Ordonnancement multicoeurs optimisé de chaînes sur AFF3CT

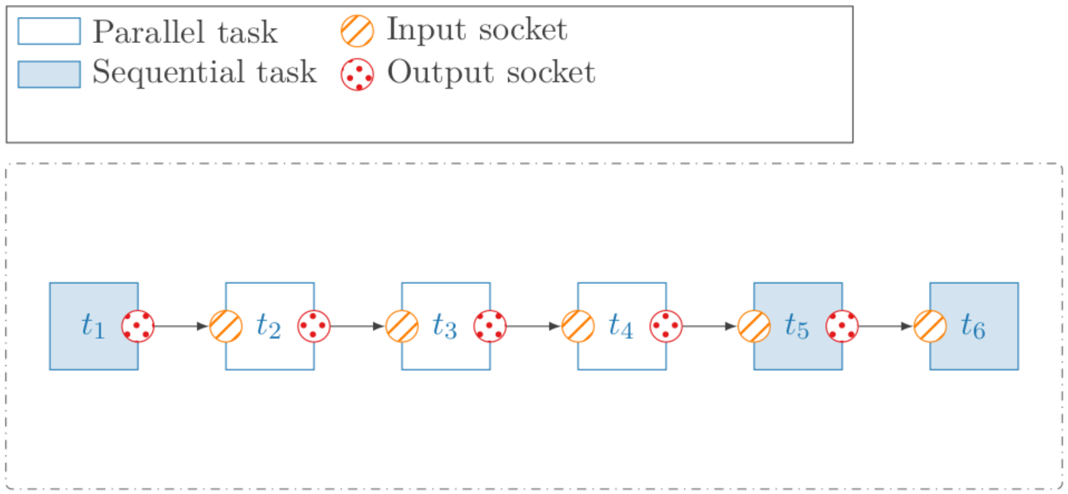
Diane ORHAN

diane.orhan@inria.fr

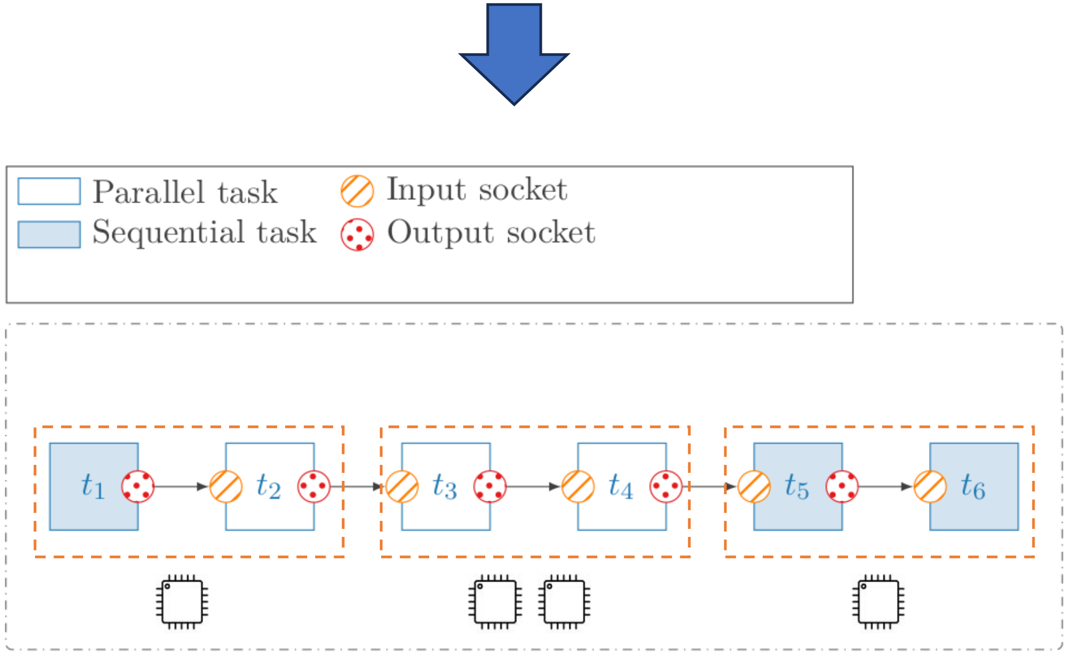
28th November 2023

Starting point

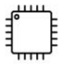
- 1 filter = 1 task/bloc implemented (AFF3CT)
- How to split the computation for a given architecture while having good performances ?
- For now: done manually by the user



Pipeline user description



Pipeline user description

 *1 core with 1 thread*

Main issue

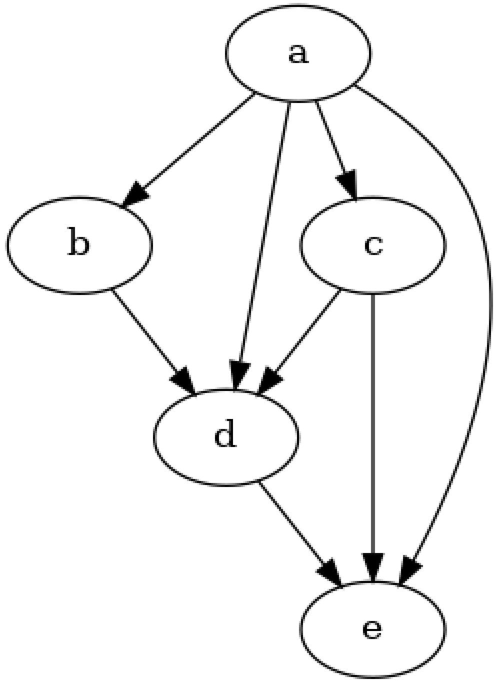
For a given multi-core architecture, with identical and homogeneous resources, **automatically** partition radio chain.

Objectives:

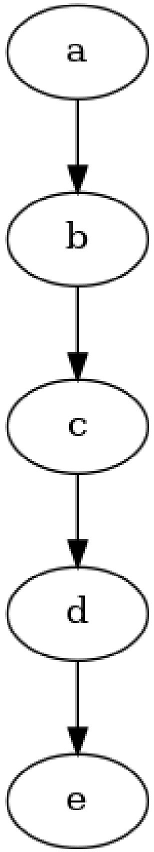
1. Maximize frame processing throughput
2. Minimize use of resources

What does it mean ?

1. Automatic parallelization of a **task chain**
2. Automatic parallelization of **several task chain** on the same architecture
3. Automatic parallelization of a DAG



Directed Acyclic Graph



Task chain

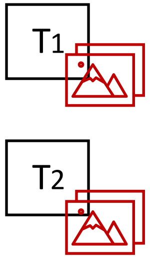
What about GNU Radio ?

