Powering Sparse Polynomials

Roman Pearce

Simon Fraser University

September 2012

Joint work with Michael Monagan, Simon Fraser University

supported by MITACS NCE of Canada and NSERC of Canada

Timeline

Hired by Mike to “make polynomials in Maple fast.”

2007	CASC	*Polynomial Division using Dynamic Arrays, Heaps, and Packed Exponent Vectors*
2008	JSC	*Sparse Polynomial Division Using a Heap*
2009	ISSAC	*Parallel Sparse Polynomial Multiplication Using Heaps*
2010	PASCO	*Parallel Sparse Polynomial Division Using Heaps*
2012	CASC	**Sparse Polynomial Powering Using Heaps**

$\Longrightarrow$ *Maple 17: A High Performance System For Polynomials*

Tom Coates' Example

$$f = xy^3z^2 + x^2y^2z + xy^3z + xy^2z^2 + y^3z^2 + y^3z + 2y^2z^2 + 2xyz + y^2z + yz^2 + y^2 + 2yz + z$$

Expand f^{50} (472226 terms out of a possible 1.54×10^6)

... also $f^{100}, f^{150}, f^{200}, f^{300}, f^{500}$, *and higher!*

... *and in more variables!*

Algorithm? Multiply $f \cdot f \cdot f \cdots f$?!

k	terms f^k	Maple 16	Magma 2.17	Singular 3.1.4	our multiply
10	4246	0.030	0.010	0.010	0.000
20	31591	0.403	0.210	0.240	0.030
30	104036	2.537	1.200	1.470	0.260
40	243581	9.062	3.620	4.930	0.970
50	472226	23.131	9.260	12.460	2.620
60	811971	49.572	19.100	26.660	5.730
70	1284816	95.654	36.390	50.180	10.950
250	57636126	–	–	–	40 min

Tom Coates' Example

$$f = xy^3z^2 + x^2y^2z + xy^3z + xy^2z^2 + y^3z^2 + y^3z + 2y^2z^2 + 2xyz + y^2z + yz^2 + y^2 + 2yz + z$$

Expand f^{50} (472226 terms out of a possible 1.54×10^6)

... also $f^{100}, f^{150}, f^{200}, f^{300}, f^{500}$, *and higher!*

... *and in more variables!*

Algorithm? Multiply $f \cdot f \cdot f \cdots f$ *?!*

k	terms f^k	Maple 16	Magma 2.17	Singular 3.1.4	our multiply
10	4246	0.030	0.010	0.010	0.000
20	31591	0.403	0.210	0.240	0.030
30	104036	2.537	1.200	1.470	0.260
40	243581	9.062	3.620	4.930	0.970
50	472226	23.131	9.260	12.460	2.620
60	811971	49.572	19.100	26.660	5.730
70	1284816	95.654	36.390	50.180	10.950
250	57636126	–	–	–	40 min

Why Multiplying is Slow

$$f = xy^3z^2 + x^2y^2z + xy^3z + xy^2z^2 + y^3z^2 + y^3z + 2y^2z^2 + 2xyz + y^2z + yz^2 + y^2 + 2yz + z$$

It slowly builds the result: (number of terms)

i	$f^{i-1} \times f = f^i$	i	$f^{i-1} \times f = f^i$
2	13 x 13 = 58	20	27190 x 13 = 31591
3	58 x 13 = 158	21	31591 x 13 = 36443
4	158 x 13 = 335	22	36443 x 13 = 41768
5	335 x 13 = 611	23	41768 x 13 = 47588
	...		...
10	3145 x 13 = 4246	40	225980 x 13 = 243581
11	4246 x 13 = 5578	41	243581 x 13 = 262073
12	5578 x 13 = 7163	42	262073 x 13 = 281478
13	7163 x 13 = 9023	43	281478 x 13 = 301818

Square and Multiply?

