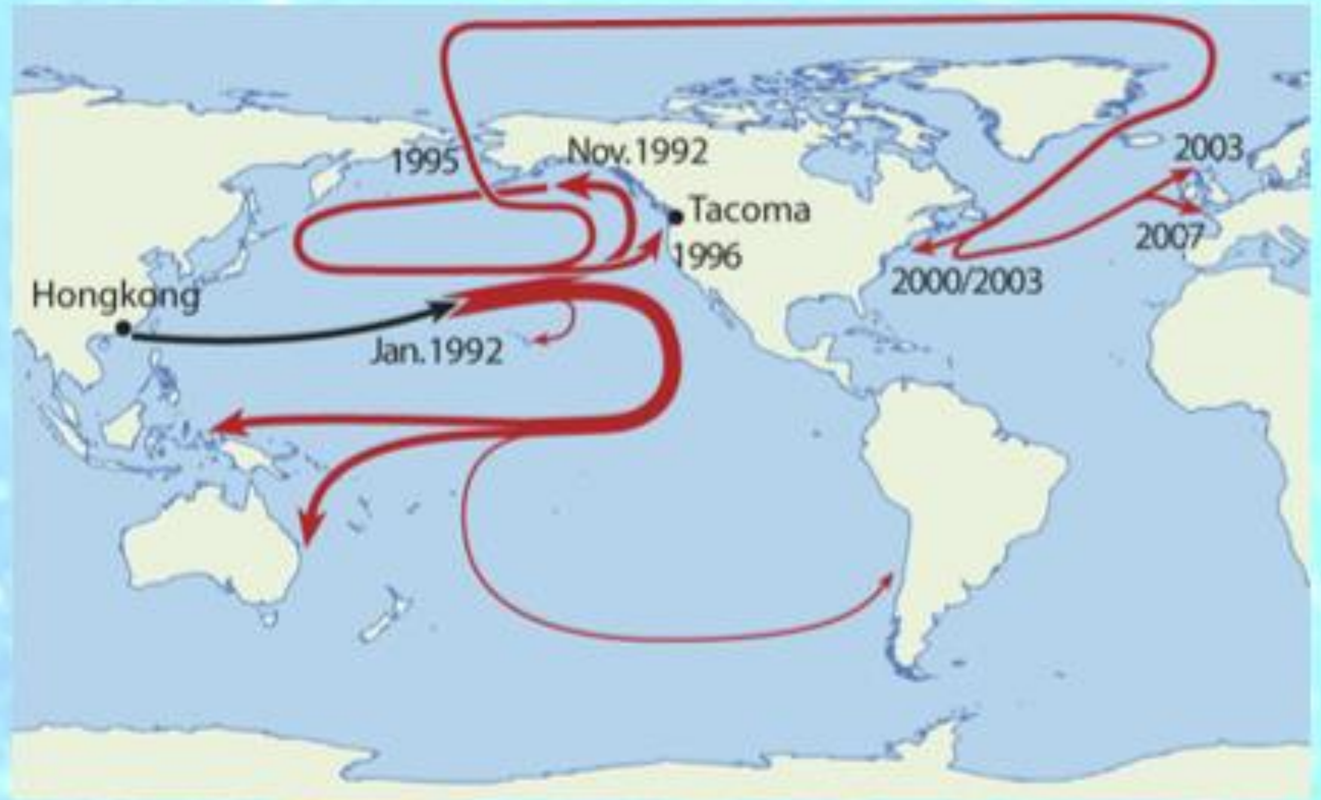
Towards a "Fluid computer"

Eva Miranda (UPC-CRM)

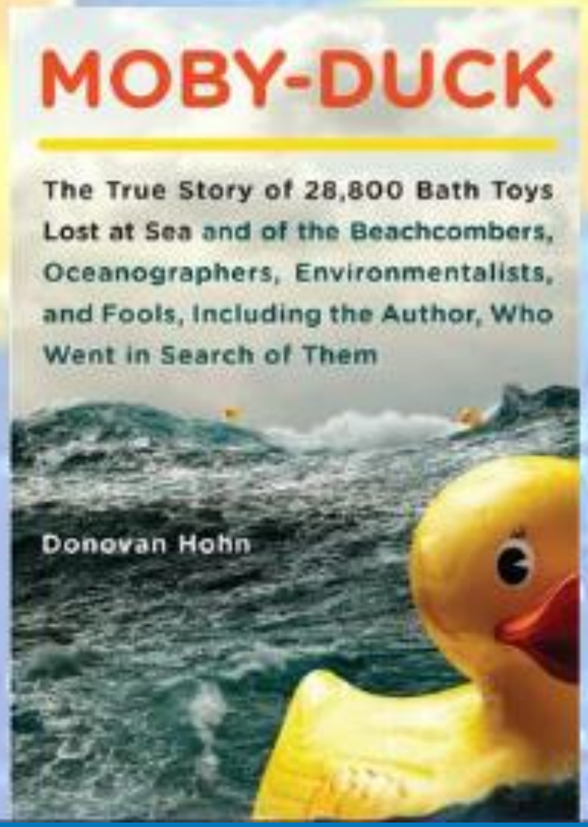
FoCM 2023

June 2023

The trip of the friendly floatees



- **January 10, 1992:** The carrier Ever Laurel departed from Hong-Kong with destination Tacoma. The carrier lost the cargo during a storm including 29000 rubber ducks.
- **November 16, 1992:** 10 Rubber ducks appeared in Sitka, Alaska.
- **July 2007:** One rubber duck show ups in Scotland. Though many more were expected.