- For now:
  - Version 3.0
  - Thread Per Block (*task*) Scheduling
  - Thread management by the OS/runtime

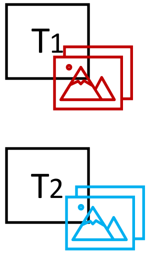
- Future development:
  - Version 4.0
  - Allow more than one block per thread
  - Could use OTAC

State of the art: Pipeline workflow scheduling

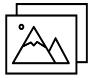
- Task parallelism

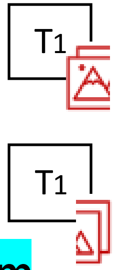


- Pipeline parallelism

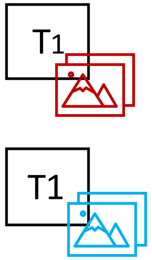


- Data parallelism





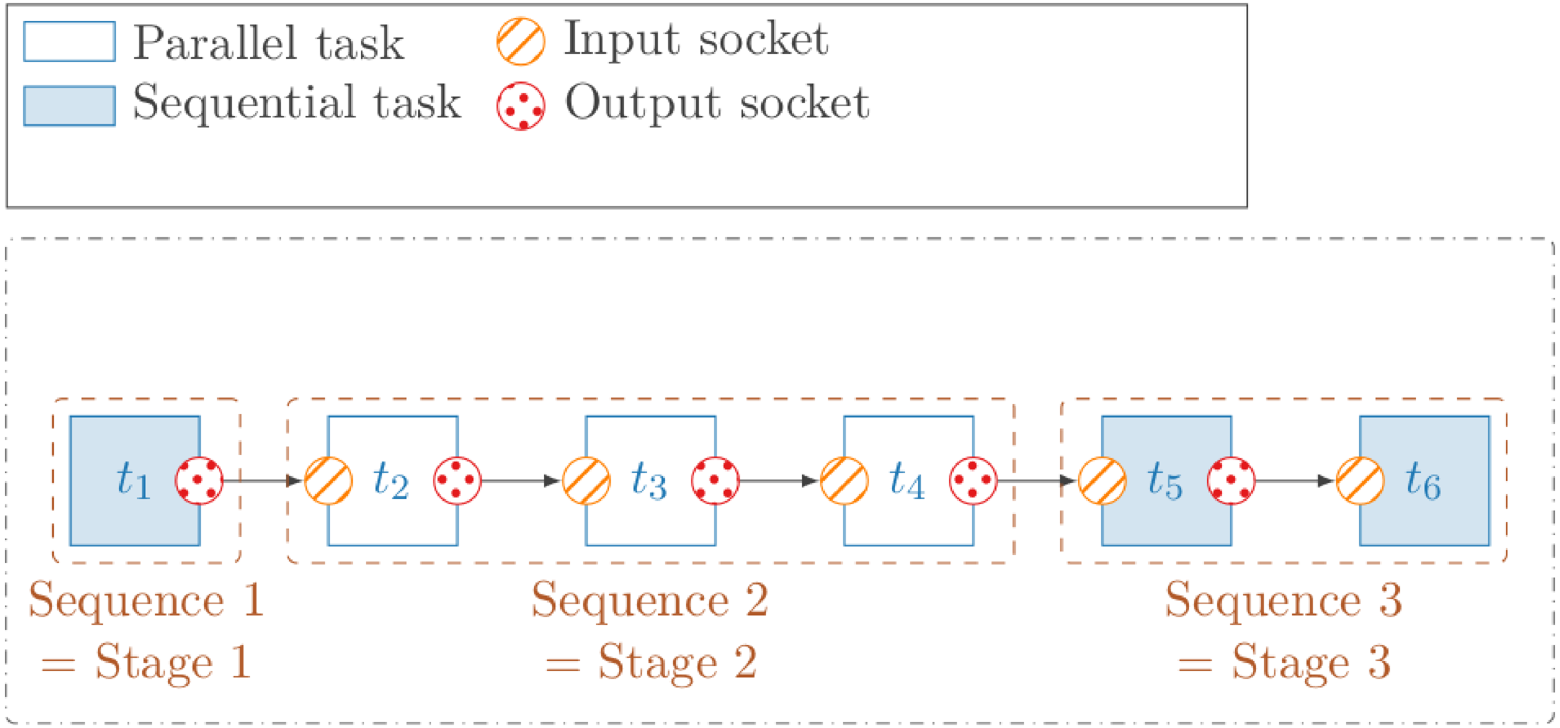
- Replicated parallelism



Anne Benoit, Ümit V. Çatalyürek, Yves Robert, and Erik Saule. 2013. A Survey of Pipelined Workflow Scheduling: Models and Algorithms. ACM Comput. Surv. 45, 4, Article 50 (aug 2013), 36 pages. https://doi.org/10.1145/2501654.2501664

