EECS Electrical Engineering and Computer Sciences

BERKELEY PAR LAB

PARALLEL COMPUTING LABORATORY

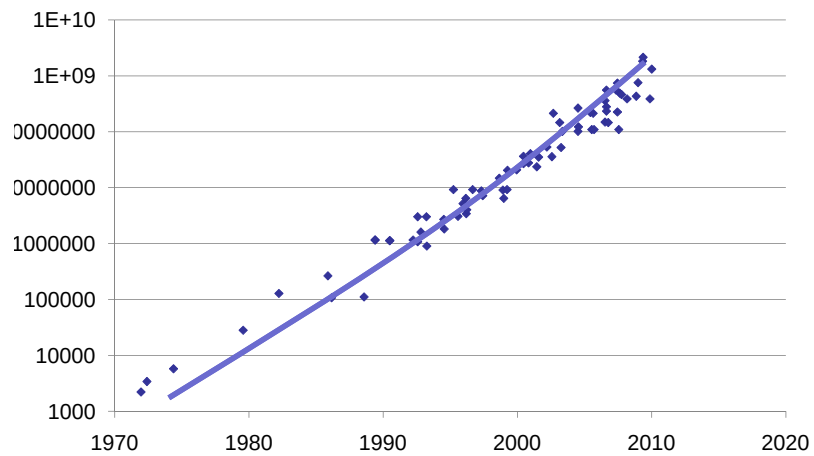
The Future of Microprocessor-based Computation (is Highly Parallel)

Kurt Keutzer
and the PALLAS Team
Michael Anderson, Bryan Catanzaro, Jike Chong,
Katya Gonina, Dorothea Kolossa, Chao-Yue Lai,
Mark Murphy, Bor-Yiing Su, Narayanan Sundaram
UC Berkeley

EECS Electrical Engineering and Computer Sciences

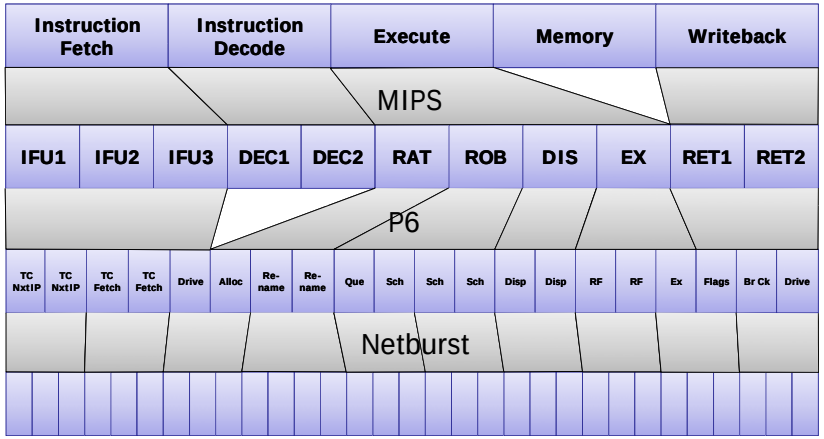
Moore's Law: Transistor Count

BERKELEY PAR LAB



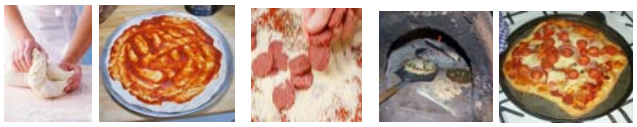
2

Pipeline Evolution



Prescott

What's wrong with that?



