

An External Memory Relational Product

Steffan Christ Sølvesten, Jaco van de Pol

September 10, 2023



$$RelProd(S, T) \equiv (\exists \vec{x}. S(\vec{x}) \wedge T(\vec{x}, \vec{x}')) [\vec{x}' / \vec{x}]$$

$$RelProd(S, T) \equiv (\exists \vec{x}. S(\vec{x}) \wedge T(\vec{x}, \vec{x}'))[\vec{x}'/\vec{x}]$$

$$\exists x_i. \phi \equiv \phi[x_i/\top] \vee \phi[x_i/\perp]$$



$$\exists x_i. \phi \equiv \phi[x_i/\top] \vee \phi[x_i/\perp]$$



$$\exists x_i. \phi \equiv \phi[x_i/\top] \vee \phi[x_i/\perp]$$



$$\exists x_i. \phi \equiv \phi[x_i/\top] \vee \phi[x_i/\perp]$$



$$\exists x_i. \phi \equiv \phi[x_i/\top] \vee \phi[x_i/\perp]$$



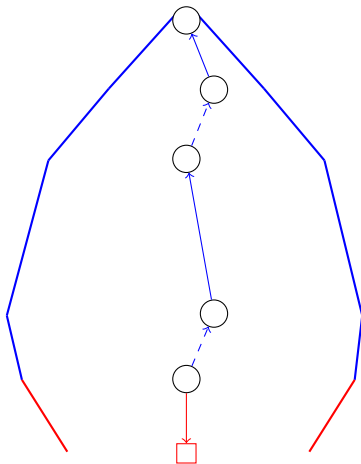
$$\exists x_i. \phi \equiv \phi[x_i/\top] \vee \phi[x_i/\perp]$$



$$\exists x_i. \phi \equiv \phi[x_i/\top] \vee \phi[x_i/\perp]$$



$$\exists x_i. \phi \equiv \phi[x_i/\top] \vee \phi[x_i/\perp]$$



$\exists \vec{x}. \phi(\dots)$ **(Push)**

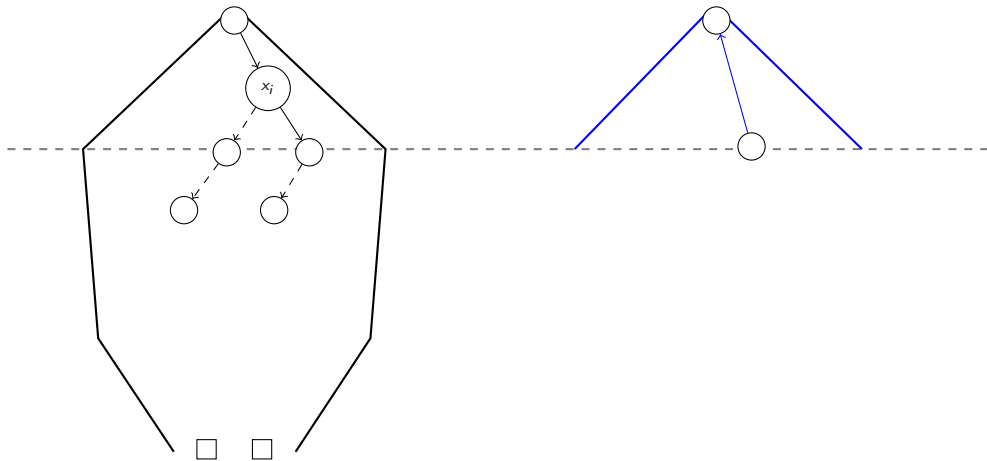


$\exists \vec{x}. \phi(\dots)$ **(Push)**



$\exists \vec{x}. \phi(\dots)$ (Push)



$$\exists \vec{x}. \phi(\dots) \quad (\text{Push})$$


$\exists \vec{x}. \phi(\dots)$ (Push)



$\exists \vec{x}. \phi(\dots)$ **(Push)**



$\exists \vec{x}. \phi(\dots)$ (Push)



$\exists \vec{x}. \phi(\dots)$ (Push)



$\exists \vec{x}. \phi(\dots)$ (Push)