It's **WORSE!** (this is known)

i	$f^{i/2} \times f^{i/2}$	=	f^i	time
2	13 x 13	=	58	0.000
4	58 x 58	=	335	0.000
8	335 x 335	=	2253	0.000
16	2253 x 2253	=	16473	0.090
32	16473 x 16473	=	125873	5.460
64	125873 x 125873	=	983905	19 min

Dense arithmetic? $x \to t,\ y \to t^{2k+1},\ z \to t^{3(2k+1)k+1}$

k	$\deg(f,t)$	Magma 2.17	our multiply	new method
40	19686	1.470	0.968	0.159
70	59646	28.260	10.833	0.941
100	121206	93.640	48.932	3.026
150	271806	*FAIL*	276.320	10.880
250	753006	–	40 min	68.626

Square and Multiply?

It's **WORSE!** (this is known)

i	$f^{i/2} \times f^{i/2}$	=	f^i	time
2	13 x 13	=	58	0.000
4	58 x 58	=	335	0.000
8	335 x 335	=	2253	0.000
16	2253 x 2253	=	16473	0.090
32	16473 x 16473	=	125873	5.460
64	125873 x 125873	=	983905	19 min

Dense arithmetic? $x \to t,\ y \to t^{2k+1},\ z \to t^{3(2k+1)k+1}$

k	$\deg(f,t)$	Magma 2.17	our multiply	new method
40	19686	1.470	0.968	0.159
70	59646	28.260	10.833	0.941
100	121206	93.640	48.932	3.026
150	271806	*FAIL*	276.320	10.880
250	753006	–	40 min	68.626

Main Result

We compute $\mathbf{f^k}$ for about the cost of $\mathbf{f^{k-1}} \times \mathbf{f}$.

The idea came from Euler's formula for power series:

$$f = f_0 + f_1 x + f_2 x^2 + \cdots + f_d x^d$$

$$g_0 = f_0^k$$

$$g_i = \frac{1}{i f_0} \sum_{j=1}^{\min(d,i)} ((k+1)j - i) f_j g_{i-j} \quad \text{for} \quad i = 1 \ldots kd \quad \in \mathbf{O(kd^2)}$$

$g = f^3$		1	$9x$	$33x^2$	$63x^3$	$66x^4$	$36x^5$	$8x^6$
	1							
f	$3x$	$9x$	$54x^2$	$99x^3$	$0x^4$	$-198x^5$	$-216x^6$	
	$2x^2$	$12x^2$	$90x^3$	$264x^4$	$378x^5$	$264x^6$		

Main Result

We compute $\mathbf{f^k}$ for about the cost of $\mathbf{f^{k-1}} \times \mathbf{f}$.

The idea came from Euler's formula for power series:

$$f = f_0 + f_1 x + f_2 x^2 + \cdots + f_d x^d$$

$$g_0 = f_0^k$$

$$g_i = \frac{1}{i f_0} \sum_{j=1}^{\min(d,i)} ((k+1)j - i) f_j g_{i-j} \quad \text{for} \quad i = 1 \ldots kd \quad \in \mathbf{O(kd^2)}$$

	$g = f^3$	1	$9x$	$33x^2$	$63x^3$	$66x^4$	$36x^5$	$8x^6$
	1							
f	$3x$	$9x$	$54x^2$	$99x^3$	$0x^4$	$-198x^5$	$-216x^6$	
	$2x^2$	$12x^2$	$90x^3$	$264x^4$	$378x^5$	$264x^6$		

□

Main Result

We compute $\mathbf{f^k}$ for about the cost of $\mathbf{f^{k-1}} \times \mathbf{f}$.

