ALPINES

Algorithms and parallel tools for integrated numerical simulations

INRIA Rocquencourt, LJLL, UPMC

Members

• X. Claeys (MdC), L. Grigori (DR), F. Hecht (PR), F. Nataf (DR)

Associated members

• J. Burman (MdC Paris-Sud), J. Beauquier (PR Paris-Sud)

Currently 6+3 Phd students, +2 postdoctoral researcher, 1 software engineer

Research group

- Lab. J.-L. Lions, UPMC
 - Mesh generation techniques, FEM
 - F. Hecht (Professor)
 - Domain decomposition, PDEs
 - F. Nataf (Director of Research CNRS)
 - Boundary element methods, modelling of electromagnetic wave propopagation X. Claeys (MdC)
- INRIA
 - Linear algebra and high performance algorithms L. Grigori (Director of Research)
- University Paris 11 (associate members)
 - Fault tolerance
 - J. Burman (MdC), J Beauquier (Professor)
- Currently 9 Phd students, 2 postdoctoral researchers, 1 software engineer

Methodology

- Mesh generation for parallel computation
 - Exploit the formalism of FreeFEM, a language dedicated to finite elements
 - F. Hecht, F. Nataf
- Solvers for numerical linear algebra
 - Design of domain decomposition and multilevel direction preserving preconditioners
 - High performance computing for boundary element methods
 - X. Claeys, L. Grigori, F. Hecht, F. Nataf
- Computational kernels for numerical linear algebra
 - Design of novel numerical algorithms that minimize communication
 - J. Beauquier, J. Burman, L. Grigori, F. Hecht
- Integration and validation in numerical simulations
 - All members

CA-ILU0 Block Filtering Decomposition (BFD), Nested Filtering Factorization (NFF)

R. Fezzani, P. Kumar, L. Grigori, R. Lacroix, S. Moufawad, F. Nataf, L. Qu, K. Wang INRIA, LJLL, UPMC

> ANR Petal and Petalh projects http://petal.saclay.inria.fr/

ILU0 with nested dissection and ghosting

Let α_0 be the set of equations to be solved by one processor For j = 1 to s do Find $\beta_j = ReachableVertices (G(U), \alpha_{j-1})$ Find $\gamma_j = ReachableVertices (G(L), \beta_j)$ Find $\delta_j = Adj (G(A), \gamma_j)$ Set $\alpha_j = \delta_j$ end Ghost data required: $x(\delta), A(\gamma, \delta),$ $L(\gamma, \gamma), U(\beta, \beta)$

