

大規模並列環境における疎行列固有値解法の性能予測

西 田 晃^{†1}

本稿では、科学技術振興機構 CREST 事業等の支援により開発を進めている並列反復解法ライブラリ Lis の大規模並列環境への適用を念頭に、疎行列ベクトル積の事前性能評価をもとにした疎行列固有値解法の性能予測手法の妥当性について検討する。

Performance Prediction of Eigensolvers for Sparse Matrices on Massively Parallel Computing Environments

AKIRA NISHIDA ^{†1}

This study discusses the validity of the performance prediction of eigensolvers based on the preliminary measurement of sparse matrix vector products, considering the application of parallel iterative solver library Lis to massively parallel computing environments.

1. 背 景

本研究では、平成 14-19 年度科学技術振興機構 CREST 事業の一環として、反復解法ライブラリ Lis ^{*1} を開発、配布し、様々な並列計算機上で大規模な線型方程式を解くための環境を提供してきた。また平成 20 年度からは九州大学情報基盤研究開発センターにおいて、大規模並列環境への対応を中心に研究を行うとともに、同年 11 月より疎行列固有値解法に対応した新版を公開している。

Lis は、Krylov 部分空間法を中心とする多様な反復解法を実装したライブラリである。同様な目的のライブラリとして、線型方程式については Argonne 米国立研究所の並列反復解

^{†1} 九州大学情報基盤研究開発センター

Research Institute for Information Technology, Kyushu University

^{*1} <http://www.ssisc.org/lis/>

法ライブラリ PETSc ^{*2} や Lawrence Berkeley 米国立研究所による並列反復解法ライブラリ Hypre ^{*3} などを挙げるができる。固有値解法については、Valencia 工科大学による SLEPc ^{*4} (PETSc を用いて開発されている) や、Colorado 大学による BLOPEX ^{*5} (Hypre を用いている) などに疎行列を対象とした固有値解法が実装されている ^{*6}。また、逐次コードが対象であるが、California 大学において、PETSc において疎行列ベクトル積部の性能最適化を行うためのインタフェース OSKI ^{*7} 等が開発されている。NAS Parallel Benchmark ^{*8} には、疎行列ベクトル積を多用する共役勾配法コード CG が含まれており、計算機システムの性能評価に用いられている。

Lis ではこれらの機能を単一のライブラリにおいて実現するとともに、多様な前処理アルゴリズムを実装している。また、近年一般的となったマルチコア環境に適したハイブリッド並列処理に対応している点も特徴として挙げるができる。表 1-3 に現時点で対応している固有値解法、線型方程式解法、行列格納形式の一覧を示す。

表 1 Lis で利用可能な固有値解法

Power Iteration
Inverse Iteration
Approximate Inverse Iteration
Conjugate Gradient
Lanczos Iteration
Subspace Iteration
Conjugate Residual

疎行列固有値解法の適用例として、2 次元 Helmholtz 問題を考える。矩形領域 $[0, l] \times [0, m]$ の膜の振動を表す 2 次元 Laplace 作用素を 5 点中心差分により離散化した場合、対応する行列の固有値は

$$\pi^2 \left(\frac{\sigma^2}{l^2} + \frac{\tau^2}{m^2} \right), \sigma, \tau \in N \quad (1)$$

^{*2} <http://www-unix.mcs.anl.gov/petsc/>

^{*3} <http://computation.llnl.gov/casc/hypr/>

^{*4} <http://www.grycap.upv.es/slepc/>

^{*5} <http://www-math.cudenver.edu/~aknyazev/software/BLOPEX/>

^{*6} <http://www.netlib.org/utk/people/JackDongarra/la-sw.html>

^{*7} <http://bebop.cs.berkeley.edu/oski/>

^{*8} <http://www.nas.nasa.gov/Resources/Software/npb.html>

表 2 Lis で利用可能な線形方程式解法

CG	CR
BiCG	BiCR
CGS	CRS
BiCGSTAB	BiCRSTAB
GPBiCG	GPBiCR
BiCGSafe	BiCRSafe
BiCGSTAB(l)	BiCRSTAB(l)
Jacobi	Gauss-Seidel
SOR	Orthomin(m)
TFQMR	MINRES
GMRES(m)	FGMRES(m)
IDR(s)	

