

Modular Code Generation

- Code generated for a composite block independently from used contexts:

SDF Director

This model shows a simple periodogram spectral estimate of a modulated sinusoid in noise. The top-level parameters control the carrier frequency, the signal frequency, and the noise level. Notice that the two peaks are centered at the carrier frequency, with their distance from the carrier given by the signal frequency. The sample rate is assumed to be 8kHz.

The blocks with red outlines are hierarchical.
 -Right click or Command-click (Mac) and select "Open Actor".
 -These generate sinusoids, one for the signal and the other for the carrier
 -signal and the other for the carrier

- carrierFrequency: 2000.0
- signalFrequency: 500.0
- noiseStandardDeviation: 0.1

The Expression₂ block calculates a mathematical expression, as shown.

Select "Run Window" from the View menu to execute the model,
 -or click on the red triangle in the toolbar.
 -Try changing the parameters in the run window or on the diagram.

Author: Edward A. Lee

SDF Director

This model illustrates the MultInstanceComposite actor, where it is used to compute a subsection of the discrete Fourier transform (DFT) using the Goertzel kernel. The parameter NDFTCoefficients specifies the number of bins to calculate.

- SinePeriod: 8.0
- NDFTCoefficients: 10
- DFTBlockLength: 64
- FirstDFTCoefficient: roundToInt(DFTBlockLength/SinePeriod-NDFTCoefficients/2)

Author: Zoltan Kemenczy (Research In Motion)

SDF Director

Generate a sine wave.

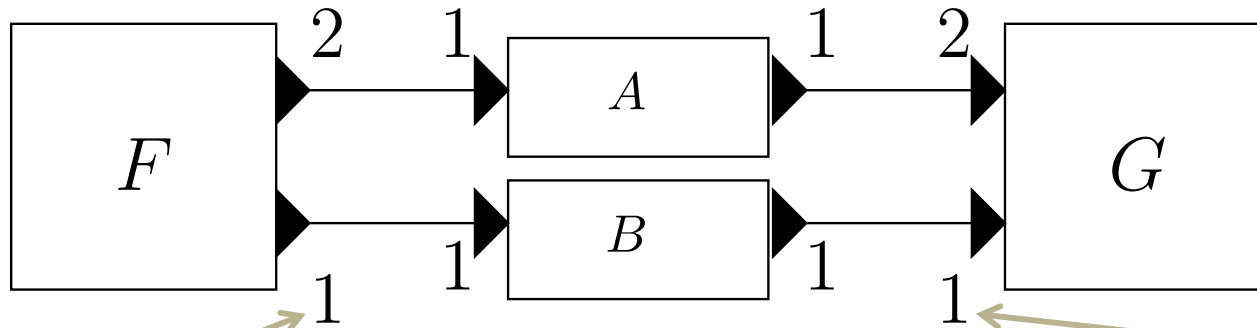
[Documentation](#)

- frequency: 440.0
- phase: 0.0

Motivations

- Reusability
 - Incremental compilation
 - IP protection
- Modularity
 - Unit verification and testing
 - Parallelization
 - Scalability
- Reduction of runtime overhead
 - Speeding up simulations

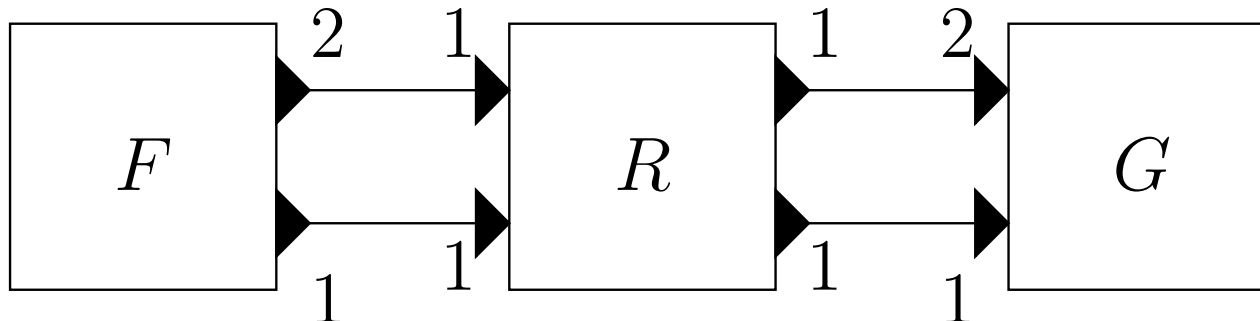
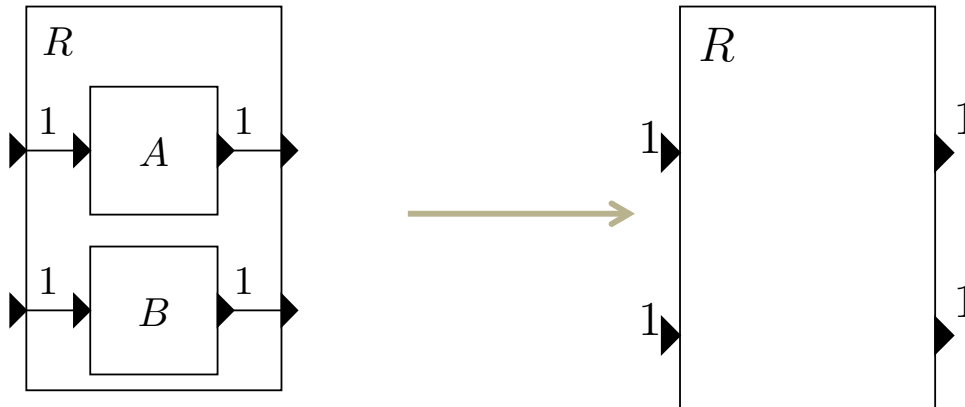
Naïve SDF Code Generation



