

<b>Part II – Convex hulls</b>	<b>125</b>
3.1.2 An example: sorting a numbers using merge-sort	125
<b>Chapter 7 – Polytopes</b>	<b>127</b>
7.1 Definitions	127
3.2.1 Overview	127
3.2.2 An example: computing the intersections of line segments	127
3.3 Vertical decompositions	135
3.3.1 Vertical decomposition	135
3.3.2 Vertical decompositions and simplified decompositions	141
3.4 Exercises	141
3.5 Bibliographical notes	144
<b>Preface</b>	<b>xv</b>
<b>Translator's preface</b>	<b>xix</b>
<b>Acknowledgments</b>	<b>xxi</b>
<b>Part I – Algorithmic tools</b>	<b>1</b>
<b>Chapter 1 – Notions of complexity</b>	<b>3</b>
1.1 The complexity of algorithms	3
1.1.1 The model of computation	3
1.1.2 Notions of complexity	4
1.1.3 Asymptotic behavior, notation	7
1.2 Optimality, lower bounds	9
1.2.1 The complexity of a problem	9
1.2.2 The example of sorting: decision trees	10
1.2.3 Lower bounds by transforming one problem into another	11
1.3 Bibliographical notes	12
<b>Chapter 2 – Basic data structures</b>	<b>13</b>
2.1 Terminology and features of the basic data structures	14
2.1.1 Lists, heaps, and queues	14
2.1.2 Dictionaries and priority queues	15
2.2 Balanced search trees	16
2.2.1 Graphs, trees, balanced trees	16
2.2.2 Red-black trees	17
2.3 Dictionary on a finite universe	25
2.4 Exercises	26
2.5 Bibliographical notes	30
<b>Chapter 3 – Deterministic methods used in geometry</b>	<b>32</b>
3.1 The divide-and-conquer method	33
3.1.1 Overview	33

3.1.2	An example: sorting $n$ numbers using merge-sort . . . . .	35
3.2	The sweep method . . . . .	36
3.2.1	Overview . . . . .	36
3.2.2	An example: computing the intersections of line segments . .	37
3.3	Vertical decompositions . . . . .	40
3.3.1	Vertical decompositions of line segments . . . . .	40
3.3.2	Vertical decompositions and simplified decompositions . . . .	42
3.4	Exercises . . . . .	43
3.5	Bibliographical notes . . . . .	44
<b>Chapter 4 – Random sampling</b>		<b>46</b>
4.1	Definitions . . . . .	46
4.1.1	Objects, regions, and conflicts . . . . .	46
4.1.2	Random sampling . . . . .	49
4.2	Probabilistic theorems . . . . .	50
4.2.1	The sampling theorem . . . . .	51
4.2.2	The moment theorem . . . . .	55
4.3	Exercises . . . . .	57
4.4	Bibliographical notes . . . . .	62
<b>Chapter 5 – Randomized algorithms</b>		<b>63</b>
5.1	The randomized incremental method . . . . .	64
5.2	Off-line algorithms . . . . .	65
5.2.1	The conflict graph . . . . .	65
5.2.2	An example: vertical decomposition of line segments . . . . .	69
5.3	On-line algorithms . . . . .	75
5.3.1	The influence graph . . . . .	75
5.3.2	An example: vertical decomposition of line segments . . . . .	81
5.4	Accelerated incremental algorithms . . . . .	84
5.4.1	The general method . . . . .	84
5.4.2	An example: vertical decomposition of a polygon . . . . .	87
5.5	Exercises . . . . .	88
5.6	Bibliographical notes . . . . .	92
<b>Chapter 6 – Dynamic randomized algorithms</b>		<b>95</b>
6.1	The probabilistic model . . . . .	96
6.2	The augmented influence graph . . . . .	97
6.3	Randomized analysis of dynamic algorithms . . . . .	101
6.4	Dynamic decomposition of a set of line segments . . . . .	110
6.5	Exercises . . . . .	121
6.6	Bibliographical notes . . . . .	123

