

TOWARDS A COMPUTATIONAL EFFICIENT, IMPLICITLY RESTARTED RATIONAL KRYLOV METHOD

Daan Camps, Karl Meerbergen, Raf Vandebril

KU LEUVEN

ILAS Conference, Leuven - July 12, 2016



INTRODUCTION

OVERVIEW

1. Krylov methods: Arnoldi & RKS
2. Implicit restart of Krylov methods
3. Numerical example
4. Conclusion & outlook

KRYLOV METHODS: ARNOLDI & RKS

Definition Krylov subspace

Given a matrix $A \in \mathbb{C}^{N \times N}$ and a non-zero vector $v \in \mathbb{C}^N$:

$$\mathcal{K}_m(A, v) = \text{span}\{v, Av, \dots, A^{m-1}v\}$$

ARNOLDI'S METHOD

[ARNOLDI, 1951]

Arnoldi's method

```
function [V,H] = Arnoldi(A,v,m)
    v = v/norm(v,2); V(:,1) = v;
    for i = 1:m
        V(:,i+1) = A*V(:,i);
        for j = 1:i
            H(j,i) = V(:,j)'*V(:,i+1);
            V(:,i+1) = V(:,i+1) - H(j,i)*V(:,j);
        end
        H(i+1,i) = norm(V(:,i+1),2);
        V(:,i+1) = V(:,i+1)/norm(V(:,i+1),2);
    end
end
```

Arnoldi's method

- Orthonormal basis V_i of \mathcal{K}_i
- Partial reduction to upper Hessenberg form
- Ritz values: $H_i y_j = \theta_j y_j$

$$A V_i = V_i H_i + h_{i+1,i} v_{i+1} e_i^T$$

or

$$A V_i = V_{i+1} \underline{H}_{i+1,i}$$

$$\Rightarrow V_i^* A V_i = H_i$$

Arnoldi Decomposition

$$A = V_i \begin{matrix} \\ = \\ \end{matrix} V_{i+1} \begin{pmatrix} & \\ & H_i \end{pmatrix}$$

Rotational Arnoldi Decomposition

 A V_i V_{i+1}

$$\begin{matrix} \rightarrow & \times \\ \downarrow & \times \\ \downarrow & \times \\ \downarrow & \times \\ \downarrow & \times \\ \downarrow & \times \\ \downarrow & \times \\ 0 & & & & & & & \end{matrix}$$

RATIONAL KRYLOV SEQUENCES

[RUHE, 1984], [RUHE, 1994], [RUHE, 1998]

Definition RKS

Given a matrix pencil $(A, B) \in \mathbb{C}^{N \times N}$ and a non-zero vector $v \in \mathbb{C}^N$:

$$\text{span}\{v, S_1v, S_1S_2v, \dots\} \text{ with}$$
$$S_i = (\alpha_i A + \beta_i B)^{-1}(\gamma_i A + \delta_i B)$$

RKS method

```
function [V,K,L] = RKS(A,B,v,m)
    v = v/norm(v,2); V(:,1) = v;
    for i = 1:m
        [alpha,beta,gamma,delta] = choose_parameters();
        t = determine_cont_vec();
        w = (alpha*A + beta*B) \ (gamma*A + delta*B) * V(:,1:i-1) * t;
        [w, h] = orthogonalise(w,V(1:i-1));
        V(:,i+1) = V(:,i+1)/norm(V(:,i+1),2)
        K(1:i+1,i) = alpha * h - gamma * t;
        L(1:i+1,i) = -beta * h + delta * t;
    end
end
```

RKS method

- Partial reduction to upper Hessenberg pencil (L, K)
- Poles: $-\beta_i/\alpha_i$

$$A V_{i+1} \underline{K}_i = B V_{i+1} \underline{L}_i$$

with

$$\begin{aligned}\underline{K}_i &= \underline{H}_i D_{\alpha,i} - \underline{T}_i D_{\gamma,i} \\ \underline{L}_i &= -\underline{H}_i D_{\beta,i} + \underline{T}_i D_{\delta,i}\end{aligned}$$

Special cases

Arnoldi:

- $(\alpha, \beta, \gamma, \delta) = (0, 1, 1, 0)$
- $S_i = (0A + 1I)^{-1}(1A + 0I)$

Special cases

Shift-and-invert Arnoldi:

- $(\alpha, \beta, \gamma, \delta) = (1, \beta, 0, 1)$
- $S_i = (1A + \beta I)^{-1}(0A + 1I)$

Special cases

Extended Krylov:

- $(\alpha, \beta, \gamma, \delta) = (0, 1, 1, 0)$ for A
- $(\alpha, \beta, \gamma, \delta) = (1, 0, 0, 1)$ for A^{-1}
- $S_i^+ = (0A + 1I)^{-1}(1A + 0I)$
- $S_i^- = (1A + 0I)^{-1}(0A + 1I)$

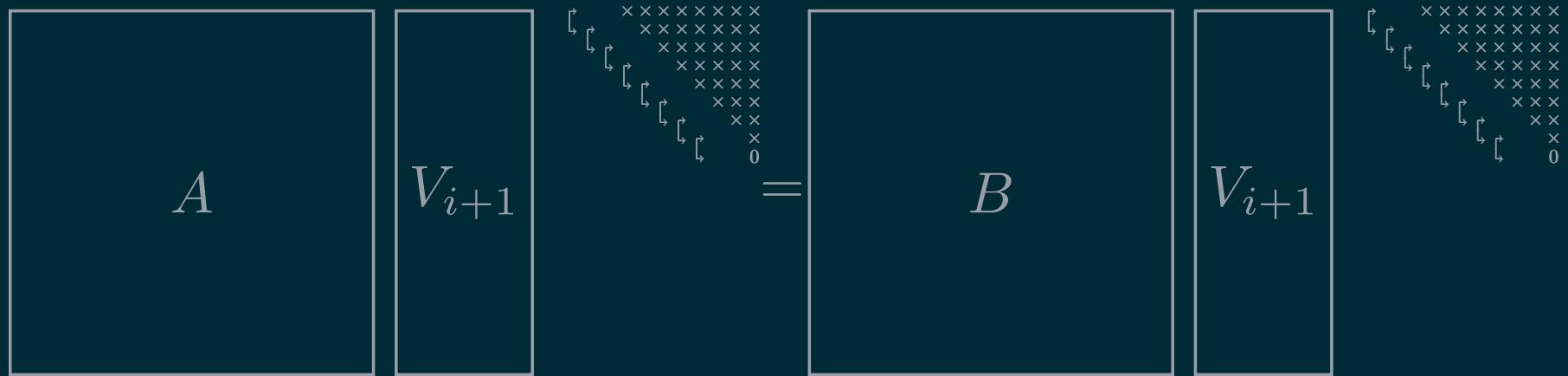
RKS Decomposition

$$A \begin{pmatrix} V_{i+1} \\ K_i \end{pmatrix} = B \begin{pmatrix} V_{i+1} \\ L_i \end{pmatrix}$$

Rotational RKS Decomposition

$$A \begin{bmatrix} V_{i+1} \end{bmatrix} = B \begin{bmatrix} V_{i+1} \end{bmatrix}$$

The matrix A is represented by a block-diagonal structure. The left block is labeled A . The right block is labeled V_{i+1} . Between them is an equals sign followed by another block-diagonal structure. This second structure has two blocks: the left one is labeled B and the right one is labeled V_{i+1} . Above the B block is a matrix with a specific pattern of 'x' characters and zeros, indicating a rotation matrix.



Special case: Extended Krylov Decomposition

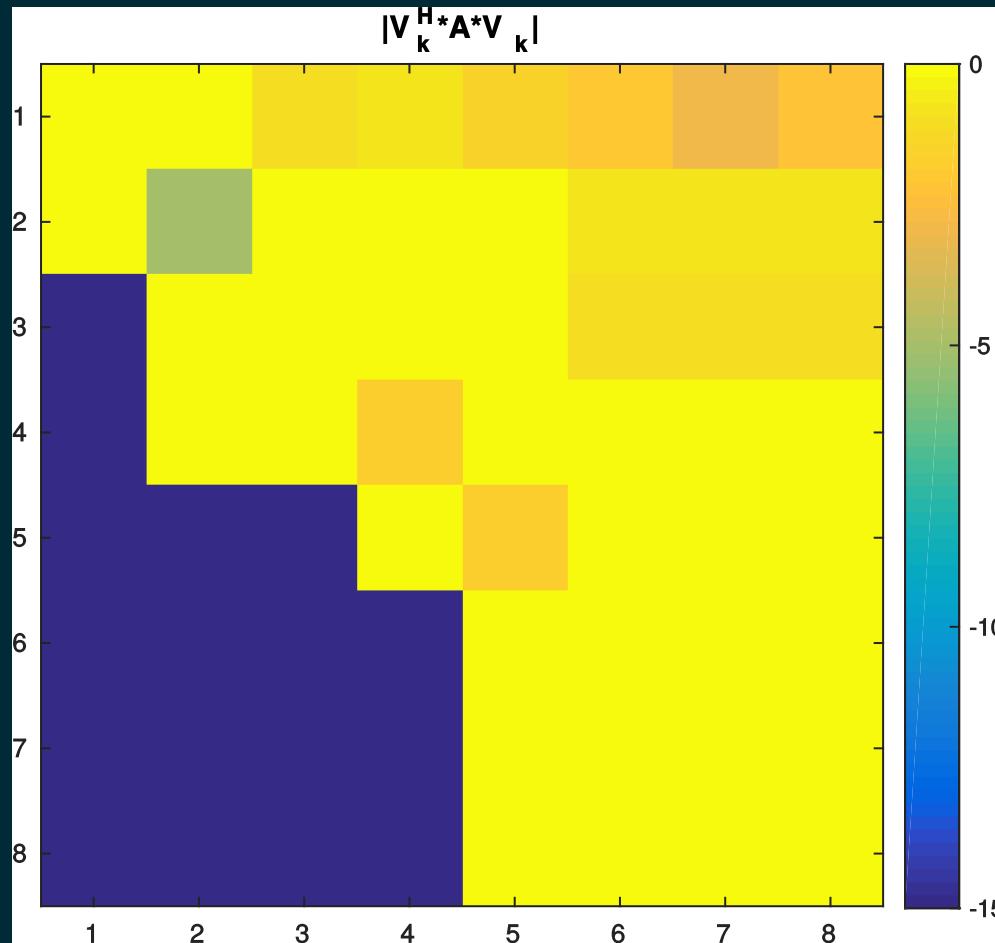
$$A \begin{pmatrix} V_{i+1} \\ \vdots \end{pmatrix} = \begin{pmatrix} V_{i+1} \\ \vdots \end{pmatrix} K_i + L_i$$

Rotational Extended Krylov Decomposition

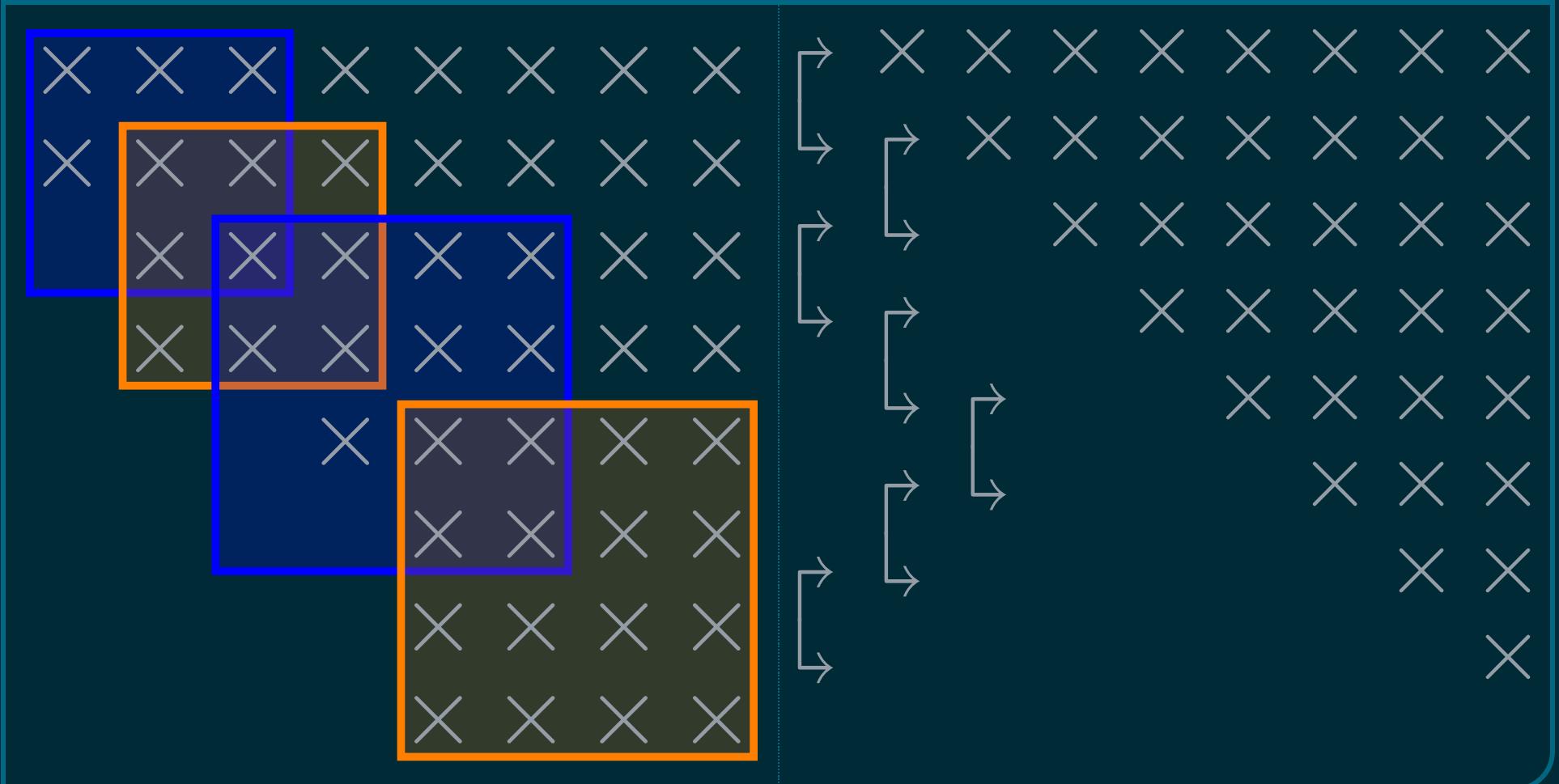
$$A \begin{pmatrix} V_{i+1} \\ \vdots \\ 0 \end{pmatrix} = \begin{pmatrix} V_{i+1} \\ \vdots \\ 0 \end{pmatrix}$$

The diagram illustrates the Rotational Extended Krylov Decomposition. On the left, a large square matrix A is shown. To its right is a vertical vector consisting of V_{i+1} followed by several rows of zeros, indicated by a series of downward-pointing arrows. This vector is multiplied by a square matrix on the right, which also consists of V_{i+1} followed by zero rows. The entire equation is preceded by an equals sign.

Structure in projected counterpart



Structure in projected counterpart



IMPLICIT RESTART OF KRYLOV ITERATIONS



SORENSEN'S IMPLICITLY RESTARTED ARNOLDI METHOD (IRA)

[SØRENSEN, 1992]

Problem

As Arnoldi's method proceeds:

- cost of orthogonalisation increases
- new basis vector needs to be stored
- cost of computing Ritz values increases

Solution

Implicit restart of the iteration:

$$\underline{H} - \mu I = [Q \quad q] \begin{bmatrix} R \\ 0 \end{bmatrix}$$

$$\underline{H} \leftarrow Q^* \underline{H} Q_{i,i-1}$$

$$V \leftarrow V Q$$

Improved solution

Implicit QR step on the factorised Hessenberg matrix by
chasing the rotations [VANDEBRIL, 2011]

Advantages

- Optimal computational complexity
- Multiple shifts at once or tightly packed shifts
- ...

Ingredients

- implicit Q-theorem [FRANCIS, 1961]
- elementary operations with Givens rotations

Bring a (pattern of) rotations through an upper triangular matrix

$$\left[\begin{array}{cccc} \times & \times & \times & \times \\ & \times & \times & \times \\ & & \times & \times \\ & & & \times \end{array} \right] = \left[\begin{array}{cccc} \times & \times & \times & \times \\ & \times & \times & \times \\ & & \times & \times \\ & & & \times \end{array} \right] = \left[\begin{array}{cccc} \times & \times & \times & \times \\ & \times & \times & \times \\ & & \times & \times \\ & & & \times \end{array} \right]$$

Fusion of two rotations

$$\begin{array}{c} \curvearrowleft \\ \curvearrowright \end{array} \quad \begin{array}{c} \curvearrowright \\ \curvearrowright \end{array} = \begin{array}{c} \curvearrowright \\ \curvearrowright \end{array}$$

Shift-through of a pattern of 3 rotations

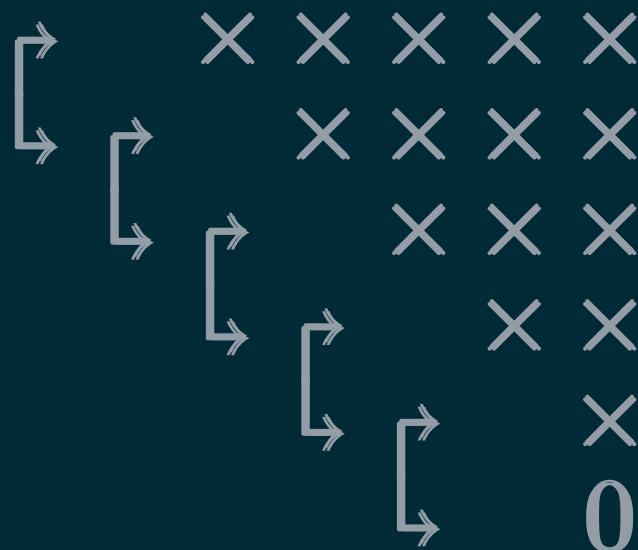
$$\begin{array}{c} \curvearrowright \\ \curvearrowright \end{array} \quad \begin{array}{c} \curvearrowright \\ \curvearrowright \end{array} \quad \begin{array}{c} \curvearrowright \\ \curvearrowright \end{array} = \begin{array}{c} \curvearrowright \\ \curvearrowright \end{array} \quad \begin{array}{c} \curvearrowright \\ \curvearrowright \end{array}$$

ROTATIONALLY IMPLICITLY RESTARTED ARNOLDI METHOD (RIRA)

