

# Artificial intelligence in video games



In strategy games like [Freeciv](#), the game AI must deal with large amounts of information

In [video games](#), **artificial intelligence** (AI) is used to generate responsive, adaptive or [intelligent](#) behaviors primarily in [non-player characters](#) (NPCs) similar to human-like intelligence. [Artificial intelligence](#) has been an integral part of video games since their inception in the 1950s.<sup>[1]</sup> The role of AI in video games has expanded greatly since its introduction. Modern games often implement existing techniques from the field of artificial intelligence such as [pathfinding](#) and [decision trees](#) to guide the actions of NPCs. Additionally, AI is often used in mechanisms which are not immediately visible to the user, such as [data mining](#) and [procedural-content generation](#).<sup>[2]</sup>

## Overview[[edit](#)]

The term "game AI" is used to refer to a broad set of [algorithms](#) that also include techniques from [control theory](#), [robotics](#), [computer graphics](#) and [computer science](#) in general, and so video game AI may often not constitute "true AI" in that such techniques do not necessarily facilitate computer learning or other standard criteria, only constituting "automated computation" or a predetermined and limited set of responses to a predetermined and limited set of inputs.<sup>[3][4][5]</sup>

Many industry and corporate voices<sup>[[who?](#)]</sup> claim that so-called video game AI has come a long way in the sense that it has revolutionized the way humans interact with all forms of technology, although many<sup>[[who?](#)]</sup> expert researchers are skeptical of such claims, and particularly of the notion that such technologies fit the definition of "intelligence" standardly used in the cognitive sciences.<sup>[3][4][5]</sup> Industry voices<sup>[[who?](#)]</sup> make the argument that AI has become more versatile in the way we use all technological devices for more than their intended purpose because the AI allows the technology to operate in multiple ways, allegedly developing their own personalities and carrying out complex instructions of the user.<sup>[6]</sup>

However, many<sup>[[who?](#)]</sup> in the field of AI have argued that video game AI is not true intelligence, but an advertising buzzword used to describe computer programs that use simple sorting and matching algorithms to create the illusion of intelligent behavior while bestowing software with a misleading aura of scientific or

technological complexity and advancement.<sup>[3][4][5]</sup> Since game AI for NPCs is centered on appearance of intelligence and good gameplay within environment restrictions, its approach is very different from that of traditional AI.

## History<sup>[[edit](#)]</sup>

Game playing was an area of research in AI from its inception. One of the first examples of AI is the computerised game of [Nim](#) made in 1951 and published in 1952. Despite being advanced technology in the year it was made, 20 years before [Pong](#), the game took the form of a relatively small box and was able to regularly win games even against highly skilled players of the game.<sup>[1]</sup> In 1951, using the [Ferranti Mark 1](#) machine of the [University of Manchester](#), [Christopher Strachey](#) wrote a [checkers](#) program and [Dietrich Prinz](#) wrote one for [chess](#).<sup>[7]</sup> These were among the first computer programs ever written. [Arthur Samuel](#)'s checkers program, developed in the middle 50s and early 60s, eventually achieved sufficient skill to challenge a respectable amateur.<sup>[8]</sup> Work on checkers and chess would culminate in the defeat of [Garry Kasparov](#) by IBM's [Deep Blue](#) computer in 1997.<sup>[9]</sup> The first [video games](#) developed in the 1960s and early 1970s, like *[Spacewar!](#)*, *[Pong](#)*, and *[Gotcha](#)* (1973), were games implemented on [discrete logic](#) and strictly based on the competition of two players, without AI.

Games that featured a [single player](#) mode with enemies started appearing in the 1970s. The first notable ones for the [arcade](#) appeared in 1974: the [Taito](#) game *[Speed Race](#)* ([racing video game](#)) and the [Atari](#) games *[Qwak](#)* (duck hunting [light gun shooter](#)) and *[Pursuit](#)* ([fighter aircraft dogfighting simulator](#)). Two text-based computer games from 1972, *[Hunt the Wumpus](#)* and *[Star Trek](#)*, also had enemies. Enemy movement was based on stored patterns. The incorporation of [microprocessors](#) would allow more computation and random elements overlaid into movement patterns.