Finite-buffer
schedule:FAABG

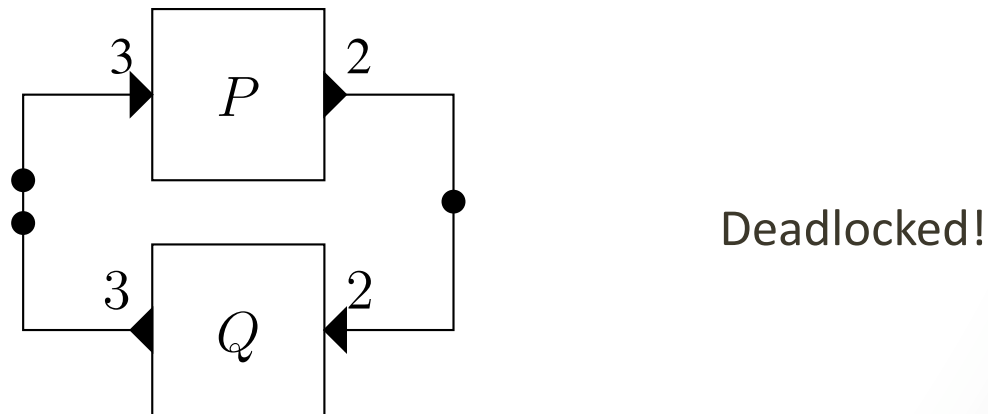
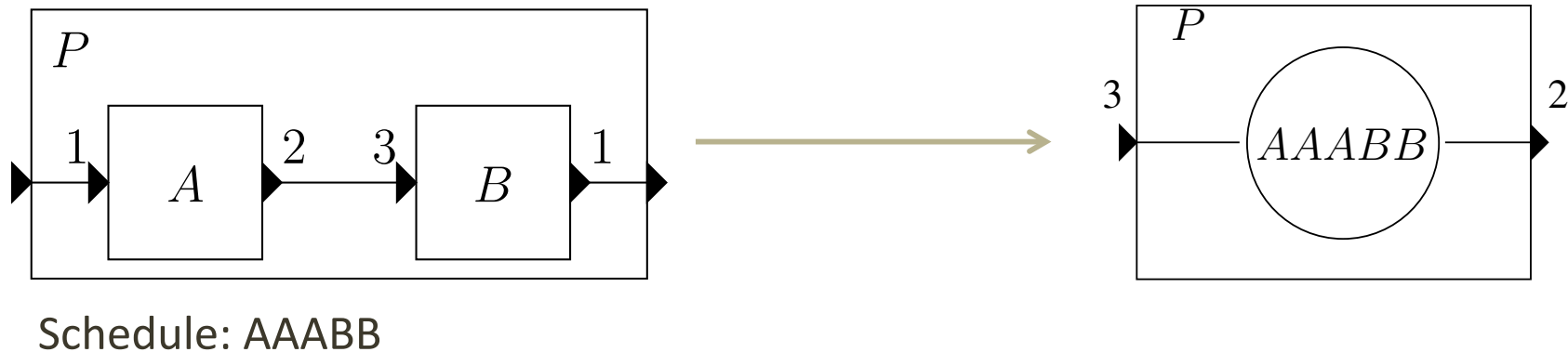
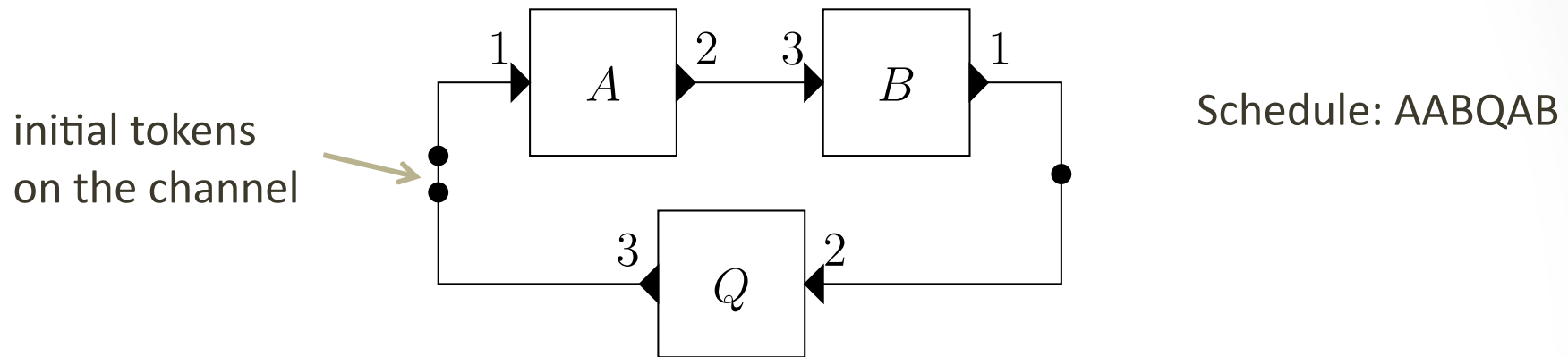
tokens consumed at
the port each time
the actor fires