An example

Initial situation

V_{i+1}



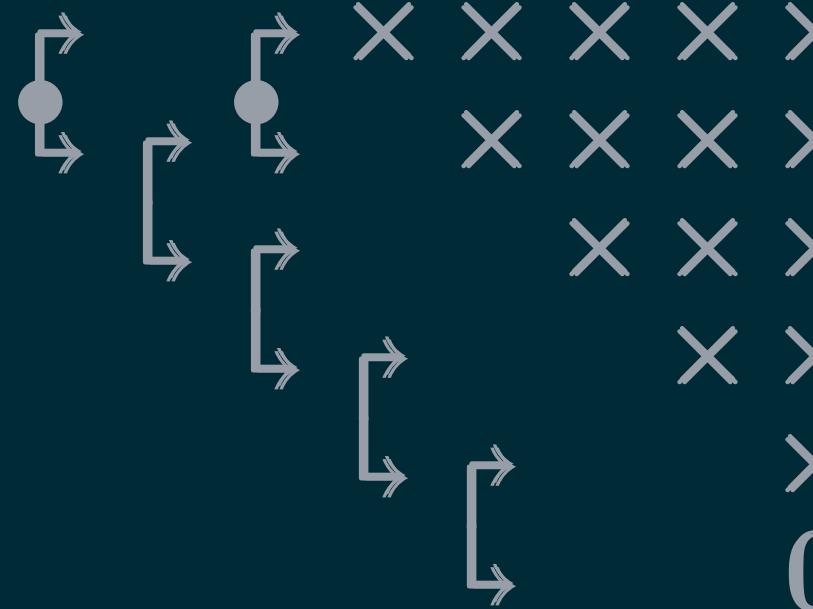
First similarity transformation

V_{i+1}

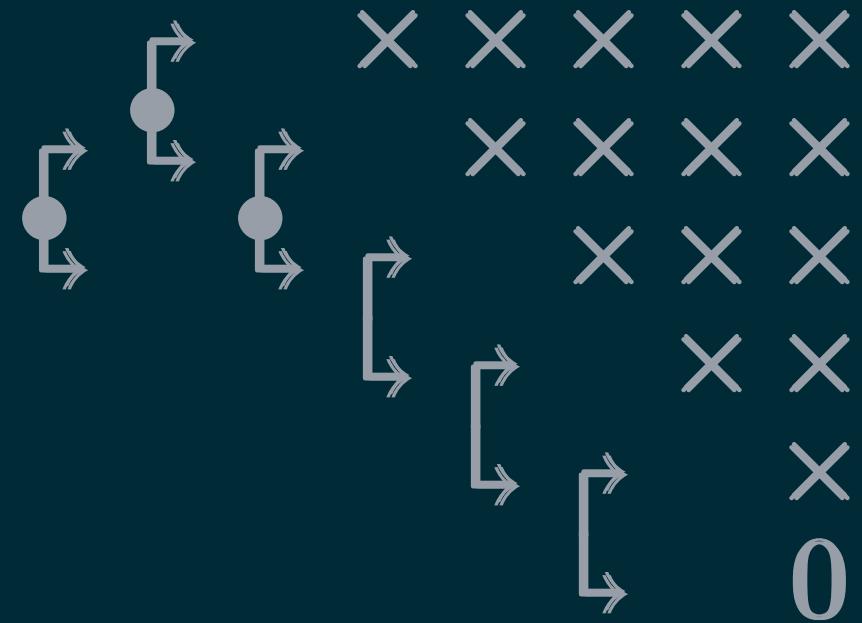
$$\begin{matrix} & * & & & & \\ * & \xrightarrow{\quad} & \xrightarrow{\quad} & \xrightarrow{\quad} & \xrightarrow{\quad} & \xrightarrow{\quad} \\ & X & X & X & X & X & 0 \\ & X & X & X & X & X & \\ & X & X & X & & & \\ & X & X & & & & \\ & X & & & & & \\ & 0 & & & & & \end{matrix}$$

Fusion & right rotation through upper triangular

$$V_{i+1}$$

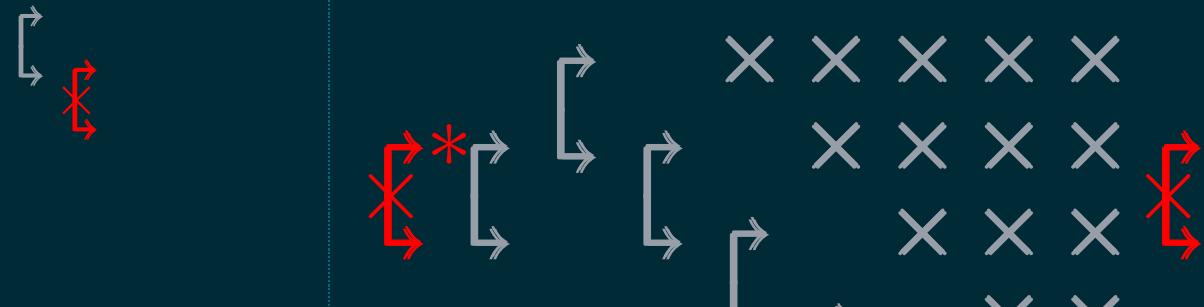


Shift-through

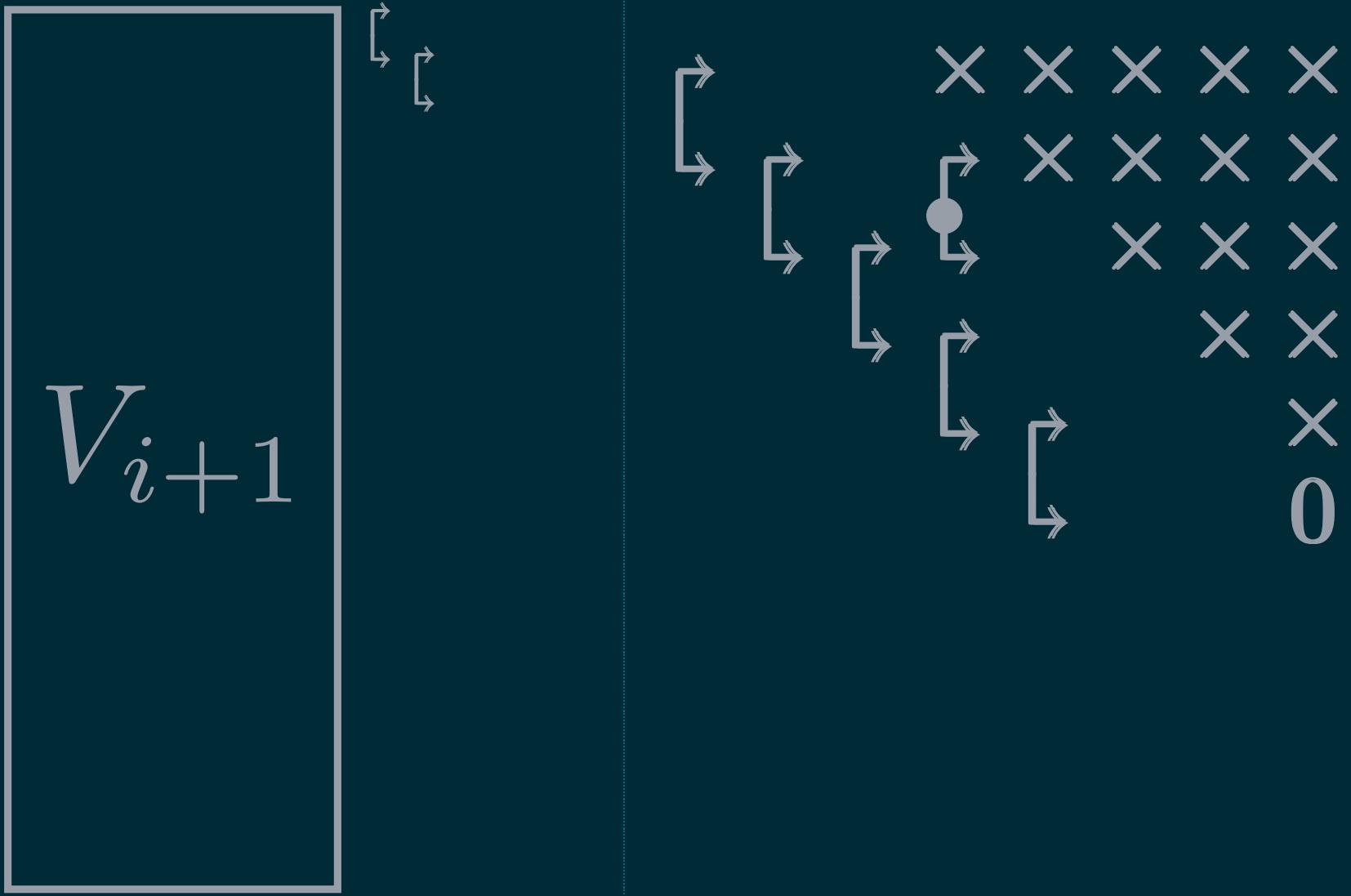


Second similarity transformation

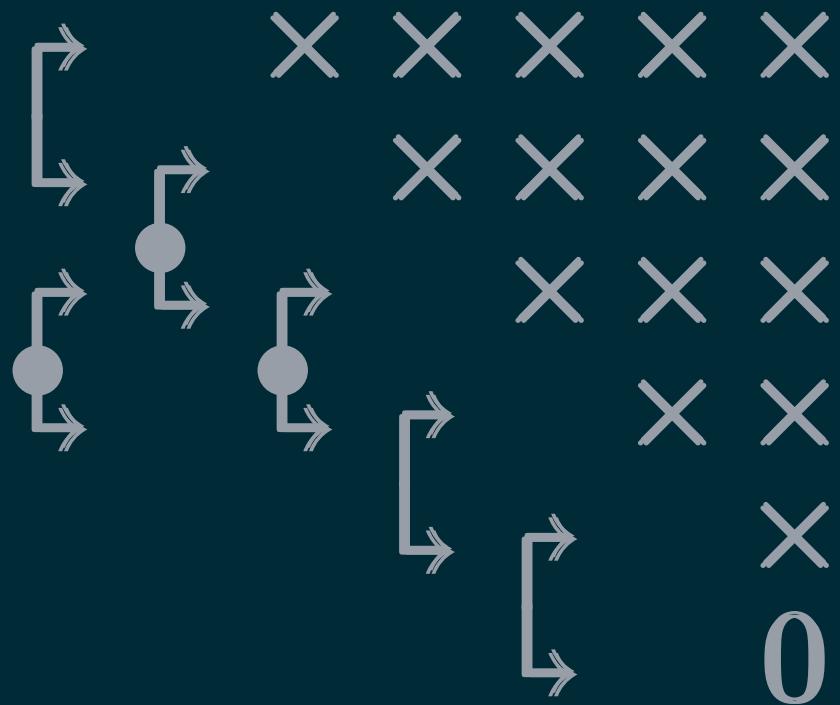
$$V_{i+1}$$



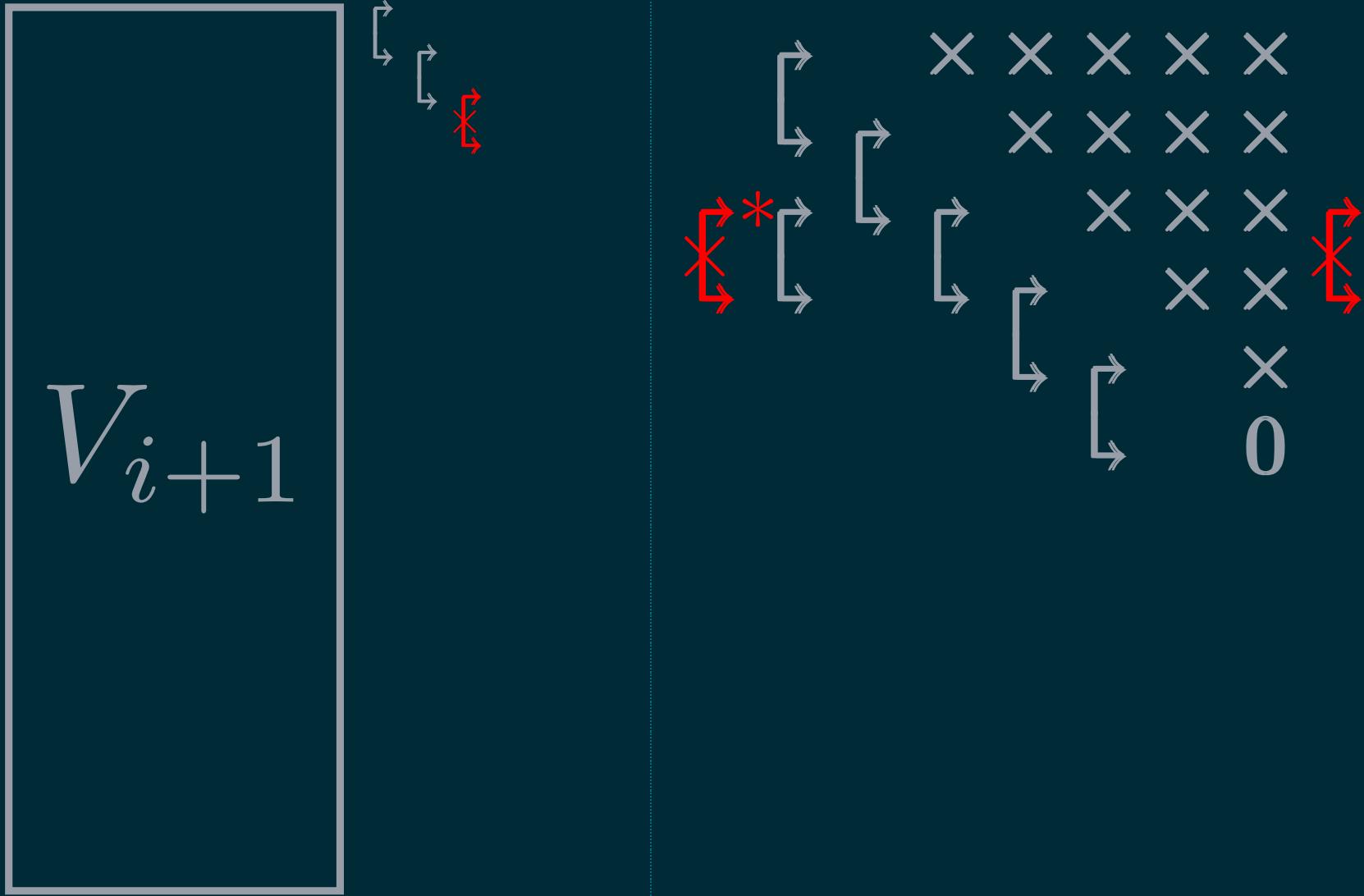
Fusion & right rotation through upper triangular



Shift-through



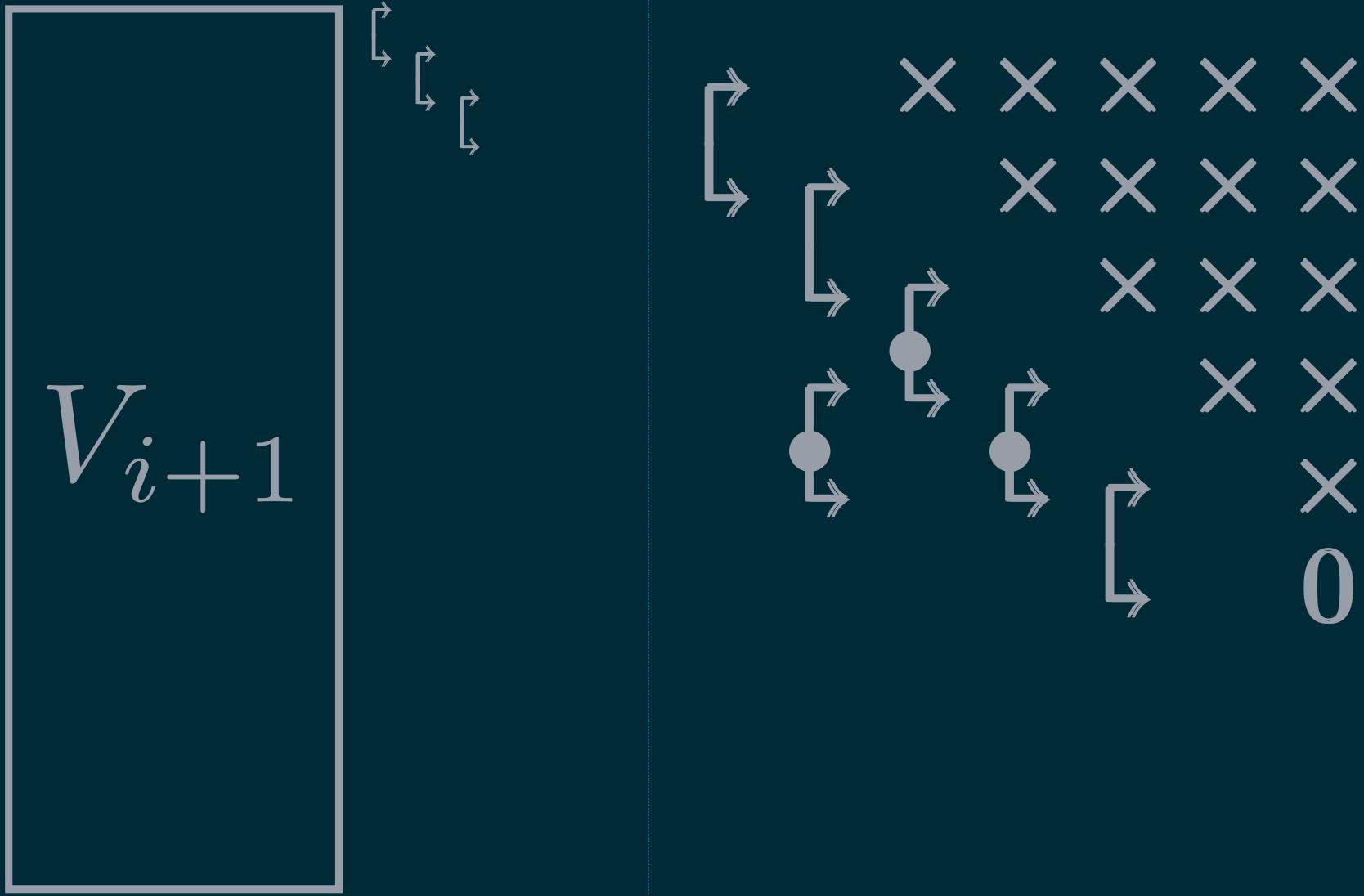
Third similarity transformation



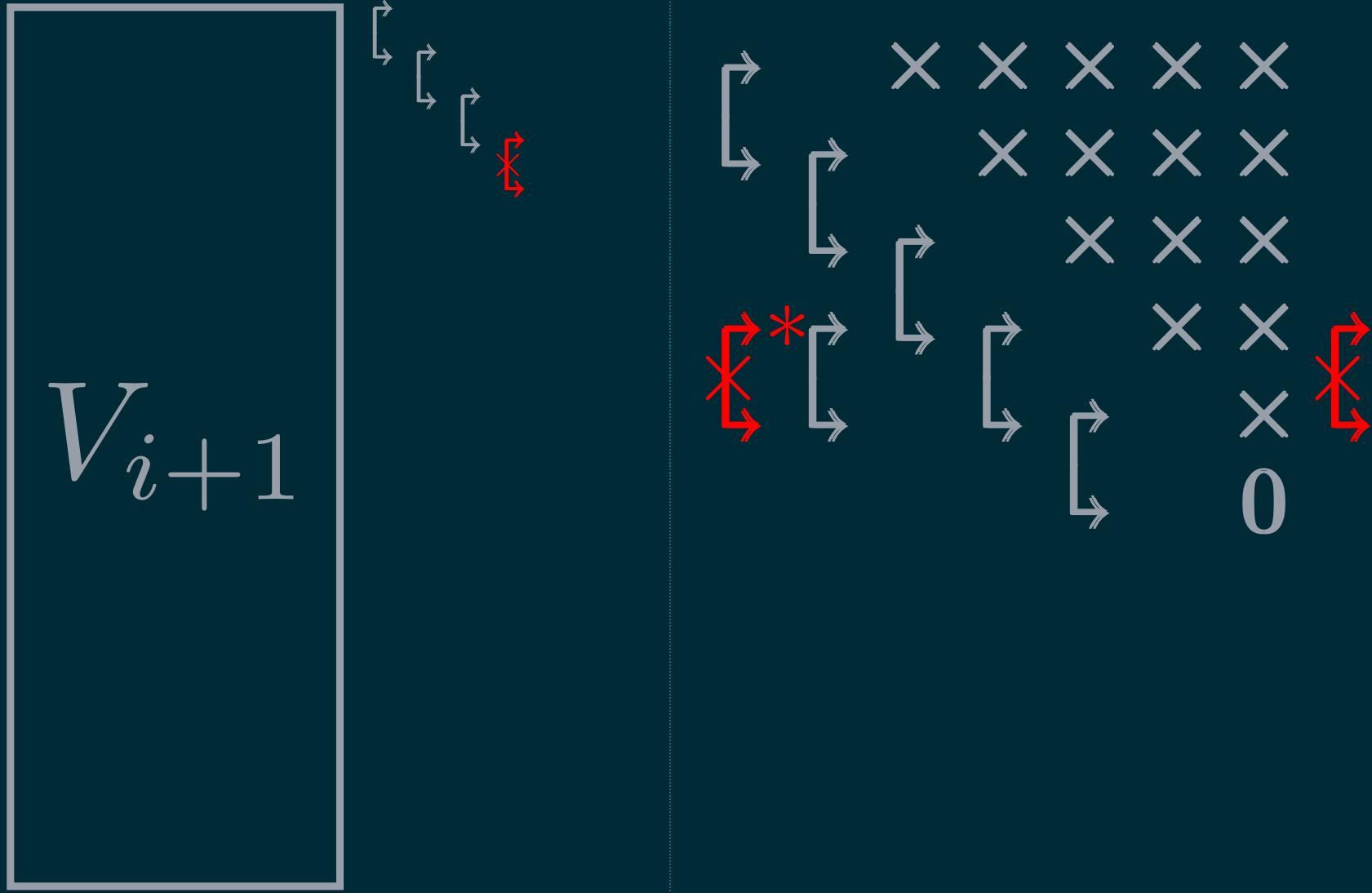
Fusion & right rotation through upper triangular



