Attacking cryptosystems by means of virus machines

Mario J. Pérez-Jiménez

Research Group on Natural Computing
Dpt. Computer Science and Artificial Intelligence
University of Seville, Spain
Academia Europaea (The Academy of Europe)

www.cs.us.es/~marper marper@us.es

20th Brainstorming Week on Membrane Computing
Sevilla, Spain, January 24-26, 2024

www.nature.com/scientificreports

scientific reports

Check for updates

OPEN

Attacking cryptosystems by means of virus machines

Mario J. Pérez-Jiménez[1,2], Antonio Ramírez-de-Arellano[1,2]✉ & David Orellana-Martín[1,2]

The security that resides in the *public-key cryptosystems* relies on the presumed computational hardness of mathematical problems behind the systems themselves (e.g. the *semiprime factorization problem* in the RSA cryptosystem), that is because there is not known any polynomial time (classical) algorithm to solve them. The paper focuses on the computing paradigm of *virus machines* within the area of Unconventional Computing and Natural Computing. Virus machines, which incorporate concepts of virology and computer science, are considered as number computing devices with the environment. The paper designs a virus machine that solves a generalization of the semiprime factorization problem and verifies it formally.

LIFE!

LIFE!

⋆ Replication of the genetic material.

LIFE!

- ⋆ Replication of the genetic material.
- ⋆ Synthesis of proteins.

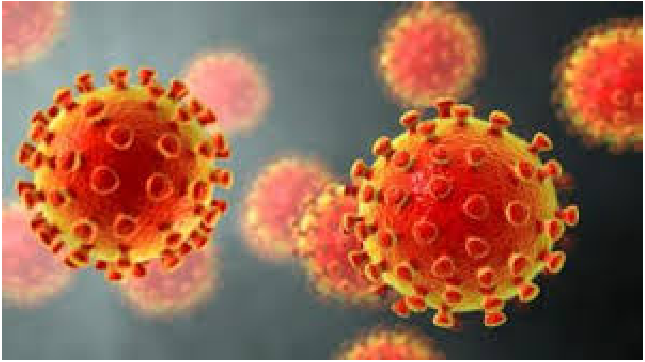
LIFE!

- ⋆ Replication of the genetic material.
- ⋆ Synthesis of proteins.
- ⋆ Production of energy.

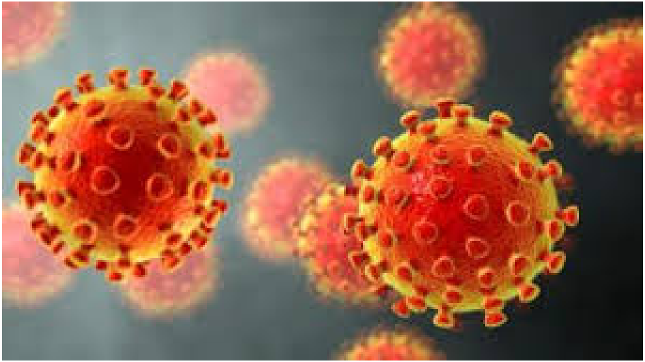
LIFE!

- ⋆ Replication of the genetic material.
- ⋆ Synthesis of proteins.
- ⋆ Production of energy.
- ⋆ Execution of metabolic procceses.

Viruses

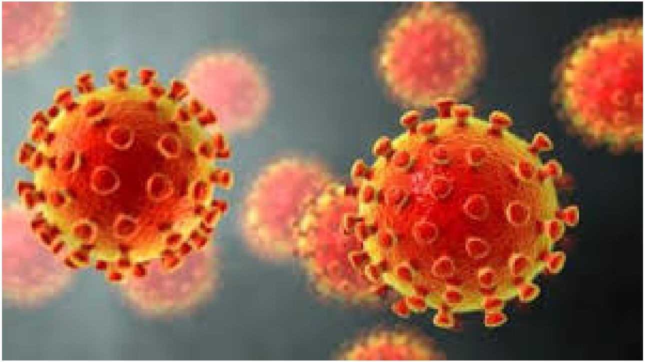


Viruses



Small parasitic biological agents that cannot reproduce by itself.

