# The Quantum Coreworld: Competition and Cooperation in an Artificial Ecology

Alexander (Sasha) Wait[1]

[1] Harvard Medical School, Department of Genetics, 77 Avenue Louis Pasteur, Boston, MA 02115, USA.
await@genetics.med.harvard.edu

## Abstract

Evolving systems, in principle, can exploit any tool in their genetic repertoire. This work specifies a computational chemistry consistent with the rules of Quantum Mechanics and a distributed artificial ecology that permits intervention by interested participants. A preliminary demonstration of two programs that use Quantum operations to authenticate one another is described. Competitors cannot impersonate the cooperating Quantum users without themselves using Quantum mechanics. Some limitations of this example are discussed. Readers can also visit the Quantum Coreworld ecology on the Internet at http://science.fiction.org.

## Introduction

Does the underlying physics of a living system change its properties in a qualitative way? Could a different physics permit entirely new types of life? The aim of the Quantum Coreworld project is to engineer, or discover, a toy life-form with a different underlying physics. The success of such an organism—at exploiting available resources before competitors or at cooperating with genetically identical friends—must depend on its use of Quantum operations. If this endeavor required delicate control of large Quantum systems, there would be no way to get started with current technology. As it happens, however, interesting Quantum operations can be simulated on ordinary digital computers.

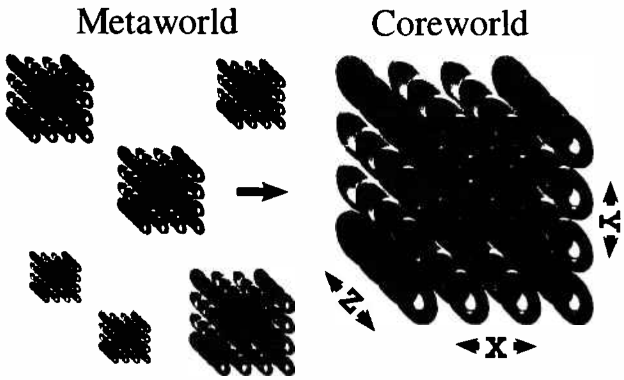


**Figure 1. A Quantum Coreworld in the Metaworld.**

The Quantum Coreworld was built by integrating "libquantum" (Butscher and Weimer 2003) with "pMars" (Ma et al. 1995), the *de facto* standard for the game Corewar (Dewdney 1984). The whole ecology—referred to as the Metaworld in Figure 1—combines elements from Avida (Adami and Brown 1994, Adami 1998, Adami and Lenski 2002), Tierra and Network Tierra (Ray 1992, 2000, 2004) and ongoing Internet Corewar tournaments (Dewdney 1987, Pihlaja 2004). Its name is inspired by the Coreworld, the first artificial life simulator based on Corewar (Rasmussen et al. 1990). This article provides an overview of the new world; more details can be found at: http://genetics.med.harvard.edu/~await/alife9.

Quantum strangeness is particularly obvious when unexpected correlations show up in spatially separated Quantum systems. To give organisms a better chance of exhibiting this strangeness, the world consists of isolated films of locally interacting compartments or cores. Films interact by periodic exchanges, random "bubbles", along vertical columns of cores. In Figure 1, the cores are donuts arranged into films along the Z axis. An energy source—energy is called privilege in the Quantum Coreworld—warms the world from behind, into the page, so that some cores are better suited to life than others. Cores along the Z axis furthest from the reader will have more privilege added by the energy source. Along the X and Y axes the greatest likelihood that privilege will be added is in the center, an intermediate likelihood at the center edges and the lowest likelihood at the corners. As time passes cores are cleared, or "washed away", so that life remains mostly in the middle of the world. This process is nearly the inverse of adding new privilege, so the cores at the corners are most likely to be cleared, center edges have an intermediate likelihood and cores at the center the lowest likelihood.

Each Coreworld in the Metaworld consists of a fixed number of molecules. These molecules are the basic constituent parts of cores and organisms, and operate according to the rules of the computational chemistry. Important global parameters—such as the dimensions of the world in units of cores—are referred to as the climate of the Coreworld. Shifts in the climate and opportunities for outside intervention are measured in epochs. The world's evolution is completely determined given a description of interventions (if any), the climate, and every molecule. At the beginning of every epoch, a Quantum Coreworld simulator starts with this description and produces a new one that describes the world at the end of the epoch. The simulator also reports the temporal and spatial distribution of genotypes in the world.

