Educational systems need appropriate animations and simulations

Ruddy Lelouche

Educational systems need appropriate animations and simulations

Ruddy LELOUCHE

Département d'Informatique, UNIVERSITÉ LAVAL, Québec G1K 7P4, CANADA

Phone: (+1-418) 656 2131 (2597) Fax: (+1-418) 656 2324 E-mail: LELOUCHE@IFT.ULAVAL.CA

Abstract

Animations and simulations are often presented as tools for learning, although not all animations are appropriate for that purpose. The goal of this paper is to get things straight in that respect. After giving a short definition of graphic animations and simulations, this paper contends that, in educational systems, these should be used only if they are related to the subject to be learned, to the learning process, and/or to the learner's characteristics, thus discrediting animations used solely for marketing purposes. Then a simple but educationally relevant simulation is discussed in a simulation-based system to help practising and learning algorithmics. Finally, the paper proposes a guideline establishing general relationships between learning task categories, knowledge types, and animation or simulation types.

Keywords: *multimedia system, graphical animation, graphical simulation, interaction, pedagogical purpose (of the animation), learning goal (of the student), virtual reality, special education, learning activity, knowledge type.*

Résumé

Les animations et les simulations sont souvent présentées comme des outils d'apprentissage, bien que toutes les animations ne soient pas aptes à remplir ce rôle. Le but de cet article est de remettre les pendules à l'heure à ce sujet. Après avoir brièvement (re)défini les animations et simulations graphiques, cet article soutient que, dans les systèmes éducatifs, celles-ci ne devraient être utilisées que si elles ont quelque chose à voir avec la matière à apprendre, avec le processus d'apprentissage, et/ou avec les caractéristiques de l'apprenant, laissant ainsi de côté les animations utilisées uniquement à des fins de marketing. Ensuite, nous présentons et discutons une simulation simple mais pédagogiquement pertinente incluse dans un système d'aide à l'apprentissage et à la pratique de l'algorithmique.Enfin, l'article propose une ligne directrice établissant des relations générales entre les types d'activités d'apprentissage, les types de connaissances mises en œuvre, et les types d'animations et de simulations.

Mots clés: *système multimédia, animation graphique, simulation graphique, interaction, but pédagogique (de l'animation), but d'apprentissage (de l'étudiant), réalité virtuelle, éducation spécialisée, activité d'apprentissage, type de connaissances.*

Introduction

Educational systems have tremendously evolved during the recent years [Lelouche, 1998]. In particular the emphasis of multimedia and hypermedia techniques has made such systems more attractive. These techniques have then been made popular through the wide use of Internet and in particular of the Worldwide Web. A notorious application of such techniques is the use of *graphical animations and simulations*, which, however, have been in effect and widely used for a long time before and independently of the Internet. Since transmission hiccups and delays over a wide-area network may affect the smoothness of such effects, very likely, their effectiveness is still bigger outside the Internet. For example, realistic animations are present particularly in computer-based video games which, even if distributed on the Internet (e.g. through e-commerce), are most often executed from a CD-ROM, or even from the local hard disk, rather than directly from the Internet. Besides games, another important use of animations is educational systems, where they have a wider and wider impact.

In this paper, we try to examine the roles and the effects of graphic animation and simulations on educational systems. The first section recalls what animations and simulations are. Section 2, aimed at "setting the stage", presents several general aspects of animation uses in educational systems, giving examples as appropriate. Section 3 gives a more detailed example of animations in a simple system built to acquire programming skills. Finally, section 4 establishes some guidelines relating various types of animations to various types of pedagogical purposes.

1 . What are animations and simulations?

Although often associated to film making (for example, the *Animation Journal* [Furniss, 1991] is exclusively related to that area), *animation* is a product, a branch, of computer graphics [Foley & al., 1990; Laybourne, 1998]. However, all that is animation is not always graphic (indeed some use only ASCII text!), but this paper will concentrate on graphic animations. All that is graphic is not always animation either. Animations are precisely those graphic displays where something is *changing over time*. Depending on the case at hand, that something may be appearing or disappearing, moving, or evolving (e.g. changing shape and/or colour, etc.). The graphic can be a line drawing, a cartoon, or a picture; when animated, these lead respectively to line animations, animation cartoons, and movies (e.g. videos).