表 3 Lis で利用可能な行列格納形式

Compressed Row Storage	(CRS)
Compressed Column Storage	(CCS)
Modified Compressed Sparse Row	(MSR)
Diagonal	(DIA)
Ellpack-Itpack generalized diagonal	(ELL)
Jagged Diagonal	(JDS)
Block Sparse Row	(BSR)
Block Sparse Column	(BSC)
Variable Block Row	(VBR)
Dense	(DNS)
Coordinate	(COO)

で与えられる。 $l = m = 20$ として、Lis に実装した部分空間反復法により絶対値最小のものから順に 6 個の固有対を求めた結果を図 1 に示す。なお内部の解法には逆反復法、前処理なし共役勾配法を使用している。図 2, 3 に示すように、九州大学の日立 SR16000 の 1 コア上でこの計算を行った場合、行列ベクトル積の割合は約 75%を占める。問題サイズを 200×200 等としてもこの割合はほとんど変わらない。

このように、反復解法においては、疎行列ベクトル積が計算時間の大半を占めることが多く、性能の最適化は最も重要な課題のひとつである。

しかしながら、疎行列ベクトル積の性能は、行列の形状、並列化手法、メモリの階層構造等によって大きく性能が変化するため、すべての場合に最適な解法を見出すのは難しい。本研究では平成 20 年度に、異なるデータ格納形式・並列化手法における疎行列ベクトル積の性能を事前に評価するためのベンチマークプログラムを Lis 上に実装したが、ここではその有効性について、九州大学先端的計算科学研究プロジェクトの一環として本センターの計算環境を用いて評価したので、その結果について報告する。

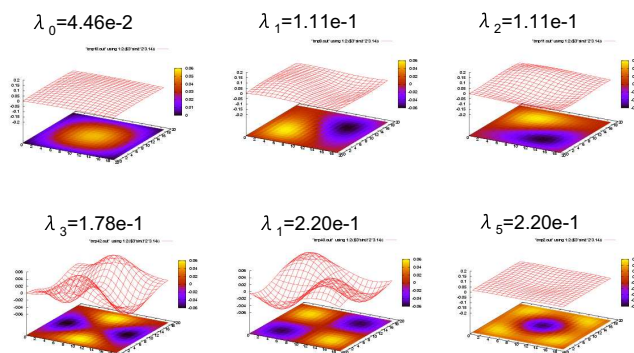


図 1 2次元 Helmholtz 問題（膜の振動）におけるモード

%	cumulative	self	self	self	total	name
time	seconds	seconds	calls	ms/call	ms/call	
75.0	0.06	0.06	7692	0.01	0.01	.lis_matvec_crs [8]
12.5	0.07	0.01	17571	0.00	0.00	.lis_vector_dot [9]
12.5	0.08	0.01	725	0.01	0.01	.lis_vector_axpyz [10]
0.0	0.08	0.00	16846	0.00	0.00	.lis_vector_axpy [11]

図 2 部分空間反復法における各処理の割合（日立 SR16000 上、問題サイズ 20×20 。Lis のプロファイリングオプションによる。）

75.7	44.30	44.30	66984	0.66	0.66	.lis_matvec_crs [8]
9.4	49.77	5.47	135837	0.04	0.04	.lis_vector_axpy [9]
5.7	53.13	3.36	136700	0.02	0.02	.lis_vector_dot [10]
3.4	55.12	1.99	66984	0.03	0.03	.lis_vector_xpay [11]

図 3 部分空間反復法における各処理の割合（日立 SR16000 上、問題サイズ 200×200 ）。

2. 性能評価

ここでは、非零要素数の比較的小さい 3 点中心差分による離散化 1 次元 Poisson 方程式を用いて、係数行列とベクトル $(1, \dots, 1)^T$ との積を各行列格納形式について計算し、非零要素数と経過時間から FLOPS 値を求めるベンチマークを実施した。この問題の場合、係数行列は 3 重対角となるため、非零要素数の割合は非常に小さく、実問題における疎行列ベクトル積の性能の下限に近い値が得られると期待される。同様なベンチマークとして、ベ

クトル間演算によりメモリ帯域幅を評価する STREAM Benchmark *1 などが挙げられる。なお、測定は strong scaling, weak scaling の双方の性能の推移を調べるため、使用コア数と問題サイズの双方を変化させて行なった。事前評価において MPI 版の性能が最も高かったことから、以下では MPI 版について評価した。