⇒ Half of the work performed on one processor

	1 2	3	4	56	7	8	9	10	101	51	52	53 5	4 55	56	57	58	59	60	463	232	233	234 2	235 23	6 237	238	239 24	0 241	332	282	283 28	4 285	286 2	287 28	8 289 2	90 291
	11 1	2 13	14	15 16	i 17	18	19	20	102	61	62	63 6	4 65	66	67	68	69	70	464	242	243	244 2	45 24	6 247	248	249 25	0 251	333	292	293 29	4 295	296 2	97 29	8 299 3	00 301
	21 2	2 23	24 3	25 26	27	28	29	30	103	71	72	73 7	4 75	76	77	78	79	80	465	252	253	254 2	255 25	6 257	258	259 26	0 261	334	302	303 30	4 305	306 3	807 30	8 309 3	10 311
	31 3	2 33	34 3	35 36	37	38	39	40	104	81	82	83 8	4 85	86	87	88	89	90	466	262	263	264 2	265 26	6 267	268	269 27	0 271	335	312	313 31	4 315	316 3	317 31	8 319 3	20 321
I	41 4	2 43	44 4	45 46	47	48	49	50	105	91	92	93 9	4 95	5 96	97	98	99 1	100	467	272	273	274 2	275 27	6 277	278	279 28	0 281	336	322	323 32	4 325	326 3	327 32	8 329 3	30 331
l	211 21	2 213	214 2	15 21	6 217	218	219 2	220	221	222	223 2	24 22	25 22	6 227	7 228	229	230 2	231	468	442	443	444 4	45 44	6 447	448	449 45	0 451	452	453	454 43	5 456	457 4	58 45	9 460 4	61 462
-	106 10	7 108	109 1	10 11	1 112	113	114 1	115	206	156	157 1	58 15	59 16	0 161	162	163	164 1	165	469	337	338	339 3	340 34	1 342	343	344 34	5 346	437	387	388 38	39 390	391 3	92 39	3 394 3	95 396
	116 11	7 118	119 1	20 12	1 122	123	124-1	125	207	166	167 1	68 16	59 17	0 171	172	173	174 1	175	470	347	348	349 3	350 35	1 352	2 353	354 35	5 356	438	397	398 39	9 400	401 4	02 40	3 404 4	05 406
	126 12	7 128	129 1	30 13	1 132	133	134 1	135	208	176	177 1	78 17	9 18	0 181	182	183	184 1	185	471	357	358	359 3	360 36	1 362	2 363	364 36	5 366	439	407	408 40	9 410	411 4	12 41	3 414 4	15 416
	136 13	7 138	139 1	40 14	1 142	143	144-1	145	209	186	187 1	88 18	9 19	0 191	192	193	194 1	195	472	367	368	369 3	370 37	1 372	373	374 37	5 376	440	417	418 41	9 420	421 4	22 42	3 424 4	25 426
	146 14	7 148	149 1	50 15	1 152	153	154 1	155	210	196	197 1	98 19	9 20	0 201	202	203	204 2	205	473	377	378	379 3	880 38	1 382	383	384 38	5 386	441	427	428 42	9 430	431 4	32 43	3 434 4	35 436
								L																											
Г]				D	oma	in &	gho	st ec	quati	ons					1	Dom	ain 8	è gho	ost ec	quatio	ns	Γ			(Jhost	t data	from	
1			_ 1	Doma	ain 1	1	-			- -	ar ha	claus	ande	ubet	itotia	on		1.			-	for fo	NPUL 91	rd eni	hetiti	ution		- I '				117701	at colu	ution a	ector

5 point stencil on a 2D grid

CA-ILU0 with AMML reordering and ghosting

- Reduce volume of ghost data by reordering the vertices using Alternating Min-Max Layers (AMML) reordering:
 - First number the vertices at odd distance from the separators
 - Then number the vertices at even distance from the separators
- CA-ILU0 computes a standard ILU0 factorization