HOMES AND PROPERTY › HOME PAGE

Thousands of rubber ducks to land on British shores after 15 year journey

VIEW COMMENTS

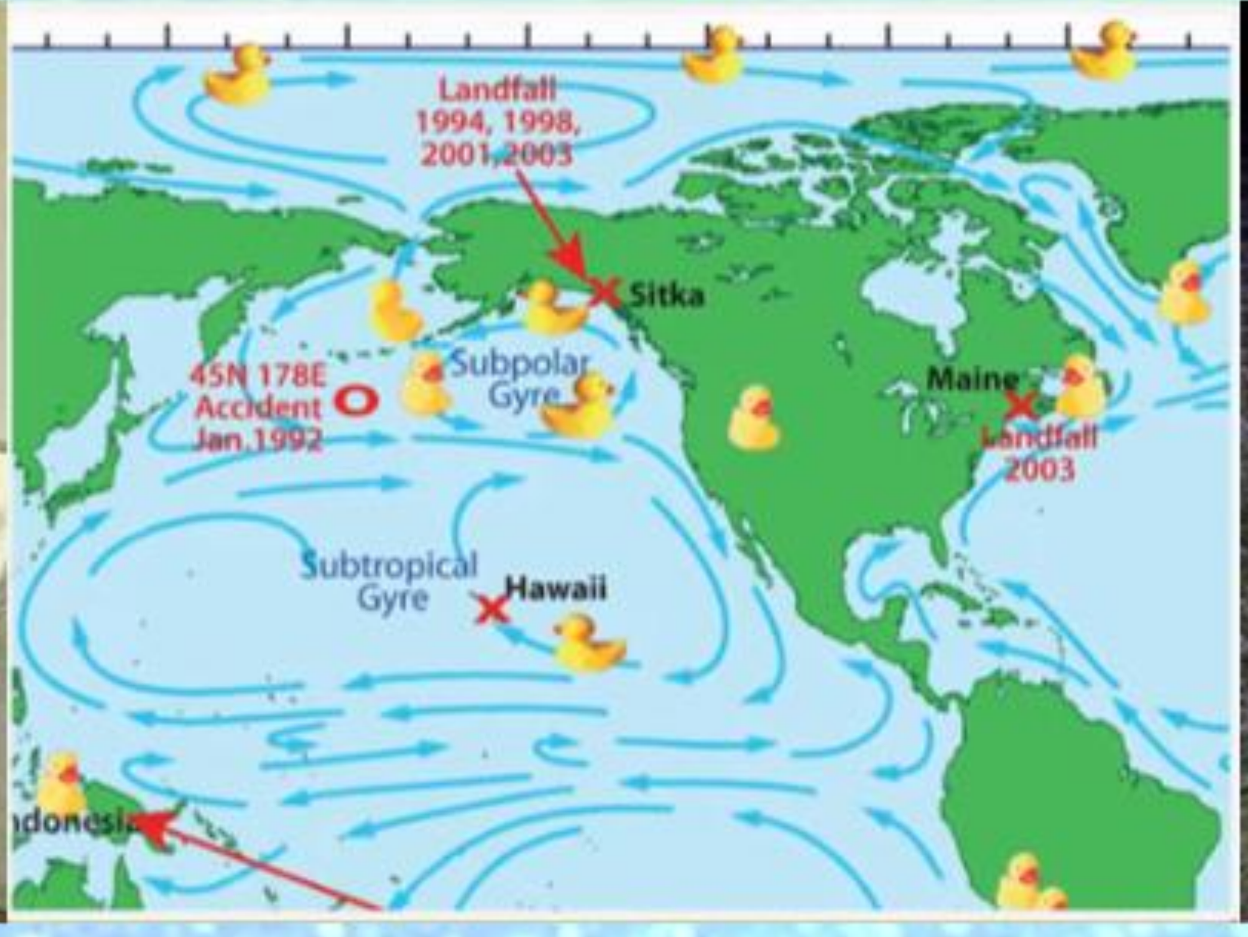
27 June 2007

They were toys destined only to bob up and down in nothing bigger than a child's bath - but so far they have floated halfway around the world.

The armada of 29,000 plastic yellow ducks, blue turtles and green frogs

What did we learn from the 29000 ducks?

- Curtis Ebbesmeyer studies ocean currents by tracking the movement of drifting things-from icebergs to message bottles.

- Using OSCURS (Ocean Surface Currents Simulation), a computer simulator developed by Seattle oceanographer Jim Ingraham, Ebbesmeyer tracked the oceanic movement of all kinds of flotsam.
- Thanks to the friendly floatees predictions about currents could be made.