Regarding simulations, we first exclude the *simulation discipline* [Yurcik, 1999], that is math-based rather than graphic-based (although simulation packages often include animated simulations for marketing reasons), to concentrate on *graphic simulations*. The latter are of two types. *Static simulations* are still displays (real pictures or computer-generated), and therefore are not aminations (nothing changes over time); their interest is the quality of their effect rendering (ex.: surface textures, landscapes). On the other hand, *dynamic simulations* are animations. What makes both categories simulations is their content and their interactive nature. Contentwise, they demonstrate some property, real or not, of the displayed scene: a body, an event, a phenomenon, etc. As to their nature, they allow the user to interact with the displayed scene, by varying its speed or pausing it if animated, by focusing on a particular area (zooming, etc.), or even by influencing the animation outcome itself (this is the case with certain simulation-based games, in particular role-playing games). Both categories of simulations are used in educational systems; the displayed scene and the demonstrated properties are then likely to relate to the subject matter. However, in this article, we are not interested in static simulations, which are simply akin to good pictures.

From a technical standpoint, animations and simulations may be either prepared in advance, or generated on demand. The former approach requires important storage capacity and bandwidth (e.g. "introduction" parts of games), while the latter needs a fast processor and a lot of memory (*Incredible machines* game). The animation may also combine both approaches: partly pre-computed, and processor-completed at run time (e.g. *Quake* game).

2. Animations in educational systems

Animations are certainly very popular, were it only because of their attractiveness, in particular in spectacular games and movies. But what exactly do they bring to educational systems? Are they improving them, and if so on what grounds, or are they essentially glitter patched on them to make them more glamorous, more attractive, and thus easier to sell? In my opinion, to be educationally relevant, they should certainly not be like the highlighted word "FREE" — or some other more tendentious buzzword — written on the top of certain billboard advertisements. The proof: you have certainly seen and maybe entirely read such an ad, but have you ever called the person who posted it?...

This section successively examines the relationships of animations with subject matter, with the learning process, with the student's interests, and, at last, with marketing.

2.1 Animations and subject matter

In an educational setting, the first purpose of any object presented to the student is that it should be *descriptive of the contents to be learned* (e.g. a course). That certainly holds in particular for animated scenes. Many examples of such animations can be found in physics (e.g. in kinematics or dynamics), in engineering (e.g. working of a four-stroke engine, creation or absorption of a photon [Henderson & al., 1979]), in astronomy (e.g. planets and stars movements), in molecular biology (e.g. human genome [Human Genome Project, 2001]), in certain areas of computer science (e.g. sorting algorithms [Brummund, 1997; Harrison, 2001; McCauley, 2001]). That also holds for static animations, quite useful in such areas as descriptive natural sciences (e.g. botany or zoology) or physical sciences (e.g. mechanical constraints in static physics or in materials science).

Such animations or simulations are certainly very appropriate, if not the most appropriate, because they definitely help the learner understand the subject, process or law that is being presented.

However, to meet most effectively that goal, such an animation or simulation should *concentrate on what is to be learned*, and nothing else; if too realistic, it may deter the learner from the very subject at hand. For example, to present the working of a four-stroke engine to a mechanics student, a line drawing or a cartoon may be sufficient. Indeed, a movie showing a real engine inside a real car hood may accidentally focus the student's attention on an inappropriate place, e.g. a red label put somewhere by some service person, especially if that label happens to be placed on a moving part. A similar and common situation occurs when driving an automobile: as a driver, my attention may be attracted by a set of blinking lights at night. If these lights signal some dangerous spot, fine! they appropriately warn me of that danger. But if they turn out to merely signal a restaurant or an entertainment place, I may feel frustrated to have had to concentrate uselessly on something unrelated to my driving. For an experienced driver, the blinking lights may simply mean frustration; for a student or tired driver, they may become dangerous by themselves...

2.2 Animations and learning process

Even when it bears a relationship with the subject matter, a good educational animation should actually *augment or at least facilitate the student's learning*. It certainly does if it makes the contents easier to *understand*, like in the examples above, which depict the very phenomenon or law that the student is supposed to understand. Other animations could simply *illustrate the subject matter*, without being as crucial to the learning process. By making the content more palatable, they probably facilitate its ingestion!