	3 4 2 3 1 2 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1 1	0 5 4 8 4 21 18 40 29 51 35	27 48 20 34 9 210 114 130	2 4 2 3 1 5 2 4 1 5 1 2 1 3 1 2 1	9 4 3 9 3 22 19 41 28 52 34	26 47 19 35 8 21: 13: 12: 13: 13:	5 2 3 1 7 1 9 1	28 49 12 26 15 42 27 53 33	25 46 36 7 21 13 13 13	5 5 4 2 1 8 1 3 1 0 1 0 1	43 50 21 41 11 228 116 143 126 155	1 1 3 (21 12 12 12	6 7 5 13 13 20 22 21	31 33 42 10 130 147 145 146	3 5 4 1 21 10 10 10 10	1 6 9 7 0 8	03 05 02 04 01 31 06 09 07 10 08	54 56 53 55 51 212 150 160 158 161	8: 8: 8 6 2 22 5 16 5 19 19 19 9 19	5 3 4 1 1 99 1 99 1 99 1 99 1 99 1 99 1 9	71 72 59 91 57 64 92 74 77	91 92 72 83 62 16 19 18 20 20	7 5 7 2 7 2 7 3 8 4 30 3	80 98 73 90 59 218 165 190 178 202 184	7 9 7 8 6 22 10 11 20 12	9 4 9 60 25 63 89 79 01 85	100 96 63 219 167 193 200 200	0 7 5 7 5 8 5 9 2 7 10 3 19 1 11 4 18 0 18	77 78 70 88 24 58 62 91 75 83 82	94 92 93 64 220 166 198 187 188	66 68 65 22 15 17 17 17 17 17	5 7 2 0 3	46 46 46 47 47 47 47 47 47 47 47 46	33 36 54 57 55 73 70 72 59 71 38	(247 249 246 237 442 353 351 354 352	7 20 5 20 5 20 7 20 2 30 2 30 2 30 2 30 2 30 2 30 2 30	52 54 53 71 59 42 59 47 57 58 56	258 259 265 238 445 343 373 358 364 363	27 25 27 24 34 37 36 38 38	6 2(9 2) 6 2) 5 2 8 4 8 3 4 5 3) 1 3) 4 3) 1 3)	50 2 78 2 54 2 56 2 40 2 46 4 45 3 70 3 59 3 33 3 55 3	261 277 255 267 241 447 346 369 360 382 366	280 281 257 269 244 460 350 374 362 386 385) 251 1 252 7 250 9 260 4 239 9 448 9 344 9 344 9 344 9 344 9 344 9 355 9 355 9 355	1 272 2 274 0 273 5 275 9 243 8 461 4 349 2 380 5 378 7 379 6 377	2 234 4 236 3 233 3 233 3 233 3 233 1 443 9 337 0 340 8 338 9 341 7 339	1 5 5 2 3 3 7 0 3 1 0	334 336 333 335 332 462 437 440 438 441 439	284 286 283 285 282 444 387 390 388 391 389	32(32) 32) 32) 32) 32) 32) 42) 42) 42) 42) 42) 42) 42)	0 29 2 29 1 29 3 31 1 28 7 45 5 39 8 42 5 40 7 40 5 40	07 : 08 : 06 : 15 : 37 : 20 4 01 4 03 4 02 4	330 331 302 319 292 450 397 424 407 436 435	0 30 1 32 2 20 3 2 20 5 42 7 30 4 4 7 40 5 42 5 4	06 26 99 14 88 51 93 19 04 31 11	30 32 30 31 29 45 39 42 40 43 41	09 29 03 18 03 55 08 23 08 34 14	307 325 300 313 289 450 394 418 405 412	7 3 5 3 0 3 3 3 9 2 0 4 4 3 5 4 5 4 5 4 2 4	10 28 04 17 94 54 99 22 09 33 15	308 324 301 310 290 445 395 410 400 415	8 31 4 32 1 30 2 31 0 29 9 45 5 40 7 42 6 41 9 43 3 41	11 27 05 16 05 21 10 32 16		
	-					-]	Do	m	air	n 1				_		 		Do for	ma ba	in ck	& wa	gl ard	hos I st	st e 1b:	eq sti	uat tut	ior	ns 1								-	l f	Do	ma for	in d wa	& g rd :	gho sut	st e osti	qua tuti	atior on	15						_		Gł	105 FTE	st (ent	da t s	ıta olu	fro	om on	ve	cto	or		

Comparison with Block Jacobi

- Block Jacobi is another preconditioner which does not require communication •
- Tests for a boundary value problem (Achdou, Nataf), 40x40x40 grid •

(

٠

Block Jacobi using ILU0 - BJ-ILU0

$$-div(\kappa(x)\nabla u) = f \quad in \Omega$$

$$u = 0 \quad on \partial \Omega_D$$

$$\frac{\partial u}{\partial n} = 0 \quad on \partial \Omega_N$$

$$\Omega = [0,1]^3, \partial \Omega_N = \partial \Omega \setminus \partial \Omega_D$$

$$\kappa \text{ jumps from 1 to 10^3}$$
Methods tested:
• Natural ordering NO+ILU0
• CA-ILU0 - kway+AMML(1)+ILU0
• Block Jacobi using LU - BJ+ILU0

32

64

128

-kway+ILU0

512

256

Number of Processors

-kway+AMML(1)+ILU0

1024

Motivation

BOILU0 - Case 2 - 30 x 30 x 16 Relative residual vs number of iterations

Solve

 $M^{-1}Ax = M^{-1}b$

Incomplete LU has plateaus

in the convergence Often due to the presence of few low eigenvalues



Filtering factorization

- Preconditioner M satisfies a filtering property for input A and set of vectors T
 MT = AT or T^TM = T^TA
- Filtering vectors **T** are chosen to improve the convergence
- Complementary with incomplete LU factorization

Preserving directions of interest

- Pointwise approximate factorization satisfying a row-sum criteria, Dupont, Kendall, and Rachford (1968), Gustafsson (1978)
 - Improves the condition number of the preconditioned matrix for matrices arising from finite difference approximation of second order elliptic equations
- Nested factorization, Appleyard, Cheshire (1983)
 - If $t^T r_0 = 0$, then at any iteration $t^T r_k = 0$, this ensures a mass conservation property
- Filtering factorization, Wagner, Wittum (1997), Achdou, Nataf (2001)
- Direction preserving semiseparable approximation of SPD matrices, Gu, Li, Vassilevski (2010)
 - If the near null-space of the original fine grid matrix is preserved, then view the preconditioner as a coarse discretization matrix
 - Conditioning analysis performed by Napov, components dropped are orthogonal to components preserved
- Multigrid methods
 - Bootstrup AMG (Karsten Kahl)

