游戏开发中的人工智能(影印版)

AI for Game Developers

O'REILLY®
東南大學出版社

David M. Bourg & Glenn Seemann 著

O'REILLY®

游戏开发中的人工智能(影印版)

你的游戏是否有角色不能任意走动？是否有角色走进障碍物？是否有非玩家角色不能按照团队运动？现在你就可以掌握高级人工智能（AI）技术以解决这些问题。不管你是编程新手或者是个仅仅想快速学习AI的熟练游戏编程人员，你都会发现《游戏开发中的人工智能》对于理解并应用AI到你的游戏中是非常合适的入门书籍。本书正是为你提供游戏开发方面高级、有用的AI技术的。如果你曾试图使用AI延长你的游戏的生命周期，让你的游戏更加具有挑战性，更重要的是让它们更加有趣，这本书就是为你准备的。

David M. Bourg（畅销书《游戏开发中的物理学》的作者）和Glenn Seemann将用非常直观、易懂的语言给你介绍一些诸如有限状态机、模糊逻辑和神经网络之类的技术，全书使用源代码（用C和C++编写）说明这些技术。从基本的诸如追赶、躲避、基于模式的运动和聚集等游戏行为到玩家行为预测，这本书告诉你怎么应用AI给你的游戏角色提供可信的智能。这些技术包括了适合初级AI开发者的确定性（传统的）和非确定性（较新的）AI技术的混合。

其他的主题包括:

- 使用基于势函数的单一技术处理追赶、躲避、聚集和避障等问题。
- 使用包括路径点和经典A*算法解决寻径问题。
- 利用AI脚本扩充AI引擎的功能，让设计者和玩家更好地设计和玩游戏。
- 给你的游戏角色赋予基于规则的AI推理能力，包括模糊逻辑和有限状态机。
- 使用概率分析和诸如贝叶斯推理的高级技术处理不确定性问题。

ISBN 7-5641-0049-4

9 787564 100490 >

Visit O'Reilly on the Web at *www.oreilly.com*

ISBN 7-5641-0049-4

定价：78.00元

O'Reilly Media, Inc. 授权东南大学出版社出版

游戏开发中的人工智能(影印版)

AI for Game Developers

David M. Bourg & Glenn Seemann

O'REILLY®

Beijing • Cambridge • Farnham • Köln • Paris • Sebastopol • Taipei • Tokyo

O'Reilly Media, Inc. 授权东南大学出版社出版

东南大学出版社

图书在版编目(CIP)数据

游戏开发中的人工智能/(美)布尔格(Bourg,D. M.),(美)西曼(Seemann,G.)著—影印本. —南京:东南大学出版社,2005. 5

书名原文:AI for Game Developers

ISBN 7—5641—0049—4

Ⅰ.游... Ⅱ.①布... ②西... Ⅲ.人工智能—应用—游戏—软件开发—英文 Ⅳ.①TP311.52 ②TP18

中国版本图书馆 CIP 数据核字(2005)第 046962 号

江苏省版权局著作权合同登记

图字:10—2005—075 号

书　　名/ 游戏开发中的人工智能(影印版)
书　　号/ ISBN 7—5641—0049—4/TP·10
责任编辑/ 张烨
封面设计/ Ellie Volckhausen,张健
出版发行/ 东南大学出版社
地　　址/ 南京四牌楼 2 号　邮编 210096
经　　销/ 各地新华书店
印　　刷/ 扬中市印刷有限公司
开　　本/ 787 毫米×1092 毫米　16 开本　24.5 印张
版　　次/ 2005 年 6 月第 1 版　2005 年 6 月第 1 次印刷
印　　数/ 0001—2000 册
定　　价/ 78.00 元(册)

O'Reilly Media，Inc. 介绍

O'Reilly Media，Inc. 是世界上在 UNIX、X、Internet 和其他开放系统图书领域具有领导地位的出版公司，同时是联机出版的先锋。

从最畅销的 The Whole Internet User's Guide & Catalog（被纽约公共图书馆评为 20 世纪最重要的 50 本书之一）到 GNN（最早的 Internet 门户和商业网站），再到 WebSite（第一个桌面 PC 的 Web 服务器软件），O'Reilly Media，Inc. 一直处于 Internet 发展的最前沿。

许多书店的反馈表明，O'Reilly Media，Inc. 是最稳定的计算机图书出版商——每一本书都一版再版。与大多数计算机图书出版商相比，O'Reilly Media，Inc. 具有深厚的计算机专业背景，这使得 O'Reilly Media，Inc. 形成了一个非常不同于其他出版商的出版方针。O'Reilly Media，Inc. 所有的编辑人员以前都是程序员，或者是顶尖级的技术专家。O'Reilly Media，Inc. 还有许多固定的作者群体 —— 他们本身是相关领域的技术专家、咨询专家，而现在编写著作，O'Reilly Media，Inc. 依靠他们及时地推出图书。因为 O'Reilly Media，Inc. 紧密地与计算机业界联系着，所以 O'Reilly Media，Inc. 知道市场上真正需要什么图书。