5 stage pizza pipeline – 1 pizza/minute

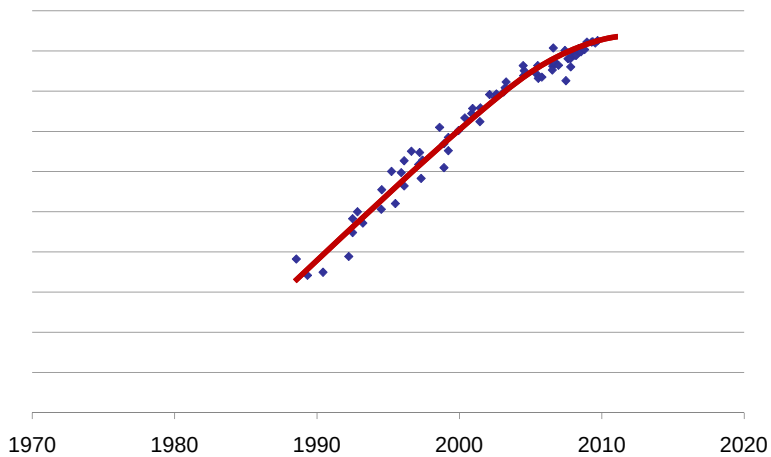
10 stage pizza pipeline – 1 pizza 30/seconds

20 stage pizza pipeline – 1 pizza/20 seconds

http://jansdough.janktheproofer.com/Make-Pizza-tutorial.htm

→31 stage pizza pipeline – too much heat in the kitchen

Results: Pipelined Sequential Performance



5

Alternative: Parallelism

10 stage pizza pipeline – 1 pizza 30/seconds

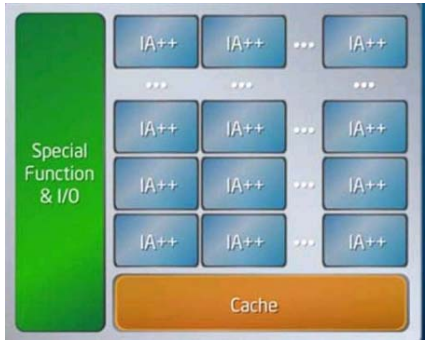
two10 stage pizza pipelines – 1 pizza 15/seconds

four10 stage pizza pipelines – 1 pizza 7.5/second???

6

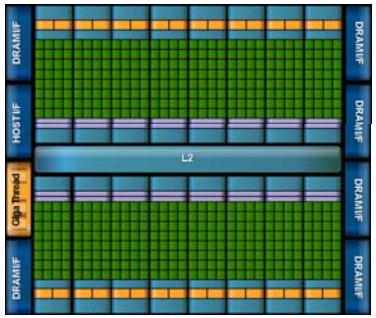
Parallelism in HW

- Intel: Larrabee



32 processors
each 16-wide vector unit

- Nvidia: Fermi

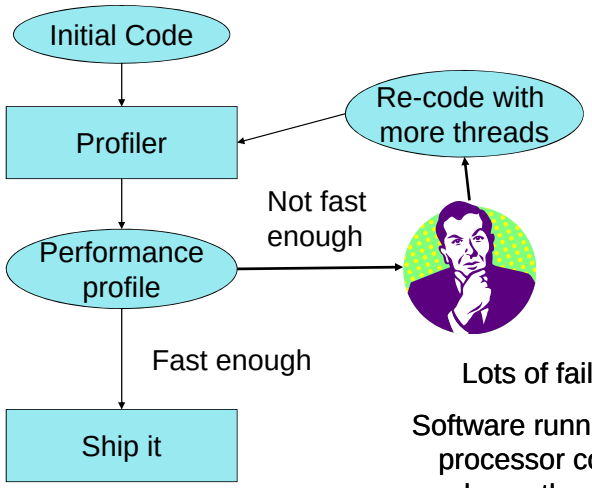


16 processors
each 32-wide vector unit

We can build them … but …

- No question we can build these processors
- Big question: How do we program 32 – 512 processors to solve your favorite application?
  - Perform content-based image retrieval
  - Large-vocabulary speech recognition
  - MRI image reconstruction
  - Value-at-risk analysis in quantitative finance
  - Etc.
- Parallel processors are more efficient, if we can program them.
- So … the future of microprocessors is really about software.
- That's why the rest of the talk is all about programming applications on parallel microprocessors

