TECHNISCHE UNIVERSITÄT DRESDEN

OpenModelica Workshop 2013

Chair of Construction Machines and Conveying Technology

# Comparison of Tearing Algorithms

Volker Waurich

Linköping, 04/02/2013

DRESDEN concept
Exzellenz aus Wissenschaft und Kultur

# Outline

1. the concept of tearing
2. classification of tearing algorithms
3. some tearing methods
4. evaluation

# the concept of tearing

tearing:

- symbolic method for large, sparse systems
- for linear, non-linear and mixed systems
- using graph theory
- speed up
- higher robustness

# the concept of tearing

## pre-work/preparation

- equation system
- algebraic representation
- partitioning/ precedence ordering (BLTF)

## tearing

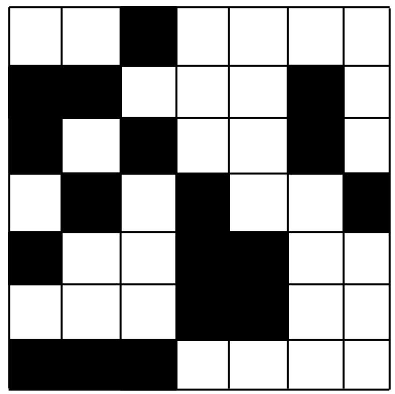
- tearing heuristic and output assignment

- numerical computation

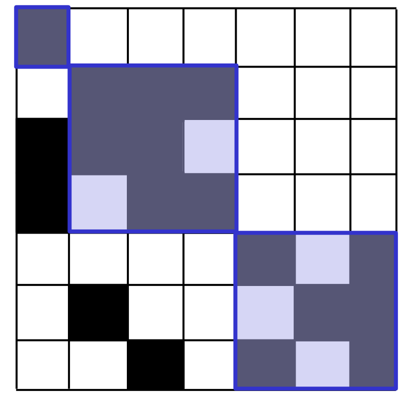
# the concept of tearing

## pre-work/preparation

$$f_1 = f(x_3)$$
$$f_2 = f(x_1 x_2, x_6)$$
$$f_3 = f(x_1, x_3, x_6)$$
$$f_4 = f(x_2, x_5, x_7)$$
$$f_5 = f(x_1 x_4, x_5)$$
$$f_6 = f(x_4, x_5\ )$$
$$f_7 = f(x_1, x_2, x_3)$$



*structure-incidence matrix*



*block lower triangular form*

**equation system**

- implicit equations

**algebraic representation**

- incidence matrix or adjacency matrix

**partitioning**

- tearing of each block (algebraic loops)

TECHNISCHE UNIVERSITÄT DRESDEN

# the concept of tearing

## basic principle of tearing

- tearing of algebraic loops
- assuming variables to be known (tearing variables)
- solve remaining equations
- iterate tearing variables with

residual equations (Newton iteration)

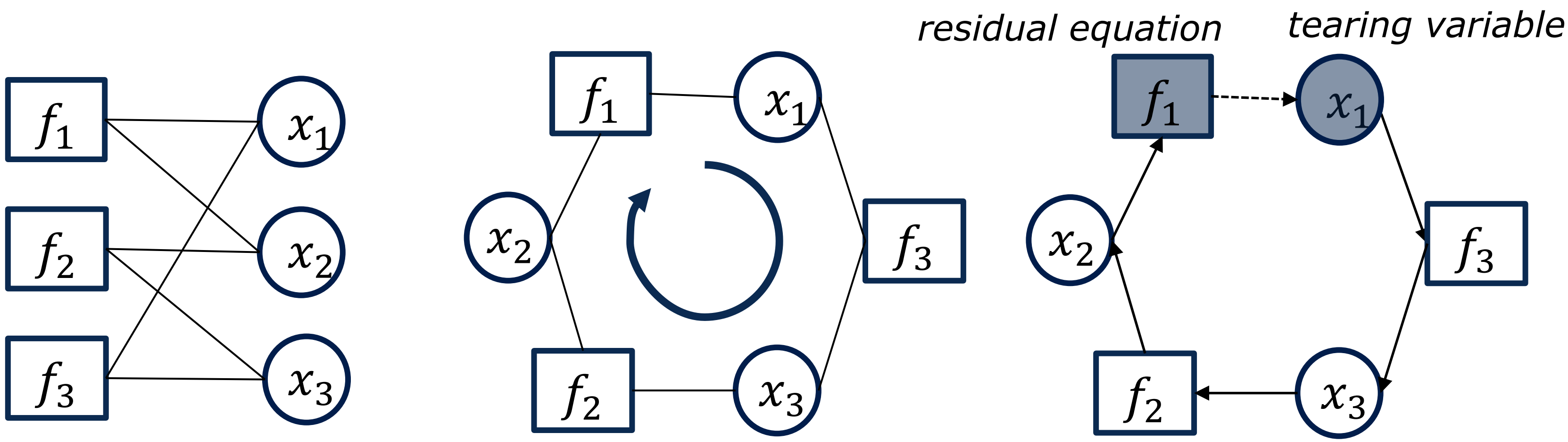
→ the aim is to choose the tearing set with the least number of variables