Light cycle characters compete to be the last one riding, in *[GLtron](#)*.

It was during the [golden age of video arcade games](#) that the idea of AI opponents was largely popularized, due to the success of *[Space Invaders](#)* (1978), which sported an increasing difficulty level, distinct movement patterns, and in-game events dependent on [hash functions](#) based on the player's input. *[Galaxian](#)* (1979) added more complex and varied enemy movements, including maneuvers by individual enemies who break out of formation. *[Pac-Man](#)* (1980) introduced AI patterns to [maze games](#), with the added quirk of different personalities for each enemy. *[Karate Champ](#)* (1984) later introduced AI patterns to [fighting games](#), although the poor AI prompted the release of a second version. *[First Queen](#)* (1988) was a [tactical action RPG](#) which featured characters that can be controlled by the computer's AI in following the leader.<sup>[10][11]</sup> The [role-playing video game](#) *[Dragon Quest IV](#)* (1990) introduced a "Tactics" system,

where the user can adjust the AI routines of [non-player characters](#) during battle, a concept later introduced to the [action role-playing game](#) genre by *Secret of Mana* (1993).

Games like *Madden Football*, *Earl Weaver Baseball* and *Tony La Russa Baseball* all based their AI in an attempt to duplicate on the computer the coaching or managerial style of the selected celebrity. Madden, Weaver and La Russa all did extensive work with these game development teams to maximize the accuracy of the games.<sup>[*[citation needed](#)*]</sup> Later sports titles allowed users to "tune" variables in the AI to produce a player-defined managerial or coaching strategy.

The emergence of new game genres in the 1990s prompted the use of formal AI tools like [finite state machines](#). [Real-time strategy](#) games taxed the AI with many objects, incomplete information, pathfinding problems, real-time decisions and economic planning, among other things.<sup>[12]</sup> The first games of the genre had notorious problems. *Herzog Zwei* (1989), for example, had almost broken pathfinding and very basic three-state state machines for unit control, and *Dune II* (1992) attacked the players' base in a beeline and used numerous cheats.<sup>[13]</sup> Later games in the genre exhibited more sophisticated AI.

Later games have used [bottom-up](#) AI methods, such as the [emergent behaviour](#) and evaluation of player actions in games like *Creatures* or *Black & White*. *Faade (interactive story)* was released in 2005 and used interactive multiple way dialogs and AI as the main aspect of game.

Games have provided an environment for developing artificial intelligence with potential applications beyond gameplay. Examples include *Watson*, a *Jeopardy!*-playing computer; and the *RoboCup* tournament, where robots are trained to compete in soccer.<sup>[14]</sup>

## Views[[edit](#)]

Many experts complain that the "AI" in the term "game AI" overstates its worth, as game AI is not about [intelligence](#), and shares few of the objectives of the academic field of AI. Whereas "real AI" addresses fields of machine learning, decision making based on arbitrary data input, and even the ultimate goal of [strong AI](#) that can reason, "game AI" often consists of a half-dozen rules of thumb, or [heuristics](#), that are just enough to give a good gameplay experience.<sup>[*[citation needed](#)*]</sup> Historically, academic game-AI projects have been relatively separate from commercial products because the academic approaches tended to be simple and non-scalable. Commercial game AI has developed its own set of tools, which have been sufficient to give good performance in many cases.<sup>[2]</sup>

Game developers' increasing awareness of academic AI and a growing interest in computer games by the academic community is causing the definition of what counts as AI in a game to become less [idiosyncratic](#). Nevertheless, significant differences between different application domains of AI mean that game AI can still be viewed as a distinct subfield of AI. In particular, the ability to legitimately solve some AI problems in games by [cheating](#) creates an important distinction. For example, inferring the position of an unseen object from past observations can be a difficult problem when AI is applied to robotics, but in a computer game a NPC can simply look up the position in the game's [scene graph](#). Such cheating can lead to unrealistic

behavior and so is not always desirable. But its possibility serves to distinguish game AI and leads to new problems to solve, such as when and how to use cheating.<sup>[[citation needed](#)]</sup>

