Trace Overview based on Spatio-Temporal Aggregation

Damien Dosimont [1 4] Lucas M. Schnorr [2 5]
Guillaume Huard [3 4] Jean-Marc Vincent [3 4]

[1] INRIA

[2] UFRGS

[3] UJF

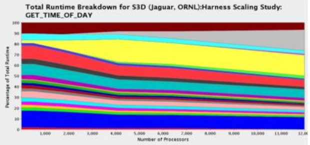
[4] *firstname.lastname@imag.fr*

[5] *schnorr@inf.ufrgs.br*

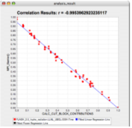
SONGS T+24 plenary meeting, January 27, 2014

Current visualization techniques bring information about system behavior

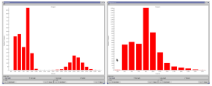
Global Analysis



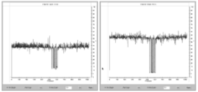
Execution Comparison



Correlations

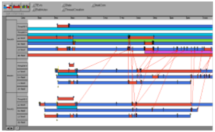


Outliers

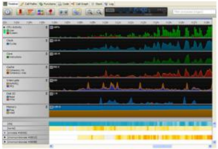


Workload

Run behavior

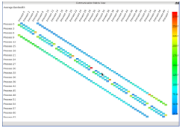


Causality relations

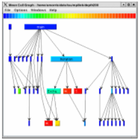


Resource usage

Structure



Communications



Call graphs

Time and space (resources) analysis scalability?

Ex: Gantt Chart is the most common technique employed by analysts...

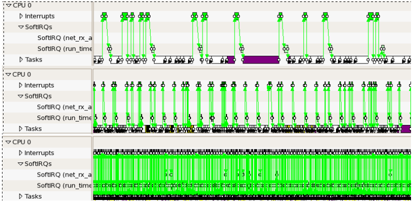


Figure 1 : *KPTrace dezoom : example of time axis scalability issues*

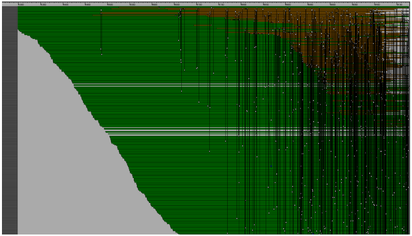


Figure 2 : *Example of space limitations : Pajé trace with 700 producers*

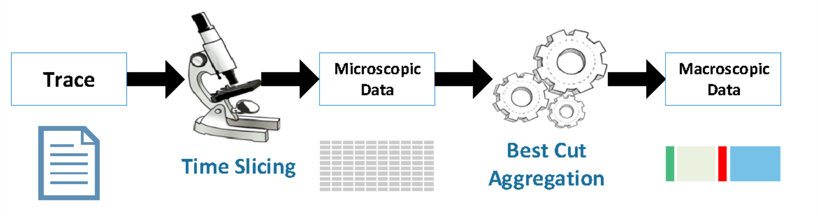
... but it does not scale to voluminous traces

Our proposal: Ocelotl

Fit to Schneiderman's methodology...

- **Overview** first, zoom and filter, then details on demand

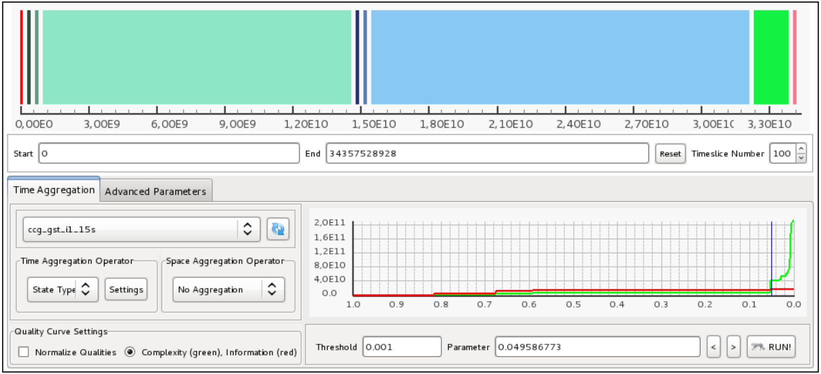
... by providing a macroscopic description of the trace...



