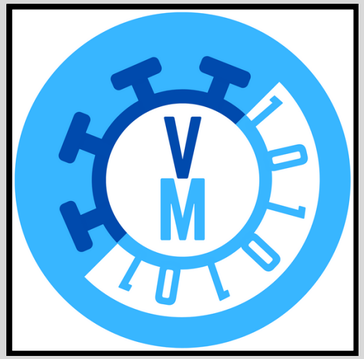
Virus Machines: Tutorial on the Formal Framewok

Antonio Ramírez de Arellano Marrero

20th BWMC &
1st Int. Workshop on Virus Machines



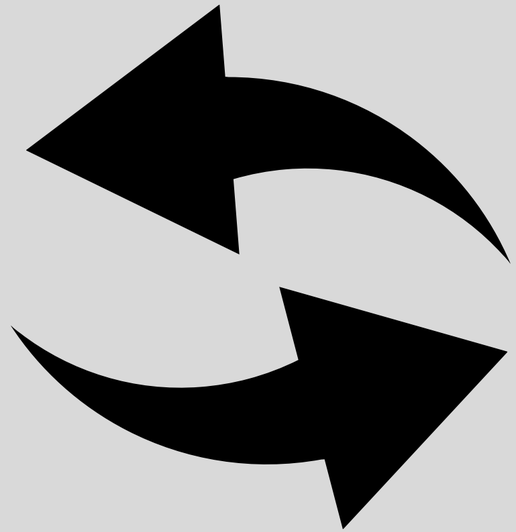
1. Introduction
2. Virus Machines (VM)
3. Formal Verification
4. Conclusion/Open Prob.

Proyect

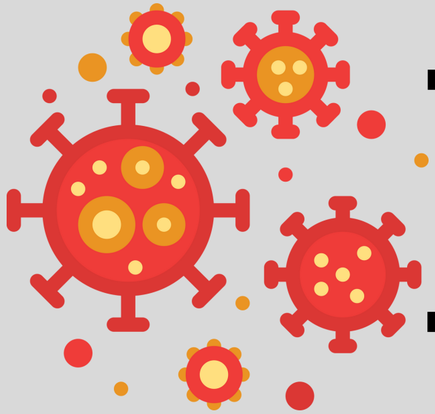
Virus Machines: Theory and Applications
Zhejiang Lab BioBit Program
(Grant No. 2022BCF05)

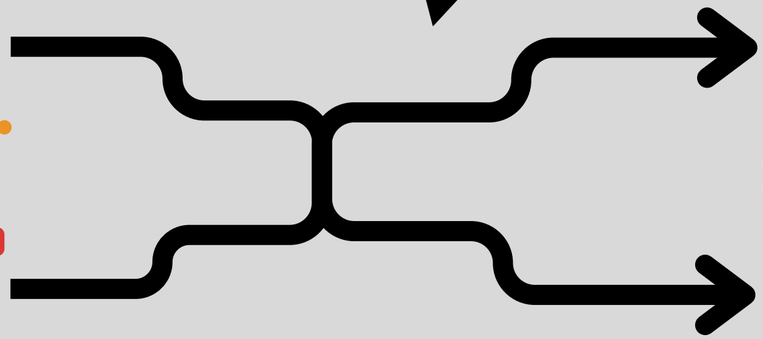
之江实验室
ZHEJIANG LAB
BioBit

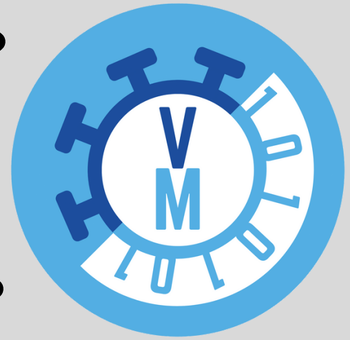
Collab.



