Computer Viruses: the Inevitability of Evolution? *

p-m agapow
Machine Intelligence Laboratory
Computer Science
La Trobe University
Bundoora Vic. 3083, Australia
Email: `agapow@latcs1.oz.au`

Abstract.

In recent years computer viruses have received much attention although their theoretical aspects have been neglected. One such aspect is the possibility of viruses as life, with the attendant characteristics of terrestrial life forms including evolution. Where this particular idea has been previously considered, it has been judged highly improbable. The author disagrees and argues that viral evolution may not only be possible but inevitable. After an introduction to the biology of computer viruses evidence of viral evolution will be presented as well as examples of how current and future trends in computing could lead to the emergence of sophisticated and novel entities. Finally the author speculates on the consequences of this.

1. Introduction

"If during the last thousand seconds you have received any High-Beyond protocol packets from 'Arbitration Arts' discard them at once. If they have been processed, then the processing site and all locally netted sites must be physically destroyed at once. We realize that this means the destruction of solar systems, but consider the alternative."

Vernor Vinge, *A Fire Upon the Deep*

Less than a decade ago, there was no term to refer to what we now call "computer viruses". Since that time we have seen an escalation that has matched, and in some ways, surpassed the intrusion of computers into our lives.[1] The extent to which viruses concern computer science in the present day can be measured by the economic impact of major viral outbreaks, some of which have been gauged at costing millions of dollars. Defence against viruses forms a major part of any concern regarding sensitive or critical data. Furthermore the design and detection of viruses has, apparently, attracted the efforts some of the more talented minds in information technology.

However, it is a more speculative but no less important prospect that occupies this paper — the idea of viruses as evolving lifeforms. The author proposes that viral biology

---

[1] One recent work on viruses traces the developments in the field on a month by month basis [?].

and present trends may be leading towards the spontaneous emergence of wholly new and highly sophisticated forms of virus that may confront us with our first examples of alien life. Before considering this, mindful of the background of potential readers, we shall first examine basic virus function.[2]

2. Virus biology

2.1. What is a computer virus?

Basically a computer virus could be defined as any segment of a larger piece of computer code that replicates itself. There are a number of entities that fit this description or are commonly associated with viruses. The best known is the *worm*, an autonomous and mobile replicating process. Others include *Trojan horses* (that masquerade as legitimate programs whilst carrying out unintended operations), *logic bombs* (which lie within legitimate programs and when triggered carry out malicious functions), *chain letters* (self-propagating electronic mail) and *rabbits* (programs which endlessly replicate until system failure). Viruses differ from these in several significant ways. First, replication is purposeful and not simply automatic. Second, the actions of the virus including its replication are triggered and modified by its environment. Finally the virus is parasitic upon another program — it in effect coerces the execution of another program towards its own purposes. Although many of the comments regarding viruses are applicable to these other entities, particularly worms, the author shall restrict his discussion to viruses.

The earliest examples of autonomously replicating entities in computing were seen in the 1960's and typically were instances of runaway programs that as a prank or by semantic error would indiscriminately and repeatedly copy themselves until system failure. Closer to the mark were efforts in fiction in the 1970's illustrated by David Gerrold's *When Harlie Was One* and John Brunner's *The Shockwave Rider* that echoed contemporary experimentation in networking.[3] While there may have been earlier or parallel development the first true virus appears to have been an experiment on an Apple II microcomputer in 1980. Virtually overnight, the occurrence of viruses snowballed until the coining of the phrase "computer virus" in 1984. The exponential increase has continued until the present day where the number of virus species easily exceeds 1000, present on virtually every platform and operating system, with new strains discovered on a weekly basis.

2.2. The "ecology" of computer viruses

Functionally, a virus masquerades as part of the code of another application. When that program is executed, so as a consequence is the virus. Typically, when the viral code is activated the virus will check for a particular triggering condition. If this condition, which could potentially be anything including a specific time or the presence of certain files, is fulfilled the virus will attempt to replicate itself. This replication is usually just the simple appending of the virus code onto another program's code, although in other cases the virus will replace or overwrite some of the host code entirely.

---

[2]For the sake of brevity throughout this document the word "virus" will refer to computer virus and not terrestrial, organic viruses unless otherwise stated.

