PIC S&T '2019

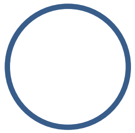
Clans Aggregation for Verification of Networking Protocols on Parallel Architectures

Dmitry A. Zaitsev
Information Technology Department
Odessa State Environmental University
Odessa, Ukraine
daze@acm.org
http://daze.ho.ua

Tatiana R. Shmeleva
Switching Systems Department
A.S.Popov Odessa National Academy of Telecommunications
Odessa, Ukraine
t.shmeleva@onat.edu.ua

Alexander A. Kostikov
Informatics and Engineering Graphics Department
Donbass State Engineering Academy
Kramatorsk, Ukraine
alexkst63@gmail.com

Elements of a Petri net



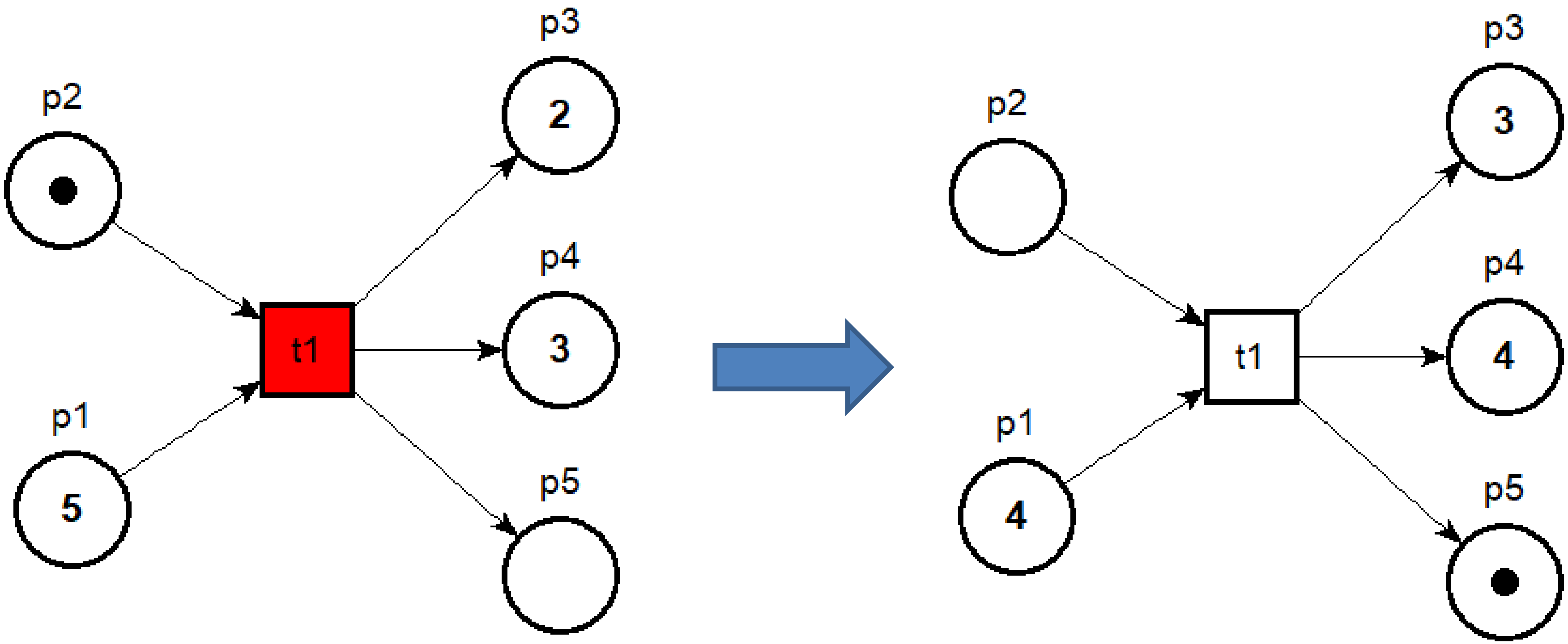
place

transition

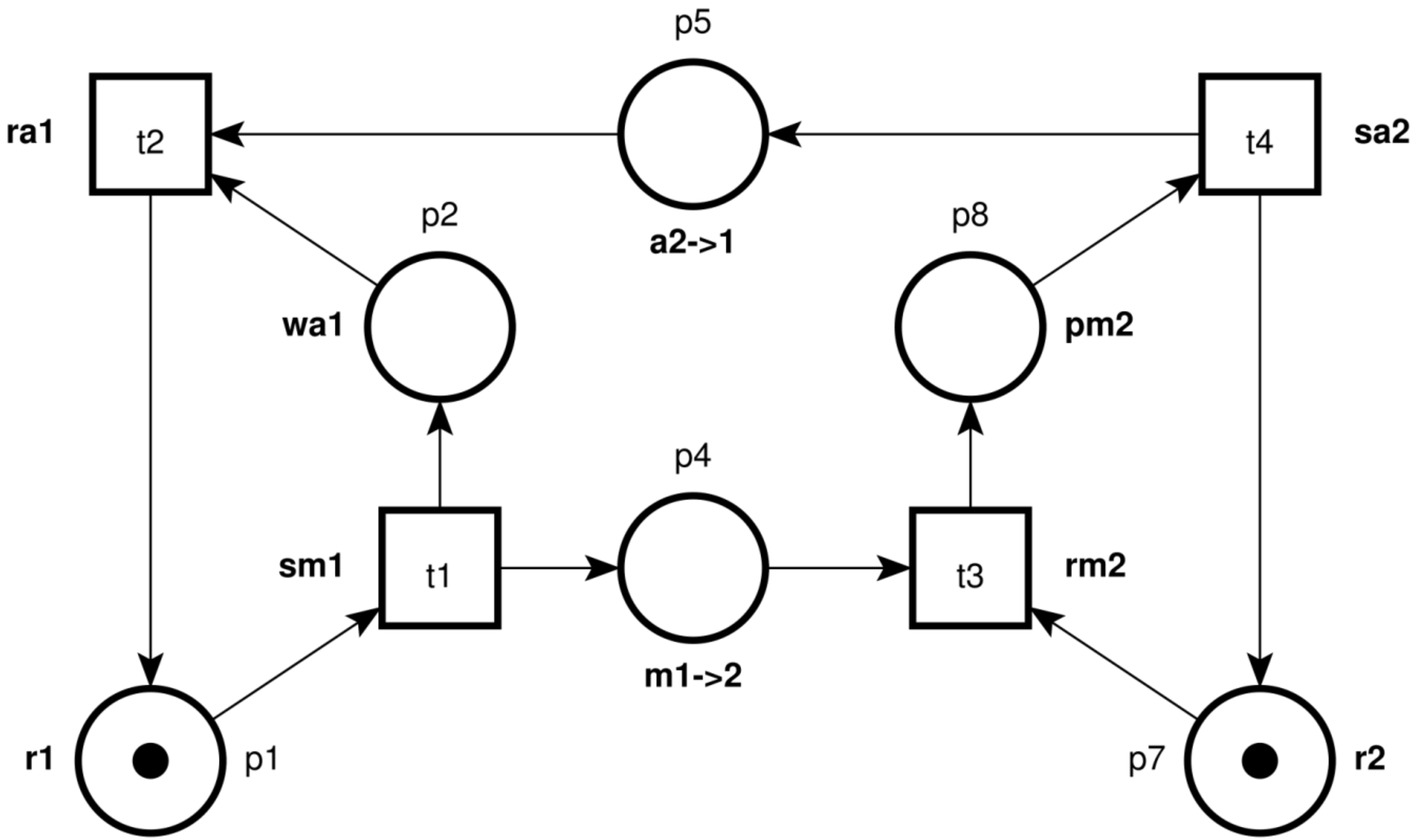
arc

token

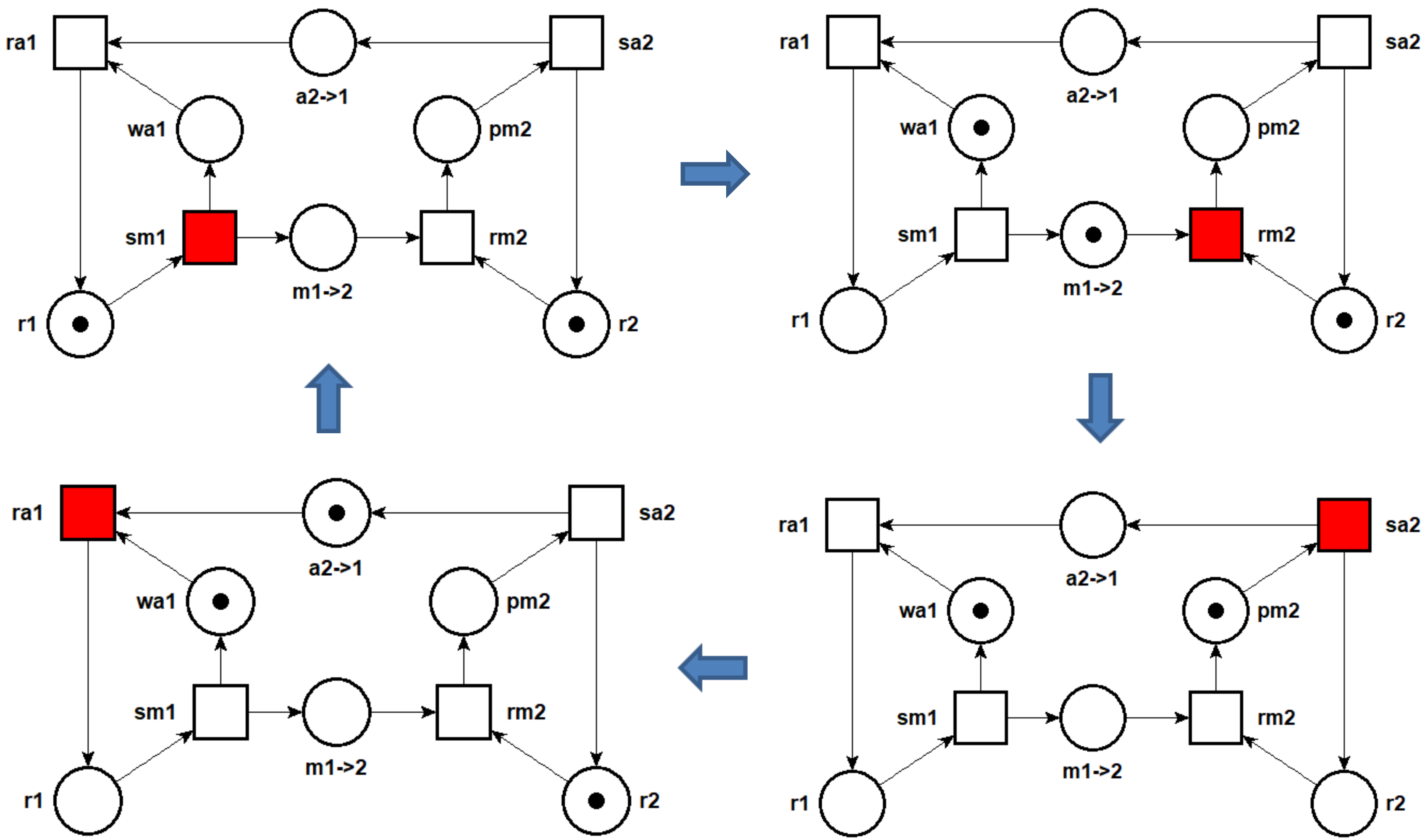
Petri net transition firing



