

Resource Allocation for Hardware Implementations of Map

Richard Townsend

Martha A. Kim

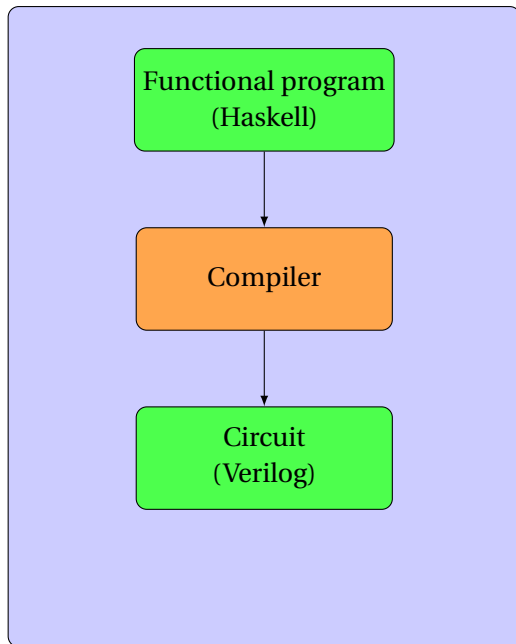
Stephen A. Edwards

Columbia University

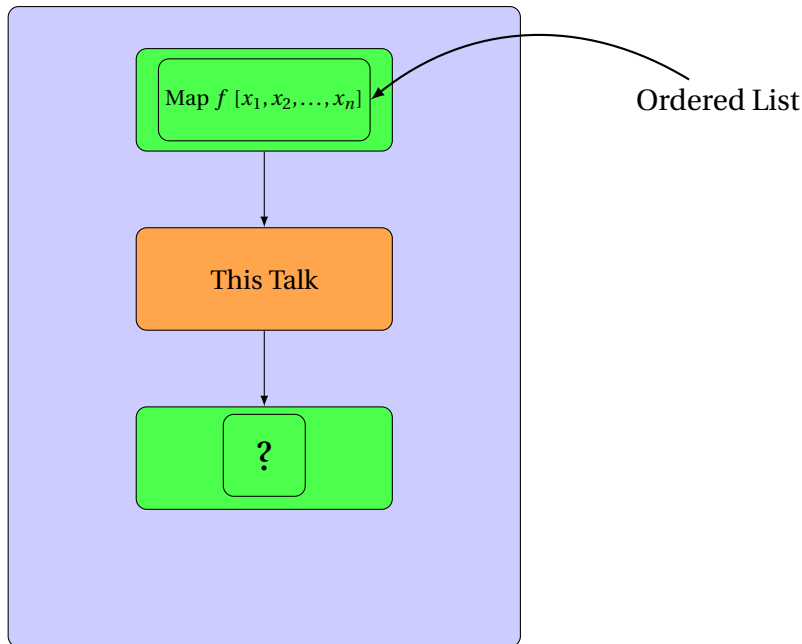
ASBD Workshop, June 15, 2014



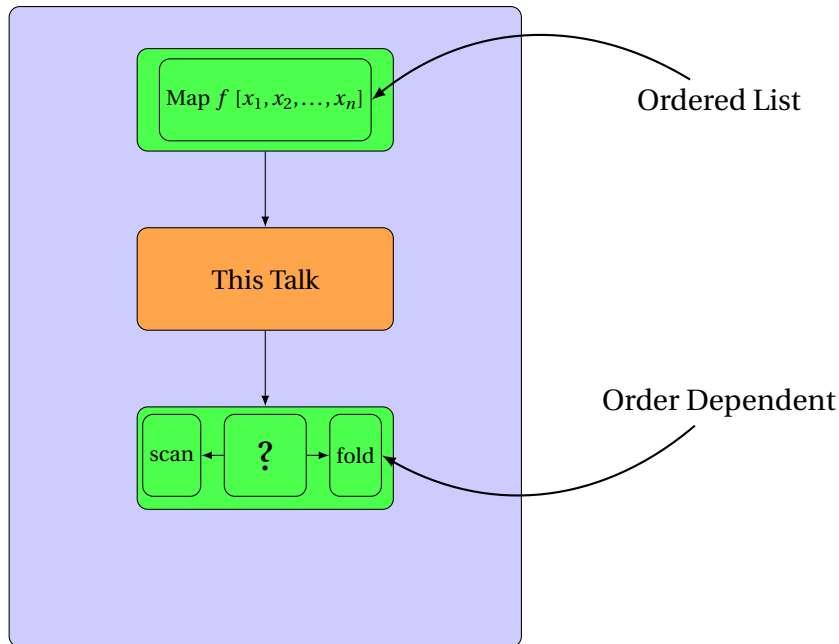
Functional Programs to Functional Hardware