... build upon an algorithm proposed by Lamarche-Perrin

- Adapted to timestamped events using **time slicing**
- Extended to **multiple event sources**

Our proposal: Ocelotl



Find a perturbation by using several level of details

a) p=1

b) p=0.4

c) p=0.049

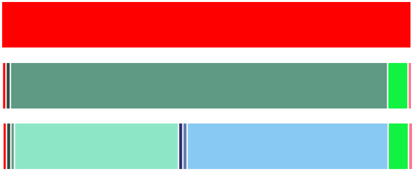


Figure 3 : *G-Streamer application perturbed execution: a) full aggregation, b) initialization and termination shown, c) perturbation detected*

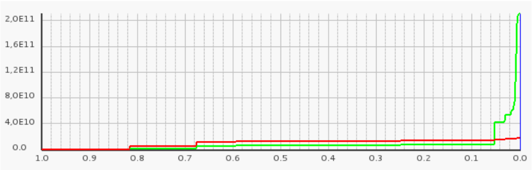


Figure 4 : *Information (red) and complexity (green) provided by aggregations*

Add semantic to understand general behavior

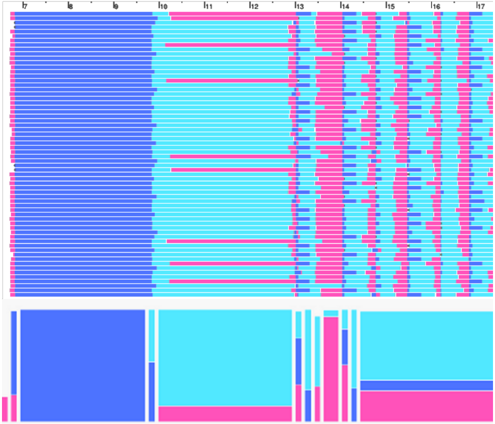


Figure 5 : *NAS Benchmark CG.A.64*

Compare several executions

a) reference

b) with perturbation

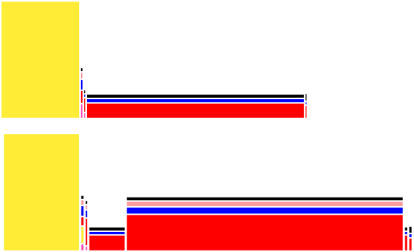


Figure 6 : *NAS Benchmark LU.A.32*

Some numbers...

G-Streamer case : 30 s

- Almost **1500** different functions, **4** threads
- **One million** of events
- **100 MB** trace (Pajé format)
- **15 seconds** to query events and pre-treatment
- Interaction is then instantaneous

Main limitations

- $<$ **10000** resources.
- $<$ **4 GB** to keep reasonable event query delay
- Efficient to decompose trace behavior in time, but unable to relate it with resources

Background: macroscopic description of a system over its structure

Lamarche-Perrin and Schnorr works

- Aggregate preferentially nodes that have close values
- Parametrized by the user to find a good compromise

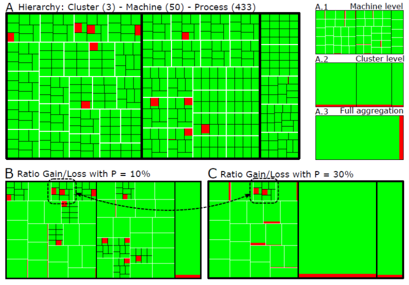


Figure 7 : *Triva treemap view example, showing influence of parameter p on node aggregation*

