The Optimizing Tensor Contraction Engine

Oak Ridge National Laboratory
David E. Bernholdt, ***Robert Harrison***

Pacific Northwest National Laboratory
So Hirata

Louisiana State University
J. Ramanujam

Ohio State University
Gerald Baumgartner, Alina Bibireata, Xiaoyang Gao, Albert Hartono, Sriram Krishnamoorthy, Sandhya Krishnan, Chi-Chung Lam, Quingda Lu, ***Russell Pitzer,*** ***P. Sadayappan***, Alexander Sibiryakov, Vaidy Sivaraman

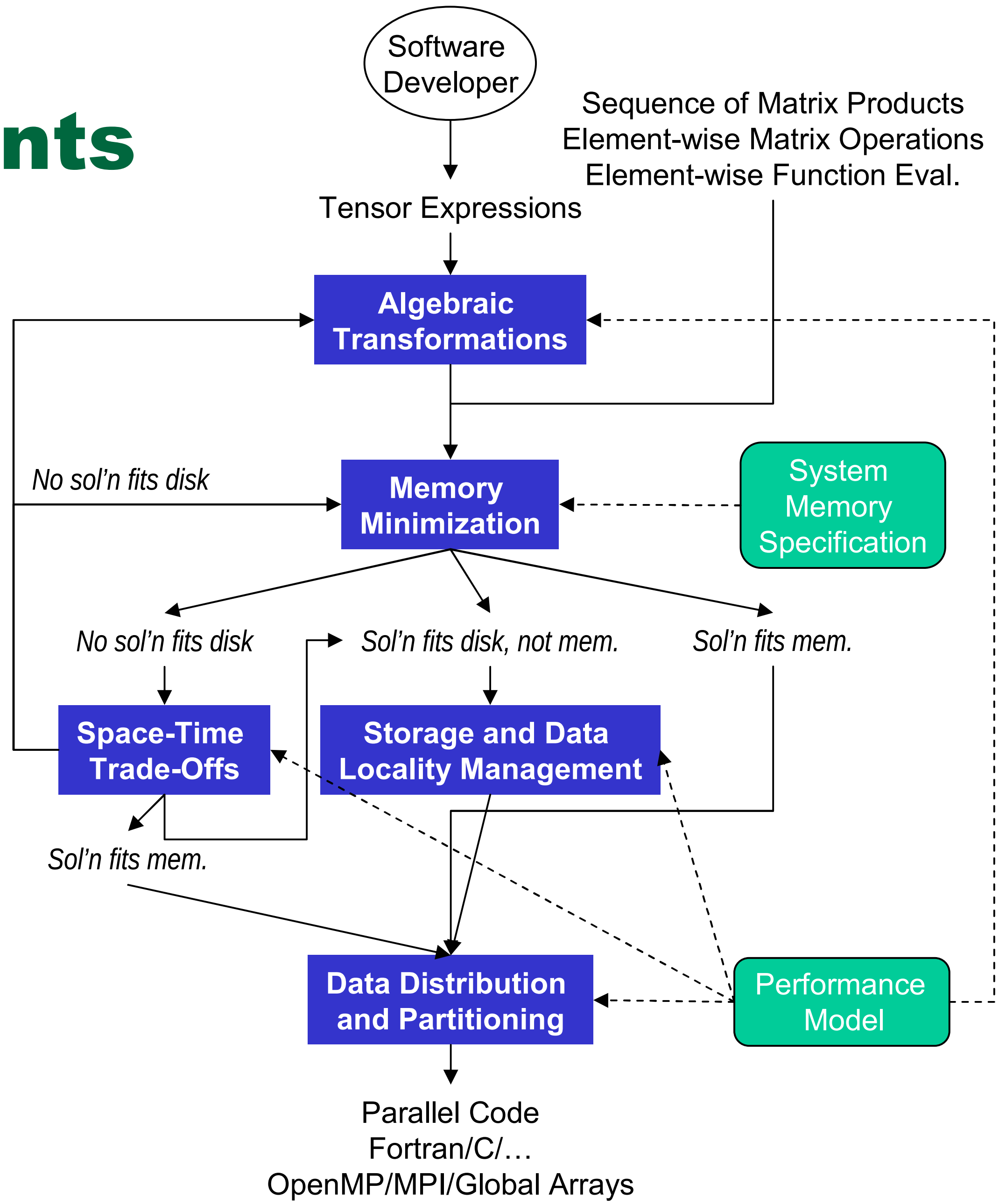
University of Waterloo
Marcel Nooijen, Alexander Auer

