Develop, License, Test, Curate Mathematical Optimization in the Real World

Matthias Miltenberger · Franziska Schlösser

Department Mathematical Optimization, Zuse Institute Berlin

June 4, 2019 · deRSE19, Potsdam

Introduction

Zuse Institute Berlin - www.zib.de

- 72113
- Non-university research institute and computing center of the state of Berlin

Research Units:

- Numerical analysis and modeling
- Visualization and data analysis
- Optimization: energy traffic linear and nonlinear IP
- Scientific information systems
- Distributed algorithms and supercomputing

President: Prof. Christof Schütte



M. Miltenberger (ZIB), F. Schlösser (ZIB): SCIP: Develop, License, Test, Curate



- manufacturing: sheet metal design
- networks: operation and design of water and gas networks
- energy production and distribution: plant design, power scheduling
- logistics: public transport, supply-chain management
- industrial engineering: mining with stockpiling constraints
- biological engineering, chemical industry, ...



Introduction (Mixed) Integer Optimization



$$\min c^{T}x + d^{T}y$$

subject to $Ax + By = b$

linear constraints

 $x \in \mathbb{Z}^n, y \in \mathbb{R}^m$



Introduction (Mixed) Integer Nonlinear Optimization



min
$$c^T x + d^T y$$

subject to $Ax + By = b$
 $G(x, y) = 0$
 $x \in \mathbb{Z}^n, y \in \mathbb{R}^m$

linear constraints nonlinear constraints

The functions $g_k \in G$ can be









SCIP is currently one of the fastest non-commercial solvers for mixed integer programming (MIP) and mixed integer nonlinear programming (MINLP).

SCIP Some Stats



SCIP is currently one of the fastest non-commercial solvers for mixed integer programming (MIP) and mixed integer nonlinear programming (MINLP).

CODE

- more than 500 000 lines of C code
- plugin based design
- easily extendable and exchangeable



DEVELOPERS

- approx. 20 active developers,
 > 30 contributors
- most work at ZIB, some in Aachen, Erlangen, Darmstadt and Lancaster

SCIP

All around the world

- runs on Linux, Windows, macOS, ...
- free for academics, available in source code: http://scip.zib.de
- active mailing list with 400+ members: scip@zib.de
- 8000 downloads per year







2002 Tobias Achterberg starts SCIP project

first industry project: Chip design verification



 $2002\,$ Tobias Achterberg starts SCIP project

- first industry project: Chip design verification
- 2004 Thorsten Koch publishes $\rm ZIMPL$ a mathematical modeling language for $\rm SCIP$ as part of his PhD



- $2002\,$ Tobias Achterberg starts SCIP project
 - first industry project: Chip design verification
- 2004 Thorsten Koch publishes $\rm ZIMPL$ a mathematical modeling language for $\rm SCIP$ as part of his PhD
- $2005\,$ First public version of SCIP



- 2002 Tobias Achterberg starts SCIP project
 - first industry project: Chip design verification
- 2004 Thorsten Koch publishes $\rm ZIMPL$ a mathematical modeling language for $\rm SCIP$ as part of his PhD
- $2005\,$ First public version of SCIP
- 2008 Gerald Gamrath starts GCG extension (Master's thesis)
 - development moved to Aachen and spawned several PhD theses



- 2002 Tobias Achterberg starts SCIP project
 - first industry project: Chip design verification
- 2004 Thorsten Koch publishes $\rm ZIMPL$ a mathematical modeling language for $\rm SCIP$ as part of his PhD
- $2005\,$ First public version of SCIP
- 2008 Gerald Gamrath starts GCG extension (Master's thesis)
 - development moved to Aachen and spawned several PhD theses
- 2009 Yuji Shinano starts ${\rm UG}$ project (massively parallel ${\rm SCIP}$ extension)



- 2002 Tobias Achterberg starts SCIP project
 - first industry project: Chip design verification
- 2004 Thorsten Koch publishes $\rm ZIMPL$ a mathematical modeling language for $\rm SCIP$ as part of his PhD
- $2005\,$ First public version of SCIP
- 2008 Gerald Gamrath starts GCG extension (Master's thesis)
 - development moved to Aachen and spawned several PhD theses
- 2009 Yuji Shinano starts ${\rm UG}$ project (massively parallel ${\rm SCIP}$ extension)
- 2010 Start of SAP cooperation (still running today)