The major limitation to strong AI is the inherent depth of thinking and the extreme complexity of the decision making process. This means that although it would be then theoretically possible to make "smart" AI the problem would take considerable processing power.<sup>[[citation needed](#)]</sup>

## Usage<sup>[[edit](#)]</sup>

### In computer simulations of board games<sup>[[edit](#)]</sup>

- [Computer chess](#)
- [Computer Go](#)
- [Computer checkers](#)
- [Computer poker players](#)
- [Computer bridge](#)
- [Computer shogi](#)
- [Computer Arimaa](#)
- [Logistello](#), which plays [Reversi](#)
- [Rog-O-Matic](#), which plays [Rogue](#)
- [Computer players of Scrabble](#)
- A variety of board games in the [Computer Olympiad](#)
- [General game playing](#)
- [Solved games](#) have a computer strategy which is guaranteed to be optimal, and in some cases force a win or draw.

### In modern video games<sup>[[edit](#)]</sup>

Game AI/heuristic algorithms are used in a wide variety of quite disparate fields inside a game. The most obvious is in the control of any [NPCs](#) in the game, although "scripting" ([decision tree](#)) is currently the most common means of control.<sup>[15]</sup> These handwritten decision trees often result in "artificial stupidity" such as repetitive behavior, loss of immersion, or abnormal behavior in situations the developers did not plan for.<sup>[16]</sup>

[Pathfinding](#), another common use for AI, is widely seen in [real-time strategy](#) games. Pathfinding is the method for determining how to get a NPC from one point on a map to another, taking into consideration the terrain, obstacles and possibly "[fog of war](#)".<sup>[17][18]</sup> Commercial videogames often use fast and simple "grid-based pathfinding", wherein the terrain is mapped onto a rigid grid of uniform squares and a pathfinding algorithm such as [A\\*](#) or [IDA\\*](#) is applied to the grid.<sup>[19][20][21]</sup> Instead of just a rigid grid, some games use irregular polygons and assemble a [navigation mesh](#) out of the areas of the map that NPCs can walk to.<sup>[19][22]</sup> As a third method, it is sometimes convenient for developers to manually select "waypoints" that NPCs should use to navigate; the cost is that such waypoints can create unnatural-looking movement. In addition, waypoints tend to perform worse than navigation meshes in complex environments.<sup>[23][24]</sup> Beyond static pathfinding, [navigation](#) is a sub-field of Game AI focusing on giving NPCs the capability to navigate in a dynamic environment, finding a path to a target while avoiding collisions with other entities

(other NPC, players...) or collaborating with them (group navigation).<sup>[*citation needed*]</sup> Navigation in dynamic strategy games with large numbers of units, such as *Age of Empires* (1997) or *Civilization V* (2010), often performs poorly; units often get in the way of other units.<sup>[24]</sup>

Rather than improve the Game AI to properly solve a difficult problem in the virtual environment, it is often more cost-effective to just modify the scenario to be more tractable. If pathfinding gets bogged down over a specific obstacle, a developer may just end up moving or deleting the obstacle.<sup>[25]</sup> In *Half-Life* (1998), the pathfinding algorithm sometimes failed to find a reasonable way for all the NPCs to evade a thrown grenade; rather than allow the NPCs to attempt to bumble out of the way and risk appearing stupid, the developers instead scripted the NPCs to crouch down and cover in place in that situation.<sup>[26]</sup>

The concept of emergent AI has recently been explored in games such as *Creatures*, *Black & White* and *Nintendogs* and toys such as *Tamagotchi*. The "pets" in these games are able to "learn" from actions taken by the player and their behavior is modified accordingly. While these choices are taken from a limited pool, it does often give the desired illusion of an intelligence on the other side of the screen.

## Video game combat AI<sup>[*edit*]</sup>

Many contemporary video games fall under the category of action, first person shooter, or adventure. In most of these types of games there is some level of combat that takes place. The AI's ability to be efficient in combat is important in these genres. A common goal today is to make the AI more human, or at least appear so.