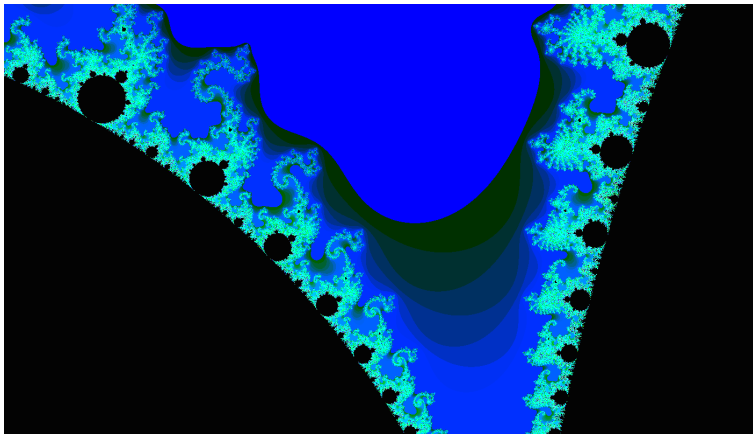
Functional Programs to Functional Hardware



Functional Programs to Functional Hardware



Functional Map vs. MapReduce



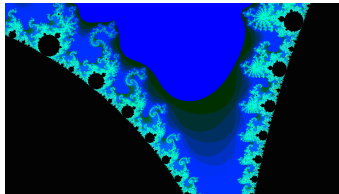
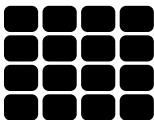
Functional Map vs. MapReduce

(0,3)	(1,3)	(2,3)	(3,3)
(0,2)	(1,2)	(2,2)	(3,2)
(0,1)	(1,1)	(2,1)	(3,1)
(0,0)	(1,0)	(2,0)	(3,0)

Functional Map vs. MapReduce

(0,3)	(1,3)	(2,3)	(3,3)
(0,2)	(1,2)	(2,2)	(3,2)
(0,1)	(1,1)	(2,1)	(3,1)
(0,0)	(1,0)	(2,0)	(3,0)

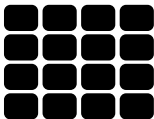
Ordered



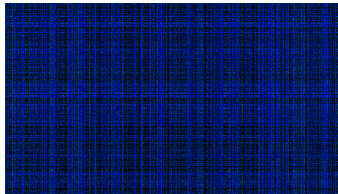
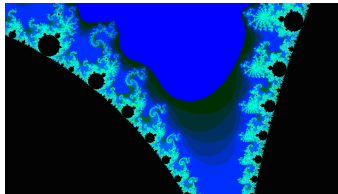
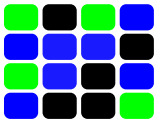
Functional Map vs. MapReduce

(0,3)	(1,3)	(2,3)	(3,3)
(0,2)	(1,2)	(2,2)	(3,2)
(0,1)	(1,1)	(2,1)	(3,1)
(0,0)	(1,0)	(2,0)	(3,0)