Copyrighted Material

Downloaded from http://direct.mit.edu/books/book/chapter-pdf/268257/9780262291392_cbt.pdf by guest on 29 September 2023

## The Chemistry

Corewar, Avida, and Tierra are part of the well known family of artificial chemistries called assembler automata (Dittrich 2000). The Corewar language, Redcode, was last standardized in 1994; this standard has sustained ongoing tournaments typically run using the pMars simulator. Tutorials for that standard are readily available (Karonen 2004). Molecules in the Quantum Coreworld are described with an extended Redcode language that supports the notion of privilege and Quantum operations. This language also supports a more biological—2004 style—biotic execution model. A biotic molecule is active in the 2004 style model if it has at least one unit of privilege. Traditional 1994 standard programs can still run without change as a type of active abiotic environment.

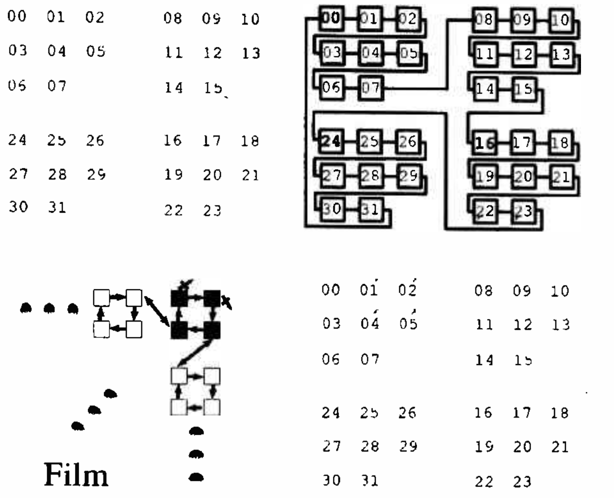


**Figure 2. Molecules in three Cores**

A typical arrangement of molecules in a core is shown in Figure 2. The number of molecules per core—referred to as core-size—is equal to the total number of molecules in the world divided by the total number of cores. There are three cores in the figure, core-size 32, corresponding to the upper right corner of the Coreworld in Figure 1. The world does not wrap around, so there is nothing beyond the edges of the world. From the core in the upper right corner of Figure 2 the accessible molecules in its neighbors are illustrated by lines connecting molecules across cores. The inset in Figure 2 represents this relationship in another way by dividing cores into four quadrants. In any case, molecules can interact with any other molecule in the same core, and with exactly one quarter of the molecules in each neighboring core. The numbering of molecules indicates their absolute position in the core. The peculiar arrangement allows the entire world to be displayed on a standard 1600x1200 display with one pixel per molecule.

Every molecule in the Coreworld has one: instruction field, modifier field, A mode, A field, B mode, B field, privilege field, qubit field, and tag. The instruction field determines the type of thing that each molecule can do.

The available instructions are listed in Table 1. The modifier specifies what field(s) an instruction uses. Modifiers are listed in Table 2. The A mode determines how the A-field is to be interpreted and the B mode determines how the B-field is to be interpreted. Valid modes are listed in Table 3. Both the A-field and B-field are integers between 0 and core-size - 1. Privilege is a positive integer. Its value can be unlimited in principle, but it cannot be greater than the total privilege in the world, which is itself bounded. The tag is a cryptographic hash of the intervention bringing that molecule into the world. Molecules with biotic tags execute under the 2004 style model—requiring privilege greater than zero to be active—while molecules with abiotic tags execute under the 1994 style model and use more traditional Corewar task-queues. Abiotic molecules, by definition, are not part of organisms, and are removed from the world at the end of each epoch.