[3]Brunner's work in particular proved remarkably prescient, in that he depicted a replicating program worm being chased by other "counterworms" across a global computer network. The global network soon became a reality, whilst "antibody" viruses have been discussed as an anti-viral measure.

The above abilities detail a fairly pedestrian "vanilla" virus. A far wider range of abilities can, and have, been seen. Some viruses (notably "Brain", allegedly the first IBM PC virus) utilize "stealth" techniques and corrupt or falsify attempts to detect them. Other "binary" viruses carry the code for themselves in several distinct locations, thwarting detection and deletion. "Polymorphic" viruses encrypt their instructions, using a different encryption key during every infection cycle. Finally viruses may deliberately mutate although this mutation, by and large, does not result in functional change and serves simply as a form of disguise.

It should be stressed that the malignant effects of viruses are, as often as not, the unintentional side-effect of the virus activities. Whilst deliberately malicious or irritating viruses are ubiquitous (for example the system-corrupting Macintosh "Scores" virus, written by a disgruntled programmer to damage his ex-employers) such overt symptoms tend to work against the replicative success of the virus. At least as many viruses are nominally benign, with any harmful effects usually being the result of memory consumption or errors caused by viral replication.

The analogy between computer and terrestrial viruses is a pervasive one and actually serves as a good model of some viral behaviour. Virus outbreaks often show epidemic-like kinetics and can be analysed in a similar way. "Contaminated" systems can be "quarantined", and systems "immunized" against disease "vectors" like virus-infected storage media. How far can this analogy be taken? Spafford [?] raised the prospect of viruses as lifeforms in their own right and then dismissed this, saying:

> ... this probably represents a flaw in our definition [of life].

The author does not intend to debate this, or the philosophical quagmire surrounding the definition of life. Instead, it is proposed that regardless of whether viruses are alive or not, they behave for all intents and purposes as if they are.

2.3. Computer viruses as artificial life-forms

Farmer and Belin [?] proposed a series of properties displayed by life, and it would be instructive to use these and contrast computer viruses with their terrestrial counterparts. This contrast is particularly keen in that terrestrial viruses occupy an ambivalent position in the classification of organisms themselves and are frequently accused of not being "real" living creatures.

The qualities listed by Farmer and Belin were:

- Life is a persistent pattern in space-time, rather than being bound to specific material objects.

- All living objects have some method of reproducing themselves.

- Organisms have some internal self-representation.

- Life displays metabolism: some conversion of the environment towards the activities of the organism.

- Living beings adapt to changes in their external and internal environments.

- Life is an aspect of the whole of an organism — portions of the whole cannot be said to be organisms in their own right.

- Life is stable to noise and minor perturbations in the environment.

• The lineage of an organism displays evolution, systematic change.

It is clear that computer and terrestrial viruses can be considered to exist as patterns in computer code and in molecules respectively. Computer code may be copied, shifted and deleted, not being tied to a specific location in memory. Terrestrial viruses direct the assembly of proteins and DNA into viral "patterns". Both types hijack the metabolism of their host and direct it towards the purpose of viral replication. Both have an internal self-representation, in an electronic binary pattern and DNA respectively. Computer viruses metabolise memory and CPU time to produce changes in that memory space, whilst terrestrial viruses utilise energy and substrates from their host in synthetic activities. Both respond to changes in their environment.[4] Both kinds display living characteristics as a property of a whole — segmentation almost always leads to total dysfunction. Terrestrial viruses have shown remarkable resistance to environmental perturbation. Computer viruses in contrast are somewhat fragile, although examples can be cited of viral adaptation and more recently examples have been seen with error-correcting code [?].

It should be stressed that the above does not form a Turing test for life, merely a list of characteristics that we find in living things commonly. However, if a possible living entity is seen to fulfill most of these conditions we must ask if it satisfies the others. Indeed some parties have concluded that evolution is an "inevitable property" of all living systems [?]. Terrestrial viruses have clearly been subject to the forces of evolution. Is the same true of computer viruses? The answer is "yes".

3. Viral evolution

3.1. Evolution — natural and artificial

Before we assert that viruses evolve, we must ask what evolution is. A succinct and elegant definition is:

> ... [the] process by which organisms come to differ from generation to generation ... change in the gene pool of a population. [?]

