

# Search Methods in Artificial Intelligence

*Search Methods in Artificial Intelligence* is designed to provide in-depth knowledge on how search plays a fundamental role in problem solving. The book is meant for undergraduate and graduate students pursuing courses in computer science and artificial intelligence. Starting with basic search, it covers a variety of complex algorithms designed for different kinds of problems. It demonstrates that search is all-pervasive in artificial intelligence and equips the reader with relevant skills.

The text begins with an introduction to search spaces that confront intelligent agents. It illustrates how basic algorithms like depth first search and breadth first search run into exponentially growing spaces. Discussions on heuristic search follow along with stochastic local search, algorithm A\*, and problem decomposition. The role of search in playing board games, deduction in logic, automated planning, and machine learning is described next. The book concludes with a coverage of constraint satisfaction.

**Deepak Khemani** has been actively working in the field of artificial intelligence (AI) for over four decades – first as a student at Indian Institute of Technology (IIT) Bombay and then as a Professor in the Department of Computer Science and Engineering at IIT Madras. Currently he is Professor at Plaksha University, Mohali. He has three well-received courses on AI on SWAYAM, a MOOC platform launched by the Government of India. He is also the author of *A First Course in Artificial Intelligence* (2013).

# Search Methods in Artificial Intelligence

**Deepak Khemani**

Cambridge University Press & Assessment  
978-1-009-28432-5 — Search Methods in Artificial Intelligence  
Deepak Khemani  
Frontmatter  
[More Information](#)



Shaftesbury Road, Cambridge CB2 8EA, United Kingdom  
One Liberty Plaza, 20th Floor, New York, NY 10006, USA  
477 Williamstown Road, Port Melbourne, VIC 3207, Australia  
314-321, 3rd Floor, Plot 3, Splendor Forum, Jasola District Centre, New Delhi – 110025, India  
103 Penang Road, #05-06/07, Visioncrest Commercial, Singapore 238467

Cambridge University Press is part of Cambridge University Press & Assessment, a department of the University of Cambridge.

We share the University's mission to contribute to society through the pursuit of education, learning and research at the highest international levels of excellence.

[www.cambridge.org](http://www.cambridge.org)  
Information on this title: [www.cambridge.org/9781009284325](http://www.cambridge.org/9781009284325)

© Deepak Khemani 2024

This publication is in copyright. Subject to statutory exception and to the provisions of relevant collective licensing agreements, no reproduction of any part may take place without the written permission of Cambridge University Press & Assessment.

First published 2024

*A catalogue record for this publication is available from the British Library*

ISBN 978-1-009-28432-5 Hardback  
ISBN 978-1-009-28433-2 Paperback

Cambridge University Press & Assessment has no responsibility for the persistence or accuracy of URLs for external or third-party internet websites referred to in this publication and does not guarantee that any content on such websites is, or will remain, accurate or appropriate.

Contents

<b>Preface</b>	<b>xi</b>
<b>Acknowledgements</b>	<b>xiii</b>
<b>1 Introduction</b>	<b>1</b>
1.1 Can Machines Think?	1
1.2 Problem Solving	4
1.3 Neural Networks	6
1.3.1 Deep neural networks	8
1.4 Symbolic AI	10
1.4.1 Symbols, language, and knowledge	10
1.4.2 Symbol systems	12
1.4.3 An architecture for cognition	13
1.5 The Core of Intelligence	14
1.5.1 Remember the past and learn from it	15
1.5.2 Understand and represent the present	18
1.5.3 Imagine the future and shape it	23
<i>A Note for the Reader</i>	26
<i>Exercises</i>	26
<b>2 Search Spaces</b>	<b>29</b>
2.1 The State Space	30
2.1.1 Generate and test	31
2.2 Search Spaces	32
2.3 Configuration Problems	33
2.3.1 The map colouring problem	33
2.3.2 The N-queens problem	34
2.3.3 The SAT problem	35
2.4 Planning Problems	35
2.4.1 The 8-puzzle	36
2.4.2 River crossing puzzles	37
2.4.3 The water jug problem	38
2.4.4 The travelling salesman problem	39
2.5 The Solution Space	42
2.5.1 Constructive methods	42
2.5.2 Perturbative methods	43
<i>Summary</i>	44
<i>Exercises</i>	44