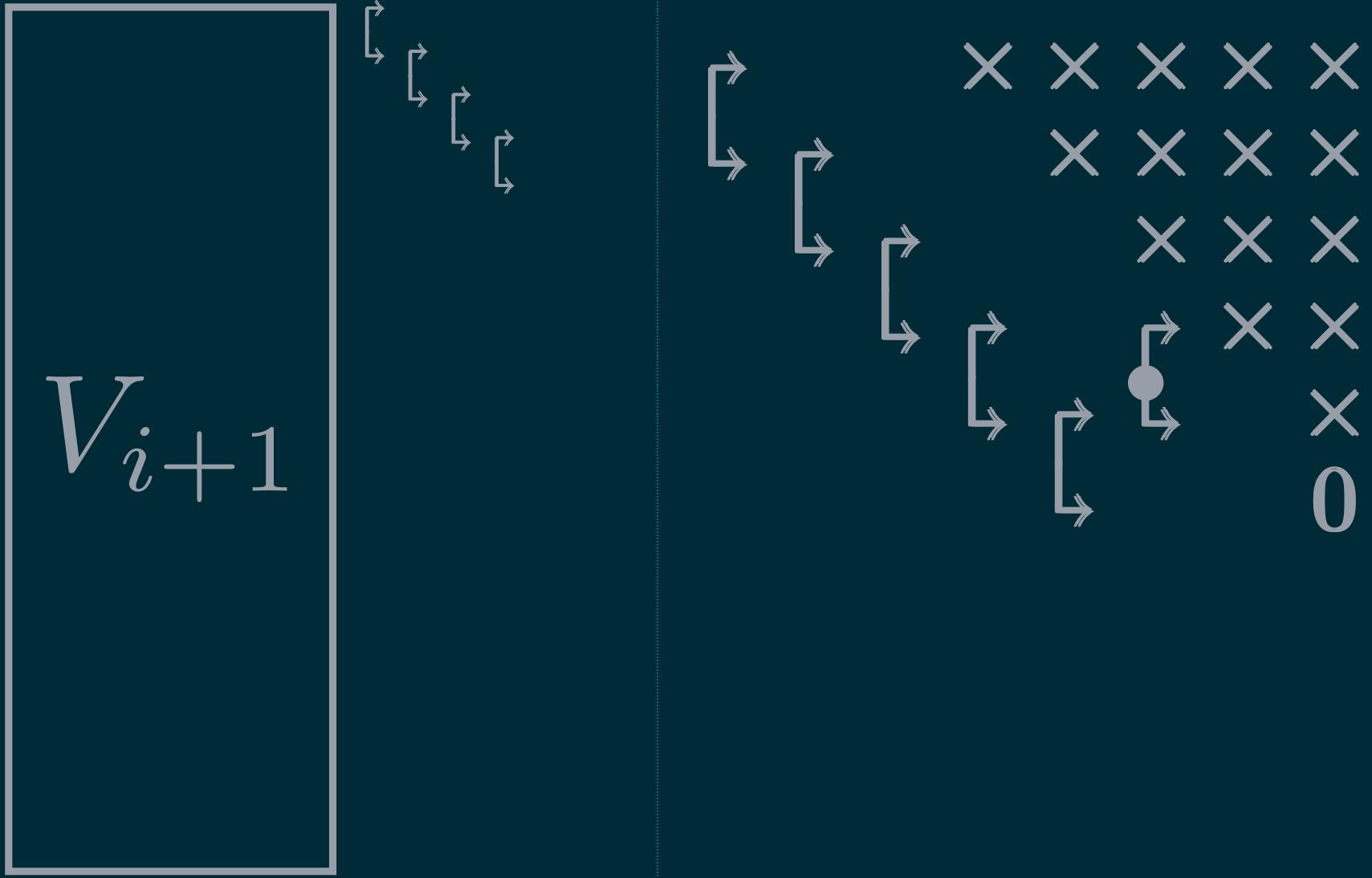
Shift-through



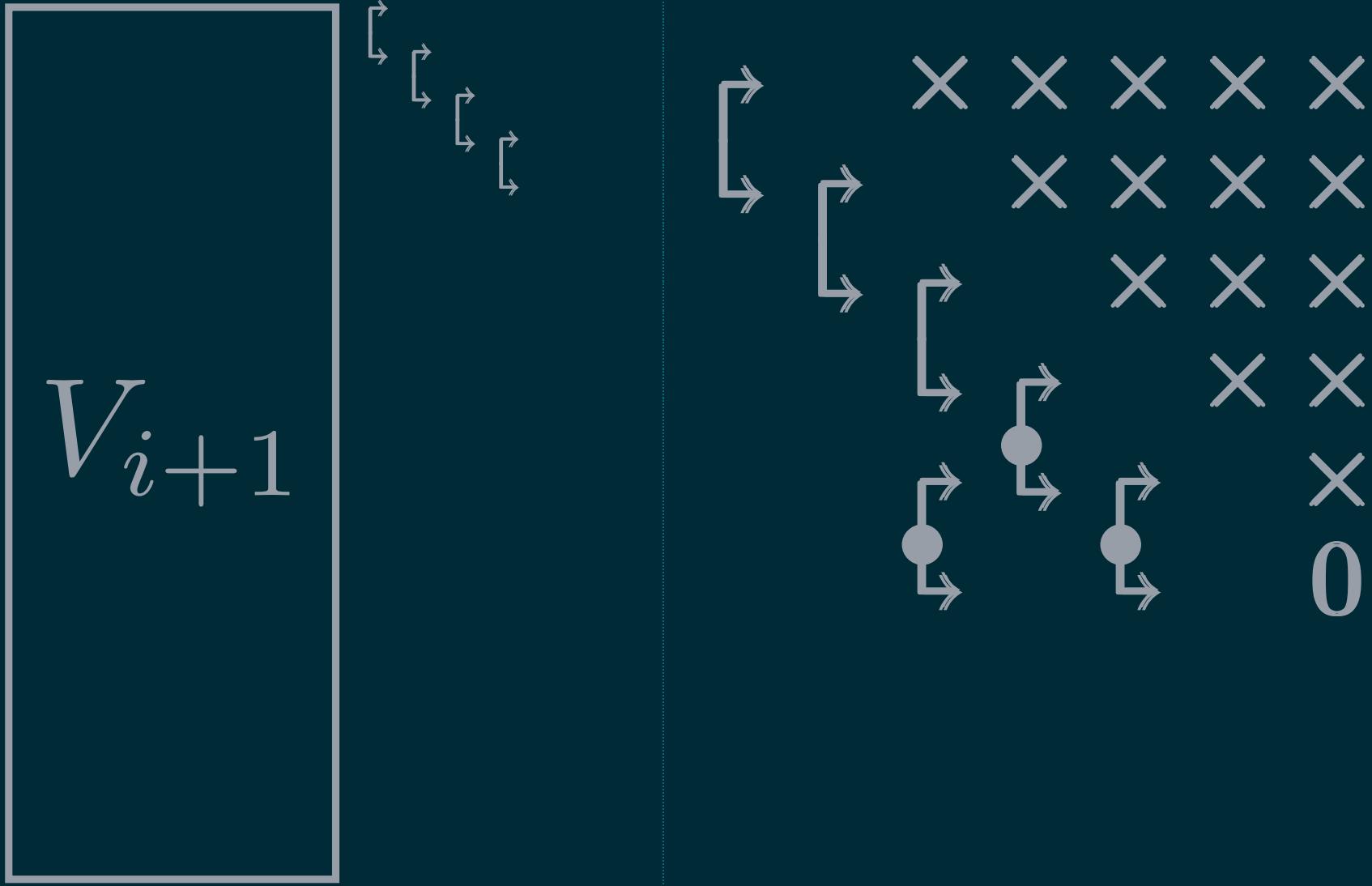
Fourth similarity transformation



Fusion & right rotation through upper triangular

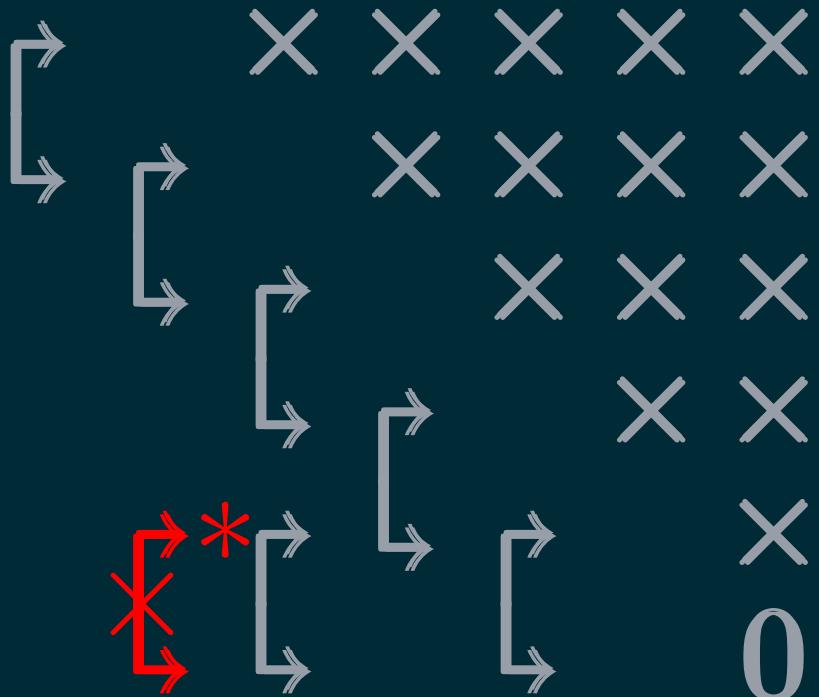
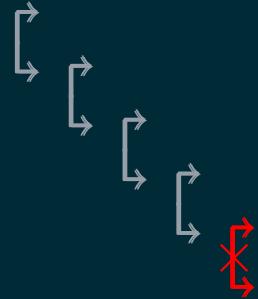


Shift-through



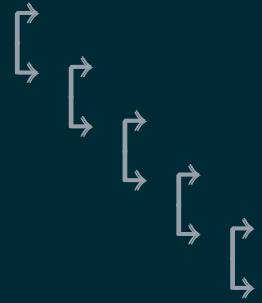
Fifth similarity transformation

V_{i+1}



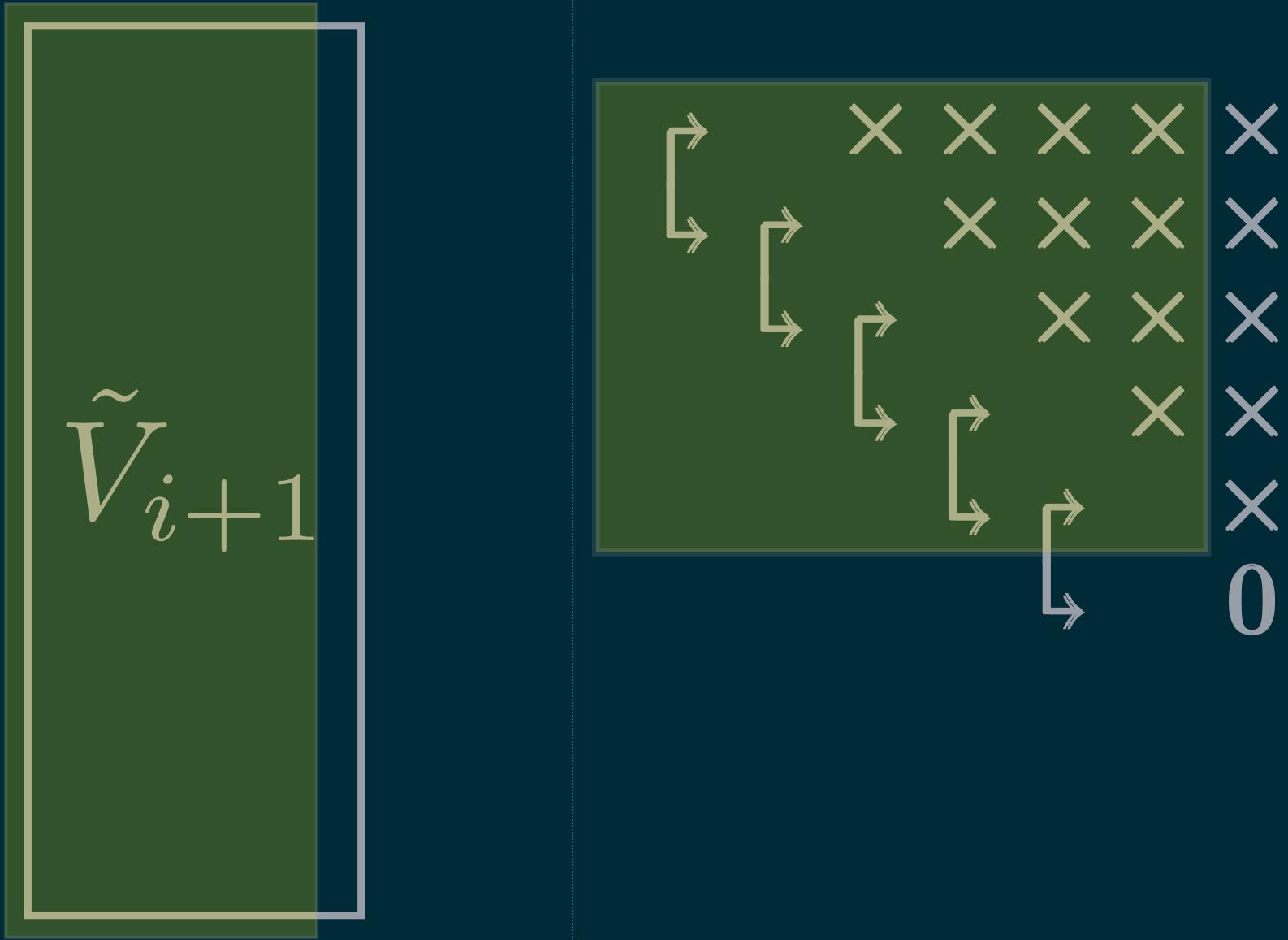
Final fusion

V_{i+1}



0

Extract reduced Arnoldi decomposition



ROTATIONALLY IMPLICITLY RESTARTED EXTENDED KRYLOV (RIREK)

An example for
 $\text{span}(v, A v, A^{-1} v, A^2 v, A^3 v A^{-2} v, A^{-3} v, A^{-4} v, A^4 v)$

STEP 1: TRANSFORM THE PENCIL TO SUITABLE COMPRESSED FORMAT

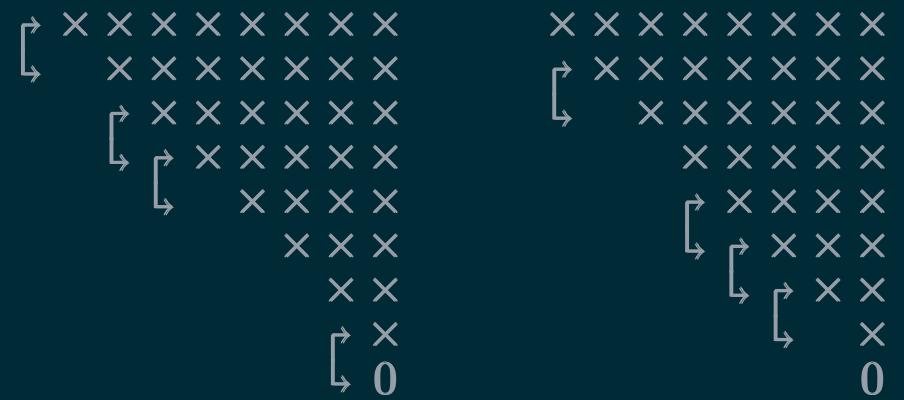
Two options:

- left removal of K rotations
- right removal of K rotations

Option 1: left removal of K rotations

$$V_{i+1}$$

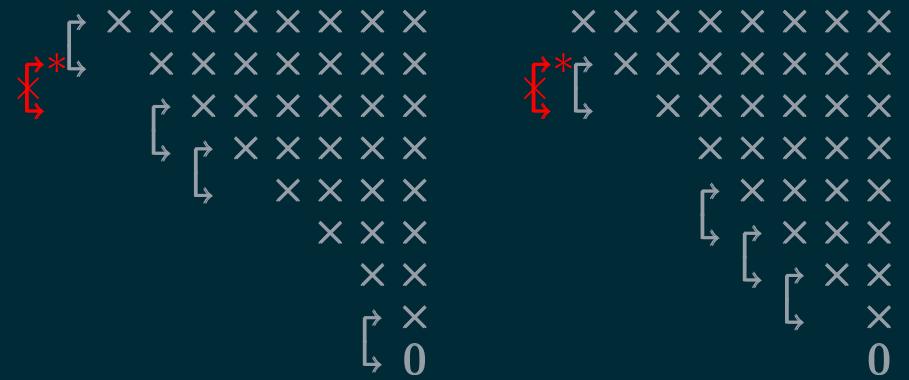
$$(\underline{L}, \underline{K})$$



Option 1: left removal of K rotations

$$V_{i+1}$$

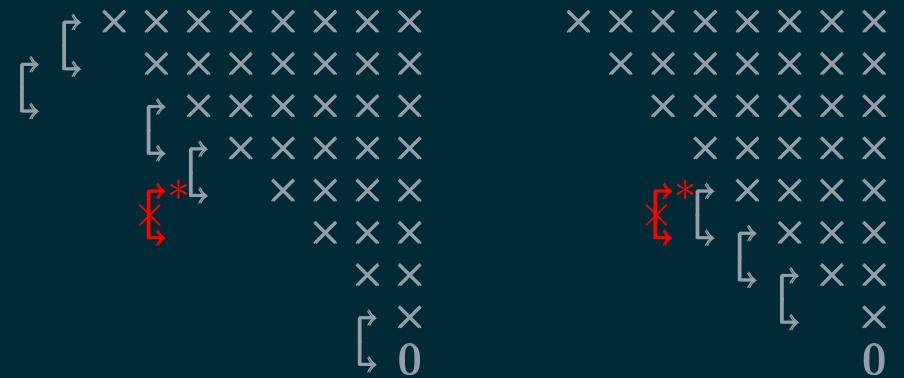
$$(\underline{L}, \underline{K})$$



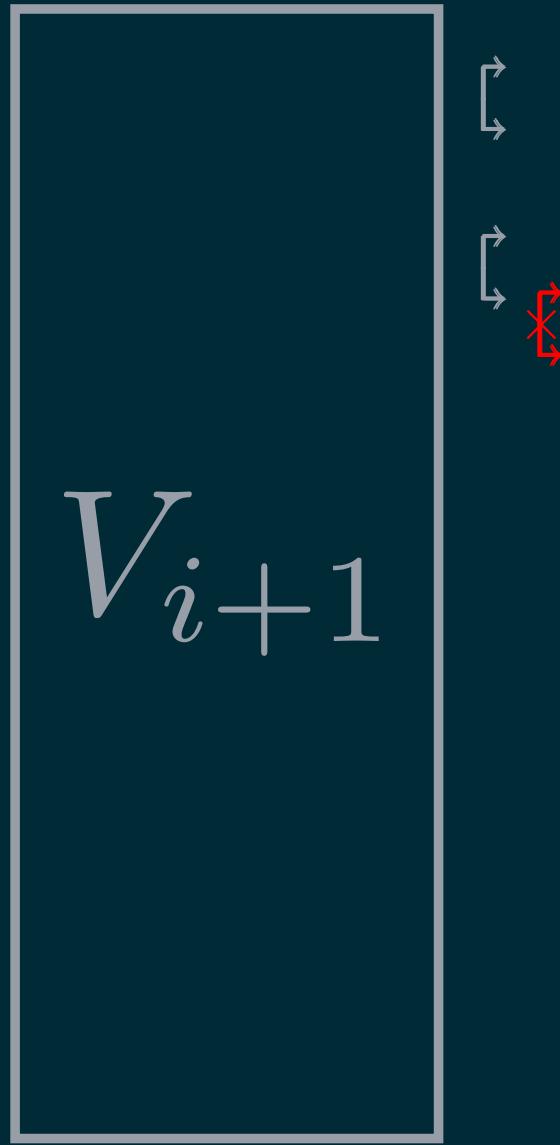
Option 1: left removal of K rotations



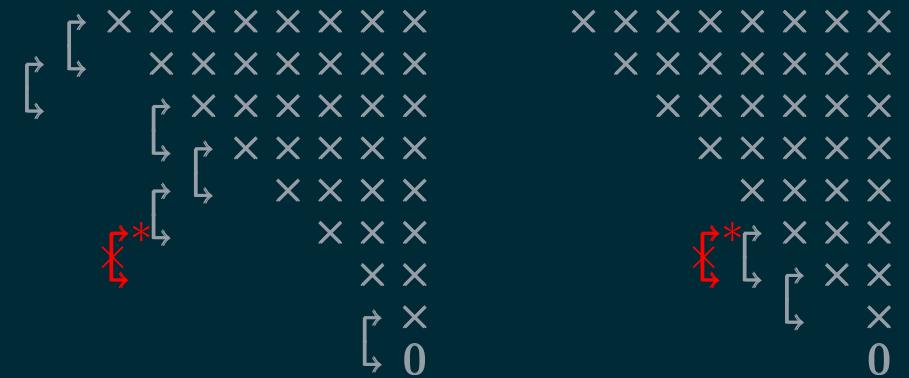
$(\underline{L}, \underline{K})$



Option 1: left removal of K rotations



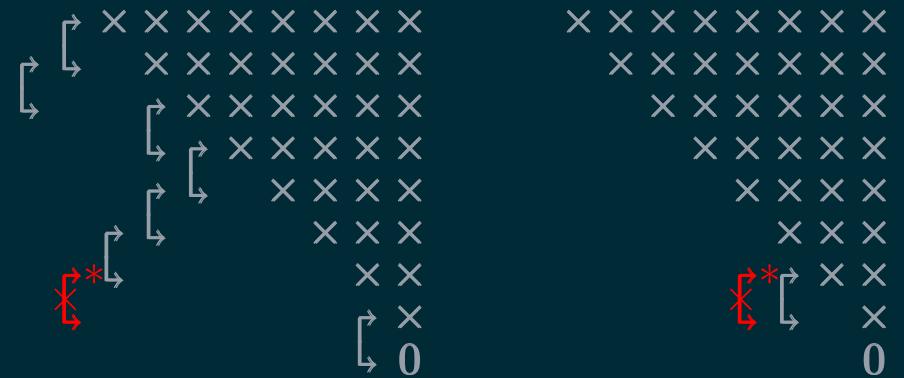
$(\underline{L}, \underline{K})$



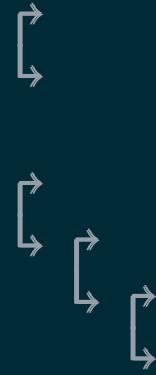
Option 1: left removal of K rotations



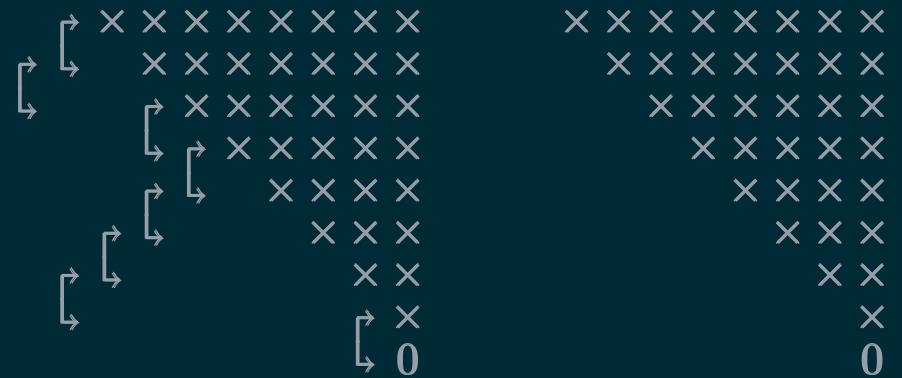
$(\underline{L}, \underline{K})$



Option 1: left removal of K rotations

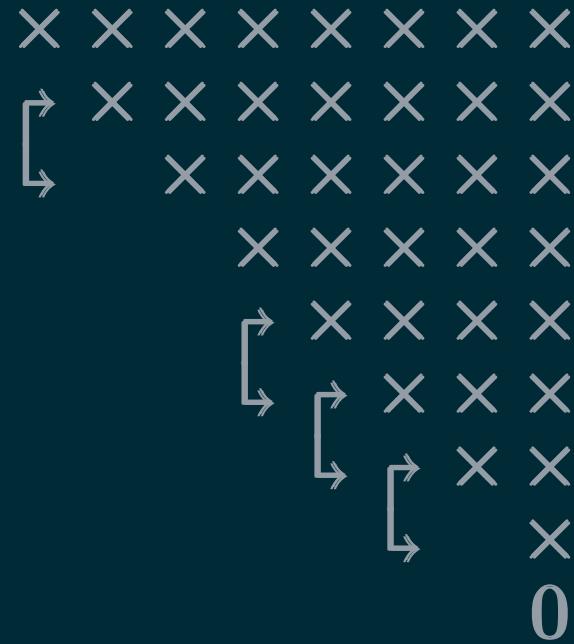
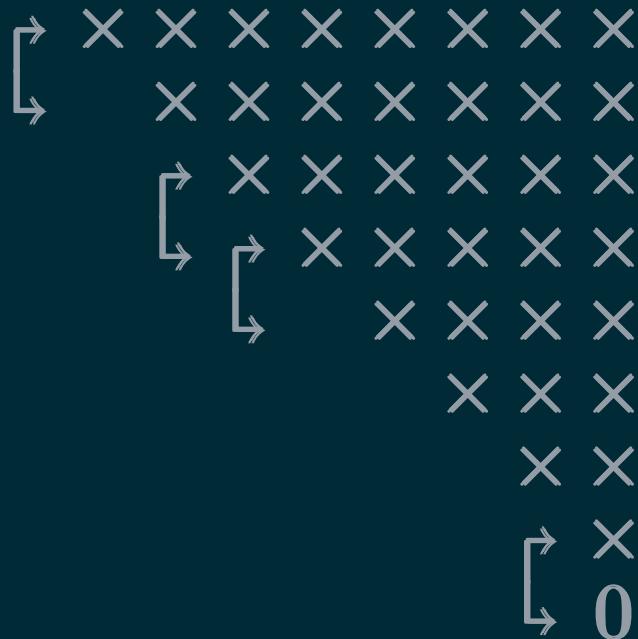


$(\underline{L}, \underline{K})$



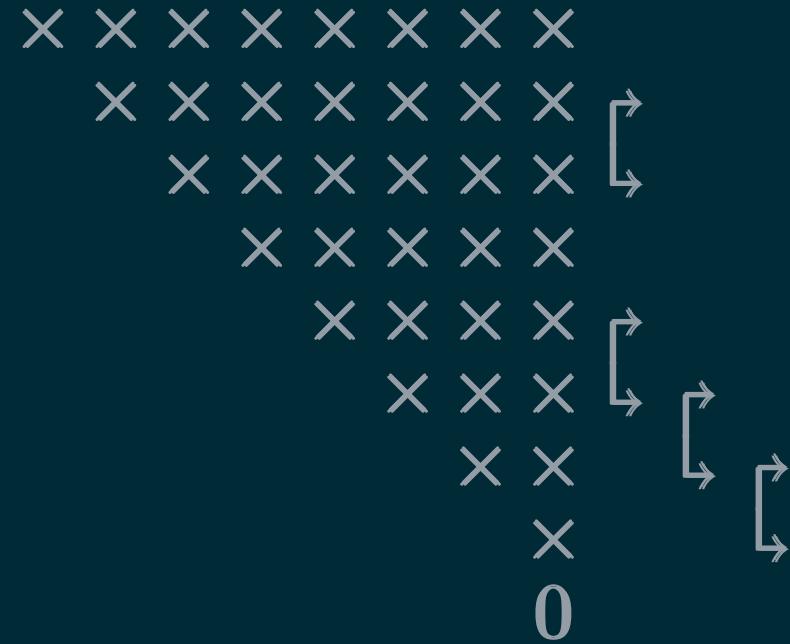
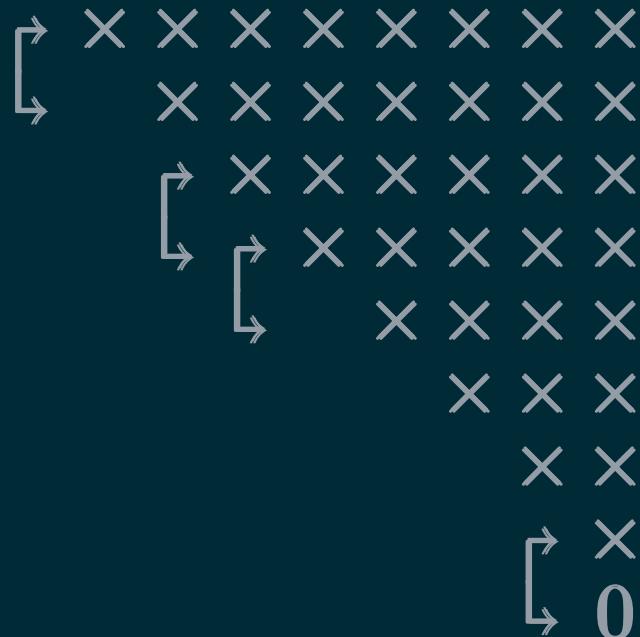
Option 2: right removal of K rotations

$(\underline{L}, \underline{K})$



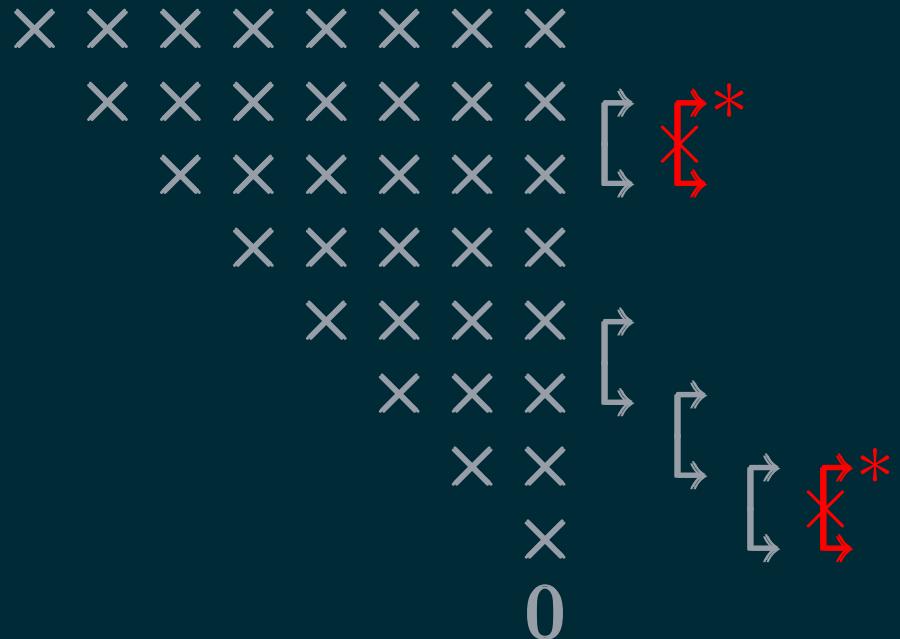
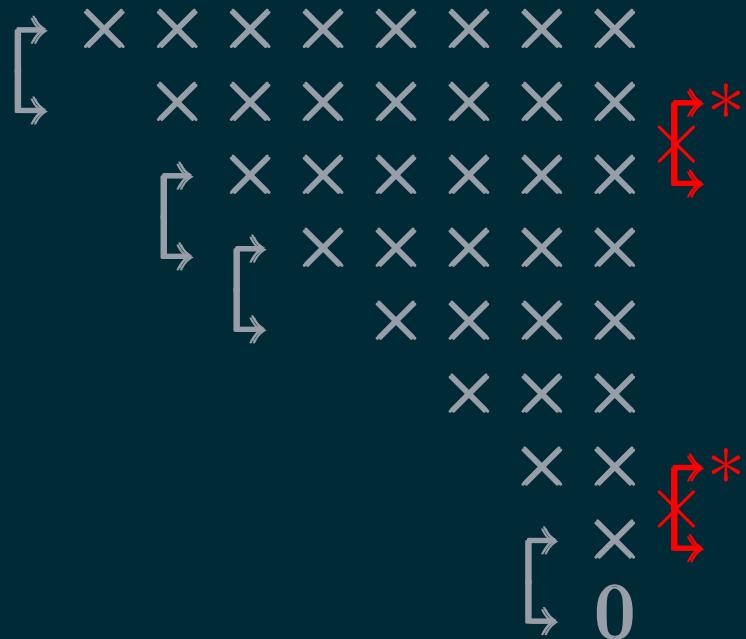
Option 2: right removal of K rotations

$$(\underline{L}, \underline{K})$$



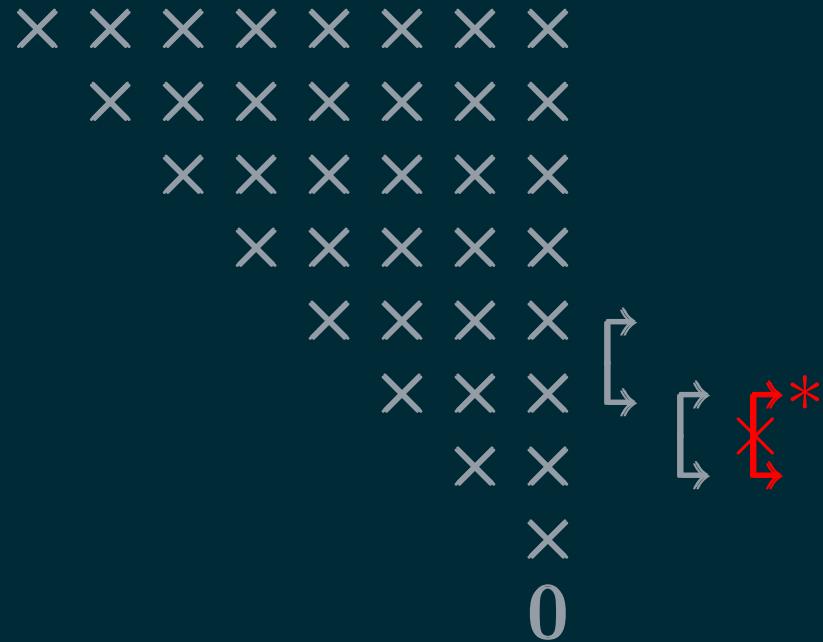
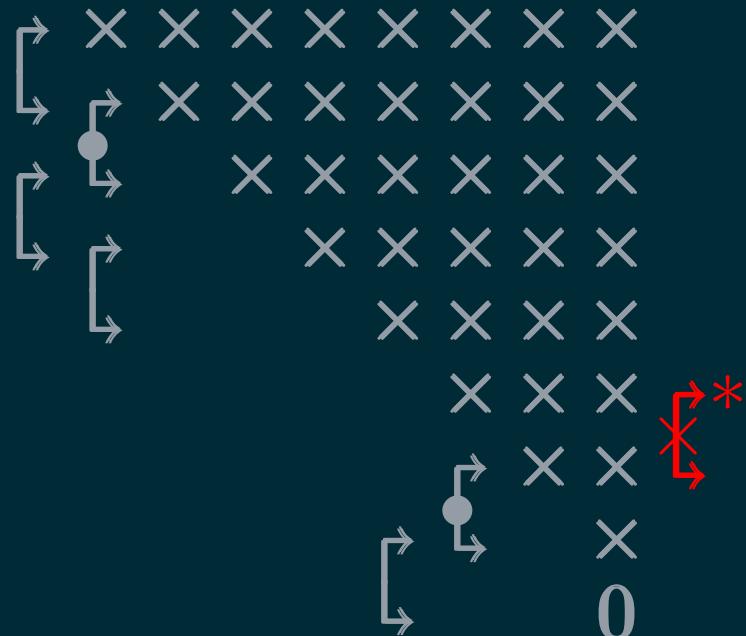
Option 2: right removal of K rotations