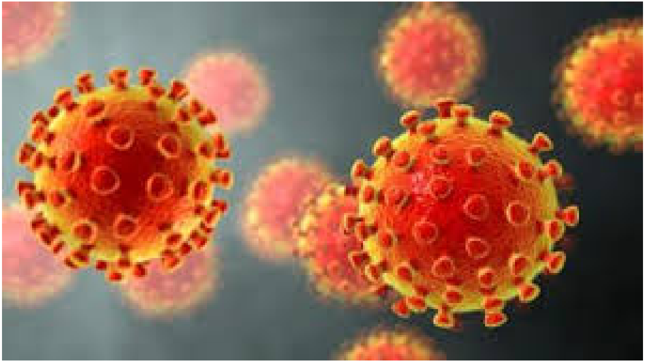
Viruses



Small parasitic biological agents that cannot reproduce by itself.

- ⋆ The most abundant parasites on Earth.

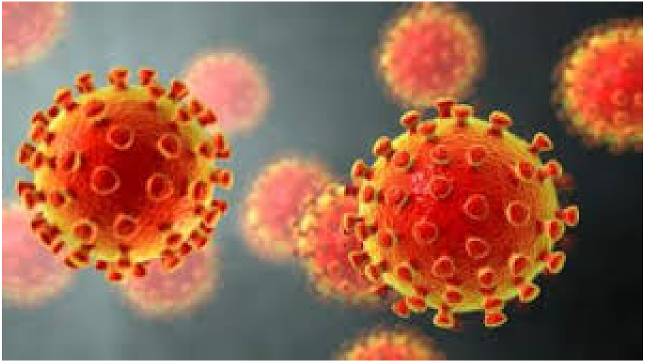
Viruses



Small parasitic biological agents that cannot reproduce by itself.

- ⋆ The most abundant parasites on Earth.
- ⋆ They have not **independent** life.

Viruses

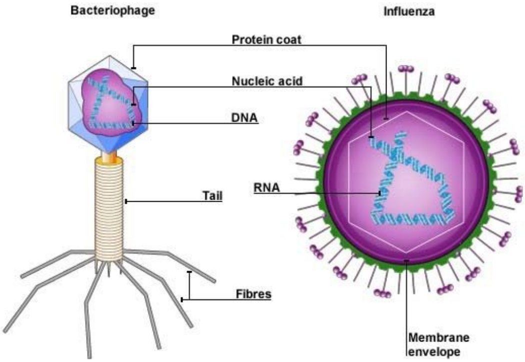


Small parasitic biological agents that cannot reproduce by itself.

- ⋆ The most abundant parasites on Earth.
- ⋆ They have not **independent** life.
- ⋆ Viruses are not lone "wolves".

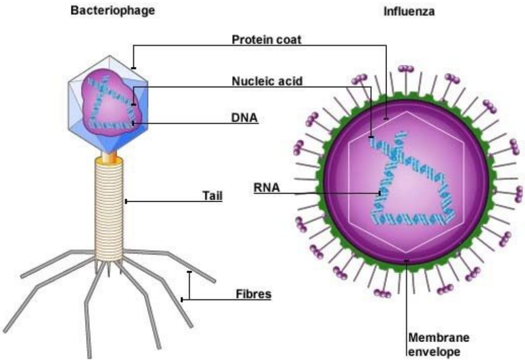
Viruses

A simple structure:



Viruses

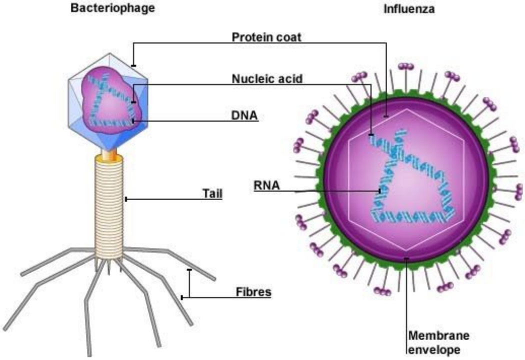
A simple structure:



⋆ Genetic material: either RNA or DNA.