Ordered



Unordered



Structure of a Hardware Implementation



f

Structure of a Hardware Implementation

f

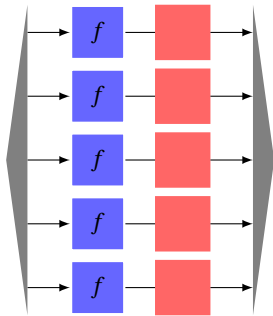
f

f

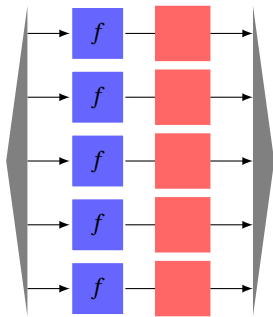
f

f

Structure of a Hardware Implementation

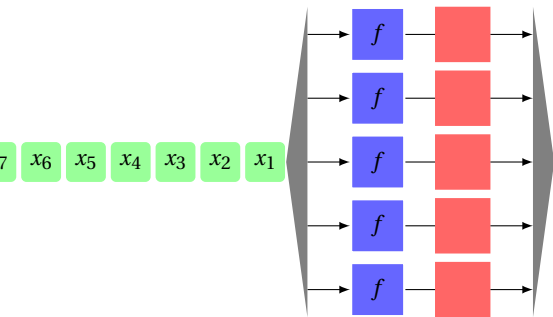


Structure of a Hardware Implementation

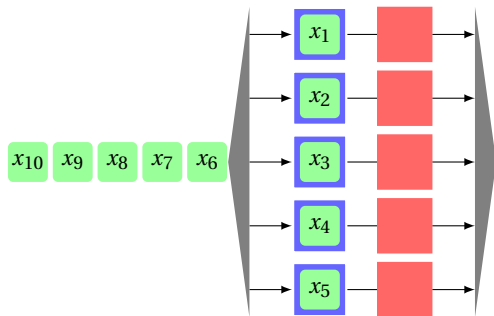


x_1 x_2 x_3 x_4 x_5 x_6 x_7 x_8 x_9 x_{10}

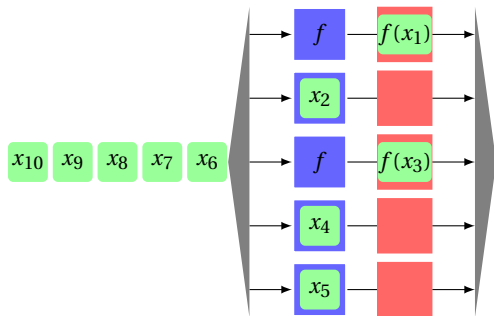
Structure of a Hardware Implementation



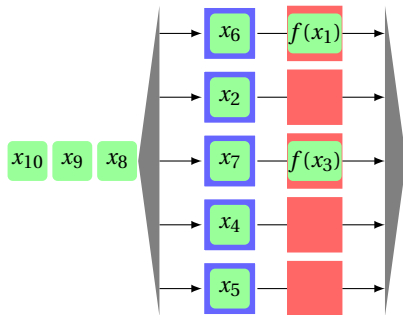
Structure of a Hardware Implementation



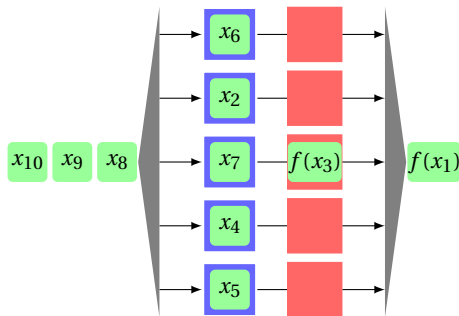
Structure of a Hardware Implementation



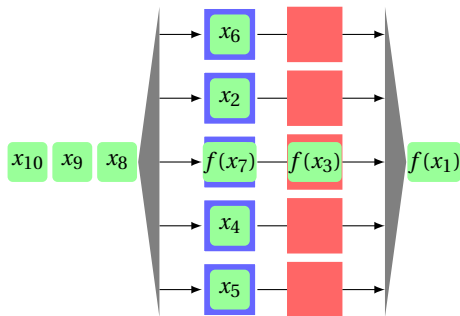
Structure of a Hardware Implementation



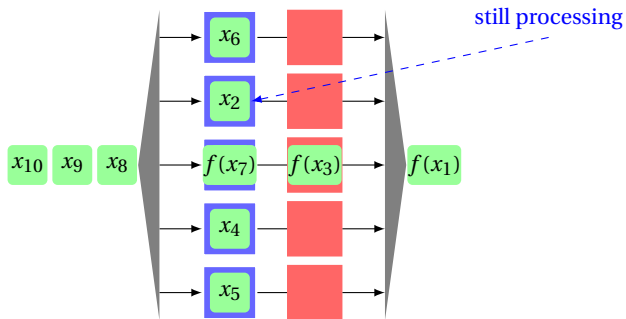
Structure of a Hardware Implementation



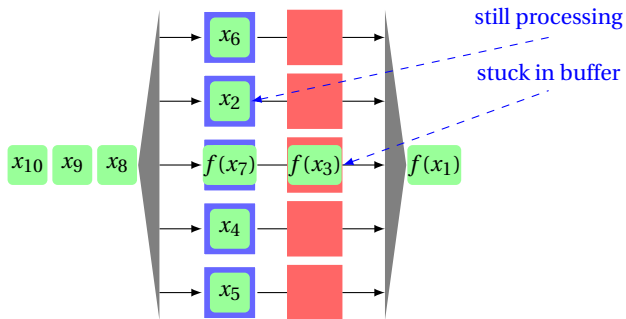
Structure of a Hardware Implementation



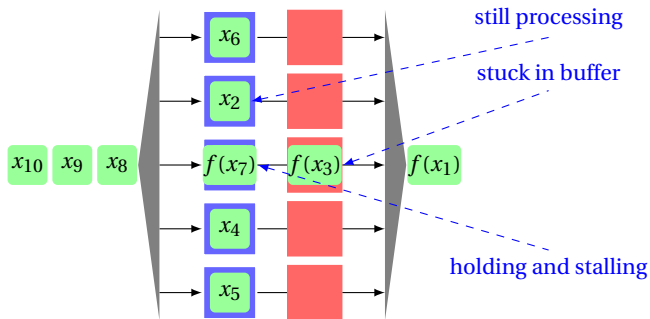
Structure of a Hardware Implementation