<b>Part II – Convex hulls</b>	<b>125</b>
<b>Chapter 7 – Polytopes</b>	<b>127</b>
7.1 Definitions	127
7.1.1 Convex hulls, polytopes	127
7.1.2 Faces of a polytope	128
7.1.3 Polarity, dual of a polytope	135
7.1.4 Simple and simplicial polytopes	140
7.2 The combinatorics of polytopes	141
7.2.1 Euler's relation	141
7.2.2 The Dehn–Sommerville relations	144
7.2.3 The upper bound theorem	145
7.2.4 Cyclic polytopes	147
7.3 Projective polytopes, unbounded polytopes	148
7.3.1 Projective spaces	148
7.3.2 Oriented projective spaces	153
7.3.3 Projective polytopes, unbounded polytopes	156
7.4 Exercises	162
7.5 Bibliographical notes	168
<b>Chapter 8 – Incremental convex hulls</b>	<b>169</b>
8.1 Representation of polytopes	170
8.2 Lower bounds	171
8.3 Geometric preliminaries	171
8.4 A deterministic algorithm	176
8.5 On-line convex hulls	180
8.6 Dynamic convex hulls	186
8.7 Exercises	195
8.8 Bibliographical notes	197
<b>Chapter 9 – Convex hulls in two and three dimensions</b>	<b>198</b>
9.1 Representation of 2- and 3-polytopes	199
9.2 Divide-and-conquer convex hulls in dimension 2	201
9.3 Divide-and-conquer convex hulls in dimension 3	205
9.4 Convex hull of a polygonal line	214
9.5 Exercises	219
9.6 Bibliographical notes	221
<b>Chapter 10 – Linear programming</b>	<b>223</b>
10.1 Definitions	224
10.2 Randomized linear programming	225
10.3 Convex hulls using a shelling	227
10.4 Exercises	236
10.5 Bibliographical notes	239

<b>Part III – Triangulations</b>	<b>241</b>
<b>Chapter 11 – Complexes and triangulations</b>	<b>243</b>
11.1 Definitions	243
11.1.1 Simplices, complexes	243
11.1.2 Topological balls and spheres, singularities	245
11.1.3 Triangulations	247
11.1.4 Polygons and polyhedra	248
11.2 Combinatorics of triangulations	250
11.2.1 Euler's relation for topological balls and spheres	250
11.2.2 The complexity of 2-complexes	251
11.2.3 The complexity of 3-triangulations	255
11.3 Representation of complexes, duality	258
11.4 Exercises	260
11.5 Bibliographical notes	262
<b>Chapter 12 – Triangulations in dimension 2</b>	<b>263</b>
12.1 Triangulation of a set of points	264
12.1.1 The complexity of computing a triangulation	264
12.1.2 An incremental algorithm	264
12.2 Constrained triangulations	266
12.3 Vertical decompositions and triangulations of a polygon	267
12.3.1 Lower bound	267
12.3.2 Triangulating monotone polygons	269
12.3.3 Vertical decomposition and triangulation of a polygon	274
12.4 Exercises	281
12.5 Bibliographical notes	287
<b>Chapter 13 – Triangulations in dimension 3</b>	<b>289</b>
13.1 Triangulation of a set of points	290
13.1.1 The size of a triangulation	290
13.1.2 The split theorem	293
13.1.3 An incremental algorithm	296
13.2 Constrained triangulations	300
13.3 Vertical and simplicial decompositions	302
13.3.1 Vertical decomposition of a polyhedral region	302
13.3.2 Simplicial decomposition of a polyhedron of genus 0	307
13.4 Exercises	317
13.5 Bibliographical notes	318
<b>Part IV – Arrangements</b>	<b>319</b>
<b>Chapter 14 – Arrangements of hyperplanes</b>	<b>321</b>
14.1 Definitions	321

14.2	Combinatorial properties	322
14.3	The zone theorem	325
14.4	Incremental construction of an arrangement	330
14.4.1	The case of dimension 2	330
14.4.2	The case of dimensions higher than 2	331
14.5	Levels in hyperplane arrangements	333
14.5.1	Definitions	333
14.5.2	Combinatorial properties of levels	335
14.5.3	Computing the first $k$ levels in an arrangement	335
14.6	Exercises	343
14.7	Bibliographical notes	350
<b>Chapter 15 – Arrangements of line segments in the plane</b>		<b>352</b>
15.1	Faces in an arrangement	353
15.2	Davenport–Schinzel sequences	353
15.3	The lower envelope of a set of functions	355
15.3.1	Complexity	356
15.3.2	Computing the lower envelope	358
15.4	A cell in an arrangement of line segments	358
15.4.1	Complexity	359
15.4.2	Computing a cell	362
15.5	Exercises	368
15.6	Bibliographical notes	371
<b>Chapter 16 – Arrangements of triangles</b>		<b>373</b>
16.1	Faces in an arrangement	374
16.2	Decomposing an arrangement of triangles	374
16.2.1	Vertical decomposition	375
16.2.2	Convex decomposition	377
16.3	The lower envelope of a set of triangles	379
16.3.1	Complexity	381
16.3.2	Vertical decomposition	383
16.3.3	Computing the lower envelope	384
16.4	A cell in an arrangement of triangles	390
16.4.1	Complexity	390
16.4.2	Vertical decomposition	394
16.4.3	Computing a cell	400
16.5	Exercises	402
16.6	Bibliographical notes	404
<b>Part V – Voronoi diagrams</b>		<b>405</b>
<b>Chapter 17 – Euclidean metric</b>		<b>407</b>
17.1	Definition	407