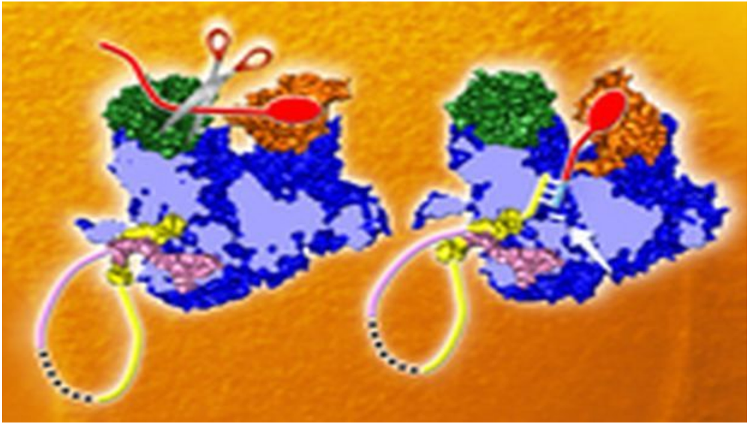
Viruses

A simple structure:



- ⋆ Genetic material: either RNA or DNA.
- ⋆ A protective protein coat.

Virus machines



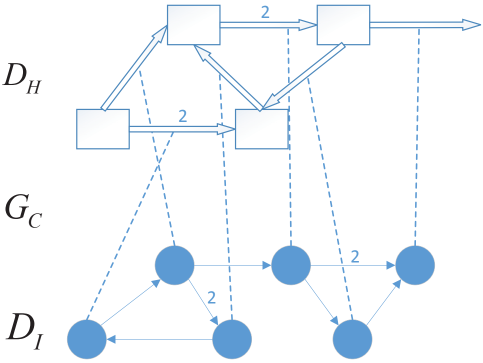
Virus machines

A new computing paradigm inspired by the manner in which viruses transmit from one host to another (introduced in 2015[1]).

[1] L. Valencia, M.J. Pérez-Jiménez, X. Chen, B. Wang, X. Zheng. Basic virus machines. In J.M. Sempere and C. Zandron (eds) **Proceedings of the 16th International Conference on Membrane Computing (CMC16)**, 17-21 August, 2015, Valencia, Spain, pp. 323-342.

Virus machines

A new computing paradigm inspired by the manner in which viruses transmit from one host to another (introduced in 2015[1]).



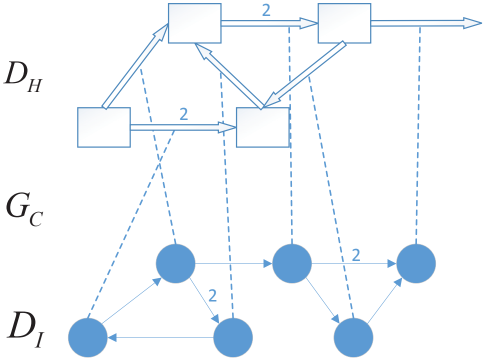
[1] L. Valencia, M.J. Pérez-Jiménez, X. Chen, B. Wang, X. Zheng. Basic virus machines. In J.M. Sempere and C. Zandron (eds) **Proceedings of the 16th International Conference on Membrane Computing (CMC16)**, 17-21 August, 2015, Valencia, Spain, pp. 323-342.

