

Compiler/Run-Time Framework for Dynamic Data-Flow Parallelization of Tiled Programs

Martin Kong¹ Antoniu Pop² R. Govindarajan³
Louis-Noël Pouchet¹ Albert Cohen⁴ P. Sadayappan¹

¹ The Ohio State University

² The University of Manchester

³ Indian Institute of Science

⁴ Inria

January 19th, 2015

IMPACT 2015

5th International Workshop on Polyhedral Compilation Techniques
Amsterdam, Netherlands



Motivating Example: Blur-Roberts

Focus of this work: removal of data-parallel barriers executed on shared-memory multi-core machines

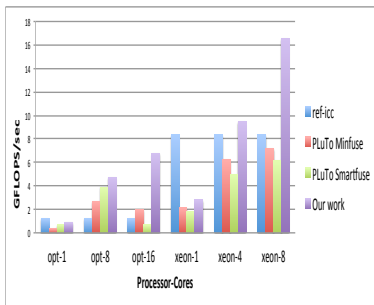
```

for (i = 1; i < N - 1; i++)
  for (j = 1; j < N - 1; j++)
S1: B[i][j] = (A[i][j] + A[i][j-1] +
             A[i][1+j] + A[i+1][j] +
             A[i-1][j] + A[i-1][j-1] +
             A[i-1][j+1] + A[i+1][j-1] +
             A[i+1][j+1])/8.0;

for (i = 1; i < N-2; i++)
  for (j = 2; j < N-1; j++)
S2: A[i][j] = abs(B[i][j]-B[i+1][j-1]) +
             abs(B[i+1][j] - B[i][j-1]);

```

- ▶ Barrier involve global consensus
- ▶ Number of synchronizations depend upon: program structure and applied transformations
- ▶ Some transformations could derive on loss of locality or parallelism



Blur-Roberts kernel performance in GFLOPS/sec for AMD Opteron 6274 (16 cores) and Intel Xeon E5-2650 (8 cores), on 1, half and all cores.

Tiled Blur-Roberts

Blur-Roberts tiled with PLuTo using the Minfuse heuristic (maximal decomposition)

```

if (_PB_N >= 3) {
  lbp=0;
  ubp=floor(_PB_N-2,32);
  #pragma omp parallel for private(lbv,ubv)
  for (t2=lbp;t2<=ubp;t2++)
    for (t3=0;t3<=floor(_PB_N-2,32);t3++)
      for (t4=max(1,32*t2);t4<=min(_PB_N-2,32*t2+31);t4++) {
        lbv=max(1,32*t3);
        ubv=min(_PB_N-2,32*t3+31);
        #pragma ivdep
        #pragma vector always
        for (t7=lbv;t7<=ubv;t7++)
          B[t4][t7] = (A[t4][t7] + A[t4][t7-1] + A[t4][1+t7] + A[1+t4][t7] + A[t4-1][t7] +
            A[t4-1][t7-1] + A[t4-1][t7+1] + A[t4+1][t7-1] + A[t4+1][t7+1])/8.0;
      }
}
if (_PB_N >= 4) {
  lbp=0;
  ubp=floor(_PB_N-3,32);
  #pragma omp parallel for private(lbv,ubv)
  for (t2=lbp;t2<=ubp;t2++)
    for (t3=0;t3<=floor(_PB_N-2,32);t3++)
      for (t4=max(1,32*t2);t4<=min(_PB_N-2,32*t2+31);t4++) {
        lbv=max(2,32*t3);
        ubv=min(_PB_N-2,32*t3+31);
        #pragma ivdep
        #pragma vector always
        for (t7=lbv;t7<=ubv;t7++)
          A[t4][t7] = (B[t4][t7]-B[t4+1][t7-1]) + (B[t4+1][t7] - B[t4][t7-1]);
      }
}

```

Good parallelism, good vectorization!

Bad locality!