| | |
|---|---|
| **DAT** | terminate process (1994) do not advance privilege (2004) |
| **MOV** | move from A to B (1994) use privilege if necessary (2004) |
| **ADD** | add A to B, result in B |
| **SUB** | subtract A from B, result in B |
| **MUL** | multiply A by B, result in B |
| **DIV** | divide B by A, result in B if A is not zero, else DAT |
| **MOD** | divide B by A, remainder in B if A is not zero, else DAT |
| **JMP** | execute at A |
| **JMZ** | execute at A if B is zero |
| **JMN** | execute at A if B is not zero |
| **DJN** | decrement B, if B is not zero, execute at A |
| **IJN** | increment B, if B is not zero, execute at A (2004 only) |
| **SLT** | skip if A is less than B |
| **SEQ**<br>**CMP** | skip if A is equal to B |
| **SNE** | skip if A is not equal to B |
| **NOP** | no operation |
| **SPL** | new abiotic task (1994 only) |
| **QOP** | Quantum operation on B, specified by A (2004 only) |
| **QCN** | Quantum not of target B controlled by A (2004 only) |

**Table 1. Valid instructions.** Note: Instruction behavior can depend on execution style: biotic (2004) or abiotic (1994).

Copyrighted Material

Downloaded from http://direct.mit.edu/books/book/chapter-pdf/268257/9780262291392_cbt.pdf by guest on 29 September 2023

| | |
|---|---|
| **.A** | Instr. read and write A-fields |
| **.B** | Instr. read and write B-fields |
| **.AB** | Instr. read A-field of A-instr. and B-field of B-instr. and write B-field |
| **.BA** | Instr. read B-field of A-instr. and A-field of B-instr. and write A-field |
| **.F** | Instr. read both A&B fields of A&B instr. and write to both A&B fields (Ato A and B to B). |
| **.X** | Instr. read both A&B fields of A&B instr. and write to both A&B fields (Ato B and B to A). |
| **.I** | Instr. read and write Instr., Modifier, Modes, A & B fields |
| **.P** | Instr. read and write privilege |
| **.Q** | Instr. read and write qubit |
| **.D*** | B pointer refers to dual core |
| **.E*** | Use privilege, execute faster |

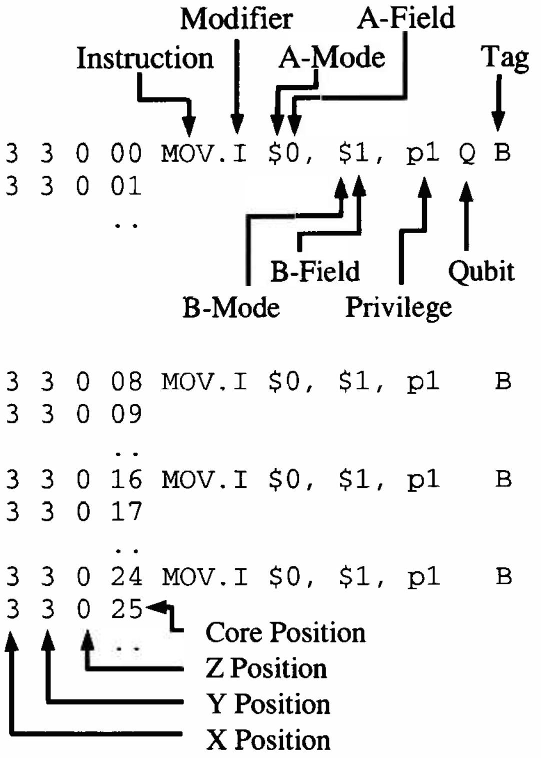
**Table 2. Valid modifiers.** Note (*) indicates modifier can be used in combination with other modifiers.

| | |
|---|---|
| **#** | immediate |
| **$** | direct |
| **@** | indirect using B-field |
| **<** | predecrement indirect using B-field |
| **>** | postincrement indirect using B-field |
| ***** | indirect using A-field |
| **{** | predecrement indirect using A-field |
| **}** | postincrement indirect using A-field |

**Table 3. Valid addressing modes.**

Several departures from 1994 Redcode can be seen in the above Tables. IJN, Increment Jump if Not zero, is analogous to DJN, Decrement Jump if Not zero. It has been discussed, but not added to Redcode in the past. In combination with the new .Q modifier—which specifies that the instruction deals primarily with the qubit field—IJN has a non-quantum application as a "concurrency primitive" (atomic test and set). It also plays a role in adding privilege, essentially spawning a new task, when used in combination with the new modifier .P, which specifies that the instruction deals primarily with the privilege field. The QOP and QCN instructions are one qubit and two qubit operations, respectively, and discussed in the example of a Quantum organism. The .D modifier specifies that the B-pointer refers to the dual core (recall Figure 2), and the .E modifier uses privilege in exchange for a greater likelihood of execution; the .E modifier is the only permanent way privilege is removed from the world.