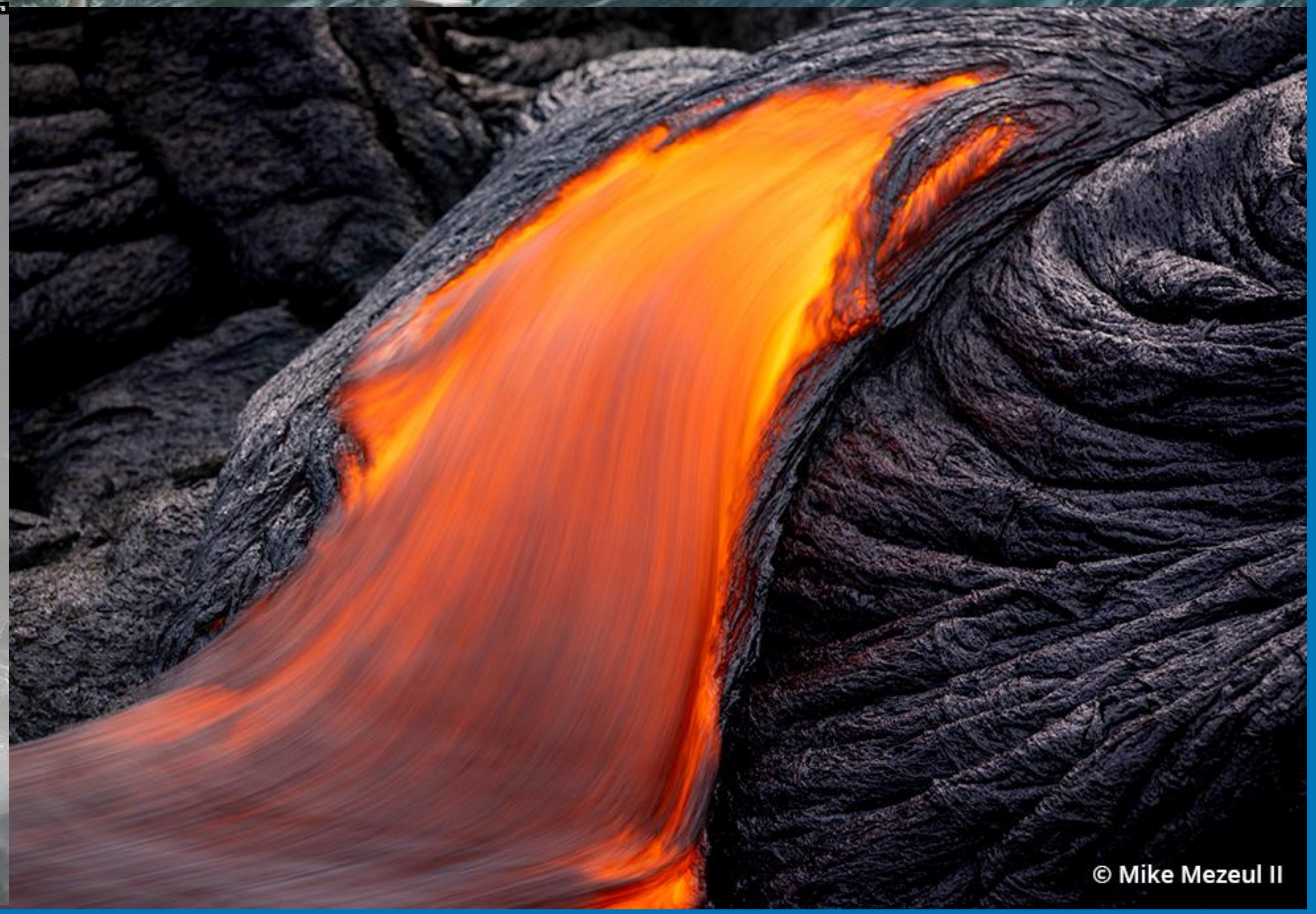

- Only 2% of the messages on a bottle are recovered.