$(\underline{L}, \underline{K})$



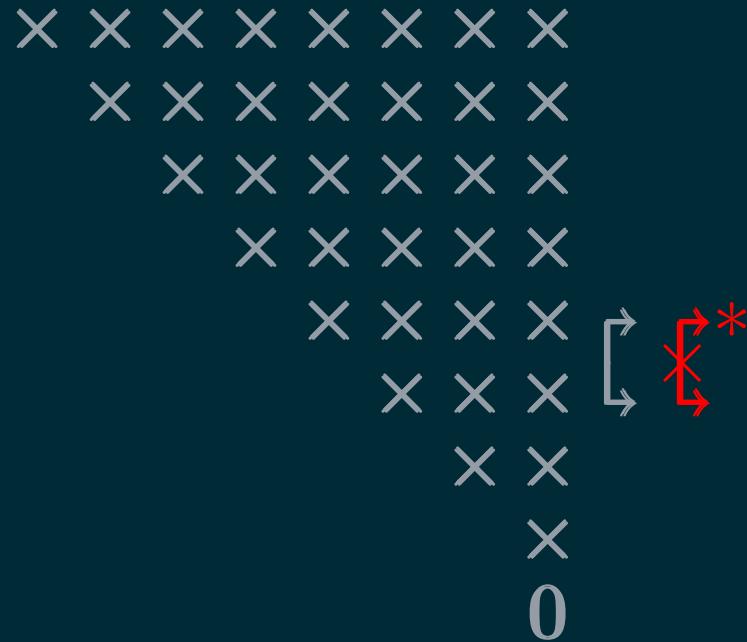
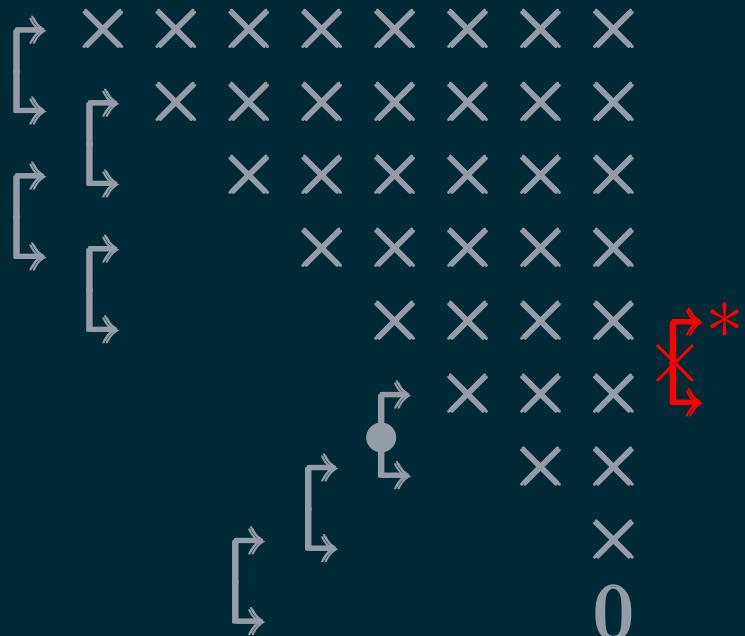
Option 2: right removal of K rotations

$$(\underline{L}, \underline{K})$$



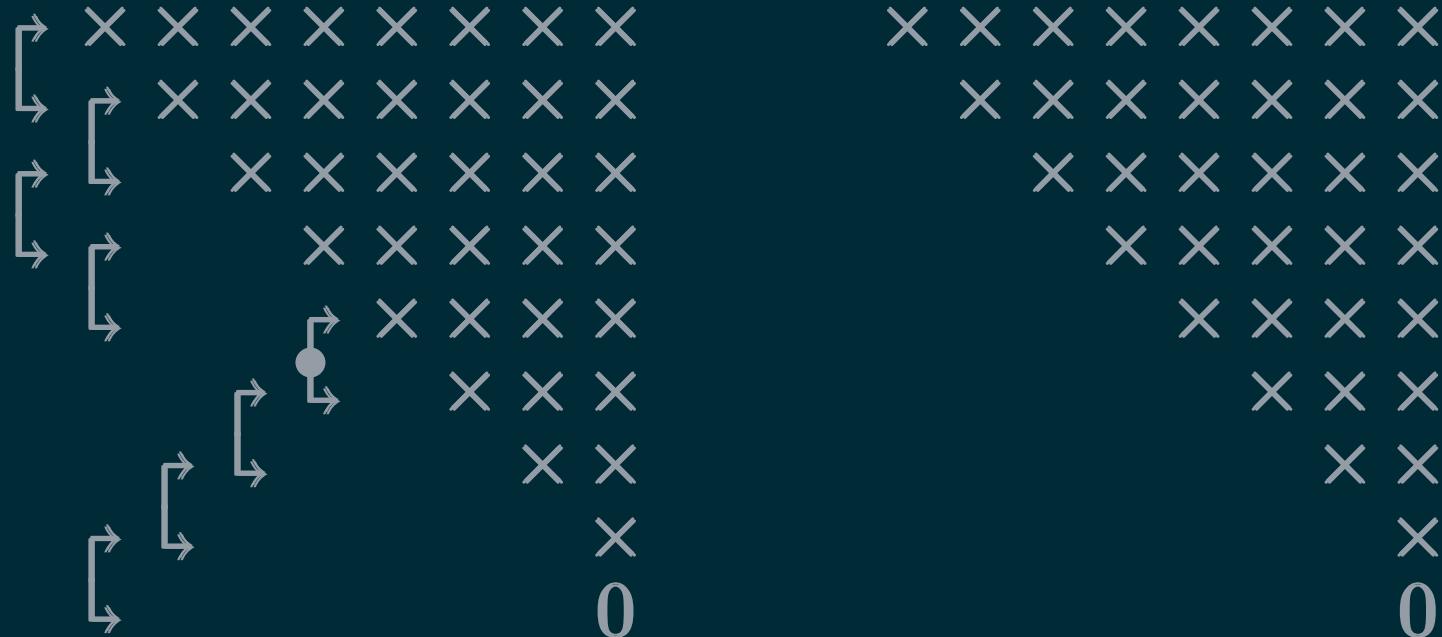
Option 2: right removal of K rotations

$(\underline{L}, \underline{K})$



Option 2: right removal of K rotations

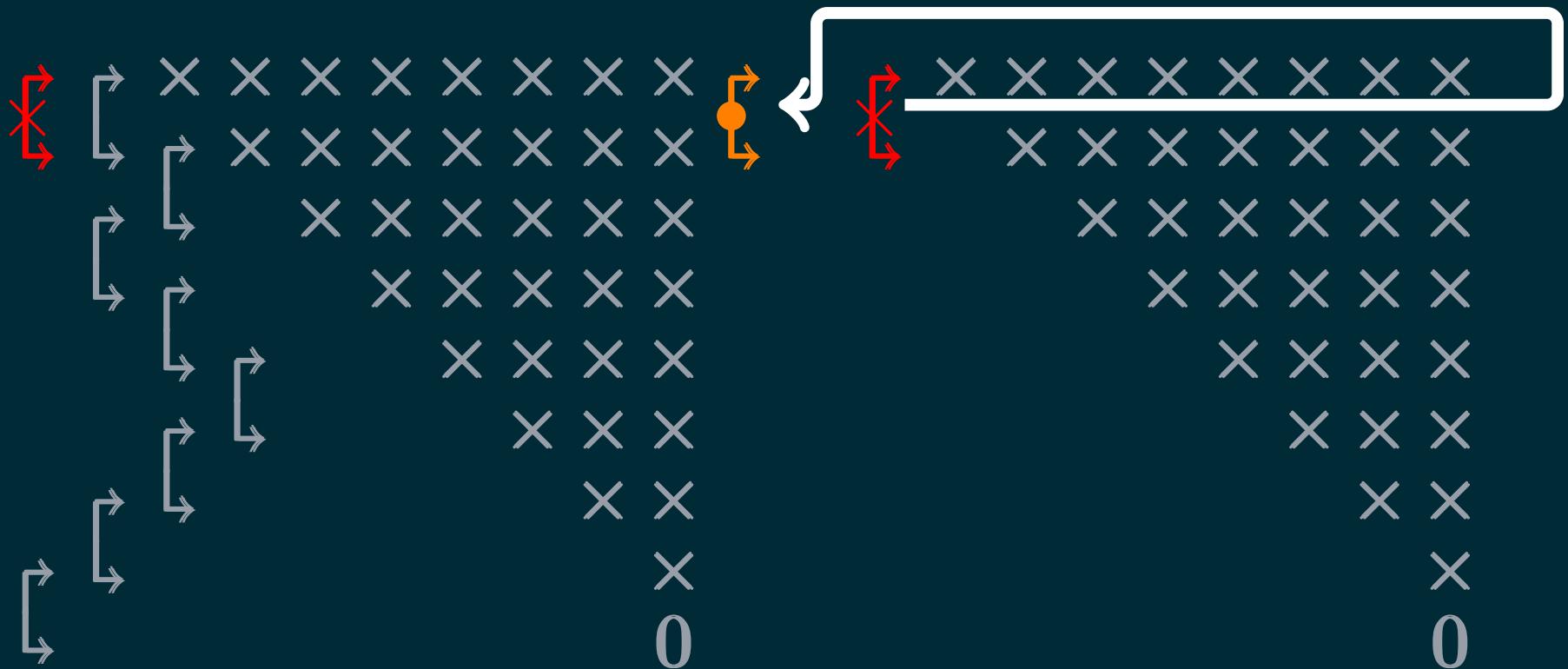
$$\left(\underline{L}, \underline{K}\right)$$