Extension of these works

Spatial AND temporal simultaneous aggregation

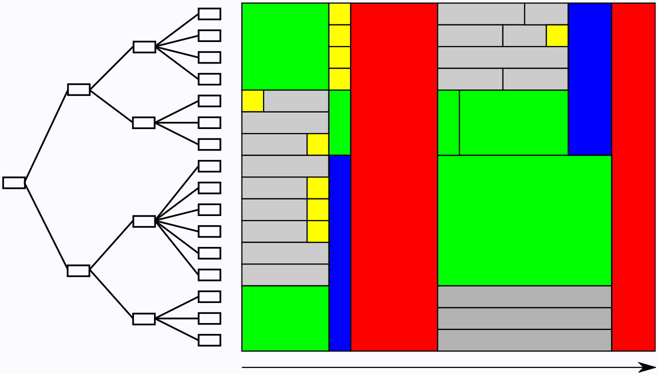


Figure 8 : *Synthetic example of spatio-temporal aggregation where space is a hierarchy and time cut into time slices*

Conclusion

Tools and FrameSoC Framework

- Official release **in June**
- Compatible with Pajé trace files, and thus OTF/Tau by using Schnorr's converters

Find use cases and analyze MPI states

- Applications that are not easy to analyze with traditional tools because of resource size
- Qualitative comparison of different executions (ex: simulation vs real application)
- Evaluate complex application/system both space and time behavior.

Links

My website

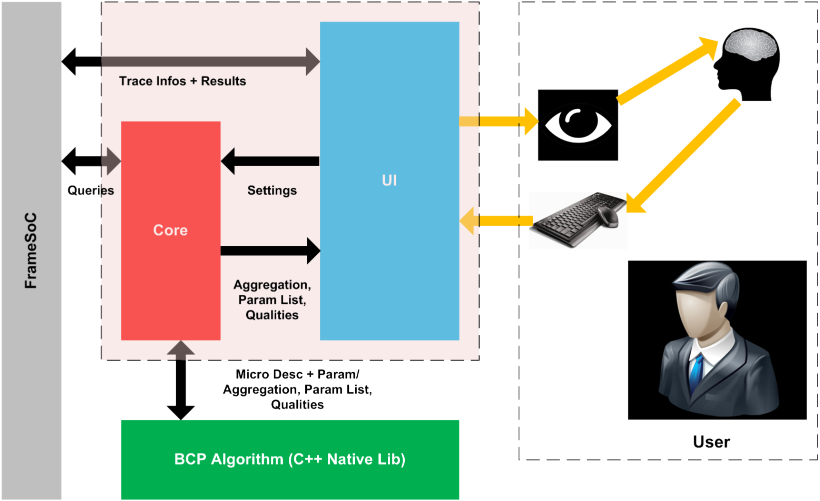
http://moais.imag.fr/membres/damien.dosimont/

Tools and libraries are available on my github

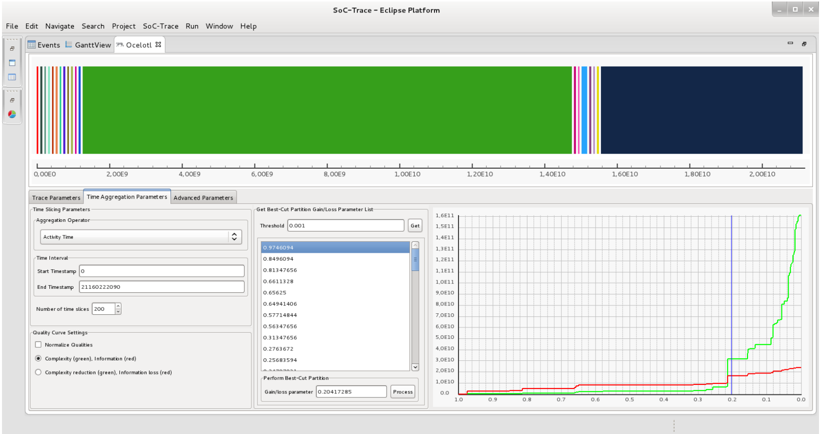
http://github.com/dosimont

Merci pour votre attention!