$\exists \vec{x}. \phi(\dots)$ **(Bounce)**



$\exists \vec{x}. \phi(\dots)$ **(Bounce)**



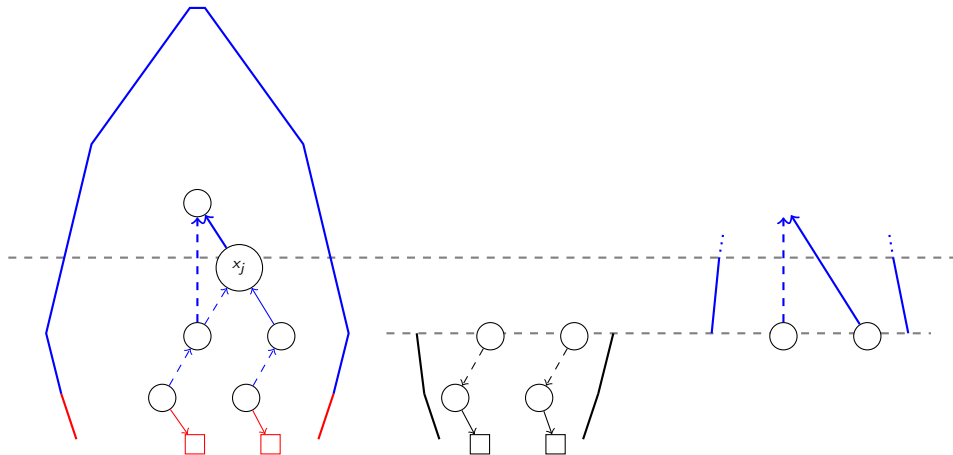
$\exists \vec{x}. \phi(\dots)$ **(Bounce)**



$\exists \vec{x}. \phi(\dots)$ **(Bounce)**



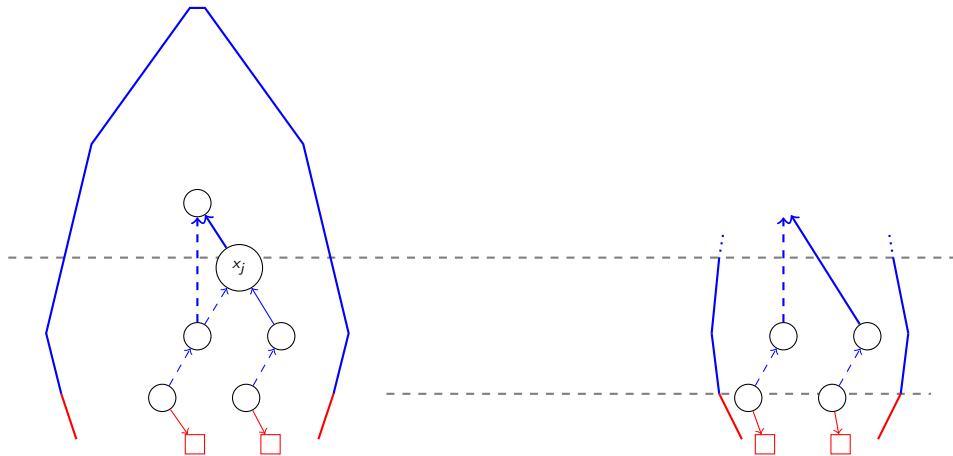
$\exists \vec{x}. \phi(\dots)$ **(Bounce)**



$\exists \vec{x}. \phi(\dots)$ **(Bounce)**



$\exists \vec{x}. \phi(\dots)$ **(Bounce)**



$\exists \vec{x}. \phi(\dots)$ **(Bounce)**



$\exists \vec{x}. \phi(\dots)$ **(Bounce)**



$\exists \vec{x}. \phi(\dots)$ **(Bounce)**



$\exists \vec{x}. \phi(\dots)$ **(Bounce)**



$\exists \vec{x}. \phi(\dots)$ **(Bounce)**



$\exists \vec{x}. \phi(\dots)$ **(Bounce)**



$$RelProd(S, T) \equiv (\exists \vec{x}. S(\vec{x}) \wedge T(\vec{x}, \vec{x}')) [\vec{x}' / \vec{x}]$$

Definition

A relabelling π is monotonic if $x_i < x_j \implies \pi(x_i) < \pi(x_j)$

Definition

A relabelling π is monotonic if $x_i < x_j \implies \pi(x_i) < \pi(x_j)$

If π is monotonic

- *1-Var / Push:*

Apply π in $O(L_N)$ extra time during the final bottom-up Reduce sweep.

- *Bounce:*

$x'_i < x'_j \implies x_i < x_j$: π can be applied during the outermost Reduce sweep.

That is, applying π can (essentially) be done for free.

Definition

A relabelling π is monotonic if $x_i < x_j \implies \pi(x_i) < \pi(x_j)$

If π is monotonic

- *1-Var / Push:*

Apply π in $O(L_N)$ extra time during the final bottom-up Reduce sweep.

- *Bounce:*

$x'_i < x'_j \implies x_i < x_j$: π can be applied during the outermost Reduce sweep.

That is, applying π can (essentially) be done for free.

If π is not monotonic

to be continued...