Virus machines

A VM of degree (p, q), $p \geq 1, q \geq 1$: $(\Gamma, H, I, D_H, D_I, G_C, n_1, \ldots, n_p, i_1)$

- ⋆ $\Gamma = \{v\}$ is the singleton alphabet (v is called *virus*).
- ⋆ $H = \{h_1, \ldots, h_p\}$, $I = \{i_1, \ldots, i_q\}$ such that $v \notin H \cup I$ and $H \cap I = \emptyset$.
- ⋆ $D_H = (H \cup \{h_0\}, E_H, w_H)$ is a weighted directed graph: $E_H \subseteq H \times (H \cup \{h_0\})$, and w_H is a mapping from E_H onto $\mathrm{N} \setminus \{0\}$.
- ⋆ $D_I = (I, E_I, w_I)$ is a weighted directed graph, where $E_I \subseteq I \times I$, w_I is a mapping from E_I onto $\mathrm{N} \setminus \{0\}$, and for each vertex $i_j \in I$ the out-degree of i_j is ≤ 2.
- ⋆ $G_C = (V_C, E_C)$ is an undirected bipartite graph, where $V_C = I \cup E_H$ being $\{I, E_H\}$ the partition associated with it. In addition, for each vertex $i_j \in I$, the degree of i_j is less than or equal to 1.
- ⋆ $n_j \in \mathrm{N}$ $(1 \leq j \leq p)$ and $i_1 \in I$.

A Virus Machine of degree $(4, 6)$



Virus machines

The virus machines are equivalent in power to Turing machines[2].

[2] X. Chen, M.J. Pérez-Jiménez, L. Valencia, B. Wang, X. Zeng. Computing with viruses. **Theoretical Computer Science**, **623** (2016), 146-159.

[3] A. Romero, L. Valencia, M.J. Pérez-Jiménez. Generating Diophantine Sets by Virus Machines. In M. Gong, L. Pan, T. Song, K. Tang, X. Zhang (eds) **Bio-Inspired Computing: Theories and Applications. The 10th International Conference (BIC-TA 2015)**, Hefei, China, September 25-28, 2015. Proceedings, pp. 331-341.

[4] A. Romero, L. Valencia, A. Riscos, M.J. Pérez-Jiménez. Computing partial recursive functions by Virus Machines. **Lecture Notes in Computer Science**, **9504** (2015), 353-368.

Virus machines

The virus machines are equivalent in power to Turing machines[2].

They have the ability to:

[2] X. Chen, M.J. Pérez-Jiménez, L. Valencia, B. Wang, X. Zeng. Computing with viruses. **Theoretical Computer Science**, **623** (2016), 146-159.

[3] A. Romero, L. Valencia, M.J. Pérez-Jiménez. Generating Diophantine Sets by Virus Machines. In M. Gong, L. Pan, T. Song, K. Tang, X. Zhang (eds) **Bio-Inspired Computing: Theories and Applications. The 10th International Conference (BIC-TA 2015)**, Hefei, China, September 25-28, 2015. Proceedings, pp. 331-341.

[4] A. Romero, L. Valencia, A. Riscos, M.J. Pérez-Jiménez. Computing partial recursive functions by Virus Machines. **Lecture Notes in Computer Science**, **9504** (2015), 353-368.

Virus machines

The virus machines are equivalent in power to Turing machines[2].

They have the ability to:

⋆ **Generate** all **diophantine sets**[3]

[2] X. Chen, M.J. Pérez-Jiménez, L. Valencia, B. Wang, X. Zeng. Computing with viruses. **Theoretical Computer Science**, **623** (2016), 146-159.

[3] A. Romero, L. Valencia, M.J. Pérez-Jiménez. Generating Diophantine Sets by Virus Machines. In M. Gong, L. Pan, T. Song, K. Tang, X. Zhang (eds) **Bio-Inspired Computing: Theories and Applications. The 10th International Conference (BIC-TA 2015)**, Hefei, China, September 25-28, 2015. Proceedings, pp. 331-341.

[4] A. Romero, L. Valencia, A. Riscos, M.J. Pérez-Jiménez. Computing partial recursive functions by Virus Machines. **Lecture Notes in Computer Science**, **9504** (2015), 353-368.

Virus machines

The virus machines are equivalent in power to Turing machines[2].