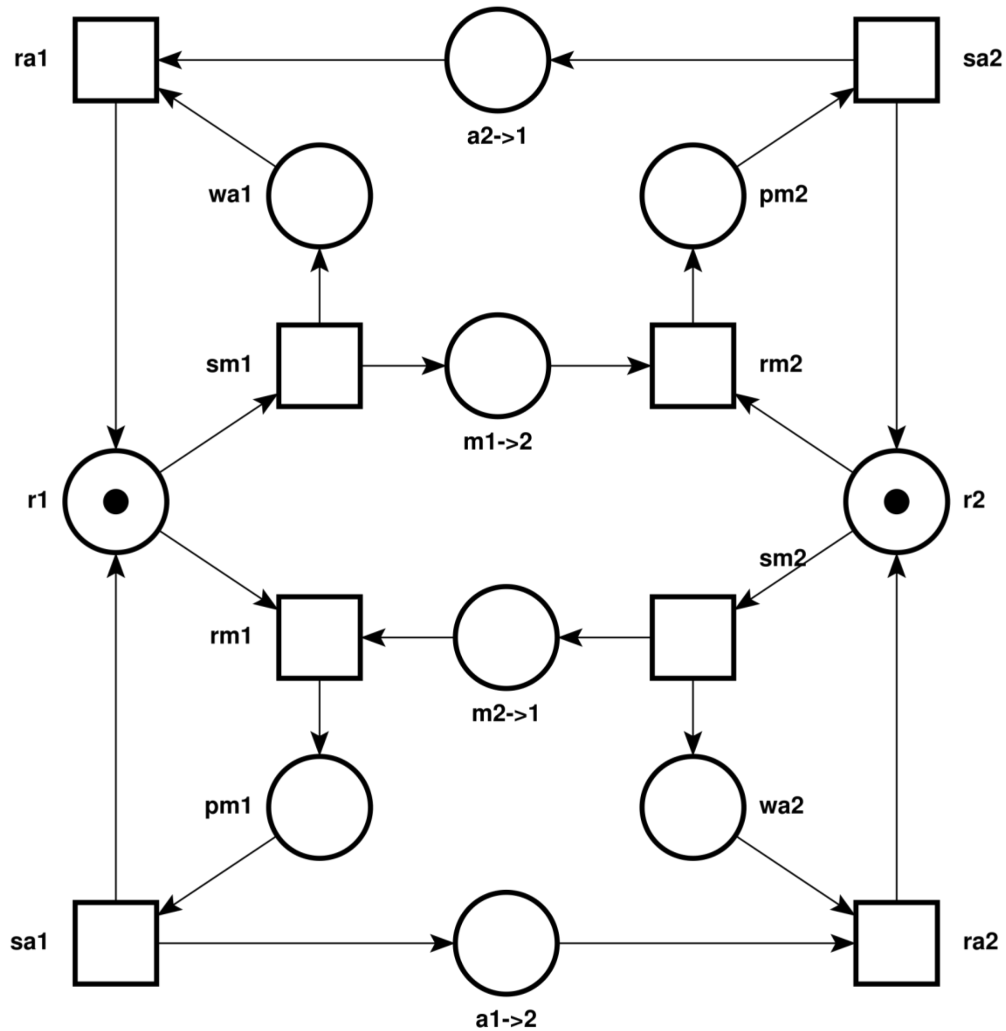
Protocol of one-way transmission with acknowledgements



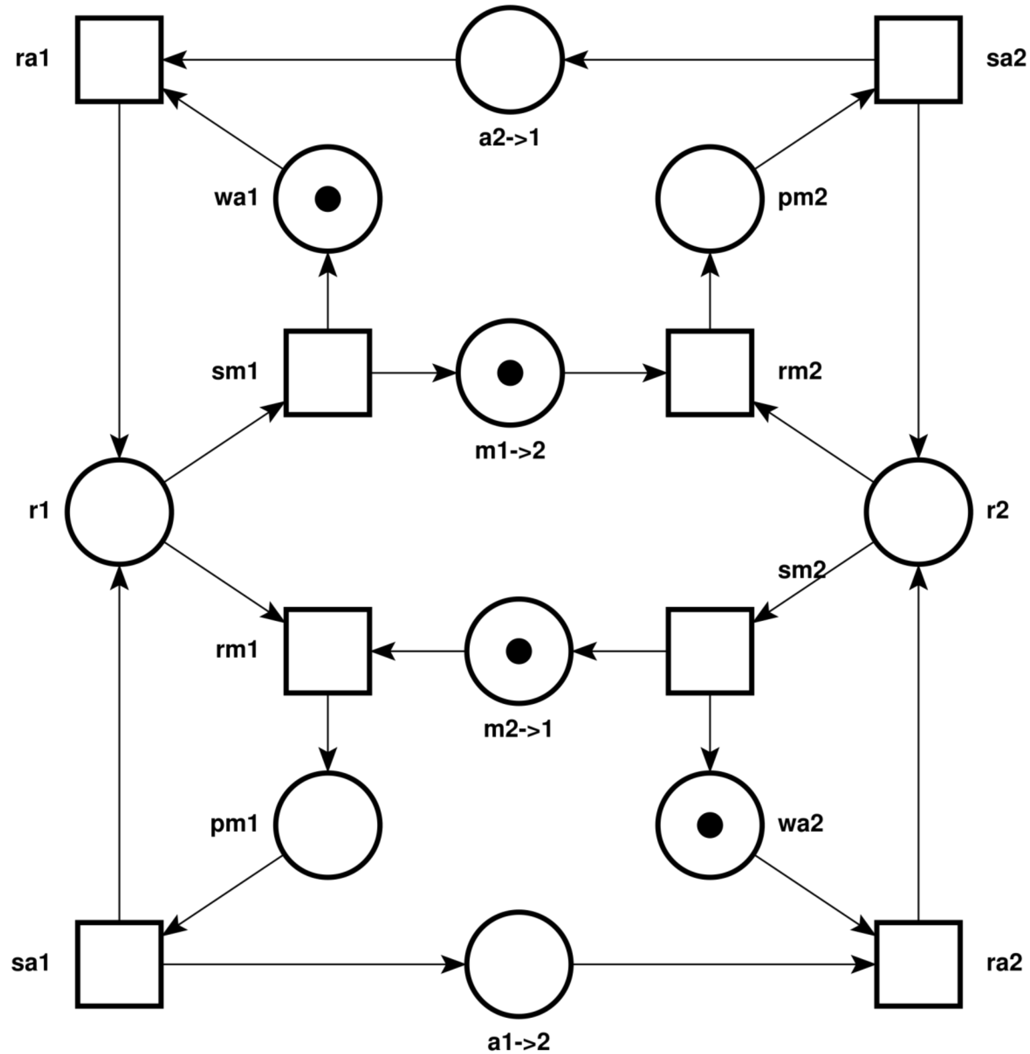
Petri net behavior



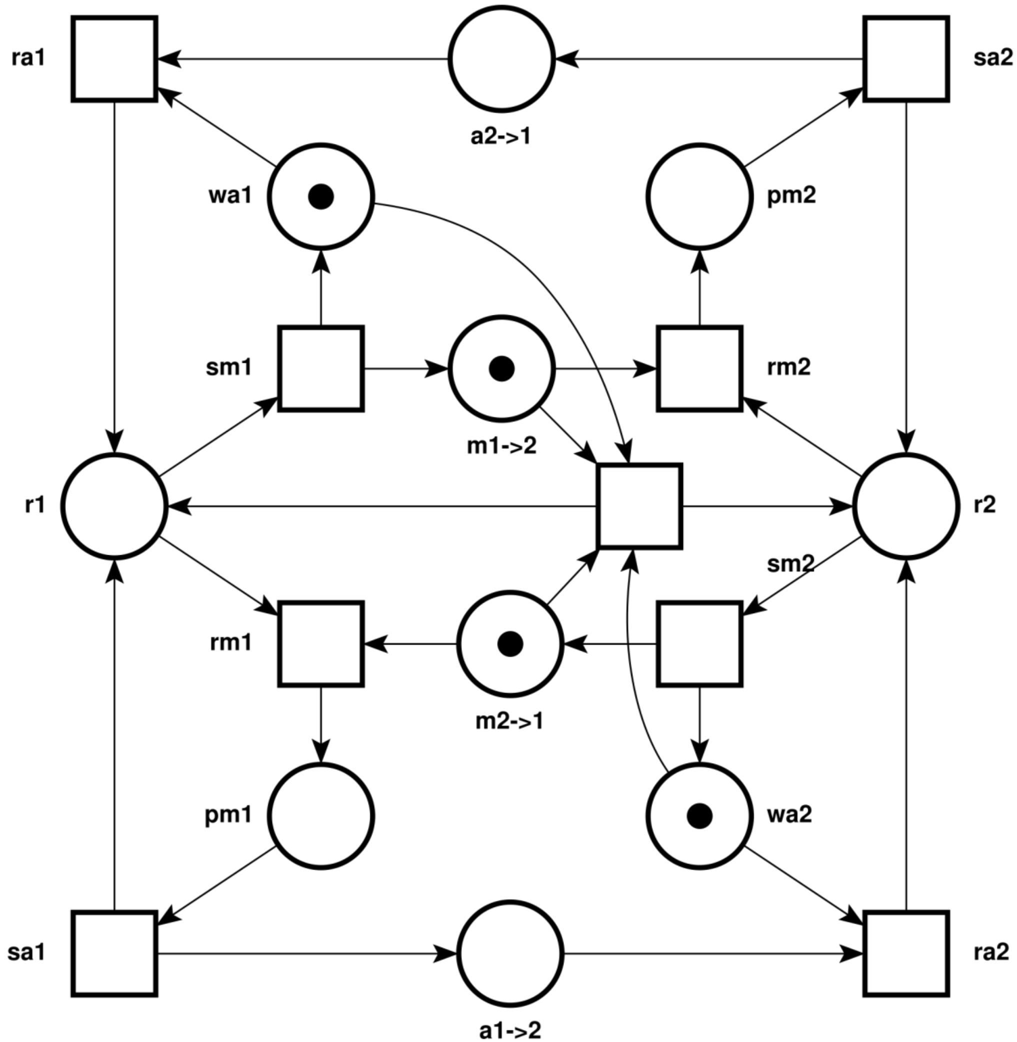
Protocol of two-way transmission with acknowledgements



Deadlock


ra1
sa2
a2->1
wa1
pm2
sm1
rm2
m1->2
r1
r2
sm2
rm1
m2->1
pm1
wa2
sa1
ra2
a1->2

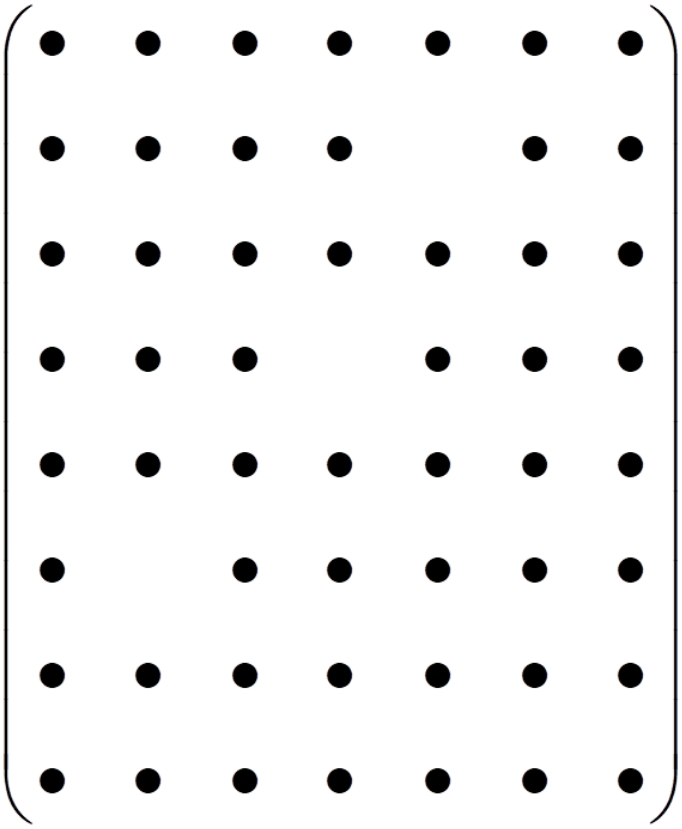
Modified protocol: collisions detection

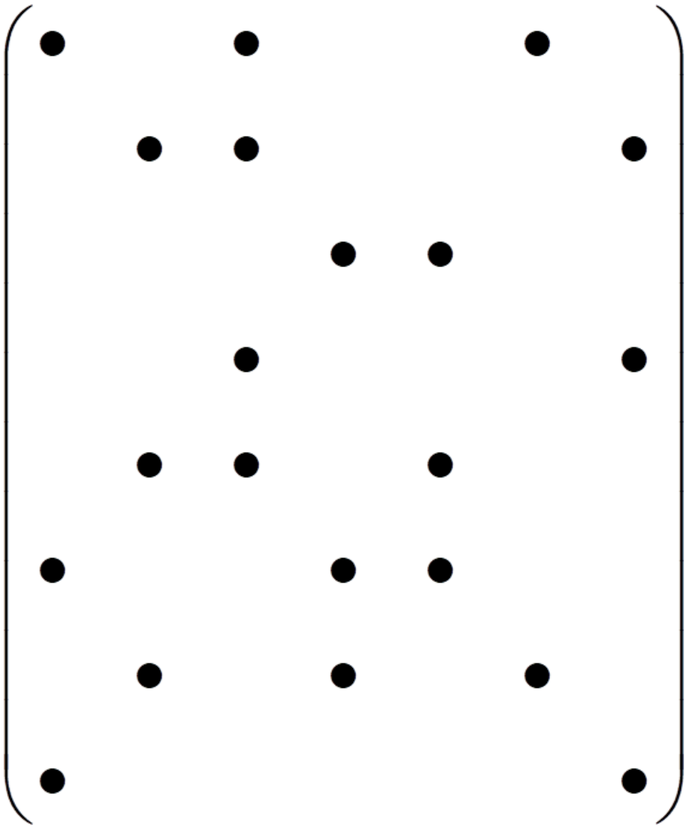


Methods of Petri net analysis

- Reachability and coverability tree
- **Fundamental equation and linear invariants – solving linear systems over integer nonnegative numbers**
- Siphons and traps
- Reduction – transformations preserving properties
- **Decomposition – divide into parts**