STEP 2: PERTURB THE PENCIL AND CHASE THE PERTURBATION DOWNWARDS

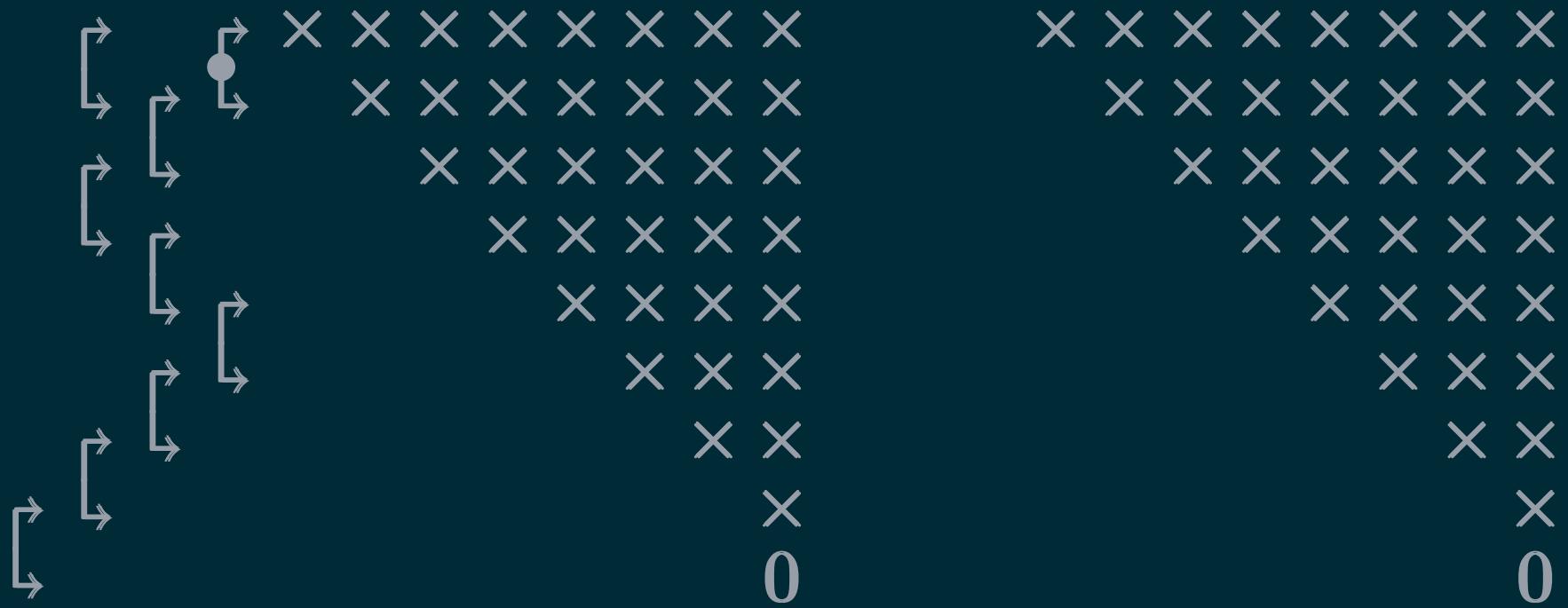
Chasing procedure

$$(\underline{L}, \underline{K})$$



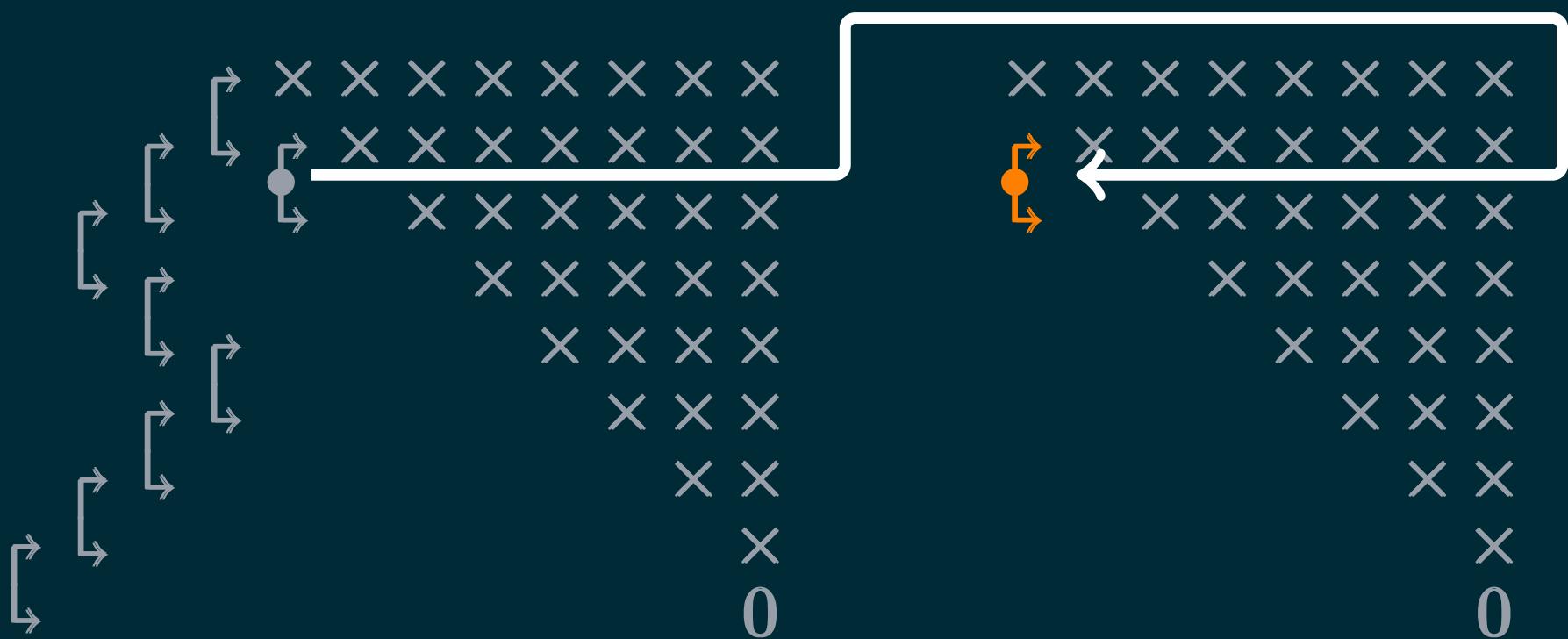
Chasing procedure

$(\underline{L}, \underline{K})$



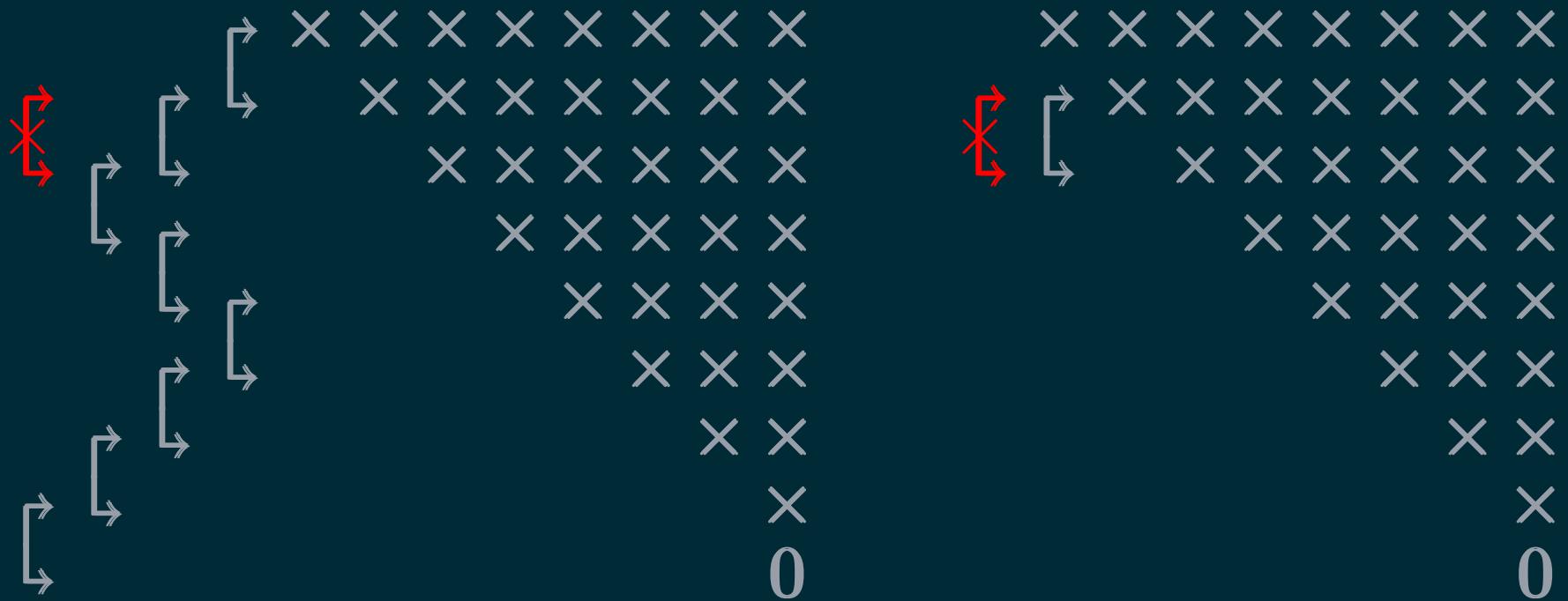
Chasing procedure

$(\underline{L}, \underline{K})$



Chasing procedure

$(\underline{L}, \underline{K})$



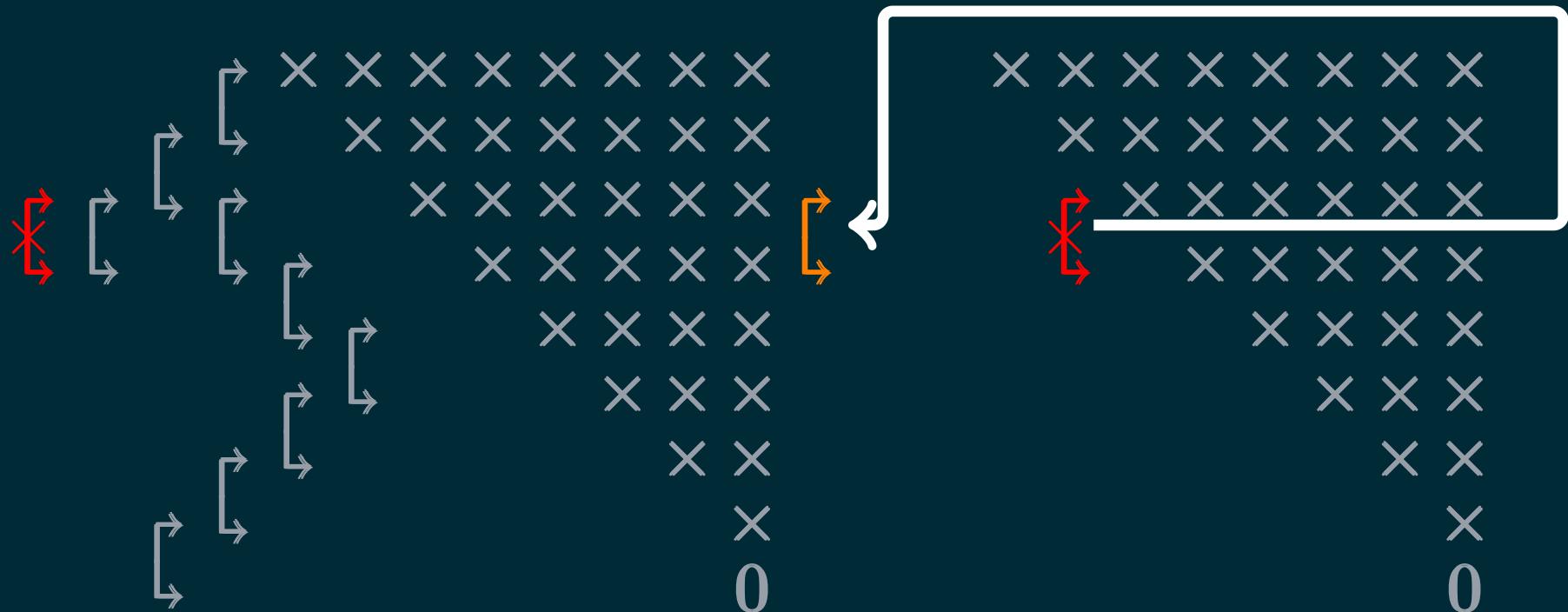
Chasing procedure

$(\underline{L}, \underline{K})$



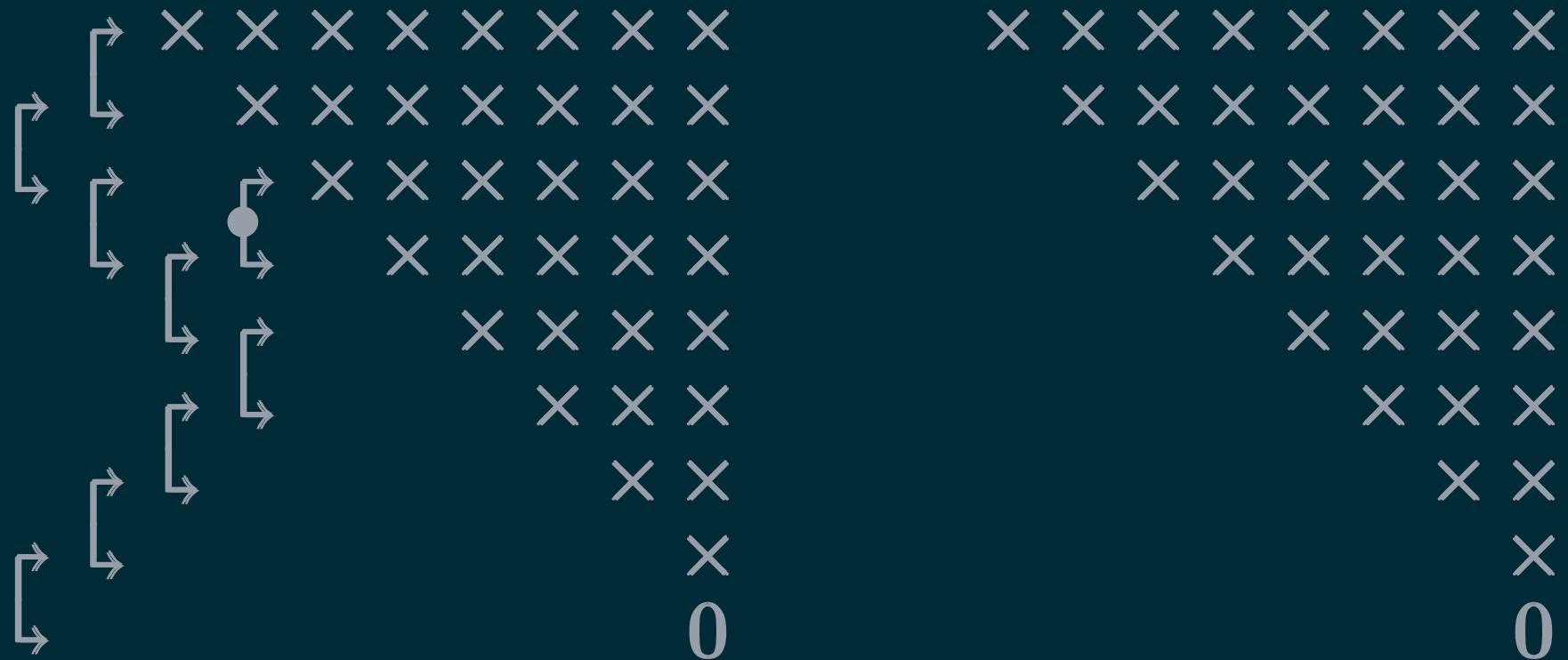
Chasing procedure

$$(\underline{L}, \underline{K})$$



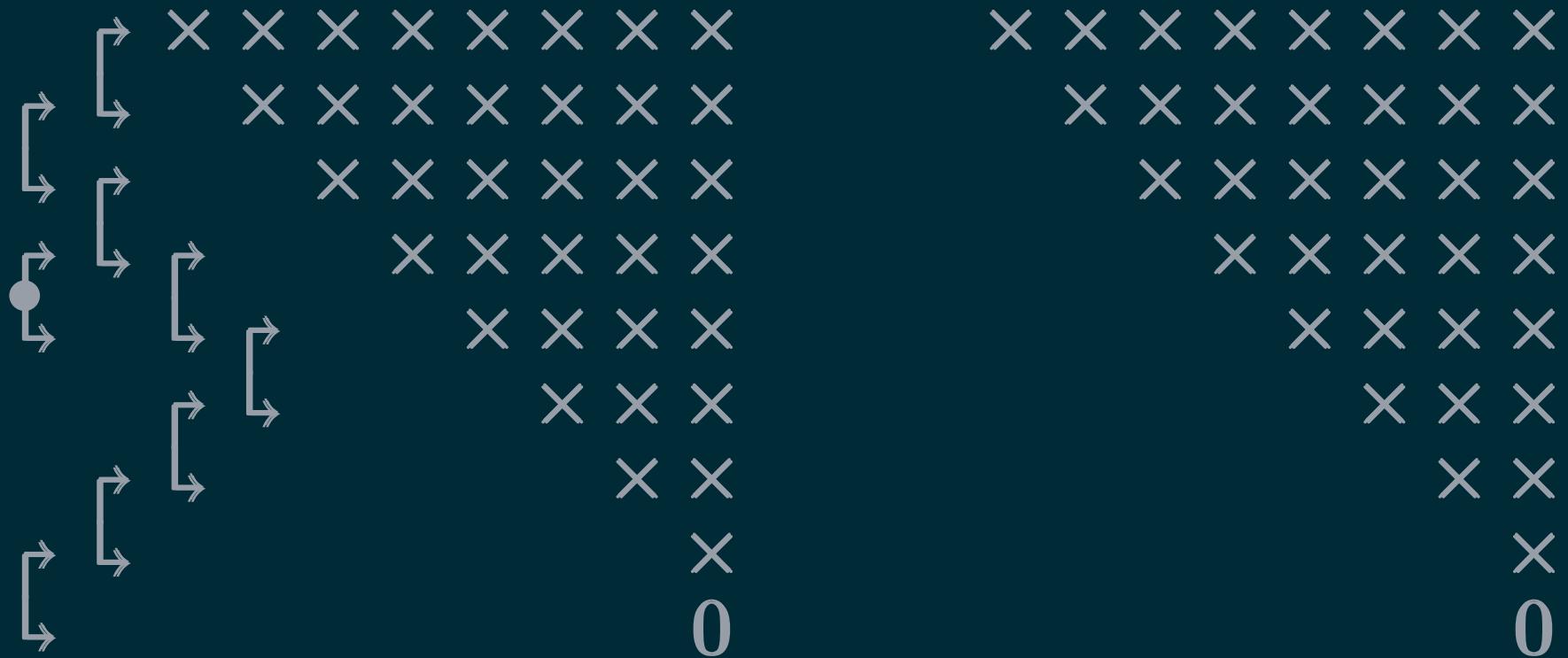
Chasing procedure

$$(\underline{L}, \underline{K})$$



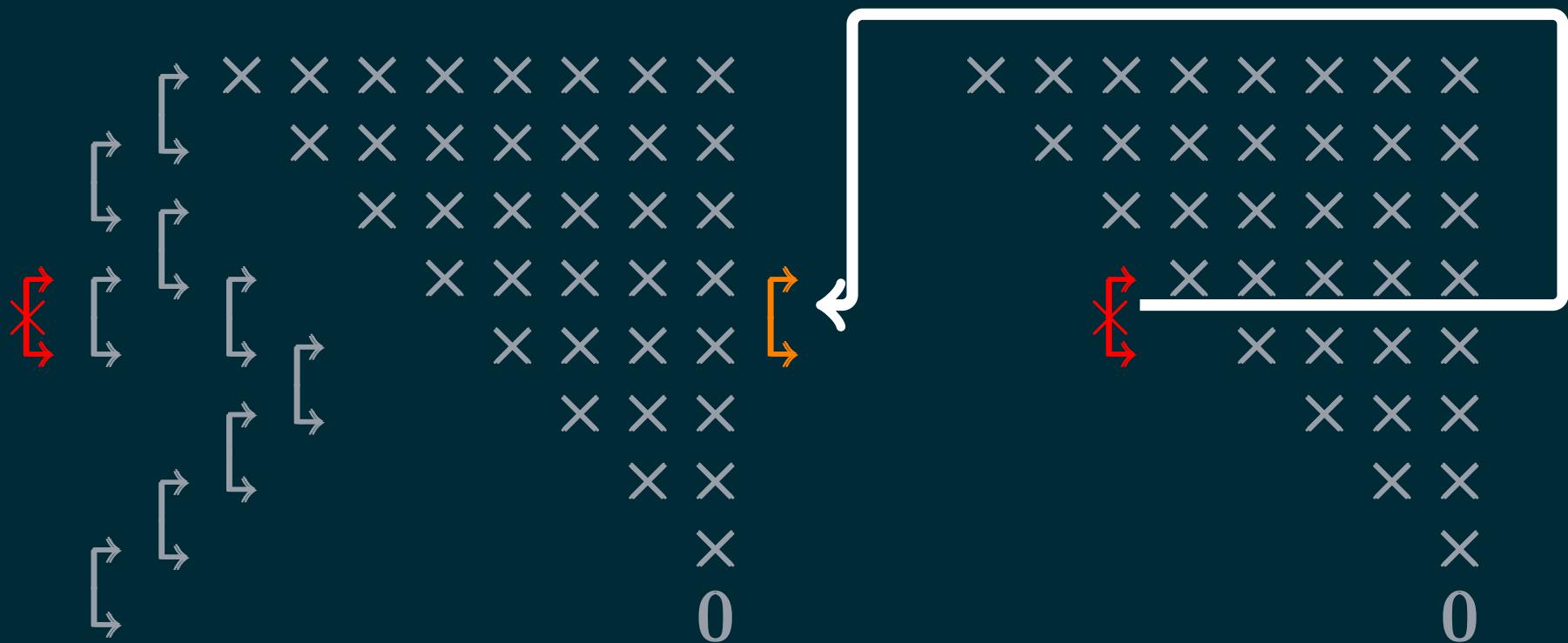
Chasing procedure

$$(\underline{L}, \underline{K})$$



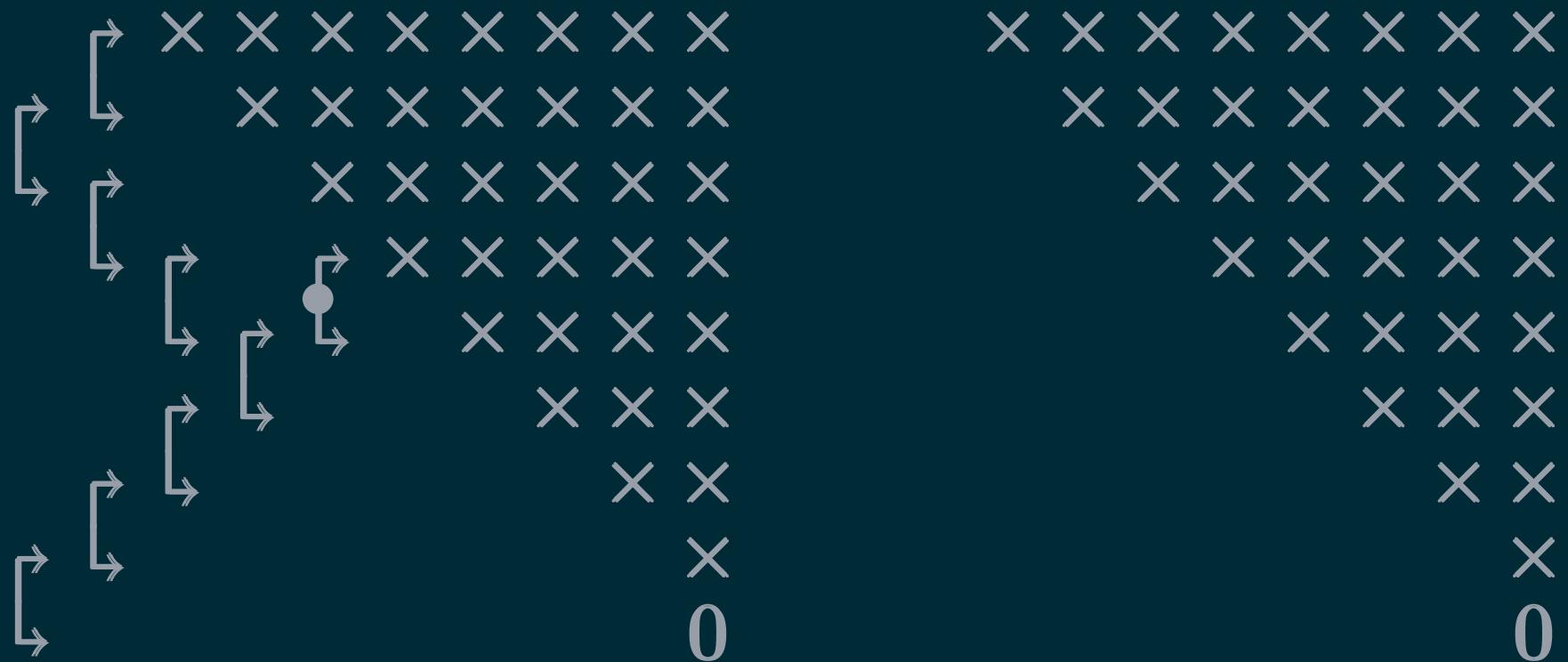
Chasing procedure

$$(\underline{L}, \underline{K})$$



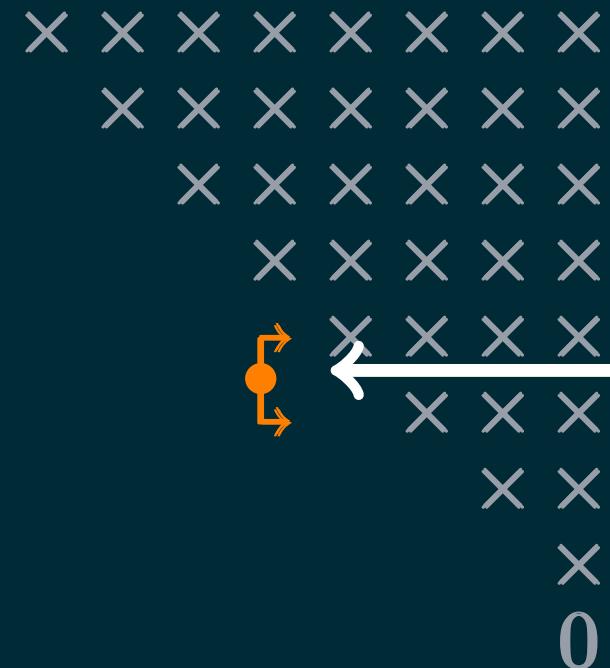
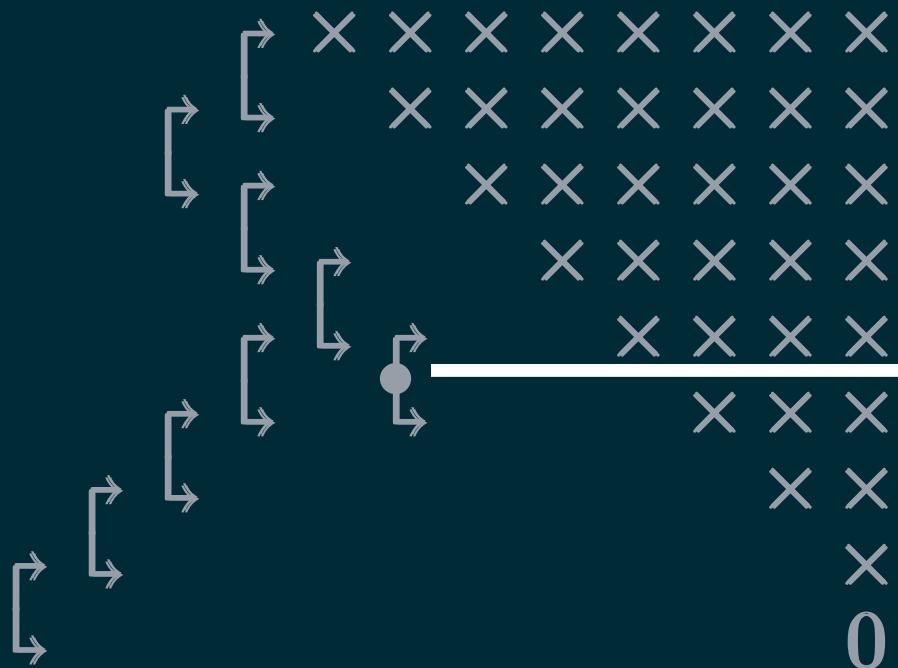
Chasing procedure