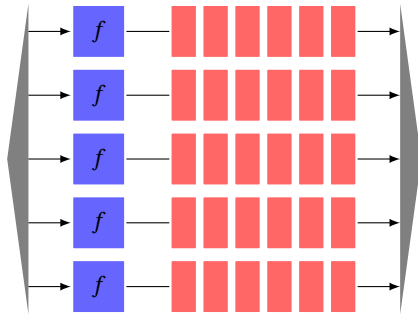
Structure of a Hardware Implementation



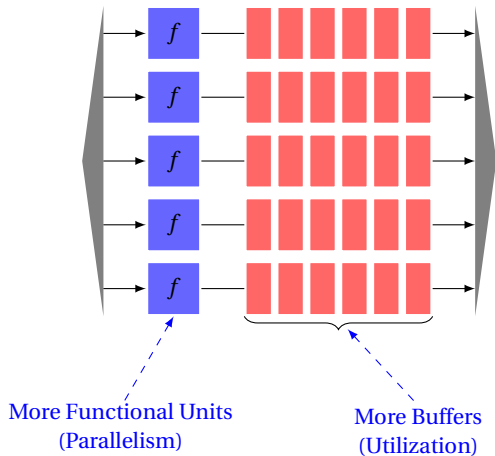
Structure of a Hardware Implementation



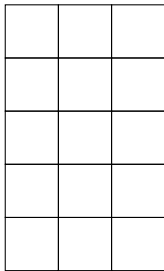
Structure of a Hardware Implementation



Structure of a Hardware Implementation

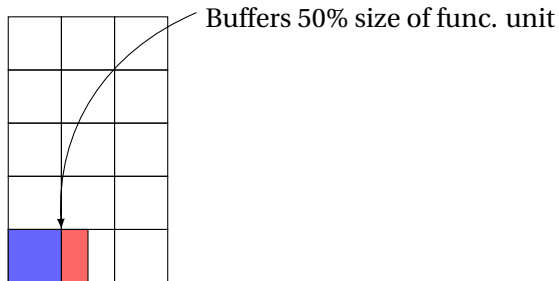


Multiple Possible Configurations...Which to Choose?



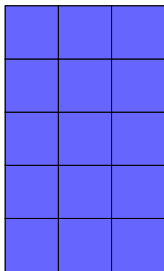
Area = 15

Multiple Possible Configurations...Which to Choose?



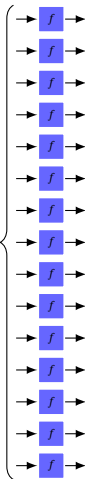
Area = 15

Multiple Possible Configurations...Which to Choose?

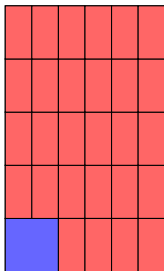


Area = 15

15 Functional Units



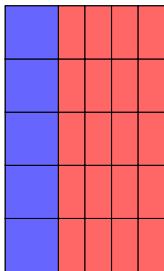
Multiple Possible Configurations...Which to Choose?



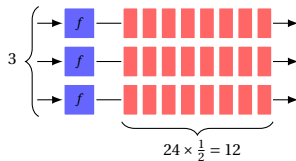
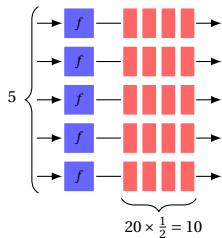
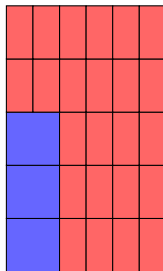
Area = 15



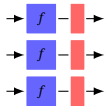
Multiple Possible Configurations...Which to Choose?