The idea came from Euler's formula for power series:

$$f = f_0 + f_1 x + f_2 x^2 + \cdots + f_d x^d$$

$$g_0 = f_0^k$$

$$g_i = \frac{1}{i f_0} \sum_{j=1}^{\min(d,i)} ((k+1)j - i) f_j g_{i-j} \;\text{ for }\; i = 1 \ldots kd \;\; \in \mathbf{O(kd^2)}$$

	$g = f^3$	1	$9x$	$33x^2$	$63x^3$	$66x^4$	$36x^5$	$8x^6$
	1							
f	$3x$	$9x$	$54x^2$	$99x^3$	$0x^4$	$-198x^5$	$-216x^6$	
	$2x^2$	$12x^2$	$90x^3$	$264x^4$	$378x^5$	$264x^6$		

□

Main Result

We compute $\mathbf{f^k}$ for about the cost of $\mathbf{f^{k-1}} \times \mathbf{f}$.

The idea came from Euler's formula for power series:

$$f = f_0 + f_1 x + f_2 x^2 + \cdots + f_d x^d$$

$$g_0 = f_0^k$$

$$g_i = \frac{1}{i f_0} \sum_{j=1}^{\min(d,i)} ((k+1)j - i) f_j g_{i-j} \quad \text{for} \quad i = 1 \ldots kd \quad \in \mathbf{O(kd^2)}$$

	$g = f^3$	1	$9x$	$33x^2$	$63x^3$	$66x^4$	$36x^5$	$8x^6$
	1							
f	$3x$	$9x$	$54x^2$	$99x^3$	$0x^4$	$-198x^5$	$-216x^6$	
	$2x^2$	$12x^2$	$90x^3$	$264x^4$	$378x^5$	$264x^6$		

□

Main Result

We compute $\mathbf{f^k}$ for about the cost of $\mathbf{f^{k-1}} \times \mathbf{f}$.

The idea came from Euler's formula for power series:

$$f = f_0 + f_1 x + f_2 x^2 + \cdots + f_d x^d$$

$$g_0 = f_0^k$$

$$g_i = \frac{1}{i f_0} \sum_{j=1}^{\min(d,i)} ((k+1)j - i) f_j g_{i-j} \ \text{ for } \ i = 1 \ldots kd \ \in \mathbf{O(kd^2)}$$

$g = f^3$		1	$9x$	$33x^2$	$63x^3$	$66x^4$	$36x^5$	$8x^6$
	1							
f	$3x$	$9x$	$54x^2$	$99x^3$	$0x^4$	$-198x^5$	$-216x^6$	
	$2x^2$	$12x^2$	$90x^3$	$264x^4$	$378x^5$	$264x^6$		

□

Main Result

We compute $\mathbf{f^k}$ for about the cost of $\mathbf{f^{k-1}} \times \mathbf{f}$.

The idea came from Euler's formula for power series:

$$f = f_0 + f_1 x + f_2 x^2 + \cdots + f_d x^d$$

$$g_0 = f_0^k$$

$$g_i = \frac{1}{i f_0} \sum_{j=1}^{\min(d,i)} ((k+1)j - i) f_j g_{i-j} \quad \text{for} \quad i = 1 \ldots kd \quad \in \mathbf{O(kd^2)}$$

	$g = f^3$	1	$9x$	$33x^2$	$63x^3$	$66x^4$	$36x^5$	$8x^6$
	1							
f	$3x$	$9x$	$54x^2$	$99x^3$	$0x^4$	$-198x^5$	$-216x^6$	
	$2x^2$	$12x^2$	$90x^3$	$264x^4$	$378x^5$	$264x^6$		

□

Main Result

We compute $\mathbf{f^k}$ for about the cost of $\mathbf{f^{k-1}} \times \mathbf{f}$.