<b>3</b>	<b>Blind Search</b>	<b>47</b>
3.1	Search Trees	48
3.2	Depth First Search	49
3.2.1	How to stop going around in circles	51
3.2.2	Reconstructing the path	53
3.2.3	The complete algorithm	55
3.2.4	Backtracking in DFS	57
3.3	Breadth First Search	58
3.4	Comparing DFS and BFS	60
3.4.1	Space complexity	61
3.4.2	Time complexity	62
3.4.3	Quality of solution	64
3.4.4	Completeness	64
3.5	Depth First Iterative Deepening	64
3.5.1	Depth bounded depth first search	65
3.5.2	Depth first iterative deepening	66
3.5.3	Space complexity	67
3.5.4	Time complexity	67
3.5.5	Quality of solution	67
3.5.6	Completeness	69
3.6	Uninformed Search	70
	<i>Summary</i>	71
	<i>Exercises</i>	72
<b>4</b>	<b>Heuristic Search</b>	<b>75</b>
4.1	Heuristic Functions	76
4.1.1	Map colouring	77
4.1.2	SAT	77
4.1.3	The 8-puzzle	77
4.1.4	Route finding	78
4.1.5	Travelling salesperson problem	80
4.2	Best First Search	81
4.2.1	Quality of solution	84
4.2.2	Completeness	85
4.2.3	Space complexity	85
4.2.4	Time complexity	86
4.3	Local Search Methods	86
4.3.1	An optimization problem	87
4.3.2	Hill climbing	88
4.3.3	Completeness	89
4.3.4	Quality of solution	89
4.3.5	Time complexity	89
4.3.6	Space complexity	90
4.4	Heuristic Search Terrains	90
4.4.1	Hill climbing in the blocks world domain	90
4.4.2	Heuristic functions	91
4.4.3	The SAT landscape	95

4.5	Neighbourhood Functions	96
4.5.1	Neighbourhood functions for the TSP	96
4.5.2	Neighbourhood functions for SAT	98
4.5.3	Variable neighbourhood descent	99
4.5.4	Beam search	101
4.6	Escaping Local Optima	103
4.6.1	Exploration versus exploitation	104
4.6.2	Tabu search	104
4.6.3	Iterated hill climbing	107
	<i>Summary</i>	108
	<i>Exercises</i>	108
<b>5</b>	<b>Stochastic Local Search</b>	<b>115</b>
5.1	Climbing Mount Improbable	116
5.1.1	Random walk: pure exploration	117
5.1.2	Stochastic hill climbing: calibrated randomness	117
5.1.3	Simulated annealing: controlled randomness	119
5.2	Evolution: Survival of the Fittest	122
5.2.1	Genetic algorithms: churning in a population of candidates	125
5.2.2	TSP: Representations and crossover operators	130
5.2.2.1	TSP: path representation	131
5.2.2.2	TSP: adjacency representation	134
5.2.2.3	TSP: ordinal representation	136
5.3	Swarm Intelligence: The Power of Many	138
5.3.1	Ant colony optimization	141
	<i>Summary</i>	144
	<i>Exercises</i>	144
<b>6</b>	<b>Algorithm A* and Variations</b>	<b>147</b>
6.1	Branch & Bound	148
6.1.1	Performance of B&B	152
6.1.2	B&B on the TSP	153
6.1.3	Higher estimates are better estimates	156
6.2	Algorithm A*	156
6.2.1	Dijkstra’s algorithm	156
6.2.2	A*	158
6.2.3	A* is admissible	162
6.2.4	Proof of admissibility	163
6.2.5	Higher is better	164
6.2.6	The monotone condition	166
6.2.7	Performance of A*	167
6.2.8	Weighted A*	167
6.3	Space Saving Versions of A*	169
6.3.1	Iterative deepening A*	169
6.3.2	Recursive best first search	170
6.3.3	Sequence alignment	171

6.3.4	Pruning CLOSED	173
6.3.4.1	Divide and conquer frontier search	173
6.3.4.2	Smart memory graph search	175
6.3.5	Beam stack search	176
6.3.6	Pruning OPEN and CLOSED	178
	<i>Summary</i>	179
	<i>Exercises</i>	180
<b>7</b>	<b>Problem Decomposition</b>	<b>185</b>
7.1	Pattern Directed Inference Systems	185
7.2	Rule Based Production Systems	189
7.2.1	The working memory in OPS5	190
7.2.2	Patterns in rules	191
7.2.3	Actions in rules	192
7.2.4	The inference engine	194
7.2.5	Conflict resolution strategies	195
7.2.6	The Rete net	197
7.2.7	The Rete algorithm	201
7.3	Problem Decomposition with And–Or Graphs	205
7.3.1	DENDRAL	206
7.3.2	Symbolic integration	209
7.3.3	Algorithm AO*	211
	<i>Summary</i>	216
	<i>Exercises</i>	216
<b>8</b>	<b>Chess and Other Games</b>	<b>221</b>
8.1	Game Theory	223
8.1.1	The prisoner’s dilemma	224
8.1.2	Types of games	225
8.2	Board Games	227
8.2.1	Strategies	229
8.2.2	Limited lookahead with the evaluation function	231
8.2.3	Algorithm Minimax	235
8.2.4	AlphaBeta pruning	236
8.2.5	Algorithm SSS*: best first search	239
8.2.6	Algorithm SSS*: an iterative version	243
8.2.7	AlphaGo, AlphaGo Zero, and AlphaZero	246
8.3	Backgammon and Scrabble: Elements of Chance	247
8.4	Contract Bridge: A Challenge for AI	250
	<i>Summary</i>	258
	<i>Exercises</i>	258
<b>9</b>	<b>Automated Planning</b>	<b>263</b>
9.1	Representation	264
9.1.1	Time and change	264
9.2	Simple Planning Domains	266
9.2.1	STRIPS	266
9.2.2	The blocks world domain	268