One of the more positive and efficient features found in modern-day video game AI is the ability to hunt. AI originally reacted in a very black and white manner. If the player were in a specific area then the AI would react in either a complete offensive manner or be entirely defensive. In recent years, the idea of "hunting" has been introduced; in this 'hunting' state the AI will look for realistic markers, such as sounds made by the character or footprints they may have left behind.<sup>[27]</sup> These developments ultimately allow for a more complex form of play. With this feature, the player can actually consider how to approach or avoid an enemy. This is a feature that is particularly prevalent in the *stealth* genre.

Another development in recent game AI has been the development of "survival instinct". In-game computers can recognize different objects in an environment and determine whether it is beneficial or detrimental to its survival. Like a user, the AI can look for cover in a firefight before taking actions that would leave it otherwise vulnerable, such as reloading a weapon or throwing a grenade. There can be set markers that tell it when to react in a certain way. For example, if the AI is given a command to check its health throughout a game then further commands can be set so that it reacts a specific way at a certain percentage of health. If the health is below a certain threshold then the AI can be set to run away from the player and avoid it until another function is triggered. Another example could be if the AI notices it is out of bullets, it will find a cover object and hide behind it until it has reloaded. Actions like these make the AI seem more human. However, there is still a need for improvement in this area.

Another side-effect of combat AI occurs when two AI-controlled characters encounter each other; first popularized in the *id Software* game *Doom*, so-called 'monster



infighting' can break out in certain situations. Specifically, AI agents that are programmed to respond to hostile attacks will sometimes attack *each other* if their cohort's attacks land too close to them.<sup>[[citation needed](#)]</sup> In the case of *Doom*, published gameplay manuals even suggest taking advantage of monster infighting in order to survive certain levels and difficulty settings.

## Uses in games beyond NPCs<sup>[[edit](#)]</sup>

[Georgios N. Yannakakis](#) suggests that academic AI developments may play roles in game AI beyond the traditional paradigm of AI controlling NPC behavior.<sup>[[2](#)]</sup> He highlights four other potential application areas:

1. **Player-experience modeling:** Discerning the ability and emotional state of the player, so as to tailor the game appropriately. This can include [dynamic game difficulty balancing](#), which consists in adjusting the difficulty in a video game in real-time based on the player's ability. Game AI may also help deduce player intent (such as [gesture recognition](#)).
2. **Procedural-content generation:** Creating elements of the game environment like environmental conditions, levels, and even music in an automated way. AI methods can generate new content or interactive stories.
3. **Data mining** on user behavior: This allows game designers to explore how people use the game, what parts they play most, and what causes them to stop playing, allowing developers to tune gameplay or improve monetization.
4. **Alternate approaches to NPCs:** These include changing the game set-up to enhance NPC believability and exploring social rather than individual NPC behavior.

Rather than procedural generation, some researchers have used [generative adversarial networks](#) (GANs) to create new content. In 2018 researchers at Cornwall University trained a GAN on a thousand human-created levels for [DOOM](#) (1993); following training, the neural net prototype was able to design new playable levels on its own. Similarly, researchers at the [University of California](#) protyped a GAN to generate levels for [Super Mario](#).<sup>[[28](#)]</sup>

## Cheating AI<sup>[[edit](#)]</sup>

[Dani Bunten](#) was once asked how to play-balance a game. Her one word answer was "Cheat." Asked what to do if gamers complained, she said, "Lie!"

Gamers always ask if the AI cheats (presumably so they can complain if they lose)

— *Terry Lee Coleman of Computer Gaming World*, 1994<sup>[[30](#)]</sup>

In the context of artificial intelligence in video games, cheating refers to the programmer giving agents actions and access to information that would be unavailable to the player in the same situation.<sup>[[31](#)]</sup> Believing that the [Atari 8-bit](#) could not compete against a human player, [Chris Crawford](#) did not fix a bug in [Eastern Front \(1941\)](#) that benefited the computer-controlled Russian side.<sup>[[32](#)]</sup> *Computer Gaming World* in 1994 reported that "It is a well-known fact that many AIs 'cheat' (or, at least, 'fudge') in order to be able to keep up with human players".<sup>[[29](#)]</sup>

