

# Contents

<b>1</b>	<b>Introduction</b>	<b>1</b>
<b>I</b>	<b>Fundamental algorithms</b>	<b>5</b>
<b>2</b>	<b>Foundations</b>	<b>7</b>
2.1	The linear programming problem and its computational forms . . . . .	7
2.2	Geometry . . . . .	9
2.3	LP Duality . . . . .	11
2.4	Basic solutions, feasibility, degeneracy and optimality . . . . .	13
<b>3</b>	<b>The Dual Simplex Method</b>	<b>17</b>
3.1	The Revised Dual Simplex Algorithm . . . . .	17
3.1.1	Basic idea . . . . .	17
3.1.2	Neighboring solutions . . . . .	18
3.1.3	Pricing . . . . .	19
3.1.4	Ratio test . . . . .	20
3.1.5	Basis change . . . . .	22
3.1.6	Algorithmic descriptions . . . . .	25
3.2	The Bound Flipping Ratio Test . . . . .	25
3.3	Dual steepest edge pricing . . . . .	30
3.4	Elaborated version of the Dual Simplex Algorithm . . . . .	34
<b>4</b>	<b>Dual Phase I Methods</b>	<b>37</b>
4.1	Introduction . . . . .	37
4.1.1	Big-M method . . . . .	37
4.1.2	Dual feasibility correction . . . . .	37
4.2	Minimizing the sum of dual infeasibilities . . . . .	38
4.2.1	Subproblem approach . . . . .	38
4.2.2	Algorithmic approach . . . . .	40
4.3	Artificial bounds . . . . .	45
4.4	Cost modification . . . . .	45
4.5	Pan's method . . . . .	47
<b>II</b>	<b>Computational techniques</b>	<b>49</b>

<b>5</b>	<b>Solving Systems of Linear Equations</b>	<b>51</b>
5.1	Introduction . . . . .	51
5.1.1	Product form of the inverse . . . . .	52
5.1.2	LU decomposition . . . . .	53
5.2	LU factorization . . . . .	54
5.3	LU update . . . . .	56
5.3.1	Forrest/Tomlin update . . . . .	57
5.3.2	Suhl/Suhl update . . . . .	59
5.4	Exploiting (hyper-)sparsity in FTran, BTran and LU-update . . . . .	61
5.4.1	Algorithms for sparse and hypersparse triangular systems . . . . .	61
5.4.2	FTran and BTran with Suhl/Suhl update . . . . .	65
<b>6</b>	<b>Numerical Stability and Degeneracy</b>	<b>67</b>
6.1	Introduction . . . . .	67
6.1.1	Numerical stability . . . . .	67
6.1.2	Degeneracy and cycling . . . . .	69
6.2	Techniques to ensure numerical stability . . . . .	70
6.2.1	Numerical tolerances . . . . .	70
6.2.2	Stabilizing ratio tests . . . . .	71
6.2.2.1	Modified standard ratio test . . . . .	72
6.2.2.2	Harris' ratio test . . . . .	73
6.2.2.3	Shifting . . . . .	74
6.2.2.4	Stabilizing bound flipping ratio test . . . . .	75
6.2.3	Refactorization, accuracy checks and stability control . . . . .	77
6.2.3.1	Refactorization for speed . . . . .	77
6.2.3.2	Refactorization for stability . . . . .	79
6.3	Techniques to reduce degeneracy and prevent cycling . . . . .	82
6.3.1	Perturbation . . . . .	82
6.3.2	Randomized pricing . . . . .	84
<b>7</b>	<b>Further computational aspects</b>	<b>85</b>
7.1	LP preprocessing, scaling and crash procedures . . . . .	85
7.2	Computation of the pivot row . . . . .	87
<b>III</b>	<b>Implementation and results</b>	<b>89</b>
<b>8</b>	<b>Implementation</b>	<b>91</b>
8.1	The Mathematical OPTimization System MOPS . . . . .	91
8.1.1	MOPS and its history . . . . .	91
8.1.2	External system architecture . . . . .	92
8.1.3	LP / MIP solution framework . . . . .	93
8.2	The dual simplex code . . . . .	96
8.2.1	Basic data structures . . . . .	96
8.2.2	Pricing . . . . .	98
8.2.2.1	Initialization and update of DSE weights . . . . .	98

---

8.2.2.2	Vector of primal infeasibilities . . . . .	100
8.2.2.3	Partial randomized pricing . . . . .	101
8.2.3	Ratio test . . . . .	101
8.2.4	FTran, BTran, LU-Update and factorization . . . . .	103
8.2.4.1	Data structures for the LU-factors . . . . .	104
8.2.4.2	Exploiting hypersparsity . . . . .	106
8.2.4.3	Forward Transformation (FTran) . . . . .	108
8.2.4.4	LU-update and factorization . . . . .	113
8.2.5	Overview . . . . .	114
<b>9</b>	<b>Numerical results</b>	<b>117</b>
9.1	Test problems . . . . .	117
9.2	Performance measures . . . . .	118
9.3	Study on dual phase 1 . . . . .	118
9.4	Chronological progress study . . . . .	121
9.5	Overall benchmarks . . . . .	124
<b>10</b>	<b>Summary and Conclusion</b>	<b>129</b>
	<b>Bibliography</b>	<b>131</b>
<b>A</b>	<b>Tables</b>	<b>137</b>