They have the ability to:

- ⋆ **Generate** all **diophantine sets**[3]
- ⋆ **Compute** all **recursive functions**[4].

[2] X. Chen, M.J. Pérez-Jiménez, L. Valencia, B. Wang, X. Zeng. Computing with viruses. **Theoretical Computer Science**, **623** (2016), 146-159.

[3] A. Romero, L. Valencia, M.J. Pérez-Jiménez. Generating Diophantine Sets by Virus Machines. In M. Gong, L. Pan, T. Song, K. Tang, X. Zhang (eds) **Bio-Inspired Computing: Theories and Applications. The 10th International Conference (BIC-TA 2015)**, Hefei, China, September 25-28, 2015. Proceedings, pp. 331-341.

[4] A. Romero, L. Valencia, A. Riscos, M.J. Pérez-Jiménez. Computing partial recursive functions by Virus Machines. **Lecture Notes in Computer Science**, **9504** (2015), 353-368.

Cryptography

Cryptography

Concerns the **security of the information** in the presence of possible **intruders**, as well as authentication and identification, providing privacy and integrity.

Cryptography

Concerns the **security of the information** in the presence of possible **intruders**, as well as authentication and identification, providing privacy and integrity.

Cryptosystems symmetrics: with secret key where the key to encrypt and decrypt the text, is the same.

Cryptography

Concerns the **security of the information** in the presence of possible **intruders**, as well as authentication and identification, providing privacy and integrity.

Cryptosystems symmetrics: with secret key where the key to encrypt and decrypt the text, is the same.

Cryptosystems asymmetrics: the issuer has both a public and a private key.

Cryptography

Concerns the **security of the information** in the presence of possible **intruders**, as well as authentication and identification, providing privacy and integrity.

Cryptosystems symmetrics: with secret key where the key to encrypt and decrypt the text, is the same.

Cryptosystems asymmetrics: the issuer has both a public and a private key.

Within asymmetric cryptography, highlights the **public-key cryptosystems**.

Cryptography

Concerns the **security of the information** in the presence of possible **intruders**, as well as authentication and identification, providing privacy and integrity.

Cryptosystems symmetrics: with secret key where the key to encrypt and decrypt the text, is the same.

Cryptosystems asymmetrics: the issuer has both a public and a private key.

Within asymmetric cryptography, highlights the **public-key cryptosystems**.

The security of the cryptosystems relies on the presumed computational hardness of a mathematical problem associated with them.

The RSA cryptosystem

[5] W. Diffie, M. Hellman. New directions in cryptography. IEEE Transactions on Information Theory, **22**, 6 (1976), 644-654.

[6] R.L. Rivest, A. Shamir, L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. CAMC, **21**, 2 (1978), 120-126.

The RSA cryptosystem

The pioneers of the **public-key cryptosystems** were W. Diffie and M. Hellman who formulated the theoretical conditions such cryptosystems should satisfy [5].

[5] W. Diffie, M. Hellman. New directions in cryptography. **IEEE Transactions on Information Theory**, **22**, 6 (1976), 644-654.

[6] R.L. Rivest, A. Shamir, L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. **CAMC**, **21**, 2 (1978), 120-126.

The RSA cryptosystem

The pioneers of the **public-key cryptosystems** were W. Diffie and M. Hellman who formulated the theoretical conditions such cryptosystems should satisfy [5].

RSA cryptosystem: R. **R**ivest, A. **S**hamir and L. **A**dleman in 1978 [6].

[5] W. Diffie, M. Hellman. New directions in cryptography. **IEEE Transactions on Information Theory**, **22**, 6 (1976), 644-654.

[6] R.L. Rivest, A. Shamir, L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. **CAMC**, **21**, 2 (1978), 120-126.

The RSA cryptosystem

The pioneers of the **public-key cryptosystems** were W. Diffie and M. Hellman who formulated the theoretical conditions such cryptosystems should satisfy [5].

RSA cryptosystem: R. **R**ivest, A. **S**hamir and L. **A**dleman in 1978 [6].

