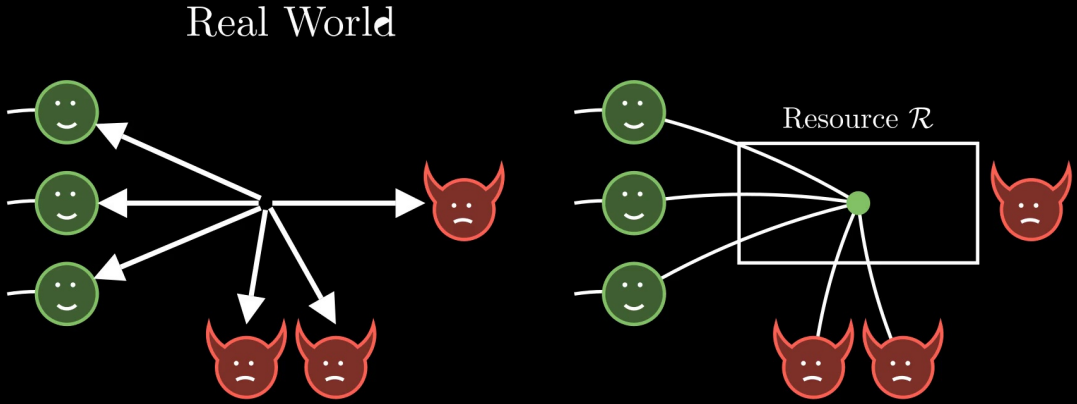
Our approach in a nutshell

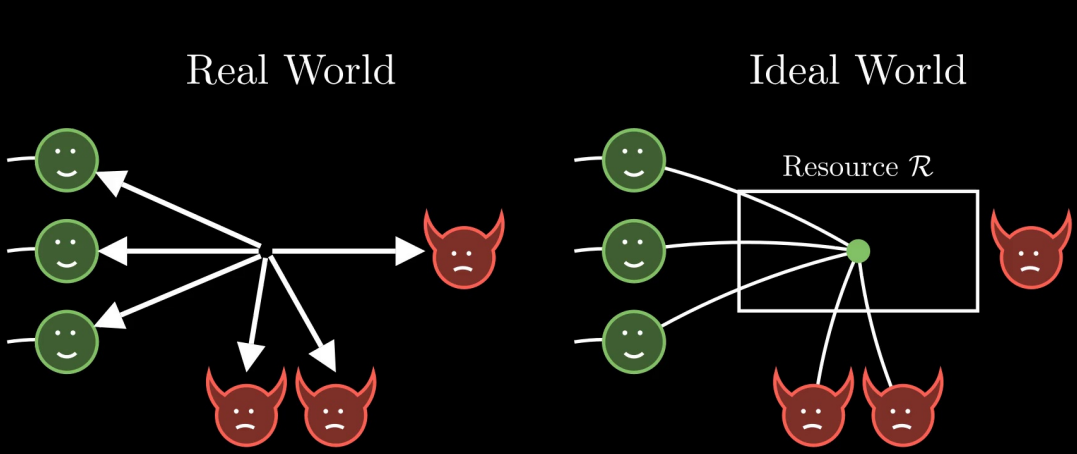


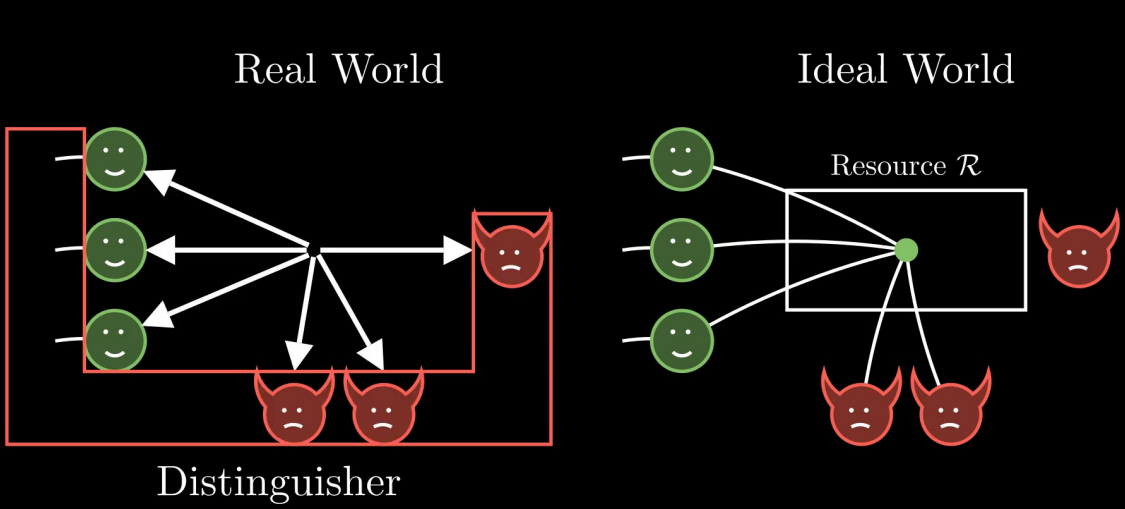


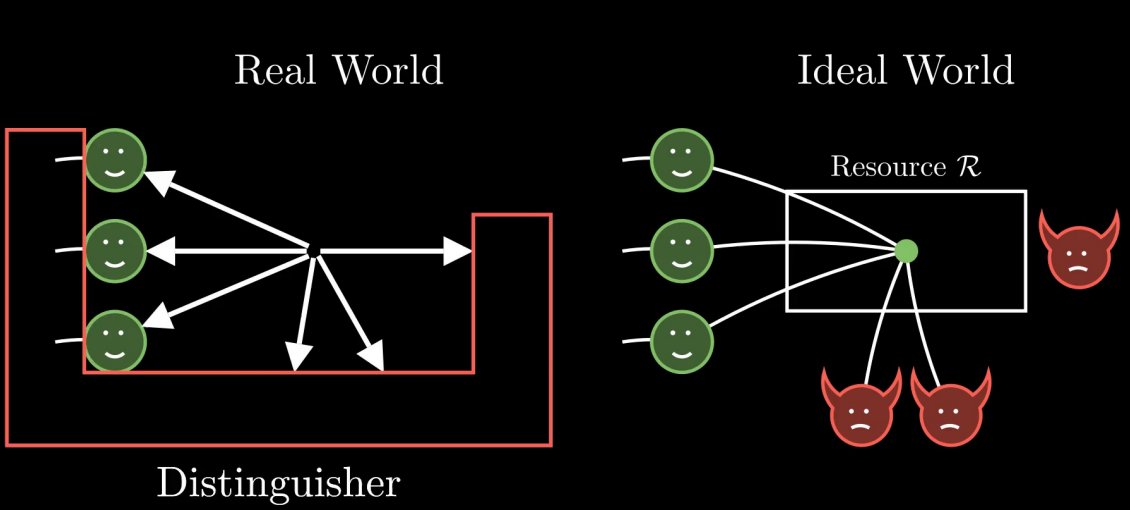

Resource $\mathcal{R}$

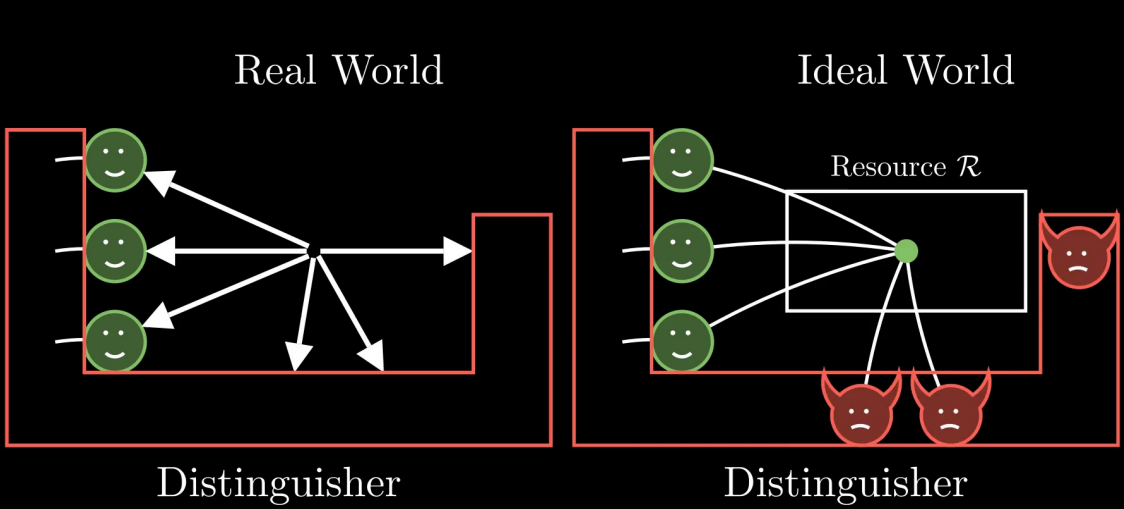

Resource $\mathcal{R}$

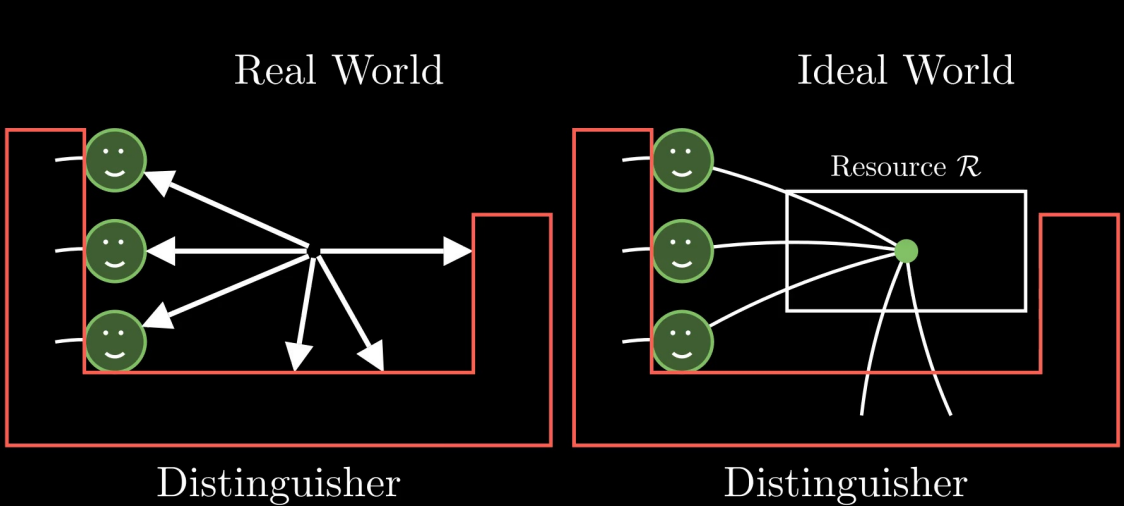

Real World
Resource $\mathcal{R}$

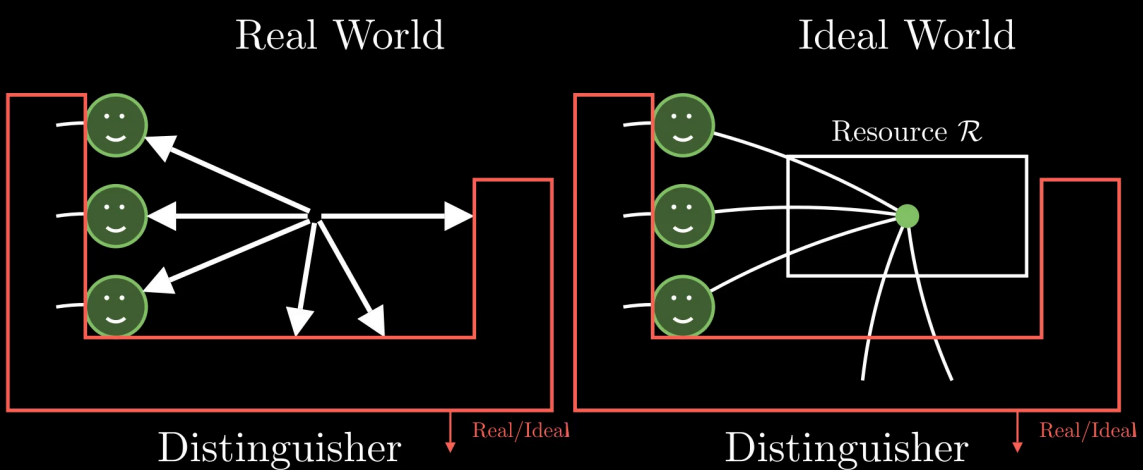

Real World
Ideal World
Resource $\mathcal{R}$

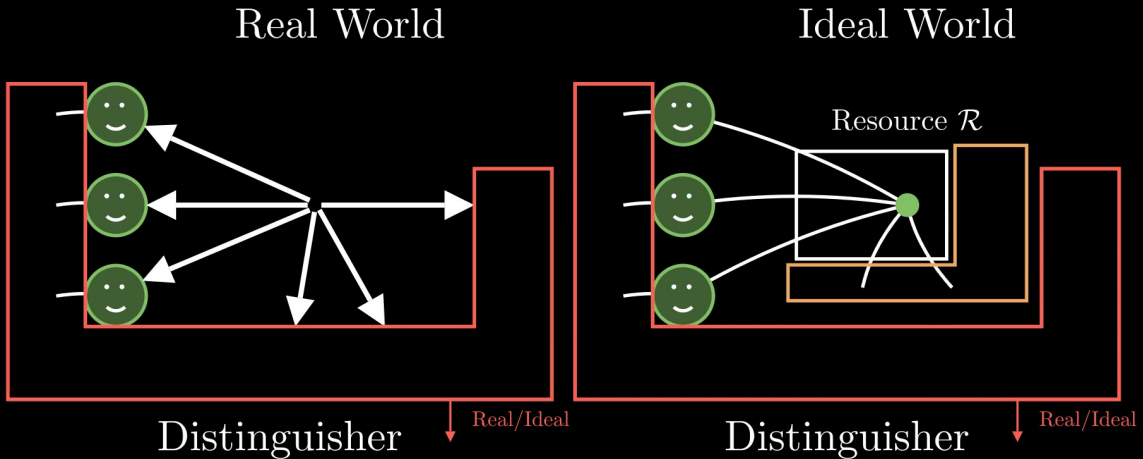

Real World
Ideal World
Resource $\mathcal{R}$
Distinguisher

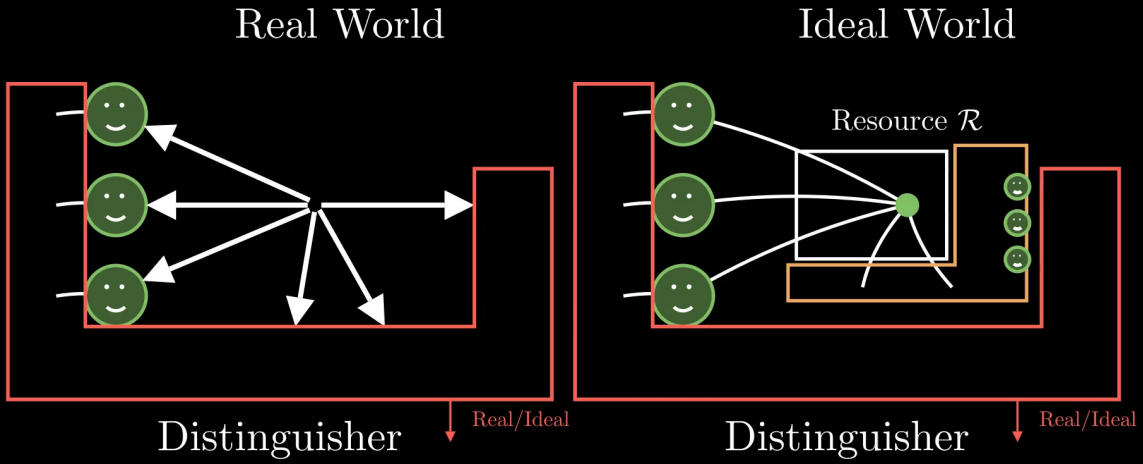

Real World
Ideal World
Resource $\mathcal{R}$
Distinguisher

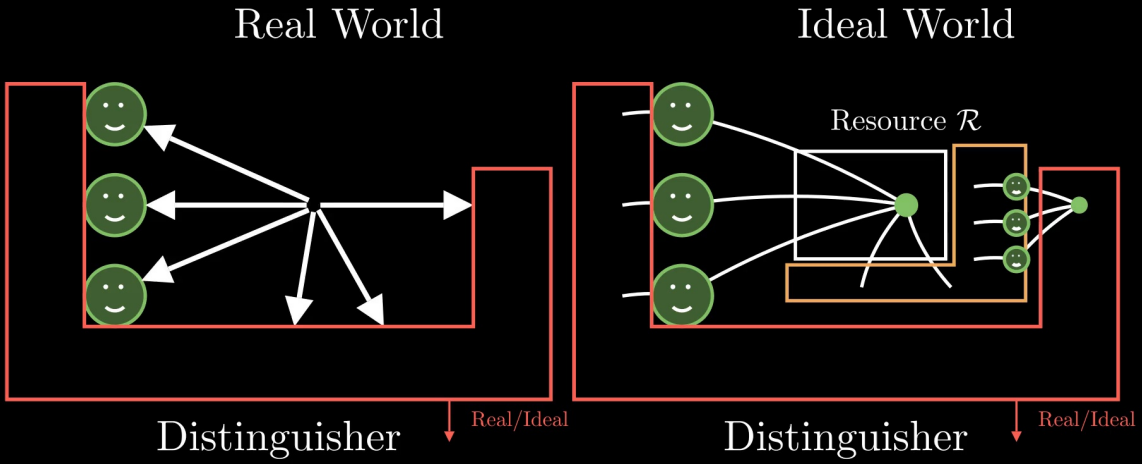

Real World
Ideal World
Resource $\mathcal{R}$
Distinguisher
Distinguisher

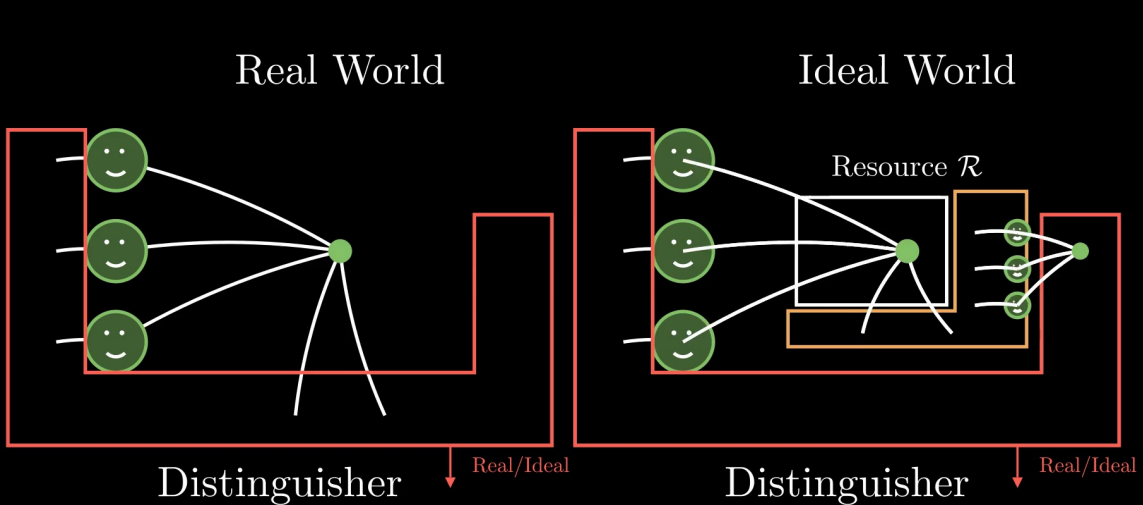

Real World
Ideal World
Resource $\mathcal{R}$
Distinguisher
Distinguisher

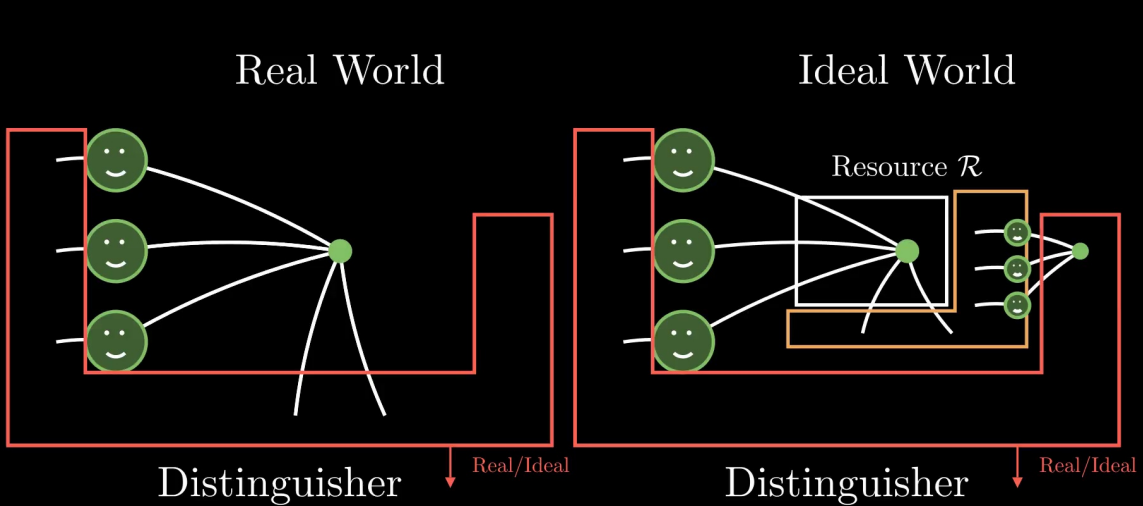

Real World
Ideal World
Resource $\mathcal{R}$
Distinguisher
Real/Ideal
Distinguisher