Although the understanding of a concept, of how a phenomenon works, is important, the learning process also involves two other aspects, namely *motivation* for learning, and *memorisation* of what is being learned. Thus animation software that addresses them also contribute to easing the learning process, although not as effectively as if they also facilitate understanding.

Finally, *experimentation*, that may be involved in all three aspects — understanding, motivation, and memorisation — is essentially present in interactive simulations.

Thus, when assessing the contribution of a proposed or existing animation or simulation to the learning process, one should examine how it addresses understanding, motivation, memorisation, and experimentation. However, one should also take into account *the time spent by the student for learning the simulation environment* vs. what it allows the student to learn, i.e. its functional capabilities.

2.3 Animations and student interests

The learning process, described in general terms above, involves at least two entities: the subject matter and the learner. So, besides the course contents examined in 2.1, it may happen that an animation tries to relate to the learning person: his/her psychology, experiences, interests, etc. That is especially true when the "knowledge" to be learned involves the learner himself or herself, who must then learn how-to-be (vs. learn what or learn how-to-do). Such situations are: driving an automobile or an aeroplane, human resources management, sociopolitical behaviours, or the integration of special-need persons into their social environment. An example of the last situation is the Brazilian AVIRC Project [Moreira da Costa & Vidal de Carvalho, 2000], an Integrated Virtual reality Environment for Cognitive Rehabilitation where the learner is a person with acquired brain injury or neuro-psychiatric disorder. Another example [Moreno & al., 2000] deals with a multimedia interface where the modalities of the performed cognitive task are adapted to the characteristics of the learner, a child with intellectual and cognitive handicaps. A most recent example is the automatic generation of learning simulation scenarios in a safe virtual environment [Ip & al., 2001], presently used for political students, but useable in other domains.

In these types of learning situations, *very realistic animations or simulations* are appropriate, because the learner is eventually bound to deal with reality. Indeed, when learning how-to-be, important details to account for may not stand out as conspicuously as when studying a law in physics, for instance. Such realistic software could then be appropriate substitutes for real life scenarios. Going farther, *virtual reality environments* may sometimes be the best and even the only way to learn, not only for cost savings, but also because other persons are at stake. Such are the cases of flight simulators, especially when developed for planes not yet constructed, of virtual surgical operations for medicine students, or of environments for special-need learners.

In my opinion however, animations geared at special individuals' interests are more difficult to assess objectively than the previous ones. Indeed a person (here the learner) is much more complex than a subject matter or even than the learning process, which have been more extensively studied. Assessment questions to be asked would be in the following directions. On what precise grounds the relations between the animation-simulation software and the learner have been established? Is a particular type of learner targeted? If so, how has s/he been modelled? And how is the animation or simulation connected to the learner's model?

2.4 Animations and marketing

Actually, a thin borderline may separate (possibly attractive, but) genuinely useful pedagogical animations and simulations from the ones whose main virtue is their (marketing) attractiveness. Indeed, many software praise their "breath-taking" animations, and the modern rendering techniques are quite fabulous [Watt & Watt, 1992]. A learner may indeed experience tremendous fun with an especially attractive animation, but if s/he does not learn better or more effectively because of it, then that animation should nevertheless be questioned.

Thus, when assessing an animation software, one should uncomplacently evaluate what are the primary *intents* and *effects* of the animation. Often it seems to be there merely to provoke a bewildered "Wahoo!". I certainly do not criticise "glitter animations" as such, but if you have a wonderfully glamorous animation to propose or to assess, maybe you should wonder whether and how it is connected *also* to the subject matter, to the general learning process, and/or to the targeted learner.

2.5 Partial conclusion

In an educational context, an animation should be related primarily to the subject matter, to the learning process, or to the learner (figure 1). Rendering effects such as realism, speed or reactivity are only secondary, if they do not strengthen these primary relationships. David Merrill [2000] stated: "Ratings based primarily on production value and appearance do little to inform you about the instructional quality of a product". This holds in particular for animations and simulations.