Dense and sparse systems





Algebraic structure

Numbers	Example	Structure	Methods
Complex	-3,2+6,25i	field	a) reduction: LU, QR; б) iteration methods
Real	0,25; -78,931		
Integer	-33; 0; 6	ring	Normal forms: Hermite, Smith
Nonnegative integer	0; 7; 55	monoid	Methods of Toudic (Silva) and Contejean

Real-life models

- Model Checking Contest – Petri net models https://mcc.lip6.fr
- Matrix Market - https://math.nist.gov/MatrixMarket
- The SuiteSparse Matrix Collection https://sparse.tamu.edu

Basic software

Structure \ Type	Dense	Sparse
Field	LAPACK	UMFPACK
Ring	4ti2	ParAd+4ti2
Monoid	4ti2	ParAd

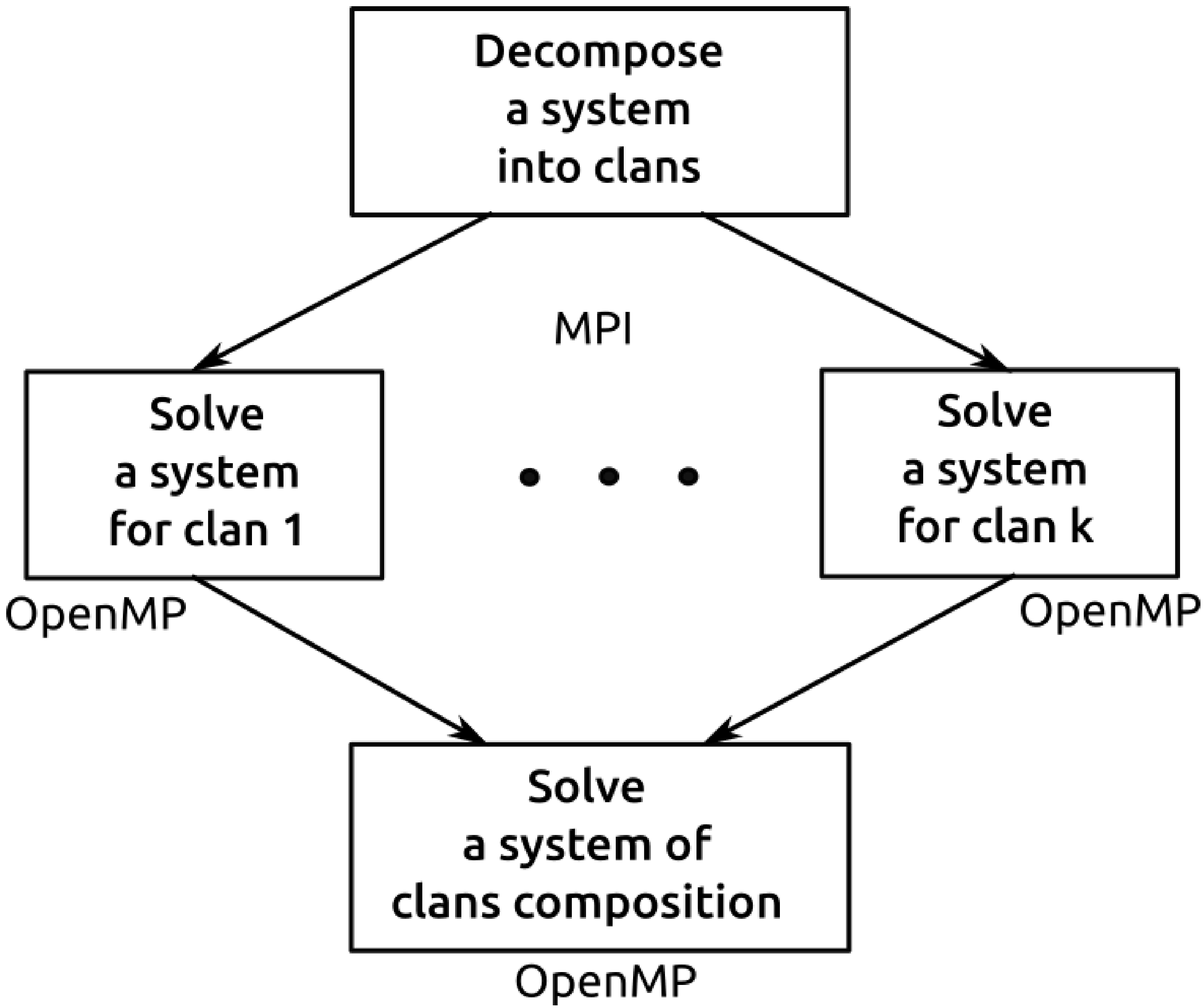
Software

- *Deborah* – decomposition into clans, 2004
- *Adriana* – solving a homogenous system via (a) simultaneous or (b) sequential composition of clans, 2005
- *ParAd* – solving a homogenous system via (a) simultaneous or (b) parallel-sequential composition of clans on *parallel architectures*, 2017

Zaitsev decomposition into clans

$$A = \begin{Vmatrix} A^{0,1} & \widehat{A}^{1} & 0 & 0 & 0 \\ A^{0,2} & 0 & \widehat{A}^{2} & 0 & 0 \\ \vdots & \vdots & \vdots & \vdots & \vdots \\ A^{0,k} & 0 & 0 & 0 & \widehat{A}^{k} \end{Vmatrix}$$

Divide and sway



A Clan – transitive closure of nearness relation

$$C1: \begin{cases} -x_2 + x_3 - x_{15} \underline{+ x_{18}} = 0 \\ -x_2 + x_4 - x_{14} \underline{+ x_{18}} = 0 \\ -x_5 + x_6 - x_{16} \underline{+ x_{18}} = 0 \\ -x_{11} + x_{12} - x_{15} \underline{+ x_{18}} = 0 \\ -x_{13} + x_8 \underline{+ x_{18}} = 0 \end{cases}$$

Two equations are ***near*** if they contain the same variable having coefficients of the same sign

Decomposition into clans

$$
\begin{cases}
-x_1 + x_2 - x_{18} = 0 \\
-x_2 + x_3 - x_{15} + x_{18} = 0 \\
-x_2 + x_4 - x_{14} + x_{18} = 0 \\
-x_4 + x_5 + x_{14} - x_{18} = 0 \\
-x_3 + x_5 + x_{15} - x_{18} = 0 \\
-x_5 + x_6 - x_{16} + x_{18} = 0 \\
-x_6 + x_7 + x_{16} - x_{19} = 0 \\
x_1 - x_7 + x_{19} = 0 \\
-x_8 + x_9 - x_{19} = 0 \\
-x_9 + x_{10} - x_{17} + x_{19} = 0 \\
-x_{10} + x_{11} + x_{17} - x_{18} = 0 \\
-x_{11} + x_{12} - x_{15} + x_{18} = 0 \\
-x_{12} + x_{13} + x_{15} - x_{18} = 0 \\
-x_{13} + x_8 + x_{18} = 0
\end{cases}
$$

$$
C1: \begin{cases}
-x_2 + x_3 - x_{15} + x_{18} = 0 \\
-x_2 + x_4 - x_{14} + x_{18} = 0 \\
-x_5 + x_6 - x_{16} + x_{18} = 0 \\
-x_{11} + x_{12} - x_{15} + x_{18} = 0 \\
-x_{13} + x_8 + x_{18} = 0
\end{cases}
$$

$$
C2: \begin{cases}
-x_1 + x_2 - x_{18} = 0 \\
-x_4 + x_5 + x_{14} - x_{18} = 0 \\
-x_3 + x_5 + x_{15} - x_{18} = 0 \\
-x_{10} + x_{11} + x_{17} - x_{18} = 0 \\
-x_{12} + x_{13} + x_{15} - x_{18} = 0
\end{cases}
$$

$$
C3: \begin{cases}
x_1 - x_7 + x_{19} = 0 \\
-x_9 + x_{10} - x_{17} + x_{19} = 0
\end{cases}
$$

$$
C4: \begin{cases}
-x_6 + x_7 + x_{16} - x_{19} = 0 \\
-x_8 + x_9 - x_{19} = 0
\end{cases}
$$

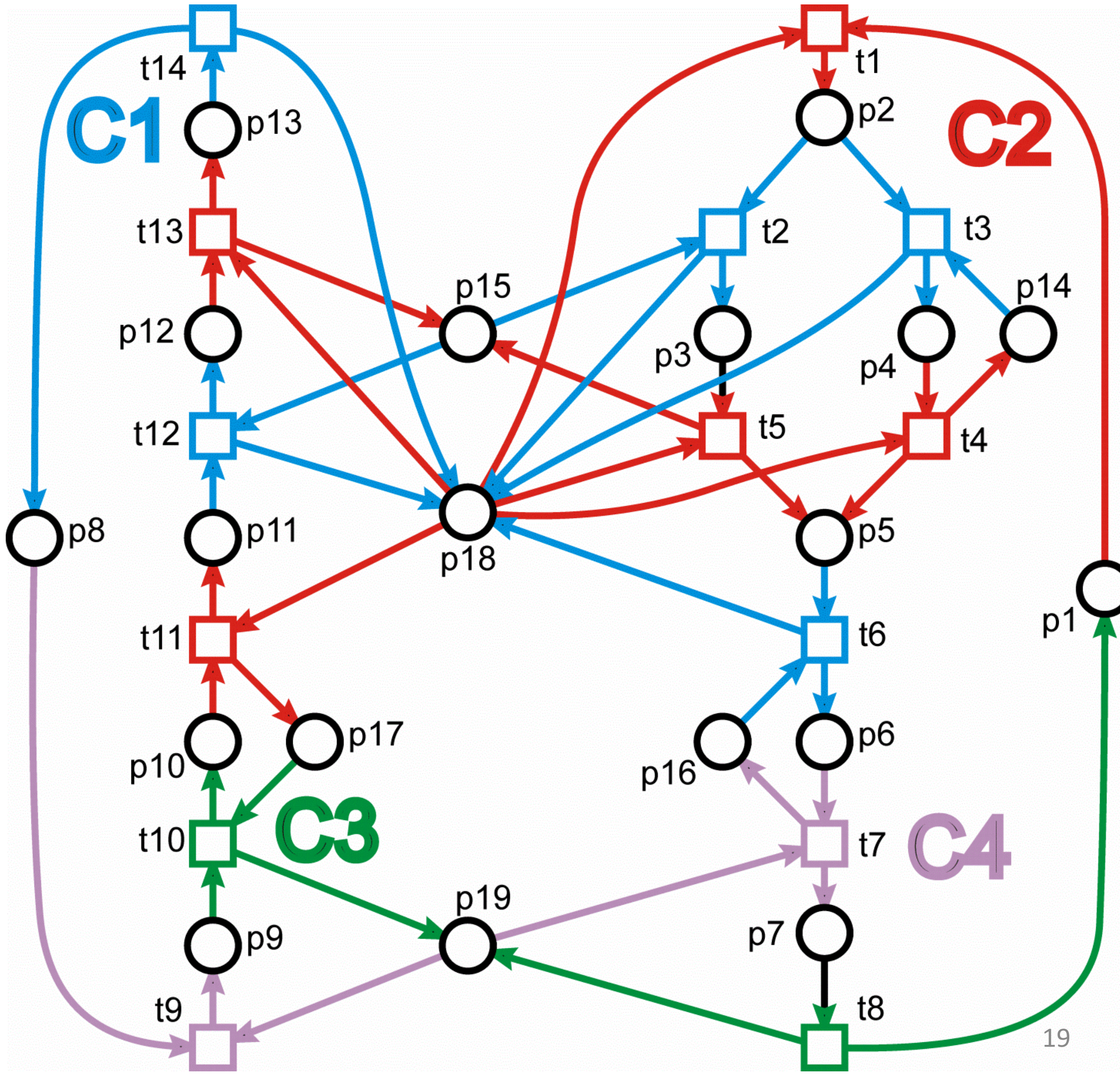
Systems and Directed bipartite graphs

Equation – transition (rectangle)