Parallelization operations available

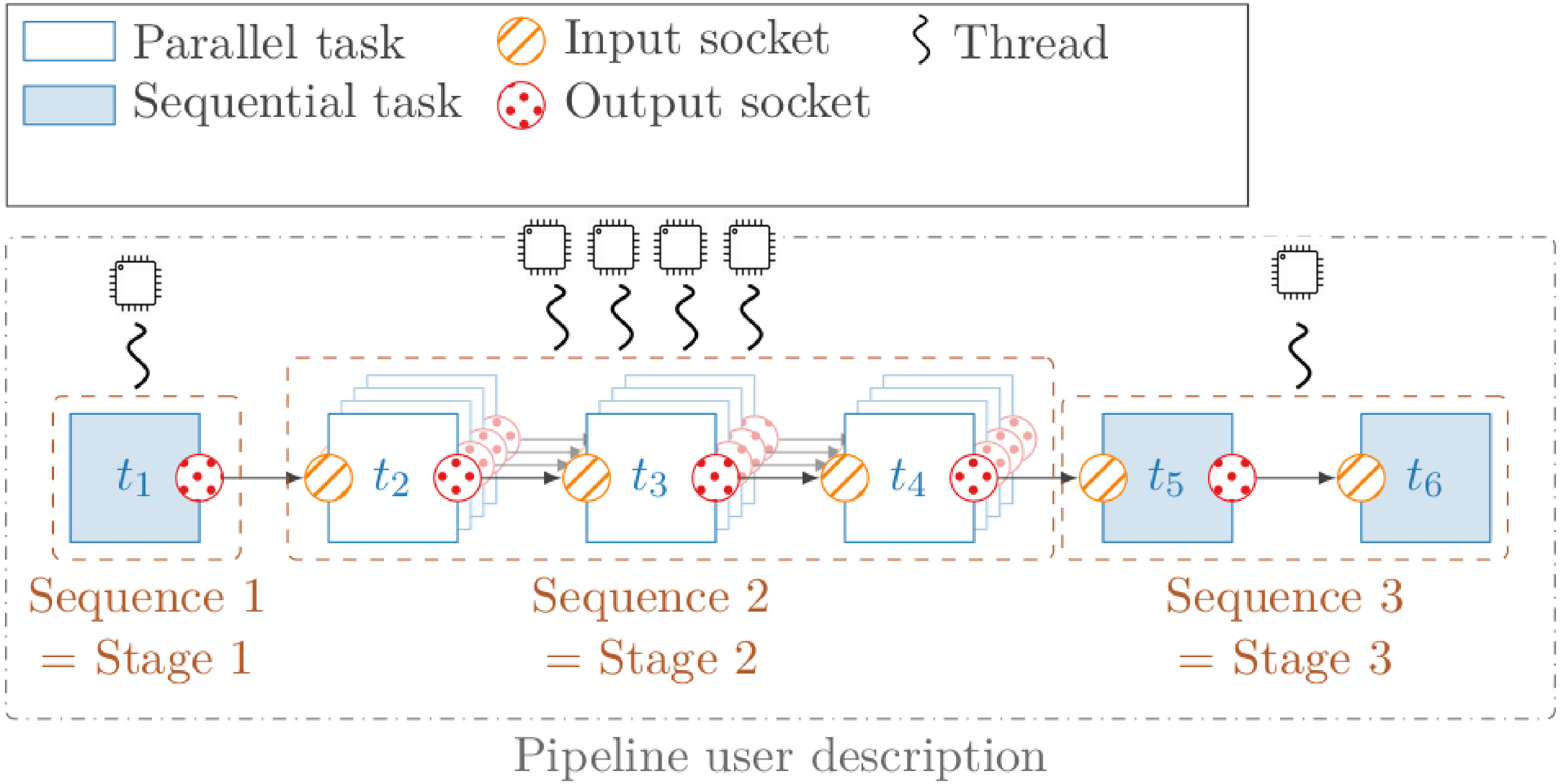
Pipeline



Pipeline user description

Parallelization operations available

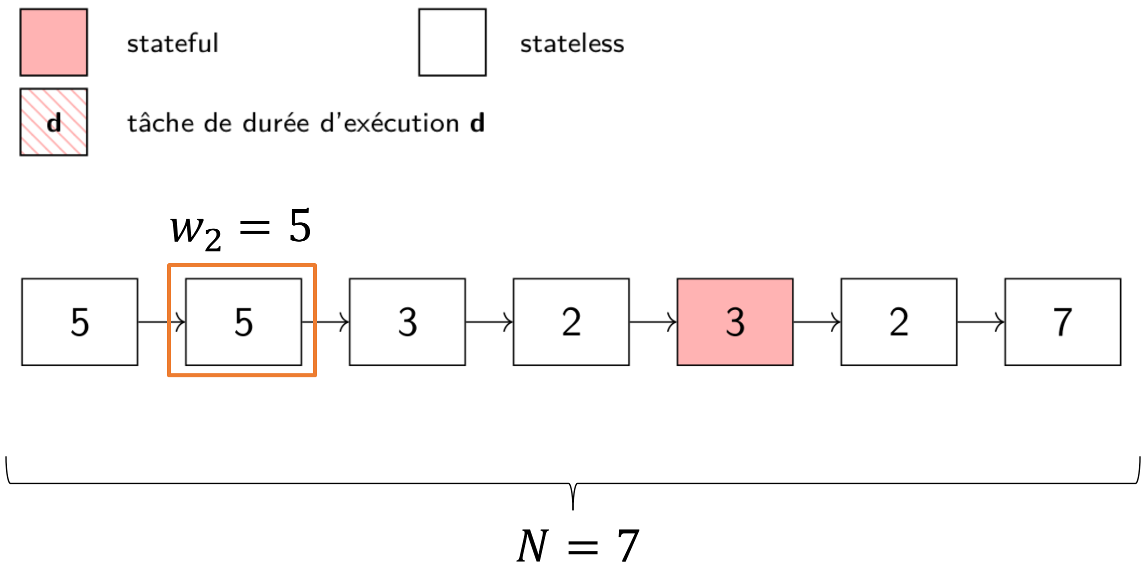
Replication



Automatic parallelization of task chain

- N : number of tasks in the chain
- P : number of available resources
- w_i : weight of a task τ_i

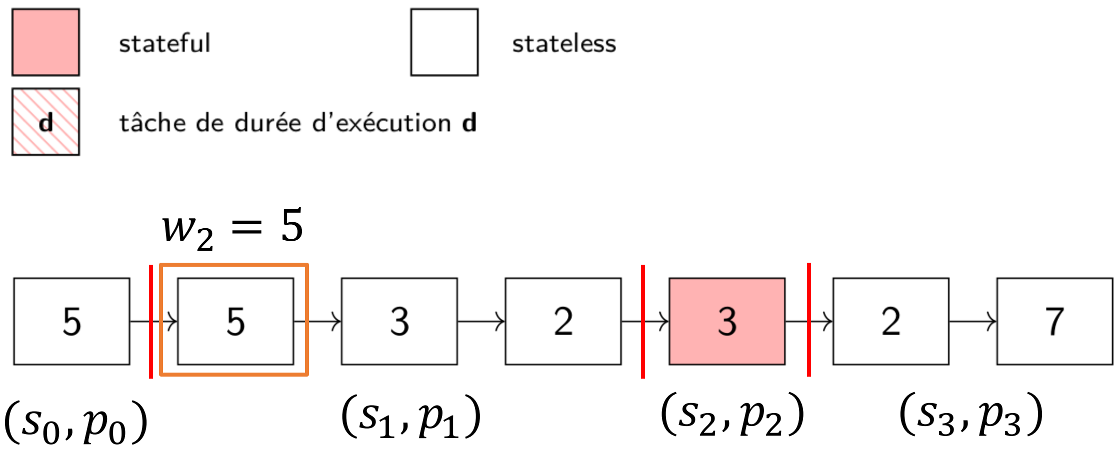
→ Find optimal partitioning in terms of throughput for a given P



Automatic parallelization of task chain

- N : number of tasks in the chain
- P : number of available resources
- w_i : weight of a task τ_i
- s_i : i^e stage
- p_i : number of resources dedicated to s_i

→ Find optimal partitioning in terms of throughput for a given P

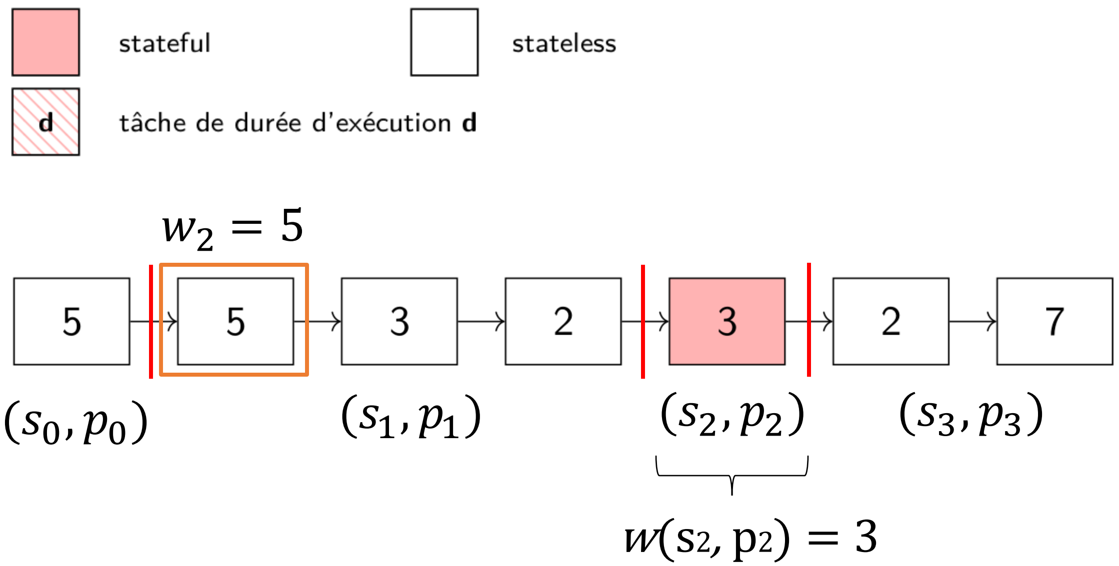


Automatic parallelization of task chain

- N : number of tasks in the chain
- P : number of available resources
- w_i : weight of a task τ_i
- s_i : i^e stage
- p_i : number of resources dedicated to s_i
- Weight of the i^e stage :

$$w(s_i, p_i) = \begin{cases} \sum_{\tau \in s_i} w_i & \text{if } s_i \text{ stateful, } p_i \geq 1 \\ \frac{1}{p_i} \sum_{\tau \in s_i} w_i & \text{if } s_i \text{ stateless, } p_i \geq 1 \\ \infty & \text{otherwise} \end{cases}$$