評価環境として、九州大学情報基盤研究開発センターに設置された富士通 PRIMEQUEST 580 (1.6GHz Intel デュアルコア Itanium2 プロセッサ ×32) サーバおよび PRIMERGY RX200S3 (3.0 GHz Intel デュアルコア Xeon プロセッサ ×2) サーバをそれぞれ DDR InfiniBand スイッチにより接続した 2 クラスタ (ノードあたり片方向理論転送性能はそれぞれ 4GB/s, 2GB/s), および日立 SR16000 サーバ (4.7GHz IBM デュアルコア POWER6 プロセッサ ×16) を DDR InfiniBand スイッチにより接続したクラスタ (ノードあたり片方向理論転送性能は 8GB/s) を用いた。

Lis の行列格納形式については昨年度の HOKKE2010 で既に述べているが、以下では、性能評価で使用された主な格納形式の特徴について再掲する。なお行列の行 (列) 番号は 0 から始まるものとする。 $n \times n$ 行列 $A = (a_{ij})$ の非零要素数を nnz とする。

Modified Compressed Sparse Row (MSR)

MSR 形式は CRS 形式を修正したものである。その違いは対角部分を分けて格納しているところである。MSR 形式は行列データを 2 つの配列 (index, value) に格納する。 ndz を対角部分の零要素数とする。

- 長さ $nnz + ndz + 1$ の倍精度配列 value は n 番目までは行列 A の対角部分を格納する。 $n + 1$ 番目の要素は使用しない。 $n + 2$ 番目からは行列 A の対角以外の非零要素の値を行方向に沿って格納する。
- 長さ $nnz + ndz + 1$ の整数配列 index は $n + 1$ 番目までは行列 A の非対角部分の各行の開始位置を格納する。 $n + 2$ 番目からは行列 A の非対角部分の配列 value に格納された非零要素の列番号を格納する。

MSR 形式での格納方法を図 4 に示す。

2 プロセス上への MSR 形式での格納方法を図 5 に示す。

Diagonal (DIA)

DIA は行列データを 2 つの配列 (index, value) に格納する。 nnd を行列 A の非零対角の本数とする。

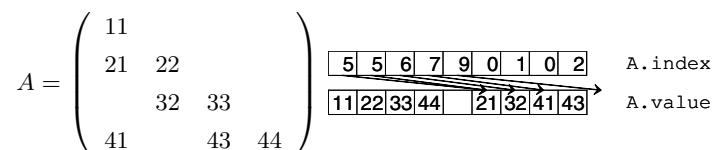


図 4 MSR 形式のデータ構造

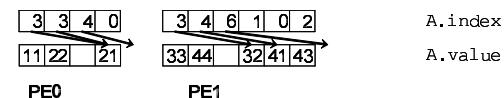


図 5 2PE 上での MSR 形式のデータ構造

- 長さ $nnd \times n$ の倍精度配列 value は行列 A の非零対角を格納する。
- 長さ nnd の整数配列 index は主対角から各対角へのオフセットを格納する。

OpenMP 版では以下のように修正している。

DIA は 2 つの配列 (index, value) に格納する。 $nprocs$ をスレッド数とする。 nnd_p を行列 A を行ブロック分割した部分行列の非零対角の本数とする。 $maxnnd$ を nnd_p の値の最大値とする。

- 長さ $maxnnd \times n$ の倍精度配列 value は行列 A を行ブロック分割した部分行列の非零対角を格納する。
- 長さ $nprocs \times maxnnd$ の整数配列 index は主対角から各対角へのオフセットを格納する。

DIA 形式での格納方法を図 6 に示す。

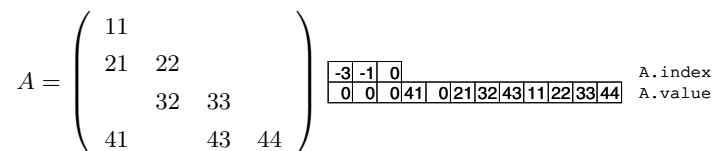


図 6 DIA 形式のデータ構造

2 プロセス上への DIA 形式での格納方法を図 7 に示す。

*1 <http://www.cs.virginia.edu/stream/>

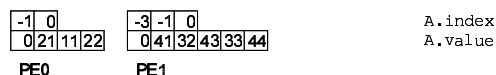


図 7 2PE 上での DIA 形式のデータ構造

万-4096 万, SR16000 上では 64 万-4096 万の範囲で測定した. 各コア数について, 最も性能の高い性能を示した格納形式と問題サイズに対応する値に下線を引いている.