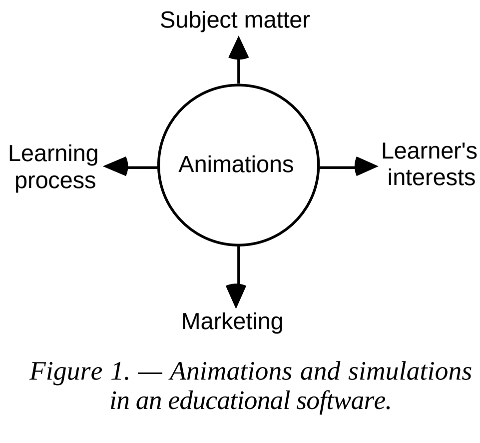

Figure 1. — Animations and simulations in an educational software.

Naturally, the present discussion, and in particular figure 1, deals with animations assuming that the student is alone with the educational system, i.e. independently from the environment in which (s)he is placed. If, for example, a human teacher is present and can refocus the learner's attention on the relevant points (e.g. the learning goals), the drawback or inadequacy of an otherwise inappropriate animation may be lessened or even eliminated.

3. An example: animations in the ALGORITHMIK simulation software

The ALGORITHMIK system [Dion, 1988; Lelouche, 1999] has been designed for college students in an introductory programming class. The main tutor objective is to help the student learn to use adequately basic control structures (IF THEN, IF THEN ELSE, WHILE DO, ITERATE *n* TIMES) and procedure calls. To that effect, ALGORITHMIK uses a micro-world called "Karel the robot" [Pattis, 1981], in which the student must program a robot to make it achieve various tasks requiring to use the learned control structures. As a major advantage, this micro-world requires no data structure, thus allowing to separate two difficulty levels —data structures and control structures — felt by the programming beginners.

ALGORITHMIK provides the student with three interactive components: a structure editor for code entry, a graphic situation editor, and a graphic simulator for executing and tracing the program (figure 2). To help the student solve his/her problem, it also includes an intelligent tutor, with which we do not deal here.

The available problems are categorized according to their assumed difficulty: decision structures, iterative structures, or all control structures. When starting a session, the student may choose a category, then his/her first problem in that category; afterwards, s/he is guided into working on more difficult problems. Alternatively, s/he may relinquish the use of the tutor and devise his/her own problem to work on. In that case however, since the system "knows" nothing about the task to be performed, the student is left alone (no tutoring).

Let us assume that the student decides to work on the first of the "all control structures" category, the super-steeplechase problem [Lelouche, 1999]. S/he is then presented with the following task description (here translated into English):

> Karel must run a super-steeplechase. The hurdles have a variable height and a zero width. The end of the run is marked by a beeper. Karel begins its run facing east; at the end of the run, it must also face east and is supposed to have picked up the beeper ending the run.

and with a display like that laid out on figure 2, showing a program window in which s/he can work on his/her program using the structure editor, and an example of initial situation.

If the student is not happy with the proposed situation, s/he may modify it using the graphical *situation editor*, which allows him/her to place the robot at the desired intersection and with the desired orientation, and to place or move bippers and wall sections around as s/he wishes. Although they do not involve animation or simulation, the structure editor and the situation editor are