Thus, virus evolution demonstrably exists on a number of mundane levels. If one was to trace the advent of new viral species it becomes clear that "old" viruses, to which defences may have formed, are often modified and re-released.[5] Indeed the simple cloning of old designs is rife within the virus production industry. Several incidents of publishing whole or partial dissembled virus code have been followed by waves of variants. In this way, viruses might be said to use humans as agents of their own evolution.

In a similar vein, viruses exercise natural selection amongst themselves. "Unfit" individuals are easily detected by anti-viral software or advertise their presence and are thus intercepted and selected against. "Fit" individuals exert their influence in a subtle manner and attempt to escape detection. Each generation of anti-viral techniques introduces new selection pressures, shaping the composition of the surviving viral population. Further, viruses that are fortunate enough to infect, deliberately or otherwise, a piece of desirable or widely distributed software enjoy considerable evolutionary success

---

[4]The triggering condition seen in computer viruses bears an uncanny resemblance to the behaviour of temperate phages in the biological arena.

[5]Curiously, more often than not this modification is done by someone other than the original creator.

compared to their kin. (Tales of infected games or programs distributed with magazines are legion.) Again we see humans as an unwitting aid to evolution.

Evidence such as the above is often regarded as inconsequential. By similar lines of reasoning, household whitegoods and cars might also be said to evolve, unreliable or old examples being discarded and replaced by new, "fit" individuals.[6] Usually the objection is that the "quasi-evolution" in the above examples is the result of external and deliberate redesign. True evolution, it is felt, occurs by virtue of chance without intent. In the case of computer viruses, this type of evolution also occurs.

The changes that occur in this type of evolution may be grouped broadly into two classes: environmental and somatic. In the first, changes in the virus behaviour are brought about as the result of its computing environment. This may be due to a number of different factors. Some viral species that are expressed in a wide number of strains are thought to have diverged due to the effect different compilers had on identical source code. In this case the actual code (or genes) of the virus is changed. Alternatively, due to the intricate way in which viruses are constructed, subtly different computing environments may bring about vastly different behaviours. The environmental differences need only be as slight as an incremental upgrade in software. Note that in this case the code of the virus need not be changed and when returned to its original environment virus behaviour may revert to its original form.

The second class of changes are those that manifest in accidental disruption of virus code, or mutation. There are plentiful opportunities for data corruption in the modern computing environment. It would be a rare user who has not faced a system crash or a corrupted disk. Stray electromagnetic radiation from electronics can disrupt magnetic media. Faults in hardware or on storage media itself can lead to the creation of new bit patterns, and hence new instructions, in programs. Faulty software can overwrite other code or result in the creation of stray pieces of orphan code.

Such changes have been observed in viruses presently extant, viruses that are fully viable with changes ranging from a single instruction and no functional difference, to multi-instruction changes resulting in vast behavioural aberrations [?, ?]. In addition, at least one case has been seen of two different viruses "crossing over" to produce offspring [?].

The above would seem to confirm that evolution, even if only on a minor scale, can occur within viral populations. It might be argued that such evolution is insufficient to result in any significant change. Modern von Neumann computer programs are computationally brittle, that is the number of viable programs to the total number of possible programs is minuscule. A random change is thus unlikely to create a viable program. This however does not prevent evolution.

The first reason is that of scale. Clearly by the above reasoning the derivatives of viruses are next to impossible, as the changes would have disrupted the viral function. Yet viable virus mutations exist. The amount of computing activity in the world has ensured a large enough number of mutations such that some were viable. Furthermore, with the expansion of computing power in the world the available gene pool for these variants is increasing exponentially each year.

3.2. *Spontaneous evolution of computer viruses*

Several parties have attempted calculations of the likelihood of viruses evolving from "inanimate code" by random mutation [?, ?, ?]. In general, the time estimated for

---

[6]It has been been claimed elsewhere that this is in fact correct [?, ?].

such an event was several times the lifespan of the universe. Yet most studies shared significant flaws in that the calculations were based on a single instance of a random sequence of code evolving into a specific virus of the same length. This is clearly incorrect and on par with watching a dog, in the hope it will turn into a cat.[7] The probability is in fact considerably better, if we consider the sum of all the data stored on computers in the world and the possibility of mutation into any virus. The chances are further boosted by the number of code sequences that are already partly homologous with potential viral code. These are ubiquitous as is evidenced by the common syndrome of virus scanning software erroneously identifying legitimate programs as viral.