A consideration in developing this chemistry was the hope that community participation would improve the chances of discovering a toy Quantum life-form. The seemingly arbitrary decision to support both 1994 and 2004 styles of Redcode is one concrete step in this direction; this allows hundreds of existing Redcode programs to be tested in the Quantum Coreworld as interventions that specify an abiotic environment. In the spirit of other scientific free software projects, such as TeXmacs (van der Hoeven 2003), I believe this openness is a natural way to do science.



```
3 3 0 00 MOV.I $0, $1, p1 Q B
3 3 0 01
      ..

3 3 0 08 MOV.I $0, $1, p1   B
3 3 0 09
      ..
3 3 0 16 MOV.I $0, $1, p1   B
3 3 0 17
      ..
3 3 0 24 MOV.I $0, $1, p1   B
3 3 0 25
      ..
```

**Figure 3. Four organisms in the Coreworld**

## Organisms and Genotypes

With the chemistry out of the way, it is possible to define organisms and genotypes in the Coreworld. In general, an individual organism is a contiguous sequence of biotic molecules; the tag of the molecule must specify it came from a "biotic" intervention. A simple one-molecule organism is shown in Figure 3. This organism is called an "Imp", and it circles around the core until it is cleared. If at least one of the molecules in an individual organism has privilege greater than zero, the organism is alive. An organism's genotype, in the Coreworld, has one genetic element for every molecule in the individual. A cryptographic hash of the encoding of these genetic elements is the unique genotype; see Deutsch (2002) for

Copyrighted Material

Downloaded from http://direct.mit.edu/books/book/chapter-pdf/268257/9780262291392_cbt.pdf by guest on 29 September 2023

the implementation. The genetic portion of a molecule is the instruction, modifier, A mode, B mode and tag. These fields are all copied by a MOV.I, along with the A-field and B-field. MOV.I requires one unit of privilege; the free privilege is adjusted, as necessary, so that privilege is conserved. In the case of the Imp, it is reduced to privilege zero, and then after one unit of privilege is moved to the next instruction, it has privilege negative one. If this occurs, the instruction with negative privilege is cleared, which is why the Imp in the biotic execution model does not leave a trail of Imps in core; the molecules are consumed to provide the necessary privilege . From time to time, the Imps in Figure 3 can catch up to each other, briefly creating a larger two-molecule organism. The Coreworld gives this larger organism its own genotype, and tracks the number of cycles that these genotypes—and all genotypes—occur core-by-core.

## Epochs

At the beginning of a new epoch, the simulator chooses one climate parameter—X, Y, Z or M—and perturbs it by adding or subtracting one. If the parameter is already at the limit of its range, the direction of change is forced. These parameters are defined so that the X dimension is equal to $2^X$. Y and Z are defined in a similar way. M is defined so that the multiplier that determines the allocation of new privilege is equal to $2^M$-1. By convention, X,Y, and Z vary from 2 to 4 and M from 1 to 5. The total number of molecules in each world is fixed to be $2^{20}$, while the X,Y, and Z dimensions of the world can vary between values of $2^2$, $2^3$, or $2^4$ cores. Thus, core-size varies between the powers of two from $2^8$ to $2^{14}$. Molecules are adjusted as necessary to accommodate a new core-size; this adjustment benefits from restricting core-sizes to powers of two. Since the X,Y, and Z dimensions are part of a Coreworld's climate, and remain constant throughout a given epoch, core-size remains constant throughout an epoch.

Interventions, as already mentioned, can occur at the beginning of an epoch. Whether or not an intervention occurs, pairs of films in the Coreworld are collapsed by picking individuals from each film, one at a time, and randomly placing them in a single film—possibly in nearby cores, but in the same location. Privilege lost in this process and throughout the past epoch—except for privilege consumed by the .E modifier—is referred to as free privilege. At the beginning of each epoch, the free privilege from the previous epoch is distributed in the world; this is done randomly or by repeatedly stamping out the molecules of intervention(s).