→ Find optimal partitioning in terms of throughput for a given P



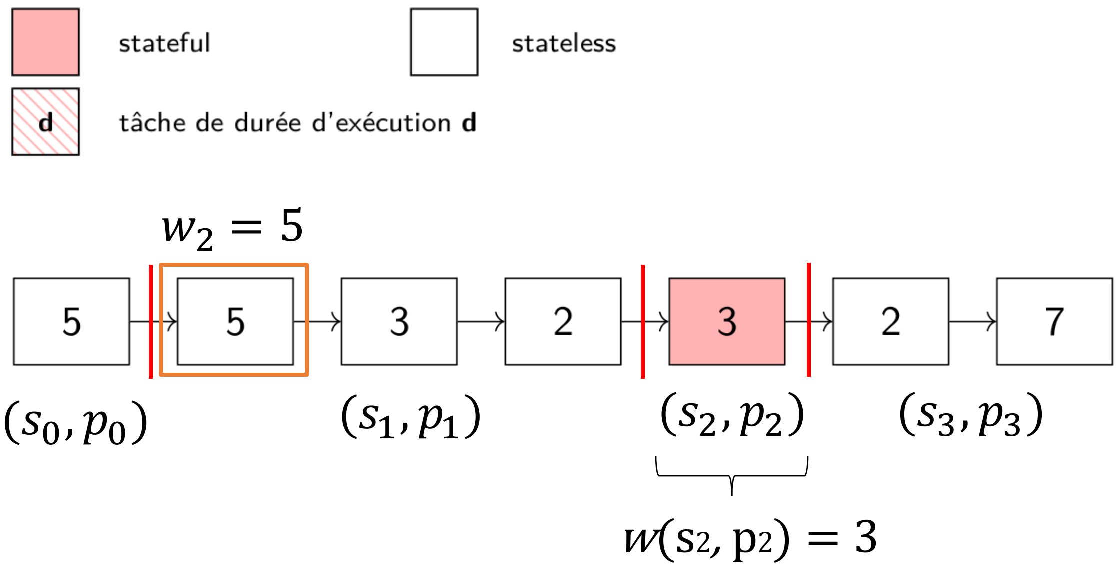
Automatic parallelization of task chain

- N : number of tasks in the chain
- P : number of available resources
- w_i : weight of a task τ_i
- s_i : i^e stage
- p_i : number of resources dedicated to s_i
- Weight of the i^e stage :

$$w(s_i, p_i) = \begin{cases} \sum_{\tau \in s_i} w_i & \text{if } s_i \text{ stateful, } p_i \geq 1 \\ \frac{1}{p_i} \sum_{\tau \in s_i} w_i & \text{if } s_i \text{ stateless, } p_i \geq 1 \\ \infty & \text{otherwise} \end{cases}$$

- $T^{-1} = \max_i w(s_i, p_i)$: the longest stage duration

→ Find optimal partitioning in terms of throughput for a given P



OTAC: Optimal maximal-packing for TAsk Chains

Goals :

1. Maximize the throughput i.e. **minimize T^{-1}**, the largest stage pipeline weight **automatically**
2. **Minimize the number of used resources**

Hypothesis :

- Identical and homogeneous resources
- No synchronization or data movement overhead

Proof of optimality: see *OTAC: Optimal Scheduling for Pipelined and Replicated Task Chains for Software-Defined Radio*, Orhan, Lima Pilla, Barthou, Cassagne, Aumage, Tajan, Jego, Leroux
https://hal.science/hal-04228117

OTAC Algorithm

Two functions:

- PROBE
  - Find if, solution with target T^{-1} and P resources, is feasible
- SOLVE
  - Use PROBE and with a binary research on T^{-1} find the best one

Algorithm 2: $\texttt{SOLVE}(N, P)$

Result: Minimal reciprocal throughput T^{-1}

$T^{-1}_{\min} \leftarrow \frac{1}{P} \sum_{\tau \in \mathcal{T}} w_\tau$;

$T^{-1}_{\max} \leftarrow T^{-1}_{\min} + \max_{\tau \in \mathcal{T}} w_\tau$;

while $T^{-1}_{\max} - T^{-1}_{\min} \geq \frac{1}{P}$ **do**

  $T^{-1}_{\text{mid}} \leftarrow \frac{T^{-1}_{\max} + T^{-1}_{\min}}{2}$;

  $done,\ T^{-1}_{\text{mid}},\ P,\ (\mathbf{n}, \mathbf{p}) \leftarrow \texttt{PROBE}(T^{-1}_{\text{mid}})$; /* *TRUE if a valid solution is found* */

  if *done* **then**

    $T^{-1}_{\max} \leftarrow T^{-1}_{\text{mid}}$; /* T^{-1} *can only get smaller* */

  else

    $T^{-1}_{\min} \leftarrow T^{-1}_{\text{mid}}$; /* T^{-1} *can only get bigger* */

  end

end

$T^{-1} \leftarrow T^{-1}_{\max}$;

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

Find at most P-1 partitions, using **pipeline** and/or **replication** parallelism, minimizing the maximum sum of weight in all partition T^{-1}.

N=9, T^{-1}= 5

P=7

Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
State	**SL**	**SL**	**SL**	**SF**	**SL**	**SL**	**SF**	**SL**	**SL**
A	**4**	**5**	**3**	**1**	**3**	**1**	**2**	**3**	**6**
P									
W									

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

Find at most P-1 partitions, using **pipeline** and/or **replication** parallelism, minimizing the maximum sum of weight in all partition T^{-1}.

N=9, $T^{-1} = 5$

P=7

Index	0	1	2	3	4	5	6	7	8
State	SL	SL	SL	SF	SL	SL	SF	SL	SL
A	4	5	3	1	3	1	2	3	6
P									
W	$W_{seq,1}$ = **12**								

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

➡ Find at most P-1 partitions, using **pipeline** and/or **replication** parallelism, minimizing the maximum sum of weight in all partition T^{-1}.

N=9, $T^{-1}= 5$

P=7

Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
State	**SL**	**SL**	**SL**	**SF**	**SL**	**SL**	**SF**	**SL**	**SL**
A	**4**	**5**	**3**	**1**	**3**	**1**	**2**	**3**	**6**
P	$p_1=\lceil w_{seq,1}/T^{-1} \rceil$								
W	**12/3 = 4**								

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

Find at most P-1 partitions, using **pipeline** and/or **replication** parallelism, minimizing the maximum sum of weight in all partition T^{-1}.