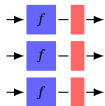
Area = 15



Workload Structure



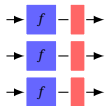
Workload Structure



Best-case



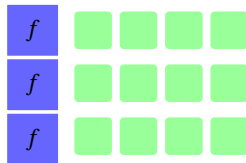
Workload Structure



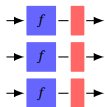
Best-case



Time →



Workload Structure



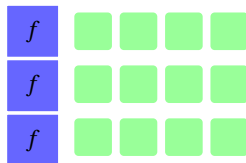
Best-case



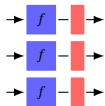
Average-case

?

Time →



Workload Structure



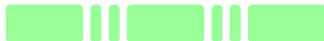
Best-case



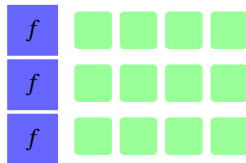
Average-case

?

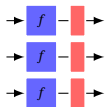
Worst-case



Time



Workload Structure



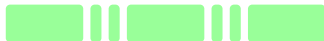
Best-case



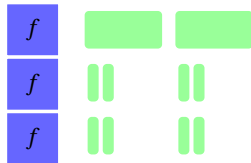
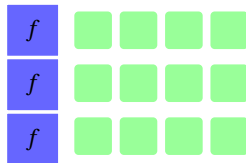
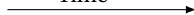
Average-case

?

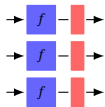
Worst-case



Time



Workload Structure



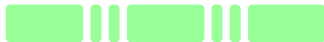
Best-case



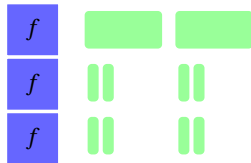
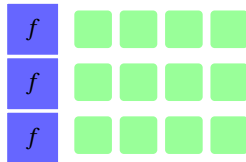
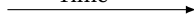
Average-case

?