How NOT to parallel program

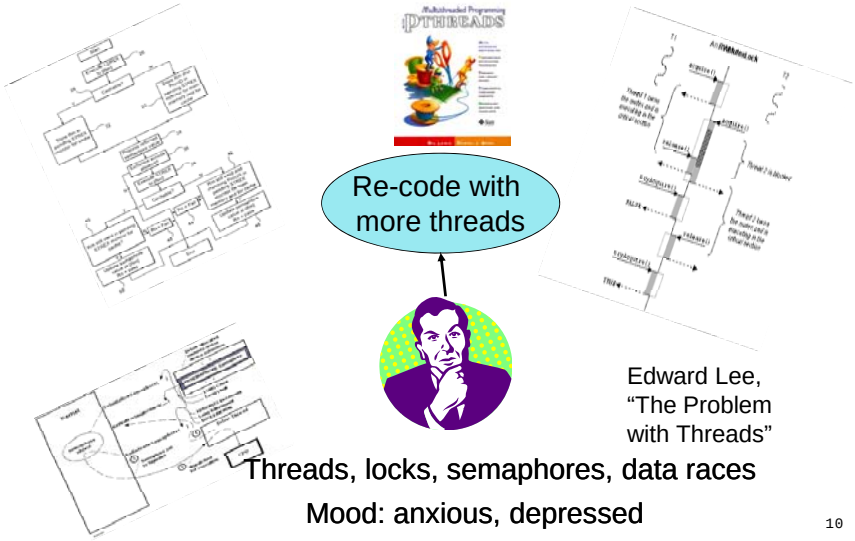


Lots of failures

Software running on N processor cores is slower than on 1

9

What is this person thinking of?

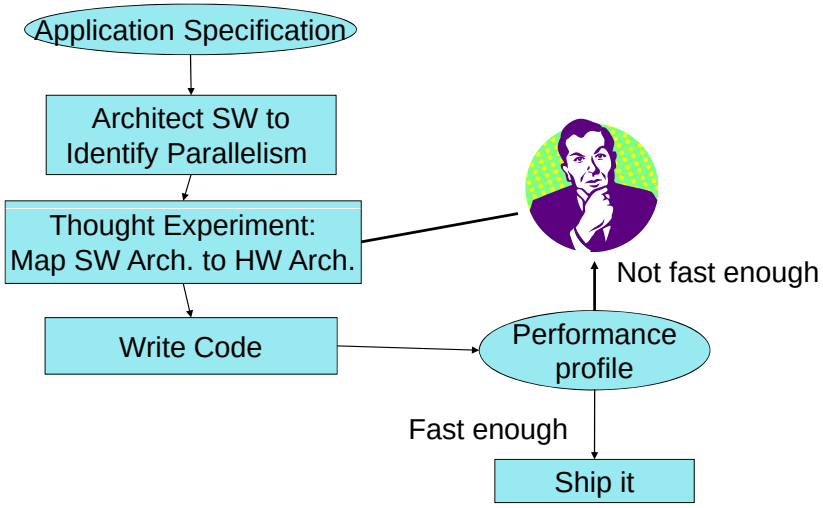


Threads, locks, semaphores, data races

Mood: anxious, depressed

10

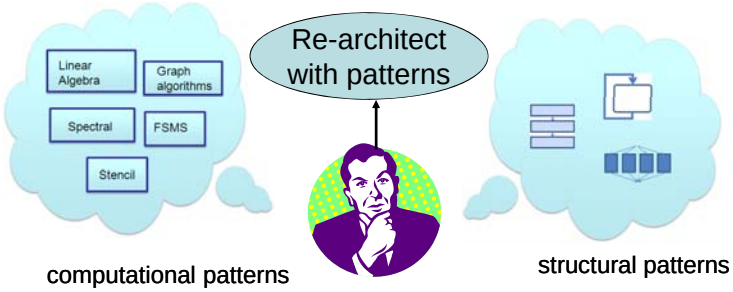
Our PALLAS Methodology



11

What is this person thinking of?