Computational complexity and fluid dynamics

In nature, fluids (like water or lava) often rebel against…what it is expected

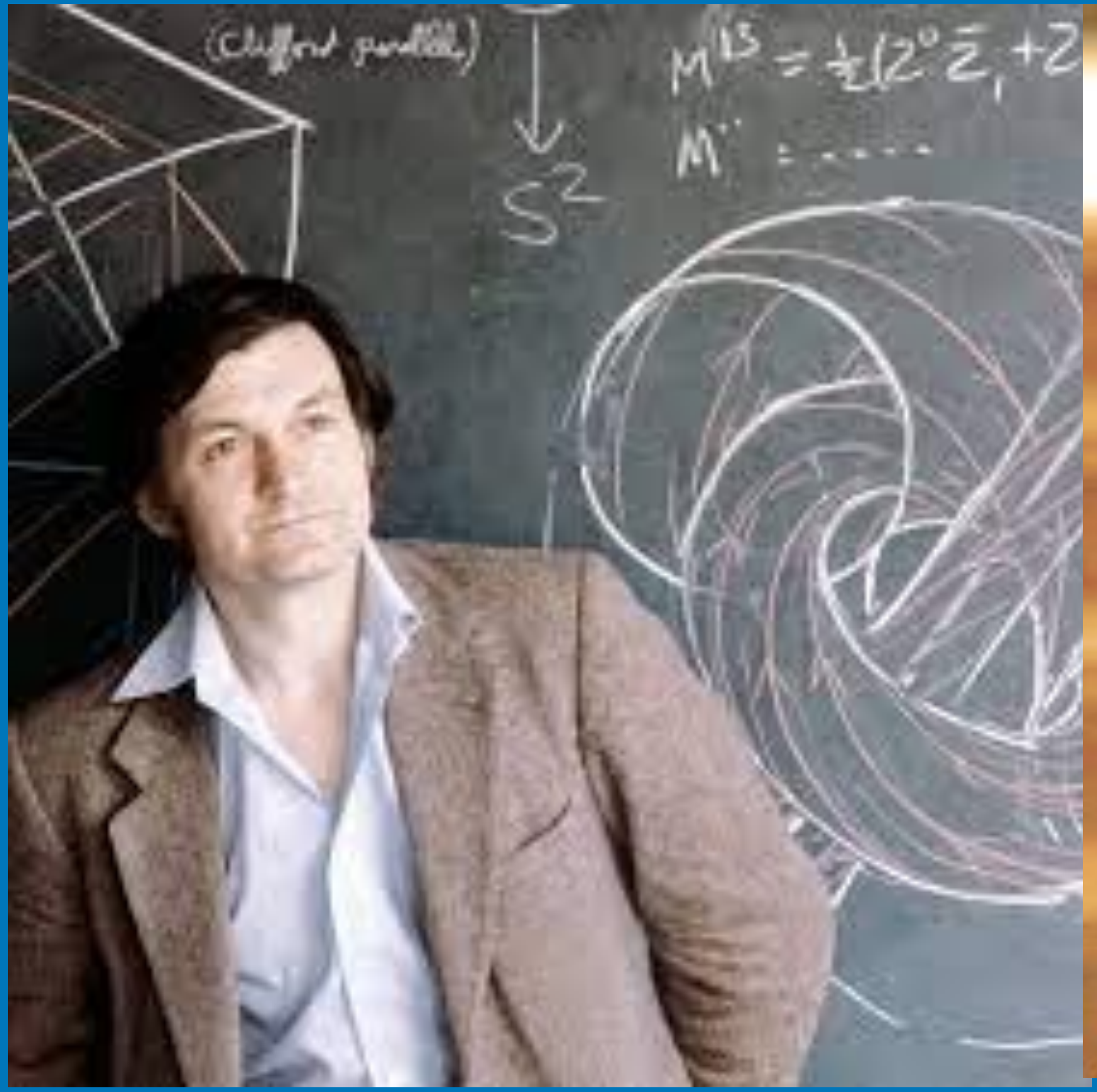


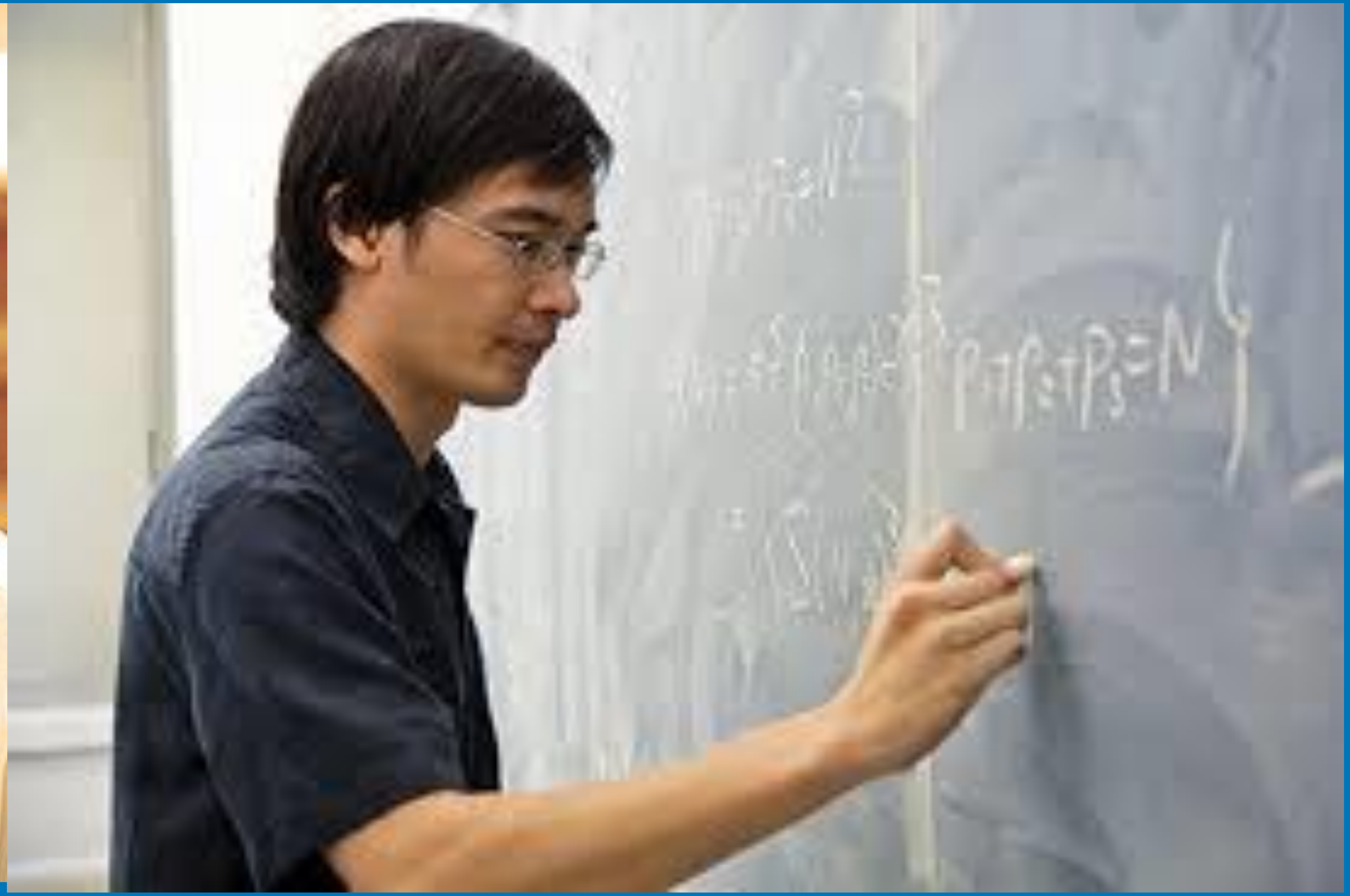


Fluid computers?

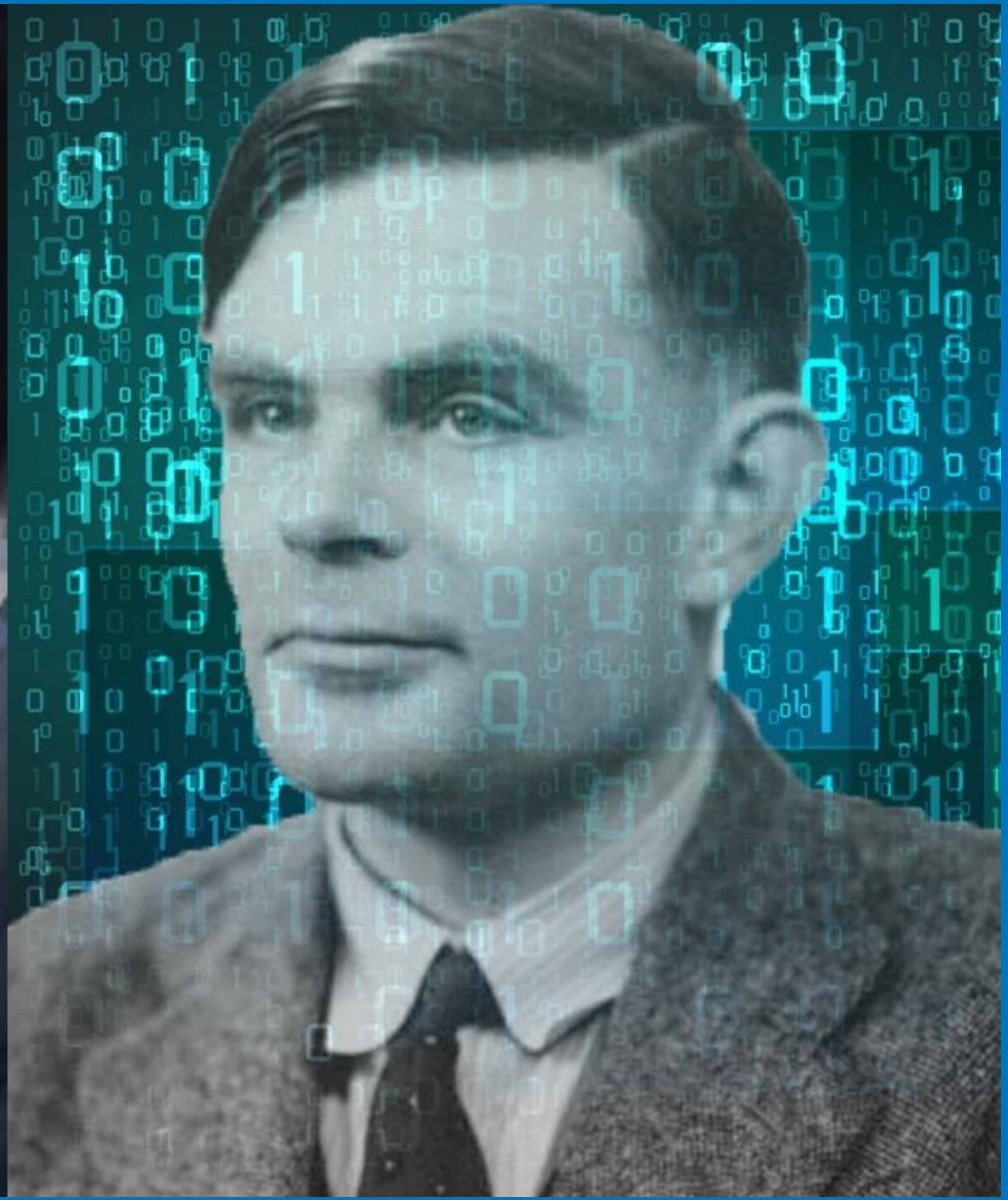
Are fluids "complicated enough" to perform computations?