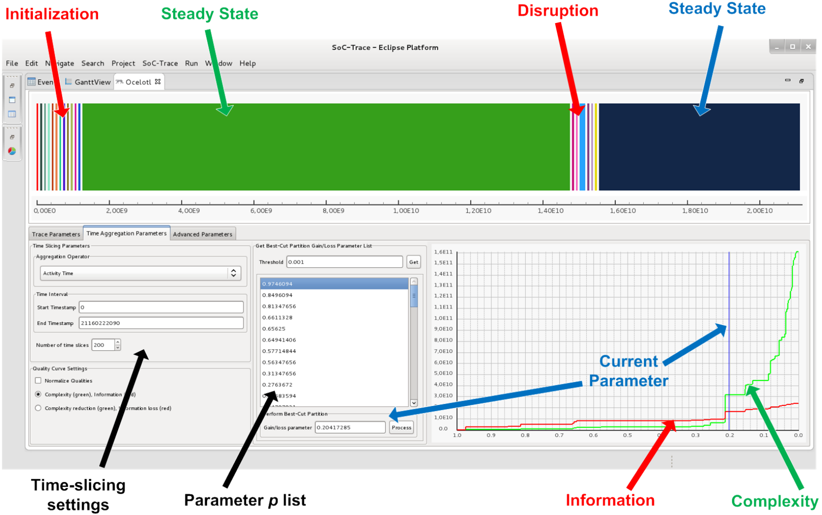
Implementation



Interface Overview

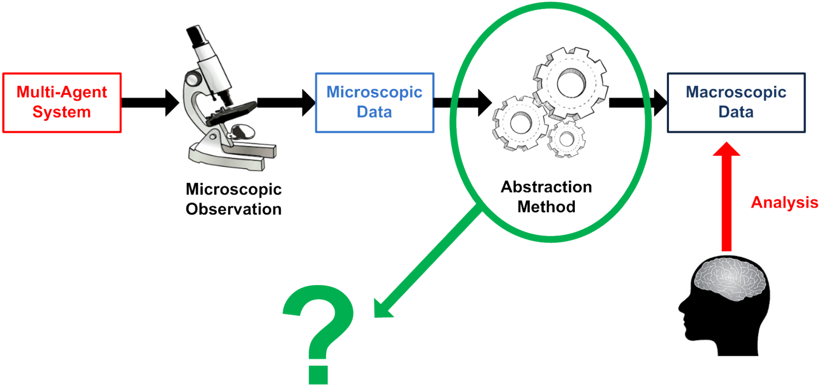


Interface Overview

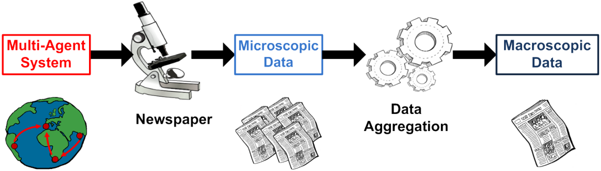


Lamarche-Perrin Works: Multi-Agent Systems

How to Build a Meaningful Macroscopic Description?