The stochastic behavior of the Quantum Coreworld is determined by the state of its pseudo-random-number generator (perturbed by interventions). A cryptographic quality generator is used—see Jenkins (1996) for the implementation—the state of the generator must be saved at the end of each epoch.

## A Quantum Redcode example

The programs used to illustrate Quantum operations in this article are reminiscent of a bacterial replicator and a temperate bacteriaphage that imparts a beneficial capability to its host. These programs use a Quantum Hadamard operation as an integral part of their survival strategy. Quantum bits and Quantum operations are defined according to the standard conventions in Quantum computation (Nielsen and Chang 2000). When the host encounters a molecule from an unknown phage—or another replicator—it eventually executes the following sequence of Redcode instructions (from "rep-qphage.red"):

```
                 ...     found something ...


modulebverify    QOP     #1,                  @attack
                 DJN.Q   >attack,
oldploc          JMP     moduleverify,             #0

                 ...     not friend ...

moduleverify     DJN     modulebverify,   #confidence

                 ...     friend ...
```

These instructions, or molecules, were specified earlier in Table 1. As mentioned, molecules consist of a segment of "code", two segments of "data" and—added in the Quantum Coreworld—a Quantum bit or qubit. Privilege is not used in this example. Since the execution style is closer to standard 1994 Redcode, two co-operating programs can gain a benefit—effectively faster execution—over a single program. An analogous, Hadamard-based advantage is possible in the new style execution model, but not completed at the time of writing.

A qubit, $|\psi>$, is defined $|\psi> = \alpha\,|0> + \beta\,|1>$ where $\alpha$ and $\beta$ are complex numbers. The states $|0>$ and $|1>$ are known as computational basis states. The probability that a qubit is in state $|0>$, when measured, is $|\alpha|^2$ and the probability that it is in state $|1>$, again when measured, is $|\beta|^2$. In the Quantum Coreworld—and physicists believe in the real world—there is no way to access $\alpha$ and $\beta$ directly. In the code above, the DJN.Q operation performs a measurement. DJN then clears the qubit to $|0>$ (setting $\alpha$=1 and $\beta$=0). We need only one more Quantum operation to understand the code. QOP #1 [address] specifies that the qubit at address should be modified by Quantum operation #1. In the Quantum Coreworld, QOP #1 is a Hadamard. The operation turns a $|0>$ into $(|0>+|1>)/\sqrt{2}$ and a $|1>$ into $(|0>-|1>)/\sqrt{2}$; conceptually, these are halfway between $|0>$ and $|1>$ (measuring either state gives a $|0>$ or $|1>$ with equal probability.) That is it!

If a pattern of #confidence (ten in the example) qubits are initialized to a one, and then disguised with a Hadamard, the pattern of ones can only be identified by

Copyrighted Material

Downloaded from http://direct.mit.edu/books/book/chapter-pdf/268257/9780262291392_cbt.pdf by guest on 29 September 2023

another Quantum program. Measurements of these qubits will return a 0 or 1 with equal probability if a Hadamard is not applied first. Other Quantum operations are available, and it is suspected that some sequences of such operations are too difficult to simulate—due to time and space constraints—on any digital computer. For example, such operations can permit the factoring of large integers that no known technique can tackle. (Shor 1994)