Worst-case

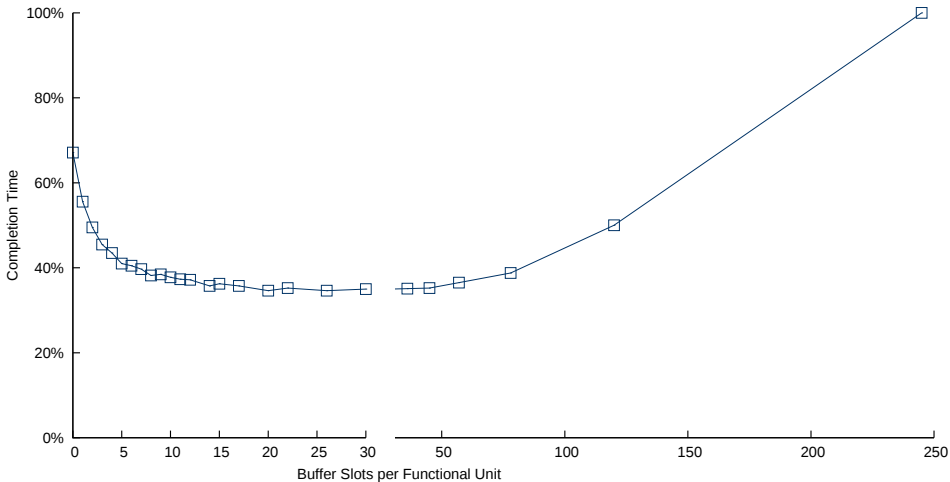


Time



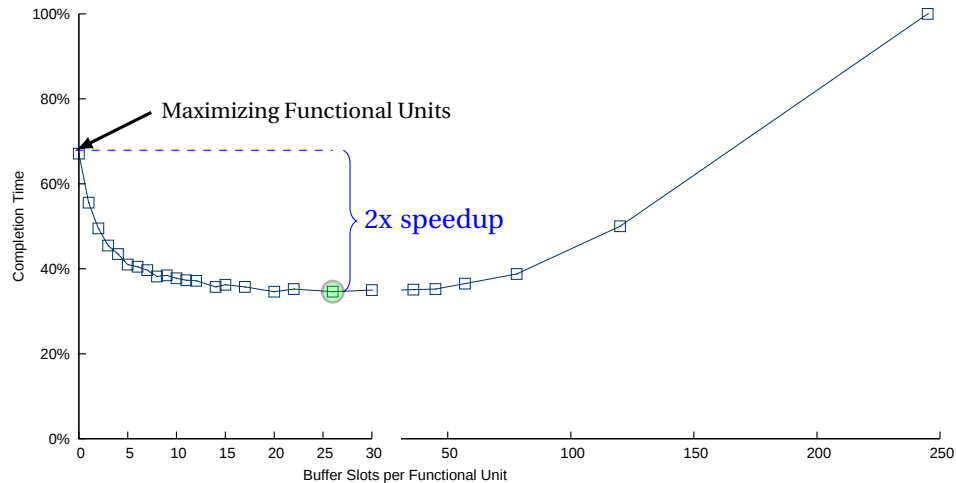
Optimal Resource Allocation

Simulator Results



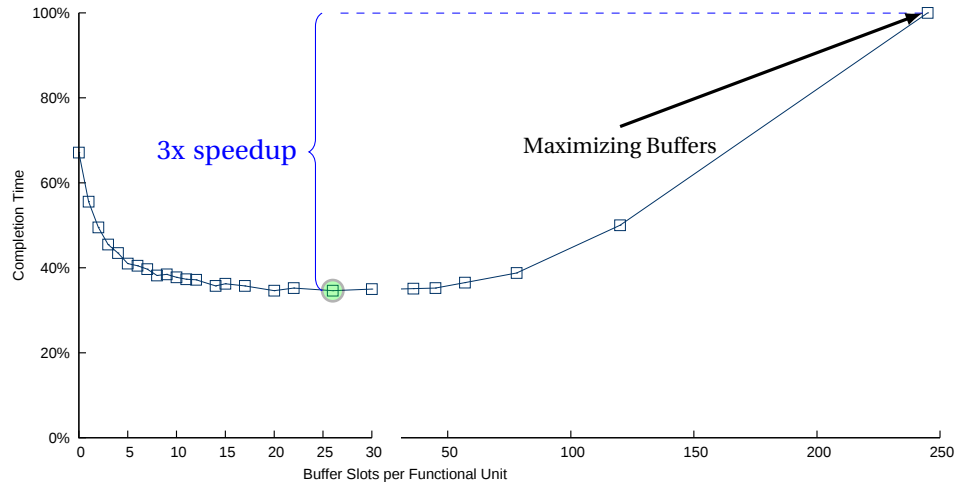
Optimal Resource Allocation

Simulator Results

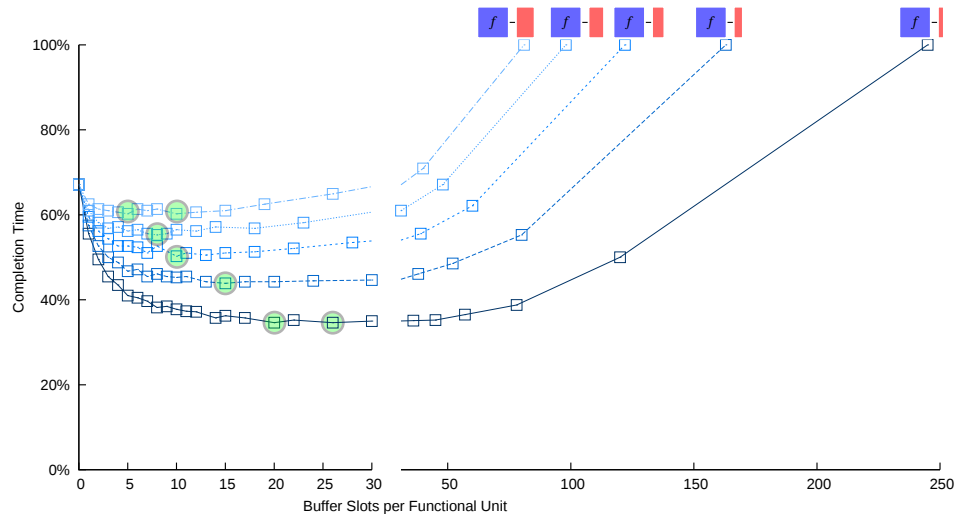