tokens produced at
the port each time
the actor fires



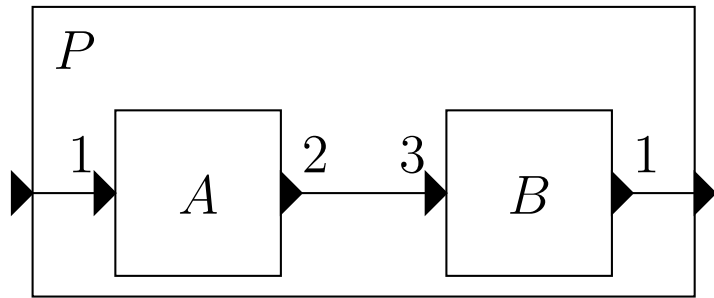
No finite-buffer
schedule!

Naïve SDF Code Generation

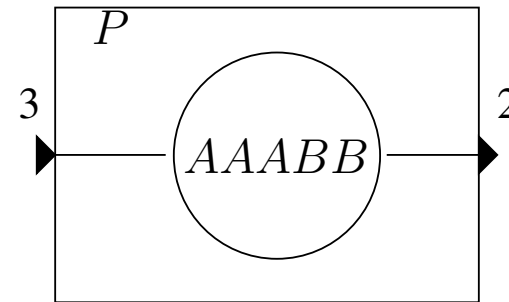


Modular SDF Code Generation

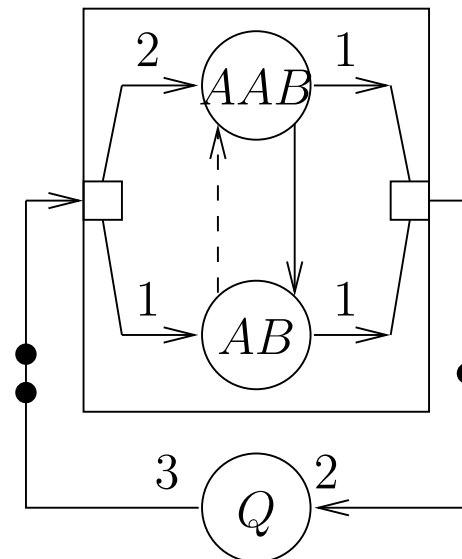
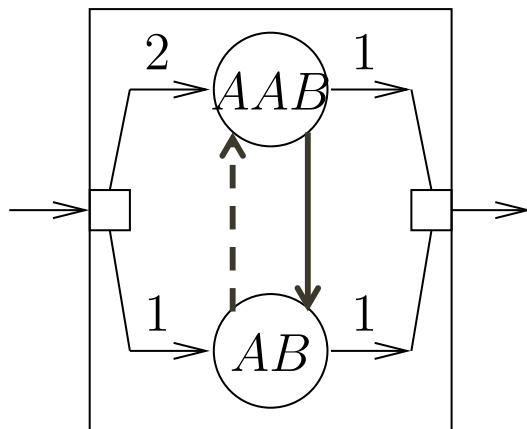
- Non-monolithic firing function
- DSSF = Deterministic SDF with Shared FIFOs



✗
monolithic



Non-monolithic How do we synthesize?