- Today's code is serial but the world is parallel
  - we just need to find the parallelism in our applications and reflect it in our software

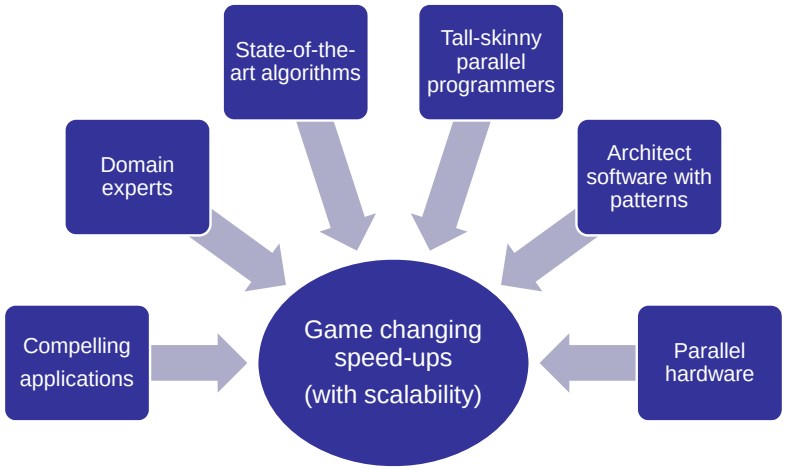


Software architecture

Mood: well-adjusted, optimistic

12

Our formula



13

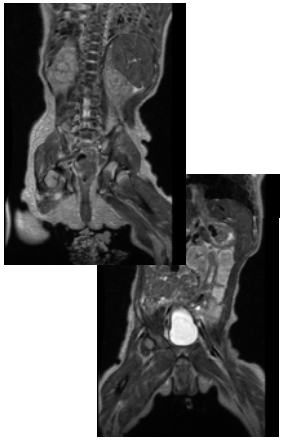
Applications

- Our formula in a bit of detail: MRI reconstruction
- Other applications where our formula has proven itself
- Applications in progress

14

Compelling Application: Fast, Robust Pediatric MRI

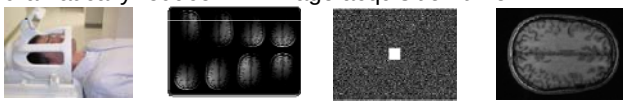
- Pediatric MRI is difficult:
  - Children cannot sit still, breathhold
  - Low tolerance for long exams
  - Anesthesia is costly and risky
- Like to accelerate MRI acquisition
  - Advanced MRI techniques exist, but require data- and compute- intense algorithms for image reconstruction
- Reconstruction must be fast, or time saved in accelerated acquisition is lost in computing reconstruction
  - Non-starter for clinical use



15

Domain Experts and State-of-the-Art Algorithms

- Collaboration with MRI Researchers:
  - Miki Lustig, Ph.D., Berkeley EECS
  - Marc Alley, Ph.D., Stanford EE
  - Shreyas Vasanawala, M.D./Ph.D., Stanford Radiology
- Advanced MRI: Parallel Imaging and Compressed Sensing to dramatically reduce MRI image acquisition time

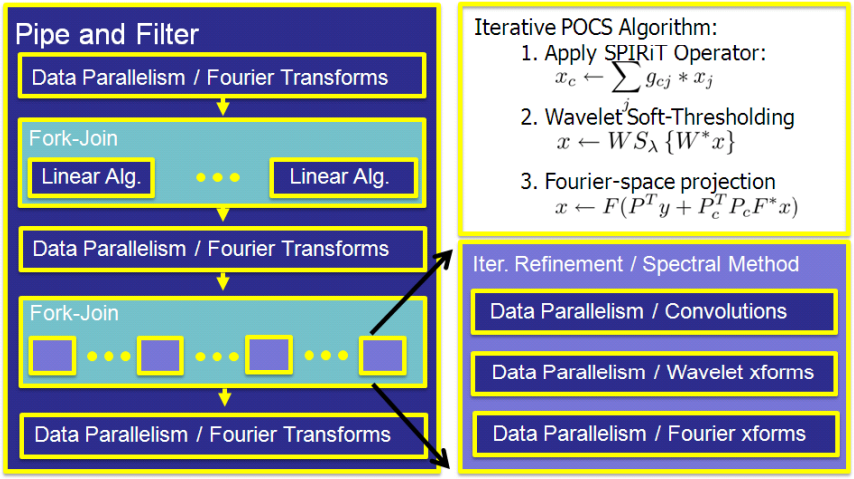


- Computational IOU: Must solve constrained L1 minimization

$$\begin{aligned} \text{minimize} \quad & ||Wx||_1 \\ \text{s.t} \quad & \mathbf{F}_\Omega x = y, \\ & ||\mathbf{G}x - x||_2 < \varepsilon \end{aligned}$$