Block Sparse Row (BSR)

BSR では行列を $r \times c$ の大きさの部分行列 (ブロックと呼ぶ) に分解する. BSR は CRS と同様の手順で非零ブロック (少なくとも 1 つの非零要素が存在する) を格納する. $nr = n/r$, $nnzb$ を A の非零ブロック数とする. BSR は行列データを 3 つの配列 (**bp**tr, **bin**dex, **value**) に格納する.

- 長さ $nnzb \times r \times c$ の倍精度配列 **value** は非零ブロックの全要素を格納する.
- 長さ $nnzb$ の整数配列 **bin**dex は非零ブロックのブロック列番号を格納する.
- 長さ $nr + 1$ の整数配列 **bp**tr は配列 **bin**dex のブロック行の開始位置を格納する.

BSR 形式での格納方法を図 8 に示す.

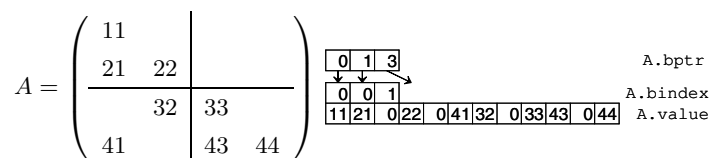


図 8 BSR 形式のデータ構造

2PE 上への BSR 形式での格納方法を図 9 に示す.

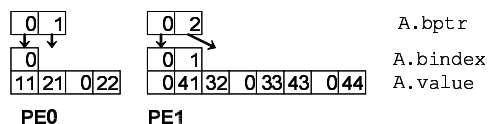


図 9 2PE 上での BSR 形式のデータ構造

3. 考 察

表 4-6 に, 上記アーキテクチャ上での 9 種の行列格納形式の MPI 版での性能を示す. なお行列次数は, 各並列度について RX200S3 上で 32 万-4096 万, PRIMEQUEST 上で 16

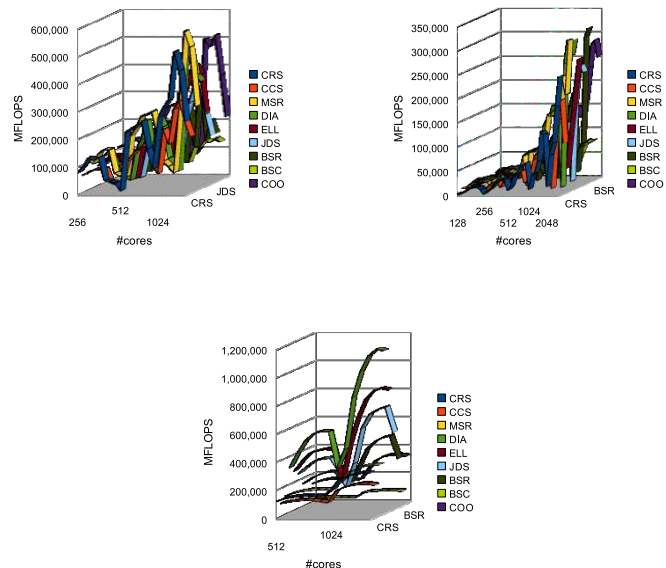


図 10 富士通 PRIMERGY RX200S3, PRIMEQUEST 580, 日立 SR16000 クラスタ上での MFLOPS 値

表 4-6 をグラフ化したものを図 10 に示す. 前節での評価結果から, 最適な格納形式が計算機アーキテクチャ, 行列サイズ等によって異なることが見て取れる. 最適な格納形式は, PRIMERGY の場合には MSR(Modified Compressed Sparse Row), PRIMEQUEST では BSR(Block Sparse Row), SR では DIA(Diagonal) 形式となっており, それぞれ異なっている. 同一コア数で問題サイズを変化させた場合の性能を見ると, PRIMERGY において FLOPS 値が最大となるのは, コア当たりの行列データサイズが約 480KB の場合である. 同じく PRIMEQUEST では約 1920KB, SR では約 480 KB であった. 図から見て取れるように, これらのサイズを越えると性能は急速に低下する. また, コア数を変化させた場合の性能は, weak scaling に関してはリニアにスケールしているが, strong scaling