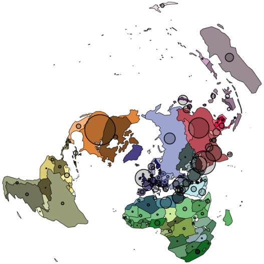
Example: Geomedia Project



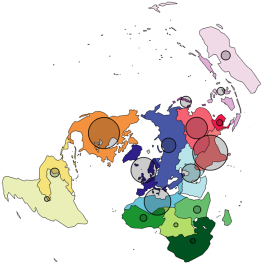
Information Loss

Complexity Reduction

Macroscopic Semantics



Resolution: Max

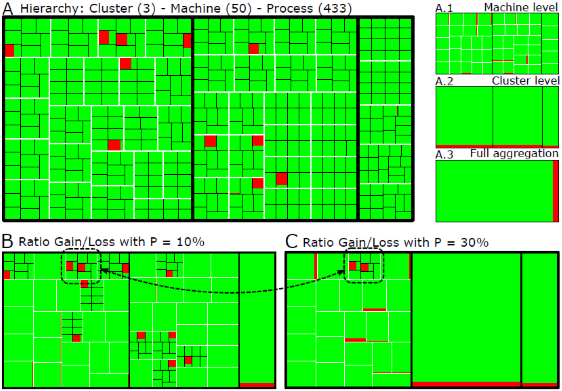


Resolution: Intermediary

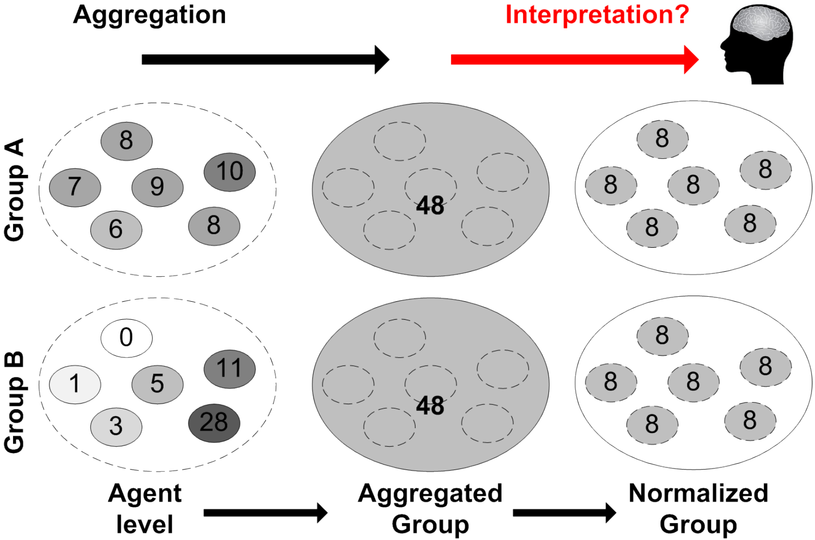
Resolution: Min

Example: Viva

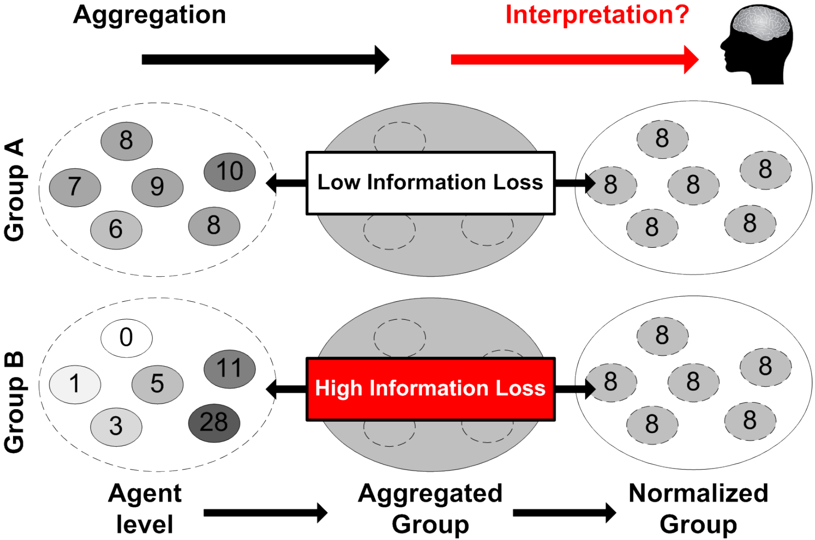
Represent Hierarchical Structure according to Value Heterogeneity