16

SW architecture of image reconstruction



17

Game-Changing Speedup

- 100X faster reconstruction
- Higher-quality, faster MRI
- This image: 8 month-old patient with cancerous mass in liver
  - 256 x 84 x 154 x 8 data size
  - Serial Recon: 1 hour
  - Parallel Recon: 1 minute
- Fast enough for clinical use
  - Software currently deployed at Lucile Packard Children's Hospital for clinical study of the reconstruction technique

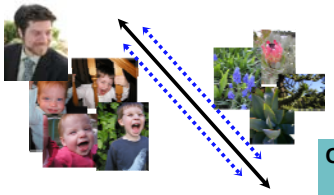
18

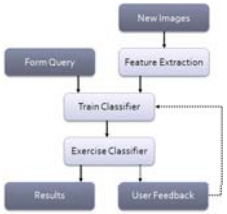
Applications

- Our formula in a bit of detail: MRI reconstruction
- Other applications where our formula has proven itself
- Applications in progress

19

Support-Vector Machine Mini-Framework





- Algorithmic changes and parallel implementation lead to performance speedup: Core 2 Duo versus G80

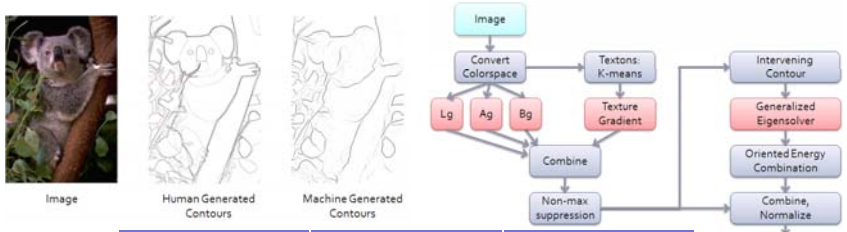
Computation	LIBSVM	Our algorithm 2-core Parallel CPU	Our algorithm, 16-core Parallel GPU
SVM Training (geo-mean)	771.8 s	---	38.5 s
SVM Classification	41.42 s	4.21 s	0.38 s

100X speed-up
793 downloads since release in 10/2008

Fast support vector machine training and classification, Catanzaro, Sundaram, Keutzer, International Conference on Machine Learning 2008

20

Image Contour Detection Mini-Framework

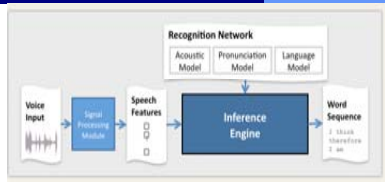


Original Serial Code (Nehalem)	C + Pthreads, our algorithms 8 threads, 2 sockets (Barcelona)	Our algorithms, GTX280
222 seconds	29.79 seconds	1.8 seconds

130X speed-up
490 downloads since release in 6/2009

"Efficient, High-Quality Image Contour Detection" Catanzaro, Su, Sundaram, Lee, Murphy, Keutzer International Conference on Computer Vision, 2009

Speech Recognition Framework



- Input: Speech audio waveform
- Output: Recognized word sequences

- Achieved 11x speedup over sequential version
- Allows 3.5x faster than real time recognition

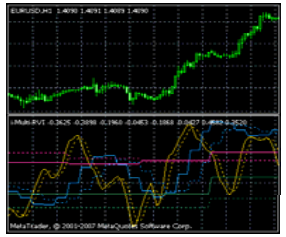
- Our technique is being deployed in a hotline call-center data analytics company
- Used to search content, track service quality and provide early detection of service issues

Scalable HMM based Inference Engine in Large Vocabulary Continuous Speech Recognition, Kisun You, Jike Chong, Youngmin Yi, Ekaterina Gonina, Christopher Hughes, Wonyong Sung and Kurt Keutzer, IEEE Signal Processing Magazine, March 2010

