Mixed-Precision Floating-Point Adaptation

Overview
- Floating-point arithmetic suffers from rounding error
- Tradeoff between faster single precision (32 bit) and more accurate double precision (64 bit)
- We have built a prototype system to auto-tune the precision level of particular components of a program
- Initial results with the Algebraic MultiGrid kernel demonstrate a nearly 2X speedup

Background
- **Mixed precision**: execute a few crucial instructions in double precision and the rest in single precision

```
1: LU ← PA
2: solve Ly=Pb
3: solve Ux_k = y
4: for k = 1,2,... do
5:    rk ← b−Ax_k−1 (*)
6:    solve Ly = Pr_k
7:    solve Uz_k = y
8:    x_k ← x_k−1 + z_k (*)
9:    convergence check
10: end for
```

Methods
- Replace double-precision instructions and values with single-precision versions
- **Basic block patching** (binary modification)
- **Mixed-precision configuration representation**

For more details, see "Automatically Adapting Programs for Mixed-Precision Floating-Point Computation" in the SC ’12 poster session.