Science



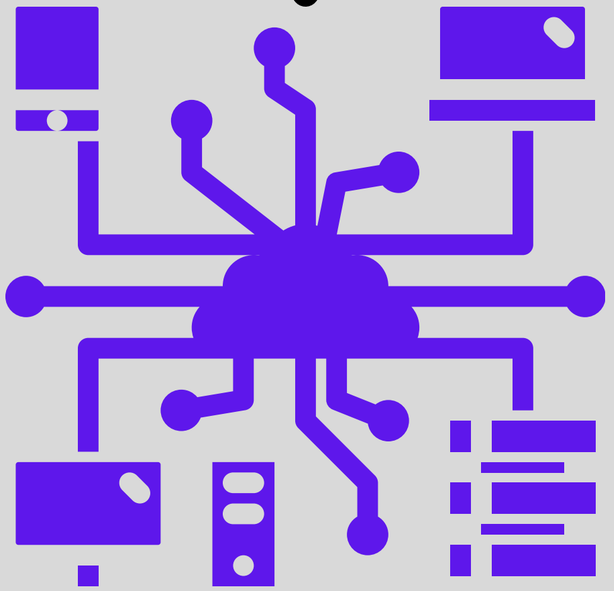




Natural Computing

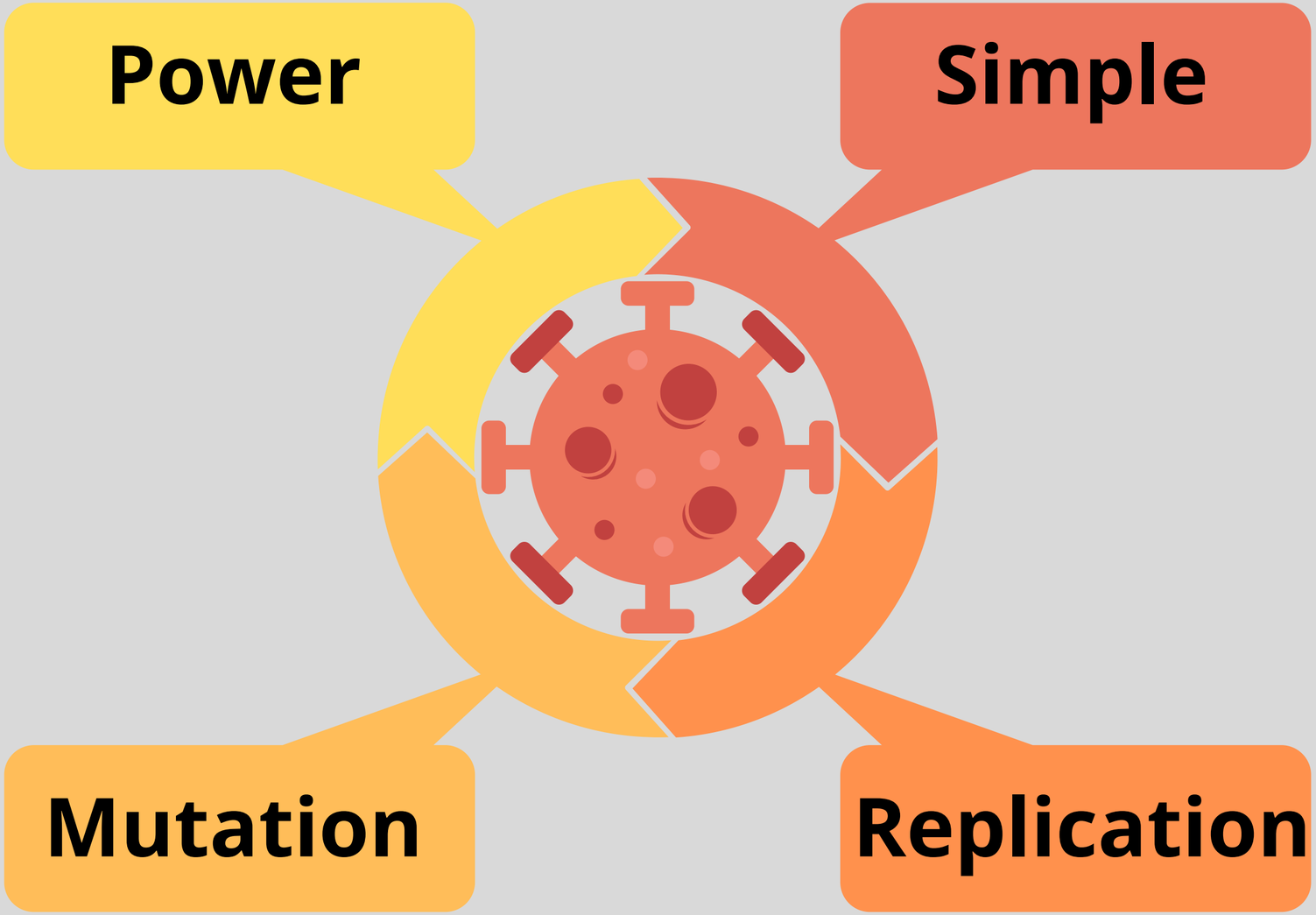
Biological Inspiration

Computing Paradigms



Computing models implementations

Bio-inspiration: Viruses



2. Virus Machines

Virus Machine of degree p,q

$$\Pi = (\Gamma, H, I, D_H, D_I, G_C, n_1, \ldots, n_p, i_1, h_{out})$$

Virus Machine of degree p,q

Singleton Alphabet

$$\Pi = (\Gamma, H, I, D_H, D_I, G_C, n_1, \ldots, n_p, i_1, h_{out})$$

Virus Machine of degree p,q

Singleton Alphabet

$$\Pi = (\Gamma, H, I, D_H, D_I, G_C, n_1, \ldots, n_p, i_1, h_{out})$$

p Hosts

Virus Machine of degree p,q

Singleton Alphabet

q Instructions

$$\Pi = (\Gamma, H, I, D_H, D_I, G_C, n_1, \ldots, n_p, i_1, h_{out})$$

p Hosts

Virus Machine of degree p,q

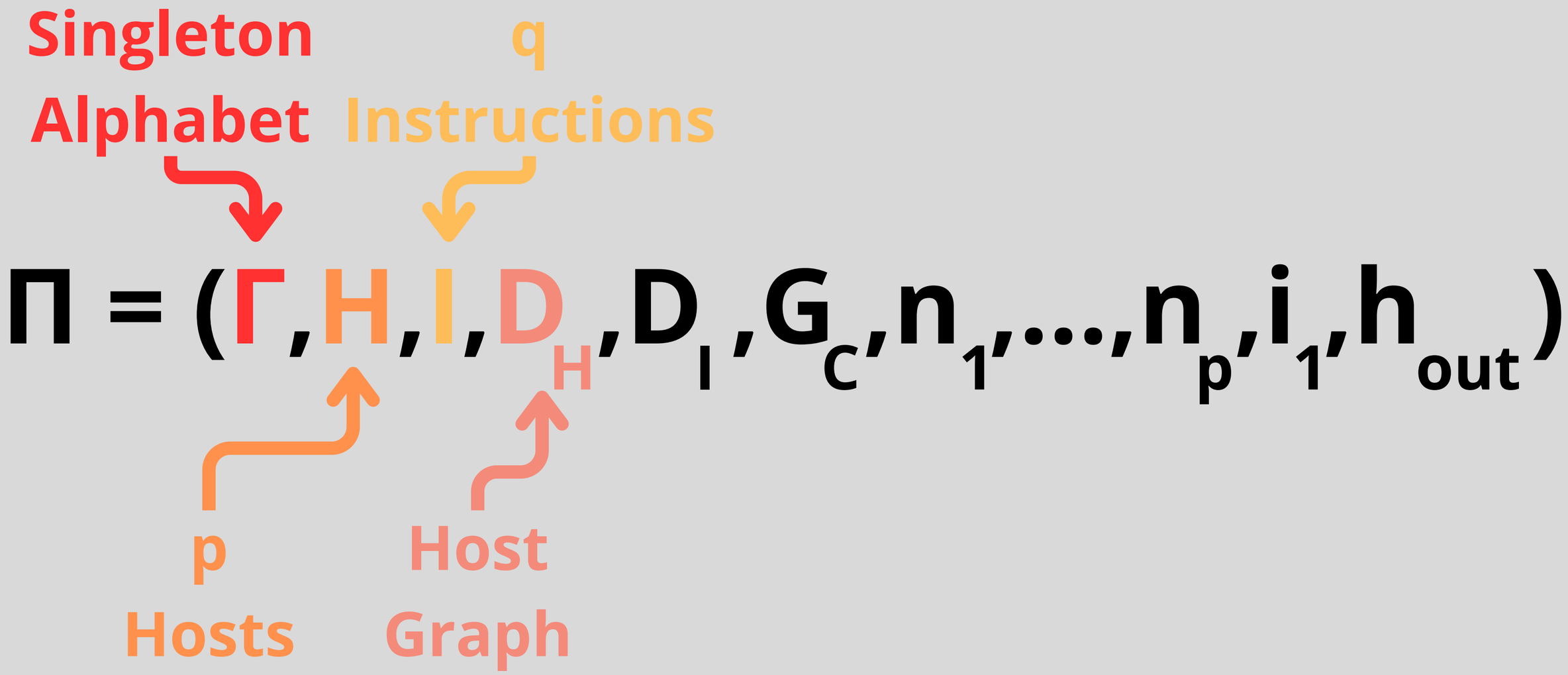
Singleton Alphabet

q Instructions

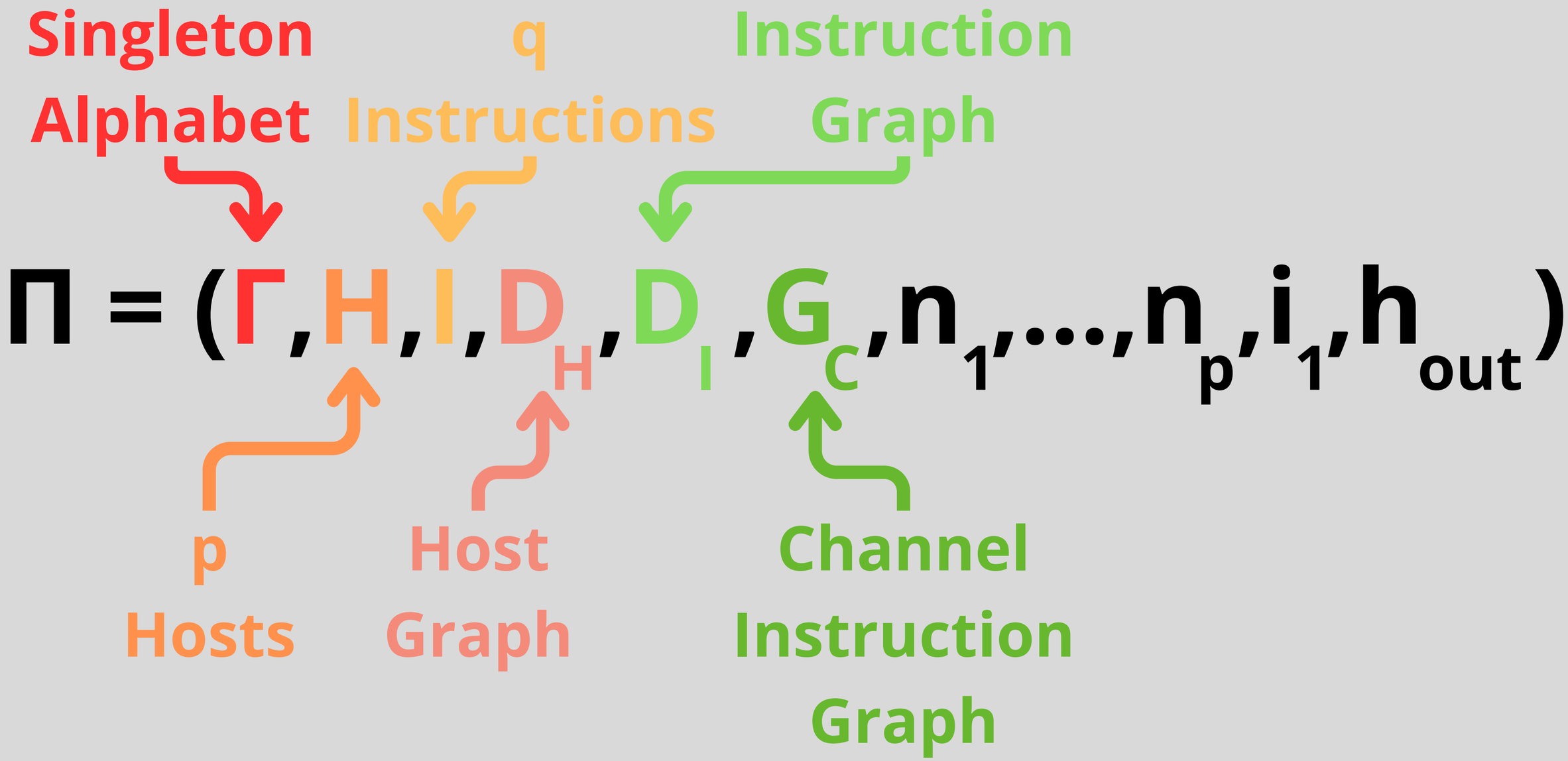
$$\Pi = (\Gamma, H, I, D_H, D_I, G_C, n_1, \ldots, n_p, i_1, h_{out})$$