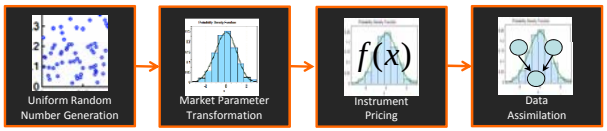
22

Computational Finance

- **Value-at-Risk Computation** with **Monte Carlo Method**
- Summarizes a portfolio's vulnerabilities to market movements
- Important to algorithmic trading, derivative usage and highly leveraged hedge funds
- Improved implementation to run **60x faster** on a parallel microprocessor



Four Steps of Monte Carlo Method in Finance



Matthew Dixon, Jike Chong, Kurt Keutzer, "Acceleration of Market Value-at-Risk Estimation", Workshop on High Performance Computing in Finance at Super Computing 2009, November 15, 2009.

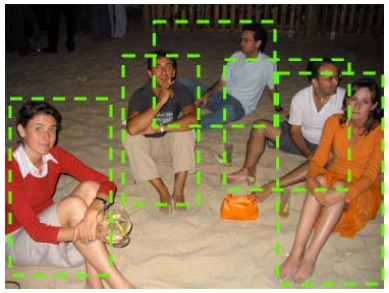
Applications

- Our formula in a bit of detail: MRI reconstruction
- Other applications where our formula has proven itself
- Applications in progress

24

Poselet Human Detection

- Can locate humans in images
- **20x speedup** through algorithmic improvements and parallel implementation



- Work can be extended to *pose estimation* for controller-free video game interfaces using ordinary web cameras

25

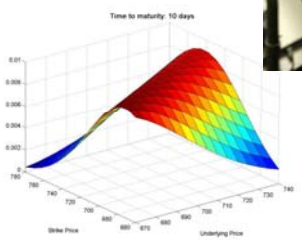
Option Pricing Application

- Price an **option -** a tradable financial security whose value depends on the value of an underlying asset and market parameters.
- Black-Sholes equation:

$$\frac{\partial V}{\partial t} + \frac{1}{2}\sigma^2 S^2 \frac{\partial^2 V}{\partial S^2} + rS\frac{\partial V}{\partial S} - rV = 0.$$

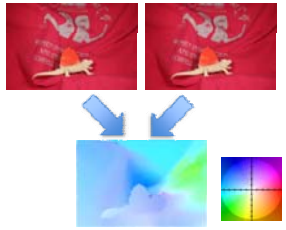
Speedup:

6x pricing 1 option
25x pricing 128 options on Larrabee



26

Optical Flow

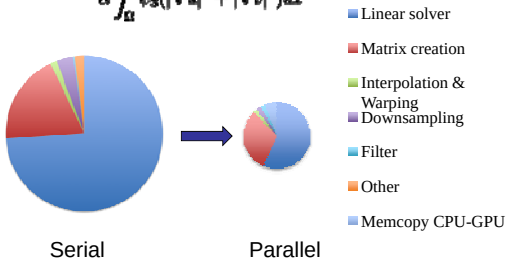


- Optical Flow involves computing the motion vectors ("flow field") between the consecutive frames of a video
- Involves solving a non-linear optimization problem

$$J(\mathbf{w}) = \int_\Omega \psi_1(|I(x+w) - I(x)|^2)dx + \gamma \int_\Omega \psi_2(|\nabla I(x+w) - \nabla I(x)|^2)dx + \alpha \int_\Omega \psi_3(|\nabla u|^2 + |\nabla v|^2)dx$$