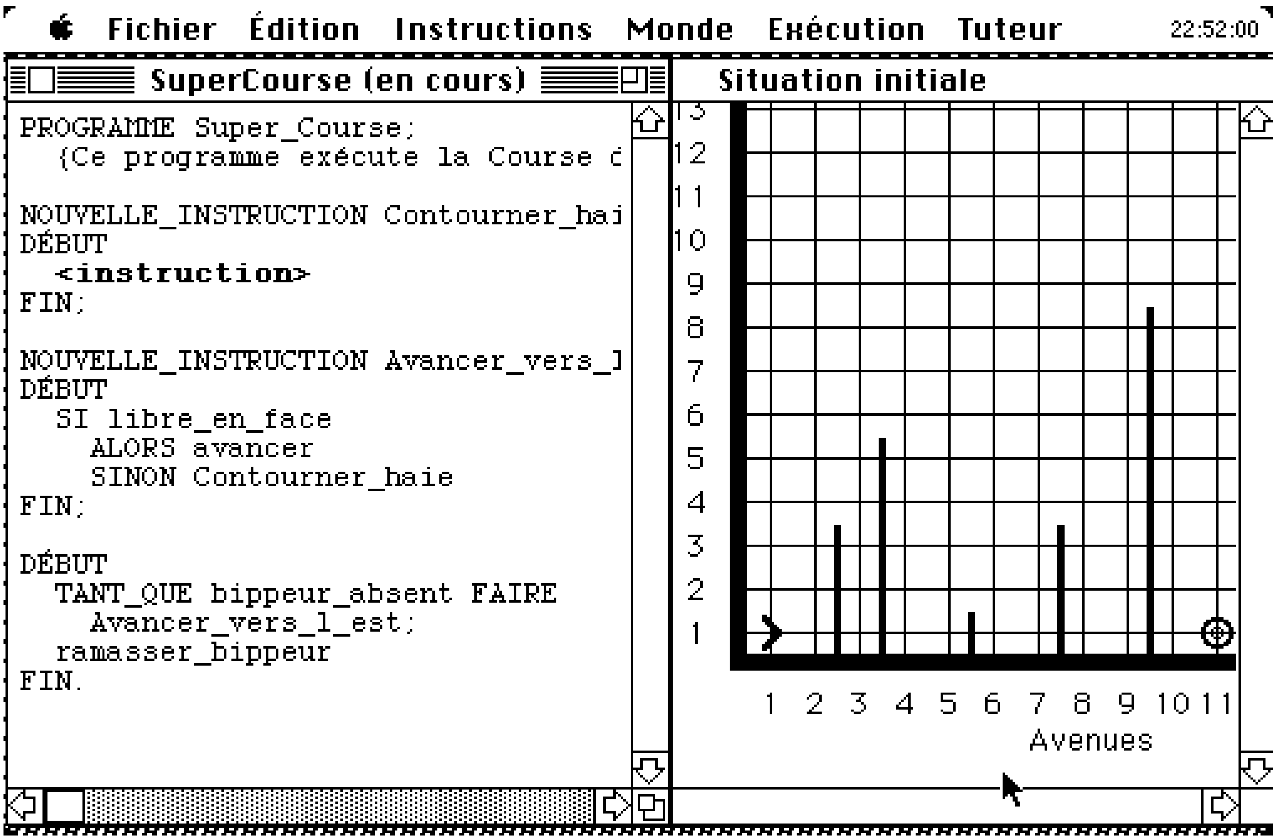

Figure 2. — ALGORITHMIK environment interface.

convenient interactive graphic tools to use, especially to vary the program situation parameters through experimentation.

The *execution simulator*, which can be activated using the *Execution* menu (figure 3), does involve animation. It is used to visualise, on the active situation, the execution of the active program window; in addition, the instruction currently executing is simultaneously highlighted in the program text. Thus, at all times, the relationship between the active program instruction and the result of its execution is visible. Besides, at any time during execution, the student can change the execution mode (*Fast speed*, *Slow speed*, or *Stepwise*), or can activate the *Pause* and *Abort execution* commands (see figure 3).

Execution
Execute •E
√ Fast speed Slow speed Stepwise
Pause Abort execution

Figure 3. — ALGORITHMIK Execution *menu.*

All these facilities help the student follow the program execution, both on the situation field and in the program code. In addition, by slowing down or even stopping the execution at any program point that s/he may find unclear, s/he can gather some information, possibly modify the local situation (Karel's location and orientation, bippers, and wall sections), in order to understand better how the program really works, and thus eventually correct his/her errors. The execution simulator capabilities are particularly useful for problems invented by the student or the teacher, where the help of the tutor is not available.

The execution simulator thus appears to be an interesting and simple alternative to more complex tools, using a more realistic robot for instance. Indeed, the ALGORITHMIK software is simple to learn and master, because only what is important is displayed and included in the controls available to the learner. These simple controls allow him/her to concentrate on the algorithm design and development (rather than play around with more sophisticated tools or displays, for instance).

4. Which animations and simulations for which purposes?

From the discussion and examples given in the previous sections, several general rules, or guidelines, can be drawn. In the table of figure 4, which is our own, we try to emphasise the roles played by animations and simulations in various learning situations.