- 2002 Tobias Achterberg starts SCIP project
 - first industry project: Chip design verification
- 2004 Thorsten Koch publishes $\rm ZIMPL$ a mathematical modeling language for $\rm SCIP$ as part of his PhD
- $2005\,$ First public version of SCIP
- 2008 Gerald Gamrath starts GCG extension (Master's thesis)
 - development moved to Aachen and spawned several PhD theses
- 2009 Yuji Shinano starts ${\rm UG}$ project (massively parallel ${\rm SCIP}$ extension)
- 2010 Start of SAP cooperation (still running today)
- $2011\,$ Switch from ${\rm CVS}$ to ${\rm Git}$ for version control



- 2002 Tobias Achterberg starts SCIP project
 - first industry project: Chip design verification
- 2004 Thorsten Koch publishes $\rm ZIMPL$ a mathematical modeling language for $\rm SCIP$ as part of his PhD
- $2005\,$ First public version of SCIP
- 2008 Gerald Gamrath starts GCG extension (Master's thesis)
 - development moved to Aachen and spawned several PhD theses
- 2009 Yuji Shinano starts ${\rm UG}$ project (massively parallel ${\rm SCIP}$ extension)
- 2010 Start of SAP cooperation (still running today)
- $2011\,$ Switch from ${\rm CVS}$ to ${\rm Git}$ for version control
- 2014 SCIP development becomes integral part of Forschungscampus MODAL



- 2002 Tobias Achterberg starts SCIP project
 - first industry project: Chip design verification
- 2004 Thorsten Koch publishes $\rm ZIMPL$ a mathematical modeling language for $\rm SCIP$ as part of his PhD
- $2005\,$ First public version of SCIP
- 2008 Gerald Gamrath starts GCG extension (Master's thesis)
 - development moved to Aachen and spawned several PhD theses
- 2009 Yuji Shinano starts ${\rm UG}$ project (massively parallel ${\rm SCIP}$ extension)
- 2010 Start of SAP cooperation (still running today)
- $2011\,$ Switch from ${\rm CVS}$ to ${\rm Git}$ for version control
- 2014 $\,{\rm SCIP}$ development becomes integral part of Forschungscampus MODAL
- 2016 First technical assistant without explicit research tasks



- stay attractive for new students, PhD candidates and postdocs
- establish an environment where everyone can grow and wants to stay
 - ZIB employs many PhD students and only few postdocs or professors
 - PhD students are usually employed for more than 5 years

attract industry projects

- forces code maturity, stability and performance
- introduces challenging real world problems
- finances development
- maintain communications
 - weekly meetings with all developers at ZIB
 - monthly online meetings with all developers

Then and Now Take home messages / Lessons learned





Source Code Management







from the official website git-scm.com:

"GIT is a free and open source distributed version control system"



from the official website git-scm.com:

"GIT is a free and open source distributed version control system"

- Free and open source
- Distributed
- Snapshot based
- Staging Area

- Branches
- User system
- History/Evolution of code

▶ ...



master



M. Miltenberger (ZIB), F. Schlösser (ZIB): SCIP: Develop, License, Test, Curate













M. Miltenberger (ZIB), F. Schlösser (ZIB): SCIP: Develop, License, Test, Curate











 $\rm GITLAB$ is an open-core self-hostable Git-repository manager, that provides Issuetracker, wiki, CI/CD and much more.



 $\rm GITLAB$ is an open-core self-hostable Git-repository manager, that provides Issuetracker, wiki, CI/CD and much more.