N=9, T^{-1}= 5

P=7

Index	0	1	2	3	4	5	6	7	8
State	SL	SL	SL	SF	SL	SL	SF	SL	SL
A	4	5	3	1	3	1	2	3	6
P		3							
W		4							

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

Find at most P-1 partitions, using **pipeline** and/or **replication** parallelism, minimizing the maximum sum of weight in all partition T^{-1}.

N=9, $T^{-1} = 5$

P=7

Index	0	1	2	3	4	5	6	7	8
State	SL	SL	SL	SF	SL	SL	SF	SL	SL
A	4	5	3	1	3	1	2	3	6
P		3 (2)	?						
W		4							

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

➡ Find at most P-1 partitions, using **pipeline** and/or **replication** parallelism, minimizing the maximum sum of weight in all partition T^{-1}.

N=9, P=7

T^{-1}= 5

Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
State	**SL**	**SL**	**SL**	**SF**	**SL**	**SL**	**SF**	**SL**	**SL**
A	**4**	**5**	**3**	**1**	**3**	**1**	**2**	**3**	**6**
P		**3 (2)**	**?**						
W		**4 (4.5)**							

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

➡ Find at most P-1 partitions, using **pipeline** and/or **replication** parallelism, minimizing the maximum sum of weight in all partition T^{-1}.

N=9, T^{-1}= 5

P=7

Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
State	**SL**	**SL**	**SL**	**SF**	**SL**	**SL**	**SF**	**SL**	**SL**
A	**4**	**5**	**3**	**1**	**3**	**1**	**2**	**3**	**6**
P	**2**								
W	**4.5**								

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

Find at most P-1 partitions, using **pipeline** and/or **replication** parallelism, minimizing the maximum sum of weight in all partition T^{-1}.

N=9, $T^{-1}= 5$

P=7

Index	0	1	2	3	4	5	6	7	8
State	SL	SL	SL	SF	SL	SL	SF	SL	SL
A	4	5	3	1	3	1	2	3	6
P	2		1						
W	4.5		4						

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

Find at most P-1 partitions, using **pipeline** and/or **replication** parallelism, minimizing the maximum sum of weight in all partition T^{-1}.

N=9, T^{-1}= 5

P=7

Index	0	1	2	3	4	5	6	7	8
State	SL	SL	SL	SF	SL	SL	SF	SL	SL
A	4	5	3	1	3	1	2	3	6
P	2		1		1				
W	4.5		4		4				

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

Find at most P-1 partitions, using **pipeline** and/or **replication** parallelism, minimizing the maximum sum of weight in all partition T^{-1}.

N=9, $T^{-1}= 5$

P=7

Index	0	1	2	3	4	5	6	7	8
State	SL	SL	SL	SF	SL	SL	SF	SL	SL
A	4	5	3	1	3	1	2	3	6
P	2		1		1		1		
W	4.5		4		4		5		

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

Find at most P-1 partitions, using **pipeline** and/or **replication** parallelism, minimizing the maximum sum of weight in all partition T^{-1}.

N=9, T^{-1}= 5

P=7

Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
State	**SL**	**SL**	**SL**	**SF**	**SL**	**SL**	**SF**	**SL**	**SL**
A	**4**	**5**	**3**	**1**	**3**	**1**	**2**	**3**	**6**
P	**2**		**1**		**1**		**1**		**2**
W	**4.5**		**4**		**4**		**5**		**3**

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC algorithm

➡ Find at most P-1 partitions, using **pipeline** and/or **replication** parallelism, minimizing the maximum sum of weight in all partition T^{-1}.

N=9, $T^{-1} = 5$

P=7

Index	0	1	2	3	4	5	6	7	8
State	SL	SL	SL	SF	SL	SL	SF	SL	SL
A	4	5	3	1	3	1	2	3	6
P	2		1		1		1		2
W	4.5		4		4		5		3

P used: 7

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

Binary search: find the best T^{-1} for P=7
Lower Bound = $\lceil 28/7 \rceil$ = 4
Upper Bound = LB + 6 = 10

Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
State	**SL**	**SL**	**SL**	**SF**	**SL**	**SL**	**SF**	**SL**	**SL**
A	**4**	**5**	**3**	**1**	**3**	**1**	**2**	**3**	**6**

Total : 28

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

➡ Binary search: find the best T^{-1} for P=7

Lower Bound = ⌈28/7⌉ = 4
Upper Bound = LB + 6 = 10

T^{-1} = (LB+UB)/2 = 7

Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
State	**SL**	**SL**	**SL**	**SF**	**SL**	**SL**	**SF**	**SL**	**SL**
A	**4**	**5**	**3**	**1**	**3**	**1**	**2**	**3**	**6**

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

Binary search: find the best T^{-1} for P=7

Lower Bound = ⌈28/7⌉ = 4
Upper Bound = LB + 6 = 10

T^{-1} = (LB+UB)/2 = 7

	Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	
	State	**SL**	**SL**	**SL**	**SF**	**SL**	**SL**	**SF**	**SL**	**SL**	
	A	**4**	**5**	**3**	**1**	**3**	**1**	**2**	**3**	**6**	P Total :
T^{-1}=7	**P**		**2**			**1**			**2**		5 ✓

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

Binary search: find the best T^{-1} for P=7

Lower Bound = 4
Upper Bound = 7

T^{-1} = (LB+UB)/2 = 5.5

	Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**	
	State	**SL**	**SL**	**SL**	**SF**	**SL**	**SL**	**SF**	**SL**	**SL**	
	A	**4**	**5**	**3**	**1**	**3**	**1**	**2**	**3**	**6**	P Total :
T^{-1}=7	**P**		**3**				**1**			**2**	6 ✓

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

➡ Binary search: find the best T^{-1} for P=7

Lower Bound = 4
Upper Bound = 7

T^{-1} = (LB+UB)/2 = 5.5

<table>
<tr><th></th><th>Index</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th></th></tr>
<tr><td></td><td>State</td><td>SL</td><td>SL</td><td>SL</td><td>SF</td><td>SL</td><td>SL</td><td>SF</td><td>SL</td><td>SL</td><td></td></tr>
<tr><td></td><td>A</td><td>4</td><td>5</td><td>3</td><td>1</td><td>3</td><td>1</td><td>2</td><td>3</td><td>6</td><td>P Total :</td></tr>
<tr><td>T^{-1}=7</td><td>P</td><td colspan="3">3</td><td colspan="4">1</td><td colspan="2">2</td><td>6</td></tr>
<tr><td>T^{-1}=5.5</td><td>P</td><td colspan="3">3</td><td colspan="3">1</td><td colspan="2">1</td><td>2</td><td>7</td></tr>
</table>

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

➡ Binary search: find the best T^{-1} for P=7

Lower Bound = 4
Upper Bound = 5.5

T^{-1} = (LB+UB)/2 = 4.75

	Index	0	1	2	3	4	5	6	7	8	
	State	SL	SL	SL	SF	SL	SL	SF	SL	SL	
	A	4	5	3	1	3	1	2	3	6	P Total :
T^{-1}=7	P		3				1			2	6 ✓
T^{-1}=5.5	P		3			1			1	2	7 ✓