Two barriers

Blur-Roberts tiled with PLuTo using the Smartfuse heuristic (fuse matching dimensions)

```

for (t1=0;t1<=floor(_PB_N-2,16);t1++) {
  lbp=max(1,32*(t1-_PB_N+2,32));
  ubp=floor(_PB_N-1,32*(t1));
  #pragma omp parallel for private(lbv,ubv)
  for (t2=lbp;t2<=ubp;t2++) {
    #pragma omp parallel for private(lbv,ubv)
    for (t3=0;t3<=floor(_PB_N-2,32);t3++)
      for (t4=max(1,32*(t2+_PB_N-2,32*(t1+1));t4++)
        for (t5=0;t5<=floor(_PB_N-2,32*(t1));t5++)
          for (t6=max(1,32*(t2+_PB_N-2,32*(t1+1));t6++)
            for (t7=0;t7<=floor(_PB_N-2,32*(t1));t7++)
              for (t8=0;t8<=floor(_PB_N-2,32*(t1));t8++)
                for (t9=0;t9<=floor(_PB_N-2,32*(t1));t9++)
                  for (t10=0;t10<=floor(_PB_N-2,32*(t1));t10++)
                    for (t11=0;t11<=floor(_PB_N-2,32*(t1));t11++)
                      for (t12=0;t12<=floor(_PB_N-2,32*(t1));t12++)
                        for (t13=0;t13<=floor(_PB_N-2,32*(t1));t13++)
                          for (t14=0;t14<=floor(_PB_N-2,32*(t1));t14++)
                            for (t15=0;t15<=floor(_PB_N-2,32*(t1));t15++)
                              for (t16=0;t16<=floor(_PB_N-2,32*(t1));t16++)
                                for (t17=0;t17<=floor(_PB_N-2,32*(t1));t17++)
                                  for (t18=0;t18<=floor(_PB_N-2,32*(t1));t18++)
                                    for (t19=0;t19<=floor(_PB_N-2,32*(t1));t19++)
                                      for (t20=0;t20<=floor(_PB_N-2,32*(t1));t20++)
                                        for (t21=0;t21<=floor(_PB_N-2,32*(t1));t21++)
                                          for (t22=0;t22<=floor(_PB_N-2,32*(t1));t22++)
                                            for (t23=0;t23<=floor(_PB_N-2,32*(t1));t23++)
                                              for (t24=0;t24<=floor(_PB_N-2,32*(t1));t24++)
                                                for (t25=0;t25<=floor(_PB_N-2,32*(t1));t25++)
                                                  for (t26=0;t26<=floor(_PB_N-2,32*(t1));t26++)
                                                    for (t27=0;t27<=floor(_PB_N-2,32*(t1));t27++)
                                                      for (t28=0;t28<=floor(_PB_N-2,32*(t1));t28++)
                                                        for (t29=0;t29<=floor(_PB_N-2,32*(t1));t29++)
                                                          for (t30=0;t30<=floor(_PB_N-2,32*(t1));t30++)
                                                            for (t31=0;t31<=floor(_PB_N-2,32*(t1));t31++)
                                                              for (t32=0;t32<=floor(_PB_N-2,32*(t1));t32++)
                                                                for (t33=0;t33<=floor(_PB_N-2,32*(t1));t33++)
                                                                  for (t34=0;t34<=floor(_PB_N-2,32*(t1));t34++)
                                                                    for (t35=0;t35<=floor(_PB_N-2,32*(t1));t35++)
                                                                      for (t36=0;t36<=floor(_PB_N-2,32*(t1));t36++)
                                                                        for (t37=0;t37<=floor(_PB_N-2,32*(t1));t37++)
                                                                          for (t38=0;t38<=floor(_PB_N-2,32*(t1));t38++)
                                                                            for (t39=0;t39<=floor(_PB_N-2,32*(t1));t39++)
                                                                              for (t40=0;t40<=floor(_PB_N-2,32*(t1));t40++)
                                                                                for (t41=0;t41<=floor(_PB_N-2,32*(t1));t41++)
                                                                                  for (t42=0;t42<=floor(_PB_N-2,32*(t1));t42++)
                                                                                    for (t43=0;t43<=floor(_PB_N-2,32*(t1));t43++)
                                                                                      for (t44=0;t44<=floor(_PB_N-2,32*(t1));t44++)
                                                                                        for (t45=0;t45<=floor(_PB_N-2,32*(t1));t45++)
                                                                                          for (t46=0;t46<=floor(_PB_N-2,32*(t1));t46++)
                                                                                            for (t47=0;t47<=floor(_PB_N-2,32*(t1));t47++)
                                                                                              for (t48=0;t48<=floor(_PB_N-2,32*(t1));t48++)
                                                                                                for (t49=0;t49<=floor(_PB_N-2,32*(t1));t49++)
                                                                                                  for (t50=0;t50<=floor(_PB_N-2,32*(t1));t50++)
                                                                                                    for (t51=0;t51<=floor(_PB_N-2,32*(t1));t51++)
                                                                                                      for (t52=0;t52<=floor(_PB_N-2,32*(t1));t52++)
                                                                                                        for (t53=0;t53<=floor(_PB_N-2,32*(t1));t53++)
                                                                                                          for (t54=0;t54<=floor(_PB_N-2,32*(t1));t54++)
                                                                                                            for (t55=0;t55<=floor(_PB_N-2,32*(t1));t55++)
                                                                                                              for (t56=0;t56<=floor(_PB_N-2,32*(t1));t56++)
                                                                                                                for (t57=0;t57<=floor(_PB_N-2,32*(t1));t57++)
                                                                                                                  for (t58=0;t58<=floor(_PB_N-2,32*(t1));t58++)
                                                                                                                    for (t59=0;t59<=floor(_PB_N-2,32*(t1));t59++)
                                                                                                                      for (t60=0;t60<=floor(_PB_N-2,32*(t1));t60++)
                                                                                                                        for (t61=0;t61<=floor(_PB_N-2,32*(t1));t61++)
                                                                                                                          for (t62=0;t62<=floor(_PB_N-2,32*(t1));t62++)
                                                                                                                            for (t63=0;t63<=floor(_PB_N-2,32*(t1));t63++)