Real/Ideal


Real World
Ideal World
Resource $\mathcal{R}$
Distinguisher
Real/Ideal
Distinguisher
Real/Ideal


Real World
Ideal World
Resource $\mathcal{R}$
Distinguisher
Real/Ideal
Distinguisher
Real/Ideal

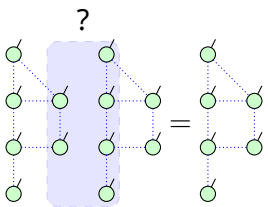

Real World
Ideal World
Resource $\mathcal{R}$
Distinguisher
Real/Ideal
Distinguisher
Real/Ideal


Real World
Ideal World
Resource $\mathcal{R}$
Distinguisher
Real/Ideal
Distinguisher
Real/Ideal


Real World
Ideal World
Resource $\mathcal{R}$
Distinguisher
Real/Ideal
Distinguisher
Real/Ideal

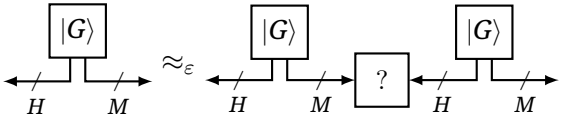
First goal: find a merging map

For correctness: can we find a map ? (run by the simulator) that can merge graph like:



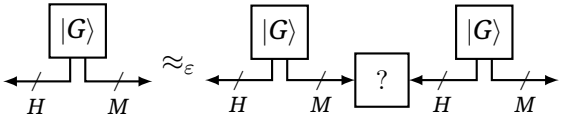
First goal: find a merging map

For correctness: can we find $\boxed{?}$ (run by the simulator) such that:

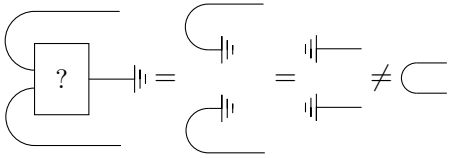


First goal: find a merging map

For correctness: can we find $\boxed{?}$ (run by the simulator) such that:



Impossible! (non-signaling)

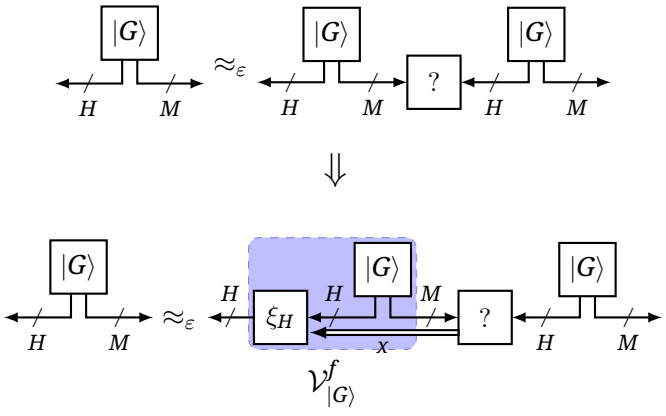


Two possibilities to avoid the no-go theorem

- Option 1: Change the functionality
- Option 2: Change the protocol

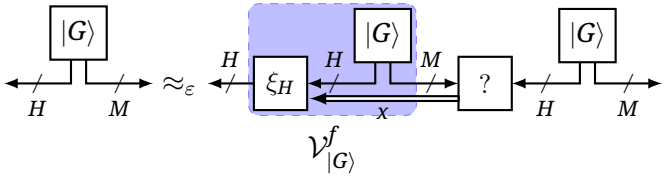
Let’s go for option 1 first!

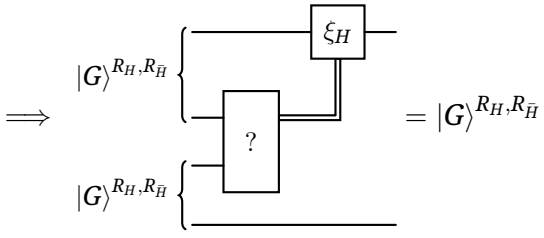
Option 1: change the functionality



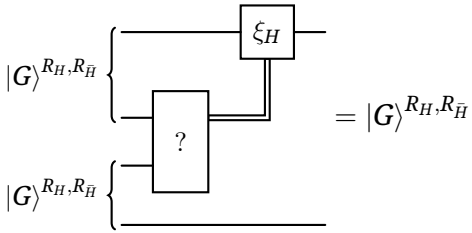
= Mergeable state (f restricts corrections: $f(x)$ must be true)

Mergeable states



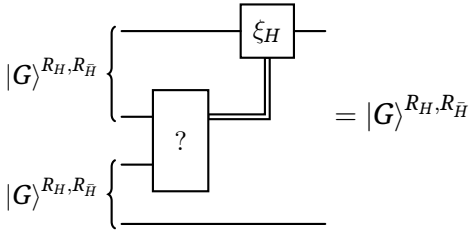


Mergeable states

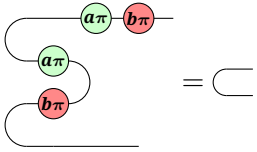


Close to **entanglement swapping** (with additional constraints), but for arbitrary states!

Mergeable states

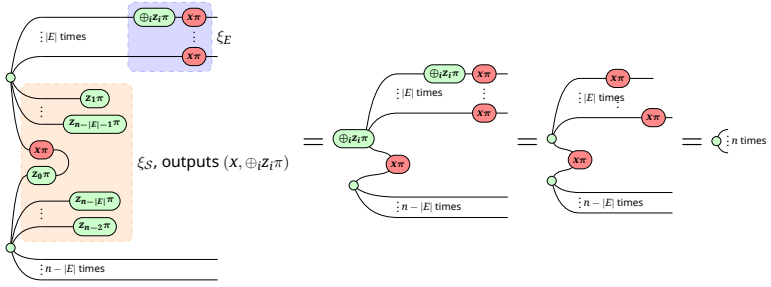


Some states are trivially mergeable, like Bell pairs:



Mergeable states

GHZ states are also easily mergeable:



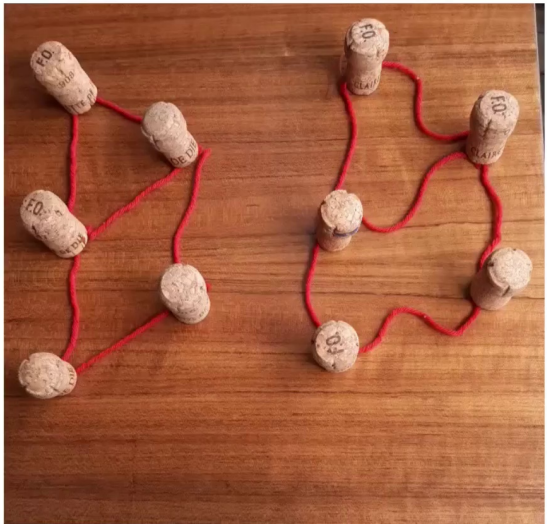
Mergeable states

Are all graph states mergeable?

Mergeable states

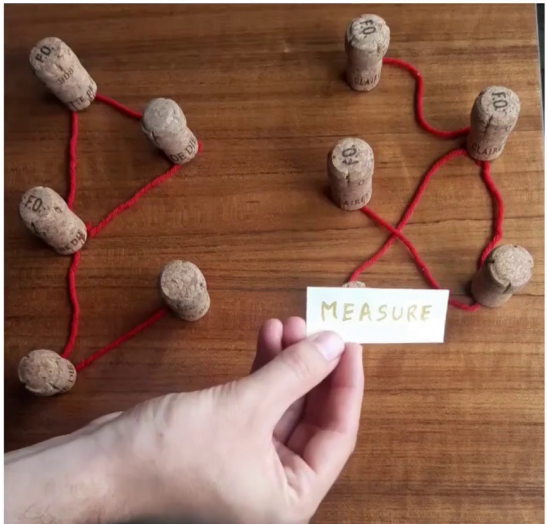
Are all graph states mergeable?