→ only heuristic methods exist to choose variables in polynomial time (proven to be NP-hard)

→ solvability has to be considered

# the concept of tearing

## basic principle of tearing




1. bipartite graph
2. algebraic loop
3. tear the loop

4. solving with x1 → f3 for x3 → f2 for x2 (output assignment)

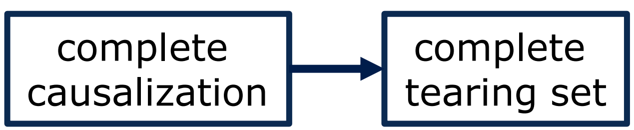
5. Newton iteration $x_{1new} = x_{1old} - \frac{f_1(x_{1old})}{f'_1\,(x_{1old})}$

# classification

## tearing = output assignment + variable selection

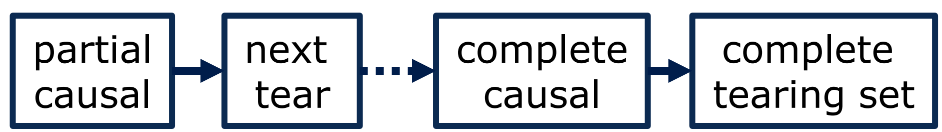
### previously matched

- output assignment is done before selection (Steward)
- works on digraph
- e.g. Steward, Ollero-Amselem

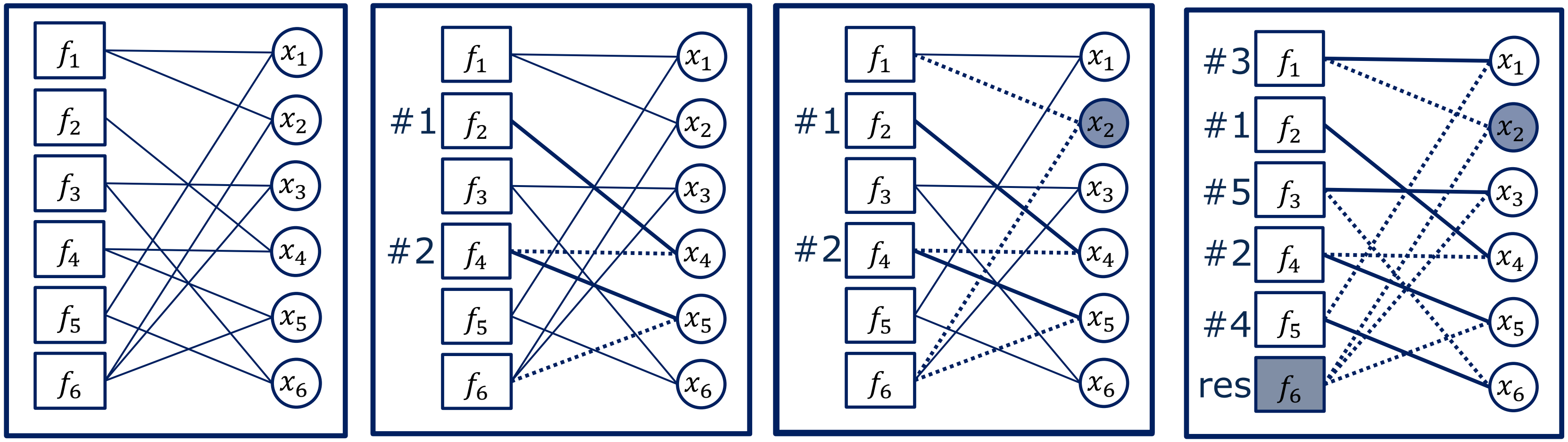



### simultaneously matched

- output assignment is done during selection (Tarjan/Cellier)
- works on bipartite graph
- e.g. Cellier, Carpanzano