The idea came from Euler's formula for power series:

$$f = f_0 + f_1 x + f_2 x^2 + \cdots + f_d x^d$$

$$g_0 = f_0^k$$

$$g_i = \frac{1}{i f_0} \sum_{j=1}^{\min(d,i)} ((k+1)j - i) f_j g_{i-j} \;\; \text{for} \;\; i = 1 \dots kd \;\; \in \mathbf{O(kd^2)}$$

$g = f^3$		1	$9x$	$33x^2$	$63x^3$	$66x^4$	$36x^5$	$8x^6$
	1							
f	$3x$	$9x$	$54x^2$	$99x^3$	$0x^4$	$-198x^5$	$-216x^6$	
	$2x^2$	$12x^2$	$90x^3$	$264x^4$	$378x^5$	$264x^6$		

□

Main Result

We compute $\mathbf{f^k}$ for about the cost of $\mathbf{f^{k-1}} \times \mathbf{f}$.

The idea came from Euler's formula for power series:

$$f = f_0 + f_1 x + f_2 x^2 + \cdots + f_d x^d$$

$$g_0 = f_0^k$$

$$g_i = \frac{1}{i f_0} \sum_{j=1}^{\min(d,i)} ((k+1)j - i) f_j g_{i-j} \quad \text{for} \quad i = 1 \ldots kd \quad \in \mathbf{O(kd^2)}$$

$g = f^3$		1	$9x$	$33x^2$	$63x^3$	$66x^4$	$36x^5$	$8x^6$
	1							
f	$3x$	$9x$	$54x^2$	$99x^3$	$0x^4$	$-198x^5$	$-216x^6$	
	$2x^2$	$12x^2$	$90x^3$	$264x^4$	$378x^5$	$264x^6$		□

Main Result

We compute $\mathbf{f^k}$ for about the cost of $\mathbf{f^{k-1}} \times \mathbf{f}$.

The idea came from Euler's formula for power series:

$$f = f_0 + f_1 x + f_2 x^2 + \cdots + f_d x^d$$

$$g_0 = f_0^k$$

$$g_i = \frac{1}{i f_0} \sum_{j=1}^{\min(d,i)} ((k+1)j - i) f_j g_{i-j} \text{ for } i = 1 \ldots kd \in \mathbf{O(kd^2)}$$

$g = f^3$		1	$9x$	$33x^2$	$63x^3$	$66x^4$	$36x^5$	$8x^6$
	1							
f	$3x$	$9x$	$54x^2$	$99x^3$	$0x^4$	$-198x^5$	$-216x^6$	
	$2x^2$	$12x^2$	$90x^3$	$264x^4$	$378x^5$	$264x^6$		

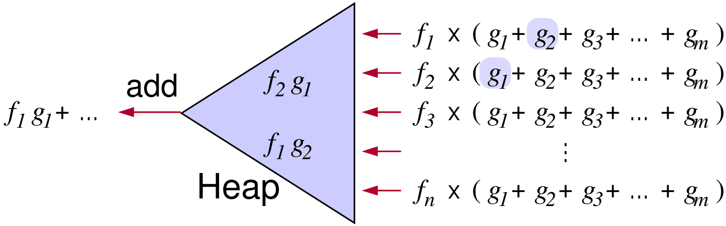
□

Sparse Version

For multivariate polynomials we use Kronecker substitution.

Merge **only** products where f_i and g_{i-j} non-zero:

$$g_i = \frac{1}{if_0} \sum_{j=1}^{\min(d,i)} ((k+1)j - i) f_j g_{i-j} \text{ for } i = 1 \ldots kd.$$



Problem: $\mathbb{Z}_p$, redundant products whose sum is zero.

Improvement

Euler's method multiplies

$$(\text{terms of } f) \times (\text{terms of } f^{k-1})$$

to compute $\mathbf{f^{k-1}}$.

But it can output $\mathbf{f^{k}}$ almost for free!

$\implies$ FPS algorithm in paper (not fully optimized)

Improvement

Euler's method multiplies

$$\text{(terms of } f) \times \text{(terms of } f^{k-1})$$

to compute $\mathbf{f^{k-1}}$.

But it can output $\mathbf{f^{k}}$ almost for free!