Arbitrary matrices

- Let *A* be partitioned into a block matrix of size N x N
- The square diagonal blocks are not necessarily of a same size



• The generalization of filtering preconditioner to arbitrary matrices is a step forward towards parallel computation

Exact factorization of arbitrary matrices

• An exact block LDU factorization of A is:

$$A = (L+D)D^{-1}(D+U)$$

$$= \begin{pmatrix} D_{11} & & \\ L_{21} & D_{22} & \\ \vdots & \ddots & \ddots & \\ L_{N1} & \cdots & L_{N,N-1} & D_{NN} \end{pmatrix} \cdot \begin{pmatrix} D_{11}^{-1} & & \\ & D_{22}^{-1} & & \\ & & \ddots & \\ & & & D_{NN}^{-1} \end{pmatrix} \cdot \begin{pmatrix} D_{11} & U_{12} & \cdots & U_{1N} \\ & D_{22} & \ddots & \vdots \\ & & & \ddots & \\ & & & D_{NN}^{-1} \end{pmatrix}$$

• Let C=L+D+U. Each block of L, D, U is computed as:

$$C_{ij} = \begin{cases} A_{ij}, i = 1, j = 1\\ A_{ij} - \sum_{k=1, L_{ik} \neq 0, U_{kj} \neq 0}^{\min(i, j) - 1} L_{ik} D_{kk}^{-1} U_{kj}, i > 1, orj > 1 \end{cases}$$

٦

Block Filtering Decomposition (BFD)

• Let *t* be a filtering vector. A BFD preconditioner *M* is written as:

 $M = (\overline{L} + \overline{D})\overline{D}^{-1}(\overline{D} + \overline{U})$

• Let $\overline{C} = \overline{L} + \overline{D} + \overline{U}$. The blocks of M are computed with the following formula, where $i, j = 1 \dots N$

$$\overline{C}_{ij} = \begin{cases} A_{ij}, & i = 1 \quad or \quad j = 1 \\ A_{ij} - \sum_{k=1, L_{ik} \neq 0, U_{kj} \neq 0}^{\min(i, j) - 1} \overline{L}_{ik} F_{kj} \overline{U}_{kj}, & i > 1 \quad or \quad j > 1 \end{cases}$$

where F_{ki} is a sparse approximation such that

$$\overline{L}_{ik}F_{kj}\overline{U}_{kj}t_{j} = \overline{L}_{ik}\overline{D}_{kk}^{-1}\overline{U}_{kj}t_{j}$$

Suitability for parallel computation

• Partition the matrix using nested dissection, thus enabling parallelism



Page 13

Results for a boundary value problem

 SKY (provided by Achdou, Nataf), discretized on a 400x400x400 grid (64 millions unknowns, 447 millions nonzeros)

$$-div(\kappa(x)\nabla u) = f \quad in \Omega$$
$$u = 0 \quad on \,\partial\Omega_D$$
$$\frac{\partial u}{\partial n} = 0 \quad on \,\partial\Omega_N$$
$$\Omega = [0,1]^3, \partial\Omega_N = \partial\Omega \setminus \partial\Omega_D$$

 κ jumps from 1 to 10^3



• Tests use GMRES (PETSc), tolerance = 10⁻⁸



Comparison with Restricted Additive Schwarz (RAS)

Settings:

- Curie supercomputer based on Bullx system, nodes composed of two eight-core Intel Sandy Bridge.
- Subdomains solved using Pardiso, separators solved using MUMPS.
- GMRES and RAS from PETSc.

NFF vs RAS, SKY 400x400x400

Subdom	Iteration	Error	Iteration	Error
256	5489	5.9e-7	268	2.2e-6
512	6126	2.7e-6	273	3.2e-6
1024	7163	1.8e-6	289	2.6e-6
2048	10000	3.7e-6	317	3.8e-6



Best student paper finalist, Qu, LG, Nataf, SC'13

Page 15