Dominance of K-Periodic schedules for evaluating the maximum throughput of a SDF

B.Bodin[1,2] A.Munier-Kordon [1,2] B.Dupont de Dinechin [2]

[1]LIP6, Université P. et M. Curie, France

[2]Kalray

Lyon, July 2014

Outline

Problem Formulation

Dataflow scheduling

Dominant set of periodicity vectors

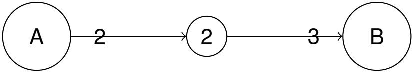
Conclusions and Perspectives

Synchronous Dataflow graph (SDF)

A simple formalism introduced by Lee and Messerschmitt 87 to model communications (DSP/parallel computation)

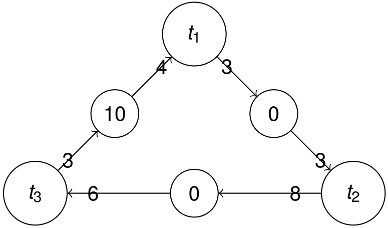
- Nodes $\rightarrow$ Actors;
- Arcs $\rightarrow$ buffers;
- Tokens $\rightarrow$ data.

A buffer between two actors A and B



Balance equation: $N_A \times 3 = N_B \times 2$

Consistence of a SDF



$$N_1 \times 3 = N_2 \times 3$$

$$N_2 \times 6 = N_3 \times 8$$

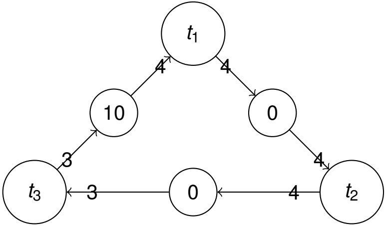
$$N_3 \times 4 = N_1 \times 3$$

$$N_1 = N_2 = 4,\ N_3 = 3$$

Definition (Lee and Messerschmitt 78)

A SDF is consistent if a repetition vector N exists.

Normalization of a SDF



$$M = PPCM(N_1, N_2, N_3) = 12$$

$$Z_1 = \frac{M}{N_1} = 4$$

$$Z_3 = \frac{M}{N_2} = 4$$

$$Z_3 = \frac{M}{N_3} = 3$$

Theorem (Marchetti and Munier 09)

A SDF is consistent iff it is normalized.

Normalized SDF

Definition

A normalized SDF is a graph $\mathcal{G} = (\mathcal{T}, \mathcal{B}, Z, M_0)$ such that

- $\mathcal{T}$ is the set of actors,
- $\mathcal{B}$ is the set of buffers,
- for any actor $t \in \mathcal{T}$, $Z_t \in \mathbb{N}^\star$ is the quantity of tokens consumed/produced by t on each adjacent buffer,
- $M_0 : \mathcal{B} \rightarrow \mathbb{N}$ is the initial marking,
- $\forall t \in \mathcal{T}$, $\ell(t)$ is the duration of one execution of t,

If N is the repetition vector of $\mathcal{G}$, there exists $M \in \mathbb{N}^\star$ such that $\forall t \in \mathcal{T}$, $N_t \times Z_t = M$.

Problem Formulation

Definition

A schedule is a function $s : \mathcal{T} \times \mathbb{N}^{\star} \to \mathbb{N}$ where $s(t, n)$ denotes the nth execution of t.
s is feasible is the numbers of tokens in any buffer remain non negative.

Definition

Let s be a feasible schedule. The throughput of an actor t following s is $\lambda^s(t) = \lim_{n\to\infty} \frac{n}{s(t,n)}$.

If $\mathcal{G}$ is consistent and strongly connected, $\forall t \in \mathcal{T}$, $\lambda^s(t) \times Z_t = \lambda^s$ is a constant.

How to evaluate efficiently the maximum throughput of $\mathcal{G}$?

Simplest way: computing the earliest schedule

Actors are performed as soon as possible until a stabilization is reached.