For example, if the agents want to know if the player is nearby they can either be given complex, human-like sensors (seeing, hearing, etc.), or they can cheat by simply asking the [game engine](#) for the player's position. Common variations include giving AIs higher speeds in racing games to catch up to the player or spawning them in advantageous positions in first person shooters. The use of cheating in AI shows the limitations of the "intelligence" achievable artificially; generally speaking, in games where strategic creativity is important, humans could easily beat the AI after a minimum of trial and error if it were not for this advantage. Cheating is often implemented for performance reasons where in many cases it may be considered acceptable as long as the effect is not obvious to the player. While cheating refers only to privileges given specifically to the AI—it does not include the inhuman swiftness and precision natural to a computer—a player might call the computer's inherent advantages "cheating" if they result in the agent acting unlike a human player.<sup>[31]</sup> [Sid Meier](#) stated that he omitted multiplayer alliances in *Civilization* because he found that the computer was almost as good as humans in using them, which caused players to think that the computer was cheating.<sup>[33]</sup> Developers say that most are honest but they dislike players erroneously complaining about "cheating" AI. In addition, humans use tactics against computers that they would not against other people.<sup>[32]</sup>

## Examples<sup>[[edit](#)]</sup>

- *[Creatures](#)* (1996)

*Creatures* is an artificial life program where the user "hatches" small furry animals and teaches them how to behave. These "Norns" can talk, feed themselves, and protect themselves against vicious creatures. It's the first popular application of machine learning into an interactive simulation. Neural networks are used by the creatures to learn what to do. The game is regarded as a breakthrough in artificial life research, which aims to model the behavior of creatures interacting with their environment.<sup>[34]</sup>

- *[Sid Meier's Alpha Centauri](#)* (1999)<sup>[[citation needed](#)]</sup>
- *[Halo: Combat Evolved](#)* (2001)

A [first-person shooter](#) where the player assumes the role of the Master Chief, battling various aliens on foot or in vehicles. Enemies use cover very wisely, and employ suppression fire and grenades. The squad situation affects the individuals, so certain enemies flee when their leader dies. A lot of attention is paid to the little details, with enemies notably throwing back grenades or team-members responding to you bothering them. The underlying "behavior tree" technology has become very popular in the games industry (especially since Halo 2).<sup>[34]</sup>

- *[F.E.A.R.](#)* (2005)

A first-person shooter where the player helps contain supernatural phenomenon and armies of cloned soldiers. The AI uses a planner to generate context-sensitive behaviors, the first time in a mainstream game. This technology used as a reference for many studios still today. The enemies are capable of using the environment very cleverly, finding cover behind tables, tipping bookshelves, opening doors, crashing through windows, and so on. Squad tactics are used to great effect. The enemies perform flanking maneuvers, use suppression fire, etc.<sup>[34]</sup>

- [\*S.T.A.L.K.E.R.\* series](#) (2007-)

A first-person shooter survival horror game where the player must face man-made experiments, military soldiers, and mercenaries known as Stalkers. The various encountered enemies (if the difficulty level is set to its highest) use combat tactics and behaviours such as healing wounded allies, giving orders, out-flanking the player or using weapons with pinpoint accuracy.<sup>[*citation needed*]</sup>

- [\*Far Cry 2\*](#) (2008)

A first-person shooter where the player fights off numerous mercenaries and assassinates faction leaders. The AI is behavior based and uses [action selection](#), essential if an AI is to multitask or react to a situation. The AI can react in an unpredictable fashion in many situations. The enemies respond to sounds and visual distractions such as fire or nearby explosions and can be subject to investigate the hazard, the player can utilize these distractions to his own advantage. There are also social interfaces with an AI but however not in the form of direct conversation but more reactionary, if the player gets too close or even nudges an AI, the player is subject to getting shoved off or sworn at and by extent getting aimed at. Other social interfaces between AI exist when in combat, or neutral situations, if an enemy AI is injured on the ground, he will shout out for help, release emotional distress, etc.<sup>[*citation needed*]</sup>

