

**Speeding up the  
Four Russians Algorithm by  
About One More Logarithmic Factor**

Timothy Chan

U of Waterloo

# The Problem: Boolean Matrix Multiplication (BMM)

Given  $n \times n$  Boolean matrices  $A, B$ ,  
compute  $C = \{c_{ij}\}$  where

$$c_{ij} = \bigvee_{k=1}^n (a_{ik} \wedge b_{kj})$$

# Previous Alg'ms

trivial

$O(n^3)$  time

Strassen'69

$O(n^{2.81})$

⋮

⋮

⋮

Coppersmith&Winograd'87

$O(n^{2.376})$

Stothers'10

$O(n^{2.373})$

Vassilevska Williams,STOC'12

$O(n^{2.3729})$

Le Gall'14

$O(n^{2.372864})$

# Today

a new BMM alg'm which is...

**SLOWER!**

but is purely **combinatorial**...

# Previous Combinatorial BMM Alg'ms

- Arlazarov&Dinic&Kronrod&Faradzhev'70 (the "4 Russians")
  - $O(n^3 / \log n)$  time
  - **technique**: table lookup
  - **refinement**: + word ops (bitwise-or)
    - $\Rightarrow O(n^3 / (w \log n))$  on  $w$ -bit word RAM  
( $w \geq \log n$ )
    - $\Rightarrow O(n^3 / \log^2 n)$

# Previous Combinatorial BMM Alg'ms

- Arlazarov&Dinic&Kronrod&Faradzhev'70  
(the “4 Russians”)
  - inspired **many** other combinatorial alg'ms...
    - \* all-pairs shortest paths (APSP) in  $O^*(n^3 / \log^2 n)$  time  
[Fredman'75; ...; C.,STOC'07; Han&Takaoka'12]
    - \* LCS & edit distance for bounded alphabet in  $O(n^2 / \log n)$   
[Masek&Paterson'80]
    - \* max unweighted bipartite matching in  $O(n^{5/2} / \log n)$  [Alt et al.'91; Feder&Motwani'91]
    - \* regular expression matching in  $O(nP / \log n)$  [Myer'92]
    - \* integer 3SUM in  $O^*(n^2 / \log^2 n)$   
[Baran&Demaine&Pătrașcu'05]
    - ⋮

# Previous Combinatorial BMM Alg'ms

- Arlazarov&Dinic&Kronrod&Faradzhev'70 (the "4 Russians")
  - inspired **many** other combinatorial alg'ms...
    - \* unweighted undirected APSP in  $O(mn / \log n)$  for  $m \gg n \log n$  [C., SODA'06]
    - \* transitive closure in  $O(mn / \log n)$
    - \* min-plus convolution in  $O^*(n^2 / \log^2 n)$  [Bremner et al.'06]
    - \* CFL reachability in  $O(n^3 / \log^2 n)$  [Chaudhuri'08]
    - \*  $k$ -cliques in  $O(n^k / \log^{k-1} n)$  [Vassilevska'09]
    - \* diameter of real-weighted planar graphs in  $O^*(n^2 / \log n)$  [Wulff-Nilsen'10]
    - ⋮

# Previous Combinatorial BMM Alg'ms

- Arlazarov&Dinic&Kronrod&Faradzhev'70 (the "4 Russians")
  - inspired **many** other combinatorial alg'ms...
    - \* discrete Fréchet distance decision in  $O^*(n^2 / \log n)$  [Agarwal et al., SODA'13]
    - \* Klee's measure problem for integers in  $O^*(n^{d/2} / \log^{d/2-2} n)$  [C., FOCS'13]
    - \* continuous Fréchet distance decision in  $O^*(n^2 / \log n)$  [Buchin et al., "4 Soviets walk the dog...", SODA'14]
    - \* real 3SUM in  $O^*(n^2 / \log n)$  [Grønlund&Pettie, FOCS'14] etc. etc. etc.
  - **notation**:  $O^*$  ignores log log factors



# Previous Combinatorial BMM Alg'ms

- Arlazarov&Dinic&Kronrod&Faradzhev'70 (the "4 Russians")
  - inspired **many** other combinatorial alg'ms...
    - \* discrete Fréchet distance decision in  $O^*(n^2 / \log n)$  [Agarwal et al., SODA'13]
    - \* Klee's measure problem for integers in  $O^*(n^{d/2} / \log^{d/2-2} n)$  [C., FOCS'13]
    - \* continuous Fréchet distance decision in  $O^*(n^2 / \log n)$  [Buchin et al., "4 Soviets walk the dog...", SODA'14]
    - \* real 3SUM in  $O^*(n^2 / \log n)$  [Grønlund&Pettie, FOCS'14] etc. etc. etc.
  - **pattern**: mostly speedup of  $\approx 1$  or 2 logs, until ...