Optimal Resource Allocation

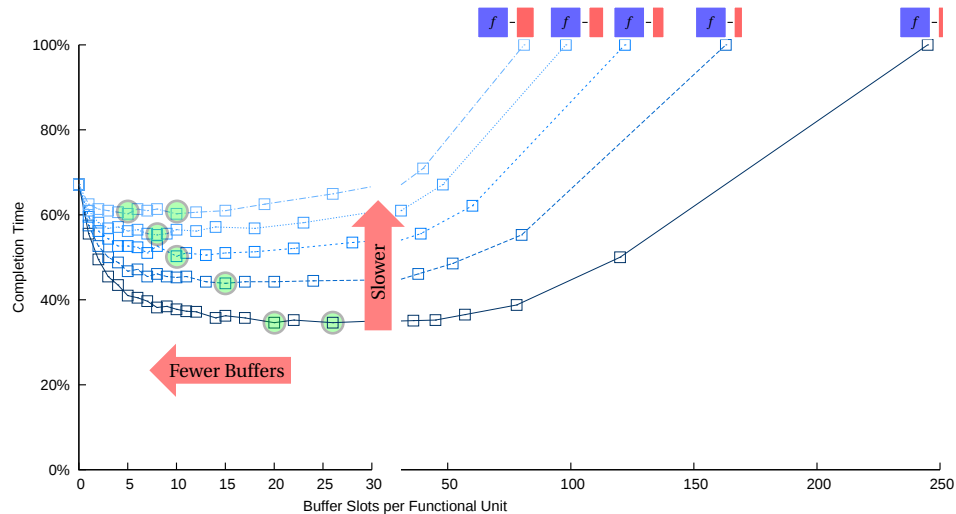
Simulator Results



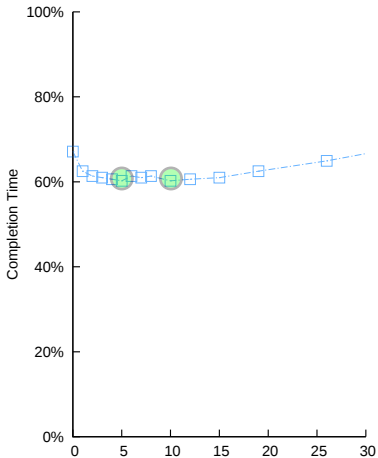
Optimal Resource Allocation



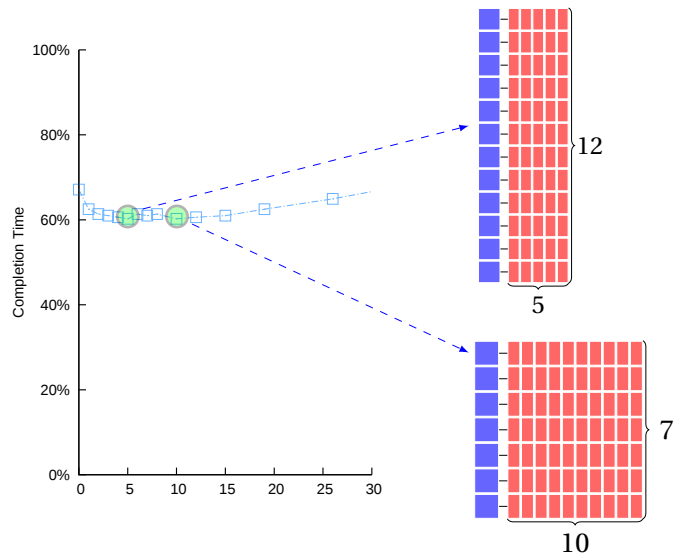
Optimal Resource Allocation



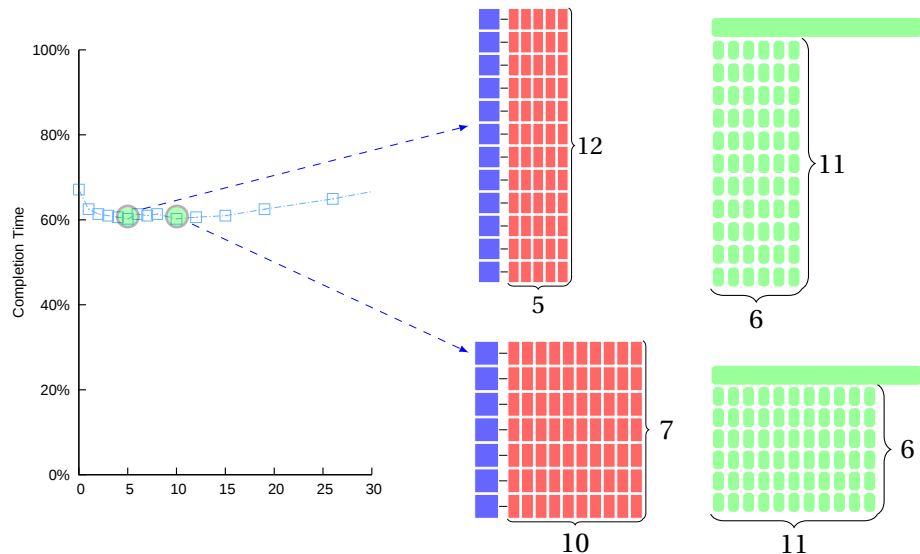
Why Are There Multiple Optima?