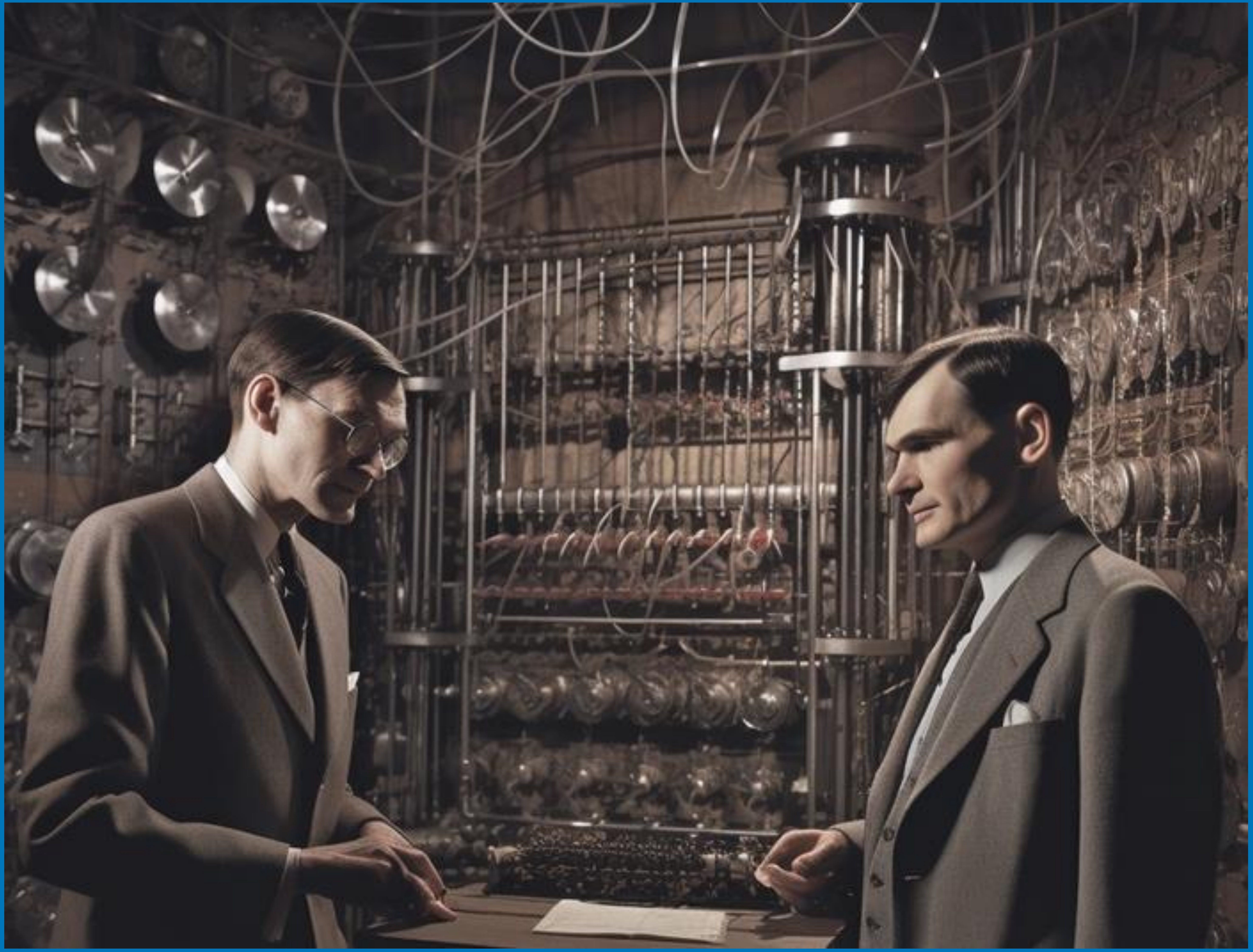


Levels of complexity and Alan Turing

Can fluids simulate any Turing machine?

Ask the experts!