Variable – place (circle)

Positive sign – incoming arc of a place

Negative sign – outgoing arc of a place



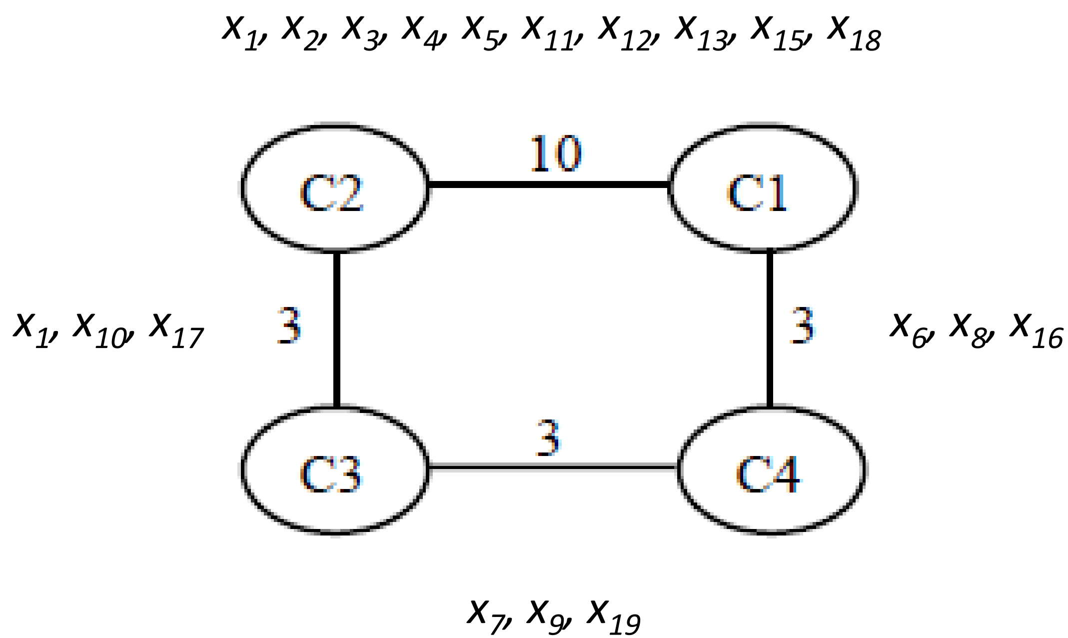
Decomposition graph

$$C1: \begin{cases} -x_2 + x_3 - x_{15} + x_{18} = 0 \\ -x_2 + x_4 - x_{14} + x_{18} = 0 \\ -x_5 + x_6 - x_{16} + x_{18} = 0 \\ -x_{11} + x_{12} - x_{15} + x_{18} = 0 \\ -x_{13} + x_8 + x_{18} = 0 \end{cases}$$

$$C2: \begin{cases} -x_1 + x_2 - x_{18} = 0 \\ -x_4 + x_5 + x_{14} - x_{18} = 0 \\ -x_3 + x_5 + x_{15} - x_{18} = 0 \\ -x_{10} + x_{11} + x_{17} - x_{18} = 0 \\ -x_{12} + x_{13} + x_{15} - x_{18} = 0 \end{cases}$$

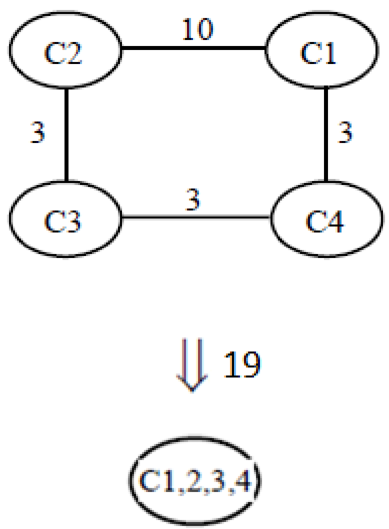
$$C3: \begin{cases} x_1 - x_7 + x_{19} = 0 \\ -x_9 + x_{10} - x_{17} + x_{19} = 0 \end{cases}$$

$$C4: \begin{cases} -x_6 + x_7 + x_{16} - x_{19} = 0 \\ -x_8 + x_9 - x_{19} = 0 \end{cases}$$

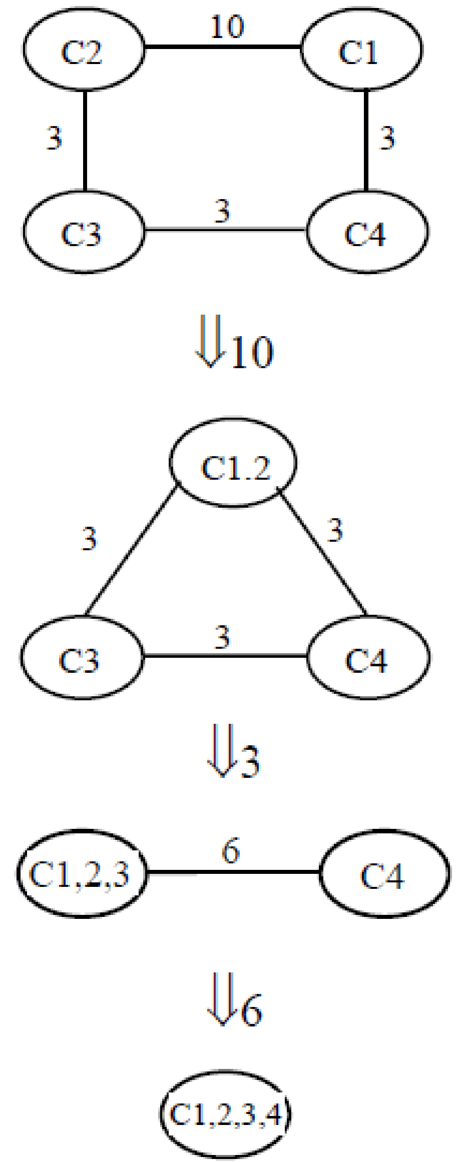


Collapse of decomposition graph

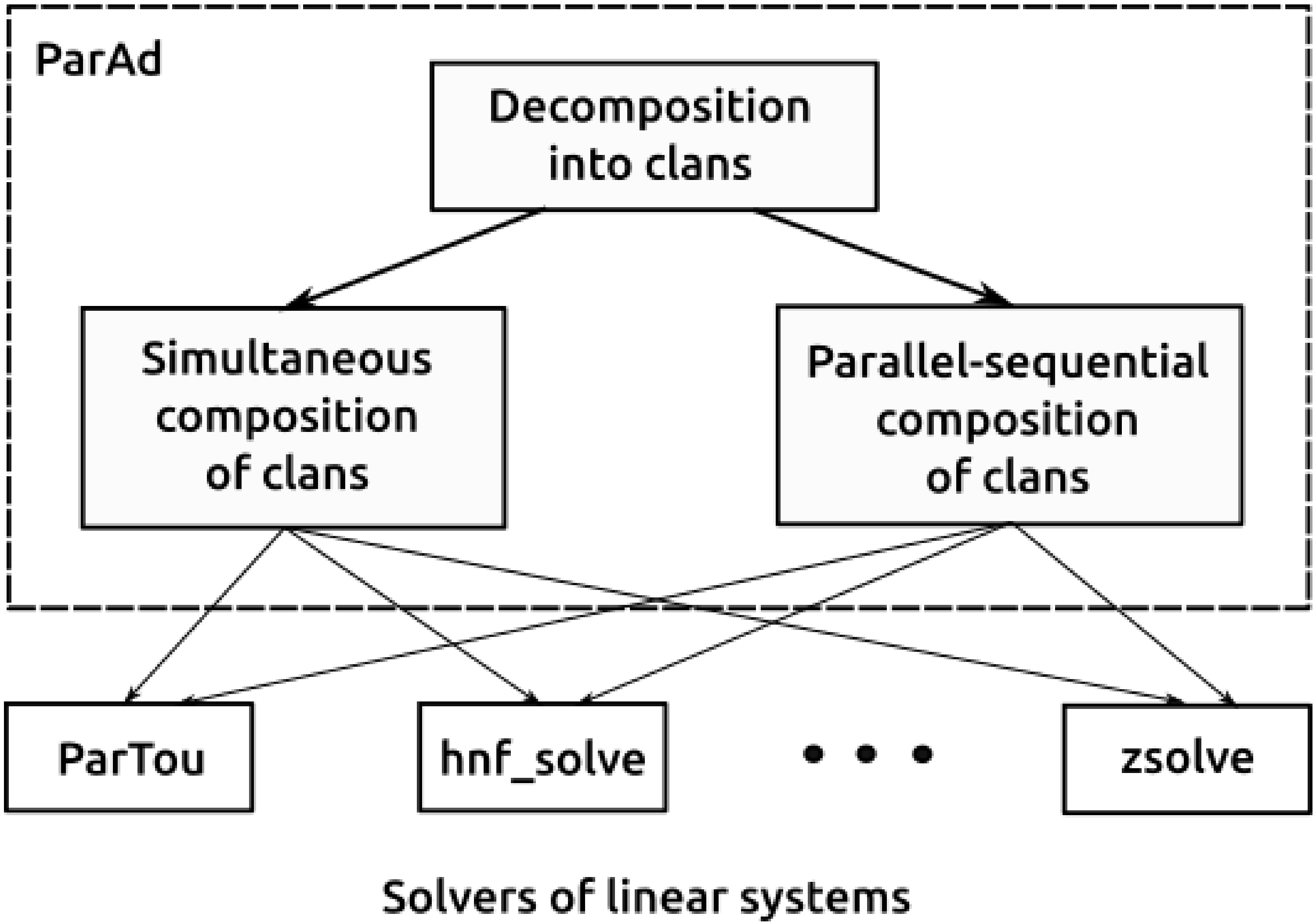
I.