p Hosts

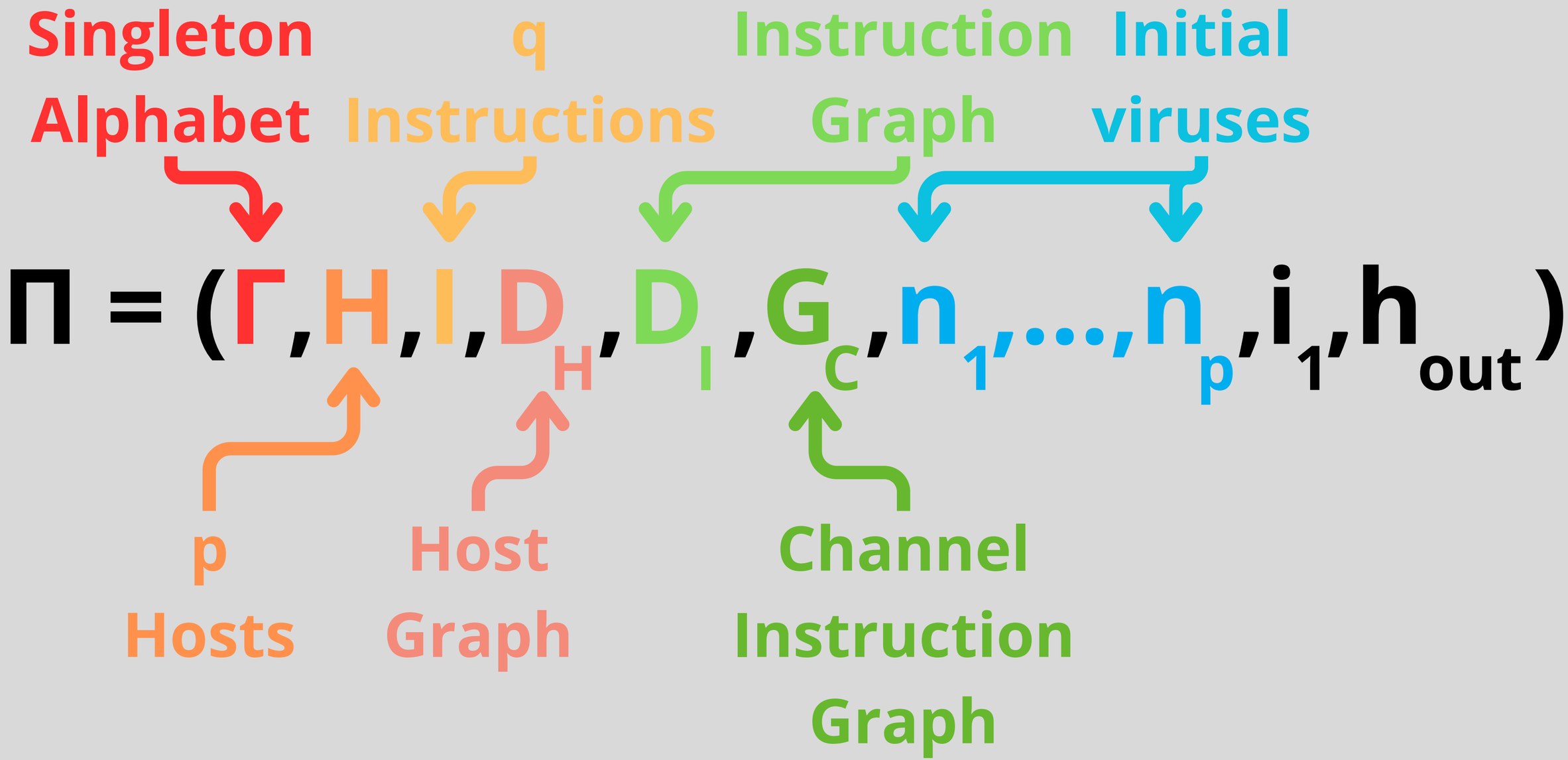
Virus Machine of degree p,q



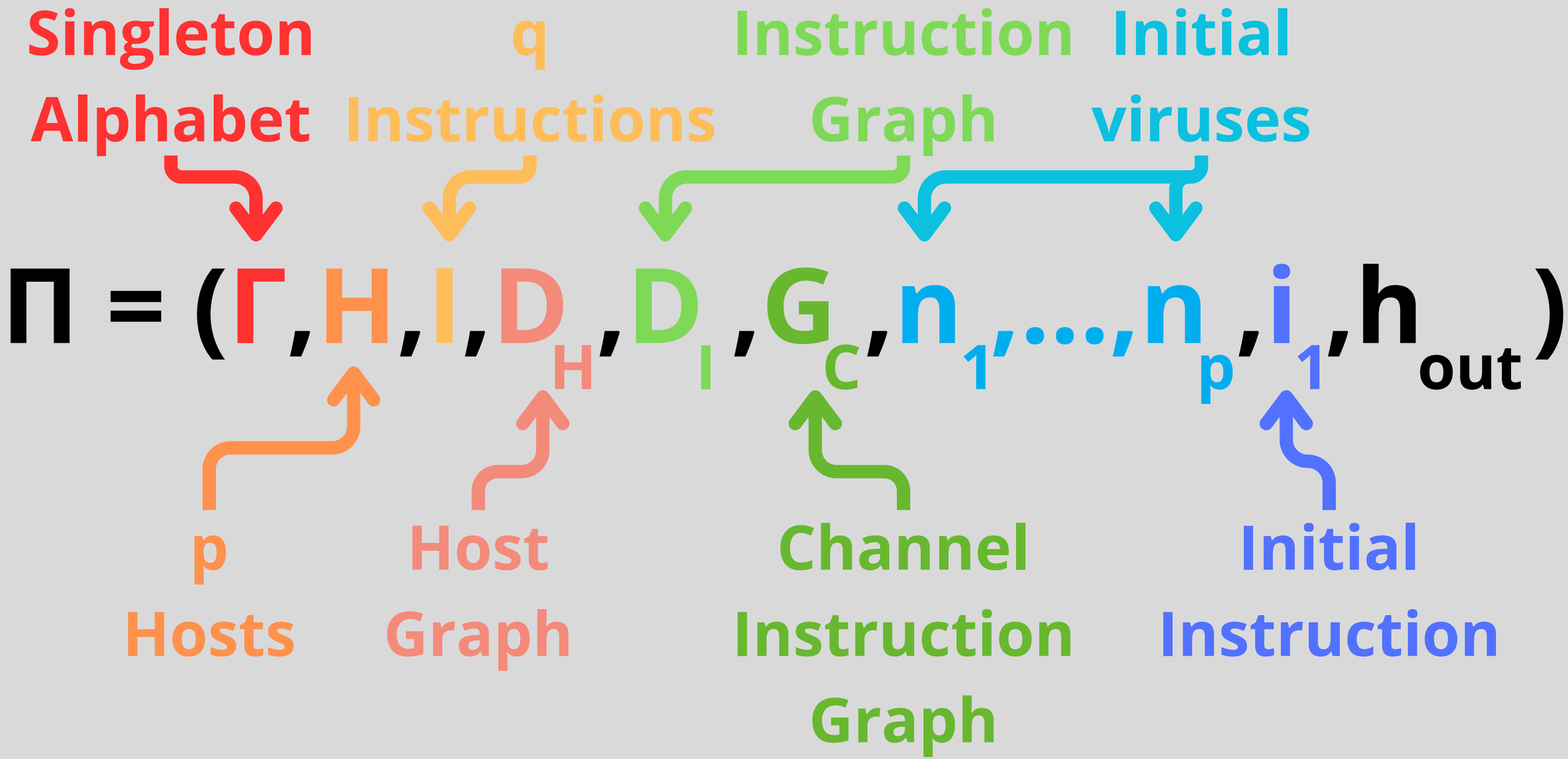
Virus Machine of degree p,q



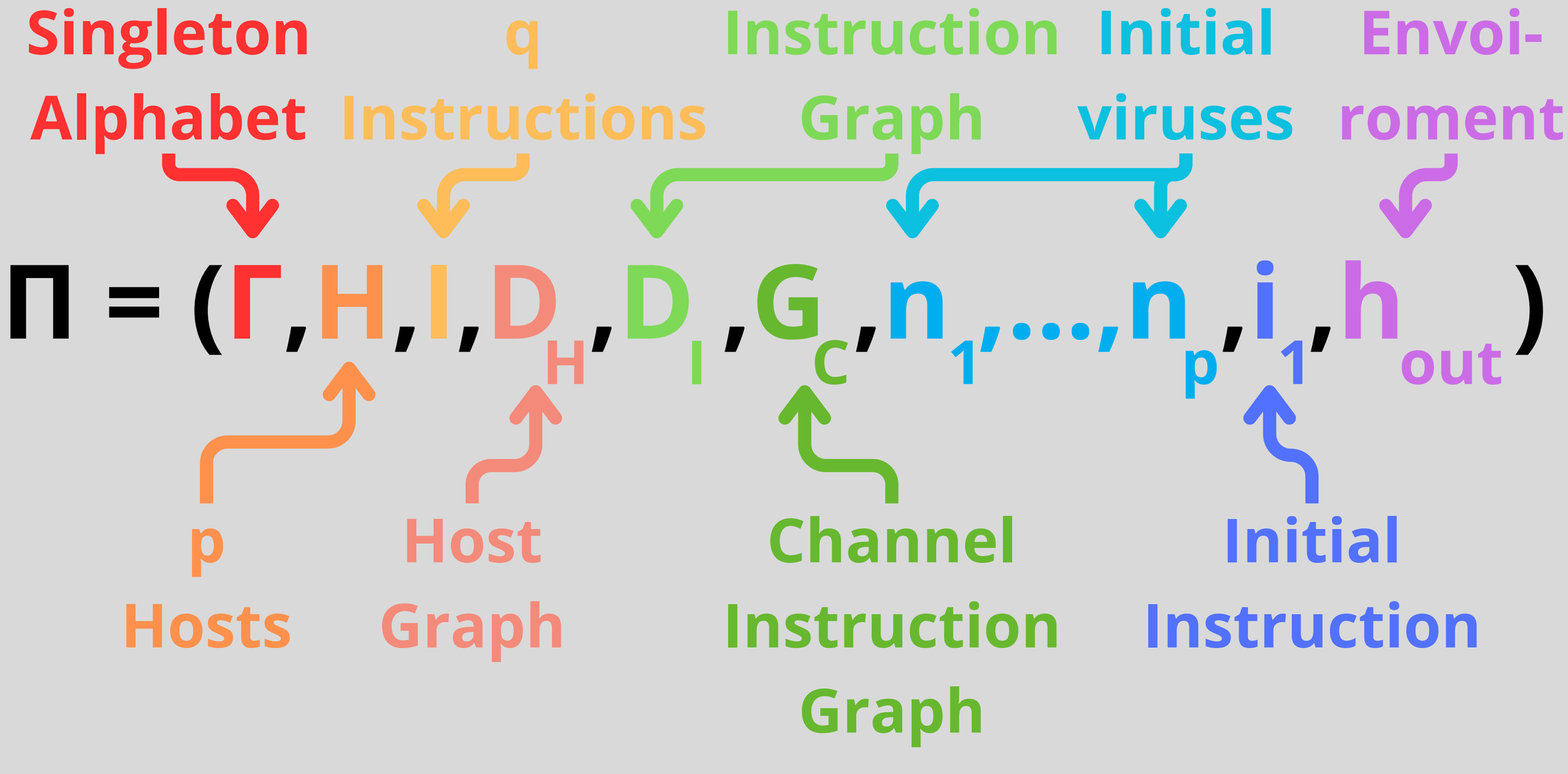
Virus Machine of degree p,q



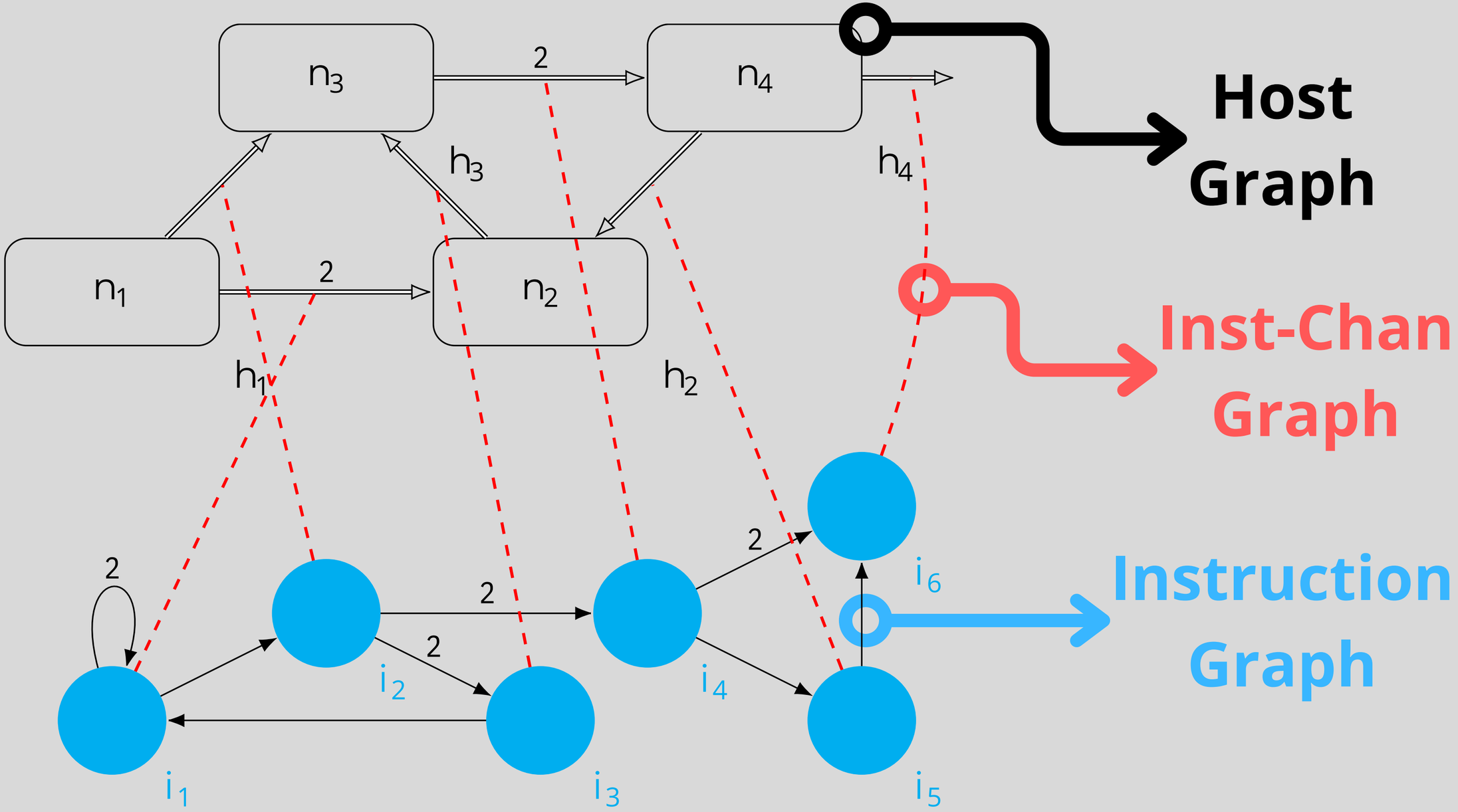
Virus Machine of degree p,q



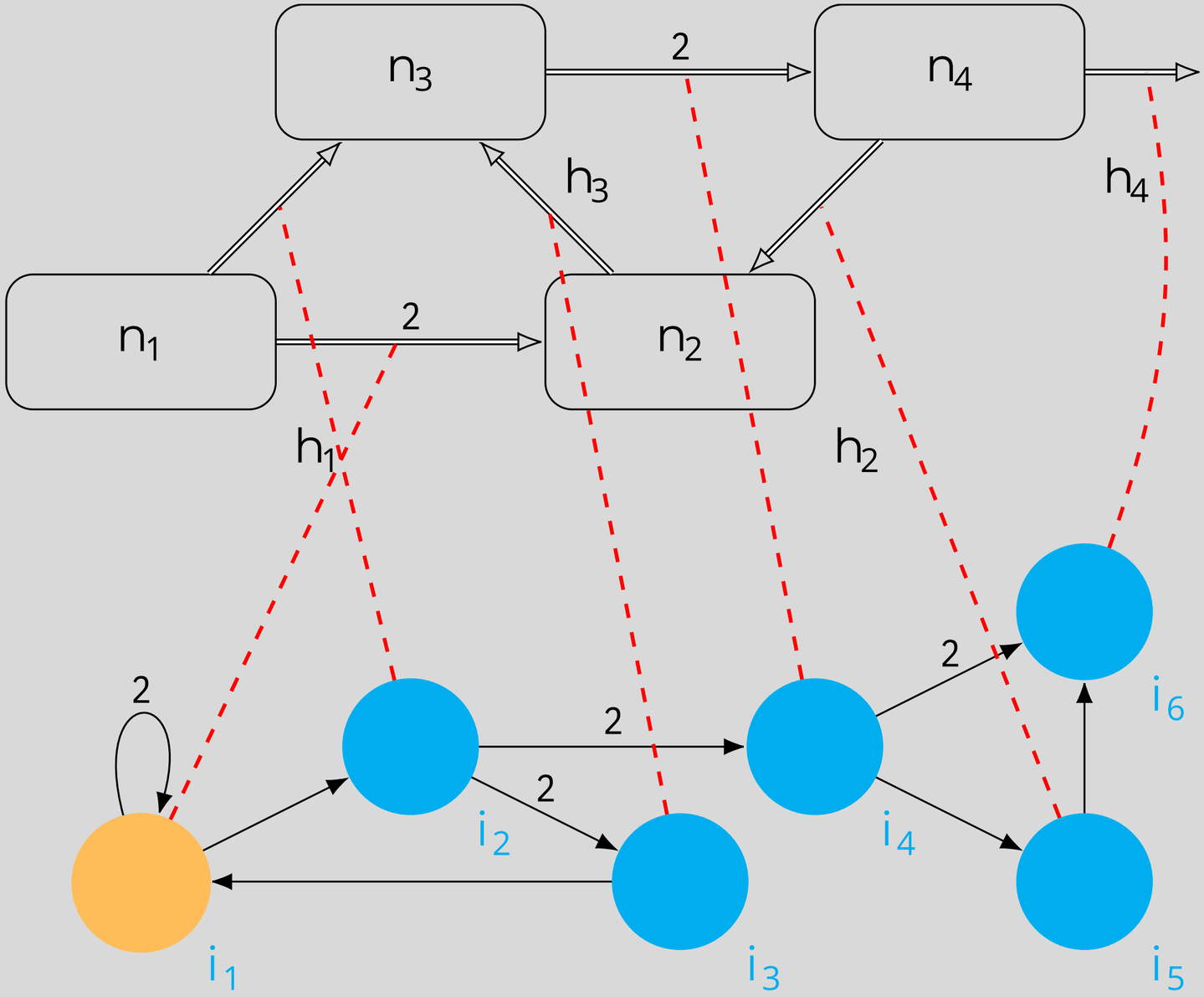
Virus Machine of degree p,q



Example of a VM of degree (p,q)