## Celliers algorithm




bipartite graph | partially causalized | tearing selection | residual equation

# tearing methods

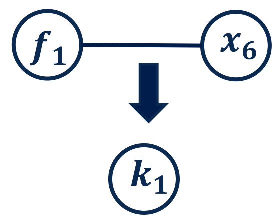
## Carpanzanos algorithm

- simultaneously matched
- similar to Cellier
- considers solvability of equations during tearing selection

→ see omcTearing (omc default)
selection weights considering rearranging effort

# tearing methods

## Stewards algorithm




| | $x_1$ | $x_2$ | $x_3$ | $x_4$ | $x_5$ | $x_6$ |
|---|---|---|---|---|---|---|
| $f_1$ | 1 | 0 | 0 | 0 | 1 | 1 |
| $f_2$ | 0 | 1 | 0 | 1 | 0 | 0 |
| $f_3$ | 0 | 1 | 1 | 1 | 0 | 0 |
| $f_4$ | 1 | 0 | 1 | 0 | 0 | 0 |
| $f_5$ | 1 | 0 | 0 | 1 | 0 | 1 |
| $f_6$ | 0 | 1 | 0 | 0 | 1 | 0 |

structure-incidence matrix

| | $k_1$ | $k_2$ | $k_3$ | $k_4$ | $k_5$ | $k_6$ |
|---|---|---|---|---|---|---|
| $k_1$ | 0 | 0 | 0 | 0 | 1 | 1 |
| $k_2$ | 0 | 0 | 1 | 0 | 0 | 0 |
| $k_3$ | 0 | 1 | 0 | 1 | 0 | 0 |
| $k_4$ | 0 | 0 | 0 | 0 | 1 | 0 |
| $k_5$ | 1 | 0 | 1 | 0 | 0 | 0 |
| $k_6$ | 0 | 1 | 0 | 0 | 0 | 0 |

adjacency matrix
k1: f1 x6
k2: f2 x2
k3: f3 x4
k4: f4 x3
k5: f5 x1
k6: f6 x5

| | $k_1$ | $k_2$ | $k_3$ | $k_4$ | $k_5$ | $k_6$ |
|---|---|---|---|---|---|---|
| $k_1$ | 0 | 0 | 0 | 0 | 1 | 1 |
| $k_2$ | 0 | 0 | 1 | 0 | 0 | 0 |
| $k_3$ | 0 | 1 | 0 | 1 | 0 | 0 |
| $k_4$ | 0 | 0 | 0 | 0 | 1 | 0 |
| $k_5$ | 1 | 0 | 1 | 0 | 0 | 0 |
| $k_6$ | 0 | 1 | 0 | 0 | 0 | 0 |

loop finding
l1: k1-k5
l2: k2-k3
l3: k4-k5-k3
l4: k6-k2-k3-k4-k5-k1

| | $k_1$ | $k_2$ | $k_3$ | $k_4$ | $k_5$ | $k_6$ |
|---|---|---|---|---|---|---|
| $l_1$ | 5 | 0 | 0 | 0 | 1 | 0 |
| $l_2$ | 0 | 3 | 2 | 0 | 0 | 0 |
| $l_3$ | 0 | 0 | 4 | 5 | 3 | 0 |
| $l_4$ | 6 | 3 | 4 | 5 | 1 | 2 |

find tearing set

# tearing methods

## Ollero-Amselems algorithm

- previously matched
- works with contraction of nodes in the digraph
- if: contraction causes self-loops