表 4 富士通 PRIMERGY RX200 (最大 1024 コア) 上での MFLOPS 値

Number of Cores	Matrix Size	CRS	CCS	MSR	DIA	ELL	JDS	BSR	BSC	COO
256	320,000	79,971	62,914	117,220	94,965	72,419	66,009	75,901	34,533	108,240
	640,000	102,456	73,881	96,213	109,342	90,586	82,174	81,057	33,195	111,615
	1,280,000	133,384	81,591	133,827	114,187	104,314	88,277	89,181	36,242	130,256
	2,560,000	126,770	84,017	147,222	135,090	110,410	91,162	92,259	36,607	136,550
	5,120,000	137,880	86,302	147,273	87,060	110,733	88,562	71,242	35,614	127,409
	10,240,000	53,771	39,415	63,264	85,170	50,664	37,028	36,720	34,498	44,336
	20,480,000	49,407	31,846	53,510	40,590	33,395	24,412	33,309	30,772	38,431
	40,960,000	44,613	30,834	48,461	29,706	27,215	22,625	32,379	28,025	33,430
512	640,000	15,912	110,924	198,596	137,190	141,282	116,965	138,846	58,250	206,489
	1,280,000	226,528	134,610	227,167	176,990	166,472	176,119	70,517	67,772	249,321
	2,560,000	242,836	155,165	269,108	245,240	203,168	170,978	209,920	71,503	259,986
	5,120,000	248,695	171,501	273,391	108,167	208,392	130,243	209,442	74,960	253,590
	10,240,000	269,080	167,838	285,855	265,449	207,972	178,610	158,010	70,763	251,585
	20,480,000	168,381	103,319	197,100	223,250	167,391	109,687	80,448	68,799	81,568
1,024	40,960,000	95,596	71,052	106,777	108,864	75,170	57,961	66,370	58,936	78,280
	1,280,000	281,330	229,923	88,180	80,330	340,150	224,007	266,437	115,788	101,712
	2,560,000	316,582	176,554	65,128	62,372	339,971	293,639	123,135	114,880	498,256
	5,120,000	344,332	303,388	294,445	467,522	410,871	360,115	398,420	142,548	471,112
	10,240,000	518,507	334,499	575,733	518,820	437,741	354,273	149,955	132,404	512,934
	20,480,000	450,324	258,798	519,395	408,332	402,622	253,117	220,736	147,159	368,535
40,960,000	263,132	204,498	306,291	389,037	265,313	171,764	172,972	134,104	216,321	

については、概ね問題サイズ 256 万以下の場合に飽和が見られる。これは 512 コアの場合、コア当たりの問題サイズが約 120KB となる大きさである。なお、並列度、問題サイズの違いによる最適な格納形式の変化は、ここでは観察できなかった。

4. ま と め

本稿では、現在開発を進めている並列反復解法ライブラリ Lis への適用を目的として、複数アーキテクチャ上での疎行列ベクトル積の事前性能評価の妥当性について検討した。疎行列ベクトル積の性能は、行列の形状、並列化手法、メモリの階層構造等によって大きく性能が変化するため、すべての場合に最適な解法を見出すのは難しい。したがって、現実的な解として、本研究で試みた局所的な性能解析を一般化し、より多様なデータ格納手法について評価を可能にすることが考えられる。本研究では、今後これらの手法について、実装、評価を行っていく予定である。