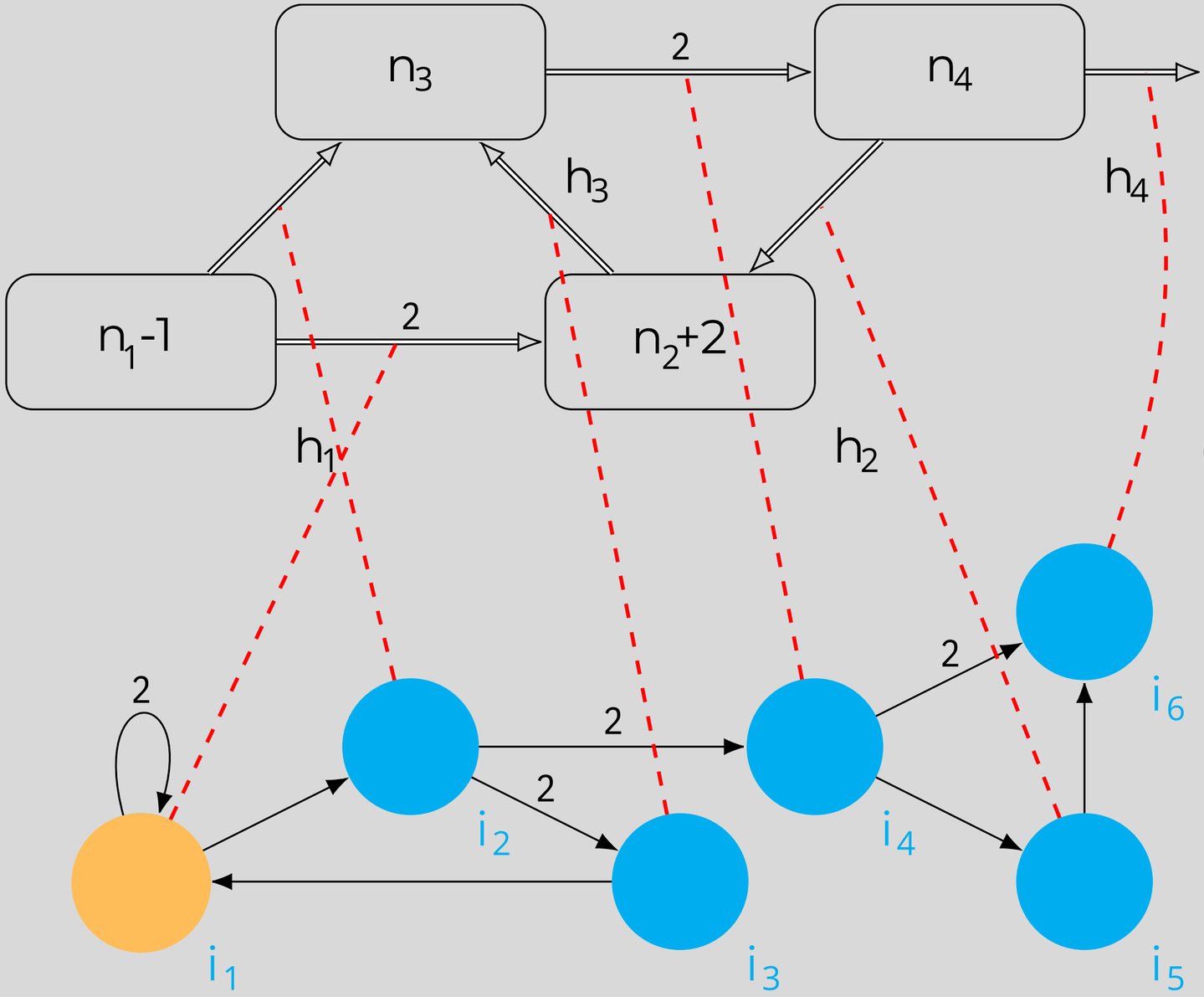
Example of a VM of degree (4,6)



Initial Configuration

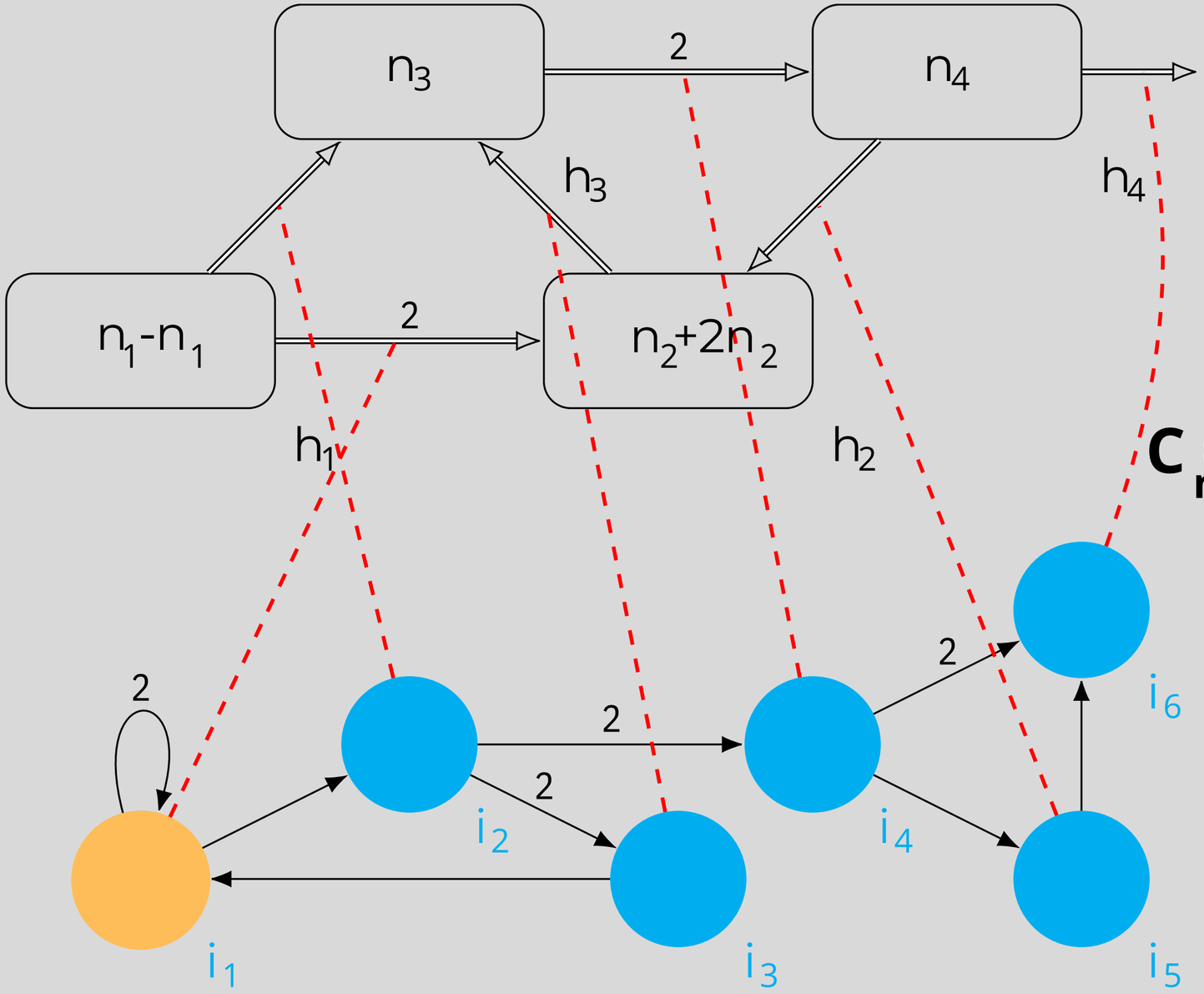
$C_0 = (n_1, n_2, n_3, n_4, i_1, 0)$

Example of a VM of degree (4,6)



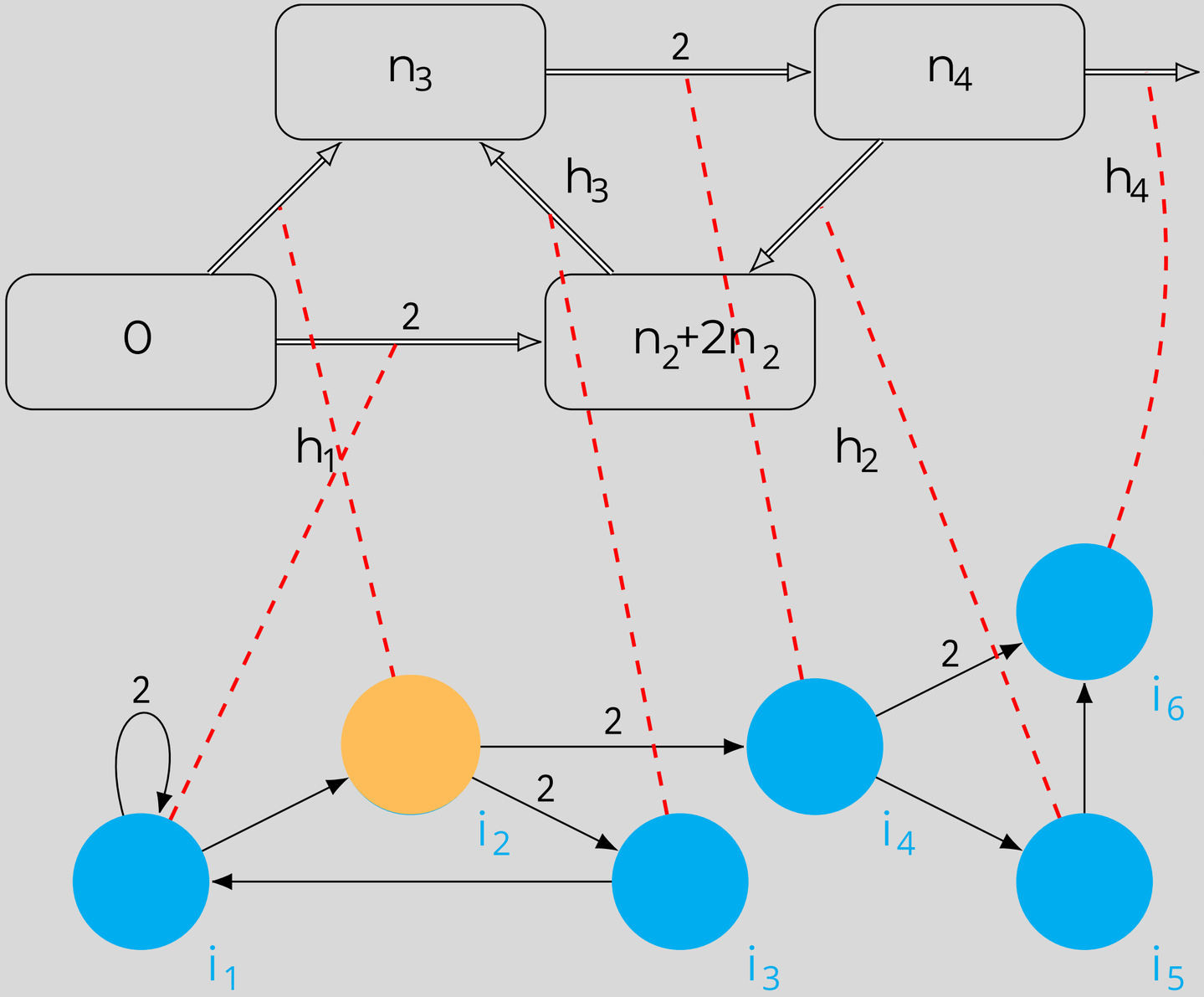
$C_1 = (n_1-1, n_2+2, n_3, n_4, i_1, 0)$

Example of a VM of degree (4,6)