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

➡ Binary search: find the best T^{-1} for P=7

Lower Bound = 4
Upper Bound = 5.5

T^{-1} = (LB+UB)/2 = 4.75

<table>
<tr><th></th><th>Index</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th></th></tr>
<tr><td></td><td>State</td><td>SL</td><td>SL</td><td>SL</td><td>SF</td><td>SL</td><td>SL</td><td>SF</td><td>SL</td><td>SL</td><td></td></tr>
<tr><td></td><td>A</td><td>4</td><td>5</td><td>3</td><td>1</td><td>3</td><td>1</td><td>2</td><td>3</td><td>6</td><td><u>P Total :</u></td></tr>
<tr><td>T^{-1}=7</td><td>P</td><td colspan="3">3</td><td colspan="4">1</td><td colspan="2">2</td><td>6 ✓</td></tr>
<tr><td>T^{-1}=5.5</td><td>P</td><td colspan="3">3</td><td colspan="3">1</td><td colspan="2">1</td><td>2</td><td>7 ✓</td></tr>
</table>

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

Binary search: find the best T^{-1} for P=7

Lower Bound = 4
Upper Bound = 5.5

T^{-1} = (LB+UB)/2 = 4.75

<table>
<tr><th></th><th>Index</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th></th></tr>
<tr><td></td><td>State</td><td>SL</td><td>SL</td><td>SL</td><td>SF</td><td>SL</td><td>SL</td><td>SF</td><td>SL</td><td>SL</td><td></td></tr>
<tr><td></td><td>A</td><td>4</td><td>5</td><td>3</td><td>1</td><td>3</td><td>1</td><td>2</td><td>3</td><td>6</td><td>P Total :</td></tr>
<tr><td>T^{-1}=7</td><td>P</td><td colspan="3">3</td><td colspan="4">1</td><td colspan="2">2</td><td>6 ✓</td></tr>
<tr><td>T^{-1}=5.5</td><td>P</td><td colspan="3">3</td><td colspan="3">1</td><td colspan="2">1</td><td>2</td><td>7 ✓</td></tr>
<tr><td>T^{-1}=4.75</td><td>P</td><td colspan="3">4</td><td colspan="2">1</td><td colspan="2">1</td><td colspan="2">2</td><td>8 ✗</td></tr>
</table>

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

Binary search: find the best T^{-1} for P=7

Lower Bound = 4.75
Upper Bound = 5.5

$T^{-1} = (LB+UB)/2 = 5.125$

<table>
<tr><th></th><th>Index</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th></th></tr>
<tr><td></td><td>State</td><td>SL</td><td>SL</td><td>SL</td><td>SF</td><td>SL</td><td>SL</td><td>SF</td><td>SL</td><td>SL</td><td></td></tr>
<tr><td></td><td>A</td><td>4</td><td>5</td><td>3</td><td>1</td><td>3</td><td>1</td><td>2</td><td>3</td><td>6</td><td><u>P Total :</u></td></tr>
<tr><td>$T^{-1}=7$</td><td>P</td><td colspan="3">3</td><td colspan="4">1</td><td colspan="2">2</td><td>6 ✓</td></tr>
<tr><td>$T^{-1}=5.5$</td><td>P</td><td colspan="3">3</td><td colspan="3">1</td><td colspan="2">1</td><td>2</td><td>7 ✓</td></tr>
<tr><td>$T^{-1}=4.75$</td><td>P</td><td colspan="3">4</td><td colspan="2">1</td><td colspan="2">1</td><td colspan="2">2</td><td>8 ✗</td></tr>
</table>

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

➡ Binary search: find the best T^{-1} for P=7

Lower Bound = 4.75
Upper Bound = 5.5

T^{-1} = (LB+UB)/2 = 5.125

<table>
<tr><th></th><th>Index</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th></th></tr>
<tr><td></td><td>State</td><td>SL</td><td>SL</td><td>SL</td><td>SF</td><td>SL</td><td>SL</td><td>SF</td><td>SL</td><td>SL</td><td></td></tr>
<tr><td></td><td>A</td><td>4</td><td>5</td><td>3</td><td>1</td><td>3</td><td>1</td><td>2</td><td>3</td><td>6</td><td>P Total :</td></tr>
<tr><td>$T^{-1}=7$</td><td>P</td><td colspan="3">3</td><td colspan="4">1</td><td colspan="2">2</td><td>6 ✓</td></tr>
<tr><td>$T^{-1}=5.5$</td><td>P</td><td colspan="3">3</td><td colspan="3">1</td><td colspan="2">1</td><td>2</td><td>7 ✓</td></tr>
<tr><td>$T^{-1}=4.75$</td><td>P</td><td colspan="3">4</td><td colspan="2">1</td><td colspan="2">1</td><td colspan="2">2</td><td>8 ✗</td></tr>
<tr><td>$T^{-1}=5$</td><td>P</td><td colspan="2">2</td><td colspan="2">1</td><td colspan="2">1</td><td colspan="2">1</td><td>2</td><td>7 ✓</td></tr>
</table>

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

➡ Binary search: find the best T^{-1} for P=7

Lower Bound = 4.75
Upper Bound = 5.125

UB-LB = 0.375 < 1 with non-integer values < 1/P

<table>
<tr><th></th><th>Index</th><th>0</th><th>1</th><th>2</th><th>3</th><th>4</th><th>5</th><th>6</th><th>7</th><th>8</th><th></th></tr>
<tr><td></td><td>State</td><td>SL</td><td>SL</td><td>SL</td><td>SF</td><td>SL</td><td>SL</td><td>SF</td><td>SL</td><td>SL</td><td></td></tr>
<tr><td></td><td>A</td><td>4</td><td>5</td><td>3</td><td>1</td><td>3</td><td>1</td><td>2</td><td>3</td><td>6</td><td><u>P Total :</u></td></tr>
<tr><td>$T^{-1}=7$</td><td>P</td><td colspan="3">3</td><td colspan="4">1</td><td colspan="2">2</td><td>6 ✓</td></tr>
<tr><td>$T^{-1}=5.5$</td><td>P</td><td colspan="3">3</td><td colspan="3">1</td><td colspan="2">1</td><td>2</td><td>7 ✓</td></tr>
<tr><td>$T^{-1}=4.75$</td><td>P</td><td colspan="3">4</td><td colspan="2">1</td><td colspan="2">1</td><td colspan="2">2</td><td>8 ✗</td></tr>
<tr><td>$T^{-1}=5$</td><td>P</td><td colspan="2">2</td><td colspan="2">1</td><td colspan="2">1</td><td colspan="2">1</td><td>2</td><td>7 ✓</td></tr>
</table>

SF = stateful (task **not** replicable)
SL = stateless (task replicable)

OTAC Algorithm

Binary search: find the best T^{-1} for P=7

Lower Bound = 4.75
Upper Bound = 5.125

UB-LB = 0.375 < 1 with non-integer values < 1/P

Index	**0**	**1**	**2**	**3**	**4**	**5**	**6**	**7**	**8**
State	**SL**	**SL**	**SL**	**SF**	**SL**	**SL**	**SF**	**SL**	**SL**
A	**4**	**5**	**3**	**1**	**3**	**1**	**2**	**3**	**6**
P	**2**		**1**		**1**		**1**		**2**
W	**4.5**		**4**		**4**		**5**		**3**

P Total :
7

For this specific problem with P=7, T^{-1}=5

Evaluation

1. Synthetic chains
2. Concrete use-case: DVB-S2

Evaluation: synthetic chains

Workload:

- 1000 randomly generated chains
- 20 tasks each
- Computational time [100,35000] μs
- Stateless ratio SR $\in$ [0, 1]

Variation of SR from 0 to 1 with step 0.1

P = {5, 10, 15, 25}

Evaluation: synthetic chains

Comparison to other algorithms ?