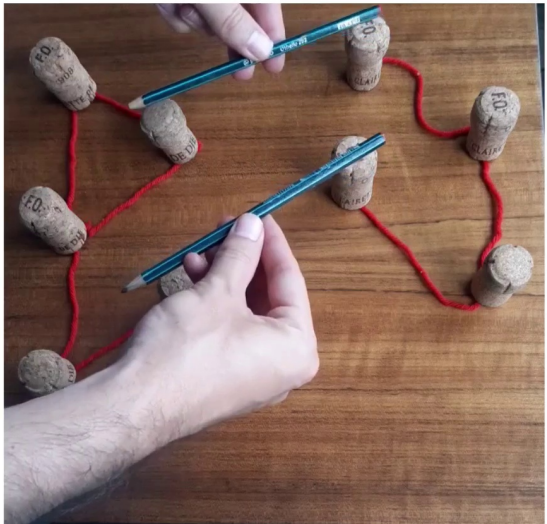
$\Rightarrow$ Our first contribution: **Yes!**




C2

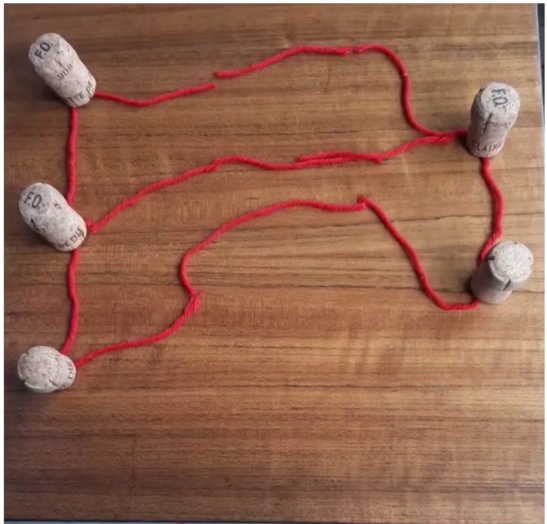

SWAP
&
CNOT


MEASURE




BELL
MEASUREMENT

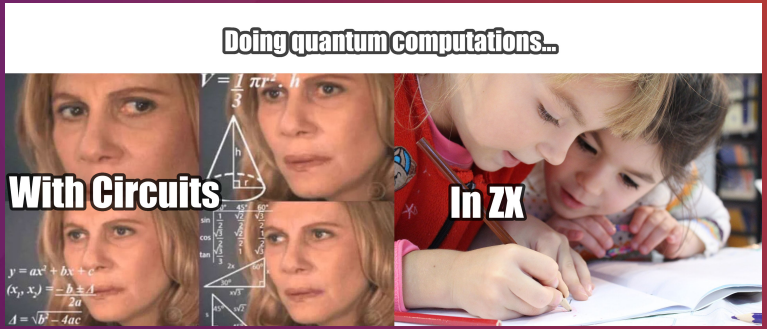

BELL
MEASUREMENT



Merging operation

- How to know which SWAP/CNOT gate to apply?
- Can all the corrections be applied on a single side? (Spoiler: magically, yes)

Introduction to the scalable ZX-calculus



Want to know more?

Full proof:

https://www.youtube.com/watch?v=slZWkYTOKnI

Interpretation of ZX-calculus' generators:

$$[\![\ n \overset{\alpha}{\cdots} m\ ]\!] = |0\rangle^{\otimes m}\langle 0|^{\otimes n} + e^{i\alpha}|1\rangle^{\otimes m}\langle 1|^{\otimes n}$$

$$[\![\ n \overset{\alpha}{\cdots} m\ ]\!] = |+\rangle^{\otimes m}\langle +|^{\otimes n} + e^{i\alpha}|-\rangle^{\otimes m}\langle -|^{\otimes n}$$

$$[\![\ \times\ ]\!] = |00\rangle\langle 00| + |10\rangle\langle 01| + |01\rangle\langle 10| + |11\rangle\langle 11|$$

$$[\![\ -\square- \ ]\!] = |+\rangle\langle 0| + |-\rangle\langle 1|$$

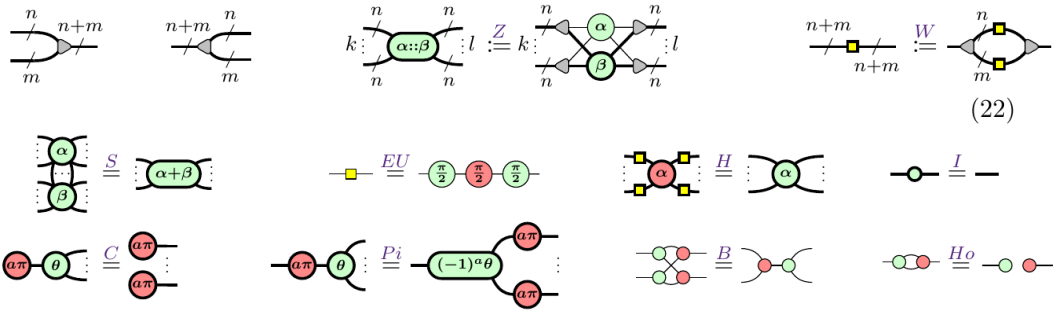
$$[\![\ — \ ]\!] = |0\rangle\langle 0| + |1\rangle\langle 1|$$

$$[\![\ \cap \ ]\!] = |00\rangle + |11\rangle$$

$$[\![\ \cup \ ]\!] = \langle 00| + \langle 11|$$

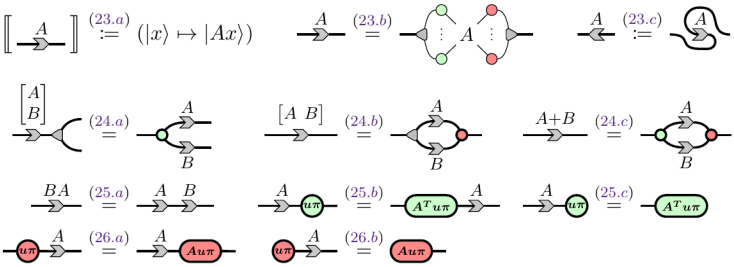
$$[\![\ \ ]\!] = (1)$$

Introduction to scalable ZX-calculus



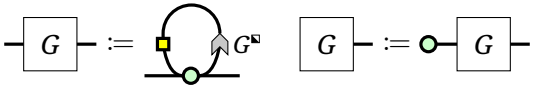
Introduction to scalable ZX-calculus

Matrices:

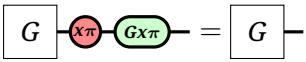


Introduction to the scalable ZX-calculus

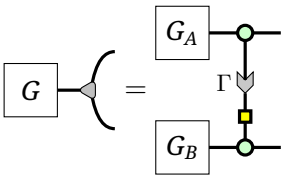
Let $G = \begin{bmatrix} G_A & \Gamma^T \\ \Gamma & G_B \end{bmatrix}$ be the adjacency matrix of a graph G, then:



Stabilizer:



Split a Graph state in 2 parts:



How to merge any graph state

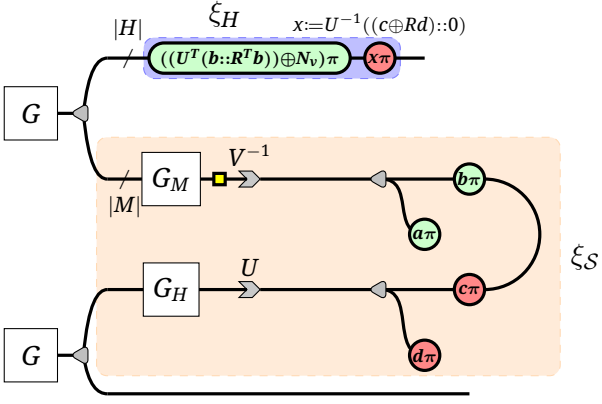
Merging graph states: step 1

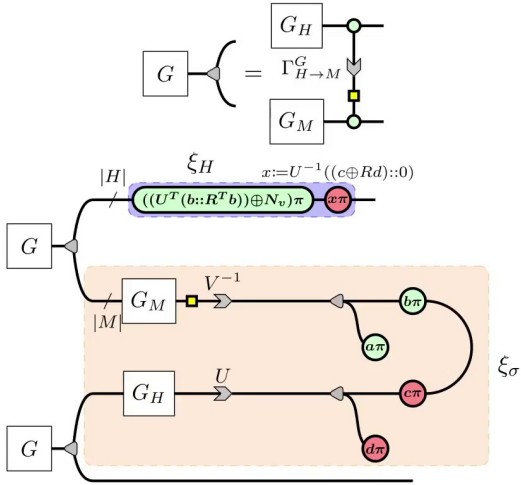
Step 1: If $G = \begin{bmatrix} G_A & \Gamma^T \\ \Gamma & G_B \end{bmatrix}$, use Gaussian elimination on Γ to find $r \in \mathbb{N}$, U, V, and R such that:

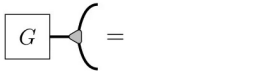
$$\Gamma = V \begin{bmatrix} I_r & R \\ \mathbf{0} & \mathbf{0} \end{bmatrix} U \tag{1}$$

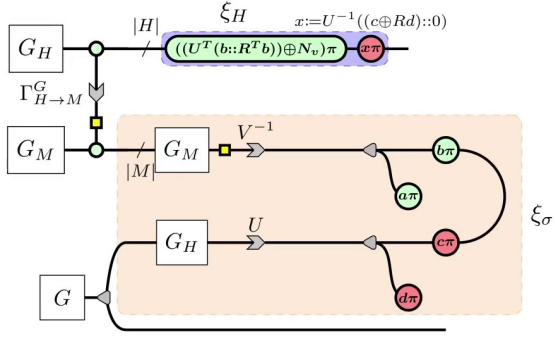
Merging graph states: step 2

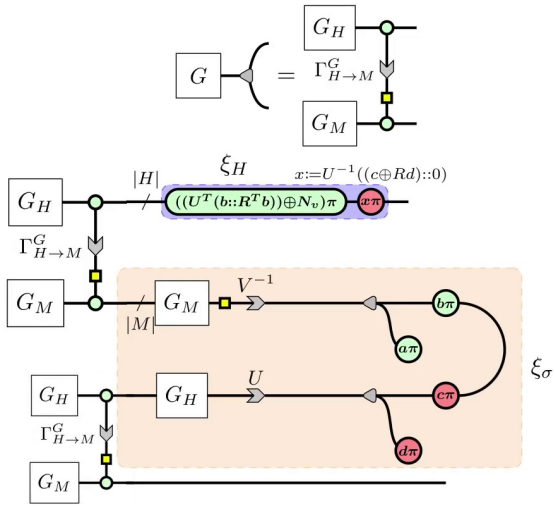
Claim: this map merges the graph states:

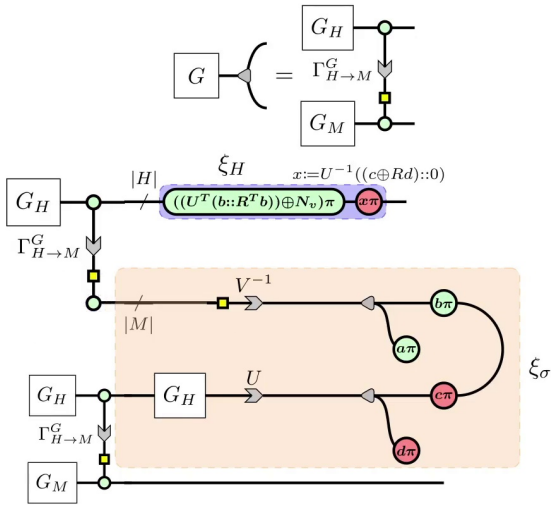


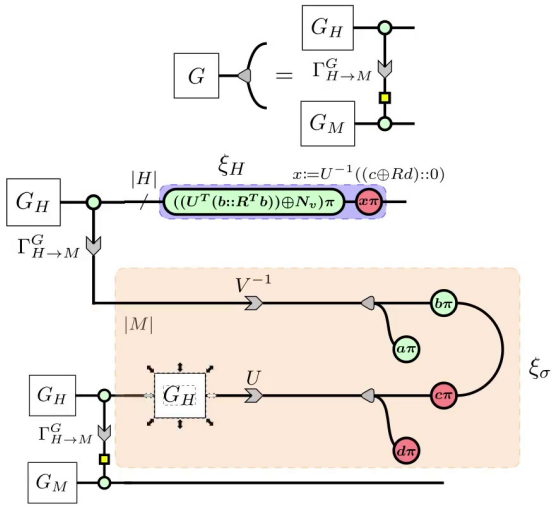

G
$=$
$\Gamma^{G}_{H\to M}$
G_H
G_M
ξ_H
$|H|$
$x{:=}U^{-1}((c\oplus Rd)::0)$
$((U^T(b::R^Tb))\oplus N_v)\pi$
$x\pi$
G
$|M|$
G_M
V^{-1}
$b\pi$
$a\pi$
ξ_σ
G_H
U
$c\pi$
G
$d\pi$

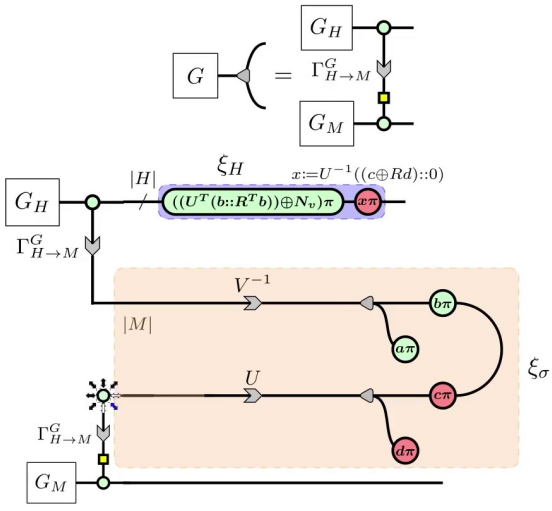

G
=

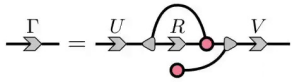

G_H
$|H|$
ξ_H
$x{:=}U^{-1}((c \oplus Rd)::0)$
$((U^T(b::R^Tb)) \oplus N_v)\pi$
$x\pi$
$\Gamma^G_{H\to M}$
G_M
$|M|$
G_M
V^{-1}
$b\pi$
$a\pi$
G_H
U
$c\pi$
$d\pi$
G
ξ_σ

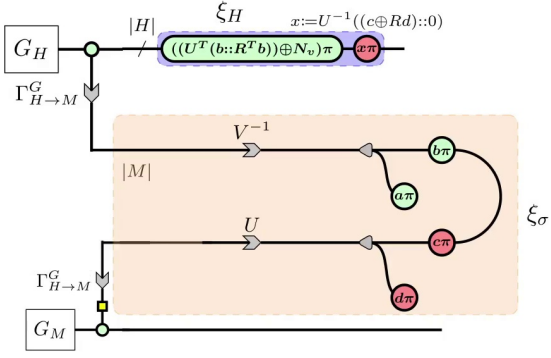

G
$=$
G_H
$\Gamma^G_{H\to M}$
G_M
ξ_H
$|H|$
$x{:=}U^{-1}((c\oplus Rd)::0)$
G_H
$((U^T(b::R^Tb))\oplus N_v)\pi$
$x\pi$
$\Gamma^G_{H\to M}$
G_M
G_M
V^{-1}
$|M|$
$b\pi$
$a\pi$
ξ_σ
G_H
G_H
U
$c\pi$
$\Gamma^G_{H\to M}$
$d\pi$
G_M

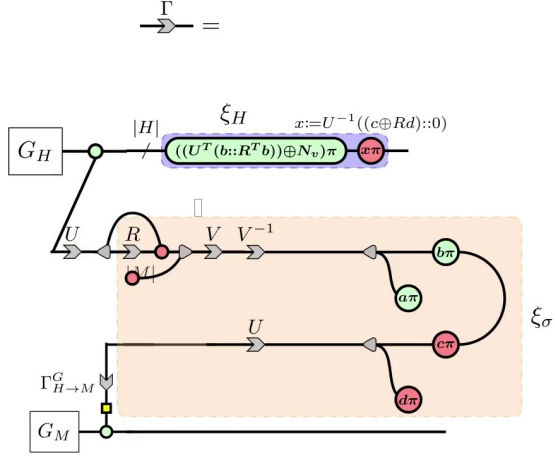

G
$=$
G_H
$\Gamma^G_{H\to M}$
G_M
ξ_H
$|H|$
$x{:=}U^{-1}((c\oplus Rd)::0)$
$((U^T(b::R^Tb))\oplus N_v)\pi$
$x\pi$
V^{-1}
$|M|$
$b\pi$
$a\pi$
U
$c\pi$
$d\pi$
ξ_σ

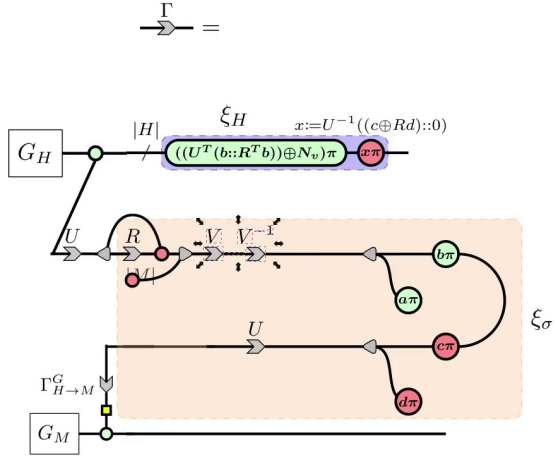

G
$=$
G_H
$\Gamma^G_{H\to M}$
G_M
ξ_H
$|H|$
$x{:=}U^{-1}((c\oplus Rd)::0)$
G_H
$((U^T(b::R^Tb))\oplus N_v)\pi$
$x\pi$
$\Gamma^G_{H\to M}$
V^{-1}
$|M|$
$b\pi$
$a\pi$
ξ_σ
G_H
G_H
U
$c\pi$
$\Gamma^G_{H\to M}$
$d\pi$
G_M

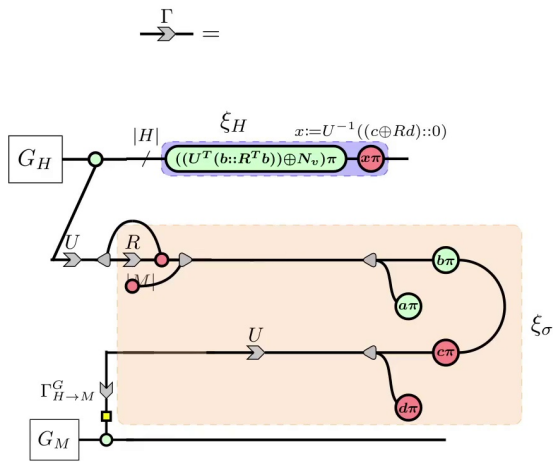

G
$=$
G_H
$\Gamma^G_{H\to M}$
G_M
ξ_H
$|H|$
$x{:=}U^{-1}((c{\oplus}Rd)::0)$
G_H
$((U^T(b::R^Tb))\oplus N_v)\pi$
$x\pi$
$\Gamma^G_{H\to M}$
V^{-1}
$|M|$
$b\pi$
$a\pi$
ξ_σ
U
$c\pi$
$\Gamma^G_{H\to M}$
$d\pi$
G_M