Next, we have the evidence contributed by computational ecosystems. These are best typified by their progenitor, Core Wars, and Ray's Tierra model [?]. Briefly, in these models a number of processes in parallel execute instructions held in a shared block of memory. Processes may spawn other processes, overwrite instructions and in effect compete for memory and execution time as resources. In Tierra, mutation is introduced by the random flipping of bits in memory.

The electronic ecologies that eventuated were astounding. The instruction sequences that serve as "organism" analogues spontaneously developed sophisticated behaviours.[8] Among these were complex patterns in memory, flocking, parasitism and immunity, predator-prey relationships and sexual reproduction.

Given that the instructions and operations of the organisms are those of a simple von Neumann assembly language, the ecologies can be seen as a model of viral evolution, indicating that given sufficient time there is no barrier to viruses spontaneously developing sophisticated behaviour. It is also instructive to observe how little code is required to achieve seemingly complex strategies. The progenitor organism in Tierra used 80 instructions.[9] All of the behaviours described were exhibited by creatures of the order of thousands of instructions or less. By comparison, viruses also tend towards the thousands of instructions, although examples have been demonstrated that were less than 40 bytes [?]. The actions that we normally associate with higher organisms may arise in viruses more easily than expected.

3.3. Human influence on the evolution of computer viruses

The role of human intervention in aiding or hindering evolution is an unknown quantity. We have already reflected as to how humanity in some ways is serving as an unwitting vector of natural selection and evolution for the viral population. It is possible that any nascent virus species may be directly exterminated by anti-viral techniques. Yet one must doubt the efficiency of this extermination. Doubts have been expressed as to whether there is any perfect method of virus detection [?]. Further, in a world where computing power is moving more and more into the hands of the layman, blanket adherence to anti-viral strategies is difficult to enforce. Despite an intervening 6 years the first IBM PC virus is still responsible for 7% of all virus outbreaks on that platform, whilst copies of the Internet Worm were extent for up to a year after it's "containment".[10] Viruses with subtle propagation strategies may escape detection for months or even years, perhaps hibernating in non-volatile storage.

---

[7]A more meaningful analogy might be the estimated 10^{120} years it would take for a 100-amino acid protein to arise by sheer chance, cold comfort to the protein-based amongst us.

[8]Authors own work and [?, ?].

[9]That in the author's system used 50.

[10]Information gathered from Virus-l electronic mailing list, maintained by Ken Van Wyk, CERT.

Deliberate and sincere design may indeed directly contribute to viral evolution. Autonomously replicating entities have been considered as useful tools for information gathering, anti-virus software and distributed computing [?, ?]. Widespread use of such agents may provide a "gene-pool" of sophisticated code for mutation and evolution. It has already been mentioned how the increase in computing activity will lead to more opportunity for mutation. At the same time, the move towards interconnection and compatibility provides a vast and rich ecosystem for the development of new species. Reason and experience with computer ecologies tells us that the smaller the instruction set of a program, the more likely a perturbation will result in a functional program. It follows therefore that the recent popularity of RISC (reduced instruction set) computers may lead to an upsurge in virus development.

Conversely, inadvertent human action might disrupt viral evolution. Viruses often rely on subtle facets of machine architectures and operating systems. The march of progress in computer technology may literally overnight obliterate entire viral species, as the computational environment in which a virus thrives is either junked or changed to the point where the virus is dysfunctional. An example of this would be the supplanting of the Apple II and CP/M machines, with the consequent reduction in viruses arising on those machines. Yet again there are exceptions. Outdated computer technology can persist for long periods, particularly in economically disadvantaged areas or times, where the money for anti-viral technology is also less likely to be available. With backwards and sideways compatibility appearing more and more in new products, the Apple-IBM PowerPC being a good example, viruses may also transfer themselves to new and compatible ecosystems. Finally, it is not impossible for a viral species to cross computing platform barriers. A programmer may adapt a pre-existing virus for a new environment. Alternatively, a virus written in a high level language may be able to survive the transition to a different platform. Thompson [?] has already described a virus using the AT&T C compiler as a host that can infect other C compilers and persist across platforms, software upgrades and changes of operating system.