Speedup

32x linear solver
7x overall



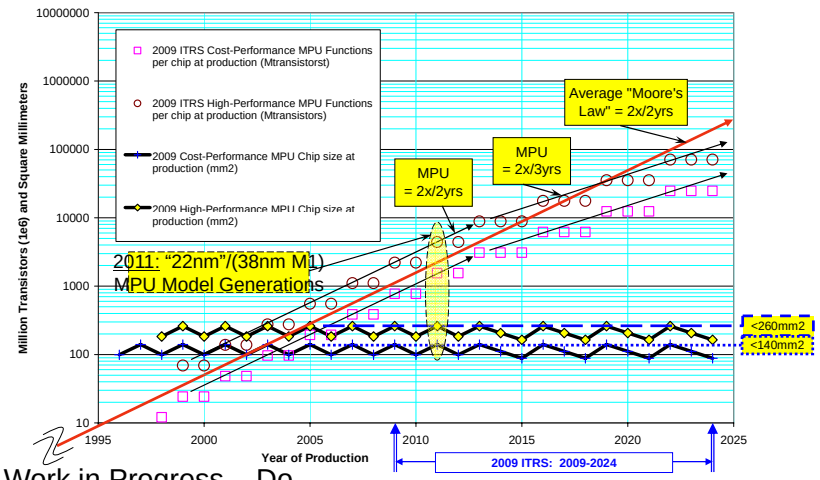
27

ITRS Roadmap

Including '11 snapshot Analysis

Figure 9b

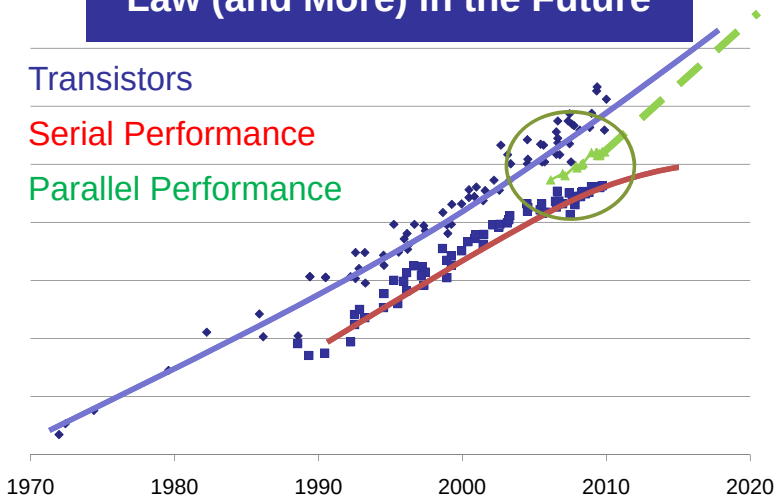
Logic



Work in Progress – Do Not Publish!

28

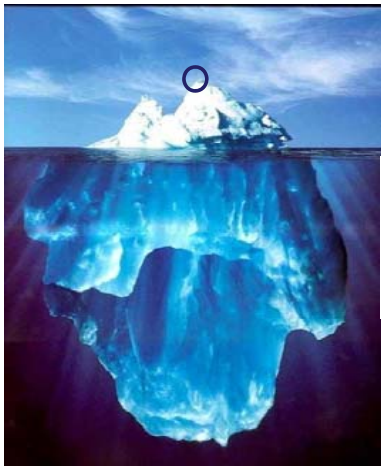
Parallelism Delivers Moore's Law (and More) in the Future



29

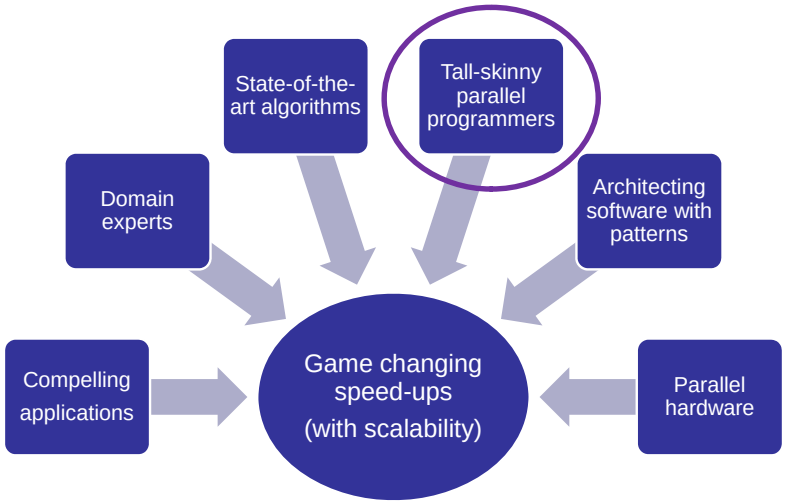
Tip of the Iceberg

- Today we're only exploiting the smallest tip of the iceberg of computation that will be available in the future