Turing machines and the halting problem

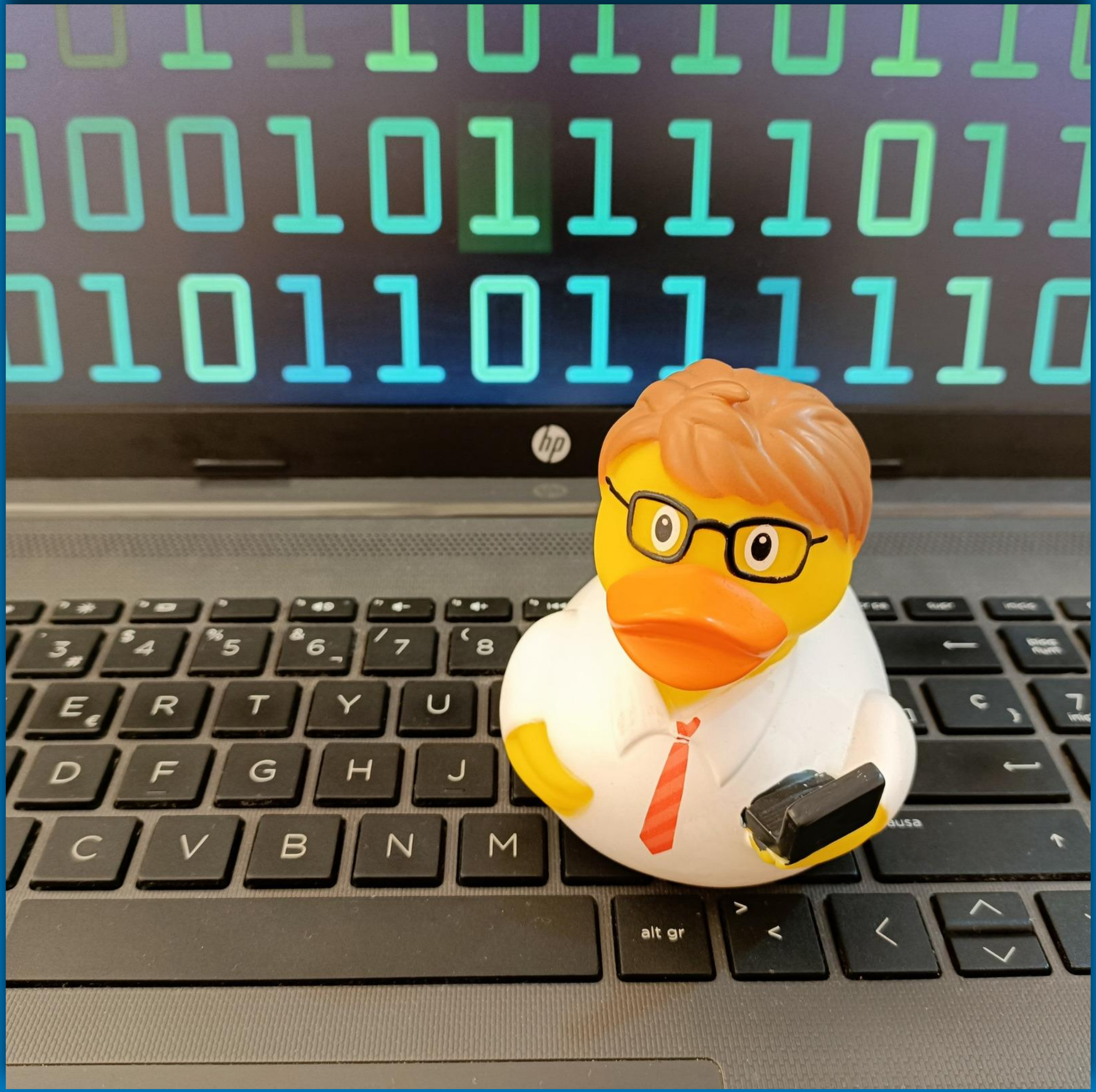
In computability theory, the halting problem is the problem of determining, from a description of an arbitrary computer program and an input, whether the program will finish running (halting state), or continue to run forever.

Turing, 1936: The halting problem is **undecidable**.

Alan Turing proved in 1936 that a general algorithm to solve the halting problem for all possible program-input pairs cannot exist.

What does Turing have to do with the rubber ducks?

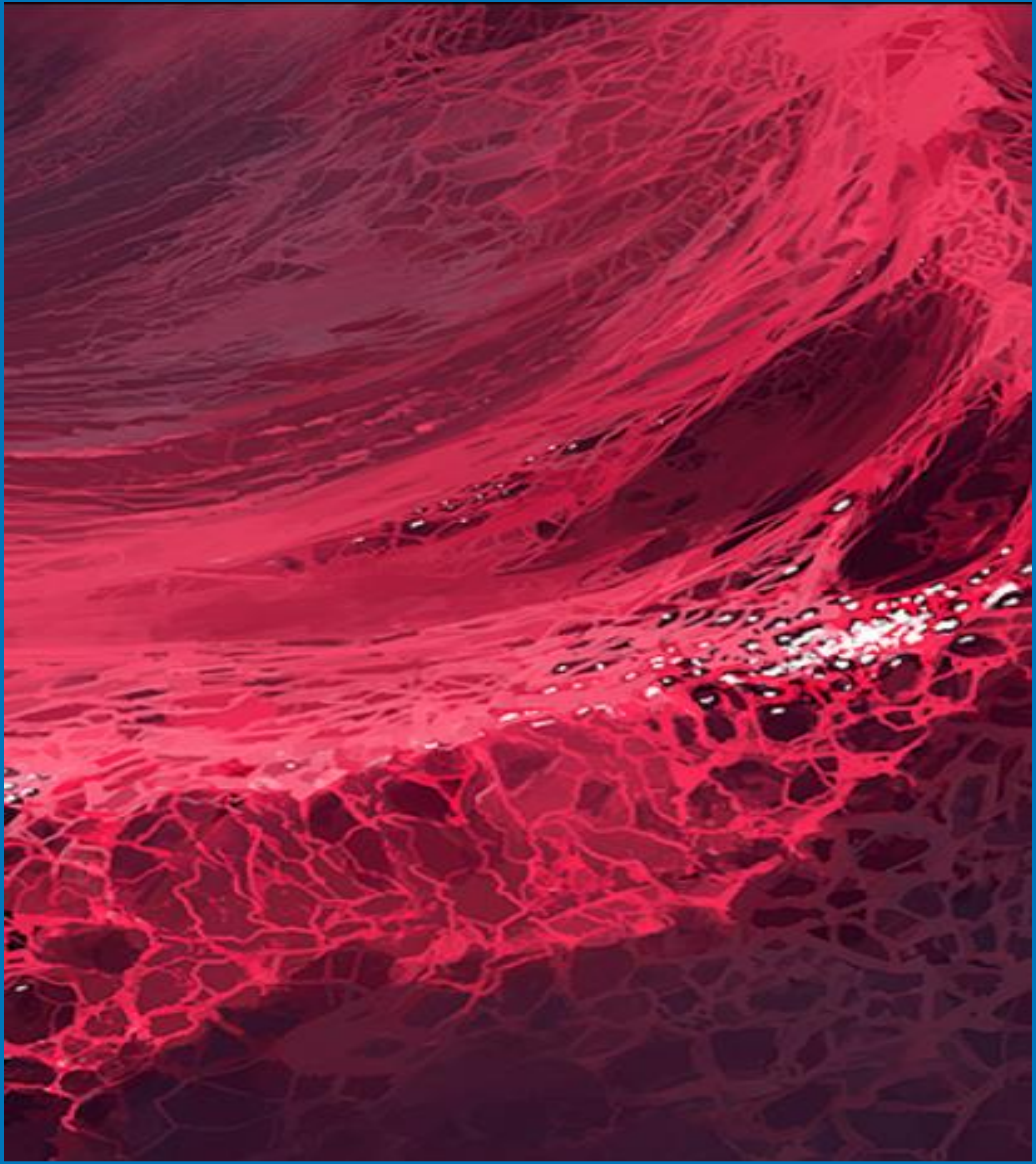
- The method **OSCURS** used by Ingraham and Ebbesmeyer could not localise all the lost rubber ducks.
- Only a 2% of the messages in bottles are recovered.
- What if finding the rubber ducks is an undecidable problem?
- Can we associate a Turing machine or supercomputer to the trajectories of the rubber ducks?