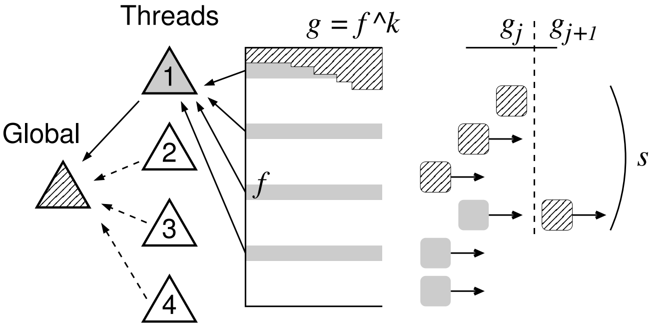
$\Longrightarrow$ FPS algorithm in paper (not fully optimized)

$f = c_1x_1 + c_2x_2 + \cdots + c_tx_t$. f^k generates $\binom{t+k-1}{k}$ terms.

t	k	Magma	Singular	multiply	binomial	**FPS**
3	100	0.010	0.050	0.026	0.001	0.001
3	500	3.480	12.750	4.560	0.055	0.069
4	50	0.120	0.180	0.033	0.005	0.007
4	200	74.360	44.610	13.151	0.521	0.714
6	30	63.270	1.170	0.173	0.039	0.057
6	40	–	6.670	1.471	0.222	0.531
8	25	–	10.700	1.504	0.452	0.649
8	35	–	148.970	28.342	5.927	13.828

Parallelism

PROBLEM: data dependency among terms of $g = f^k$.



Solution: reduce parallelism *dynamically* if program stalls.

No OS interaction. Communication with memory barriers.

Parallel Benchmark

$f = (1 + x + y)^{15}$ $t = 136$							Magma	Singular
k	*SUMS*	4 cores	*FPS*	*RMUL*	4 cores	*BINA*	*FFT*	*RMUL*
20	0.536	0.149	0.685	1.514	0.429	1.553	0.49	12.33
40	3.157	0.846	4.181	15.833	4.406	16.375	5.49	134.59
60	9.263	2.478	12.552	65.276	17.927	66.790	27.27	522.59
80	20.439	5.402	28.110	182.717	49.830	187.178	56.42	–
120	64.117	16.618	88.688	–	–	–	325.60	–

$f = (1 + w + x + y + z)^4$ $t = 70$							Magma	Singular
k	*SUMS*	2 cores	*FPS*	*RMUL*	2 cores	*BINA*	*FFT*	*RMUL*
4	0.005	0.005	0.003	0.003	0.003	0.003	0.30	0.01
8	0.068	0.062	0.048	0.071	0.047	0.072	1.24	1.01
12	0.711	0.440	1.021	0.955	0.589	0.995	10.84	10.40
16	2.311	1.297	3.784	5.238	3.120	5.443	65.50	46.49
20	5.852	4.755	10.337	17.164	10.065	17.790	218.14	166.02
24	12.313	11.350	22.643	44.008	25.513	45.489	391.42	394.08
28	23.430	22.754	45.458	97.179	56.745	100.277	(*)	–

Further Reading

M. Monagan, R. Pearce. Sparse Polynomial Powering Using Heaps.
CASC 2012 Proceedings `http://www.cecm.sfu.ca/~rpearcea/`

R. Fateman. On the computation of powers of sparse polynomials.
Studies in Applied Math., 53 (1974), pp. 145–155.

R. Fateman. Polynomial multiplication, powers, and asymptotic analysis.
SIAM J. Comput. **3**, 3 (1974), pp. 196–213.

S.C. Johnson. Sparse polynomial arithmetic.
ACM SIGSAM Bulletin, **8** (3) 63–71, 1974.

Monagan, Pearce. Parallel Sparse Polynomial Multiplication Using Heaps.
ISSAC 2009 Proceedings, ACM Press, 295–315.

Monagan, Pearce. Parallel Sparse Polynomial Division Using Heaps.
PASCO 2010 Proceedings, ACM Press, 105–111, 2010.