Information Loss



Information Loss



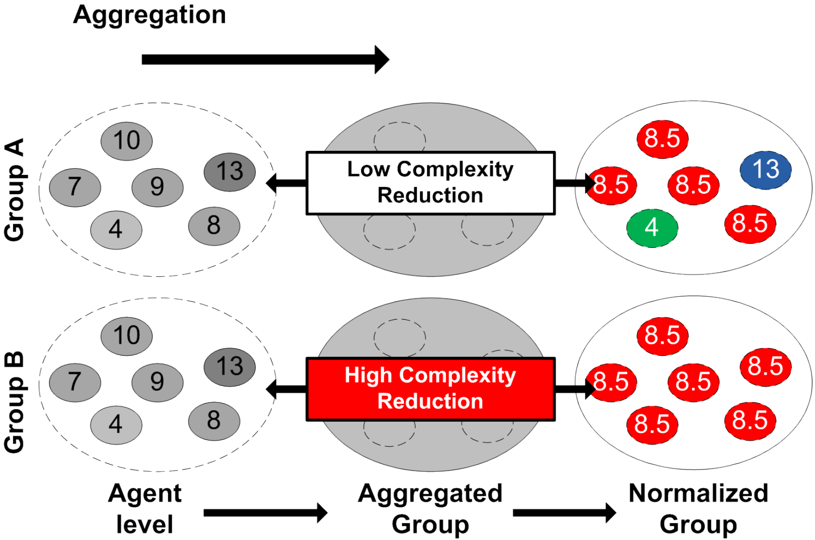
Information Loss Measure

Kullback-Leibler Divergence

$\text{loss}(A||e) = \sum_{e \in A} v(e) \times \log_2 \left(\frac{v(e)}{v(A)} \right)$ in bits/x

- Quantity of information than one **loses** by using an **aggregated description** instead of the **microscopic description**

Complexity Reduction



Complexity Reduction Measure

Shannon Entropy

$\mathrm{H}(v) = \sum (v(i) \times \log_2 v(i))$ in bits/x

Entropy Reduction

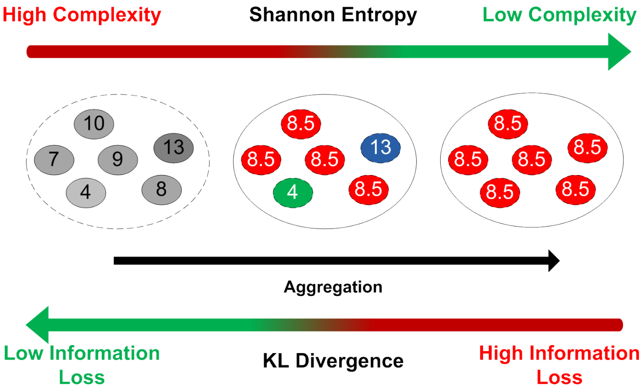
$\mathrm{gain}(A||e) = \mathrm{H}(A) - \mathrm{H}(e)$ in bits/x

- Quantity of information than one **saves** by encoding the **aggregated description** instead of the **microscopic description**

Compromise Finding between Information Loss and Complexity Reduction

Parametrized Information Criterion

$$\text{pIC}(\mathcal{A}) = p \times \text{gain}(\mathcal{A}) - (1 - p) \times \text{loss}(\mathcal{A})$$