RSA cryptosystem: the first public-key cryptosystem verifying the Diffie-Hellman conditions.

---

[5] W. Diffie, M. Hellman. New directions in cryptography. **IEEE Transactions on Information Theory**, **22**, 6 (1976), 644-654.

[6] R.L. Rivest, A. Shamir, L. Adleman. A method for obtaining digital signatures and public-key cryptosystems. **CAMC**, **21**, 2 (1978), 120-126.

The RSA cryptosystem

The RSA cryptosystem

The underlying problem of the **RSA** system is the **semiprime factorization problem**:

The **RSA** cryptosystem

The underlying problem of the **RSA** system is the **semiprime factorization problem**: "**given a semiprime number, find its decomposition**"

The RSA cryptosystem

The underlying problem of the **RSA** system is the **semiprime factorization problem**: "**given a semiprime number, find its decomposition**"

(Semiprime: the product of exactly two prime numbers).

The RSA cryptosystem

The underlying problem of the **RSA** system is the **semiprime factorization problem**: "**given a semiprime number, find its decomposition**"

(Semiprime: the product of exactly two prime numbers).

This problem can be characterized by the following partial function FACT: for each semiprime $x = y \cdot z$, with $y \geq z \geq 2$, we have $\texttt{FACT}(x) = z$.

The RSA cryptosystem

The underlying problem of the **RSA** system is the **semiprime factorization problem**: "**given a semiprime number, find its decomposition**"

(Semiprime: the product of exactly two prime numbers).

This problem can be characterized by the following partial function FACT: for each semiprime $x = y \cdot z$, with $y \geq z \geq 2$, we have $\texttt{FACT}(x) = z$.

The *semiprime factorization problem* is conjectured to be a **computationally hard** problem.

The RSA cryptosystem

The underlying problem of the **RSA** system is the **semiprime factorization problem**: "**given a semiprime number, find its decomposition**"

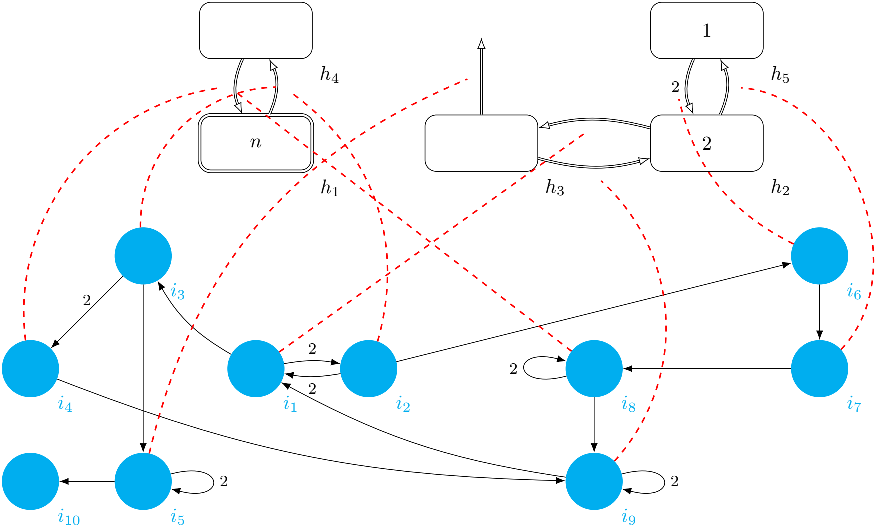
(Semiprime: the product of exactly two prime numbers).

This problem can be characterized by the following partial function FACT: for each semiprime $x = y \cdot z$, with $y \geq z \geq 2$, we have $\texttt{FACT}(x) = z$.

The *semiprime factorization problem* is conjectured to be a **computationally hard** problem.

Any "large" semiprime input n for **RSA** can be used as the *modulus* for both public and private keys.

A VM computing the partial function FACT



THANK YOU

FOR YOUR ATTENTION!