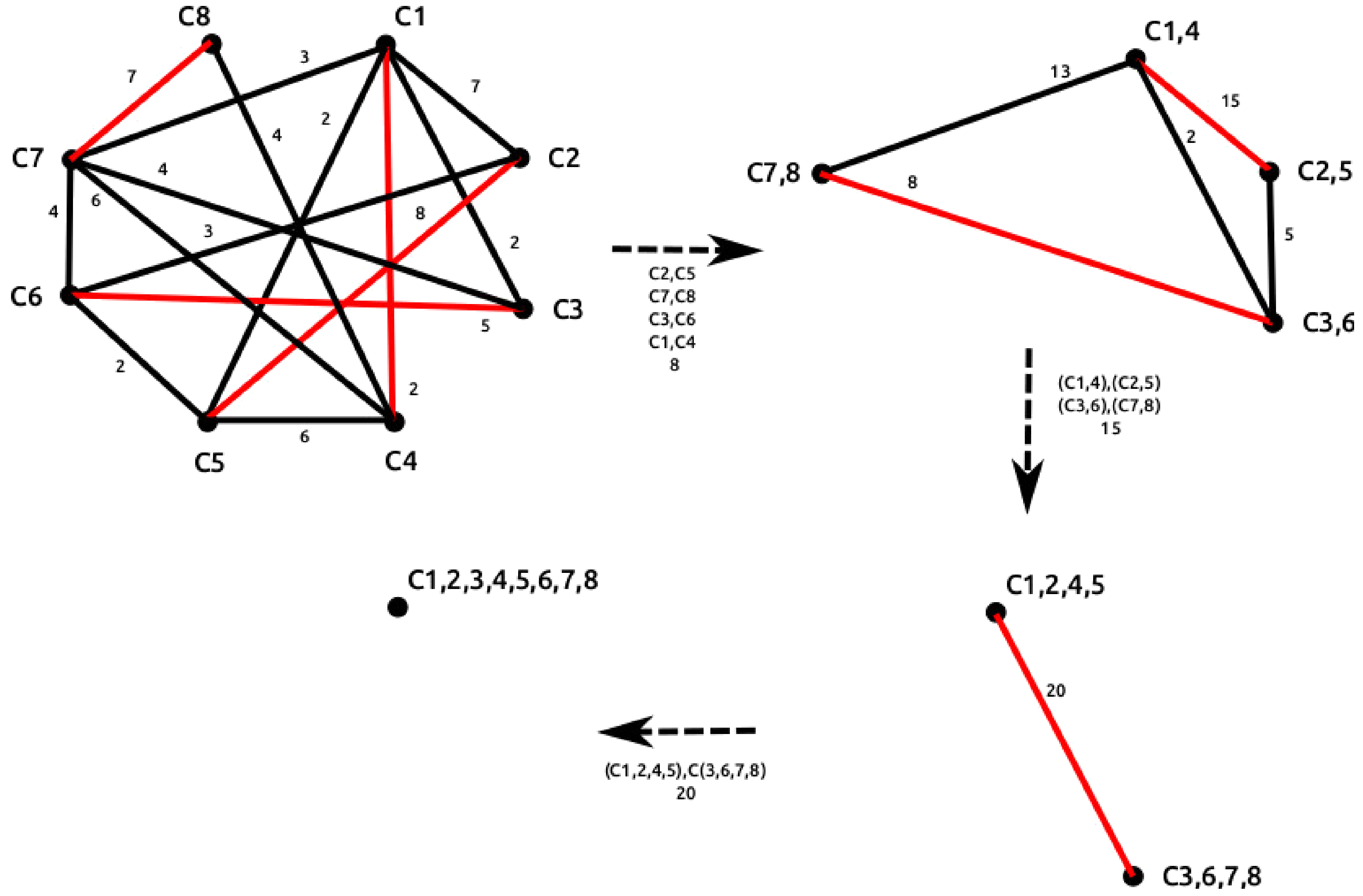
II.