- Issues
- Merge Requests
- Milestones
- integrates MATTERMOST (a messaging platform)

…

Continuous testing

Types of tests for SCIP

- Unit/end-to-end tests
- Different platform/compiler
- Code analysis
- Code Coverage



Continuous testing Testing and JENKINS



Types of tests for SCIP

- Unit/end-to-end tests
- Different platform/compiler
- Code analysis
- Code Coverage

Schedule

- Regularly (i.e. every night)
- run tests on Merge Request
- update websites/install software on code update

Continuous testing Testing and JENKINS



Types of tests for SCIP

- Unit/end-to-end tests
- Different platform/compiler
- Code analysis
- Code Coverage

Schedule

- Regularly (i.e. every night)
- run tests on Merge Request
- update websites/install software on code update

from the official website jenkins.io:

"The leading open source automation server, JENKINS provides hundreds of plugins to support building, deploying and automating any project."













Performance analysis



What is a testrun?

- Run SCIP for a list of problem files (instances)
- Collect the output
- Concatenate output into big files





Performance analysis Infrastructure





Performance analysis Logfiles

top - 11:35:28 up 3 days, 31 min, 1 user, load average: 1.04, 1.05, 1.01

Linux opt18 4.4.0-142	-peneric #168-Ubuntu	SMP Wed Jan 16 21:00:45 UT0	C 2019 x05_64 x86_64 x86_64 GNU/Linux
-----------------------	----------------------	-----------------------------	---------------------------------------

Tasks:	: 272	total,	- 2 r	unning, i	27D slee	ping,	0 stop	pped,	0 zombie		
4Cpu(1		4.2 us,	12.0	sy, 0.0	ni, 83.	5 id, 0	-3 ss;	. 0.0	hi, 0.1	si, 0.0 st	
KiB Me	(a) 1 (1)	32830348	tota	1, 174814	076 free	, 9850	36 use	ed, 14	364236 but	f/cache	
KIB SI	ap:	8389994	tota	1, \$3834	504 free		0 US4	ed. 33	207332 ava	il Men	
PID	USER.	PR	NI	VIRT	RES	SER S	ACPU.	NEM	TINE+	CONMAND	
28721	reot	20		0		8 R	93.8	0.0	11:54.43	kworker/0:3	
1	root	20		185500	6156	4016 S	0.0	0.0	8:22.64	systemd	
2	root	20		0		8 S	0.0	0.0	0:00.33	kthreadd	
3	reat	20		0		0.5	0.0	0.0	0:00.46	ksoftirgd/0	
7	reat	20		0		0.5	0.0	0.0	0:49.90	rcu sched	
	root	20		0		0.5	0.0	0.0	0:00.00	rcu bh	

9 reot rt 0 0 0 0 0 0.0 0.0 0.0 0:00.02 nigration/0 rt 0 0 0 0 0.5 0.0 0.0 0:00.02 nigration/0

hard time limit: 60:12:00 hard man limit: 5600

SLIEM (ebt0 - 21027305

d01 /nfs/OPTI/jenkins/workspace/SCIP cmake custombranch testrum/check/instances/CP/linking.cip

Thu Feb 14 11:35:28 CET 2019

e83 1558148528

SCIP version 6.0.2 [precision: 8 byte] [memory: block] [mode: optimized] [LP solver: SoPlex 4.0.2] [GitHash: ad80505] Copyright (C) 2002-2019 Konrad-Zune-Zentrum fuer Informationstechnik Berlin (ZIB)

External codes: ReadLine 6.3 SoPlex 4.0.2 CppAD 20180000.0 2L18 1.2.8 GMP 6.1.0 bliss 0.73p Ipopt 3.12.11	OBI library for command line aftism (pup corp/reallow) Library Forgenamics (below devideout 2 librar fortices for line (copter zih.de) (GiBKah: 572264/) Algorithmic Differentizites of C+ algorithmic devideout by B. Bell (max.coli-er.org/CDAB) General propse compression Library by J. Guily and K. Alger (Librar) OBF Mittgle Frectizion Arithmic Library averaigned by T. Grandum (coptile.org) Compression Distance and the start of the s
--------------------------------------------------------------------------------------------------------------------------------	-----------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------------

user parameter file <scip.set> not found - using default parameters

SCIP» SCIP» set load /nfs/OPTI/jenkins/workspace/SCIP_cmake_custombranch_testrun/ch enkins/workspace/SCIP cmake_custombranch_testrun/che

