Back to Basics: Homogeneous Representations of Multi-Rate Synchronous Dataflow Graphs

Robert de Groote, Philip Hölzenspies, Jan Kuper, and Hajo Broersma

UNIVERSITY OF TWENTE.

Computer Architectures for Embedded Systems Group
Dept. of Electrical Engineering, Mathematics and Computer Science
University of Twente, Enschede, The Netherlands
http://caes.ewi.utwente.nl

MEMOCODE 2013
Multi-Rate Synchronous Dataflow Graphs (1/3)

- Capture task graphs
- Potential parallelism and interactions explicit
- Well suited for modelling DSP applications
- Annotations for analysis
Rates, auto-concurrency
Consistency, Iteration, Periodicity
Homogeneous, Cyclo-Static, Scenario-Aware, ...
Multi-Rate Synchronous Dataflow Graphs (3/3)

Throughput Analysis
- (Average) number of graph iterations per time unit
- Find critical cycle

Buffer Analysis
- Determine buffer capacities required for minimal throughput
- Make all cycles equally critical
Throughput Analysis

- Algorithms available for Homogeneous SDF Graphs (marked graphs)
- Transform MRSDF graph into HSDF graph
- Transformation described in [1], [2], ...


MRSDF to HSDF Transformation

- Represent individual firings in an iteration
MRSDF to HSDF Transformation

- Represent individual firings in an iteration
- Represent each token by a single edge
MRSDF to HSDF Transformation

- Represent individual firings in an iteration
- Represent each token by a single edge

Analysis: compute critical cycle (MCR)
Analysis: compute critical cycle (MCR)
Analysis: compute critical cycle (MCR)
Analysis: compute critical cycle (MCR)

Robert de Groote (University of Twente)
Back to Basics

10/18/2013 6 / 22
MRSDF Graphs - Exact Analysis (2/3)

Analysis: compute critical cycle (MCR)

Robert de Groote (University of Twente)
HSDF-based approach abandoned due to high complexity

- State-Space Exploration used instead [1]

HSDF-based approach abandoned due to high complexity

- State-Space Exploration used instead [1]

Exact Analysis: costly, but useful?

HSDF-based approach abandoned due to high complexity
  ▶ State-Space Exploration used instead [1]

Exact Analysis: costly, but useful?
  ▶ Only need guarantees

Construct linear bounds:

- Upper bound on token consumption times: \( \hat{\alpha}_c \)
- Lower bound on token production times: \( \check{\alpha}_p \)
Construct linear bounds:

- Upper bound on token consumption times: $\hat{\alpha}_c$
- Lower bound on token production times: $\tilde{\alpha}_p$
Translate each actor and channel into an edge \((i, j)\):

- \(\gamma\): Transfer rate ratio
- \(\epsilon\): Rate-independent delay
- \(\delta\): Rate-dependent delay

Translate each actor and channel into an edge \((i,j)\):

- \(\gamma\): Transfer rate ratio
- \(\epsilon\): Rate-independent delay
- \(\delta\): Rate-dependent delay
- \(s\): Firing start time
- Compute maximum rate, \(r\)

Translate each actor and channel into an edge \((i, j)\):
- \(\gamma\): Transfer rate ratio
- \(\epsilon\): Rate-independent delay
- \(\delta\): Rate-dependent delay
- \(s\): Firing start time
- Compute maximum rate, \(r\)

\[
\begin{align*}
\maximize r \\
\text{s.t. } s(j) &\geq s(i) + \epsilon(i, j) + \frac{\delta(i,j)}{r(i)} \\
r(j) &= \gamma(i, j) \cdot r(i)
\end{align*}
\]

Existing exact analysis of MRSDF graphs

- Data-driven transformation into HSDF
- Redundancy in resulting HSDF

Existing approximate analysis

- No upper bound on rate - no sense of error
- Opaque solution from an LP
Existing exact analysis of MRSDF graphs
  ▶ Data-driven transformation into HSDF
  ▶ Redundancy in resulting HSDF

Existing approximate analysis
  ▶ No upper bound on rate - no sense of error
  ▶ Opaque solution from an LP

No common ground!
Status Quo on analysis:


Periodic timed synchronous systems

- Mathematics: Max-Plus algebra (constraints)
- HSDF Graph: Linear Shift-Invariant system
- MRSDF Graph: Linear Shift-varying system
$t_a(k) = t_c(k - 1) + 1$
$t_b(k) = t_a(k - 2) + 2$
$t_c(k) = t_d(k) + 3$
$t_d(k) = \max\{t_b(k), t_a(k), t_c(k)\} + 5$
\[
\begin{bmatrix}
t_a \\
t_b \\
t_c \\
t_d
\end{bmatrix}
(k) = \bigoplus_i A_i
\begin{bmatrix}
t_a \\
t_b \\
t_c \\
t_d
\end{bmatrix} \otimes (k - i)
\]
\[
\begin{bmatrix}
t_a \\
t_b \\
t_c \\
t_d \\
\end{bmatrix} (k_0 + k) = \begin{bmatrix}
t_a \\
t_b \\
t_c \\
t_d \\
\end{bmatrix}(k_0) + 9k
\]
Structural invariants:
  - Repetition vector, $q$
Structural invariants:

- Repetition vector, \( q \)
\[ c_v \cdot s_{uv} = p_v \cdot s_{vw} \]

\[ p_u \cdot q_u = c_v \cdot q_v \quad p_v \cdot q_v = c_w \cdot q_w \]

Structural invariants:

- Repetition vector, \( q \)
\[ c_v \cdot s_{uv} = p_v \cdot s_{vw} \]

\[ p_u \cdot q_u = c_v \cdot q_v \quad p_v \cdot q_v = c_w \cdot q_w \]

\[ N = p_u \cdot q_u \cdot s_{uv} = c_v \cdot q_v \cdot s_{uv} = p_v \cdot q_v \cdot s_{vw} = \ldots \]

**Structural invariants:**

- Repetition vector, \( q \)
- Normalisation vector, \( s \)
\[ c_v \cdot s_{uv} = p_v \cdot s_{vw} \]

\[ p_u \cdot q_u = c_v \cdot q_v \quad p_v \cdot q_v = c_w \cdot q_w \]

\[ \mathcal{N} = p_u \cdot q_u \cdot s_{uv} = c_v \cdot q_v \cdot s_{uv} = p_v \cdot q_v \cdot s_{vw} = \ldots \]

Structural invariants:

- Repetition vector, \( q \)
- Normalisation vector, \( s \)
- Normalised token count, \( \mathcal{N} \)
\[ t_v(k) = t_u(\ldots) + \tau \]
\[ t_v(k) = t_u(...k \cdot c - d...) + \tau \]
\[ t_v(k) = t_u \left( \left\lfloor \frac{k \cdot c - d}{p} \right\rfloor \right) + \tau \]
$t_v(k) = t_u \left( \left\lceil \frac{k \cdot c - d}{p} \right\rceil \right) + \tau$

$t_v(k + mq_v) = ...$
\[ t_v(k) = t_u \left( \left\lceil \frac{k \cdot c - d}{p} \right\rceil \right) + \tau \]

\[ t_v(k + mq_v) = t_u \left( \left\lceil \frac{(k + mq_v) \cdot c - d}{p} \right\rceil \right) + \tau \]
MRSDF Analysis - Exact Homogeneous Representations

\[ t_v(k) = t_u \left( \left\lceil \frac{k \cdot c - d}{p} \right\rceil \right) + \tau \]

\[ t_v(k + mq_v) = t_u \left( \left\lceil \frac{k \cdot c - d}{p} \right\rceil + mq_u \right) + \tau \]
\[ t_v(k) = t_u \left( \left\lfloor \frac{k \cdot c - d}{p} \right\rfloor \right) + \tau \]

\[ t_v(k + mq_v) = t_u \left( \left\lfloor \frac{k \cdot c - d}{p} \right\rfloor + mq_u \right) + \tau = t_u(k + mq_u - C) + \tau \]
MRSDF Analysis - Exact Homogeneous Representations

Robert de Groote (University of Twente)
\[ t_v(k) = t_u\left(\left\lceil \frac{k \cdot c - d}{p} \right\rceil \right) + \tau \]
MRSDF Analysis - Approx. Homogeneous Representations

\[ t_v(k) = t_u \left( \left\lceil \frac{k \cdot c - d}{p} \right\rceil \right) + \tau \]

Obtain shift-invariance by changing counting units
MRSDF Analysis - Approx. Homogeneous Representations

\[ s_{uv} \]

\[ t_v \left( \frac{k}{q_v} \right) = \]
MRSDF Analysis - Approx. Homogeneous Representations

\[ t_v(\kappa) = \]

\( u \rightarrow d \rightarrow c \rightarrow v, \tau \)

\[ S_{UV} \]
\[ t_v(\kappa) = t_u \left( \frac{1}{q_u} \left[ \frac{\kappa \cdot q_v \cdot c - d}{p} \right] \right) + \tau \]
MRDF Analysis - Approx. Homogeneous Representations

\[ S_{uv} \]

\[ t_v(\kappa) = t_u \left( \frac{1}{q_u} \left[ \frac{\kappa \cdot q_v \cdot c - d}{p} \right] \right) + \tau \]

\[ = t_u \left( \frac{1}{q_u} \left[ \frac{\kappa \cdot q_v \cdot c - d + p - 1}{p} \right] \right) + \tau \]
\[ \hat{t}_v(k) = \hat{t}_u(k - s_{uv} \cdot d) + \tau \]
\[ = \hat{t}_u(k - s_{uv} \cdot (d - p + 1)) + \tau \]
MRSDA Analysis - Approx. Homogeneous Representations

Pessimistic

Optimistic
MP3 Analysis - Example use case

MP3 decoder: $\tau = 1603621$, SRC: $\tau = 1320974$, DAC: $\tau = 5000$
Goal: Close the gap between exact and approximate analysis
Goal: Close the gap between exact and approximate analysis

- Critical Subgraph
Future Work - Towards Incremental Analysis

Goal: Close the gap between exact and approximate analysis

▶ Critical Subgraph
▶ Use bounds to zoom in on critical subgraph
Back to Basics!

- Gives us a \textit{natural} transformation from MRSDF into HSDF...
Back to Basics!

- Gives us a *natural* transformation from MRSDF into HSDF...
- ...from which we can derive bounding HSDF graphs
Conclusions

Back to Basics!

- Gives us a *natural* transformation from MRSDF into HSDF...
- ...from which we can derive bounding HSDF graphs

Properties:

- Buffer weights direct further optimization
Back to Basics!

- Gives us a *natural* transformation from MRSDF into HSDF...
- ...from which we can derive bounding HSDF graphs

Properties:
- Buffer weights direct further optimization
- Approximation gets better for large repetition vectors (= large HSDF graphs)
Conclusions

Back to Basics!

- Gives us a *natural* transformation from MRSDF into HSDF...
- ...from which we can derive bounding HSDF graphs

Properties:

- Buffer weights direct further optimization
- Approximation gets better for large repetition vectors (= large HSDF graphs)
- Perfectly suited to balance analysis accuracy and runtime
Questions ?

robert.degroote@utwente.nl