表 5 富士通 PRIMEQUEST 580 (最大 2048 コア) 上での MFLOPS 値

Number of Cores	Matrix Size	CRS	CCS	MSR	DIA	ELL	JDS	<u>BSR</u>	BSC	COO
128	160,000	2,911	10,061	7,787	12,848	5,675	10,599	1,238	3,840	10,006
	320,000	3,124	9,296	14,505	17,793	6,173	11,710	9,322	4,678	17,160
	640,000	6,337	11,800	14,157	19,305	10,847	15,172	4,371	5,270	16,521
	1,280,000	10,792	12,832	18,662	20,372	13,739	15,986	13,728	5,312	18,374
	2,560,000	14,230	12,936	20,552	22,099	17,064	15,845	20,727	5,507	19,026
	5,120,000	16,666	13,489	20,624	22,191	17,460	16,761	16,259	5,545	19,212
	10,240,000	17,610	13,111	20,759	21,955	17,811	16,137	25,662	5,393	18,332
	20,480,000	17,856	9,241	20,376	21,205	16,178	11,066	17,455	4,092	10,363
	40,960,000	13,783	5,743	13,557	8,919	7,444	5,836	12,896	3,352	6,030
256	320,000	4,045	16,883	26,464	31,335	8,341	15,856	8,808	7,676	23,925
	640,000	9,688	22,142	30,207	35,783	12,808	27,252	17,305	10,636	32,703
	1,280,000	13,288	24,729	30,329	40,498	24,990	30,438	9,093	10,190	35,927
	2,560,000	19,394	24,520	35,653	43,047	29,357	33,099	39,202	10,646	36,632
	5,120,000	27,081	13,462	37,256	42,951	31,355	31,992	41,472	10,174	32,461
	10,240,000	34,828	25,669	42,070	44,370	35,074	33,053	50,670	11,175	38,667
	20,480,000	34,821	21,095	40,251	43,321	32,986	25,921	41,562	10,461	33,294
	40,960,000	34,081	17,757	38,389	38,245	26,578	20,980	39,887	8,481	23,025
	512	640,000	12,145	34,232	41,129	49,864	20,702	38,857	5,283	18,585
1,280,000		20,447	43,105	48,999	74,273	26,168	45,985	39,283	17,740	65,793
2,560,000		40,043	46,399	73,729	75,289	42,742	59,570	17,398	21,322	65,672
5,120,000		45,834	47,035	13,226	79,768	53,887	59,567	31,086	20,675	75,461
10,240,000		58,109	23,371	81,188	81,664	64,562	62,425	78,325	20,968	64,913
20,480,000		64,961	27,243	82,354	82,506	68,163	64,839	94,042	21,184	63,837
40,960,000		68,641	41,236	84,367	87,433	72,129	59,050	95,453	19,003	61,907
1,024	1,280,000	14,062	70,765	82,342	130,256	33,593	67,787	9,867	37,283	108,022
	2,560,000	41,564	82,585	116,416	144,709	71,894	100,194	86,884	37,857	133,246
	5,120,000	47,500	90,211	111,984	150,226	90,471	111,693	34,815	40,628	145,839
	10,240,000	71,949	98,905	158,622	172,963	129,549	118,427	126,552	40,855	147,164
	20,480,000	132,783	48,226	155,822	173,604	97,500	128,355	159,309	41,873	126,670
	40,960,000	127,036	60,425	168,866	152,705	132,861	130,013	186,721	42,212	127,304
2,048	2,560,000	19,789	11,736	163,100	5,057	38,451	8,143	20,699	56,335	169,806
	5,120,000	84,155	132,738	167,903	260,617	99,806	166,773	73,210	71,662	228,740
	10,240,000	109,412	175,352	238,389	308,141	157,729	242,380	81,728	81,669	278,924
	20,480,000	183,337	193,184	200,496	275,377	218,675	247,715	301,772	77,976	285,902
	40,960,000	246,932	104,830	309,901	307,975	265,292	233,597	320,989	82,941	251,106

表 6 日立 SR16000 (最大 1024 コア) での MFLOPS 値

Number of Cores	Matrix Size	CRS	CCS	MSR	<u>DIA</u>	ELL	JDS	BSR	BSC	COO
512	640,000	120,420	93,126	130,498	321,095	281,723	248,820	163,613	44,801	132,604
	1,280,000	141,992	105,444	152,752	425,637	351,931	298,041	201,654	47,347	153,202
	2,560,000	152,784	112,695	165,611	512,934	409,460	338,827	228,076	48,726	163,585
	5,120,000	159,254	116,469	173,501	562,438	426,003	354,780	243,640	49,197	168,252
	10,240,000	162,472	117,135	177,045	<u>578,889</u>	436,540	360,528	252,131	49,083	169,885
	20,480,000	163,036	111,928	178,404	<u>578,284</u>	428,811	279,282	172,282	48,942	164,232
	40,960,000	164,723	103,265	179,485	320,600	210,106	154,457	173,452	48,669	161,849
1024	1,280,000	208,359	183,055	256,753	454,013	476,371	324,884	283,708	85,407	234,766
	2,560,000	281,195	207,955	301,063	823,316	662,531	574,398	407,415	94,894	297,710
	5,120,000	305,895	224,378	328,094	1,007,026	793,112	672,595	461,758	97,668	328,069
	10,240,000	318,460	232,762	345,028	1,116,494	852,176	704,170	489,948	98,685	338,894
	20,480,000	326,748	234,844	353,670	<u>1,164,711</u>	869,295	723,901	503,813	98,628	338,963
	40,960,000	328,333	224,007	357,180	<u>1,154,665</u>	859,846	542,728	339,557	96,130	330,024