< custombranch testrun/check/../settiros/default.set

Lipitreeds - 1 CLDs set tising (log) SCDs set display free 100006 display free - 10000000000000000000000000000000000	trus/ch
<_custombranch_testrum/check/results201902141133_1/a	dm_time.short.1_linking.s
<k adm_time.short.1_linking.sc<="" results201902141133_1="" th=""><th>ipoptspx_bugfix_201902141</th></k>	ipoptspx_bugfix_201902141
<pre><m_time.short.l_linking.scipoptspx_bugfix_2019021411< pre=""></m_time.short.l_linking.scipoptspx_bugfix_2019021411<></pre>	
saved parameter file dbg.default.isd=PTLfjenkins/avarkpare/SCIP_cmake_custombranch_testrum <th>ults201902141133_1/adm_time.short.1_linking.scipoptspx_bugfix_20190214 /check/</th>	ults201902141133_1/adm_time.short.1_linking.scipoptspx_bugfix_20190214 /check/

read problem </nfs/0PTI/ienkins/workspace/SCIP cmake custombranch testrun/check/instances/CP/Linking.cip-



Linux optc-05-07 4.4.0-97-generic #120-Ubs @01 /nfs/0PTI/jenkins/workspace/SCIP_COMP+ Sun Nov 5 03:46:04 CET 2017 srun: defined options for program 'srun' srun: srun: user 'adm_timo' srun: gid + 3893 : /nfs/OPTI/ienkins/v srun: out 1 (default) srun: nodes (set) 7356103 (default) srun: partition srun: profile srun: job name SCIPlinking' srun: reservation (null) srun: wckey (null) srun: cpu freq min 4294967294 srun: cpu freg max 4294967294 srun: cpu freq gov srun: wait.for.switches : .1 srun: distribution unknown srun: cpu bind srun: nen bind default srun: verbose srun: slurmd debug srun: immediate false srun: label output srun: unbuffered IO srun: overcommit srun: checkpoint_dir : /var/slurm/checkpoi : (null) srun: conment : (null) srun: dependency srun: gos (null) srun: constraints men-per-node+6858M srun: geometry (null) Vec srun: network (null) srun: propagate NONE srun: prolog (null) srun: epilog (null) srun: mail type NONE srun: mail user (null) srun: task prolog (null) srun: task epilog srun: multi prop 00 srun: sockets-per-node : -2 srun: cores-per-socket srun: ntasks.ner.node srun: ntasks-per-socket : srun: ntasks-per-core -2 4294967294 srun: plane size srun: core-spec NA srun: power srun: sicp : gdb -batch-sile srun: remote command Srun: remote command : gdb -batch-site bt full --args /nfs/OPTI/jenkins/workspace srun: debug: propagating SLUMM PRID PROCE srun: debug: propagating UMASK=0022 auth plugin for Munge (http: srun: debug: srun: debug: requesting job 7356183, user srun: debug: cous 1, tasks 1, name odb, r Entering slurm step launch srun: debug: srun: debug: mpi type = (null) srun: debug: Using mpi/none srun: debug: Entering _msg_thr_create() srun: debug: initialized stdio listening

SCIP version 4.0.2

branching score function ('s'um, 'p'roduct, 'g'ustient # [type: char, advanced: TRUE, range: (spq), defailt: p]

branching score factor to weigh downward and upward ga # [type: real, advanced: TRUE, range: [0,1], default: 0, branching/scorefac = 0.367

should branching on binary variables be preferred? # [type: beel, advanced: FALSE, range: (TRUE,FALSE), def branching/preferbinary = FALSE

minimal relative distance of branching point to bounds
[type: real, advanced: FALSE, range: [0,0.5], default:
branching/clamp = 0.2

strategy for normalization of LP gain when updating ps # [type: char, advanced: FALSE, range: (dls), default: s

should updating pseudo costs for continuous variables
[type: bool, advanced: FALSE, range: (TRUE.FALSE), def
branching/delavascottpdate = TRUE