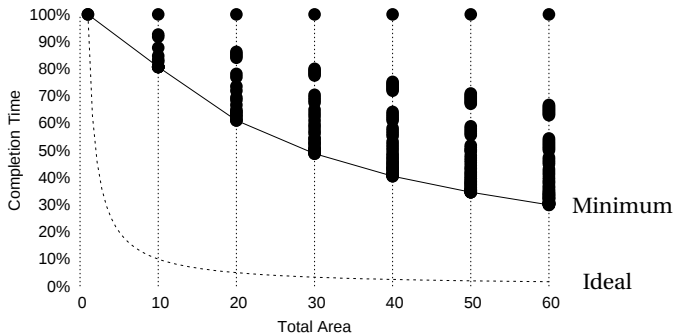
Why Are There Multiple Optima?



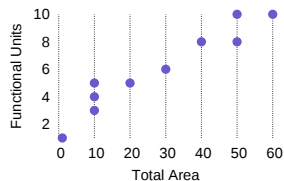
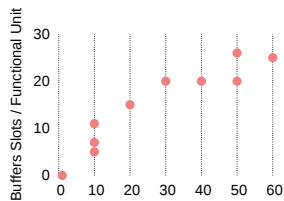
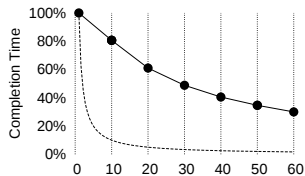
Why Are There Multiple Optima?



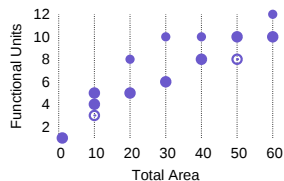
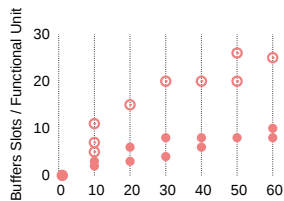
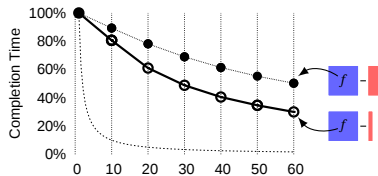
Performance Scales with Area



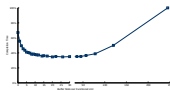
Performance Scales with Area



Performance Scales with Area

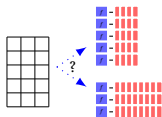
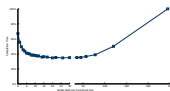


Conclusions



Area trade-off is important...

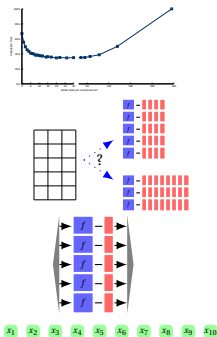
Conclusions



Area trade-off is important...

...and non-obvious

Conclusions

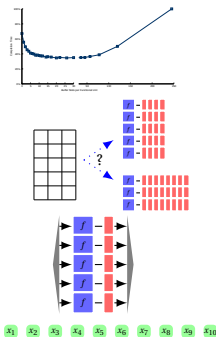


Area trade-off is important...

...and non-obvious

Model helps explore design space

Conclusions



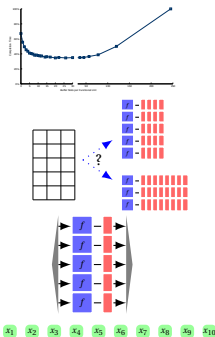
Area trade-off is important...

...and non-obvious

Model helps explore design space

Synthesize Efficient Hardware Implementation of Map

Conclusions



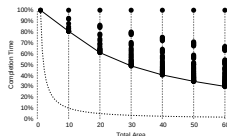
Area trade-off is important...

...and non-obvious

Model helps explore design space

Synthesize Efficient Hardware Implementation of Map

Scan Map Fold



Enhance our abstraction