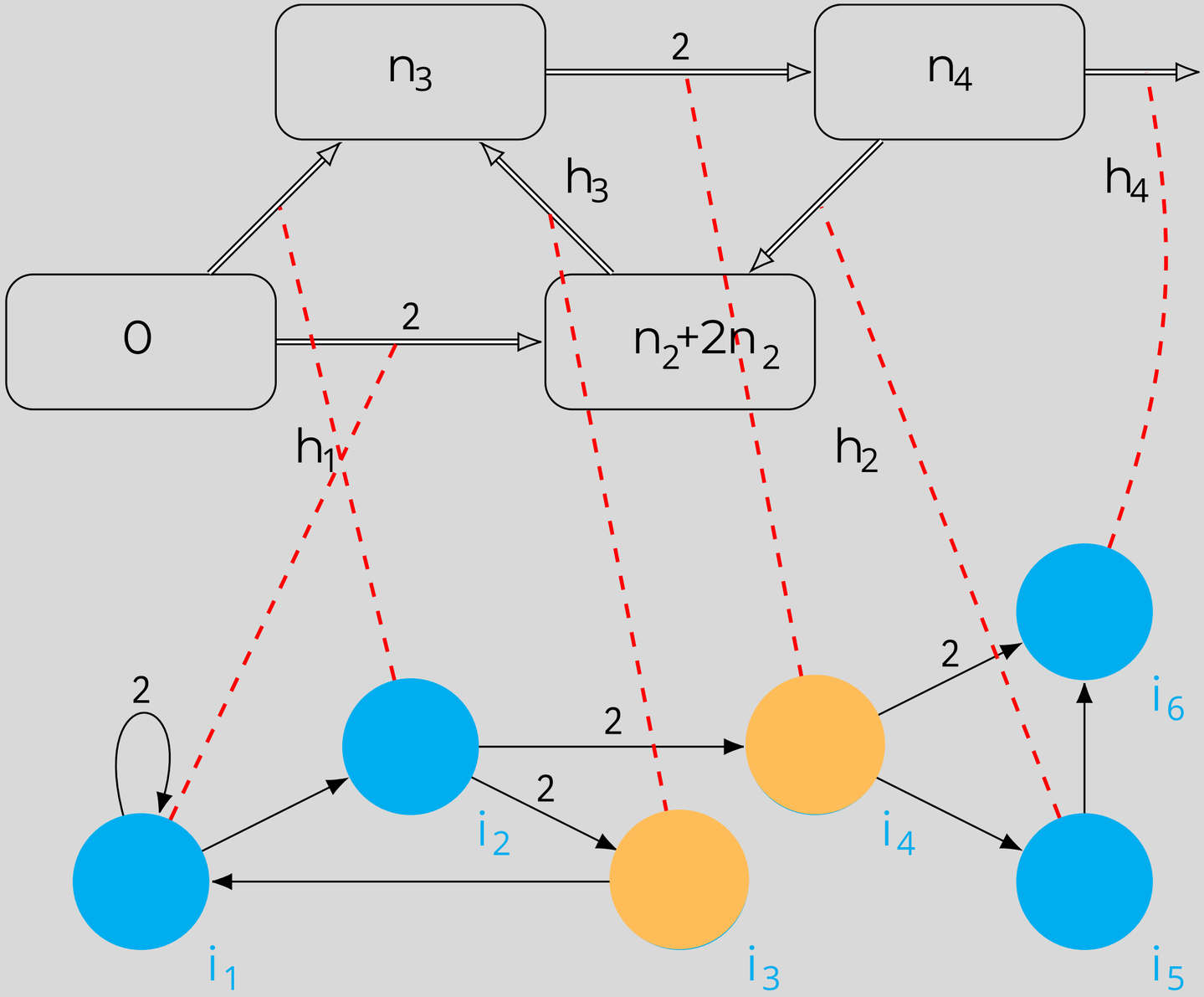
$$C_n = (n_1-n_1, n_2+2n_2, n_3, n_4, i_1, 0)$$

Example of a VM of degree (4,6)



$$C_{n_1+1}(0,n_2+2n_1,n_3,n_4,i_2,0)$$

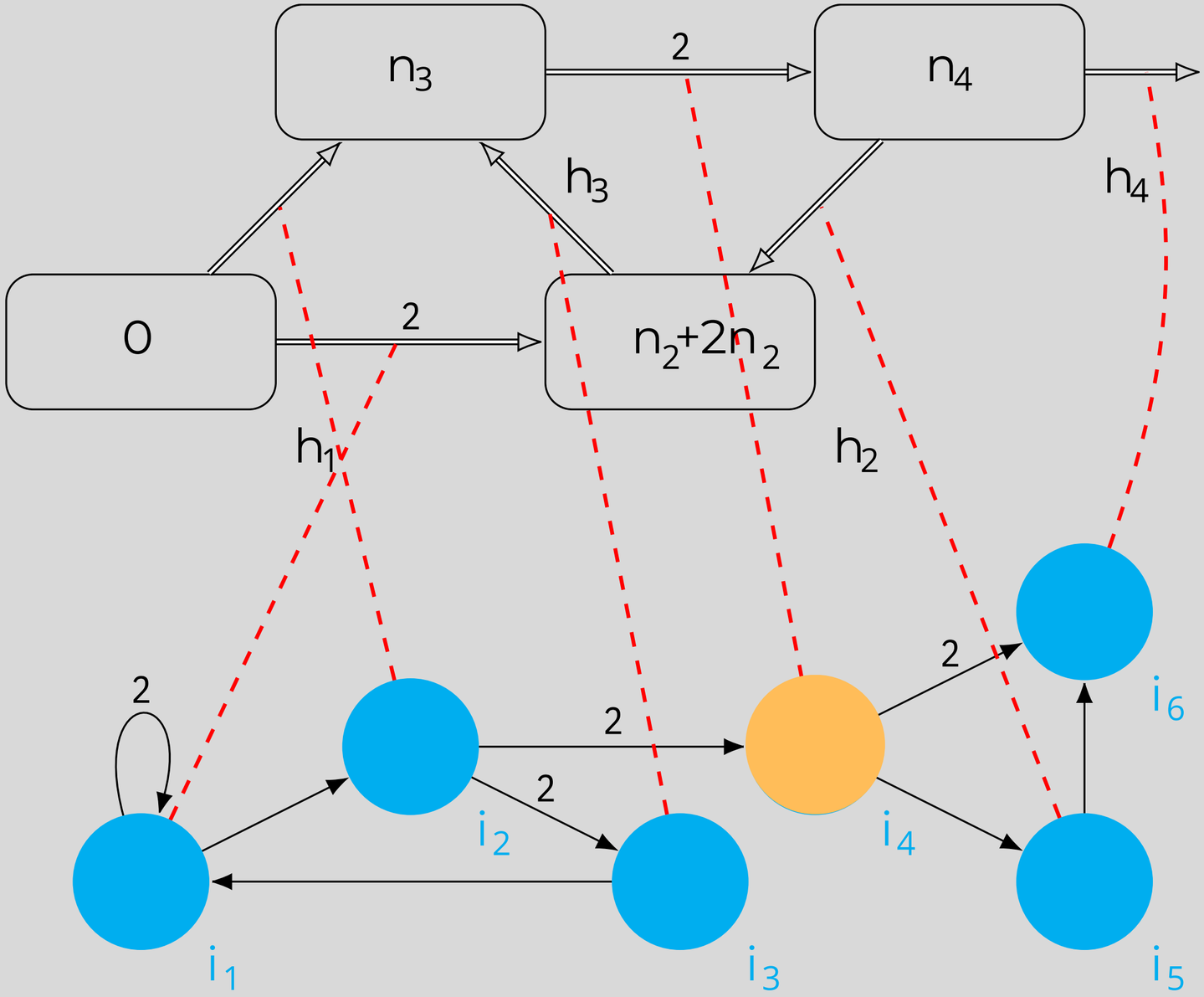
Example of a VM of degree (4,6)



$$C_{n_1+2}(0,n_2+2n_1,n_3,n_4,i_3,0)$$

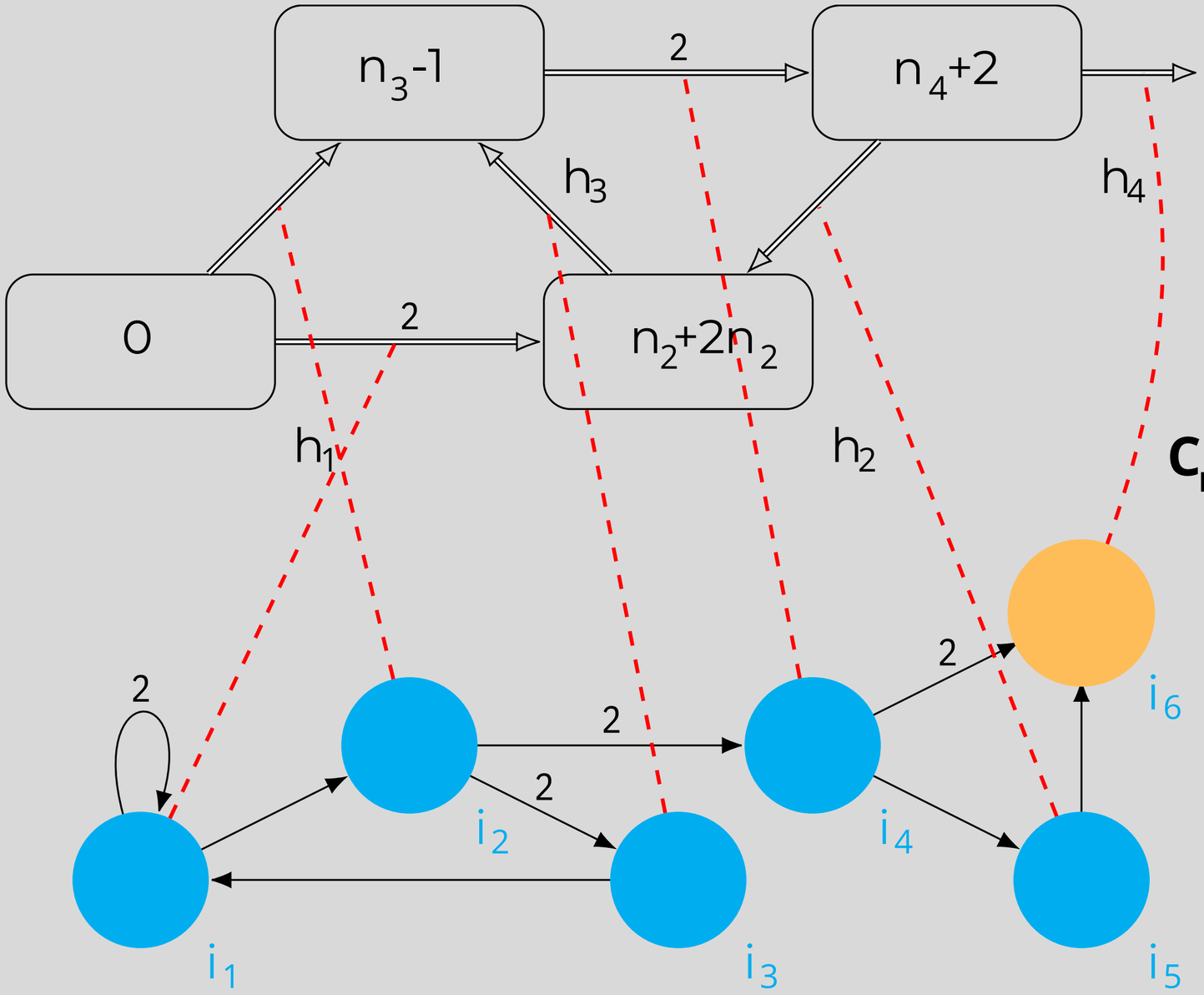
$$C_{n_1+2}(0,n_2+2n_1,n_3,n_4,i_4,0)$$

Example of a VM of degree (4,6)