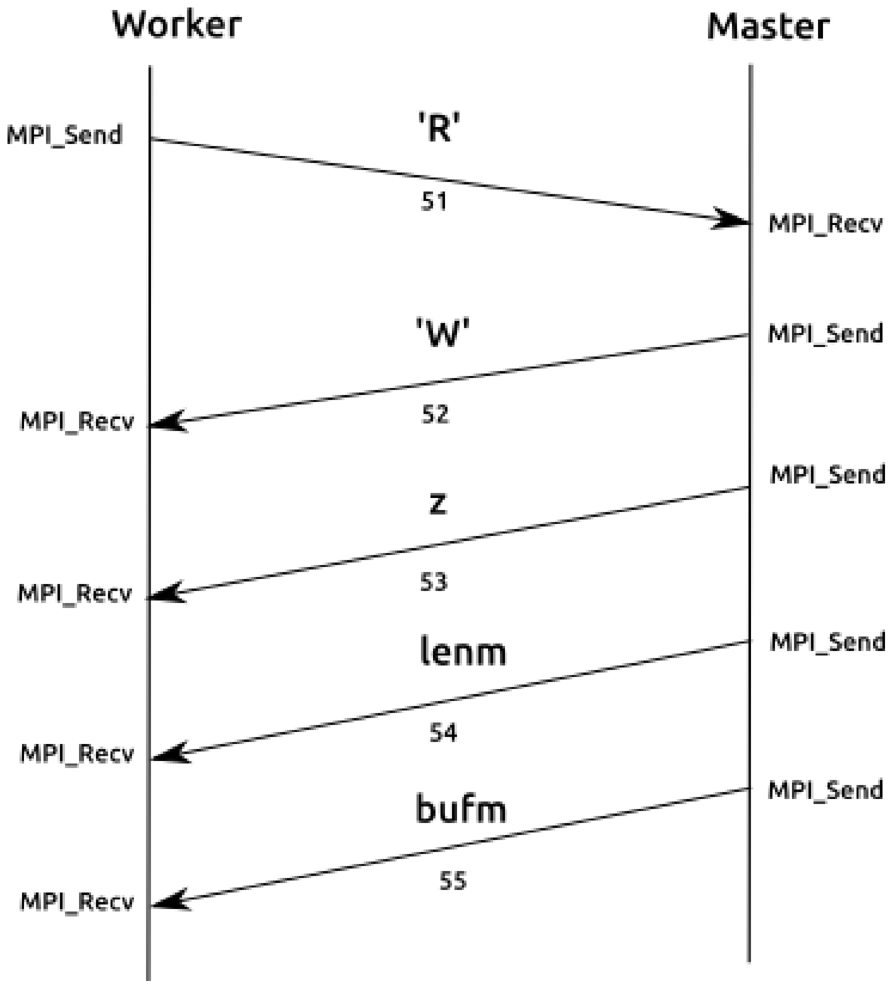
ParAd – Parallel Adriana

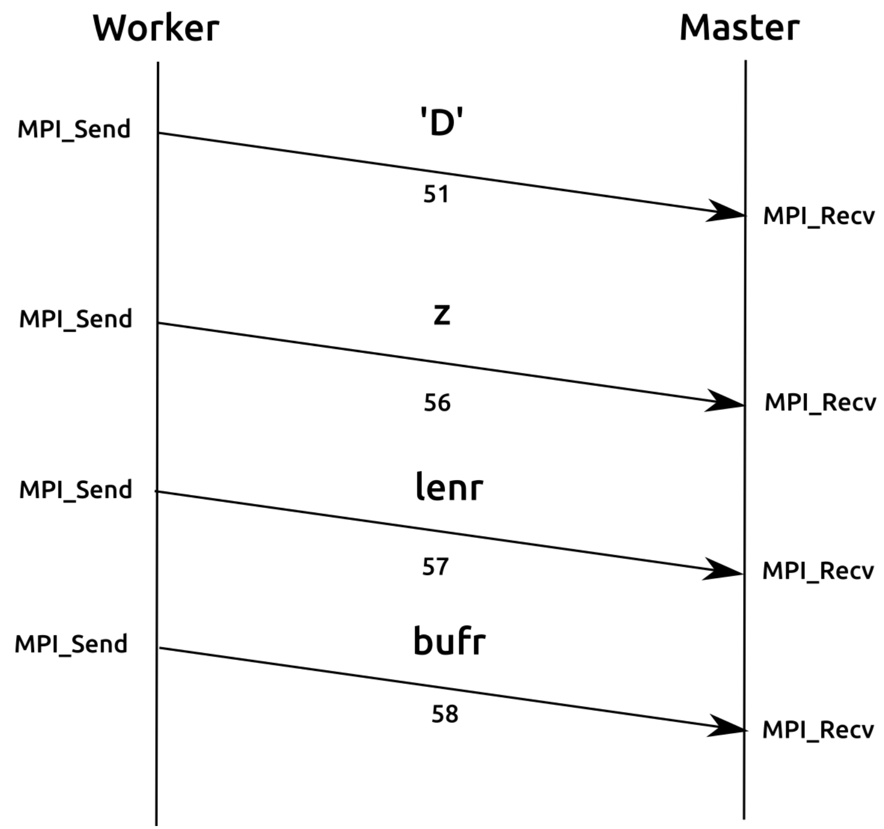


Parallel-sequential Composition of Clans

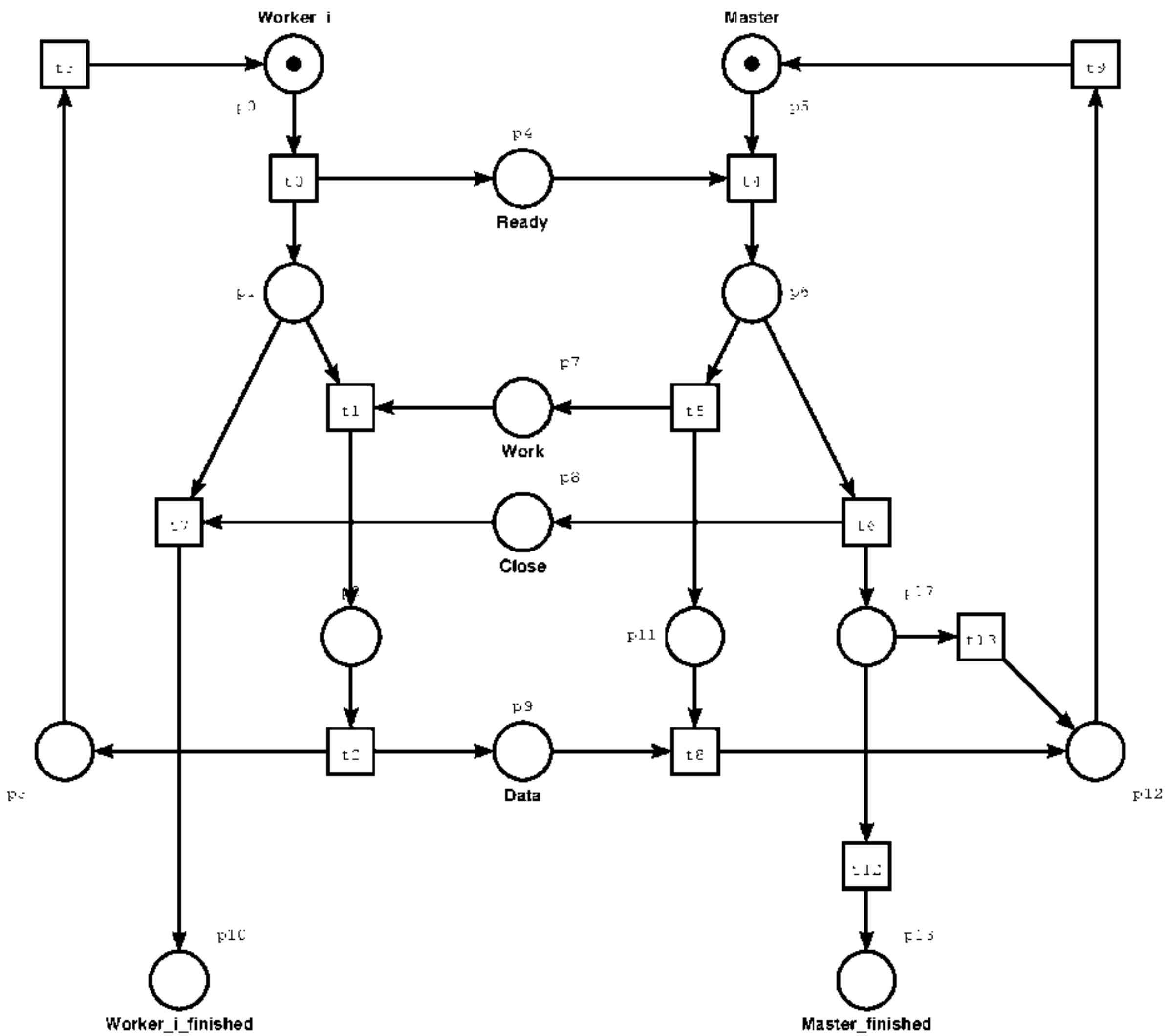


Protocols of data transmission

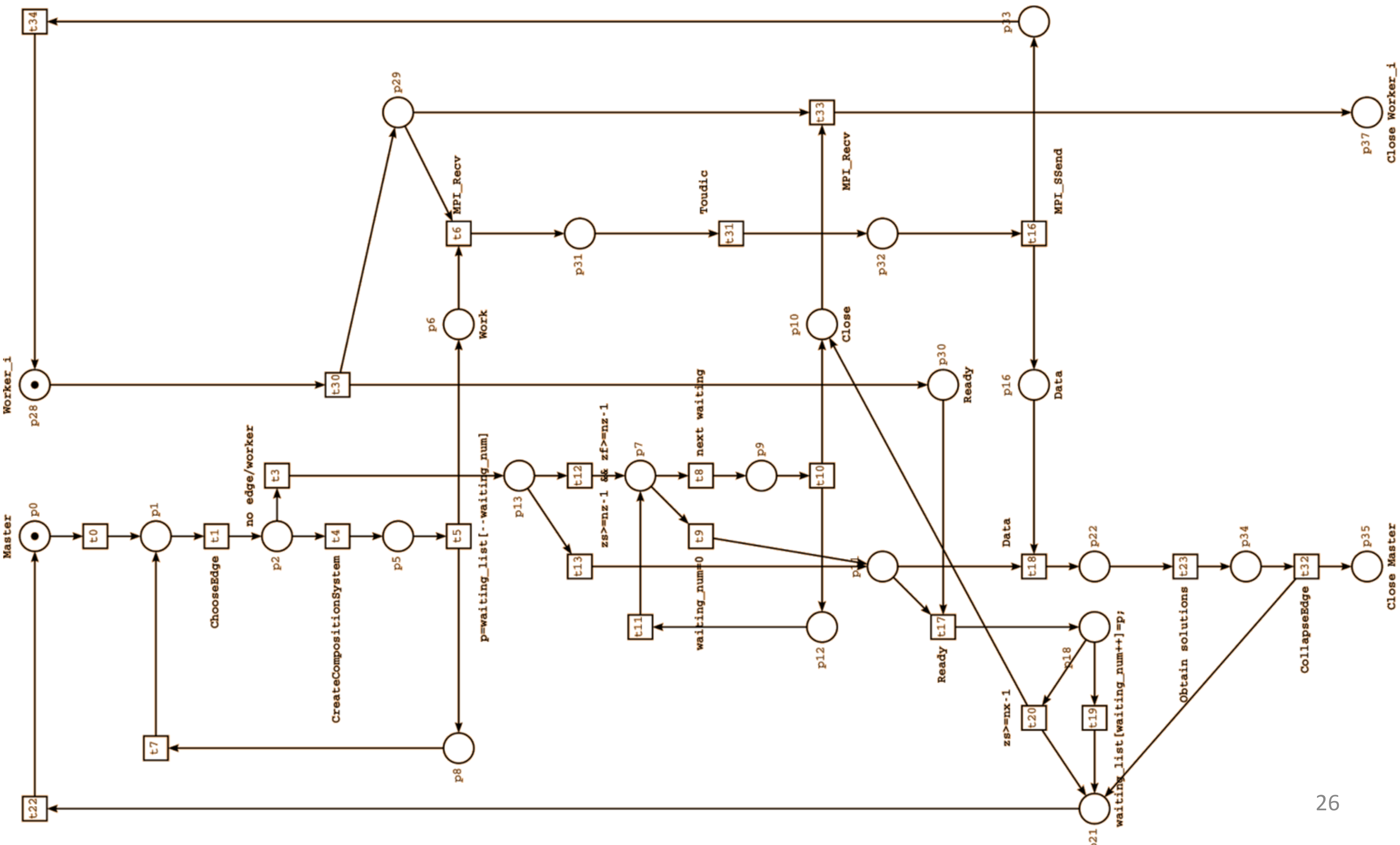




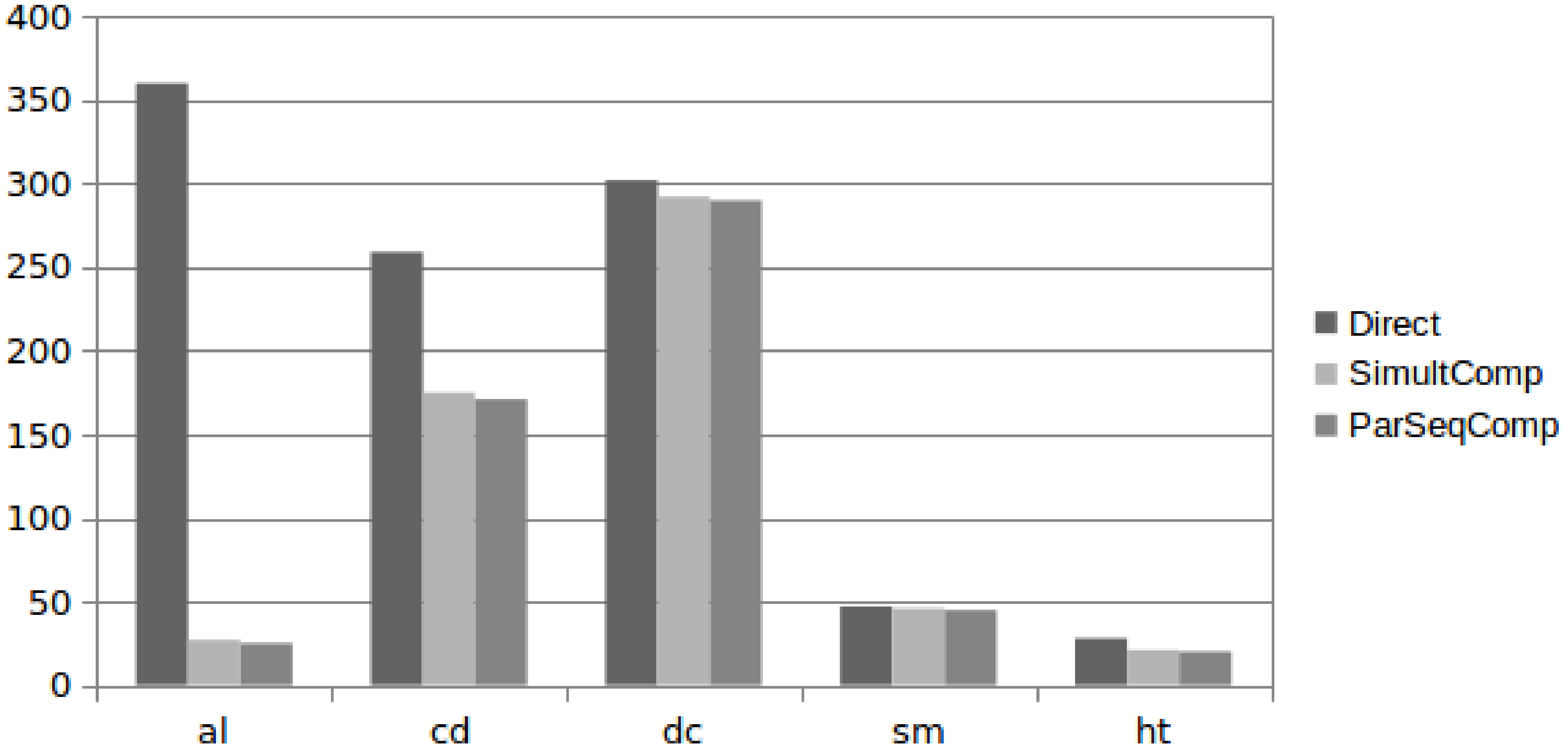
Master-Worker Basic Communication Protocol



Parallel-Sequential Composition Communication Protocol



Speed-up because of clans



Run ParAd

- **Run with mpirun**

```
>mpirun -n 5 ./ParAd -c -r zsolve tcp.spm tcp-pi.spm
>mpirun -n 10 ./ParAd -s -t -d 1 tcp.spm tcp-ti.spm
```

- **Run with Slurm**

```
>srun -N 10 ./ParAd -s -t -d 1 tcp.spm tcp-ti.spm
```

- **SPM – simple sparse matrix format:**

```
i j a[i][j]
```

- **Check decomposability (Matrix Market Format)**

```
>toclans lp_cre_d.mtx
```

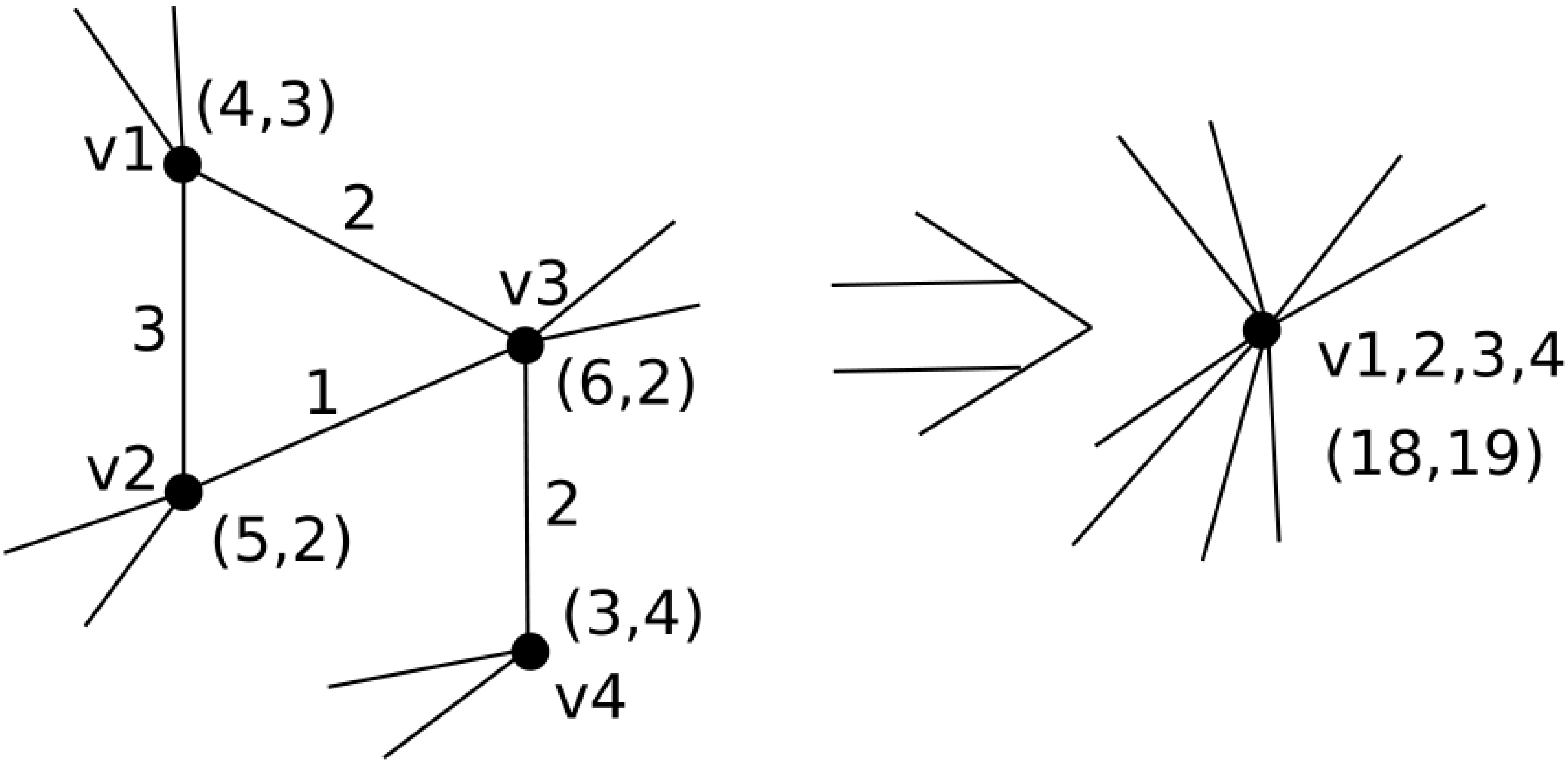
Load balancing

- ***Dynamic*** **on demand – appoint a clan to a free node (version 1.1)**
- ***Static*** **– aggregate minimal clans into big clans according to the number of available computing nodes**
- ***Hybrid*** **– pre-aggregation to equal size and then dynamic scheduling clans to nodes (version 1.2)**