should pseudo costs be updated also in diving and prob # [type: beel, advanced: FALSE, range: (TRUE,FALSE), def branching/dixingpscost = TRUE

should all strong branching children be reparded even
[type: beel, advanced: TRUE, range: (TRUE,FALSE), defa
branching/forceallchildren = FALSE

child node to be regarded first during strong branchin # [type: char, advanced: TRUE, range: (aduh), default: a branching/firstabchild = a

should LP solutions during strong branching with propa # [type: beal, advanced: TRUE, range: (TRUE,FALSE), defa branching/checksol = TRUE

should LP solutions during strong branching with propa # [type: bool, advanced: TRUE, range: (TRUE,FALSE), defo branching/rewsdsDsol = TRUE

score adjustment near zero by adding epsilon (TRUE) or # [type: bool, advanced: TRUE, range: (TRUE,FALSE), defa branching/sumadjustscore = FALSE

should automatic tree compression after the presolving # [type: bod, advanced: TRUE, range: (TRUE,FALSE), defa compression/emails = FALSE

should coeflict analysis be enabled?
[type: bool, advanced: FALSE, range: (TRUE.FALSE), def
coeflict/reable = TRUE

should conflicts based on an old cutoff bound be remov # [type: beel, advanced: TRUE, range: (TRUE,FALSE), defm conflict/cleanboundexceedings = TRUE

should propagation coeflict analysis be used? # [type: bool, advanced: FALSE, range: (TRUE,FALSE), def coeflict/usegrop = TRUE

should infeasible LP conflict analysis be used? ('o'ff
[type: char, advanced: FALSE, range: (oc0b), default:
conflict/useinflp = b

should bound exceeding LP conflict analysis be used? (# Itype: cher, advanced: PALSE, range: (ocdb), default:

should infeasible/bound exceeding strong branching con # [type: bool, advanced: MALSE, range: (TRUE,FALSE), def conflict/usesb = TRUE

@Permutation 0 aSeed a @Settings default (BinName scipoptspx bugfix 201902141133/bin/scip SNodeLinit 2100000000 MenLinit 5000 ØThreads 1 @FeasTol default AQueue mip-dbg @Exclusive

Performance analysis										
Goal										
Rubberband	O Search	1 Unload	2 Help							
Kubberbanu										