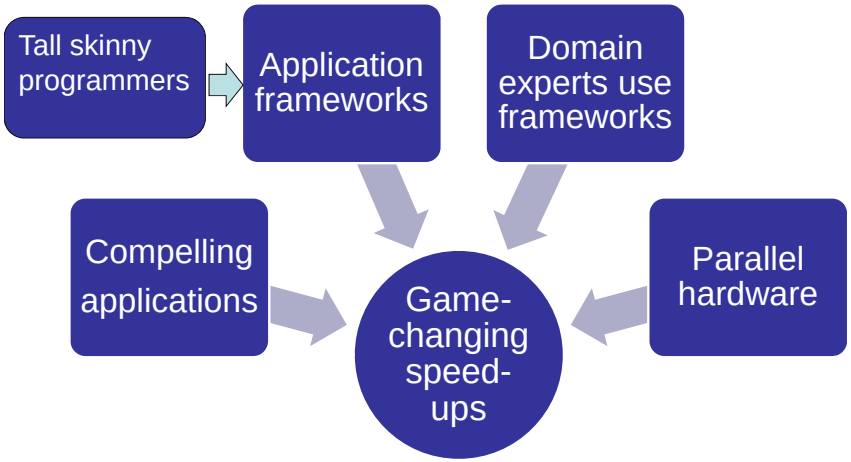
30

Limits of our formula



31

Our research formula



32

How do we deliver all those new applications to the user?

- I've talked about how to translate Moore's Law into useful applications via microprocessors.
- Our next speakers will talk about how to deliver those applications to you.

- The Future of Mobile
  - Eric Brewer
- The Future of the Cloud
  - Michael Franklin

33

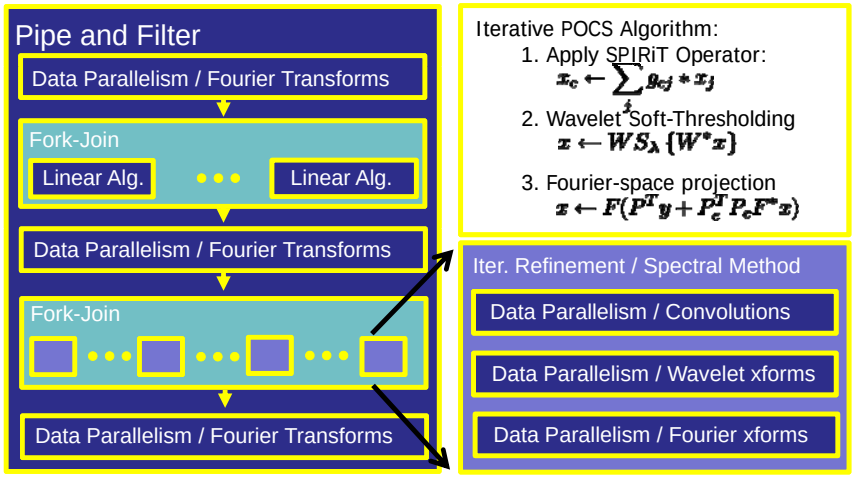
Backup

34

EECS
Electrical Engineering and Computer Sciences

SW architecture of reconstruction

BERKELEY PAR LAB



Iterative POCS Algorithm:

1. Apply SPIRiT Operator:
   $x_c \leftarrow \sum_j g_{cj} * x_j$
2. Wavelet Soft-Thresholding
   $x \leftarrow W S_\lambda \{W^* x\}$
3. Fourier-space projection
   $x \leftarrow F(P^T y + P_c^T P_c F^* x)$

Iter. Refinement / Spectral Method

- Data Parallelism / Convolutions
- Data Parallelism / Wavelet xforms
- Data Parallelism / Fourier xforms

35