出版说明

随着计算机技术的成熟和广泛应用，人类正在步入一个技术迅猛发展的新时期。计算机技术的发展给人们的工业生产、商业活动和日常生活都带来了巨大的影响。然而，计算机领域的技术更新速度之快也是众所周知的，为了帮助国内技术人员在第一时间了解国外最新的技术，东南大学出版社和美国 O'Reilly Media，Inc. 达成协议，将陆续引进该公司的代表前沿技术或者在某专项领域享有盛名的著作，以影印版或者简体中文版的形式呈献给读者。其中，影印版书籍，力求与国外图书"同步"出版，并且"原汁原味"展现给读者。

我们真诚地希望，所引进的书籍能对国内相关行业的技术人员、科研机构的研究人员和高校师生的学习和工作有所帮助，对国内计算机技术的发展有所促进。也衷心期望读者提出宝贵的意见和建议。

第一批影印图书共 10 本，涉及 Java，Unix/Linux，Python 等方面：

- 《核心 Java(影印版)》
- 《Jakarta Commons 经典实例(影印版)》
- 《Weblogic 权威指南(影印版)》
- 《Java 网络编程　第三版(影印版)》
- 《Linux 设备驱动程序　第三版(影印版)》
- 《LPI Linux 认证权威指南(影印版)》
- 《GNU Make 项目管理　第三版(影印版)》
- 《游戏开发中的人工智能(影印版)》
- 《学习 Python　第二版(影印版)》
- 《精通正则表达式　第二版(影印版)》

Preface

Recent advances in 3D visualization and physics-based simulation technology, at both the software and hardware levels, have enabled game developers to create compelling, visually immersive gaming environments. The next step in creating even more immersive games is improved artificial intelligence (AI). Advances in computing power, and in hardware-accelerated graphics in particular, are helping to free up more CPU cycles that can be devoted to more sophisticated AI engines for games. Further, the large number of resources—academic papers, books, game industry articles, and web sites—devoted to AI are helping to put advanced AI techniques within the grasp of every game developer, not just those professionals who devote their careers to AI.

With that said, wading through volumes of technical papers, text books, and web sites can be a daunting task for upcoming game AI developers. This book pulls together the information novices need so that they can get a jump-start in the field of game AI development. We present relevant theory on a wide range of topics, which we support with code samples throughout the book.

Many general game development books cover AI to some extent, however their treatment of the technology tends to be limited. This is probably because such books have to cover a lot of different topics and cannot go into great depth on any particular one. Although several very good books do focus on game AI (we list many of them in the "Additional Resources" section of this Preface), most of them are geared toward experienced AI developers and they focus on relatively specific and advanced topics. Therefore, novices likely would require companion resources that cover some of the more fundamental aspects of game AI in more detail. Still other books cover some specific game AI techniques in great detail, but are restricted to covering just those techniques.

Our book covers a wide range of game AI topics at a level appropriate for novice developers. So, if you are new to game programming or if you are an experienced

game programmer who needs to get up to speed quickly on AI techniques such as finite state machines, fuzzy logic, and neural networks, among others, this book is for you.

Assumptions This Book Makes

Because this book is targeted for beginner game AI developers, we don't assume you have any AI background. We do, however, assume you know how to program using C/C++. We also assume you have a working knowledge of the basic vector math used in games, but we have included a brief vector math refresher in the Appendix in case your skills are a little rusty.

About This Book

We didn't hope to (nor did we attempt to) cover every aspect of game AI in this book; far too many techniques and variations of techniques are used for an even larger variety of game types, specific game architectures, and in-game scenarios. Instead, we present a mix of both deterministic (traditional) and nondeterministic (newer) AI techniques aimed squarely at beginner AI developers. Here's a summary of what we cover:

Chapter 1, *Introduction to Game AI*

Here, we define game AI and discuss the current state of the art as well as the future of this technology.

Chapter 2, *Chasing and Evading*

We cover basic techniques for chasing and evading as well as more advanced techniques for intercepting. We also cover techniques applicable to both tile-based and continuous game environments.

Chapter 3, *Pattern Movement*

Pattern movement techniques are common to many video games and developers have been using them since the early days of gaming. You can use these techniques to preprogram certain behaviors such as the patrolling of a guard or the swooping in of a spacecraft.