On the horizontal axis, to a learning task, identified by the type of knowledge to be acquired, we associate the animation type(s) most typically appropriate for that learning task. We document that association with the animation capabilities educationally useful for that task, and with typical domain examples. Note that animations can only work at the task level, which is much lower than the more general learning goal of the student.

On the vertical axis, our table shows a progression from lower-level learning tasks to higher-level ones, and a parallel progression from simpler to more sophisticated animations or simulations. Naturally, the second progression is justified by the fact that the tasks to be performed, and the domains from which they are drawn, are more and more complex, and thus need to be displayed with more and more realism.

Naturally, animations or even simulations are never sufficient per se: they can only be *part of the learning* and *part of the teaching*. On the learning side, they are necessarily related with specific tasks to be performed, tasks that can be far away from more general domain-related learning goals: animation-based activities need to be complemented by more learning-oriented activities, e.g. problem solving. On the teaching side, even with animation-based activities, animation capabilities need to be complemented by other tutoring capabilities of the system or of the teacher, like providing details or explanations on the subject or phenomenon or behaviour

Learning goal (task)	*Knowledge type*	*Animation type*	*Educationally useful capabilities*	*Domain examples*
Acquaintance with the subject matter	Know what (it is)	Static simulation or representation	Emphasis on distinctive features	Classification domains (Botany, Zoology, ...)
Understanding a phenomenon or law	Know how (it works)	Line or cartoon animation	Id. + speed control, zooming effects	Theoretical domains (Geometry, Physics, Mechanics, ...)
Performing a procedure	Know-how (to do)	Realistic animation, Line or cartoon simulation	Id. + showing procedure being performed, parameter modification effects, playback, ...	Practical skill domains (Driving, Building, Troubleshooting, ...)
Dealing with a situation involving others	Know-how-to-be (attitudes and behaviours)	Realistic simulation Virtual reality	Id. + immersion into the situation to be mastered	Human domains (Human resources managt, Politics, Medical domains, Special-need learners, ...)

Figure 4. — Relationships between educational goals and animation types.

at hand, answering various kinds of questions from the learner (which become more and more difficult to answer appropriately when the targeted knowledge type evolves from know-what to know-how-to-be), etc. And above all, links have to be made between the tasks empowered by the animation or simulation and the higher-level capabilities that the student must eventually acquire.

It should also be noted that animation properties like motivation-triggering or attractiveness are absent from our table. That is because such properties can possibly apply to all animation types, and also because they cannot be measured objectively (at best through sensitivity questionnaires). Besides and most importantly, it is our strong belief that motivation or attraction alone cannot be a source for learning; they can only be modalities that can ease a learning process the source of which lies necessarily elsewhere. Although realism does augment attractiveness, we have shown in section 2.1 that too realistic an animation can deter the learner from difficult learning problems, or even mask these problems (e.g. by accidentally focusing the student's attention on some irrelevant aspect of the animation) because it does not *concentrate on what is to be learned.* Similarly, in spite of their present popularity as a research trend, we think that, although animated agents do have their advantages, they also have their limitations, especially if they distract the student from concentrating on what is to be learned.

Conclusion

This paper tried to show that animations or simulations are not a feature that necessarily augment the pedagogical value of a computer-based educational software or environment. They can do so only if some learning goal underlies them, and if their types are appropriate for helping the student meet that goal.

We hope that the presented guidelines can help people designing animations and simulations build more educationally significant ones, and can help people in charge of buying or assessing an educational software not be unduly impressed by glamorous animations the learning goal of which is absent or questionable. It would certainly be valuable to design and build a real instrument, possibly computer-based, to assess the educational merits of a given animation, taking into account the learning contexts, materials, and objectives, as well as the learner's specificities, but that endeavour is beyong the scope of this position paper.

References

Brummund P. (1997) *The Animator*. Hope College (Holland, Michigan, USA). On line at <http://www.cs.hope.edu/~alganim/ccaa/brummund.html>.

Dion P. (1988) *Conception et implantation d'un système de tutorat pour l'enseignement de l'algorithmique.* Master's thesis. U. Laval (Québec, Canada).