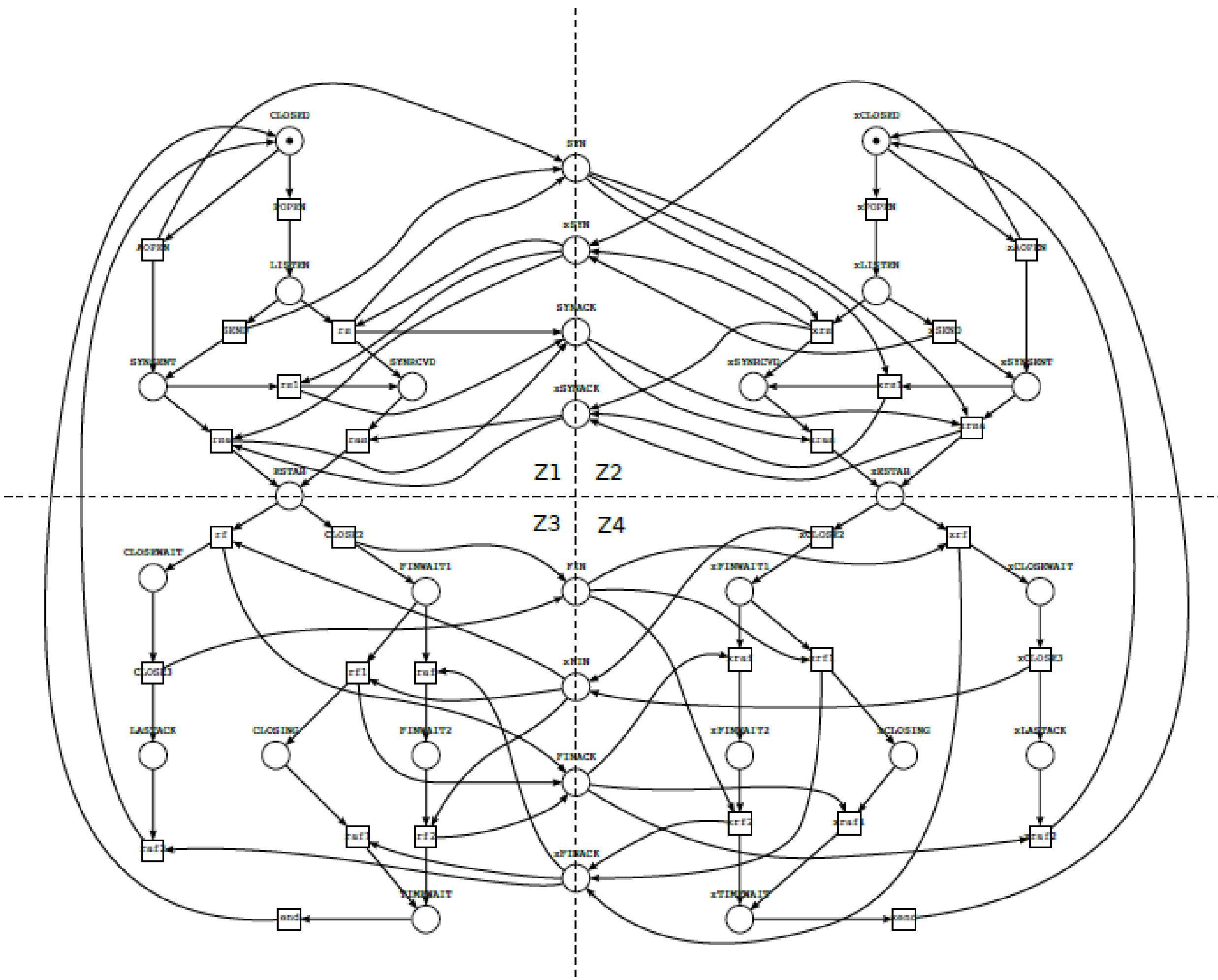
Aggregation of Clans for Workload Balancing

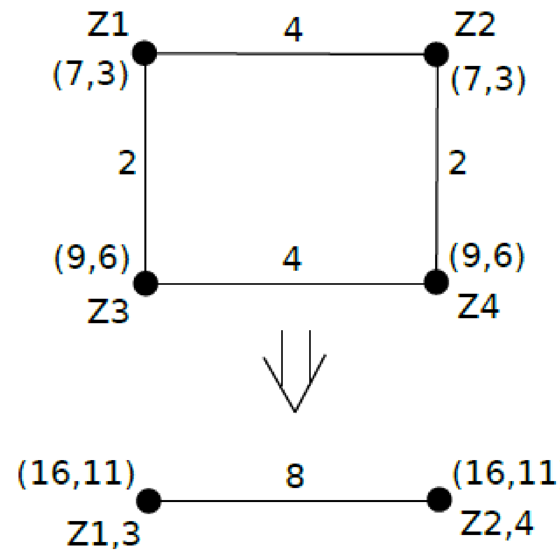
- A clan is a sum of minimal clans
- The maximal clan size restricts granulation
- Many small clans lead to heavy communication load
- Balancing: create clans having size close to the maximal
- Key: -a val
- Aggregation steeds-up about 20%