# Previous Combinatorial BMM Alg'ms

- Bansal&Williams [FOCS'09]
  - $O^*(n^3 / (w \log^{7/6} n))$  on  $w$ -bit word RAM  
or  $O^*(n^3 / \log^{2.25} n)$
  - need “advanced” techniques: regularity lemmas

# New Combinatorial BMM Alg'm

- Our result
  - $O^*(n^3 / (w \log^2 n))$  on  $w$ -bit word RAM
    - $\Rightarrow O^*(n^3 / \log^3 n)$
  - completely “elementary” techniques:  
table lookup + word ops (as in the 4 Russians)  
+ an **embarrassingly simple** divide&conquer!

# Setup

- Suffice to solve **rectangular** matrix multiplication:  
multiply a Boolean  $n \times d$  matrix  $A$  & a Boolean  $d \times n$  matrix  $B$   
in  $O(n^2)$  time, for  $d \approx^* \log^3 n$
- **Re-formulation** (with  $m = n$ ):  
given  $m$  red &  $n$  blue Boolean vectors in  $\{0, 1\}^d$ ,  
report all pairs of red & blue vectors with Boolean  
inner product = 1

# Preliminaries: Sparse Version of the 4 Russians Alg'm

- **Lemma:** If fraction of 1's in the blue vectors is  $\leq \beta$ , then problem can be solved in time

$$O^* \left( \beta \frac{dmn}{\log^2 n} + \text{lower-order terms} \right)$$

- **Proof:** let  $b = 0.01 \log_d n$ 
  - precompute inner product of the  $m$  red vectors with every possible vector  $v$  w.  $\leq b$  1's  
 $\Rightarrow O(dm d^b) = O(dmn^{0.01})$  time
  - store answer for each  $v$  in  $O(m/w)$  **words**
  - break the  $n$  blue vectors into  $\approx \beta dn/b$  chunks each w.  $\leq b$  1's & do **table lookup** for each chunk  
 $\Rightarrow O(\beta dn/b \cdot m/w)$  time **Q.E.D.**

# The Main Divide&Conquer Alg'm

0. if fraction of 1's in the blue vectors is  $\leq \beta$  then  
apply Lemma & return  $\Rightarrow O^* \left( \beta \frac{dmn}{\log^2 n} + \dots \right)$
1. find coord. position w.  $\geq \beta n$  1's in the  $n$  blue vectors;  
w.l.o.g., say it's 1st coord.  $\Rightarrow O(dm + dn)$
2. recurse on all red vectors w. 1st coord. 0 &  
all blue vectors  $\Rightarrow T_{d-1}((1-\alpha)m, n)$
3. recurse on all red vectors w. 1st coord. 1 &  
all blue vectors w. 1st coord. 0  $\Rightarrow T_{d-1}(\alpha m, (1-\beta)n)$
4. report all pairs betw'n all red vectors w. 1st coord. 1 &  
all blue vectors w. 1st coord. 1

THAT'S IT!

# The Recurrence

$$T_d(m, n) \leq \begin{cases} T_{d-1}((1-\alpha)m, n) + T_{d-1}(\alpha m, (1-\beta)n) \\ \quad + O(dm + dn) \text{ for some } \alpha, \text{ or} \\ O^* \left( \beta \frac{dmn}{\log^2 n} + \dots \right) \end{cases}$$

for any choice of  $\beta$

How to solve it: guess!

# Solving the Recurrence (Cont'd)

- try induction hypothesis like

$$T_d(m, n) \leq (1 + \delta)^d d n m^{1-\varepsilon} + \beta \frac{d m n}{\log^2 n} + \dots$$

- do the boring math... (read the paper!)
- choosing  $\delta \approx 1/d$ ,  $\varepsilon \approx^* 1/\log m$ ,  $\beta \approx^* 1/\log m$  will work...

$$\Rightarrow T_d(n, n) \leq O(n^2) + O^*\left(\frac{d n^2}{\log^3 n}\right) = O(n^2)$$

for  $d \approx^* \log^3 n$       **Q.E.D.**



# Final Remarks

1. How to define **combinatorial** alg'ms??
2. Practical or not?
3. Our technique is currently limited to BMM only...

# Final Remarks

4. Our divide&conquer alg'm is inspired by a result of Impagliazzo&Lovett&Paturi&Schneider'14 on the **dominance** problem:

given  $n$  red &  $n$  blue points in  $R^d$ ,  
report all  $K$  pairs of red & blue points  $(p, q)$  s.t.  
 $p_i > q_i \quad \forall i = 1, \dots, d$

in near  $n^{2-\Omega(1/(c^{15} \log c))} + K$  time for  $d = c \log n$

– our better analysis:  $n^{2-\Omega(1/(c \log^2 c))} + K$

– also implies a simple combinatorial alg'm for **APSP**  
in  $O^*(n^3 / \log^2 n)$  time (but worse than Williams'  
non-comb.  $n^3 / 2^{\Omega(\sqrt{\log n})}$  alg'm [STOC'14])