- Advantage $\rightarrow$ the evaluation is exact (as the earliest schedule maximizes the throughput of each actor);
- Drawback $\rightarrow$ not polynomial, a K-periodic steady state is always reached after a temporary phase. Each of them are not polynomially bounded.

Not possible to use this method in an optimization process, nor for SDF with a large number of actors.

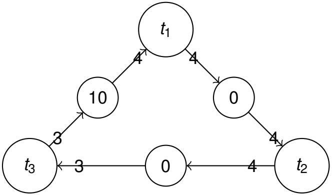
Restriction to K-Periodic schedules with $K = N$

- Many authors observed that an equivalent SDF $\mathcal{G}_{exp}$ with unit weight (i.e $Z_t = 1, \forall t \in \mathcal{T}$) may be built by expanding each actor t N_t times.
- The throughput may then be polynomially computed using classical critical circuits algorithms (Chrétienne 1982) or Max-Plus algebra (Cohen et al. 1987).

The number of nodes of $\mathcal{G}_{exp}$ is then $\sum_{t \in \mathcal{T}} N_t$ (not polynomial !).

The size of $\mathcal{G}_{exp}$ is so important, that this method cannot be considered for real life application.

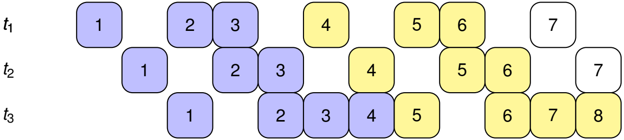
Example of K-Periodic schedule with $K = N$



$N_1 = 3$

$N_2 = 3$

$N_3 = 4$



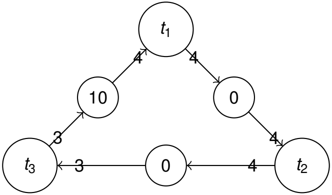
An $[3, 3, 4]$-Periodic schedule of (exact) maximum throughput $\lambda^\star = \frac{12}{5}$. The throughput of the actors are $\lambda(t_1)^\star = \frac{3}{5}$, $\lambda^\star(t_2) = \frac{3}{5}$ and $\lambda^\star(t_2) = \frac{4}{5}$.

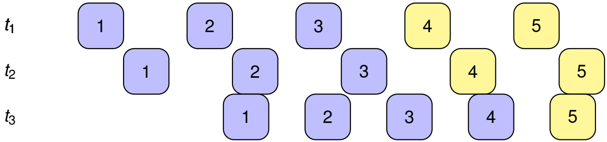
Restriction to Periodic schedule ($K_t = 1, \forall t \in \mathcal{T}$)

- The maximum throughput of a feasible periodic schedule can be computed in polynomial time (Benabid et al. 2012).
- The distance to the optimum throughput is not bounded.
- Widely used as a certificate for several optimization problems (minimization of the buffers size as example).

The evaluation of the throughput can be rather pessimistic for real-life applications.

Example of K-Periodic schedule ($K_t = 1, \forall t \in \mathcal{T}$)





An $[1, 1, 1]$-Periodic schedule of (exact) maximum throughput $\tilde{\lambda} = \frac{5}{3}$. $\tilde{\lambda}(t_1) = \frac{5}{12}$, $\tilde{\lambda}(t_2) = \frac{5}{12}$ and $\tilde{\lambda}(t_3) = \frac{5}{9}$.

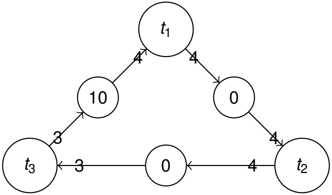
Compute the maximum throughput fixing the periodicity factor K

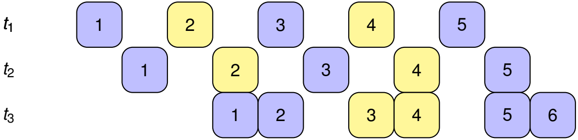
- Fix $K_t \in \{1, \cdots, N_t\}, \forall t \in \mathcal{T}$;
- Computing the maximum throughput of a K-periodic schedule can be done in time complexity $\mathcal{O}((\sum_{b=(t_i,t_j)\in\mathcal{B}} K_{t_i} \times K_{t_j})^2)$ (Bodin et al. 2012).