                                                                                                                              for (t64=0;t64<=floor(_PB_N-2,32*(t1));t64++)
                                                                                                                                for (t65=0;t65<=floor(_PB_N-2,32*(t1));t65++)
                                                                                                                                  for (t66=0;t66<=floor(_PB_N-2,32*(t1));t66++)
                                                                                                                                    for (t67=0;t67<=floor(_PB_N-2,32*(t1));t67++)
                                                                                                                                      for (t68=0;t68<=floor(_PB_N-2,32*(t1));t68++)
                                                                                                                                        for (t69=0;t69<=floor(_PB_N-2,32*(t1));t69++)
                                                                                                                                          for (t70=0;t70<=floor(_PB_N-2,32*(t1));t70++)
                                                                                                                                            for (t71=0;t71<=floor(_PB_N-2,32*(t1));t71++)
                                                                                                                                              for (t72=0;t72<=floor(_PB_N-2,32*(t1));t72++)
                                                                                                                                                for (t73=0;t73<=floor(_PB_N-2,32*(t1));t73++)
                                                                                                                                                  for (t74=0;t74<=floor(_PB_N-2,32*(t1));t74++)
                                                                                                                                                    for (t75=0;t75<=floor(_PB_N-2,32*(t1));t75++)
                                                                                                                                  for (t76=0;t76<=floor(_PB_N-2,32*(t1));t76++)
                                                                                                                                    for (t77=0;t77<=floor(_PB_N-2,32*(t1));t77++)
                                                                                                                                      for (t78=0;t78<=floor(_PB_N-2,32*(t1));t78++)
                                                                                                                                        for (t79=0;t79<=floor(_PB_N-2,32*(t1));t79++)
                                                                                                                                          for (t80=0;t80<=floor(_PB_N-2,32*(t1));t80++)
                                                                                                                                            for (t81=0;t81<=floor(_PB_N-2,32*(t1));t81++)
                                                                                                                                              for (t82=0;t82<=floor(_PB_N-2,32*(t1));t82++)
                                                                                                                                                for (t83=0;t83<=floor(_PB_N-2,32*(t1));t83++)
                                                                                                                                                  for (t84=0;t84<=floor(_PB_N-2,32*(t1));t84++)
                                                                                                                                                    for (t85=0;t85<=floor(_PB_N-2,32*(t1));t85++)
                                                                                                                                  for (t86=0;t86<=floor(_PB_N-2,32*(t1));t86++)
                                                                                                                                    for (t87=0;t87<=floor(_PB_N-2,32*(t1));t87++)
                                                                                                                                      for (t88=0;t88<=floor(_PB_N-2,32*(t1));t88++)
                                                                                                                                        for (t89=0;t89<=floor(_PB_N-2,32*(t1));t89++)
                                                                                                                                          for (t90=0;t90<=floor(_PB_N-2,32*(t1));t90++)
                                                                                                                                            for (t91=0;t91<=floor(_PB_N-2,32*(t1));t91++)
                                                                                                                                              for (t92=0;t92<=floor(_PB_N-2,32*(t1));t92++)
                                                                                                                                                for (t93=0;t93<=floor(_PB_N-2,32*(t1));t93++)
                                                                                                                                                  for (t94=0;t94<=floor(_PB_N-2,32*(t1));t94++)
                                                                                                                                                    for (t95=0;t95<=floor(_PB_N-2,32*(t1));t95++)
                                                                                                                                  for (t96=0;t96<=floor(_PB_N-2,32*(t1));t96++)
                                                                                                                                    for (t97=0;t97<=floor(_PB_N-2,32*(t1));t97++)
                                                                                                                                      for (t98=0;t98<=floor(_PB_N-2,32*(t1));t98++)
                                                                                                                                        for (t99=0;t99<=floor(_PB_N-2,32*(t1));t99++)
                                                                                                                                          for (t100=0;t100<=floor(_PB_N-2,32*(t1));t100++)

```

Good locality!

"Bad" parallelism, poor vectorization!

One barrier executed $O(n)$ times!

Our solution