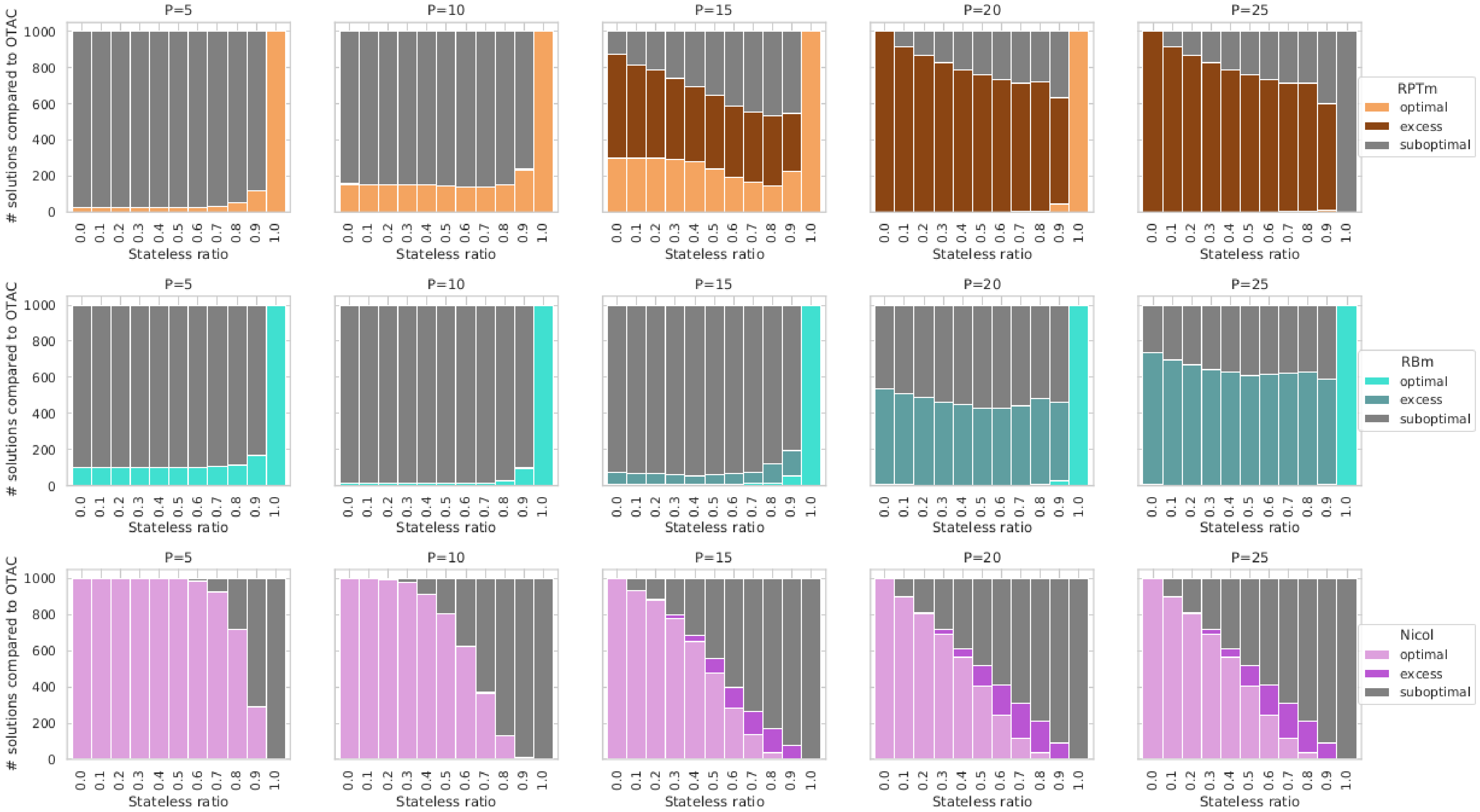
- **Nicol**: optimal algorithm for pipelining only
- **RB**: recursived-bipartitioning-based scheduler
- **RBm**: as RB + merge neighboring stateless stages together
- **RPT**: one resources per task, if N>P, fuses stages together
- **RPTm**: as RPT + merge neighboring stateless stages together

Evaluation: synthetic chains

Comparison to other algorithms ?

- **Nicol**: optimal algorithm for pipelining only
- ~~**RB**: recursived-bipartitioning-based scheduler~~
- **RBm**: as RB + merge neighboring stateless stages together
- ~~**RPT**: one resources per task, if N>P, fuses stages together~~
- **RPTm**: as RPT + merge neighboring stateless stages together

Number of solutions of different qualities for different schedulers, numbers of processors, and stateless ratios compared to OTAC.



- **Optimal**: maximum throughput with minimum number of resources.
- **Excess**: maximum throughput with extra resources.
- **Suboptimal**: suboptimal throughput.