The problem is then to find a good trade-off between:

1. the time required to evaluate the throughput;
2. the quality of the result.

Example of Periodic schedule (K fixed arbitrarily)





An $[1,1,2]$-Periodic schedule of (exact) maximum throughput $\lambda = 2 \in [\tilde{\lambda}, \lambda^{\star}]$. $\lambda(t_1) = \frac{1}{2}$, $\lambda(t_2) = \frac{1}{2}$ and $\lambda(t_3) = \frac{2}{3}$.

Non linearity of the maximum throughput

The throughput does not necessarily increase, nor remain equal while K increases.

For our example:

- The maximum throughput for $K = (1, 1, 3)$ equals $\frac{5}{3}$;
- The maximum throughput for $K = (1, 1, 2)$ equals 2;

Which vectors K are pertinent ?

A Dominant set of periodicity vectors

Theorem

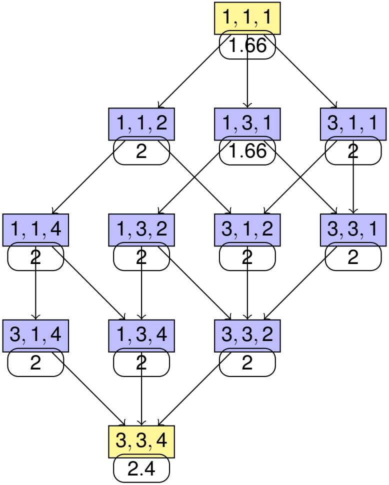
The maximum throughput $\lambda^\star$ for $K = N$ reaches the maximum throughput of $\mathcal{G}$.

Theorem (Bodin et al.)

Let be two vectors K and K' and their respective maximum throughput λ and λ'. Let suppose that, for every actor $t \in \mathcal{T}$, K_t is a divisor of K'_t. Then, $\lambda' \geq \lambda$.

The set $\mathcal{K} = \{K / \forall t \in \mathcal{T}, K_t | N_t\}$ contains at least one vector leading to the maximum throughput of $\mathcal{G}$.

An order relation for the set $\mathcal{K}$ of periodicity vectors



A Dominant set of periodicity vectors

Theorem (Bodin et al.)

Let be two vectors K and K' such that, for any actor $t \in \mathcal{T}$, $K'_t = \gcd(K_t, N_t)$. The respective maximum throughput λ and λ' are such that $\lambda' \geq \lambda$.

Since $N_3 = 4$, the throughput λ' for $K' = (1, 1, 3)$ is not better than for $K = (1, 1, 1)$.

Corollary

For any vector $K \notin \mathcal{K}$, there exists a vector $\tilde{K} \in \mathcal{K}$ with $K > \tilde{K}$ and the respective associated throughput $\tilde{\lambda} \geq \lambda$.

$\mathcal{K}$ is the set of pertinent values of K

Example of an ordered set $\mathcal{K}$ for 4 actors

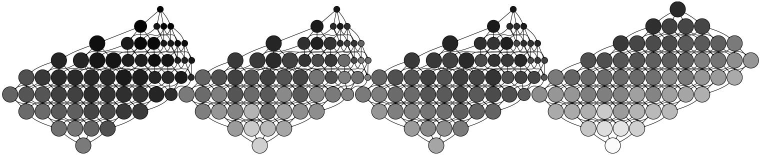


Figure: The darker is the node, the faster is the computation. The larger it is, the better the solution is.

How to find good vectors K ?

Conclusions and Perspectives

1. Building easily a periodic achedule for any fixed vector K is possible;
2. An original characterization of the set $\mathcal{K}$ of dominant periodicity vectors.

Next open question is: how to choose elements from $\mathcal{K}$ to get a fast and accurate evaluation of the maximum throughput of $\mathcal{G}$?