(AAB) (Q) (AB)

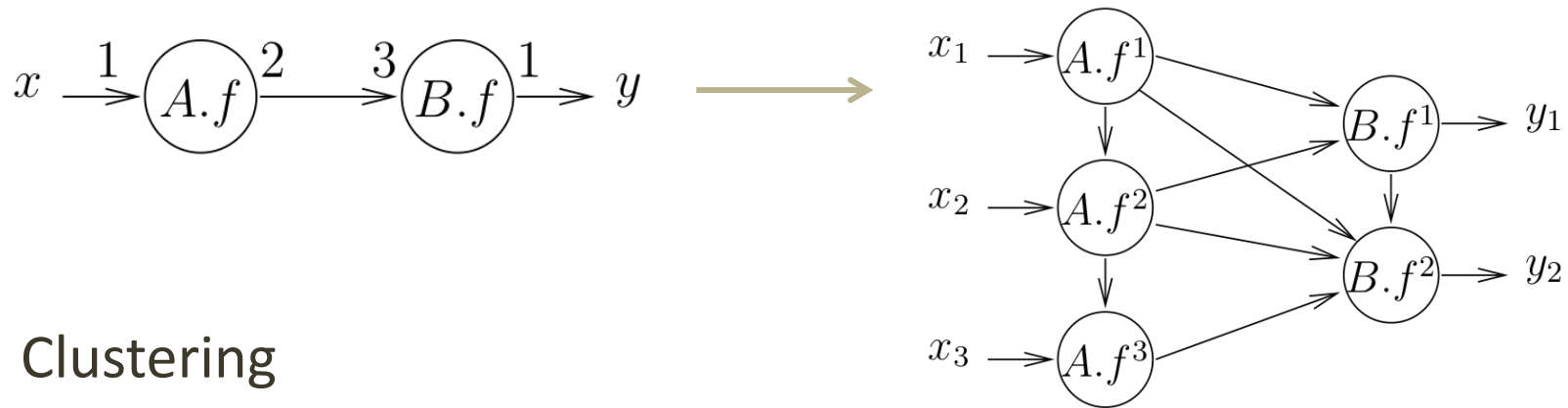
Can we do

(AAB)(Q)(AAB)

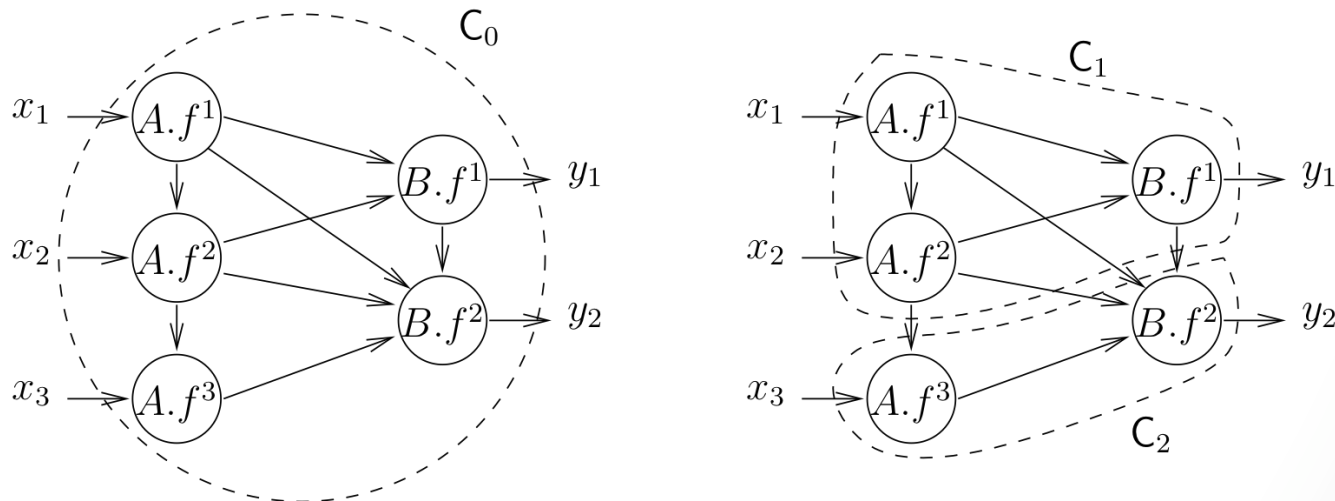
?