Chapter 4, *Flocking*

The flocking method we examine in this chapter is an example of an A-life algorithm. In addition to creating cool-looking flocking behavior, A-life algorithms form the basis of more advanced group movement.

Chapter 5, *Potential Function Based Movement*

Potential-based movement is relatively new in game AI applications. The cool thing about this method is that it can handle chasing, evading, swarming, and collision avoidance simultaneously.

Chapter 6, *Basic Pathfinding and Waypoints*

Game developers use many techniques to find paths in and around game environments. In this chapter, we cover several of these methods, including waypoints.

Chapter 7, *A* Pathfinding*

No treatment of pathfinding is complete without addressing the workhorse algorithm of pathfinding; therefore, we devote this whole chapter to the A* algorithm.

Chapter 8, *Scripted AI and Scripting Engines*

Programmers today often write scripting engines and hand off the tools to level designers who are responsible for creating the content and defining the AI. In this chapter, we explore some of the techniques developers use to apply a scripting system in their games, and the benefits they receive.

Chapter 9, *Finite State Machines*

Finite state machines are the nuts and bolts of game AI. This chapter discusses the fundamentals of finite state machines and how to implement them.

Chapter 10, *Fuzzy Logic*

Developers use fuzzy logic in conjunction with or as a replacement for finite state machines. In this chapter, you'll learn the advantages fuzzy techniques offer over traditional logic techniques.

Chapter 11, *Rule-Based AI*

Technically, fuzzy logic and finite state machines fall under the general heading of rules-based methods. In this chapter, we cover these methods as well as other variants.

Chapter 12, *Basic Probability*

Game developers commonly use basic probability to make their games less predictable. Such cheap unpredictability enables developers to maintain substantial control over their games. Here, we cover basic probability for this purpose as well as lay the groundwork for more advanced methods.

Chapter 13, *Decisions Under Uncertainty—Bayesian Techniques*

Bayesian techniques are probabilistic techniques, and in this chapter we show how you can use them for decision making and for adaptation in games.

Chapter 14, *Neural Networks*

Game developers use neural networks for learning and adaptation in games—in fact, for anything from making decisions to predicting the behavior of players. We cover the most widely used neural network architecture, in detail.

Chapter 15, *Genetic Algorithms*

Genetic algorithms offer opportunities for evolving game AI. Although developers don't often use genetic algorithms in games, their potential for specific applications is promising, particularly if they are combined with other methods.

Appendix, *Vector Operations*

This appendix shows you how to implement a C++ class that captures all of the vector operations that you'll need when writing 2D or 3D simulations.

All the chapters in this book are fairly independent of each other. Therefore, you generally can read the chapters in any order you want, without worrying about missing material in earlier chapters. The only exception to this rule is Chapter 12, on basic probability. If you don't have a background in probability, you should read this chapter before reading Chapter 13, on Bayesian methods.

Also, we encourage you to try these algorithms for yourself in your own programs. If you're just getting started in game AI, which we assume you are if you're reading this book, you might want to begin by applying some of the techniques we present in simple arcade-style or board games. You also might consider programming a *bot* using extensible AI tools that are increasingly becoming standard for first-person shooter games. This approach will give you the opportunity to try out your AI ideas without having to program all the other non-AI aspects of your game.

Conventions Used in This Book

The following typographical conventions are used in this book:

Plain text

Indicates menu titles, menu options, menu buttons, and keyboard accelerators (such as Alt and Ctrl).

Italic

Indicates new terms, URLs, email addresses, filenames, file extensions, pathnames, directories, and Unix utilities.

`Constant width`

Indicates commands, options, switches, variables, attributes, keys, functions, types, classes, namespaces, methods, modules, properties, parameters, values, objects, events, event handlers, XML tags, HTML tags, macros, the contents of files, or the output from commands.

`Constant width bold`

Shows commands or other text that should be typed literally by the user.

Constant width italic

Shows text that should be replaced with user-supplied values.

Bold

Variables shown in bold are vectors as opposed to scalar variables, which are shown in regular print.

Additional Resources

Although we attempt to cover a wide range of AI techniques in this book, we realize we can't compress within these pages everything there is to know about AI in game development. Therefore, we've compiled a short list of useful AI web and print resources for you to explore should you decide to pursue game AI further.

Here are some popular web sites related to game development and AI that we find helpful:

- The Game AI Page at *http://www.gameai.com*
- AI Guru at *http://www.aiguru.com*
- Gamasutra at *http://www.gamasutra.com*
- GameDev.net at *http://www.gamedev.net*
- AI Depot at *http://ai-depot.com*
- Generation5 at *http://www.generation5.org*
- The American Association for Artificial Intelligence at *http://www.aaai.org*

Each web site contains information relevant to game AI as well as additional links to other sources of information on AI.