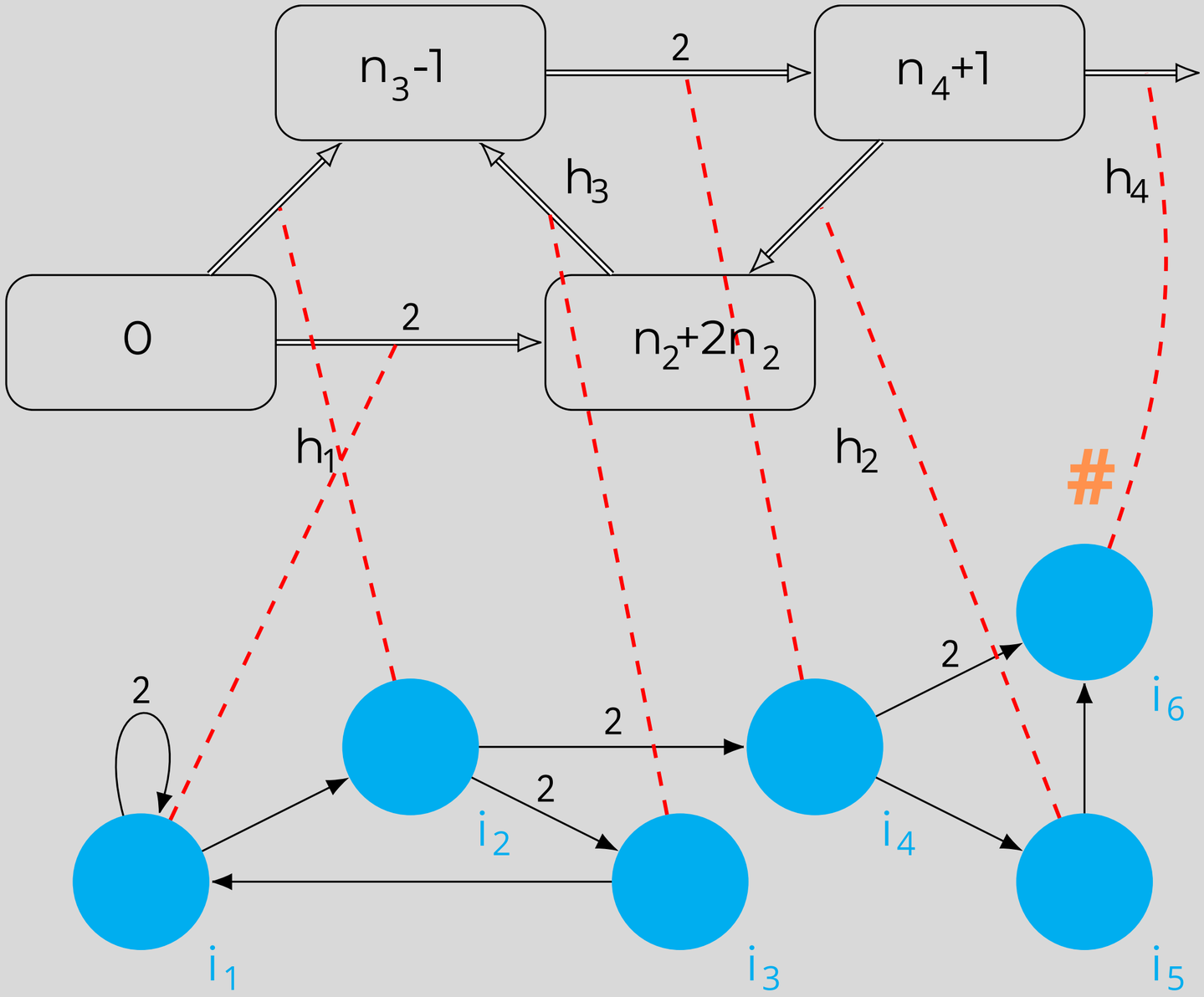
$$C_{n_1+2}(0,n_2+2n_1,n_3,n_4,i_4,0)$$

Example of a VM of degree (4,6)



$C_{n_1+3}(0,n_2+2n_1,n_3-1,n_4+2,i_6,0)$

Example of a VM of degree (4,6)



Halting Configuration

$C_{n_1+4} = (0, n_1+2n_2, n_3-1, n_4+1, \#, 1)$

Past

Universality

- Diophantine sets[1]
- Partial recursive functions[2]
- Register machines[3]

Present

Problem solving

- Pairing functions[4]
- Decision problems[5]
- Cryptosystems[6]

1. Á. Romero-Jiménez et al. ***Generating diop. sets by VM***, Com. Comp. & Inf. Sc. 2015
2. Á. Romero-Jiménez et al. ***Comp. partial rec. func. by VM.*** LNCS 2015
3. L. Valencia et al. ***Comuting with viruses***, *TCS 2016*
4. A. Ramírez-de-Arellano et al. ***Using virus machines to compute pairing funcitons***, IJNS 2023
5. A. Ramírez-de-Arellano et al. ***Generating, computing and recognizing with VM***, TCS 2023
6. M. J. Pérez-Jiménez et al. ***Attacking cryptosystems by means of VM***, SR 2023

3. Formal Verification

Virus Machine of degree p,q,r

$$\Pi = (\Gamma, H, H_r, I, D_H, D_I, G_C, n_1, \ldots, n_p, i_1, h_{out})$$

$$H_r = \{h_1, \ldots, h_r\} \subseteq H$$

Input

Initial config. of $\Pi+(a_1,\ldots,a_r)$

$(a_1, \ldots, a_r) \in N^r$

$C_0 = (n_1+a_1, \ldots, n_r+a_r, n_{r+1}, \ldots, n_p, i_1, 0)$

Definition

A Virus Machine Π of degree p,q,r computes the partial function $f : -N^r \rightarrow N$ if for each $x \in N^r$ all computations of Π+x verify:

- Do not halt if f(x) is not defined;
- Halt and the output is z = f(x)

Example. Addition

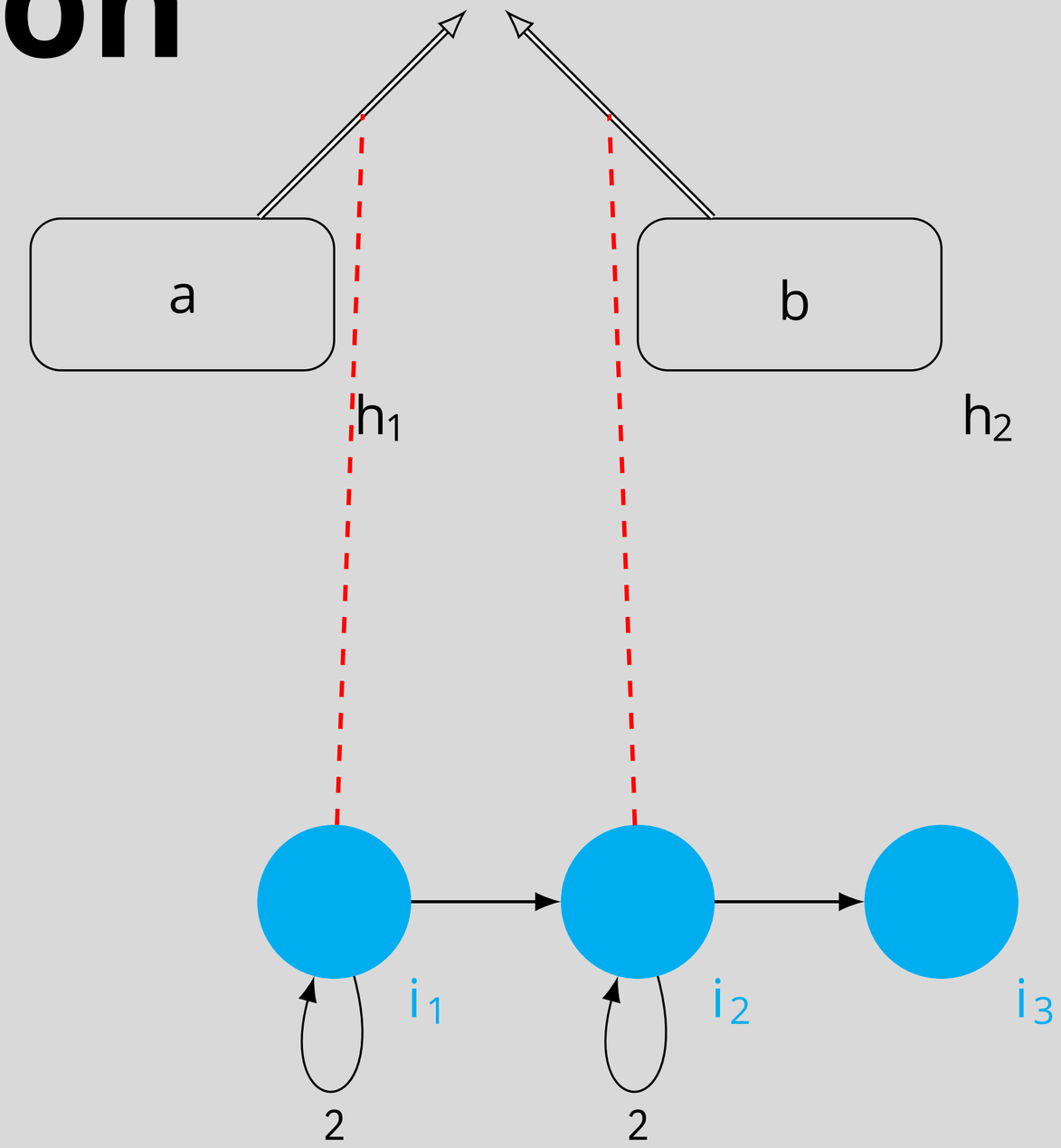
$\Pi_{ADD}+(a,b)$

Initial Conf.

$C_0 = (0+a,0+b,i_1,0)$

Halting Conf.

$C_{a+b+2} = (0,0,\#,a+b)$



Example. Subtraction

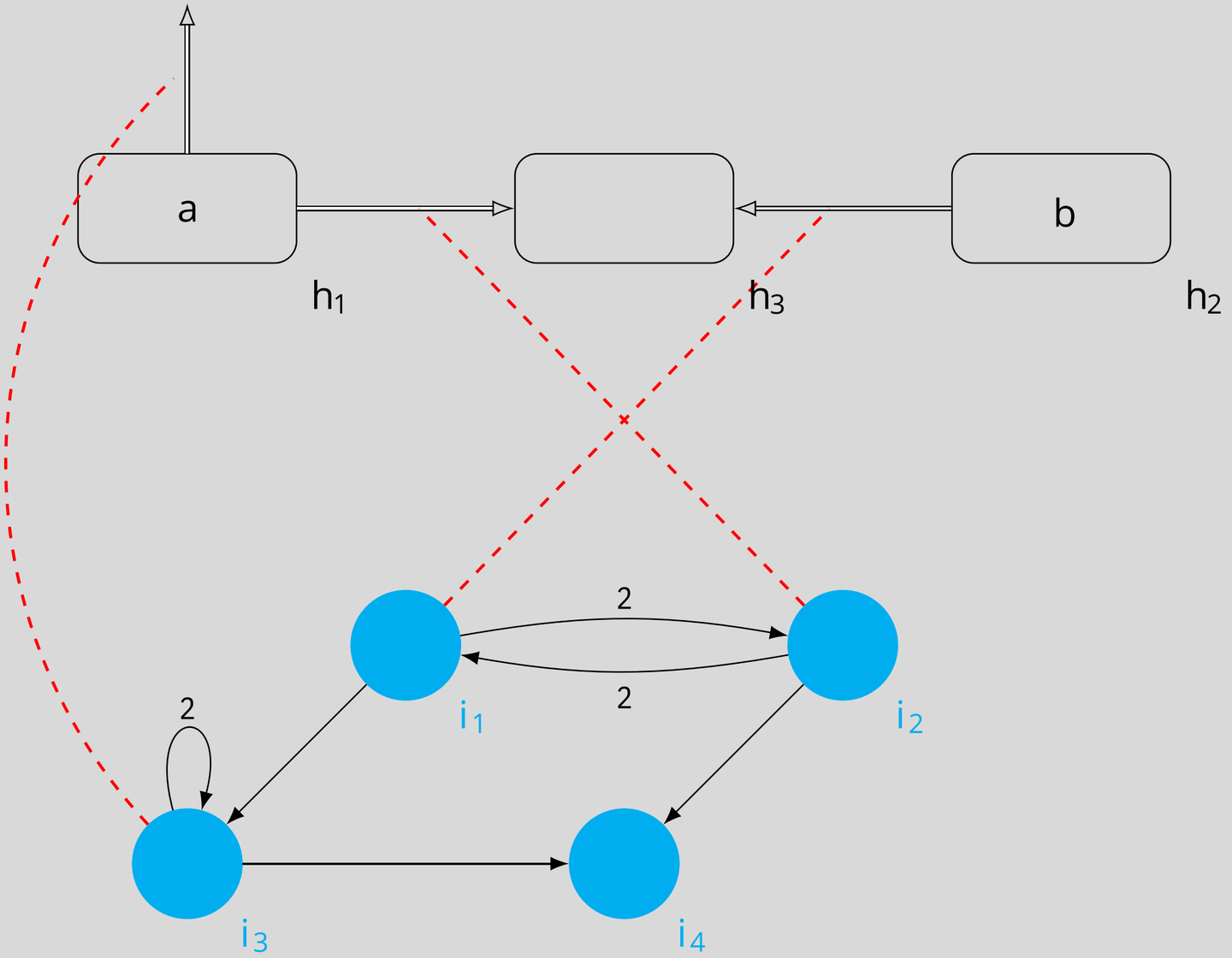
$\Pi_{SUB}+(a,b)$

Initial Conf.

