Computer Games — Boon or Bane for AI Research?

Alexander Nareyek

Computer Science Department
Carnegie Mellon University
5000 Forbes Avenue
Pittsburgh, PA 15213-3891
USA

alex@ai-center.com
http://www.ai-center.com/home/alex/

Will applications like computer games finally help break down the walls of the ivory tower? Or is it simply a silly distraction, undermining the seriousness of our research field in the public view? In this article, I present my opinion. As a precaution: I believe that points of view can rarely be advanced by harmonious speeches accompanied by general nodding. Instead, a friendly but spicy discussion can help find and clarify positions in a much better way. Therefore, please do not be offended if you feel your adrenaline levels rising.

At first, let me clarify what I mean by "computer games". This article is not about board/card/puzzle games, like chess or Rubik's Cube. Games like chess are actually good examples of what I would like research to steer away from. While chess has served as a kind of Drosophila for AI research, and has led some great advances in search, nowadays, most chess research is so over-specialized that I can hardly call it research anymore — it is application development.

Does computing a rebel's behavior in the galactic civil war in a Star Wars computer game require sophisticated technology? You bet! Suddenly, one has to deal with optimization in real time, a highly dynamic and complex environment, incomplete knowledge, very restricted computation power and memory, ... and forget about all these nice things like closed-world assumptions. Completeness properties and complexity classes below undecidability? Hah, are you kidding?

Tackling the features above may sound difficult to begin with, and I do not recommend that anyone tries to realize them all at once. However, the important thing is to keep these features in mind when initially striving for a subset. Algorithm design must take place in a holistic view of the complete set of target features, and not for single properties in isolation. I refer to this as "**multi-objective algorithm design**". There are lots of trade-offs to be considered when striving for these features. Trade-offs between speed and memory are well known, but the same applies to other features.

Currently, nearly all AI research focuses exclusively on computational efficiency, neglecting other goal features. Because of the **trade-offs** involved in multi-objective tasks, this leads to the somewhat counter-intuitive effect that – although focusing on something obviously important like computation speed – research steers away from possible applications, further widening the gap between research and application. What is the point in specializing an algorithm for solving a machine-scheduling problem 75% faster if the result is an extremely specialized version that cannot be reasonably adapted anymore to handle dynamic changes, like a machine breakdown or the change of an order? Oh yes, I forgot, you can publish a paper that will surely be cheerfully praised by the academic community because you beat some abstract toy-problem benchmark...

It is much easier to compare approaches and results with respect to only one feature, which may be a reason why everyone seems to bow to the deity of computational efficiency. It is certainly correct that in general, different applications need different feature sets to be supported, which makes a direct comparison somewhat difficult. Looking at different feature sets, however, does not mean that we enter the realm of arbitrary interpretation. Richer metrics to compare and guide solution development must be developed, and respective benchmarks be established that consider richer feature dimensions. Given our human – real-time, dynamic and only partially known – life environment, it is hard to believe that a niche area for systems that neglect most of these features would receive much attention in this extended feature view.

You may now see why I do not like games like chess as a research object. Solving chess in one year from now instead of three may be good for political reasons. The scientific contribution, however, is hardly anything notable because the respective methods for this boring one-feature-focused problem are already on such a specialized level that an application for anything else is practically ruled out.

"Real" computer games, on the other hand, make it necessary to keep a great number of features in focus. With their mind-boggling complexity of large-scale simulations with realistic physics and vast interaction possibilities, they can easily be seen as an equivalent to the real world with simplified sensing and actuator procedures. In contrast to the real world, experiments can easily be controlled and repeated.

There are of course also several research groups working in the direction of considering a larger number of relevant features. Initiatives like RoboCup (which I do not consider to be a computer game; but nevertheless, it has some common features) have spurred this development. The same is true for the robotics area, and more generally, in areas where people have to contribute to a real application and things have to actually work.

Unfortunately, this application focus often turns out to be only "yet another case study", and results are rarely analyzed from a more general point of view. An application like computer games is no different in this respect, and the right balance between application-specific "hacking" and general technology/approaches is not easily achieved. It is important to keep this in mind -

especially in an exciting application area like computer games, where lots of tremendously impressive technology blends together with your own. From looking over paper submissions and presentations in this area, I have to say that many supervisors seem to let their students loose on this great looking "stuff", and provide little guidance on relevant research questions.

Even though I do not want this article to be considered as another facet of the "theory vs. application" struggle for research orientation, I consider it vitally important for research groups to work on real applications on occasion. Simply to get a feeling about what the important features and issues are. "Beautiful" concepts, algorithms and proofs are great. But if they do not address any relevant needs, it is for the arts or entertainment department, not for computer science. In contrast to basic research in areas like, say, physics, computer science makes it very easy to set up virtual problems with no relation to the real world at all. Do not get me wrong here — I do not say that we need more focus on application-oriented research. I only wish that researchers have a clear idea of what the important features really are. It sounds obvious and not much to learn, but in my experience, I have only seen people with some practical experience develop a feel for this.

I think that using a computer-game application for research can provide a great opportunity to re-focus research efforts toward more relevance, forcing everyone to think about and involve many important features of our world. It is of course not the only option, but it is fun, your students will be extremely motivated, and it makes for some great presentations.

Some words of warning: All the talk about the billions of dollars that the computer games industry is making should not lead you to believe that there is plentiful research funding to be collected from that industry. The way the computer games industry is currently structured does not provide much leeway to invest in something other than the current game under development. Thus, if you want to get into the area as a new funding source, I strongly suggest that you look into other areas.

Another issue that might turn out to be a problem is getting a suitable game platform and interface for research. While there are a few games that let you write AI scripts in game-specific languages, there is hardly anything available that provides more basic interfaces on a level of ordinary programming languages. Similar to other application areas, it is unlikely that you will get the source code from the companies unless the technology is sufficiently outdated. Public open-source projects on the internet may be an alternative but are also nowhere close to current commercial technology. An additional problem regarding a good game platform is that the game content often involves a horrific level of violence. There are alternatives out there if you care about this, but it further complicates your search for a suitable platform.

Building up some contacts with the industry certainly helps in sorting out these problems. And it is really worth it! It is an application area at the pinnacle of technological progress, substantially driving the whole computer industry by its impact, and provides an awfully inspiring environment.

We are currently in progress of setting up a lab, which will be one of the very few spots in the world that combine AI research with the application area of computer games. I am hoping that many research groups will join us in making use of this highly challenging application area. People who consider the area simply as a silly distraction are most often unfamiliar with it or fear the loss of the domination of their beloved one-dimensional evaluation criterion of computational efficiency.

Hardly anything that AI research produced so far made its way into computer games - with reason. How relevant is your research?

Alexander Nareyek received his diploma and Ph.D. from the TU Berlin. Since 2002, he is on an Emmy Noether fellowship of the German Research Foundation (DFG), and is guest researcher at Carnegie Mellon University. His main research interests include the generation and execution of behavior plans for goal-driven intelligent agents. He is also active in the application area of computer games and serves as chairperson of the IGDA's Artificial Intelligence Interface Standards Committee (AIISC).