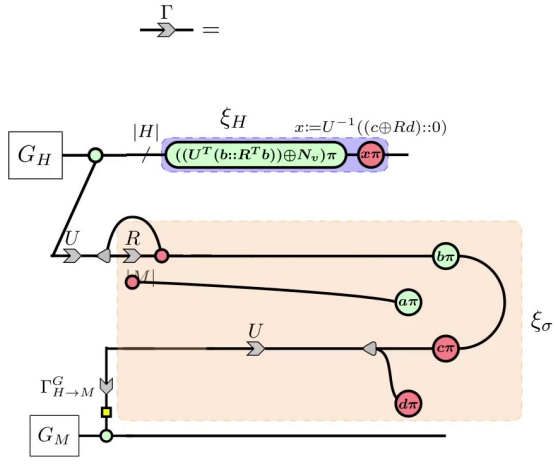

Γ
=
U
R
V

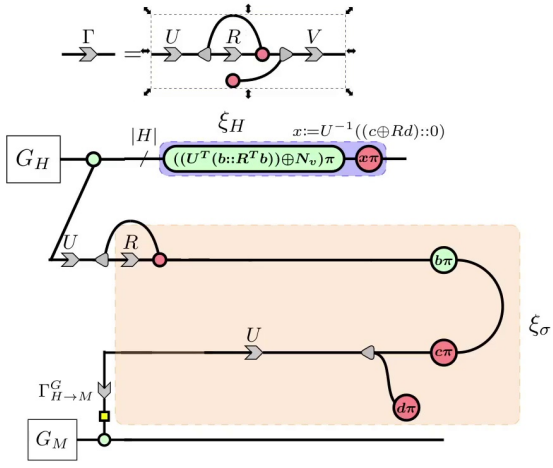

G_H
$|H|$
ξ_H
$x{:=}U^{-1}((c \oplus Rd)::0)$
$((U^T(b::R^Tb)) \oplus N_v)\pi$
$x\pi$
$\Gamma^G_{H\to M}$
V^{-1}
$|M|$
$b\pi$
$a\pi$
ξ_σ
U
$c\pi$
$\Gamma^G_{H\to M}$
$d\pi$
G_M

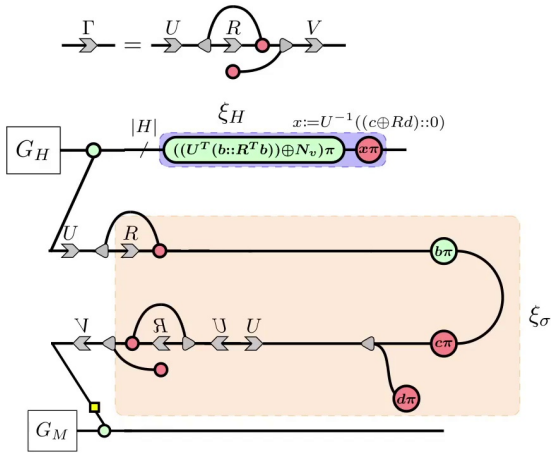

Γ =
ξ_H
$x{:=}U^{-1}((c \oplus Rd)::0)$
G_H
$|H|$
$((U^T(b::R^Tb)) \oplus N_v)\pi$
$x\pi$
U
R
V
V^{-1}
$|H|$
$b\pi$
$a\pi$
ξ_σ
U
$c\pi$
$\Gamma^G_{H\to M}$
$d\pi$
G_M

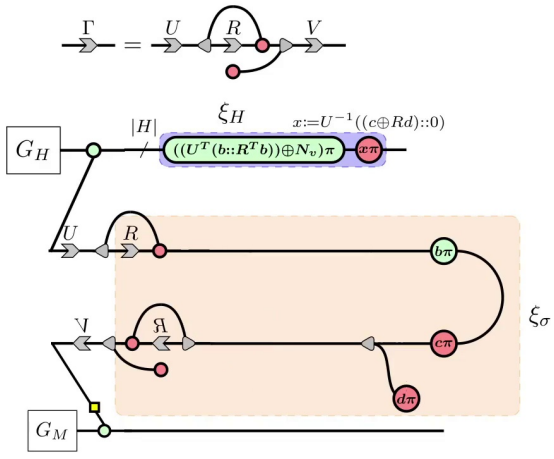

Γ
=
G_H
$|H|$
ξ_H
$x := U^{-1}((c \oplus Rd)::0)$
$((U^T(b::R^Tb)) \oplus N_v)\pi$
$x\pi$
U
R
V
V^{-1}
$b\pi$
$a\pi$
ξ_σ
U
$c\pi$
$d\pi$
$\Gamma^G_{H\to M}$
G_M

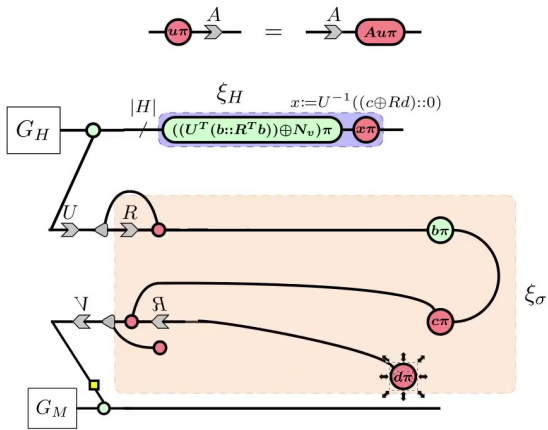

Γ =
G_H
$|H|$
ξ_H
$x{:=}U^{-1}((c \oplus Rd)::0)$
$((U^T(b::R^Tb)) \oplus N_v)\pi$
$x\pi$
U
R
$b\pi$
$a\pi$
ξ_σ
U
$c\pi$
$\Gamma^G_{H\to M}$
$d\pi$
G_M

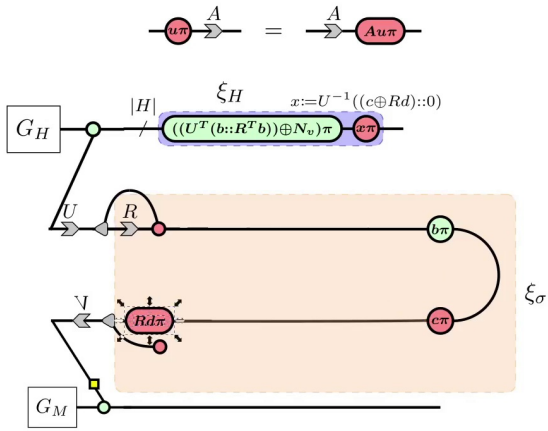

Γ =
ξ_H
$x := U^{-1}((c \oplus Rd) :: 0)$
G_H
$|H|$
$((U^T(b :: R^T b)) \oplus N_v)\pi$
$x\pi$
U
R
$b\pi$
$a\pi$
ξ_σ
U
$c\pi$
$d\pi$
$\Gamma^G_{H \to M}$
G_M

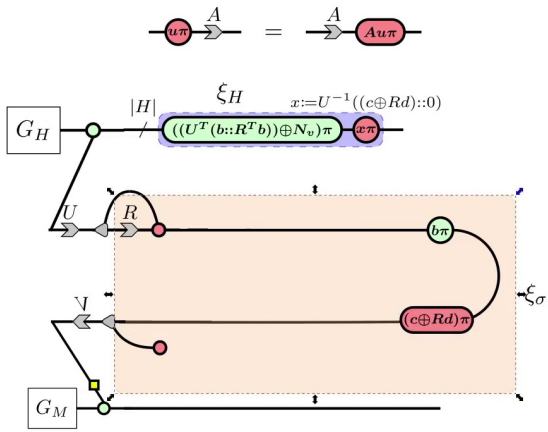

Γ
=
U
R
V
ξ_H
$x{:=}U^{-1}((c \oplus Rd)::0)$
G_H
$|H|$
$((U^T(b::R^Tb)) \oplus N_v)\pi$
$x\pi$
U
R
$b\pi$
ξ_σ
U
$c\pi$
$d\pi$
$\Gamma^G_{H\to M}$
G_M

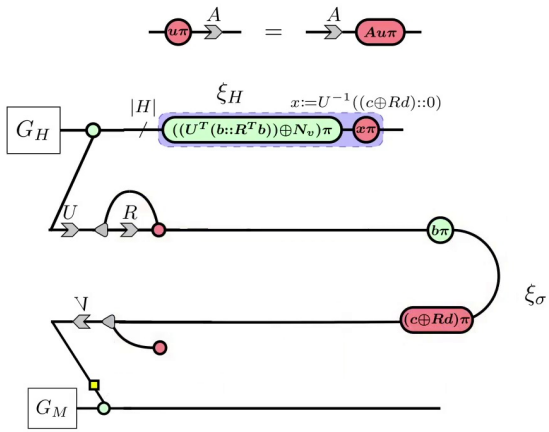

Γ
=
U
R
V
ξ_H
$x{:=}U^{-1}((c \oplus Rd)::0)$
G_H
$|H|$
$((U^T(b::R^Tb)) \oplus N_v)\pi$
$x\pi$
U
R
$b\pi$
ξ_σ
$c\pi$
$d\pi$
G_M

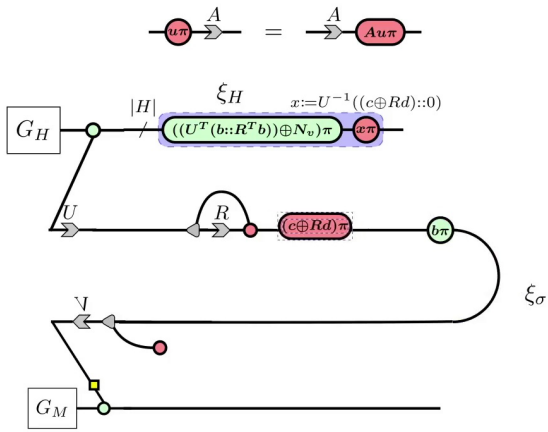

Γ
=
U
R
V
ξ_H
$x{:=}U^{-1}((c \oplus Rd)::0)$
G_H
$|H|$
$((U^T(b::R^Tb)) \oplus N_v)\pi$
$x\pi$
U
R
$b\pi$
ξ_σ
V
R
$c\pi$
$d\pi$
G_M

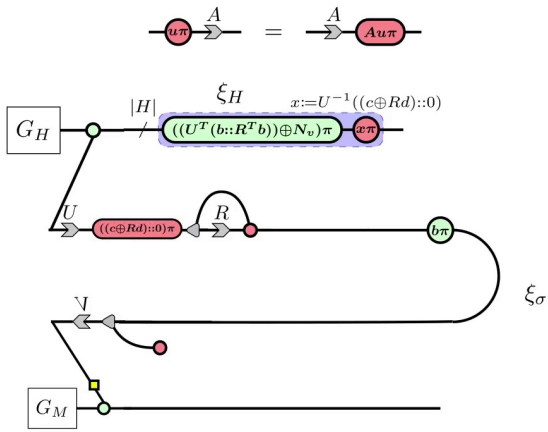

A
uπ
=
A
Auπ
ξH
|H|
x:=U^{-1}((c⊕Rd)::0)
G_H
((U^T(b::R^T b))⊕N_v)π
xπ
U
R
bπ
cπ
dπ
ξσ
G_M