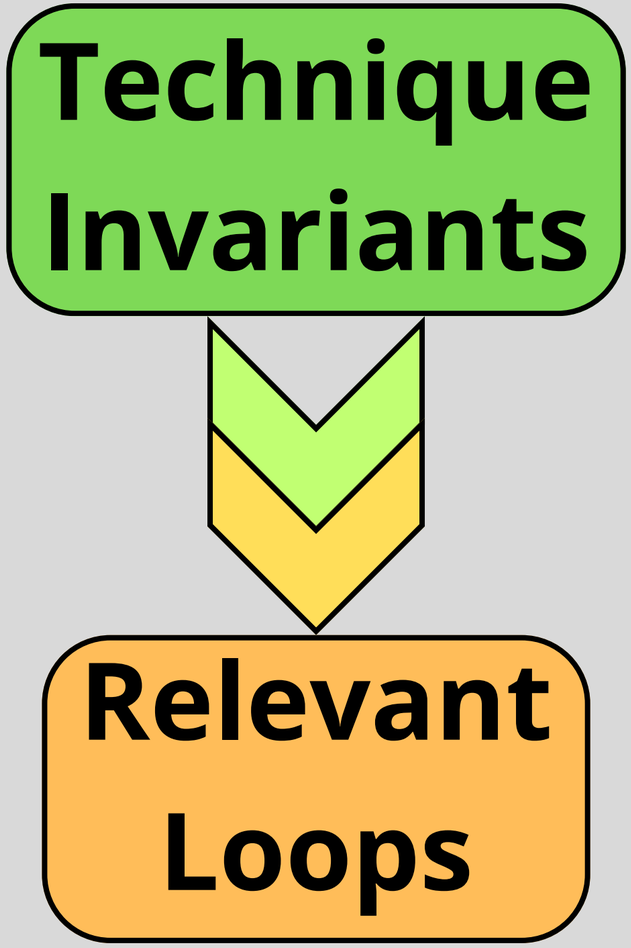
$C_0 = (0+a,0+b,0,i_1,0)$

Halting Conf.

$C_{a+b+2} = (0,0,2b,\#,a-b)$



Formal verification



$\Pi_{ADD}+(a,b)$

$$\varphi(k) \equiv C_k = (a - k, b, i_1, k), \text{ for } 0 \le k \le a$$

$$\varphi'(k) \equiv C_{a+1+k} = (0, b - k, i_2, a + k), \text{ for } 0 \le k \le b$$

$\Pi_{SUB}+(a,b)$

$$\varphi(k) \equiv C_k = (a-\lfloor k/2 \rfloor, b-\lceil k/2 \rceil, k, i_{1+mod(k,2)}, 0), \text{ for } 0 \le k \le \min\{2(a+1)-1, 2b)\}$$

$$\varphi'(k) \equiv C_{2b+k} = (a-b-k, 0, k, i_3, k), \text{ for } 0 \le k \le a-b$$

Example. Remainder

$\Pi_{Rem}+(a,b)$

Initial Conf.

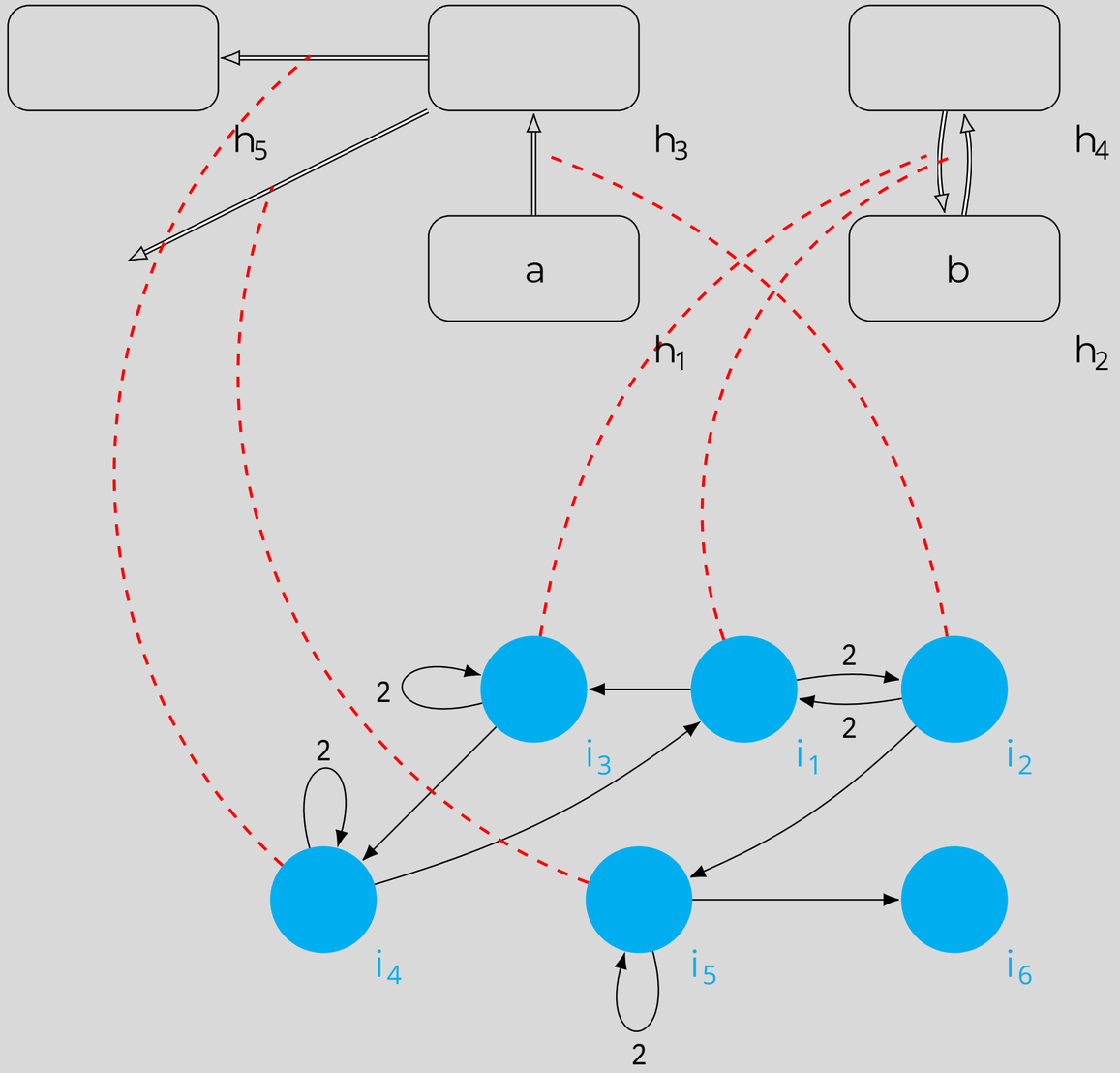
$C_0 = (0+a,0+b,0,0,0,i_1,0)$

Invariant

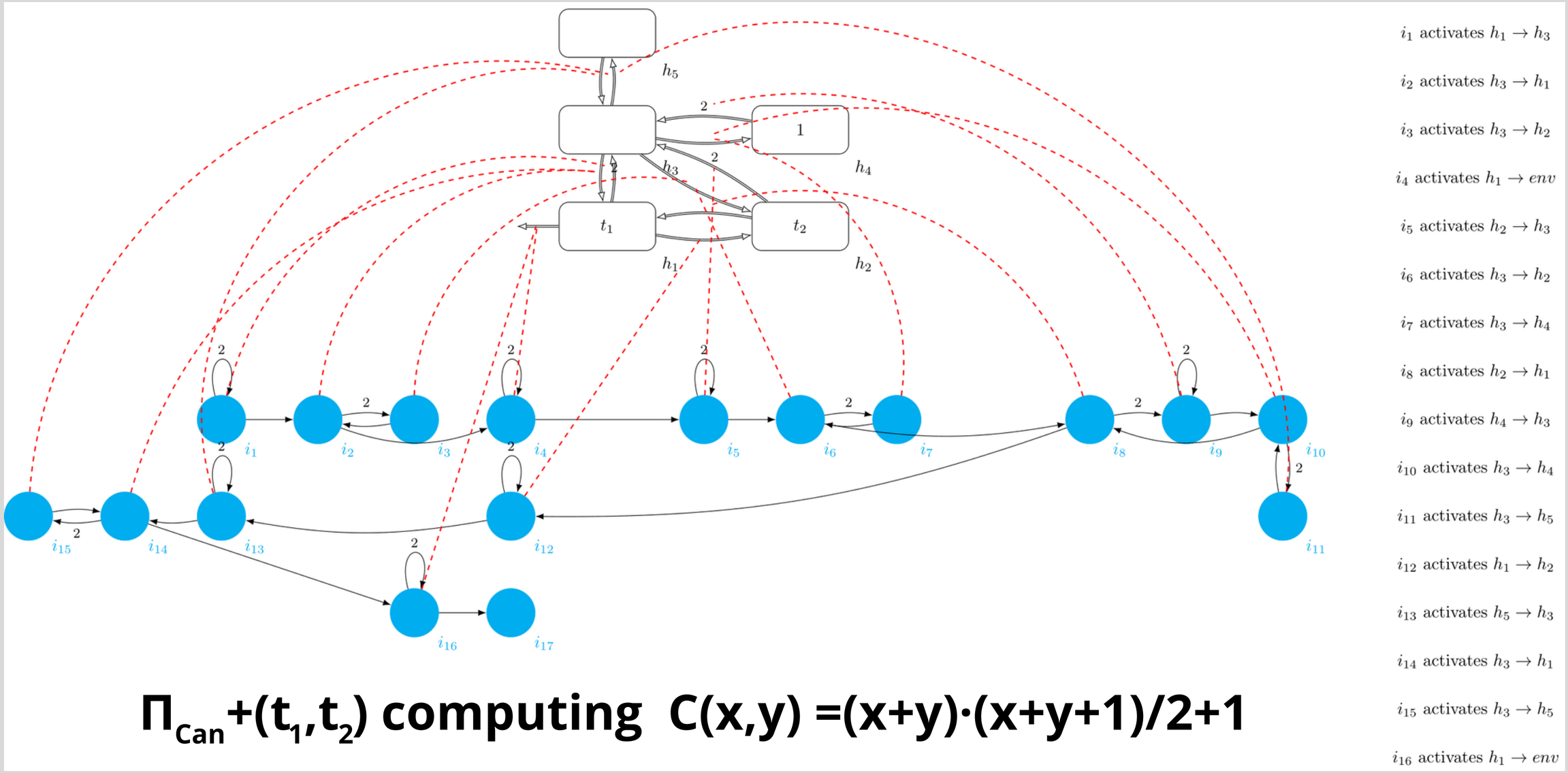
$\varphi(k) \equiv C_{k(4b+3)} = (a-b\cdot k,b,0,0,b\cdot k,i_1,0)$,
for $0 \le k \le \lfloor a/b \rfloor$

Halting Conf.

$C_{a+b+2} = (0,0,2b,\#,a-b)$



Example. Cantor Pairing Func.