9.3	State Space Planning	272
9.3.1	Forward state space planning	273
9.3.2	Domain independent heuristics	274
9.3.3	Backward state space planning	277
9.4	Goal Stack Planning	280
9.4.1	Linear planning	281
9.4.2	Sussman anomaly	284
9.5	Partial Order Planning	286
9.5.1	A two armed robot	293
9.6	Algorithm Graphplan	295
9.6.1	The planning graph	295
9.6.2	Heuristics and solutions from the planning graph	300
9.7	Planning as Satisfiability	302
9.7.1	Direct encoding	303
9.7.2	The planning graph as SAT	306
9.8	Richer Planning Domains	307
9.8.1	Durative actions	308
9.8.2	Metric domains	310
9.8.3	Conditional effects	310
9.8.4	Contingent planning	311
9.8.5	Trajectory constraints and preferences	312
9.8.6	Coordination in multi-agent systems	312
9.8.7	Epistemic planning	313
	<i>Summary</i>	315
	<i>Exercises</i>	315
<b>10</b>	<b>Deduction as Search</b>	<b>319</b>
10.1	Logical Connectives	320
10.1.1	Truth tables	321
10.2	Entailment and Proof	323
10.2.1	Soundness	324
10.2.2	Completeness	326
10.3	First Order Logic	328
10.3.1	Terms and domains	329
10.3.2	Atomic formulas	330
10.3.3	Quantifiers, formulas, and sentences	331
10.4	Deduction in FOL	333
10.4.1	Implicit quantifier form	335
10.4.2	Unification	337
10.4.3	Forward chaining	339
10.4.4	Backward chaining and deductive retrieval	341
10.4.5	Prolog	343
10.4.6	Incompleteness of forward and backward chaining	346
10.4.7	The resolution refutation method	347
10.4.8	FOL with equality	351



x | Contents

10.5	The Family of Logics	355
10.5.1	Horn clause logic and Prolog	356
10.5.2	Description logics	357
10.5.3	Default reasoning	359
10.5.4	Event calculus	360
10.5.5	Epistemic logic	361
	<i>Summary</i>	362
	<i>Exercises</i>	362
<b>11</b>	<b>Search in Machine Learning</b> by Sutanu Chakraborti	<b>367</b>
11.1	Decision Trees	371
11.2	<i>k</i> -Nearest Neighbour	376
11.3	Bayesian Classification	378
11.4	Artificial Neural Networks	382
11.5	<i>K</i> -MEANS Clustering	389
	<i>Summary</i>	391
	<i>Exercises</i>	391
<b>12</b>	<b>Constraint Satisfaction</b>	<b>393</b>
12.1	Constraints: Clearing the Fog	394
12.1.1	The map colouring problem	394
12.1.2	The <i>N</i> -queens puzzle	395
12.2	Algorithm BACKTRACKING	398
12.2.1	Static variable ordering	400
12.2.2	Dynamic variable ordering	402
12.3	Constraint Propagation	403
12.3.1	Arc consistency	404
12.3.2	The Waltz algorithm	408
12.3.3	Path consistency	411
12.3.4	<i>i</i> -Consistency	414
12.3.5	Directional consistency	415
12.4	Lookahead Search	417
12.4.1	Algorithm forward checking	419
12.4.2	Algorithm DAC-Lookahead	422
12.4.3	Algorithm AC-Lookahead	425
12.5	Informed Backtracking	426
12.5.1	Gaschnig’s backjumping	427
12.5.2	Graph based backjumping	429
12.5.3	Conflict directed backjumping	432
	<i>Summary</i>	434
	<i>Exercises</i>	435
	<b>Appendix: Algorithm and Pseudocode Conventions</b> by S. Baskaran	<b>441</b>
	<b>References</b>	<b>449</b>
	<b>Index</b>	<b>461</b>

## Preface

This book is meant for the serious practitioner-to-be of constructing intelligent machines. Machines that are aware of the world around them, that have goals to achieve, and the ability to imagine the future and make appropriate choices to achieve those goals. It is an introduction to a fundamental building block of artificial intelligence (AI). As the book shows, search is central to intelligence.