Foley J., A. van Dam, S. Feiner & J. Hugues (1990) *Computer Graphics, 2nd ed.* Addison-Wesley (Reading, MA).

Furniss M. R., ed. (1991) *Animation Journal*. Published since 1991, now at AJ Press (Irvine, CA, U.S.A.). Online at <http://www.animationjournal.com/>.

Harrison J., ed. (2001) *Sorting Algorithms*. University of British Colombia (Vancouver, BC, Canada). On line at <http://www.cs.ubc.ca/spider/harrison/Java/sorting-demo.html>.

Henderson G., R. C. Rittenhouse, J. C. Wright & J. L. Holmes (1979) "How a photon is created or absorbed". *Journal of Chemical Education*, Vol. 56, p. 631-635. An electronic version with interactive graphics and animations is on line at <http://jchemed.chem.wisc.edu/JCEWWW/Articles/DynaPub/DynaPub.html>.

Human Genome Project (2001) *The Human Genome Project: Exploring our Molecular Selves*. Free, multimedia educational kit available on line at <http://www.nhgri.nih.gov/educationkit/>

Ip. A., R. Linser & S. Naidu (2001) "Simulated worlds: rapid generation of Web-based role-play". Proceedings of *AusWeb 01*, the Seventh *Australian World-Wide Web Conference*, 21-25 April 2001, Coffs harbour (NSW, Australia). On line at <http://ausweb.scu.edu.au/aw01/papers/refereed/ip/paper.html>

Laybourne K. (1998) *The Animation Book: a Complete Guide to Animated Film-Making - From Flip-Books to Sound Cartoons to 3-D Animation.* Three Rivers Pr. (N.-Y., U.S.A.).

Lelouche R. (1999) "Can a student-controlled environment and the model-tracing methodology go together?". *Advanced Research in Computers and Communications in Education, Vol. 1* (G. Cummings, T. Okamoto, L. Gomez, eds.). IOS Press (Amsterdam, Netherlands), p.938-945.

Lelouche R. (1998) "How education can benefit from computers: a critical survey". Invited conference presented at *CALISCE 98*, the 4th International Conference on *Computer-Aided Learning and Instruction in SCience and Engineering*, Goteborg (Sweden), 15-17 June 1998, p. 19-32.

McCauley R., ed. (2001) "Algorithms and Algorithm Animation". *Computer Science Education Links*. Department of Computer Science, College of Charleston (Charleston, SC, U.S.A.). Available on line at <http://www.cs.cofc.edu/~mccauley/edlinks/#algorithms>

Merrill D. (2000) "Does your instruction rate 5 stars?". Keynote address, Proceedings of *IWALT 2000*, the *International Workshop on Advanced Learning Technology*, Palmerston North (New Zealand), 4-6 December 2000. IEEE Computer Society (Los Alamitos, CA, U.S.A.), p. 8-11.

Moreira da Costa R. M. & L. A. Vidal de Carvalho (2000) “Virtual reality in cognitive retraining”. Proceedings of *IWALT 2000*, the *International Workshop on Advanced Learning Technology*, Palmerston North (New Zealand), 4-6 December 2000. IEEE Computer Society (Los Alamitos, CA, U.S.A.), p. 221-224.

Moreno L., M. C. Gonzalez, R. M. Aguilar, J. Estévez, J. Sánchez & C. Barroso (2000) “Adaptive multimedia interface for users with intellectual and cognitive handicaps”. *Intelligent Tutoring Systems*, Proceedings of *ITS 2000* (Montreal, Canada, June 2000). LNCS 1839. Springer (Berlin & Heidelberg, Germany), p. 363-372.

Pattis R. (1981) *Karel the Robot. a Gentle Introduction to the Art of Programming*. J. Wiley (New York, U.S.A.).

Watt A. & M. Watt (1992) *Advanced Animation and Rendering Techniques: Theory and Practice*. Addison-Wesley (Readings, MA, U.S.A.).

Yurcik W. (1999) *The Simulation Education Home Page*. Applied Computer Science Department, Illinois State University (Normal, IL, U.S.A.). On line at <http://www.acs.ilstu.edu/faculty/wjyurci/nsfteachsim/indexnew.html>.