$\Pi_{Can}+(t_1,t_2)$ computing $C(x,y) = (x+y)\cdot(x+y+1)/2+1$

1. **A. Ramírez-de-Arellano et al. Using VM to compute pairing functions, International Journal of Neural Systems 2023**

Example. Cantor Pairing Func.

Initial Conf.

$$C_0 = (0+t_1, 0+t_2, 0, 1, 0, i_1, 0)$$

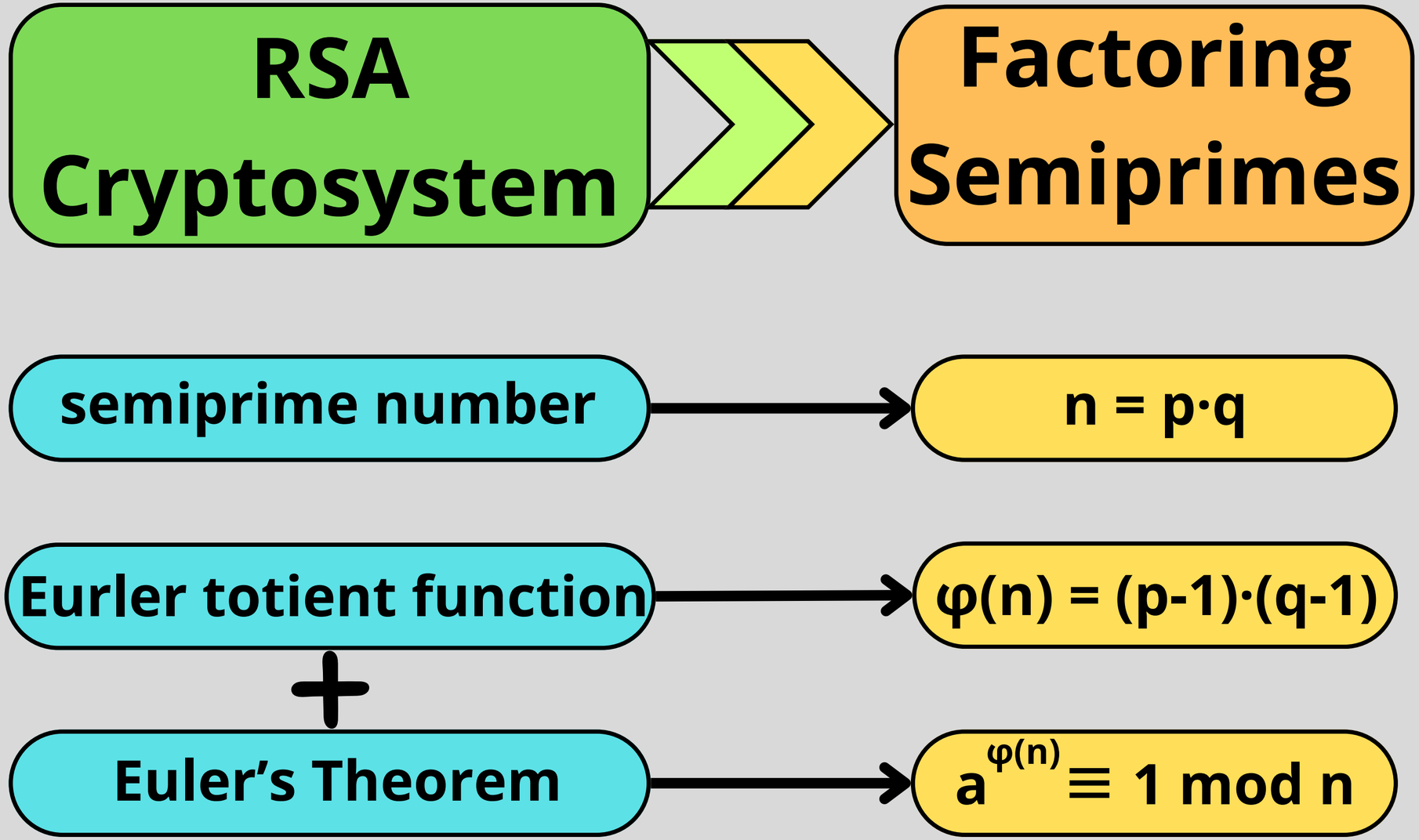
Invariant

$$\varphi(k) \equiv C_{y(k,t_1,t_2)} = (k,\ t_1+t_2-k, 0,\ t_1+t_2+1,\ k\cdot(t_1+t_2+1),\ i_8, t_1),\ \text{for } 0 \le k \le t_1+t_2$$

Halting Conf.

$$C_{y(t_1+t_2,t_1,t_2)+N} = (0, t_1+t_2, 0, t_1+t_2+1, \#, C(t_1,t_2))$$

Attacking Cryptosystmems

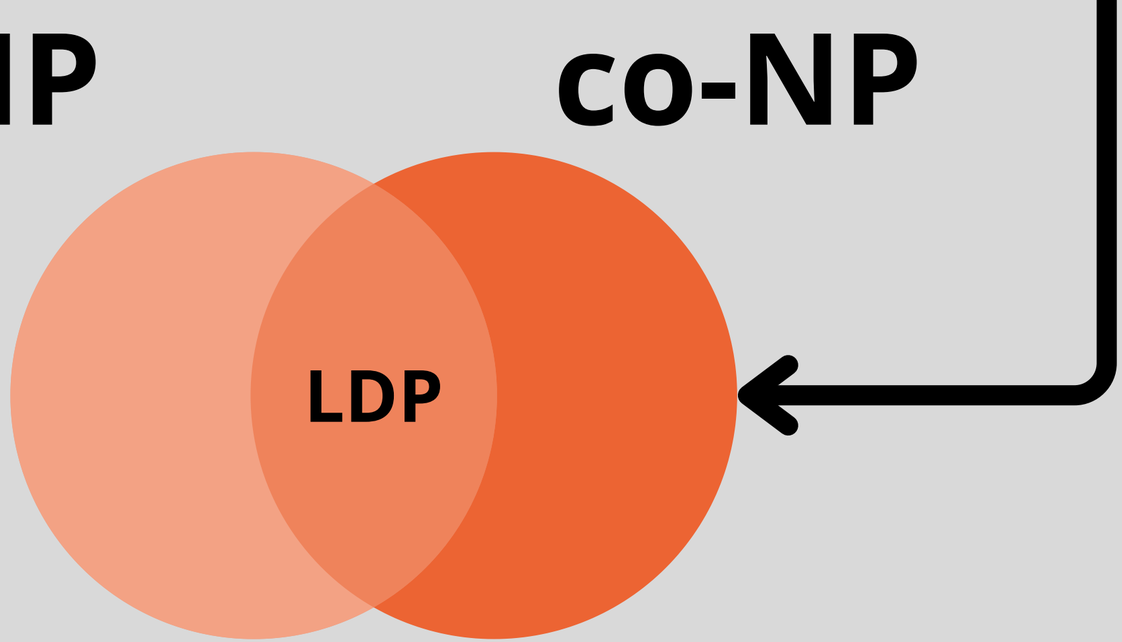


Attacking Cryptosystmems

Least Divisor Problem (LPD)
“Given a semiprime n>0,
find its least prime divisor”

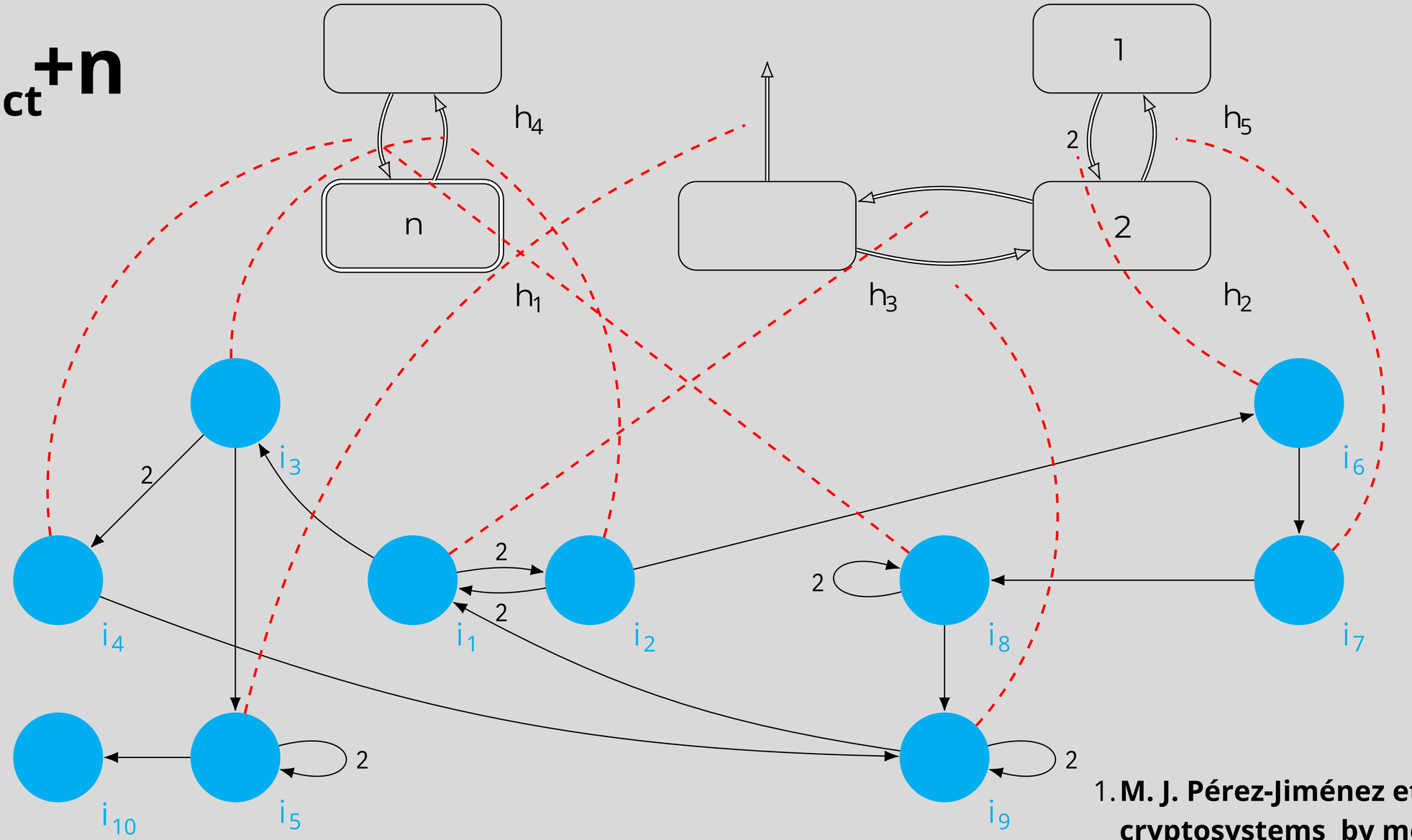
Decision version (LPD)
“given two natural numbers x, y,
determine whether or not
x has a factor less than y”

Generalization (LDP)
Fact(n) = p
where p is the least
prime divisor of n



VM Computing Fact[1]

$\Pi_{Fact}+n$



1. **M. J. Pérez-Jiménez et al. Attacking cryptosystems by means of VM, Scientific Reports 2023**

VM Computing Fact

Initial Conf.

$$C_0 = (0+n,2,0,0,1,i_1,0)$$

Invariant

$$\varphi(k) \equiv C_{y(k)} = (n,\ k+1,0,0,1,i_1,0),\ \text{for } 0 \le k \le p-1,\ \text{where } n = p \cdot q$$

Halting Conf.

$$C_{y(p-1)+N} = (0,0,0,n,1,\#,p)$$

4. Conclusion

Conclusion

Virus Machines

- Universality
- Formal verification
- Practical examples
  - Pairing functions
  - Cryptosystems
  - Decision Problems

Open Problems

Efficiency

Parallelism

NP-Hard Problems

Parallel VM[1] extensions/variants

Real-World Appl.

Nonlinear Behavior
Dynamic Threshold

1. A. Ramírez-de-Arellano et al. Parallel VM, (A)CMC2023 Chengdu,China

Thank you!

aramirezdearellano@us.es

cs.us.es/perfiles/antonio-ramirez-de-arellano-marrero