XPS Project Report
Automatically Scalable Computation (ASC)

Jonathan Appavoo, Steve Homer, & Ajay Joshi Boston University
Tommy Unger, Tomislav Petrovic, & Schuyler Eldridge
Margo Seltzer, Ryan P. Adams, & David Brooks Harvard University
Amos Waterland, & Yu (Emma) Wang
Scaling SW Performance as function of a physical parameter
A FUNDAMENTAL QUESTION

Can we scale software performance as a function of system size (e.g. transistor count — not necessarily cpus)?
Exposing and Exploiting Structure in Execution

Execution as n bit "video" signal of system state

One column of pixels for each bit of system state (register, ram and memory-mapped I/O devices)

One row of pixels is a frame of execution and describes entire system state at a particular step of instruction execution

Execution "video" proceeds from top to bottom
Intuition
An Approach
Making things tractable
A Simple Example

• Search for a prime factor of a number
• in this case 9223371994482243049

Amos Waterland

/* factor.c */
#include <stdio.h>
#include <stdlib.h>

static long invert(long n)
{
    register long p;
    for (p = 3; p * p <= n; p += 2)
        if ((n % p) == 0)
            return p;
    return n;
}

int main(int argc, char *argv[])
{
    long p, n = 9223371994482243049;
    if (argc > 1)
        n = atol(argv[1]);
    p = invert(n);
    if (p)
        printf("p = %ld\n", p);
    return p;
}
$ time ./factor > out
$ for ((i=0;i<10;i++)); do time ./factor > out; done
$ time ./asc factor > out
$ time ./asc -a 40013f -i 23068684 factor > out
$ for ((i=0;i<10;i++)); do time ./asc -a 40013f -i 23068684 factor.net factor > out; done
$ ./asc factor.* > out.train & sleep 30; kill %1
./asc:main.c:233: batch training `factor.net' on `factor.train' ...
$ time ./asc -a 40013f -i 23068684 factor.net factor > out
Status and Directions

- **OS**: Linux ASC Processes — ASC exec flag, Address Space Management, “Fat” Binary
- **Theory**: Limits, Hardness and Complexity implications
- **Compiler**: Language Level Hyperplanes, Approximation Opportunities and state space profile information
- **HW**: Neuromorphic Acceleration, Hyperplane Support, State profiling, State Cache Acceleration