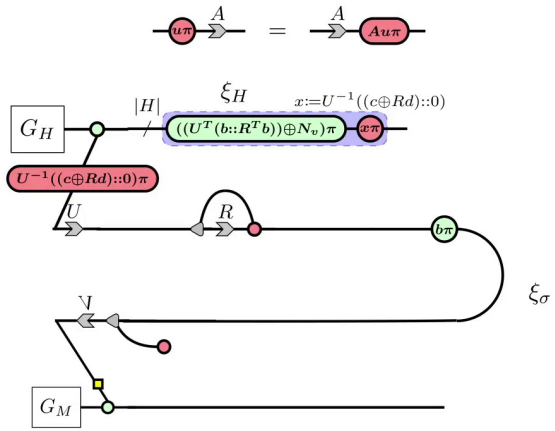

A
$u\pi$
=
A
$Au\pi$
ξ_H
$|H|$
$x:=U^{-1}((c \oplus Rd)::0)$
G_H
$((U^T(b::R^Tb)) \oplus N_v)\pi$
$x\pi$
U
R
$b\pi$
ξ_σ
V
$Rd\pi$
$c\pi$
G_M

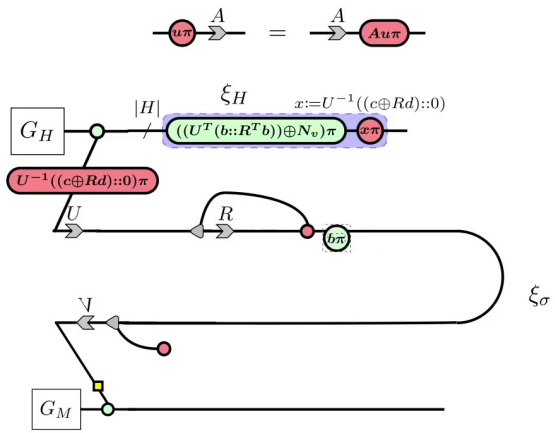

A
uπ
=
A
Auπ
ξH
|H|
x:=U⁻¹((c⊕Rd)::0)
GH
((UT(b::RTb))⊕Nv)π
xπ
U
R
bπ
ξσ
(c⊕Rd)π
GM

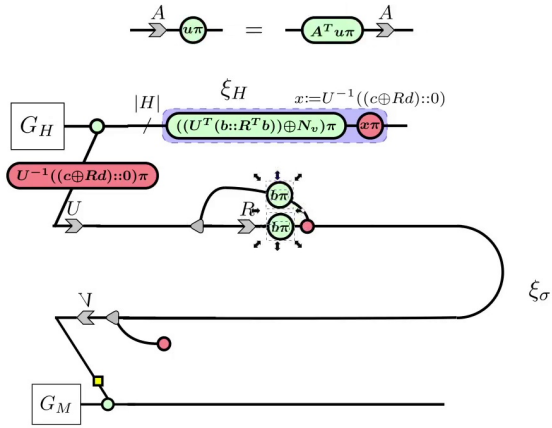

A
$u\pi$
=
A
$Au\pi$
ξ_H
$|H|$
$x:=U^{-1}((c\oplus Rd)::0)$
G_H
$((U^T(b::R^Tb))\oplus N_v)\pi$
$x\pi$
U
R
$b\pi$
ξ_σ
$(c\oplus Rd)\pi$
G_M

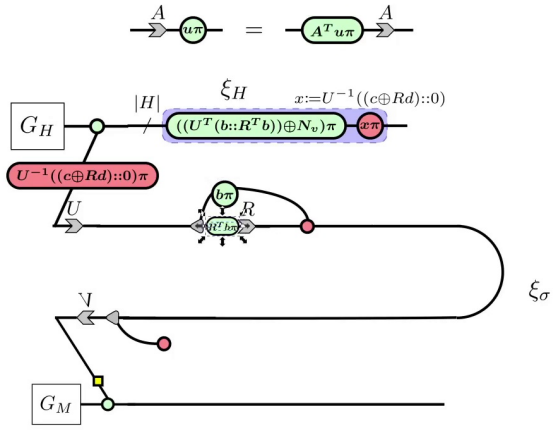

A
$u\pi$
=
A
$Au\pi$
ξ_H
$|H|$
$x{:=}U^{-1}((c \oplus Rd)::0)$
G_H
$((U^T(b::R^Tb)) \oplus N_v)\pi$
$x\pi$
U
R
$(c \oplus Rd)\pi$
$b\pi$
ξ_σ
G_M

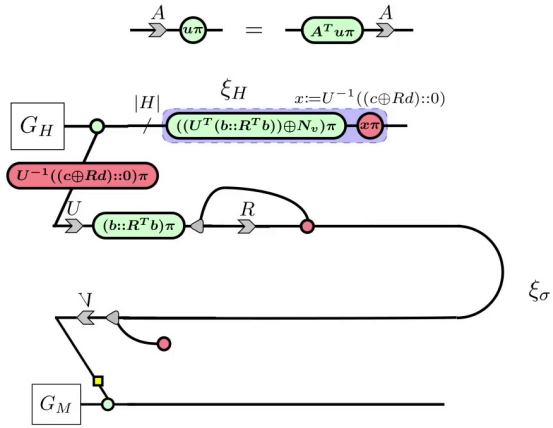

A
$u\pi$
=
A
$Au\pi$
ξ_H
$|H|$
$x{:=}U^{-1}((c\oplus Rd)::0)$
G_H
$((U^T(b::R^Tb))\oplus N_v)\pi$
$x\pi$
U
$((c\oplus Rd)::0)\pi$
R
$b\pi$
ξ_σ
M
G_M

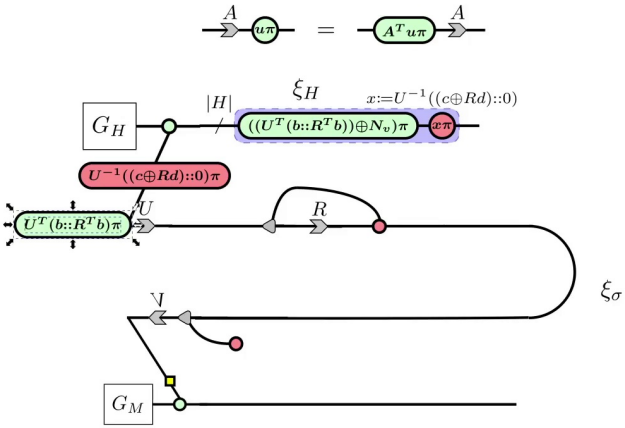

A
$u\pi$
=
A
$Au\pi$
ξ_H
$|H|$
$x := U^{-1}((c \oplus Rd)::0)$
G_H
$((U^T(b::R^T b)) \oplus N_v)\pi$
$x\pi$
$U^{-1}((c \oplus Rd)::0)\pi$
U
R
$b\pi$
ξ_σ
M
G_M

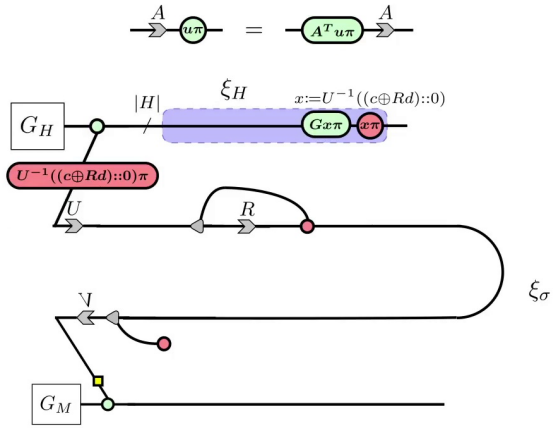

A
$u\pi$
=
A
$Au\pi$
ξ_H
$|H|$
$x := U^{-1}((c \oplus Rd) :: 0)$
G_H
$((U^T(b :: R^T b)) \oplus N_v)\pi$
$x\pi$
$U^{-1}((c \oplus Rd) :: 0)\pi$
U
R
$b\pi$
ξ_σ
G_M

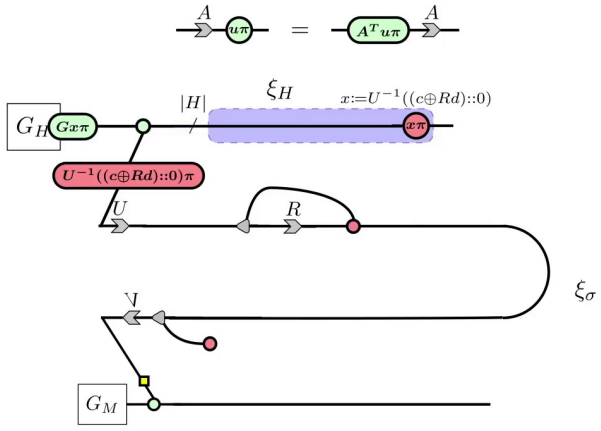

A
uπ
=
A^T uπ
A
ξ_H
|H|
x:=U^{-1}((c⊕Rd)::0)
G_H
((U^T(b::R^T b))⊕N_v)π
xπ
U^{-1}((c⊕Rd)::0)π
U
R
bπ
bπ
ξ_σ
M
G_M

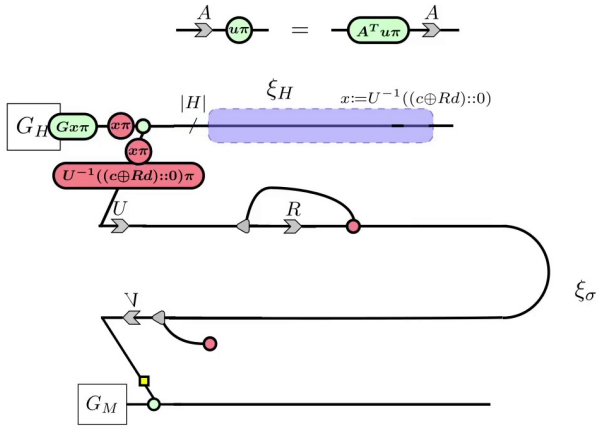

A
uπ
=
A^T uπ
A
ξ_H
|H|
x:=U^{-1}((c⊕Rd)::0)
G_H
((U^T(b::R^T b))⊕N_v)π
xπ
U^{-1}((c⊕Rd)::0)π
bπ
U
R
R^T bπ
ξ_σ
G_M