Here are several print resources that we find helpful (note that these resources include both game and nongame AI books):

- *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference* by Judea Pearl (Morgan Kaufmann Publishers, Inc.)
- *Bayesian Artificial Intelligence* by Kevin Korb and Ann Nicholson (Chapman & Hall/CRC)
- *Bayesian Inference and Decision*, Second Edition by Robert Winkler (Probabilistic Publishing)
- *AI Game Programming Wisdom* by Steve Rabin, ed. (Charles River Media)
- *AI Techniques for Game Programming* by Mat Buckland (Premier Press)
- *Practical Neural Network Recipes in C++* by Timothy Masters (Academic Press)
- *Neural Networks for Pattern Recognition* by Christopher Bishop (Oxford University Press)
- *AI Application Programming* by M. Tim Jones (Charles River Media)

Using Code Examples

This book is designed to help you get your job done. In general, you can use the code in this book in your programs and documentation. You do not need to contact us for permission unless you're reproducing a significant portion of the code. For example, writing a program that uses several chunks of code from this book does not require permission. Selling or distributing a CD-ROM of examples from O'Reilly books does require permission. Answering a question by citing this book and quoting example code does not require permission. Incorporating a significant amount of example code from this book into your product's documentation does require permission.

We appreciate, but do not require, attribution. An attribution usually includes the title, author, publisher, and ISBN. For example: "*AI for Game Developers*, by David M. Bourg and Glenn Seemann. Copyright 2004 O'ReillyMedia, Inc., 0-596-00555-5."

If you feel your use of code examples falls outside fair use or the permission given here, feel free to contact us at *permissions@oreilly.com*.

How to Contact Us

Please address comments and questions concerning this book to the publisher:

O'Reilly Media, Inc.
1005 Gravenstein Highway North
Sebastopol, CA 95472
(800) 998-9938 (in the United States or Canada)
(707) 829-0515 (international or local)
(707) 829-0104 (fax)

We have a web page for this book, where we list errata, examples, and any additional information. You can access this page at:

http://www.oreilly.com/catalog/ai

To comment or ask technical questions about this book, send email to:

bookquestions@oreilly.com

For more information about our books, conferences, Resource Centers, and the O'Reilly Network, see our web site at:

http://www.oreilly.com

Acknowledgments

We'd like to thank our editors, Nathan Torkington and Tatiana Diaz, for their skill, insight, and patience. We'd also like to express our appreciation to O'Reilly for giving us the opportunity to write this book. Further, special thanks go to all the production and technical staff at O'Reilly, and to the technical reviewers, chromatic, Graham Evans, and Mike Keith, for their thoughtful and expert comments and suggestions. Finally, we'd like to thank our respective wives, Helena and Michelle, for their support and encouragement and for putting up with our repeatedly saying, "I have to work on the book this weekend." We can't forget to thank our respective children, who always remind us that it's OK to play. On that note, remember that the most important goal of good AI in games is to make games more fun. Play on!

Table of Contents

CHAPTER 1
Introduction to Game AI

In the broadest sense, most games incorporate some form of artificial intelligence (AI). For instance, developers have used AI for years to give seemingly intelligent life to countless game characters, from the ghosts in the classic arcade game *Pac Man* to the bots in the first-person shooter *Unreal*, and many others in between. The huge variety of game genres and game characters necessitates a rather broad interpretation as to what is considered game AI. Indeed, this is true of AI in more traditional scientific applications as well.

Some developers consider tasks such as pathfinding as part of game AI. Steven Woodcock reported in his "2003 Game Developer's Conference AI Roundtable Moderator's Report" that some developers even consider collision detection to be part of game AI.* Clearly, some wide-ranging interpretations of game AI exist.

We're going to stick with a broad interpretation of game AI, which includes everything from simple chasing and evading, to pattern movement, to neural networks and genetic algorithms. Game AI probably best fits within the scope of *weak* AI (see the sidebar "Defining AI"). However, in a sense you can think of game AI in even broader terms.

In games, we aren't always interested in giving nonplayer characters human-level intellect. Perhaps we are writing code to control nonhuman creatures such as dragons, robots, or even rodents. Further, who says we always have to make nonplayer characters smart? Making some nonplayer characters dumb adds to the variety and richness of game content. Although it is true that game AI is often called upon to solve fairly complex problems, we can employ AI in attempts to give nonplayer characters the appearance of having different personalities, or of portraying emotions or various dispositions—for example, scared, agitated, and so on.

*Steven Woodcock maintains an excellent Web site devoted to game AI at *http://www.gameai.com*.