Clearly AI is not one monolithic algorithm but a collection of processes working in tandem, an idea espoused by Marvin Minsky in his book *The Society of Mind* (1986). Human problem solving has three critical components. The ability to make use of experiences stored in memory; the ability to reason and make inferences from what one knows; and the ability to search through the space of possibilities. This book focuses on the last of these. In the real world we sense the world using vision, sound, touch, and smell. An autonomous agent will need to be able to do so as well. Language, and the written word, is perhaps a distinguishing feature of the human species. It is the key to communication which means that human knowledge becomes pervasive and is shared with future generations. The development of mathematical sciences has sharpened our understanding of the world and allows us to compute probabilities over choices to take calculated risks. All these abilities and more are needed by an autonomous agent.

Can one massive neural network be the embodiment of AI? Certainly, the human brain as a seat of intelligence suggests that. Everything we humans do has its origin in activity in our brains, which we call the mind. Perched on the banks of a stream in the mountains we perceive the world around us and derive a sense of joy and well-being. In a fit of contented creativity, we may pen an essay or a poem using our faculty of language. We may call a friend on the phone and describe the scene around us, allowing the friend to visualize the serene surroundings. She may reflect upon her own experiences and recall a holiday she had on the beach. You might start humming your favourite song and then be suddenly jolted out of your reverie remembering that friends are coming over for dinner. You get up and head towards your home with cooking plans brewing in your head.

So, in principle at least one can imagine a massive neural network that could do all the above. But how would it be implemented? What kind of a training process would instil all such knowledge and memories in the neural brain? Human beings go through a lifetime of learning. A human baby, unlike a fawn, is an utterly helpless creature and needs to be nurtured for years. Taught in schools, influenced by peer groups, moulded through culture and religion, coached in sports. Every human is said to be unique, even identical twins. We celebrate this diversity, even when it is sometimes a source of crime and conflict. Are we ready for idiosyncratic machines? Or do we aim for identical assembly line robots? But what or who would they be like? And what about issues of fairness? And generation of harmful or misleading content?

The twenty-first century has seen an explosion in machine learning as exemplified by deep neural networks which outperform humans on many classification tasks, and large language models that can generate an essay, a college application, or a poem in a jiffy. Massive computing power and humungous amounts of data have made this possible. It has been very impressive, but has it peaked? Do we need to move on and seek another path to the Holy Grail, machines which autonomously solve problems for us? Instead of blanket ingestion of all data on the internet, perhaps we need to build machines which learn from human expertise to become experts in specific domains. And do useful things for us.

This book is a step in that direction. It is designed to be a complete guide to one specific aspect of problem solving – the use of search methods. It is intended to be one in-depth module for the task of building AI, and its contents can be covered in a one semester course. We begin by learning to walk with small problems, and gradually build a repertoire of search algorithms that would allow us to navigate the high seas and vast deserts. The algorithms are general purpose, but our representations are tailormade for the individual domains. We urge the interested reader to implement the algorithms described here and develop a suite of search algorithms that can be used to solve specific problems.

One common feature in all these algorithms is that they operate on symbolic data, where symbols stand for things meaningful to us, and algorithms operate upon them. This approach is, as *hypothesized* by Herbert Simon and Alan Newell in 1976, both necessary and sufficient to create AI.

Maybe one day these many algorithms and the different problems they solve will come together in an integrated entity as a step towards artificial general intelligence. But that will need advances in knowledge representation where different domains and problems can be uniformly expressed in a common language. There is work still ahead for us.

## Acknowledgements

Several people have contributed in myriad ways to this book, some directly and some indirectly. Many students in my class, both online and offline, have triggered a thought process by asking incisive questions and making insightful observations. I gratefully acknowledge all of them collectively. They have made the job of teaching and learning rewarding.

Baskaran Sankaranarayanan has been involved with my courses over the last few years, answering student queries, helping with question papers and figures, and most importantly by standardizing the way in algorithms are written in pseudo code. He has written the appendix in this book on algorithm style, and the algorithms in the book conform to that style.

Sutanu Chakraborti has been a long-time collaborator working in AI. He wrote a chapter on natural language processing in my previous book, *A First Course in Artificial Intelligence*. In this book he has written one chapter on machine learning, despite a pressing schedule.

I am indebted to the following who have read and commented upon parts of the book – Nitin Dhiman, Aditi Khemani, Kamal Lodaya, Adwait Pramod Parsodkar, Devika Sethi, and Shashank Shekhar.

I am grateful to the team at CUP for their constant support from the very beginning. Vaishali Thapliyal took up my book proposal with gusto and got reviews from external reviewers expeditiously, yielding some very valuable suggestions and feedback. When the manuscript was ready Vikash Tiwari and Ankush Kumar initiated the production process immediately. Aniruddha De and Karan Gupta have done an excellent job with proofreading and copy editing, ironing out many discrepancies and bringing uniformity to the writing style.

Finally, I would like to thank the friends and family who have supported the book writing in many ways.