Testruns 🗸

Repr. © A sepaaggr (03. Nov ® B default (14. Feb 1	17) 9)	Solv Solv SCIF SCIF	er LPSol 9 4.0.2 CPLE 9 6.0.2 SoPl	wer Ma X 12.7.1.0 de ex 4.0.2 op	ebug Short ptinized Short	iet Setting sepaaggr default	Seed Perm Tim 0 0 300 0 0 60.	elimit .0s 0s	Host opt-low mip-dbg	Commit Ti 03. Nov 2 14. Feb 2	mestamp 2017, 11:06 2019, 10:32	Uploader adm_timo adm_timo	Date Added 05. Nov 2017, 03:48 14. Feb 2019, 11:40
Details			🖬 Eval	uation		0:	Meta				盍 Setti	ngs	
Detailed data	a												
Search:												Тод	gle columns:
												N	odes ¢
Instance Name	Туре 🔅	Original Cons 🗄	Original Vars 🗄	Presolved Cons 🗄	$\textbf{Presolved Vars} ~ \boldsymbol{\varphi}$	Dual Bound 🗄	Primal Bound	🔅 Gap 🔅	Iterations (Nodes \Rightarrow	Solving Time φ	Total Time	🔅 Status 🔶
disj_conj_rail	CIP	63	53	56	50	1324.4355742297	1324.43557422	97 0.0	54.6	27	0.03	θ.	93 ok 🍵
egout	MBP	98	141	37	49	568.1007	568.10	97 0.0	54.6	1	0.02	θ.	92 ok
enigma	BP	21	100	21	100	0.0	0	0 0.0	3543.0	667	0.31	0.	31 ok
ex1266	CIP	96	181	155	150	16.3	16	3 0.0	521.6	21	1.03	1.	B3 ok
factor-mod-si6=307-B	CIP	505	648	485	529	3.0	3	0.0	15.6	1	0.05	θ.	B5 ok
findRoot	CIP	3	101	3	101	1.0	1	0.0	10.0	11	0.0	0	.0 ok
flugpl	IΡ	18	18	13	14	1201500.0	1201500	0 0.0	231.0	194	0.07	θ.	97 ok
gt2	IP	29	188	28	173	21166.0	21166	0.0	122.6	1	0.02	θ.	92 ok
j301_2	CIP	52	32	61	24	47.0	47	0.0	0.0	1	0.01	θ.	B1 ok
linking	CIP	2	16	1	12	2.0	2	0.0	0.6	1	0.0	0	.0 ok
lseu	BP	28	89	84	60	1120.0	1120	0 0.0	2983.6	302	0.67	θ.	67 ok
n3	MINLP	43	26	39	22	37.7999969901	37.79999699	91 0.0	178.6	9	0.16	θ.	16 ok
MANN_a9.clq	BP	72	45	72	45	16.0	16	0.0	89.6	6	0.02	θ.	B2 ok
ncf128-4-1	CIP	1832	2596	637	1566	14.0	14	0.0	759.0	1	0.17	0.	17 ok
ncf64-4-1	CIP	928	1308	336	795	10.0	10	0.0	597.6	41	0.15	θ.	15 ok
meanvarxsc	CIP	43	36	30	23	14.3692318921	14.36923189	0.0	66.6	5	0.16	θ.	<pre>16 fail_solution_infeasible</pre>
misc03	BP	96	160	95	138	3368.0	3360	0.0	1038.0	15	0.46	θ.,	46 ok
normalized-bs10_4_5	BP	97	66	31	20	-4.0	- 4	0.0	127.0	1	0.03	0.	03 ok
normalized-md50_25_5	BP	1564	1514	84	23	3.0	3	0 0.0	968.0	4	0.23	θ.	23 ok
normalized-md10_4_3	BP	72	62	8	7	2.0	2	0.0	4.6	1	0.0	0	.8 ok
normalized-t21218308	BP	101	169	73	169			0.0	666.0	5	0.32	θ.	32 ok
normalized-t23-100-0	CIP	166	233	159	277	4.0	4	0 0.0	2434.6	53	0.68	0.	68 ok
p0033	BP	16	33	13	28	3089.0	3889	0.0	58.0	1	0.01	θ.	91 ok

M. Miltenberger (ZIB), F. Schlösser (ZIB): SCIP: Develop, License, Test, Curate



Performance analysis Infrastructure









Performance analysis About IPET and RUBBERBAND



IPET

"Interactive Performance Evaluation Tools for Optimization Software"



"Interactive Performance Evaluation Tools for Optimization Software"

- Extract and evaluate data from logfiles
- Flexible and customizable
- Commandline tool (plus rudimentary GUI)



"Interactive Performance Evaluation Tools for Optimization Software"

- Extract and evaluate data from logfiles
- Flexible and customizable
- Commandline tool (plus rudimentary GUI)

Rubberband

is a flexible archiving platform for optimization benchmarks.



"Interactive Performance Evaluation Tools for Optimization Software"

- Extract and evaluate data from logfiles
- Flexible and customizable
- Commandline tool (plus rudimentary GUI)

Rubberband

is a flexible archiving platform for optimization benchmarks.

- Archives the logfiles in ELASTICSEARCH database
- Accessible (browser based, unique links)
- Standardized interface to IPET for benchmark evaluation



"Interactive Performance Evaluation Tools for Optimization Software"

- Extract and evaluate data from logfiles
- Flexible and customizable
- Commandline tool (plus rudimentary GUI)

Rubberband

is a flexible archiving platform for optimization benchmarks.

- Archives the logfiles in ELASTICSEARCH database
- Accessible (browser based, unique links)
- Standardized interface to IPET for benchmark evaluation

github.com/GregorCH/ipet/

github.com/ambros-gleixner/rubberband

Both python tools developed by members of the SCIP group at ZIB.



Rubberband





















Thank you for your attention! Questions?



Interested? Come visit our poster!