then: tearing variable found and removed from graph

## comparison

### previously matched

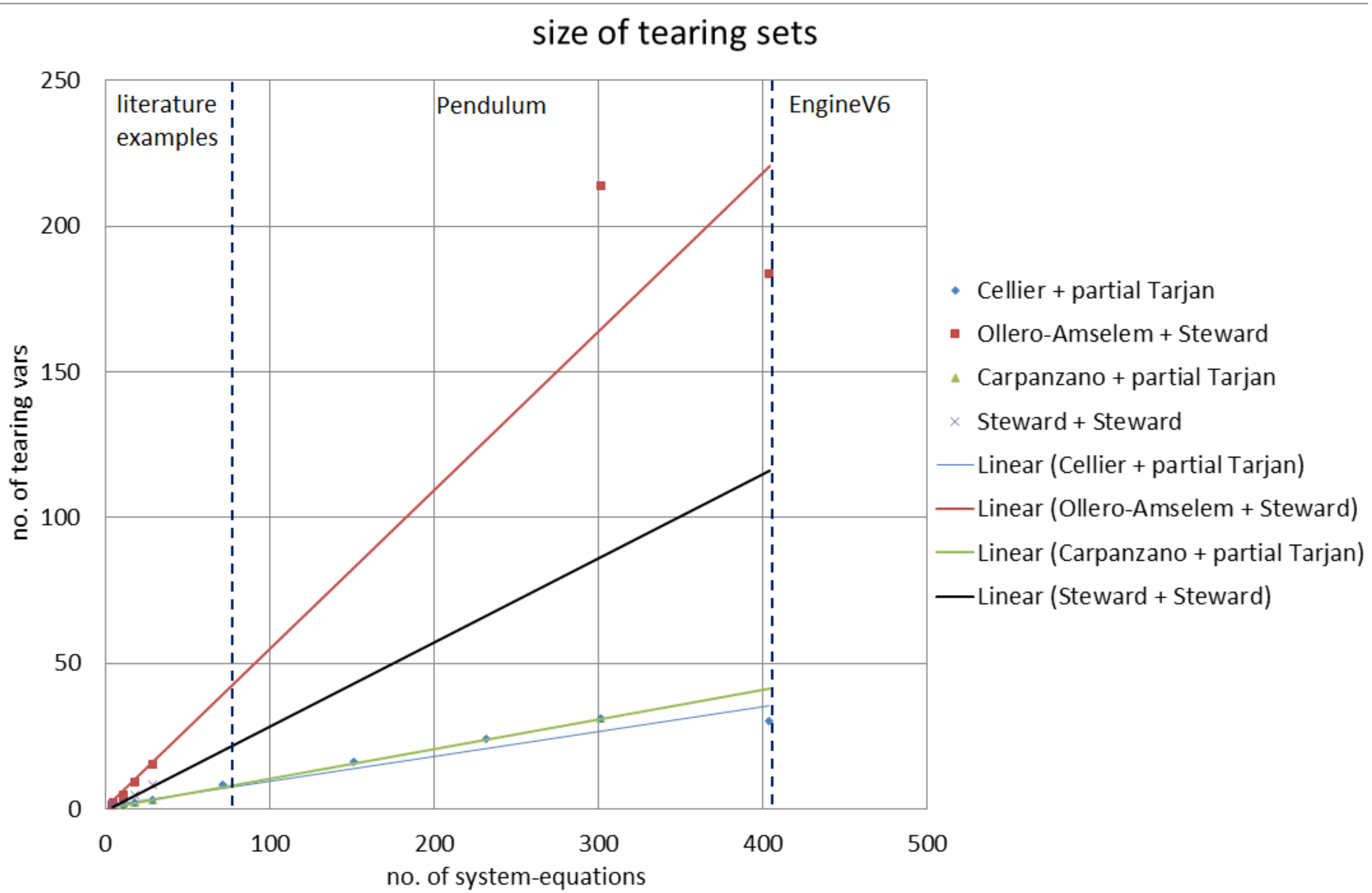
- works on matched-system-graph
- higher computational effort
- re-transformation from matched system

### simultaneously matched

- works on incidence matrix
- output-assignment, precedence-ordering and tearing at once

→ this concept will be pursued

# evaluation


size of tearing sets
literature examples
Pendulum
EngineV6
no. of tearing vars
no. of system-equations
0
50
100
150
200
250
0
100
200
300
400
500
Cellier + partial Tarjan
Ollero-Amselem + Steward
Carpanzano + partial Tarjan
Steward + Steward
Linear (Cellier + partial Tarjan)
Linear (Ollero-Amselem + Steward)
Linear (Carpanzano + partial Tarjan)
Linear (Steward + Steward)

TECHNISCHE UNIVERSITÄT DRESDEN

# conclusion

- simultaneously matched tearing method is more effective (smaller tearing set)
- less administrative overhead for the simultaneously matched method
- previous matching is not unique and may effect tearing selection
- solvability has to be considered during selection

# outlook

- finish implementation
- manual selection via annotation
- choice of residual equation
- improve tearing algorithm

TECHNISCHE UNIVERSITÄT DRESDEN

»Wissen schafft Brücken.«

thank you for your attention