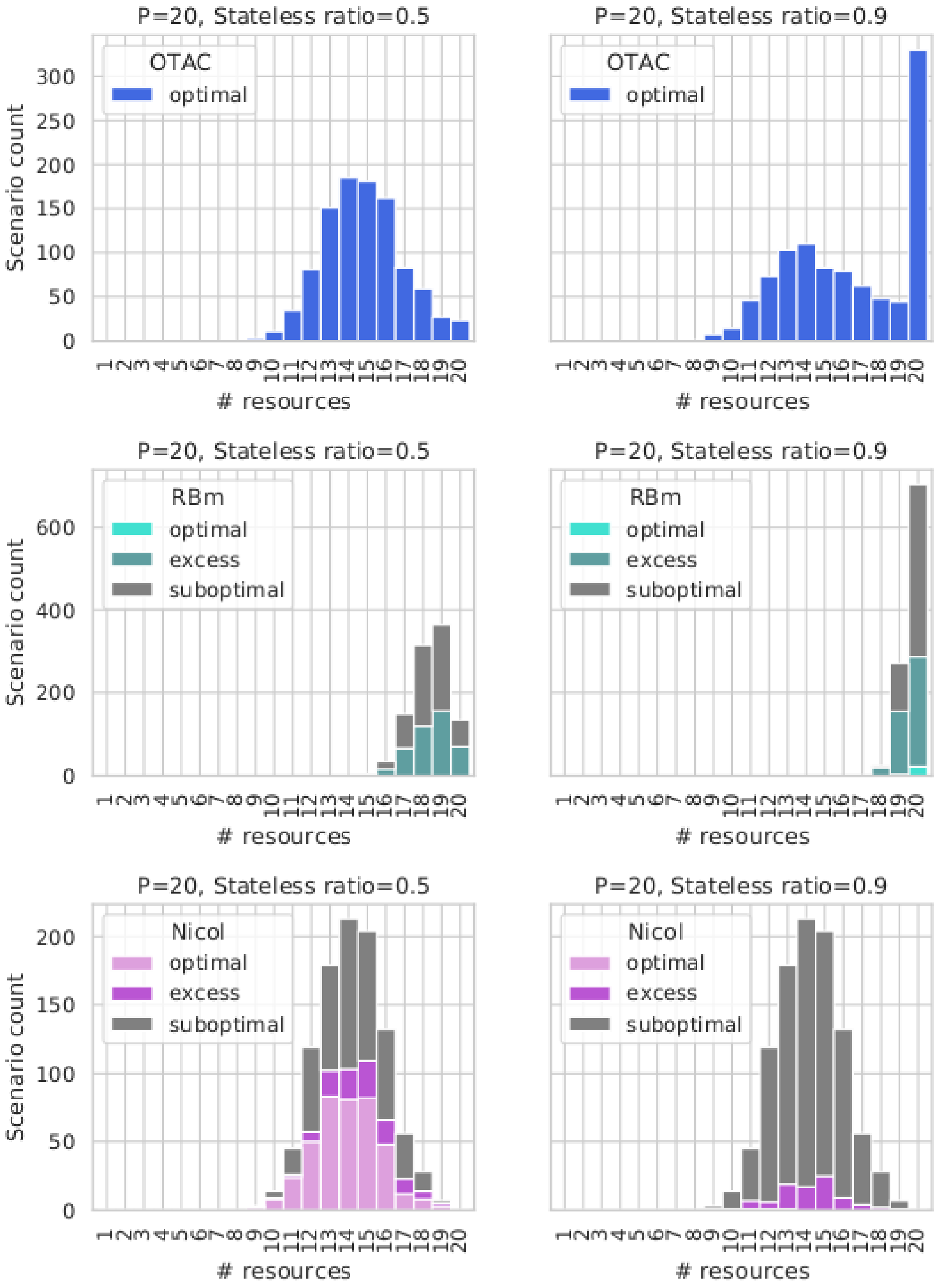
Simulation: histograms of number of resources used (P)

Schedules computed by:

- OTAC
- RBm
- Nicol

Configurations:

- Left: P = 20 and SR = 0.5
- Right: P = 20 and SR = 0.9



Evaluation: concrete use case DVB-S2

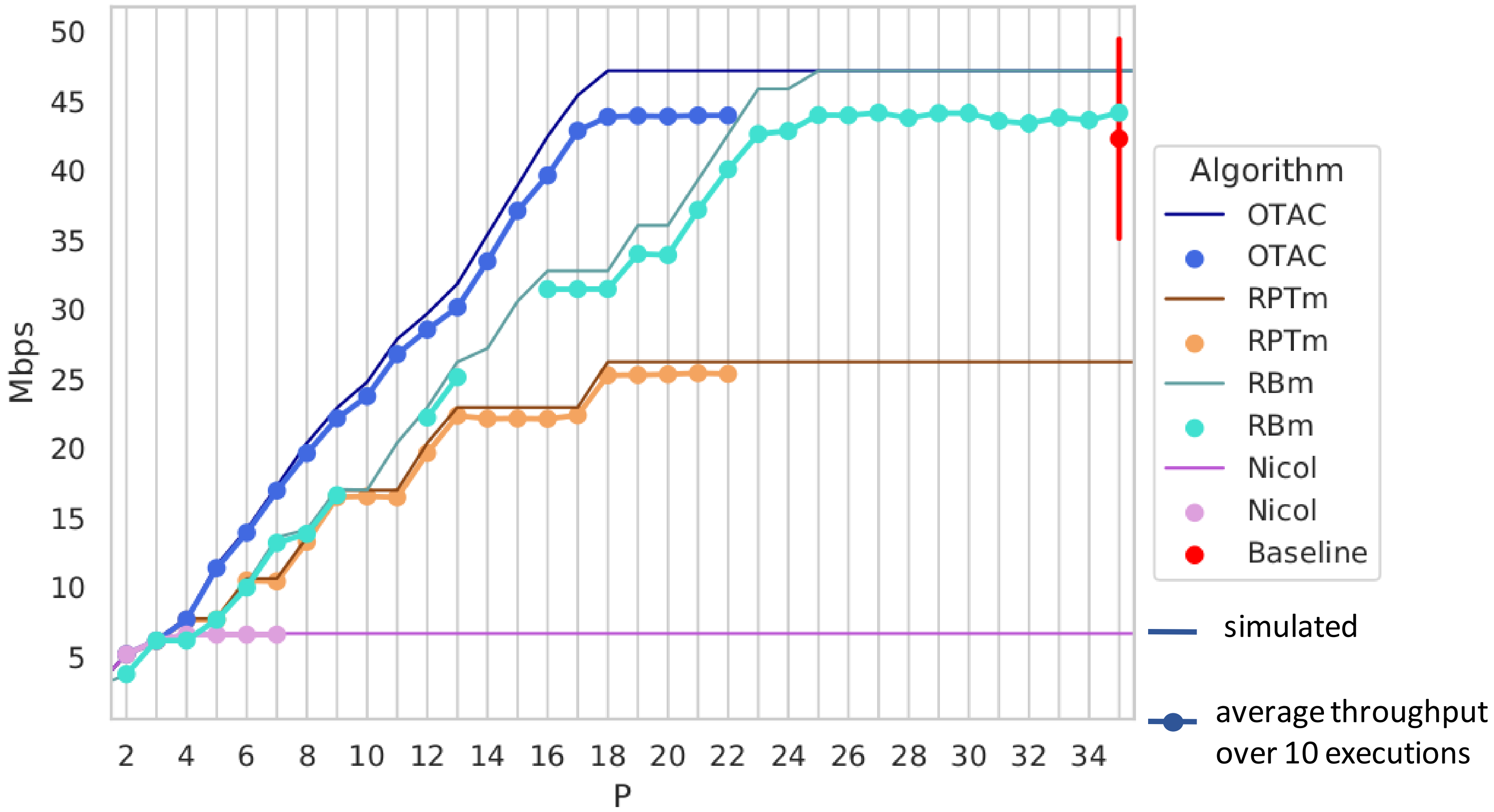
Hardware & Software environment

- 2 x 18 Intel Xeon Skylake, 3.2 GHz
- No hyperthreading
- AFF3CT v3.0.2
- gcc v12.2.0 - Python v3.8.0
- Threads pinned onto core with hwloc v2.7

DVB-S2 configuration

- Satellite transmission of video contents
- Reception chain at transmission time
- Frame packet of 16 frames
- 1.28 GB decoded by execution
- Noise 4 dB

Evaluation: concrete use case DVB-S2



	Simulated T	T average	P
OTAC	47.17	43.86	18
RPTm	26.21	25.26	18
RBm	47.17	43.98	25
Nicol	6.66	6.58	4
Baseline	47.17	42.3	35

Simulated and average throughput, number of resources needed for the best results achieved

Conclusion

Summary:

- Optimal algorithm maximizing throughput and minimizing resources

On-going:

- Automatic parallelization of several task chain on the same architecture

To go further:

- Manage more complex graphs (loop, condition, DAG)
- Is it transferable for neural network training ? => internship
- On-the-fly noise variation