Reality or science fiction?

The novel Solaris written by Stanisław Lem (1961), presents a thinking sea:

[...] *``For some time there was a widely held notion (zealously fostered by the daily press) to the effect that the 'thinking ocean' of Solaris was a gigantic brain, prodigiously well-developed and several million years in advance of our own civilization, a sort of 'cosmic yogi', a sage, a symbol of omniscience, which had long ago understood the vanity of all action and for this reason had retreated into an unbreakable silence."*

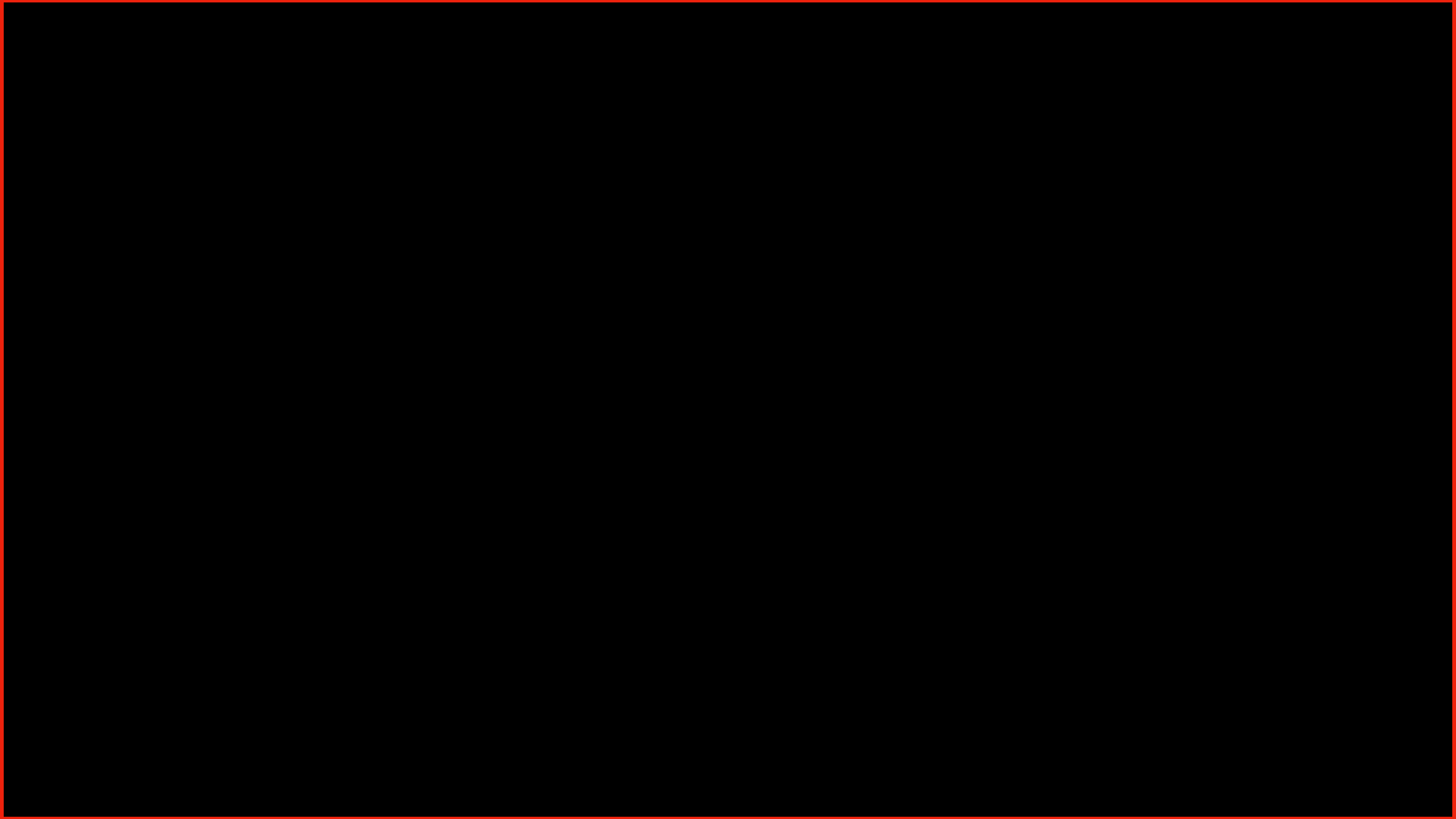


Let’s talk about Chaos!