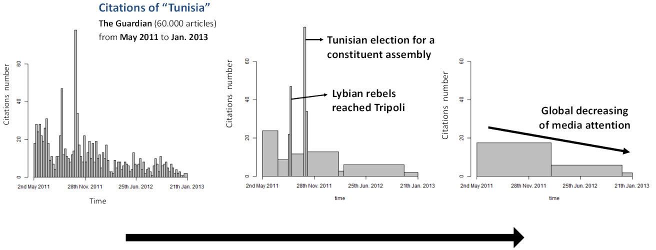


Temporal Aggregation

Temporal Aggregation principle

- Same principle but only consecutive data can be aggregated

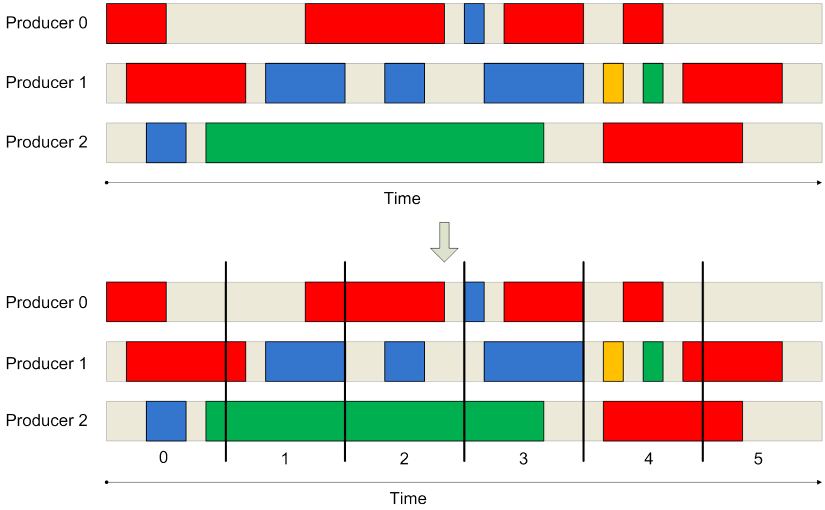
Ex: Tunisia citation



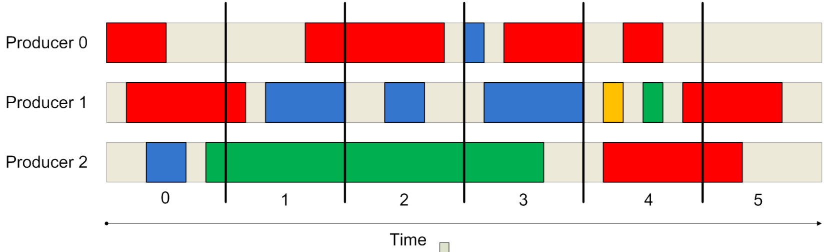
p is growing

Need of a microscopic level description

Microscopic Level: Time-Slicing



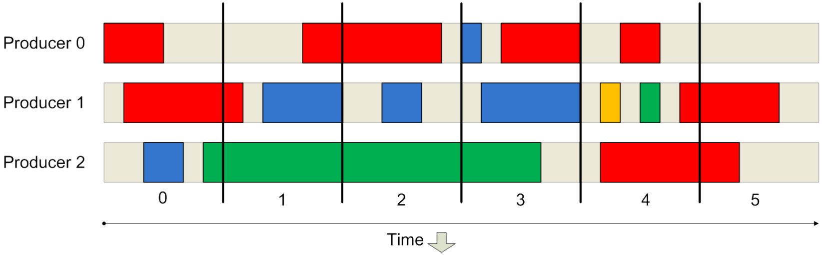
Microscopic Level: Producer Activity Time Matrix



	0	1	2	3	4	5
Producer 0	0.5	0.33	0.83	0.83	0.33	0
Producer 1	0.83	0.83	0.33	0.83	0.33	0.66
Producer 2	0.5	1	1	0.66	0.83	0.33

Part number

Microscopic Level: State Activity Time Cubic Matrix



	0	1	2	3	4	5
Producer 0	0.5	0.33	0.83	0.66	0.33	0
	0	0	0	0.16	0	0
	0	0	0	0	0	0
	0	0	0	0	0	0
Producer 1	0.83	0.16	0	0	0.16	0.66
	0	0	0.33	0.83	0	0
	0	0	0	0	0.16	0
	0	0	0	0	0.16	0
Producer 2	0	0	0	0	0.83	0.33
	0.33	0	0	0	0.33	0
	0.16	1	1	0.66	0.33	0
	0	0	0	0	0	0

Part number

Quality Computation

Gain and loss formulas: originally for scalars

012345					
01234	12345				
0123	1234	2345			
012	123	234	345		
01	12	23	34	45	
0	1	2	3	4	5

Adaptation for time-sliced description

- Vector (ex: activity time per process)
  $\text{quality}(A) = \sum_{i \in n} \text{quality}(A[i])$
- Matrix (ex: activity time per state type)
  $\text{quality}(A) = \sum_{i \in n} (\sum_{j \in m} \text{quality}(A[i][j]))$

Best-Cut Partition for a given *p*