$(\underline{L}, \underline{K})$



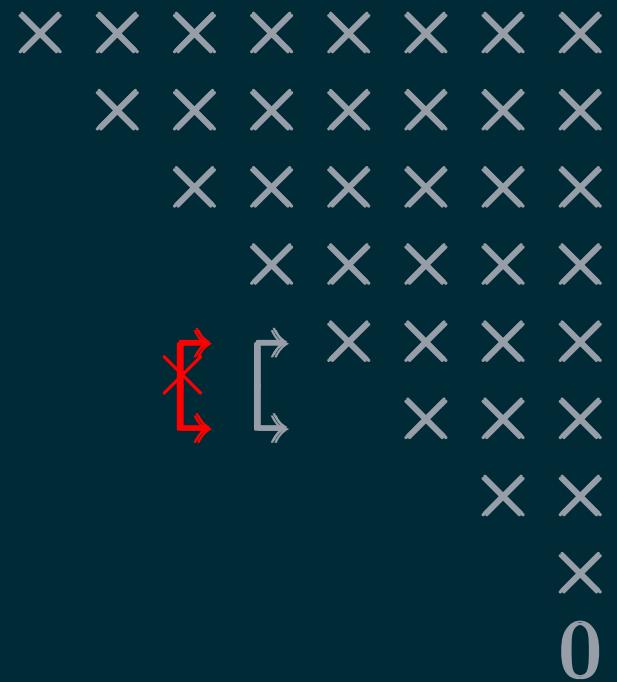
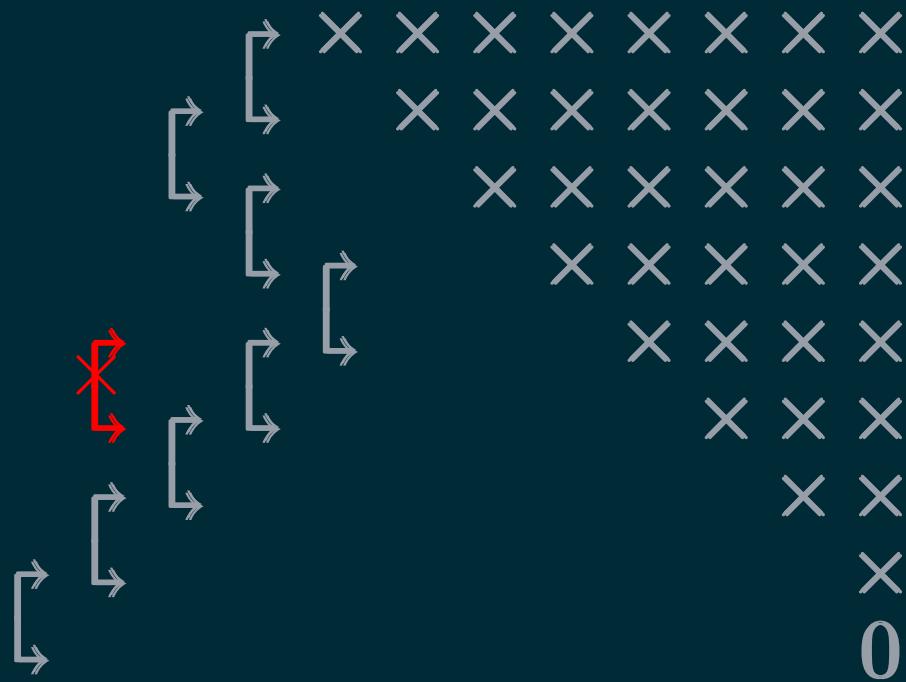
Chasing procedure

$(\underline{L}, \underline{K})$



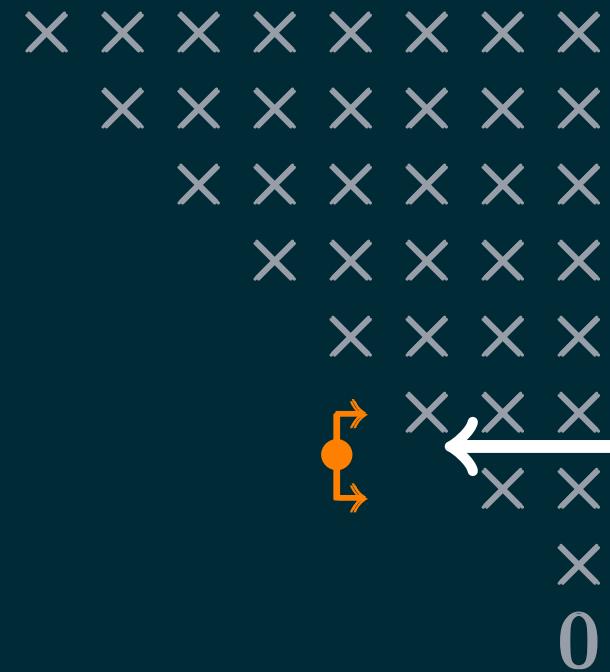
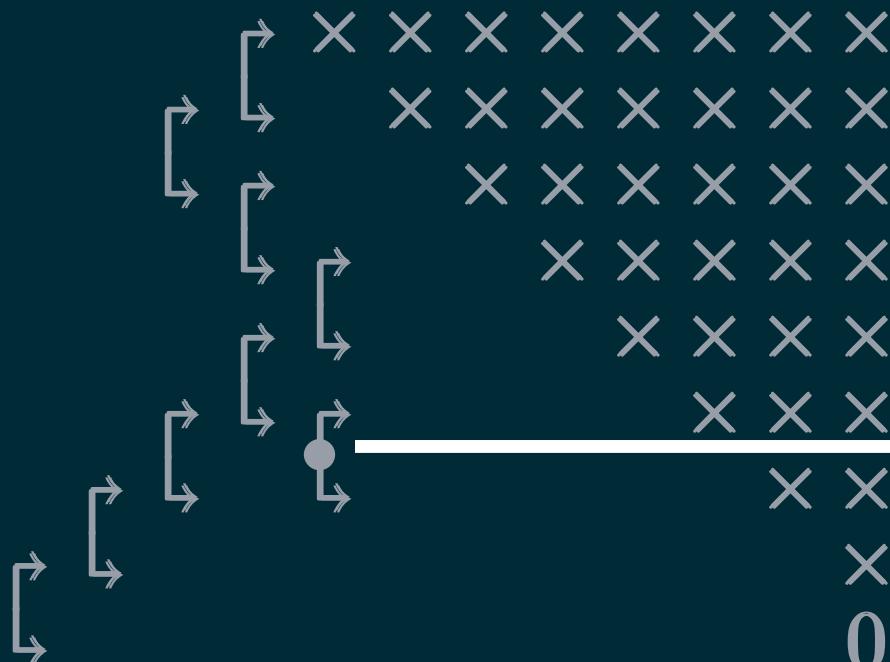
Chasing procedure

$(\underline{L}, \underline{K})$



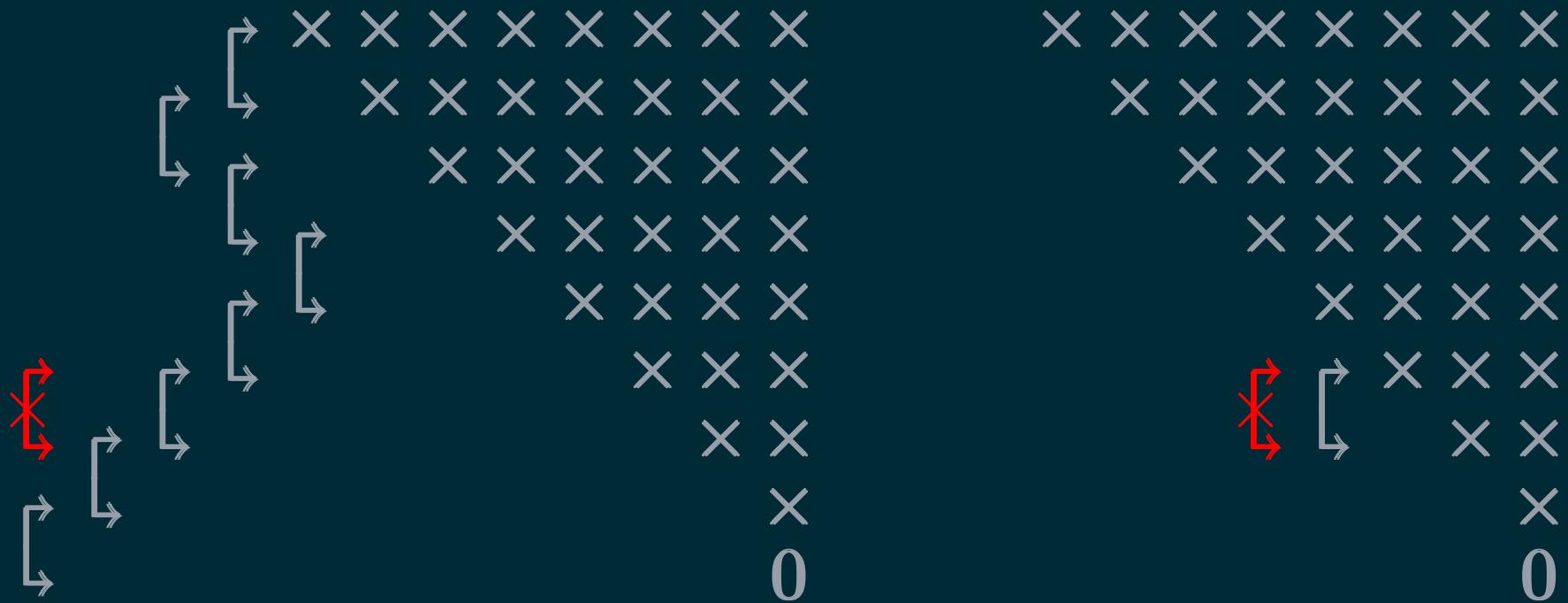
Chasing procedure

$(\underline{L}, \underline{K})$



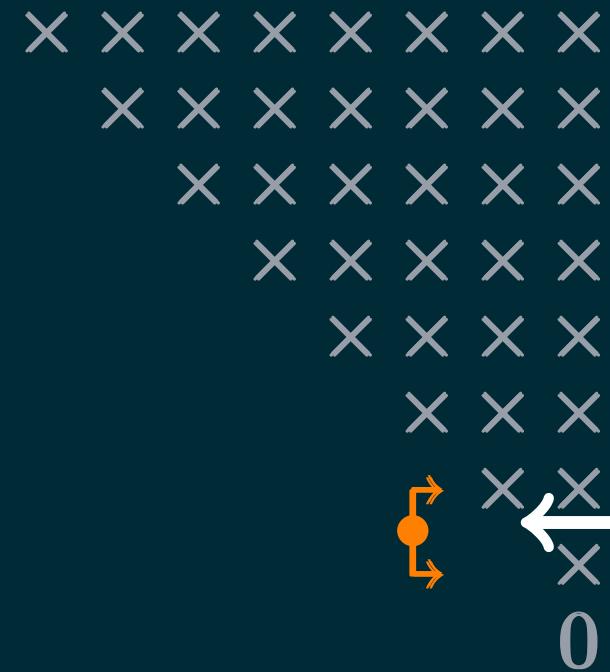
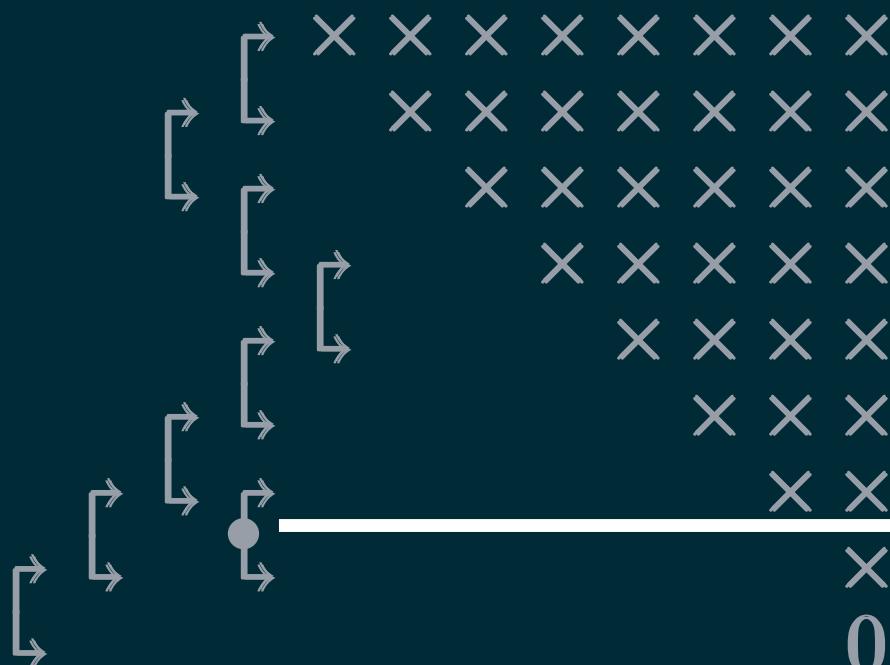
Chasing procedure