# List of Figures

2.1	A convex polyhedron in $\mathbb{R}^2$ .	9
5.1	Forrest/Tomlin update	58
5.2	Suhl/Suhl update	60
5.3	An upper triangular matrix and the corresponding nonzero graph $\mathcal{G}$	64
8.1	The external architecture of MOPS	94
8.2	LP/IP solution framework in MOPS	95
8.3	Columnwise compact storage for $\bar{A}$ before and after LP preprocessing.	96
8.4	Dense, indexed and packed storage for mathematical vectors.	97
8.5	Implementation of the bound flipping ratio test with Harris' tolerance.	102
8.6	Data structure for $\tilde{U}$ .	104
8.7	Data structure for the eta vectors of $\tilde{L}^{-1}$ (1).	105
8.8	Data structure for the eta vectors of $\tilde{L}^{-1}$ (2).	106
8.9	Data structure for depth-first-search.	107
8.10	Data-flow of FTran operation.	108
8.11	Implementation overview: initialization, refactorization and dual phase I.	115
8.12	Implementation overview: main loop.	116
9.1	Performance profile over phase 1 test set: solution time using four different dual phase 1 methods.	120
9.2	Performance profile over all test models: solution time with reduced and expanded bounds after LP preprocessing.	121
9.3	Chronological progress in the development process of the MOPS Dual Simplex code.	123
9.4	Overall benchmarks.	125

# List of Tables

2.1	Primal-dual transformation rules. . . . .	12
2.2	Dual feasibility conditions. . . . .	15
6.1	Tolerances: notation and default values. . . . .	71
8.1	History of MOPS development. . . . .	92
8.2	Improvement of MOPS LP optimization on model oil. . . . .	92
8.3	Improvement of MOPS IP optimization on model oil. . . . .	93
9.1	Benchmark of dual phase 1 methods. . . . .	119
9.2	Benchmark with reduced and expanded bounds after LP-preprocessing on original test set. . . . .	121
9.3	Progress in our dual simplex code: implementation techniques in chronological order. . . . .	122
9.4	System benchmark: solution time (including time for LP preprocess- ing) on 25 most difficult original models. . . . .	126
9.5	Benchmark of dual simplex codes (no LP preprocessing): solution time on 25 most difficult models pretreated by COIN LP preprocessing. . . . .	127
A.1	Model dimensions. . . . .	139
A.2	Results with Cplex 9.1 Dual Simplex. . . . .	141
A.3	Results with COIN LP 1.02.02 Dual Simplex. . . . .	143
A.4	Results with Soplex 1.2.1 Dual Simplex. . . . .	145
A.5	Results with MOPS 7.9 Dual Simplex. . . . .	147
A.6	Results with MOPS 7.9 Dual Simplex with expanded bounds after LP preprocessing. . . . .	149
A.7	Results on Dual Phase 1: Combined Method "Pan + Subproblem- Approach". . . . .	150
A.8	Results on Dual Phase 1: Pan's method. . . . .	151
A.9	Results on Dual Phase 1: Minimization of the sum of dual infeasibil- ities, subproblem approach. . . . .	152
A.10	Results on Dual Phase 1: Cost modification + Primal Simplex. . . . .	153
A.11	Progress in our dual simplex code: implementation techniques in chronological order. . . . .	154
A.12	Chronological progress: solution time. . . . .	155
A.13	Chronological progress: total iteration count. . . . .	156

# List of Algorithms

1	Basic steps of the dual simplex method. . . . .	17
2	Dual simplex method with simple ratio test and update of $d$ . . . . .	26
3	Dual simplex method with simple ratio test and update of $y$ . . . . .	27
4	Selection of $q$ with the BRFT. . . . .	31
5	Update of $d$ and $x_B$ for the BRFT. . . . .	31
6	Update of $x_N$ for the BRFT. . . . .	31
7	Dual simplex method with dual steepest edge pricing, bound flipping ratio test and update of $d$ . . . . .	35
8	The 2-phases dual simplex method with subproblem dual phase I. . . . .	40
9	Dual phase 1: Minimizing the sum of dual infeasibilities with simple ratio test. . . . .	43
10	Extended ratio test in dual phase I. . . . .	46
11	Dual phase 1: Pan's method. . . . .	48
12	LU-factorization. . . . .	55
13	LU-Update Suhl/Suhl (in terms of $\tilde{U}$ ) . . . . .	62
14	$Ux = b$ – dense method (for dense $b$ ) . . . . .	63
15	$Ux = b$ – sparse method (for sparse $b$ ) . . . . .	63
16	$Ux = b$ – hyper-sparse method (for very sparse $x$ ) . . . . .	65
17	Modified standard ratio test . . . . .	72
18	Bound flipping ratio test with Harris' tolerance. . . . .	78
19	HypersparsityTest . . . . .	107
20	FTran (Pseudocode) . . . . .	109
21	FTranL-F_sparse . . . . .	110
22	FTranL-U . . . . .	110
23	FTranU_sparse() . . . . .	111
24	FTranU_hyper . . . . .	111
25	FTranU_hyper_DFS . . . . .	112