17.2	Voronoi diagrams and polytopes	408
17.2.1	Power of a point with respect to a sphere	408
17.2.2	Representation of spheres	409
17.2.3	The paraboloid $\mathcal{P}$	410
17.2.4	Polarity	411
17.2.5	Orthogonal spheres	412
17.2.6	Radical hyperplane	414
17.2.7	Voronoi diagrams	414
17.3	Delaunay complexes	416
17.3.1	Definition and connection with Voronoi diagrams	416
17.3.2	Delaunay triangulations	418
17.3.3	Characteristic properties	418
17.3.4	Optimality of Delaunay triangulations	421
17.4	Higher-order Voronoi diagrams	425
17.5	Exercises	428
17.6	Bibliographical notes	431
<b>Chapter 18 – Non-Euclidean metrics</b>		<b>433</b>
18.1	Power diagrams	434
18.1.1	Definition and computation	434
18.1.2	Higher-order power diagrams	436
18.2	Affine diagrams	437
18.2.1	Affine diagrams and power diagrams	437
18.2.2	Diagrams for a general quadratic distance	438
18.3	Weighted diagrams	439
18.3.1	Weighted diagrams with additive weights	439
18.3.2	Weighted diagrams with multiplicative weights	442
18.4	$L_1$ and $L_\infty$ metrics	445
18.5	Voronoi diagrams in hyperbolic spaces	449
18.5.1	Pencils of spheres	449
18.5.2	Voronoi diagrams in hyperbolic spaces	450
18.6	Exercises	454
18.7	Bibliographical notes	457
<b>Chapter 19 – Diagrams in the plane</b>		<b>459</b>
19.1	A sweep algorithm	459
19.2	Voronoi diagram of a set of line segments	464
19.2.1	Definition and basic properties	464
19.2.2	A sweep algorithm	470
19.2.3	An incremental algorithm	471
19.2.4	The case of connected segments	476
19.2.5	Application to the motion planning of a disk	479
19.3	The case of points distributed in two planes	481
19.3.1	The two planes are parallel	481

19.3.2 The two planes are not parallel .....	485
19.4 Exercises .....	488
19.5 Bibliographical notes .....	490
<b>References</b>	<b>492</b>
<b>Notation</b>	<b>508</b>
<b>Index</b>	<b>513</b>

## A new field

Many disciplines require a knowledge of how to efficiently deal with and build geometric objects. Among many examples, one could quote robotics, computer vision, computer graphics, medical-imaging, virtual reality, or computer aided design. The first geometric results with a constructive flavor date back to Euclid and remarkable developments occurred during the nineteenth century. However, only very recently did the design and analysis of geometric algorithms find a systematic treatment: this is the topic of computational geometry which as a field truly emerged in the mid 1970s. Since then, the field has undergone considerable growth, and is now a full-fledged scientific discipline, of which this text presents the foundations.

## Contents and layout of this book

The design of efficient geometric algorithms and their analysis are largely based on geometric structures, algorithmic data structuring techniques, and combinatorial results.

A major contribution of computational geometry is to exemplify the central role played by a small number of *fundamental geometric structures* and their relation to many geometric problems.

Geometric data structures and their systematic analysis guided the layout of this text. We have dedicated a part to each of the fundamental geometric structures: convex hulls, triangulations, arrangements, and Voronoi diagrams.

In order to control the complexity of an algorithm, one must know the complexity of the objects that it generates. For example, it is essential to have a sharp bound on the number of facets of a polytope as a function of the number of its vertices: this is the celebrated upper-bound theorem proved by McMullen in 1970. *Combinatorial geometry* plays an essential role in this book and the first chapters of each part lay the combinatorial grounds and prove the basic combinatorial properties satisfied by the corresponding geometric structures.