$(\underline{L}, \underline{K})$



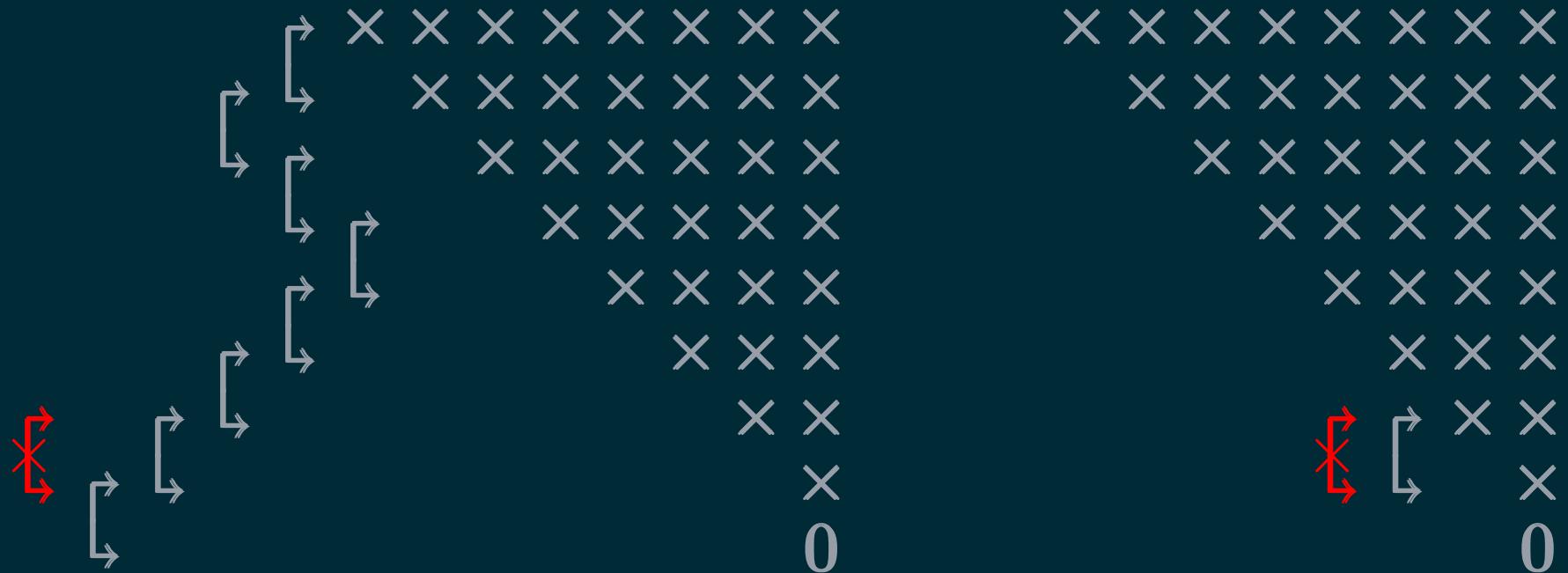
Chasing procedure

$(\underline{L}, \underline{K})$



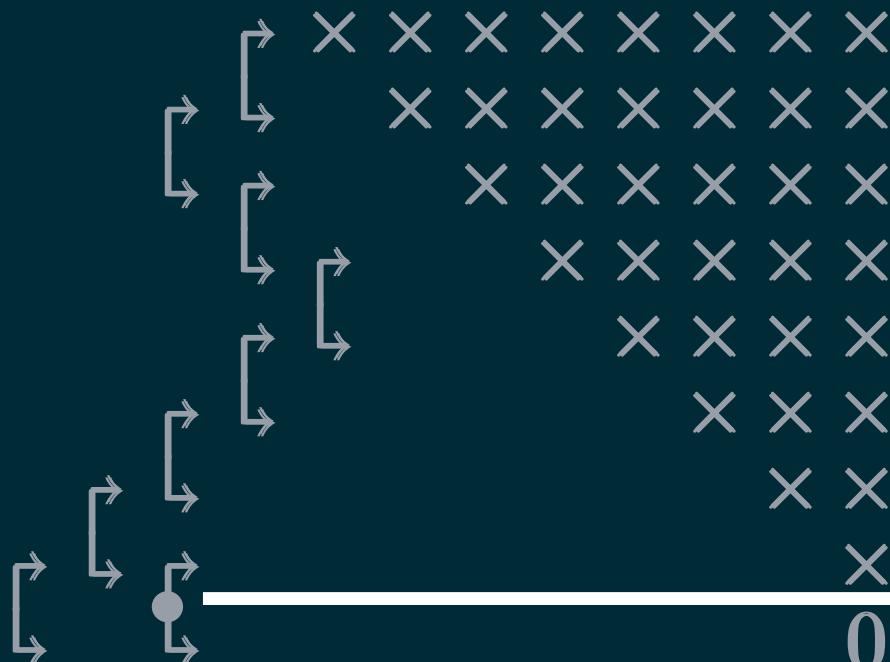
Chasing procedure

$(\underline{L}, \underline{K})$



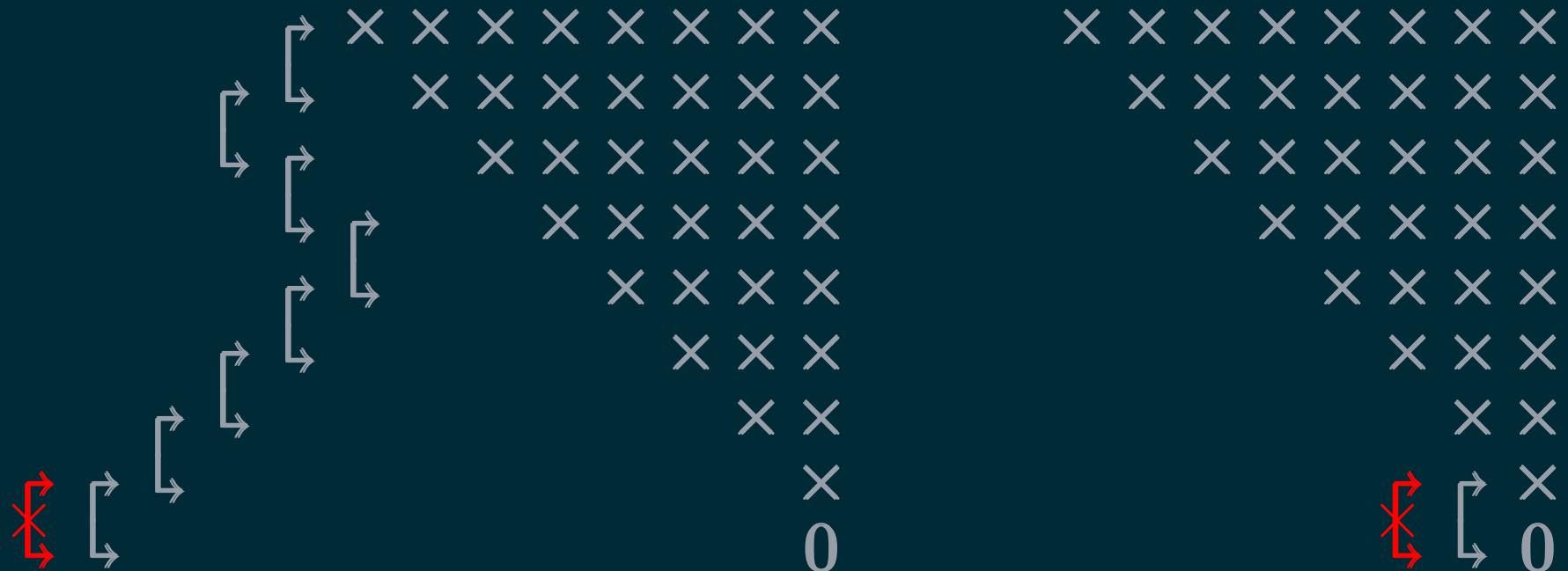
Chasing procedure

$(\underline{L}, \underline{K})$



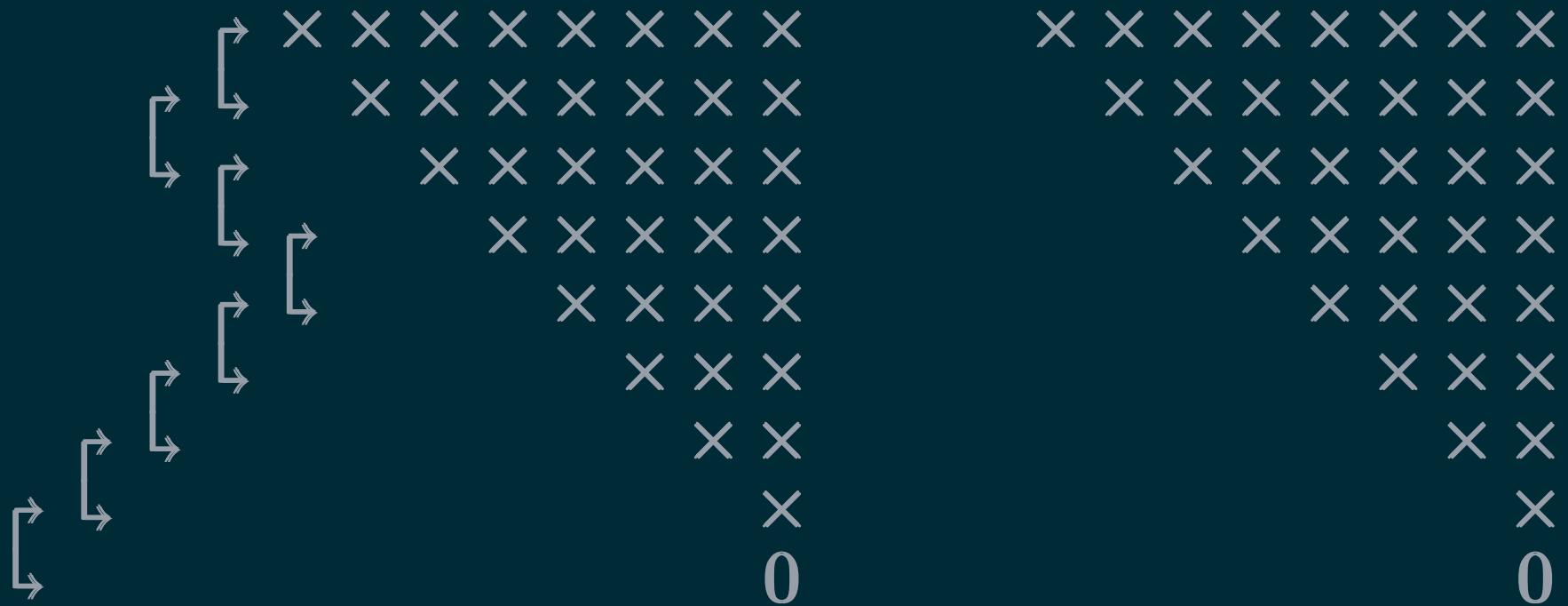
Chasing procedure

$(\underline{L}, \underline{K})$



Chasing procedure

$(\underline{L}, \underline{K})$



NUMERICAL EXAMPLE

BRUSSELATOR MODEL

[DE SAMBLANX, 1997]

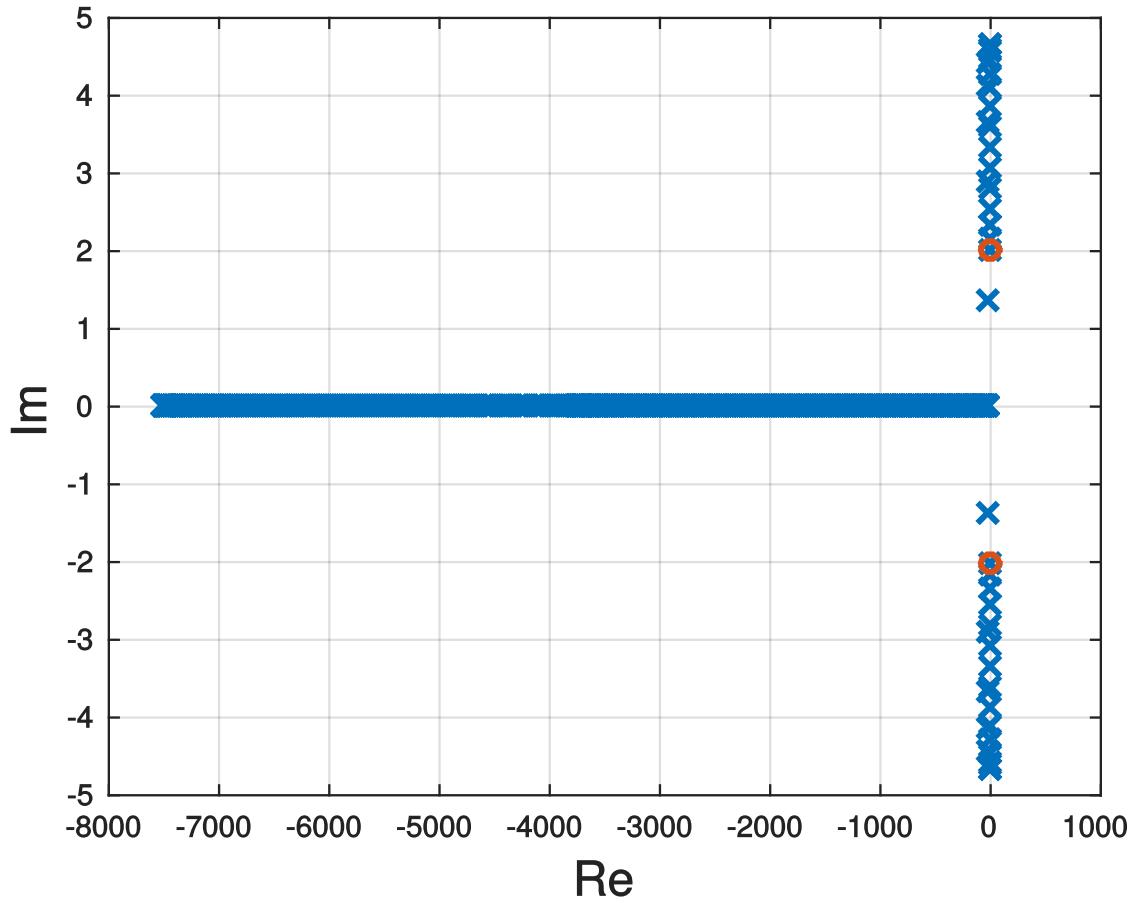
2D reaction-diffusion model

$$\frac{\partial u}{\partial t} = \frac{D_u}{L^2} \left[\frac{\partial^2 u}{\partial X^2} + \frac{\partial^2 u}{\partial Y^2} \right] - (B + 1)u + u^2v + C$$

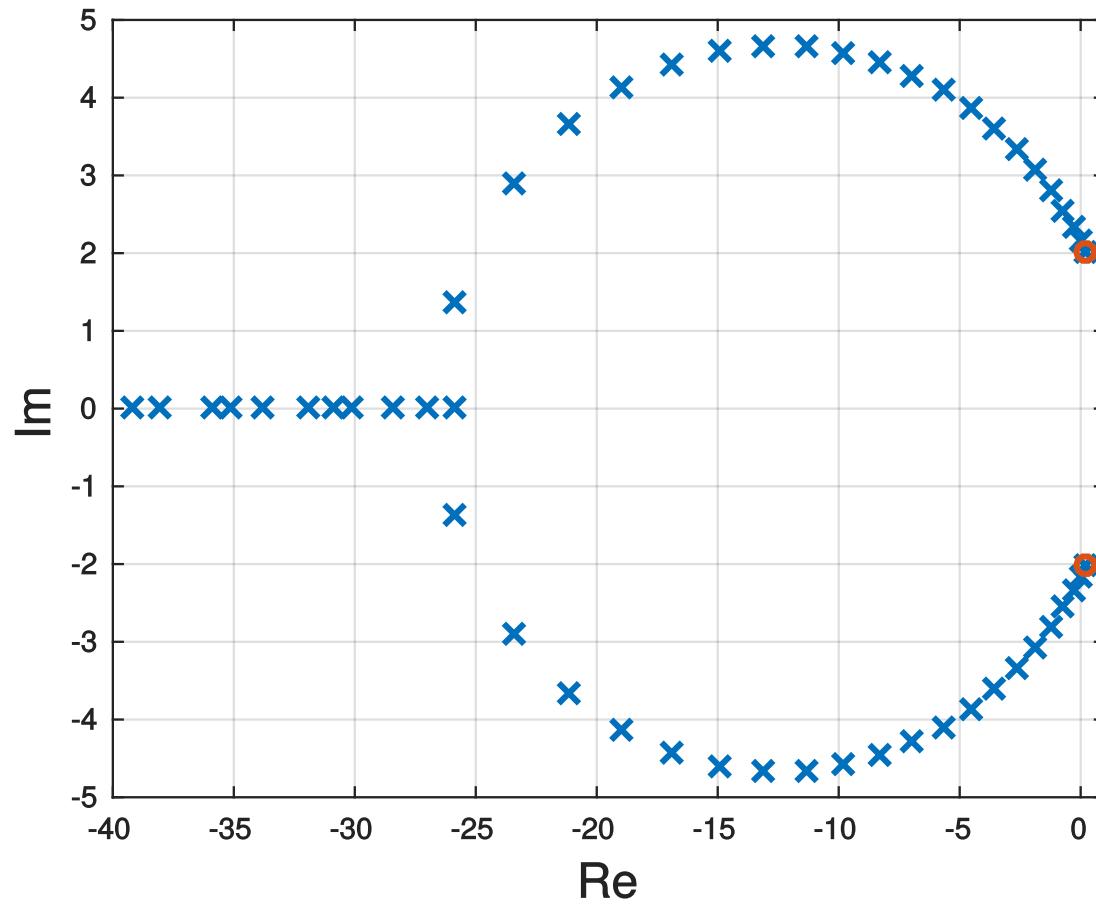
$$\frac{\partial v}{\partial t} = \frac{D_v}{L^2} \left[\frac{\partial^2 v}{\partial X^2} + \frac{\partial^2 v}{\partial Y^2} \right] - u^2v + Bu$$

- u, v : concentrations of two reactants
- homogeneous Dirichlet boundary conditions
- discretised with central differences
- parameters: $B = 5.45, C = 2, D_u = 0.004, D_v = 0.008$ and $L = 1$

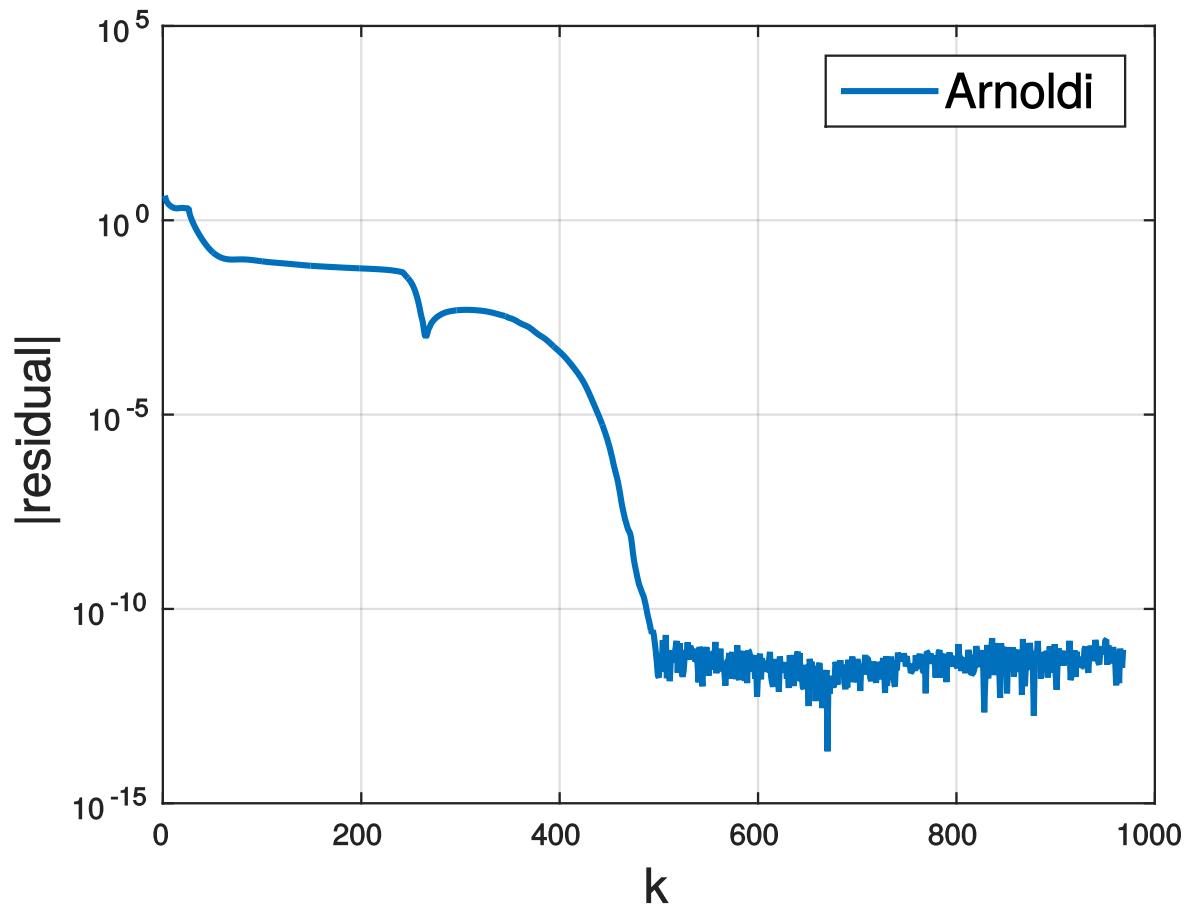
Matrix $A \in \mathbb{C}^{968 \times 968}$



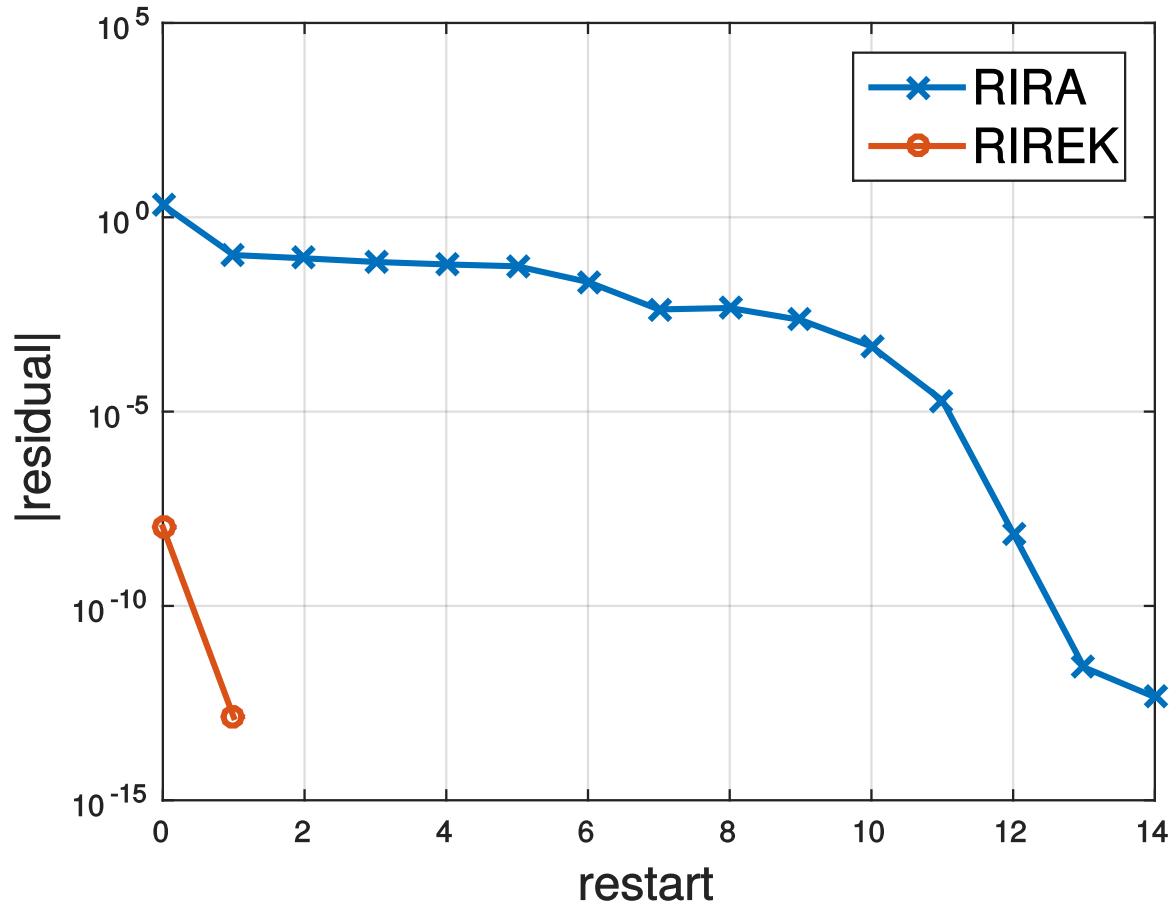
Matrix $A \in \mathbb{C}^{968 \times 968}$



Arnoldi



RIRA & RIREK



An aerial photograph of a vast, snow-covered mountain range. The terrain is rugged with deep blue shadows in the valleys and bright white snow on the ridges. A winding, dark grey path or stream bed cuts through the center of the image. At the bottom left, a small group of approximately ten people in colorful winter gear are walking along this path, their long shadows stretching across the snow.

CONCLUSION & OUTLOOK

Conclusion

- Krylov methods: Arnoldi & RKS
- Implicit restart of Arnoldi & EK in an implicit manner:
 - optimal computational complexity
 - multi-shift
 - unitary operations

Outlook

- Generalise to fully rational Krylov sequences
- Applications
- ...

REFERENCES