- [\*StarCraft II\*](#) (2010)

A [real-time strategy](#) game where a player takes control of one of three factions in a 1v1, 2v2, or 3v3 battle arena. The player must defeat his opponents by extinguishing all their units and bases. This is accomplished by creating units that are effective at countering your opponents units. Players can play against multiple different levels of AI difficulty ranging from very easy to cheater 3 (insane). The AI is able to cheat at the difficulty, cheater 1 (vision), where it begins to have vision assistance on your location and units. Cheater 2 gives the AI extra resources, while Cheater 3 give an extensive advantage over its opponent.<sup>[35]</sup>

## See also<sup>[[edit](#)]</sup>

- [AI effect](#)
- [Applications of artificial intelligence](#)
- [Behavior model](#)
- [Video game bot](#)
- [Simulated reality](#)
- [Kynapse](#) – game AI middleware, specializing in path finding and spatial reasoning
- [AiLive](#) – A suite of game AI middleware
- [xaitment](#) – graphical game AI software

### Lists

- [List of emerging technologies](#)
- [List of game AI middleware](#)
- [Outline of artificial intelligence](#)



## References[edit]

1. ^ [Jump up to: <sup>a</sup> <sup>b</sup>](#) Grant, Eugene F.; Lardner, Rex (2 August 1952). *"The Talk of the Town – It"*. *The New Yorker*.
2. ^ [Jump up to: <sup>a</sup> <sup>b</sup> <sup>c</sup>](#) Yannakakis, Georgios N (2012). *"Game AI revisited"* (PDF). *Proceedings of the 9th conference on Computing Frontiers*: 285–292. Archived from [the original](#) (PDF) on 8 August 2014.
3. ^ [Jump up to: <sup>a</sup> <sup>b</sup> <sup>c</sup>](#) Bogost, Ian (March 2017). *"'Artificial Intelligence' Has Become Meaningless"*. Retrieved 22 July 2017.
4. ^ [Jump up to: <sup>a</sup> <sup>b</sup> <sup>c</sup>](#) Kaplan, Jerry (March 2017). *"AI's PR Problem"*. MIT Technology Review.
5. ^ [Jump up to: <sup>a</sup> <sup>b</sup> <sup>c</sup>](#) Eaton, Eric; Dietterich, Tom; Gini, Maria (December 2015). *"Who Speaks for AI?"* (PDF). *AI Matters*. **2** (2).
6. ^ Eastwood, Gary. *"How video game AI is changing the world"*. CIO. Retrieved 28 February 2017.
7. ^ See ["A Brief History of Computing"](#) at AlanTuring.net.
8. ^ Schaeffer, Jonathan. *One Jump Ahead:: Challenging Human Supremacy in Checkers*, 1997,2009, Springer, ISBN 978-0-387-76575-4. Chapter 6.
9. ^ [McCorduck, Pamela](#) (2004), *Machines Who Think* (2nd ed.), Natick, MA: A. K. Peters, Ltd., ISBN 1-56881-205-1, p. 480-483
10. ^ [First Queen](#) at MobyGames
11. ^ ["Official Site"](#). *Kure Software Koubou*. Retrieved 19 May 2011. ([Translation](#))
12. ^ Schwab, 2004, p. 97-112
13. ^ Schwab, 2004, p.107
14. ^ [Emergent Intelligence in Games Archived](#) 19 February 2011 at the [Wayback Machine](#).17 February 2011.
15. ^ Good, Owen S. (5 August 2017). *"Skyrim mod makes NPC interactions less scripted, more Sims-like"*. Polygon. Retrieved 16 April 2018.
16. ^ Lara-Cabrera, R., Nogueira-Collazo, M., Cotta, C., & Fernández-Leiva, A. J. (2015). Game artificial intelligence: challenges for the scientific community.
17. ^ Yannakakis, G. N. (2012, May). Game AI revisited. In *Proceedings of the 9th conference on Computing Frontiers* (pp. 285–292). ACM.
18. ^ Hagelback, Johan, and Stefan J. Johansson. "Dealing with fog of war in a real time strategy game environment." In *Computational Intelligence and Games*, 2008. CIG'08. IEEE Symposium On, pp. 55-62. IEEE, 2008.
19. ^ [Jump up to: <sup>a</sup> <sup>b</sup>](#) Abd Algfoor, Zeyad; Sunar, Mohd Shahrizal; Kolivand, Hoshang (2015). "A Comprehensive Study on Pathfinding Techniques for Robotics and Video Games". *International Journal of Computer Games Technology*. **2015**: 1–11. doi:10.1155/2015/736138.
20. ^ Yap, Peter. "Grid-based path-finding." In *Conference of the Canadian Society for Computational Studies of Intelligence*, pp. 44-55. Springer, Berlin, Heidelberg, 2002.
21. ^ Sturtevant, N. R. (June 2012). "Benchmarks for Grid-Based Pathfinding". *IEEE Transactions on Computational Intelligence and AI in Games*. **4** (2): 144–148. doi:10.1109/TCLAI.2012.2197681.
22. ^ Goodwin, S. D., Menon, S., & Price, R. G. (2006). Pathfinding in open terrain. In *Proceedings of International Academic Conference on the Future of Game Design and Technology*.
23. ^ Nareyek, A. (2004). AI in computer games. *Queue*, 1(10).
24. ^ [Jump up to: <sup>a</sup> <sup>b</sup>](#) Cui, X., & Shi, H. (2011). A\*-based pathfinding in modern computer games. *International Journal of Computer Science and Network*