Research at ORNL supported by the Laboratory Directed Research and Development Program
Oak Ridge National Laboratory is managed by UT-Battelle, LLC for the US Dept. of Energy under contract DE-AC-05-00OR22725.

Research at PNNL supported by the Office of Basic Energy Sciences, U. S. Dept. of Energy
Research at OSUand LSU supported by the National Science Foundation Information Technology Research Program

TCE Components

- Algebraic Transformations
  - Minimize operation count
- Memory Minimization
  - Reduce intermediate storage
- Space-Time Transformation
  - Trade-offs btw storage and recomputation
- Storage Management and Data Locality Optimization
  - Optimize use of storage hierarchy
- Data Distribution and Partitioning
  - Optimize parallel layout



Memory-Minimal Forms (Loop Fusion)

for a, e, c, f
- for i, j
  - X_{aecf} += $T_{ijae}\ T_{ijcf}$

for c, e, b, k
- $T1_{cebk}$ = f1(c, e, b, k)

for a, f, b, k
- $T2_{afbk}$ = f2(a, f, b, k)

for c, e, a, f
- for b, k
  - Y_{ceaf} += $T1_{cebk}\ T2_{afbk}$

for c, e, a, f
- E += $X_{aecf}\ Y_{ceaf}$

array	space	time
X	$V^4 \rightarrow 1$	V^4O^2
T1	$V^3O \rightarrow VO$	$C_{f1}V^3O$
T2	V^3O	$C_{f2}V^3O$
Y	$V^4 \rightarrow 1$	V^5O
E	1	V^4

$$A3A = \tfrac{1}{2}\, X_{ce,af}Y_{ae,cf} + X_{\bar{c}e,a\bar{f}}Y_{\bar{a}e,c\bar{f}} + X_{\bar{c}\bar{e},af}Y_{\bar{a}\bar{e},cf}$$
$$+ X_{\bar{c}e,a\bar{f}}Y_{ae,\bar{c}\bar{f}} + X_{\bar{c}e,\bar{a}f}Y_{\bar{a}e,\bar{c}f} + X_{\bar{c}\bar{e},\bar{a}\bar{f}}Y_{\bar{a}\bar{e},\bar{c}\bar{f}})$$

$$X_{ce,af} = t_{ij}^{ce}t_{ij}^{af} \qquad Y_{ae,cf} = \langle ab\|ek\rangle\langle cb\|fk\rangle$$

a .. f: range V = 1000 .. 3000
i .. k: range O = 30 .. 100

for a, f, b, k
- $T2_{afbk}$ = f2(a, f, b, k)

for c, e
- for b, k
  - $T1_{bk}$ = f1(c, e, b, k)
- for a, f
  - for i, j
    - X += $T_{ijae}\ T_{ijcf}$
  - for b, k
    - Y += $T1_{bk}\ T2_{afbk}$
  - E += X Y

Effect of Different Optimizations

<u>AO-to-MO Transform:</u> N=150, V=140, Memory = 2GB

$$B(a,b,c,d) = \sum_{p,q,r,s} C1(s,d) x C2(r,c) x C3(q,b) x C4(p,a) x A(p,q,r,s)$$

Optimizations included & omitted	Total Disk I/O time (secs)	Total Execution Time (secs)
Fusion + Optimizing Tiling	248.43	954.87
No Fusion, Optimizing Tiling	747.83	1261.95
No Fusion, Tile size = 4^{th} root of memorySize/3	1240.85	1957.18