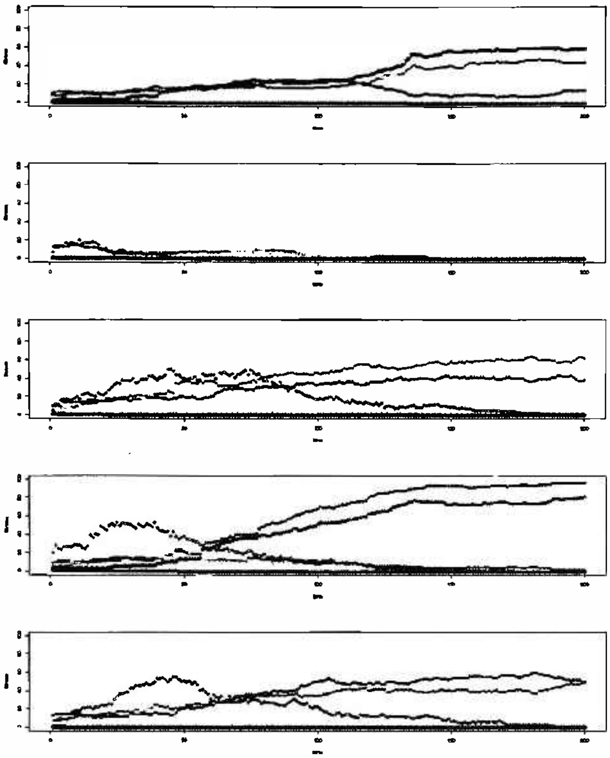


**Figure 4— Five replicators and three viruses in a single 10x10 film in the Quantum Coreworld.** Each panel is a separate run. The horizontal axis is time, and the vertical access is fitness, which is defined, for purposes of this example only, as the number of compartments in which a program is active.

## Testing the example

In Figure 4, replicators and viruses were placed at random on a ten-by-ten film in the Quantum Coreworld. The results of five runs are shown. Two of the programs are identical, except for their ability to authenticate a potentially helpful—but possibly harmful—virus. The non-quantum virus is present in the panel second from the bottom. The Quantum virus is active in the top panel. In this experimental set-up, all viruses are mixed with all programs. If we added a second or third film, we could put the viruses on one film and either the classic or Quantum host on the other. If the authentication works, then the Quantum organism will have no detrimental effect when it is in the presence of one of the other viruses, but the non-quantum organism will be harmed. Experimental setup is summarized in Table 4, and complete results are available in the supplementary information.

| ***Film 1 (all virus)*** | ***Film 2*** | ***Film 3*** |
|---|---|---|
| 10x10film (all) | no film | no film |
| 5x5film | no film | no film |
| no-quantum | quantum | no film |
| quantum | no-quantum | no film |
| no-quantum | quantum | "quick" (yellow) |
| quantum | no-quantum | "quick" (yellow) |
| no-quantum | quantum | "unlock" (cyan) |
| quantum | no-quantum | "unlock" (cyan) |
| no-quantum | quantum | "retry" (purple) |
| quantum | no-quantum | "retry" (purple) |

**Table 4. Set-up for experiment to test Quantum advantage.**

A simple classifier was used to evaluate each run. If a program is active in 23/25 compartments on each film then it "wins"; if it occupies no compartments on any film it "loses" and otherwise it draws. The results are **47 wins, 22 draws, and 31 losses** for the Qunatum organism; and **31 wins, 21 draws, and 48 losses** for the classical organism. Repetition produced comparable results. **Quantum wins!**

## Future Work

A small but significant benefit can be obtained in the Quantum Coreworld by using the Quantum Hadamard operation for authenticating a friend. If an "unfriendly" Quantum virus existed, this would neutralize much of the benefit of Quantum authentication. These programs, in any case, are not very biological, since they predate the use of privilege as described here. This does not preclude the existence of more truly Quantum life-forms in the Coreworld. One possible way to increase the chances of finding such life-forms could be to study communication in the Coreworld, since the spatial structure and more biological execution model of this article should give organisms more to communicate about. Perhaps organisms could benefit from using Quantum Cryptography (Bennett and Brassard 1984) to obtain unconditional security for communication between cores.

As currently devised, the Quantum Coreworld can evolve slowly by mutation and Natural selection, as well as through participant-engineered interventions. If—or when —an interesting Quantum organism is discovered, a next step is to evolve it *de novo.* After seeding the Metaworld with a minimal replicating ecology, we can restrict interventions to organisms written by genetic algorithms. This direction of research attempts to answer the question: "From what starting point could Quantum Mechanical life

Copyrighted Material

Downloaded from http://direct.mit.edu/books/book/chapter-pdf/268257/9780262291392_cbt.pdf by guest on 29 September 2023

evolve?" as opposed to the question being currently asked: "Could Quantum Mechanical life differ from life?"