Security, 11(1), 125-130.

25. <sup>^</sup> ["Design Techniques and Ideals for Video Games"](#). *Byte Magazine* (7.12). 1982. p. 100.
26. <sup>^</sup> Lidén, L. (2003). Artificial stupidity: The art of intentional mistakes. *AI game programming wisdom*, 2, 41-48.
27. <sup>^</sup> Schreiner, Tim. "Artificial Intelligence in Game Design." Artificial Intelligence Depot. Web. 19 November 2009. [AI-depot.com](#)
28. <sup>^</sup> ["AI creates new levels for Doom"](#). *BBC News*. 8 May 2018. Retrieved 17 May 2018.
29. <sup>^</sup> [Jump up to: <sup>a</sup> <sup>b</sup>](#) Wilson, Johnny L. (February 1994). ["Mea Culpas And Culpability"](#). Editorial. *Computer Gaming World*. p. 8.
30. <sup>^</sup> Coleman, Terry Lee (July 1994). ["He Ain't Heavy, He's My Sovereign"](#). *Computer Gaming World*. pp. 110–111.
31. <sup>^</sup> [Jump up to: <sup>a</sup> <sup>b</sup>](#) Scott, Bob (2002). "The Illusion of Intelligence". In Rabin, Steve. *AI Game Programming Wisdom*. Charles River Media. pp. 16–20.
32. <sup>^</sup> [Jump up to: <sup>a</sup> <sup>b</sup>](#) Wilson, Johnny L.; Brown, Ken; Lombardi, Chris; Weksler, Mike; Coleman, Terry (July 1994). ["The Designer's Dilemma: The Eighth Computer Game Developers Conference"](#). *Computer Gaming World*. pp. 26–31.
33. <sup>^</sup> ["The 7th International Computer Game Developers Conference"](#). *Computer Gaming World*. July 1993. p. 34. Retrieved 12 July 2014.
34. <sup>^</sup> [Jump up to: <sup>a</sup> <sup>b</sup> <sup>c</sup>](#) [AiGameDev - Top 10 Most Influential AI Games](#)
35. <sup>^</sup> ["StarCraft II"](#). *StarCraft II*. Retrieved 2017-02-28.

## Bibliography[[edit](#)]

## External links[[edit](#)]

- [Artificial Intelligence for Games](#)
- [Special Interest Group on Artificial Intelligence @IGDA](#)
- [AI Game Programming Wisdom](#) on aiwisdom.com
- [Game AI for Developers](#) on AiGameDev.com
- [AI section](#) at [GameDev.net](#)
- [The General Video Game AI Competition](#)