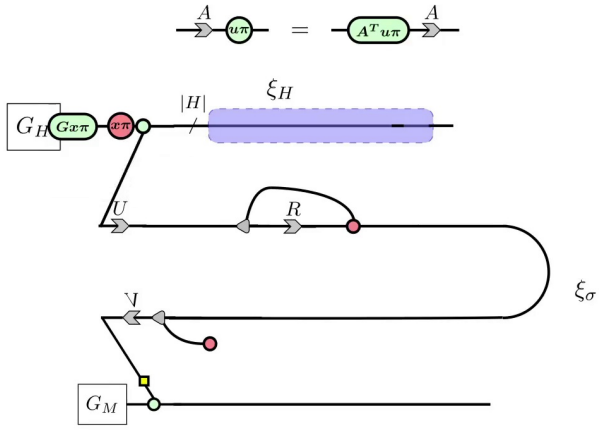

A
uπ
=
A^T uπ
A
ξ_H
|H|
x:=U^{-1}((c⊕Rd)::0)
G_H
((U^T(b::R^T b))⊕N_v)π
xπ
U^{-1}((c⊕Rd)::0)π
U
(b::R^T b)π
R
ξ_σ
G_M

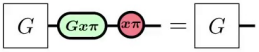

A
uπ
=
A^T uπ
A
ξ_H
|H|
x:=U^{-1}((c⊕Rd)::0)
G_H
((U^T(b::R^T b))⊕N_v)π
xπ
U^{-1}((c⊕Rd)::0)π
U^T(b::R^T b)π
U
R
ξ_σ
G_M

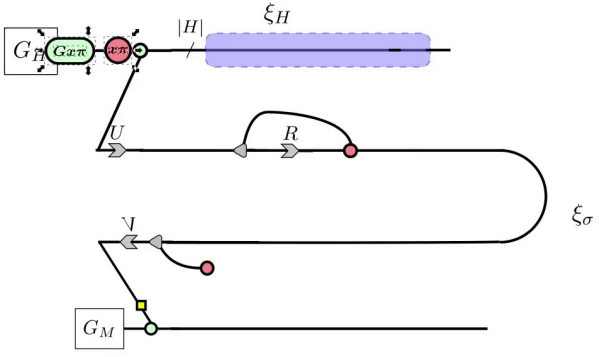

A
uπ
=
A^T uπ
A
ξ_H
|H|
x:=U^{-1}((c⊕Rd)::0)
G_H
Gxπ
xπ
U^{-1}((c⊕Rd)::0)π
U
R
ξ_σ
M
G_M

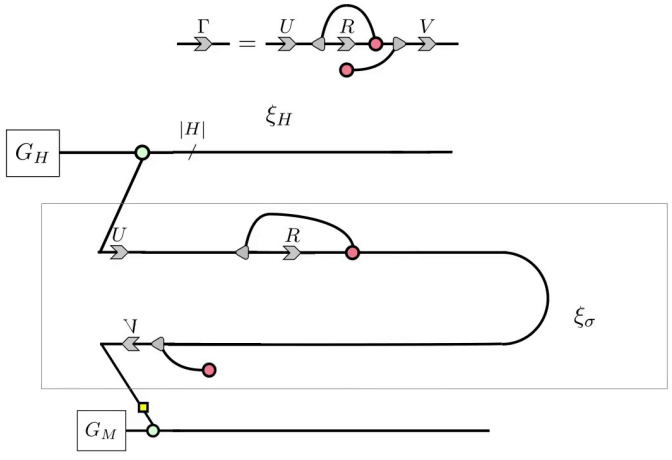

A
uπ
=
A^T uπ
A
ξ_H
|H|
x:=U^{-1}((c⊕Rd)::0)
G_H
Gxπ
xπ
U^{-1}((c⊕Rd)::0)π
U
R
ξ_σ
G_M

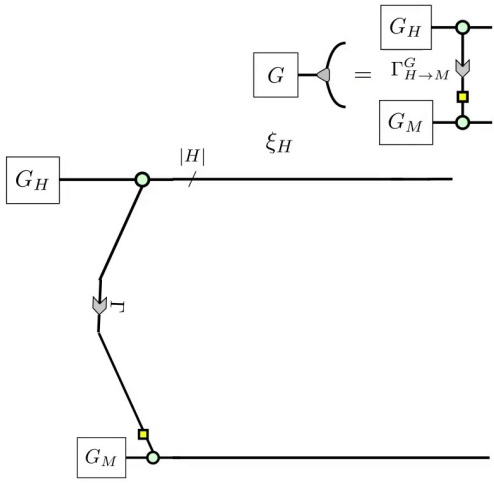

A
uπ
=
A^T uπ
A
ξ_H
|H|
x:=U^{-1}((c⊕Rd)::0)
G_H
Gxπ
xπ
xπ
U^{-1}((c⊕Rd)::0)π
U
R
ξ_σ
M
G_M

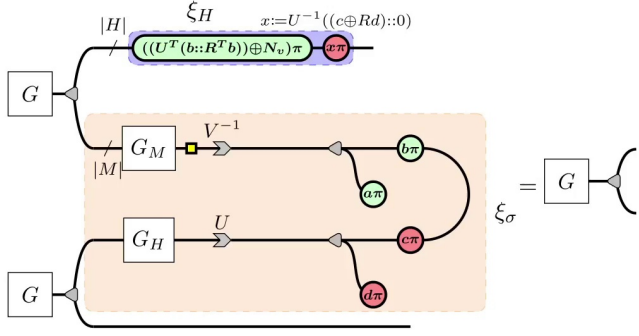

A
uπ
=
A^T uπ
A
ξ_H
|H|
G_H
Gxπ
xπ
U
R
ξ_σ
M
G_M


G
Gxπ
xπ
=
G


ξ_H
$|H|$
G_H
$Gx\pi$
$x\pi$
U
R
ξ_σ
N
G_M


Γ
=
U
R
V
ξ_H
G_H
$|H|$
U
R
V
ξ_σ
G_M


G_H
G
$=$
$\Gamma^G_{H\to M}$
G_M
ξ_H
G_H
$|H|$
Γ
G_M


ξ_H
$|H|$
$x{:=}U^{-1}((c \oplus Rd)::0)$
$((U^T(b::R^Tb)) \oplus N_v)\pi$
$x\pi$
G
G_M
V^{-1}
$|M|$
$b\pi$
$a\pi$
G_H
U
$c\pi$
G
$d\pi$
ξ_σ
$=$
G

All graph states are mergeable

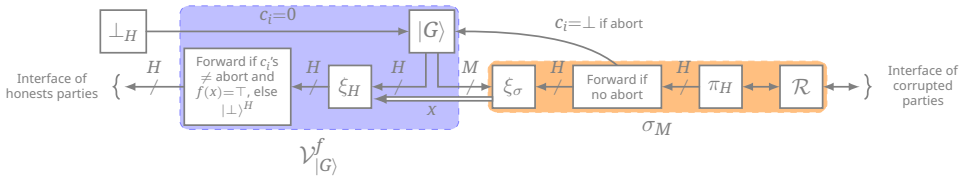
So all graph states are mergeable!

Mergeable: sufficient for security

We saw: composable $\Rightarrow$ mergeable

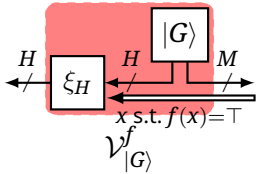
Surprisingly, **mergeable $\Rightarrow$ composable!**

Proof sketch: we consider this simulator



If distinguishable from $\pi_H\mathcal{R}$ (real world), then we use the distinguisher (second half) to distinguish the state obtained by H from $|G\rangle$. Impossible if π is secure in the game based model. $\square$

The importance of checking the corrections

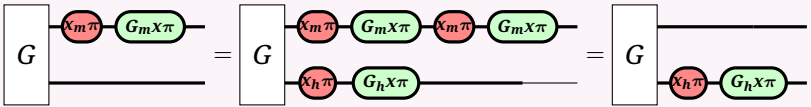
Why add f in the functionality 

?

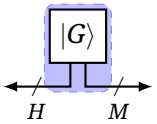
$\Rightarrow$ If the corrections are arbitrary: **resource useless** (e.g. coin-flipping).

Claim 1

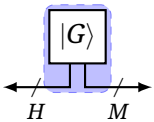
The set of allowed corrections is harmless (subset of stabilizers), as the adversary can always apply them on its side:



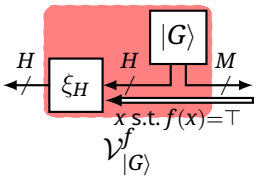
Keeping the natural functionality

– Me: Mom, can we have the natural functionality 

?

– Mom: We have the natural functionality 

at home.

The natural functionality at home: 

Keeping the natural functionality

If the adversary can apply the corrections, ***why do we need to send them to the functionality?***

$\Rightarrow$ Move corrections from functionality to protocol:

Theorem (informal)

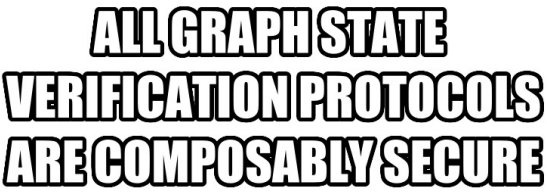
All graph state verification protocols can be turned into a composably secure protocol.

Proof sketch. Idea: add a round where all parties **apply a random stabilizer** on the final state: impossible to know if the stabilizer is applied by the honest parties, or due to non-corrected corrections occuring during the teleportation phase between the simulator and the functionality. □

Conclusion

ALL GRAPH STATE VERIFICATION PROTOCOLS ARE COMPOSABLY SECURE

Conclusion



Open questions:

- Can we generalize to other states (CV, non-graph states, qutrits…)?
- Other applications of our generalized entanglement swapping?
- Study more verification protocols


Thank you!