Aggregation idea

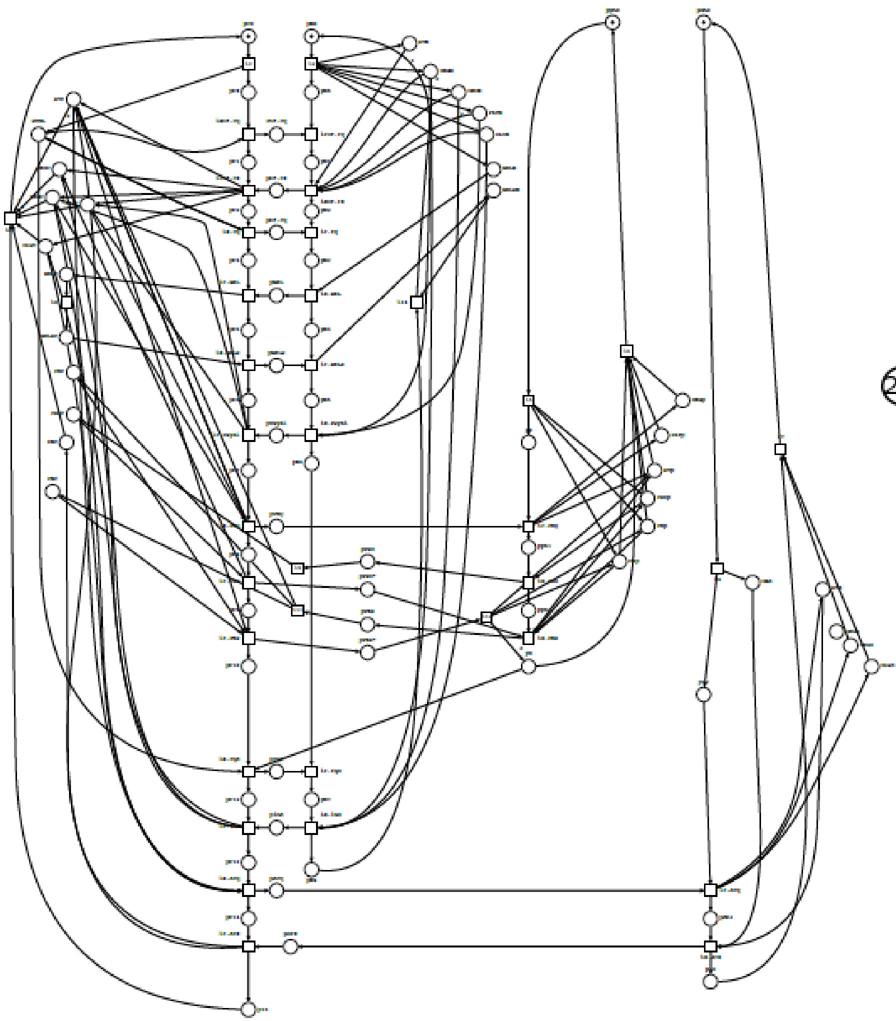


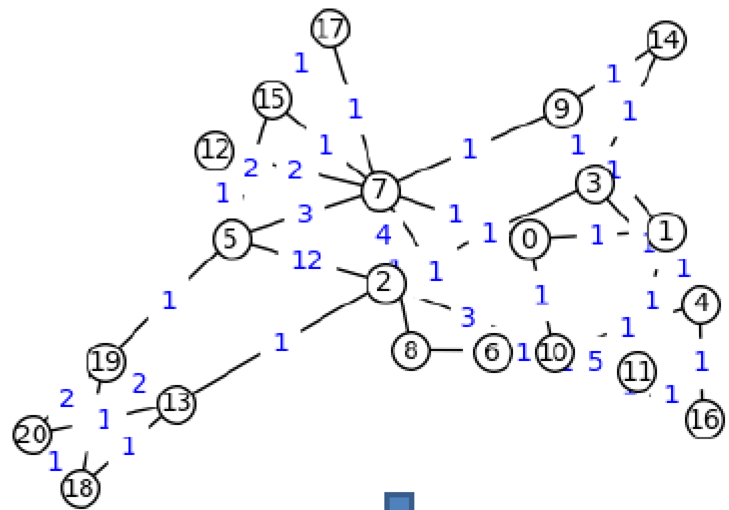
Verification of TCP



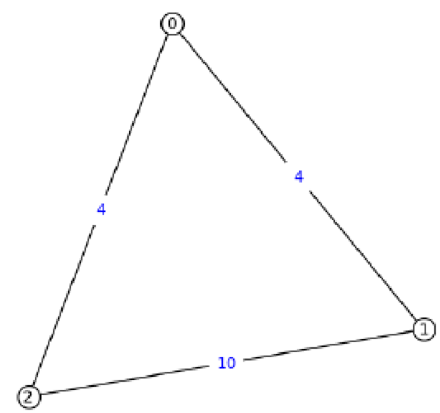


Verification of IOTP



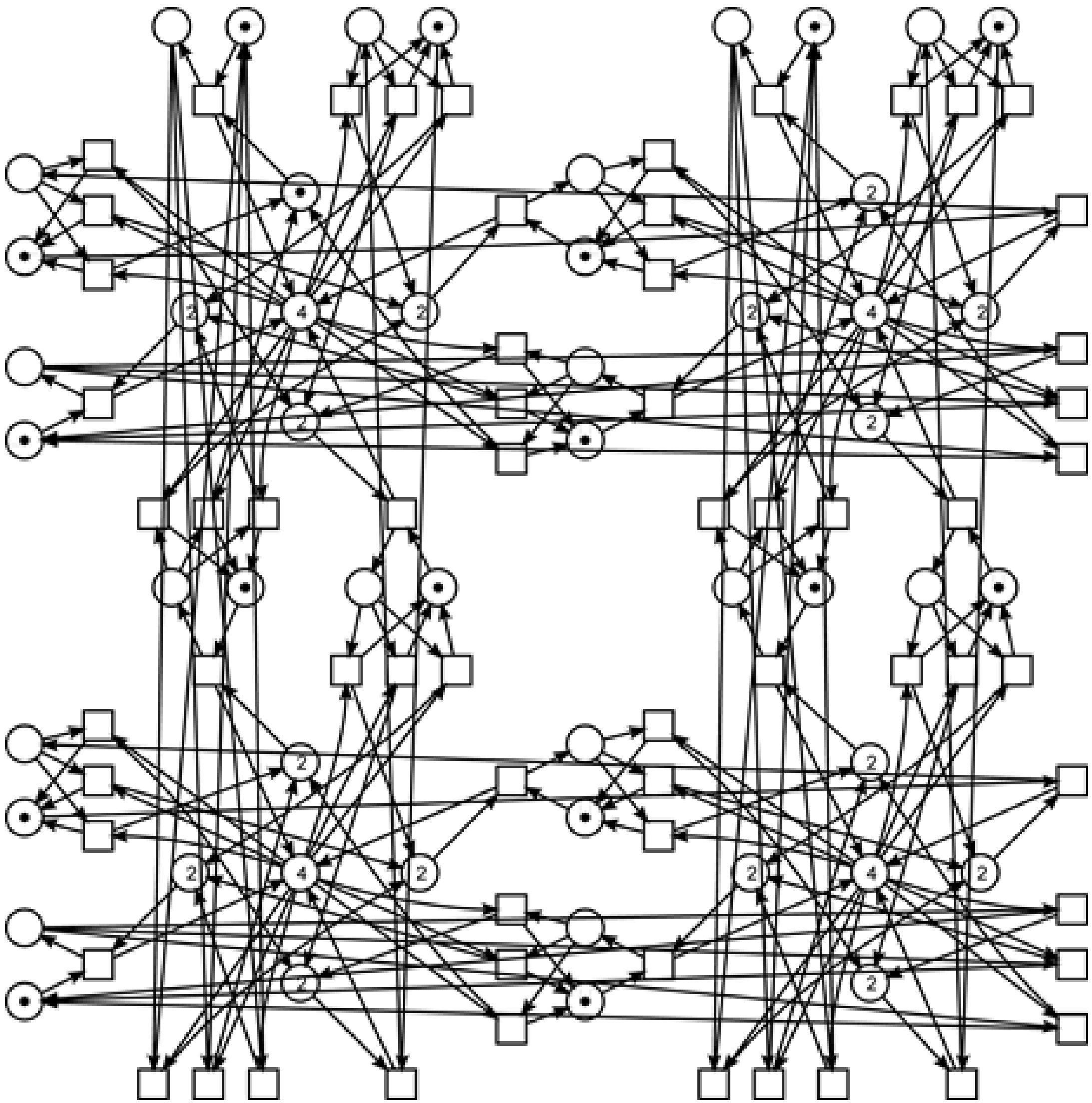


Aggregation

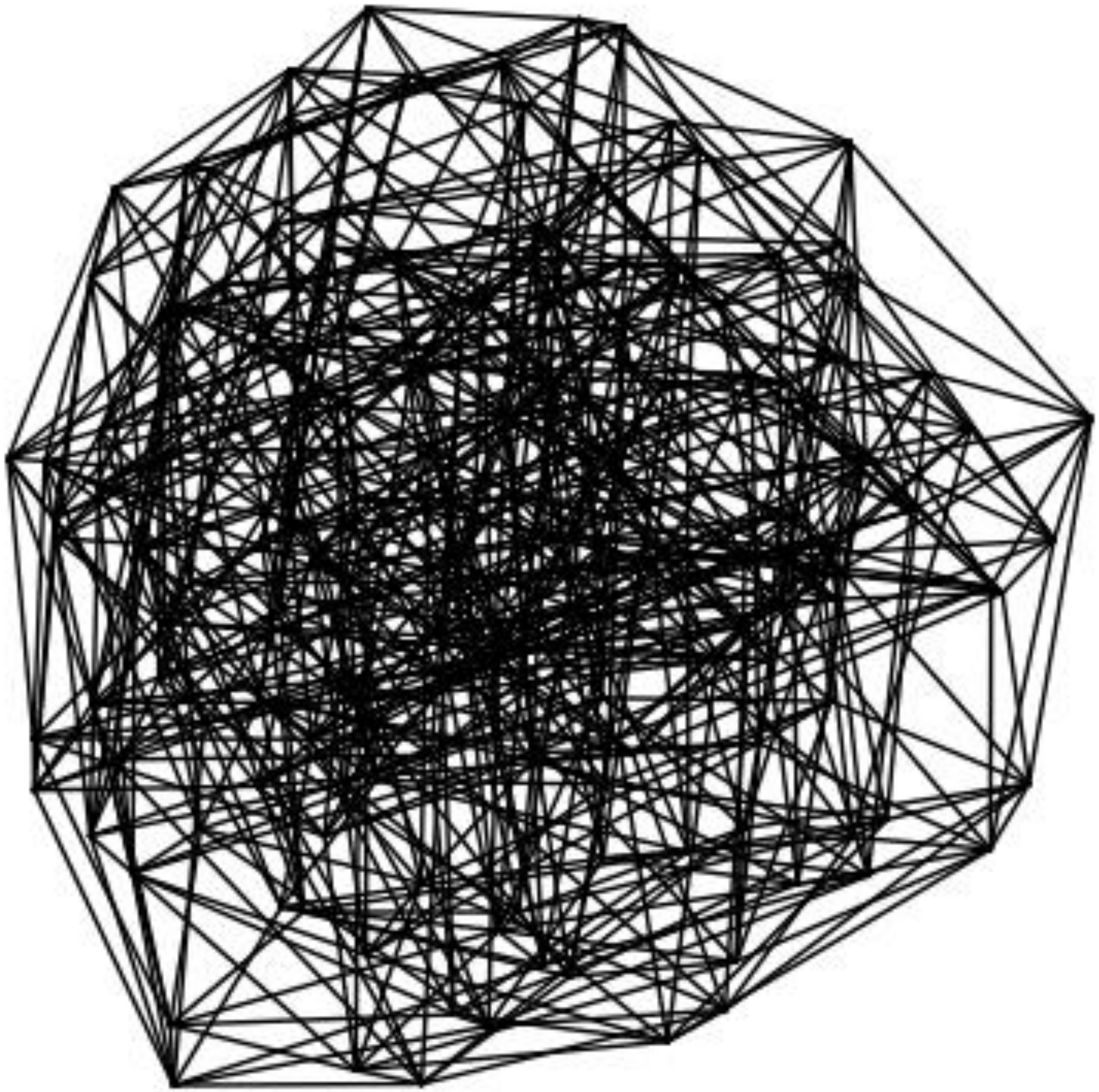


edge-cut 18, partition weight 13

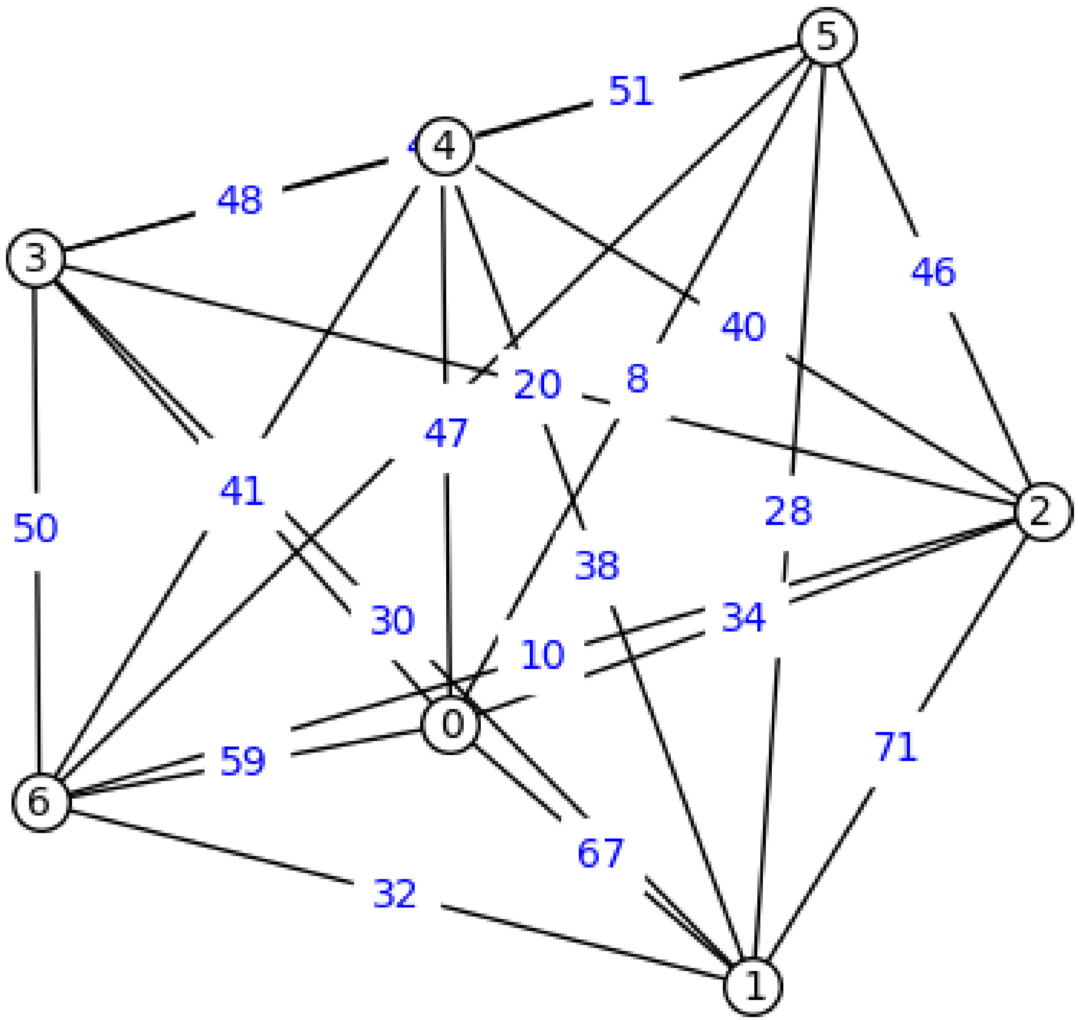
Hypertorus switching grid, 2D, 2x2 model



Decomposition of hypertorus 4D, 3x3

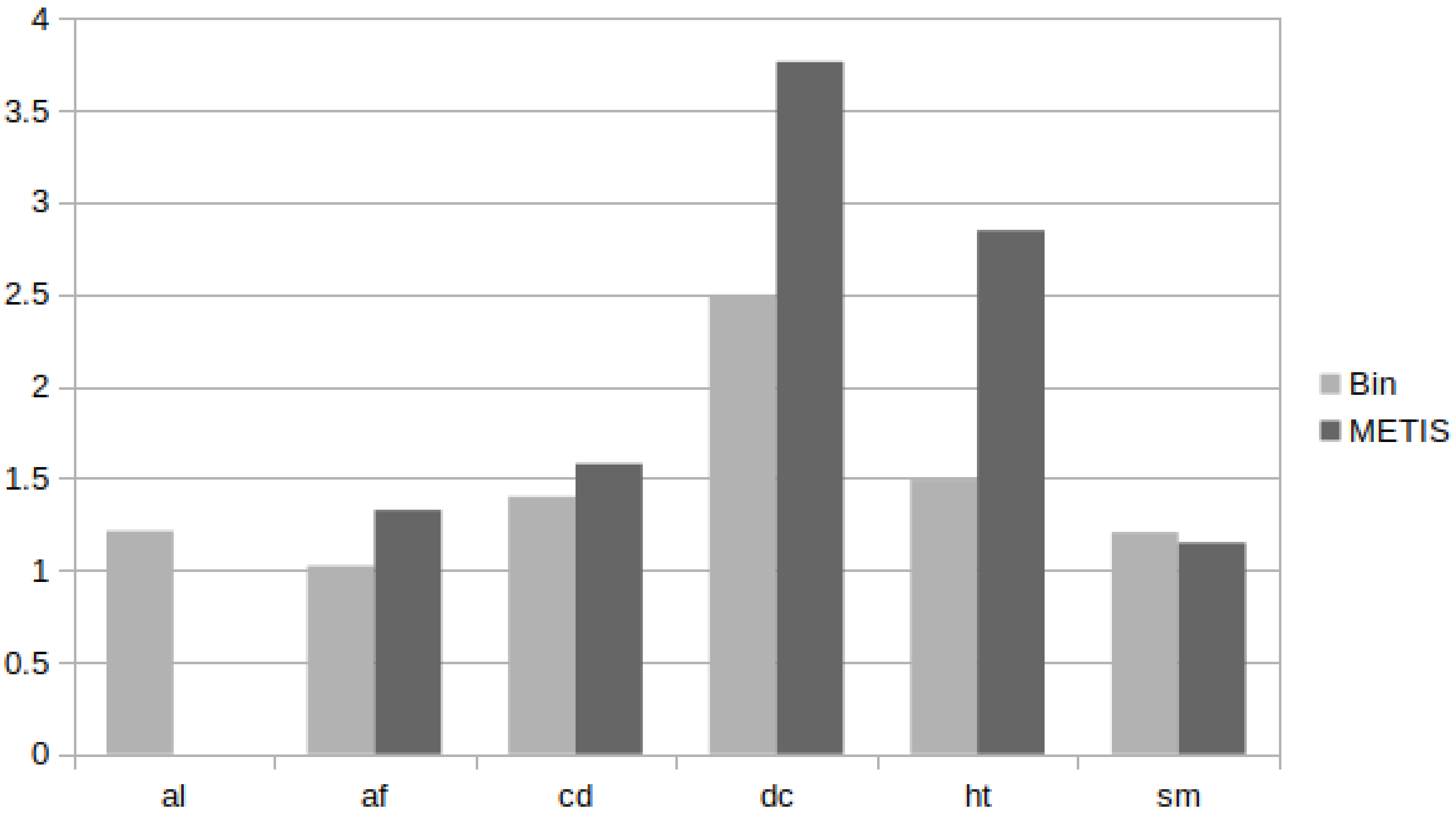


Aggregation by METIS into 7 clans



7 partitions: edge-cut 855, maximal partition weight 760

Extra speed-up because of aggregation



Conclusions

- **Additional modules for ParAd:**

a) **clans aggregation with METIS;**

b) **clans aggregation with bin packing**

- **Tested with verification of TCP, IOTP, and hypertorus grid**
- **Extra speed-up because of aggregation up to 4 times**

http://daze.ho.ua

Recent references

- ParAd, https://github.com/dazeorgacm/ParAd
- Dmitry Zaitsev, Stanimire Tomov, Jack Dongarra. Solving Linear Diophantine Systems on Parallel Architectures, IEEE Transactions on Parallel and Distributed Systems, 30(5), 2019.
- Zaitsev D.A. Sequential composition of linear systems’ clans, Information Sciences, Vol. 363, 2016, 292–307.
- D. A. Zaitsev, Clans of Petri Nets: Verification of protocols and performance evaluation of networks. LAP LAMBERT Academic Publishing, 2013.

http://daze.ho.ua