Measurements were taken on an Itanium 2 System

Pseudo Codes for AO-to-MO Transform

```
Read C4, C3, C2, C1
FOR s_T
  FOR r_T
    FOR q_T
      FOR p_T
        Read A
        FOR a, s_I, r_I, q_I, p_I
          T1[a,s_I,r_I,q_I] += C4[p,a] *
                          A[p_I,q_I,r_I,s_I]
      FOR a, s_I, r_I, q_I, b
        T2[a,b,s_I,r_I] += T1[a,s_I,r_I,q_I] *
                          C3[q,b]
    FOR a, s_I, r_I, b, c
      T3[a,b,c, s_I] += T2[a,b,s_I,r_I] *
                          C2[r,c]
  Write T3
FOR a_T
  FOR s_T
    Read T3
    FOR a_I, b, c, s_I, d
      B[a_I,b,c,d] += T3[a_I,b,c, sI] *
                          C1[s,d]
  Write B
```

Loop Fusion + Optimizing Tiling

```
FOR aT
  Read C4
  FOR rT, sT
    Read A
    FOR aI, p, q, rI, sI
      T1[aI,q,rI,sI] += C4[p,aI] *
                    A[p,q,rI,sI]
    Write T1
FOR aT, bT
  Read C3
  FOR rT
    Read T1
    FOR s, aI, bI, q, rI
      T2[aI,rI,s,bI] += T1[aI,q,rI,s] *
                    C3[q,bI]
    Write T2
Read C2, C1
FOR aT, bT
  Read T2
  FOR c, r, s, aI, bI
    T3[aI,s,bI,c] += T2[aI,r,s,bI] *
                    C2[r,c]
  Write T3
FOR aT, bT
  Read T3
  FOR c, d, s, aI, bI
    B[aI,bI,c,d] += T3[aI,s,bI,c] *
                    C1[s,d]
  Write B
```

No Fusion, Optimizing Tiling

```
FOR aT, pT
  Read C4
  FOR qT, rT, sT
    Read T1
    Read A
    FOR aI, pI, qI, rI, sI
      T1[aI,qI,rI,sI] += C4[pI,aI] *
                  A[pI,qI,rI,sI]
    Write T1
FOR aT, bT, qT
  Read C3
  FOR rT
    Read T2
    Read T1
    FOR s, aI, bI, qI, rI
      T2[aI,rI,s,bI] += T1[aI,qI,rI,s] *
                  C3[qI,bI]
    Write T2
Read C2, C1
FOR aT, bT
  Read T2
  FOR c, r, s, aI, bI
    T3[aI,s,bI,c] += T2[aI,r,s,bI] *
                C2[r,c]
  Write T3
FOR aT, bT
  Read T3
  FOR c, d, s, aI, bI
    B[aI,bI,c,d] += T3[aI,s,bI,c] *
                C1[s,d]
  Write B
```

No Fusion, Standard Tiling

CCSD Performance Itanium 2 Cluster

900 Megahertz Intel Itanium 2 processors, 4GB RAM		Sequential		4 Processors		16 Processors	
		CPU	Wall	CPU	Wall	CPU	Wall
Benzene (Medium)	**Native NWChem**	1440	1440	389	389	102	102
	Scale			**3.7**		**14.12**	
	Prototype TCE	97.6	97.6	49.5	49.5	46	46
	Scale			**1.97**		**2.12**	
	Optimizing TCE	48.5	48.5	15.6	15.6	13.3	13.3
	Scale			**3.12**		**3.65**	
Benzene (Large)	**Native NWChem**	2865	2865	806	806	212	212
	Scale			**3.55**		**13.5**	
	Prototype TCE	611	665	293	297	282	282
	Scale			**2.24**		**2.36**	
	Optimizing TCE	300	300	120	120	60.1	60.1
	Scale			**2.5**		**4.99**	

All times are in seconds

On the Drawing Board…

- More flexibility in sequencing and controlling optimizations
- Global factorization (across equations)
  - Complex problem
- Improving parallel code generation
  - Overlap of Communication and Computation
  - Multi-level parallelism
    - Threads
    - Multiple loosely coupled tasks
- More sophisticated performance models
- Develop approximate algorithms for opt.
  - Address situations where exhaustive search too expensive
    - i.e. Deliver best result spending at most 3 min on code gen.
    - … or 60 min … or 3 days …
- Generalizations beyond electronic structure