4. Discussion

What then would be the consequences of this viral evolution? At this point our thoughts must be necessarily speculative. The timeline for the advent of a wholly novel virus is uncertain and such an event may be undetectable due to the circumstances under which viruses are produced. It is clear however that with the trends detailed above, the probability of such an event increases with every passing year.

What form these new lifeforms might take is unknown and perhaps unknowable, somewhat akin to trying to predict the shape of the modern world based on observation of the Pre-Cambrian era. Our experience with computational ecosystems shows no evidence that there is any upper limit on the complexity of electronic life given sufficient time. It is possible that soon we may be faced with an alien presence in our machines, a life that could combine aspects of the biological and computational world. Such organisms may be able to rewrite their own instructions, replicate at astounding rates or hide in storage for years. As electronic life moves out of a prebiotic stage, the rate of evolution may even accelerate exponentially, causing them rise towards and beyond us.

What should our attitude be to such new life? Some may argue that it is ethically incorrect to interfere with a nascent lifeform, that we have no more right to exterminate it than we have the right to exterminate terrestrial species. There is no doubt that

there is some threat posed to the fidelity of our computing power, a not inconsequential thing where critical functions are increasingly being handed over to automated systems. Simons [?] suggested the nightmare scenario of a time where electronic and terrestrial life no longer could co-exist in peace. This prospect may seem laughably remote, but when we know so little, can any possibility be truly called improbable?

In the question of viral evolution, we have been placed in the position of attempting to gauge the probability of a situation for which we have no precedent other than the single instance of life occurring on our own planet. In many ways, this is similar to the question of life on other planets and the estimates made by some researchers, where for any conclusions to be drawn at all one must make assumptions bordering on the outrageous. The reader will notice that the title of this paper ends with a question mark. The author will not pretend that his conclusions are unassailable. Viral evolution may not be probable or even vaguely probable but it is possible. And one is aware that given sufficient time, even the most improbable event will occur.

References

[1] Ferbrache D. (1992), *A Pathology of Computer Viruses*, Berlin: Springer-Verlag.

[2] Spafford E. H. (1992), "Computer viruses — a form of artificial life?" *Artificial Life II*, Langton C. G., Taylor C., Farmer J. D. & Rasmussen S. (eds.), Menlo Park, California: Addison-Wesley, pp. 727–745.

[3] Farmer F. D. & Belin A. (1992), "Artificial Life: the coming evolution," *Artificial Life II*, Langton C. G., Taylor C., Farmer J. D. & Rasmussen S. (eds.), Menlo Park, California: Addison-Wesley, pp. 815–840.

[4] Brooks D. R. & Wiley E. O. (1988), *Evolution as Entropy*, Chicago: University of Chicago Press.

[5] Arms K. & Camp P. S. (1983), *Biology*, New York: Holt, Reinhard and Winston.

[6] Simons G. (1983), *Are Computers Alive?*, London: Harvester Press.

[7] Simons G. (1989), *The Biology of Computer Life*, Oxford: Oxford University Press.

[8] Norstad J. (1990), *Disinfectant On-Line Documentation*, Northwestern University.

[9] Burger R. (1988), *Computer Viruses — a High Tech Disease*, San Francisco: Abacus Software.

[10] Cohen F. (1989), "Computational aspects of computer viruses," *Computers and Security*, **8**, 4, pp. 230–267.

[11] Ray T. S. (1992), "An approach to the synthesis of life," *Artificial Life II*, Langton C. G., Taylor C., Farmer J. D. & Rasmussen S. (eds.), Menlo Park, California: Addison-Wesley, pp. 371–408.

[12] Rasmussen S., Knudsen C. & Feldberg R. (1992), "Dynamics of programmable matter," *Artificial Life II*, Langton C. G., Taylor C., Farmer J. D. & Rasmussen S. (eds.), Menlo Park, California: Addison-Wesley, pp. 211–254.

[13] Croall J. & MacKay I. C. (1988), "Computer viruses," *British Medical Journal*, **297**, 15 October 1988, p. 488.

[14] Shoch J. F. & Hupp J. A. (1982), "The 'Worm' programs — early experiments with a distributed computation," *Communications of the ACM*, **25**, 3, March 1982, pp. 172–180.

[15] Thompson F. (1984), "Reflections on trusting trust," *Communications of the ACM*, **27**, 8, August 1984, pp. 761–763.