On one hand, the most intriguing—and the most far-fetched—outcome of this research is the possibility that toy Quantum life-forms in the Quantum Coreworld could help us to recognize such behavior in the real world. On the other hand, the most practical benefit from studying Quantum Artificial Life might be a greater understanding of biological possibilities with more standard physics. A variety of experiments exploring phenomena in evolutionary theory are under consideration; examining multi-level selection, frequency-dependent selection, and mechanisms of diversity maintenance are possibilities.

The operations permitted in a classical world and a Quantum world—especially the operations we can model on a digital computer—are only subtly different. The Quantum Coreworld is a specific model in which these differences can be examined; this exploration can lead to a better understanding of the world as it is.

## Acknowledgments

This project could never have been started in earnest without the support of my graduate advisers in Computer Science, Gilles Brassard and Claude Crépeau. It could not have been completed without the support of my current graduate adviser in Biophysics, George Church.

## References

Adami. C. 1998. *Introduction to Artificial Life*. Santa Clara: TELOS Springer-Verlag.

Adami, C. and Brown, T. 1994. Evolutionary Learning in the 2D Artificial Life System 'Avida'. In Brooks, R. and Maes, P. editors, *Artificial Life IV*, 377-381. Cambridge, MA: MIT Press.

Adami, C. and Lenski, M. 2002. Avida. Software on the web at: http://sourceforge.net/projects/avida

Bennett C.H. and Brassard G. 1984. An update on Qunatum cryptography. In *Advances in Cryptology – proceedings of Crypto'84*, 475-480. Springer Verlag.

Butscher, B. and Weimer, H. 2003. Simulation eines quantencomputers, unpublished (in German.) Software on the web at: http://www.enyo.de/libquantum/

Deutsch, L.P. 2002. RFC1321-based (RSA-Free) MD5 library. Software on the web at: http://sourceforge.net/project/showfiles.php?group_id=42360

Dewdney, A.K. 1984. In the game called core war hostile programs engage in a battle of bits. *Scientific American*, 250:14–22.

Dewdney, A.K. 1987. A program called MICE nibbles its way to victory at the first Core War tournament, Computer Recreations, *Scientific American*, 14-20.

Dittrich, P.; Ziegler, J.; and Banzhaf, W. 2000. Artificial Chemistries - a Review. *Artificial Life*, 7:225-275.

van der Hoeven, J. 2003. Why Freedom is Important for Scientists, part of GNU TeXmacs. On the web at: http://www.texmacs.org/tmweb/about/philosophy.en.html

Jenkins, R.J. 1996. Isaac. In Fast Software Encryption, third international workshop, 41-49. Lecture Notes in Computer Science, 1039, Springer. Software on the web at: http://burtleburtle.net/bob/rand/isaacafa.html

Karonen, I. 2004. Beginner's guide to Redcode. On the web: http://vyznev.net/corewar/guide.html

Ma, A.; Sieben, N.; Strack, S.; and Wangsaw, M. 1995. PMars. Software on the web at: http://www.koth.org/pmars

Nielsen, M.A. and Chuang, I.L. 2000. *Quantum Computation and Quantum Information*. Cambridge University Press.

Pihlaja, M.J. 2004. New KOTH server. Usenet post in REC.GAMES.COREWAR on the web at: http://groups.google.com/groups?th=1f82856630ed460d

Ray, T.S. 1992. An approach to the synthesis of life. In C. G. Langton, C. Taylor, J.D. Farmer, and S. Rasmussen, editors, *Artificial Life II*, 371–408, Redwood City, CA, Addison-Wesley.

Ray, T. S. 2000. Evolution of Complexity: Tissue Differentiation in Network Tierra. ATR Journal, 40:12-13

Ray, T.S. 2004. Tierra. Software on the web at: http://www.isd.atr.co.jp/~ray/tierra/source/index.html

Rasmussen, S.; Knudsen, C.; Feldberg, R; and Hindsholm, M. 1990. The Coreworld: Emergence and evolution of cooperative structures in a computational chemistry. *Physica D*, 42:111–134.

Shor, P. 1994. Algorithms for Quantum computation: Discrete logarithms and factoring. In *35th Annual Symposium on Foundations of Computer Science*, 124-134, IEEE.

Copyrighted Material

Downloaded from http://direct.mit.edu/books/book/chapter-pdf/268257/9780262291392_cbt.pdf by guest on 29 September 2023