Synthesis

- Unfolding (non-homogeneous to homogeneous)

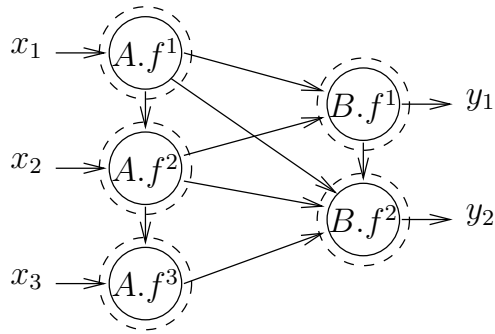


- Clustering

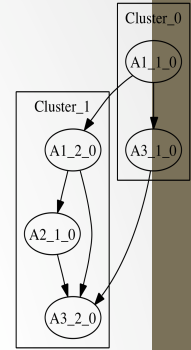
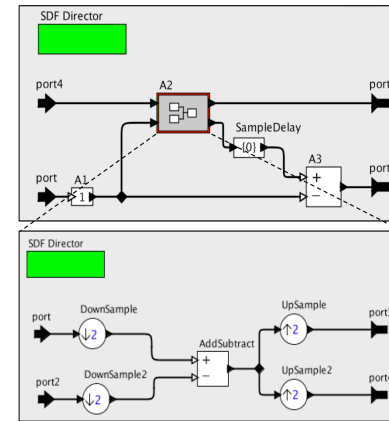
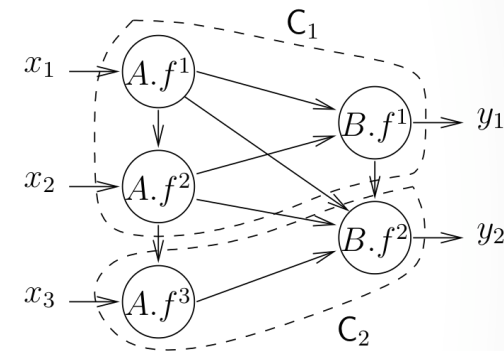


Clustering Algorithm Effectiveness

Valid clustering
yet bad!



Better
clustering



Test cases	# ins/outs	# actors	# nodes in unfolding graph	# clusters
Test	2/2	3	5	2
Entropy	1/1	15	1545	1
CD to DAT	1/1	4	156	15

Conclusions

- Hierarchical SDF models are not compositional
- Introduce DSSF profiles as a compositional representation of composite actors and show how this representation can be used for modular code generation
- Propose a synthesis algorithm that can handle hierarchical models of arbitrary depth

Future Work

- Implement more advanced clustering algorithms to reduce the number of clusters
- Understand the relations between the number of clusters, the size of generated code and performance
- Estimate throughput and delay of a code-generated model

References

- Stavros Tripakis, Dai Bui, Marc Geilen, Bert Rodiers, Edward A. Lee, “Compositionality in Synchronous Data Flow: Modular Code Generation from Hierarchical SDF Graphs”. To appear at ACM Transactions on Embedded Computing Systems (TECS)
- Roberto Lubliner, Christian Szegedy, Stavros Tripakis, “Modular code generation from synchronous block diagrams: modularity vs. code size”, POPL ‘09 Proceedings of the 36th annual ACM SIGPLAN-SIGACT symposium on Principles of programming languages Proceedings of the 36th annual ACM SIGPLAN-SIGACT symposium on Principles of programming languages.
- Joachim Falk, Joachim Keinert, Christian Haubelt, Jürgen Teich, Shuvra S. Bhattacharyya, “A Generalized Static Data Flow Clustering Algorithm for MPSoC Scheduling of Multimedia Applications”, EMSOFT ‘08 Proceedings of the 8th ACM International Conference on Embedded Software.
- Joseph Buck, “Scheduling Dynamic Dataflow Graphs with Bounded Memory Using the Token Flow Model”, PhD Thesis, EECS Department, University of California, Berkeley, 1993.

Thank you