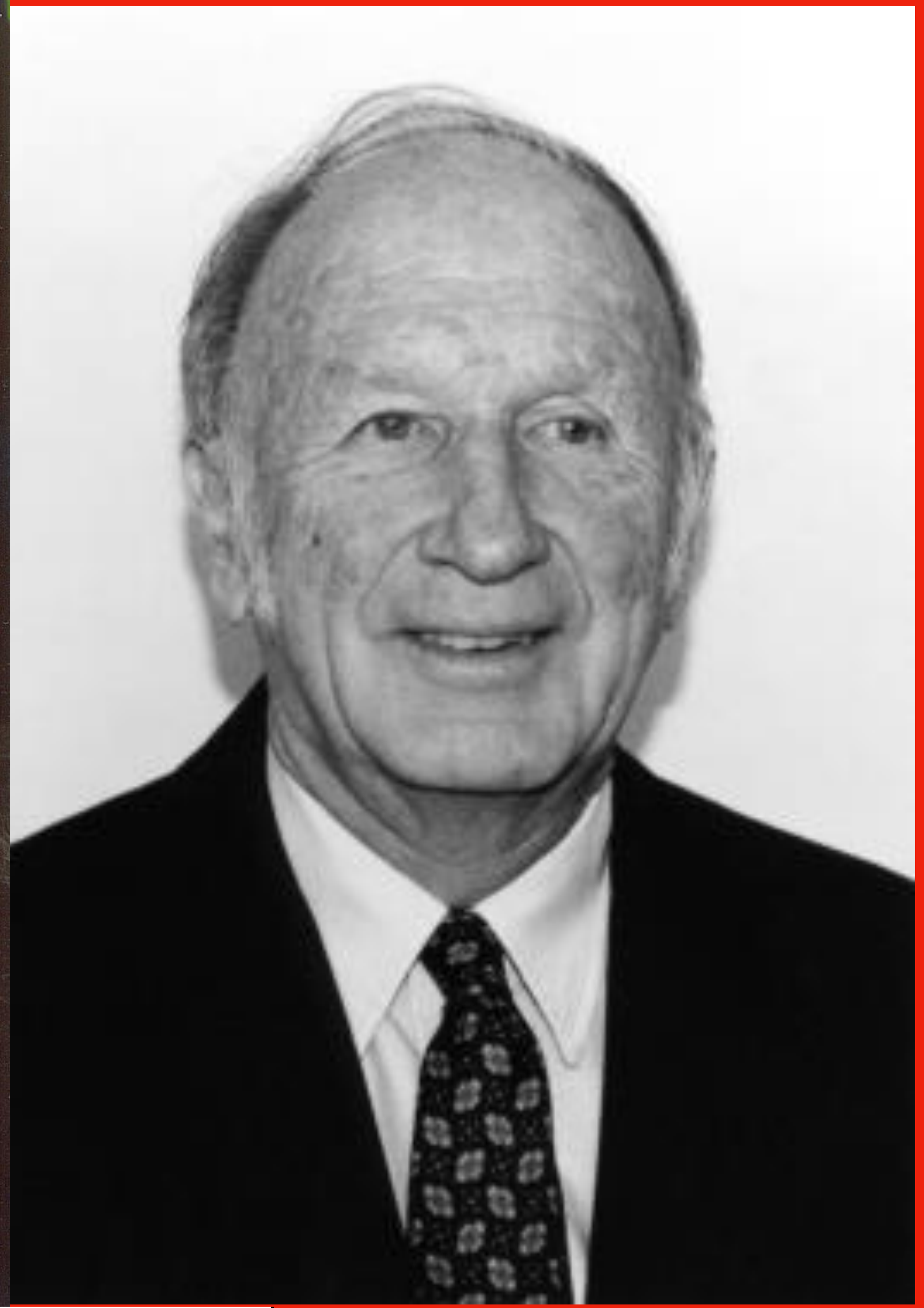
“Chaos: When the present determines the future, but the approximate present does not approximately determine the future”.

Edward Lorenz

An example of the Butterfly effect….



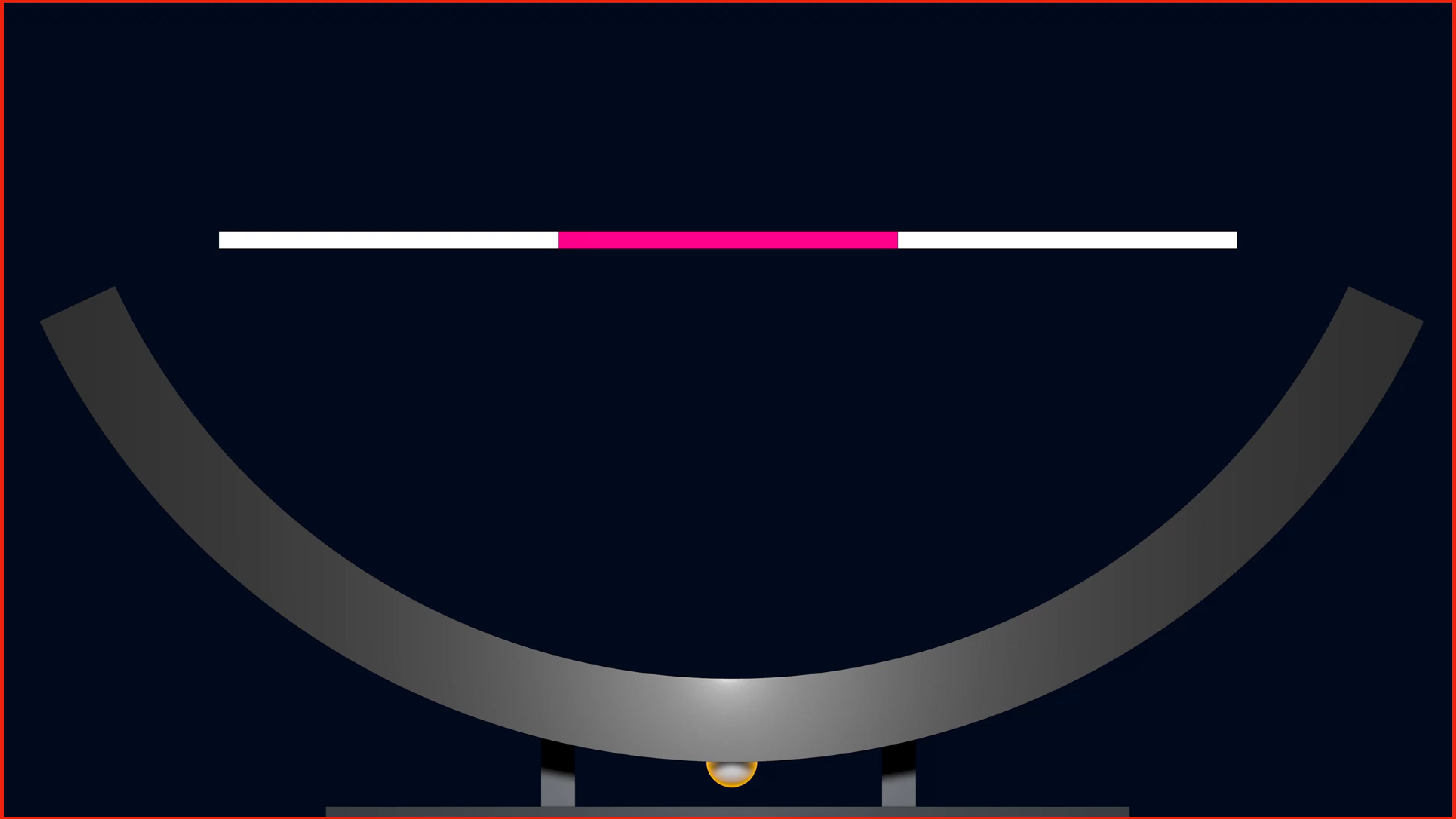
Fetter, Hamilton and Lorenz



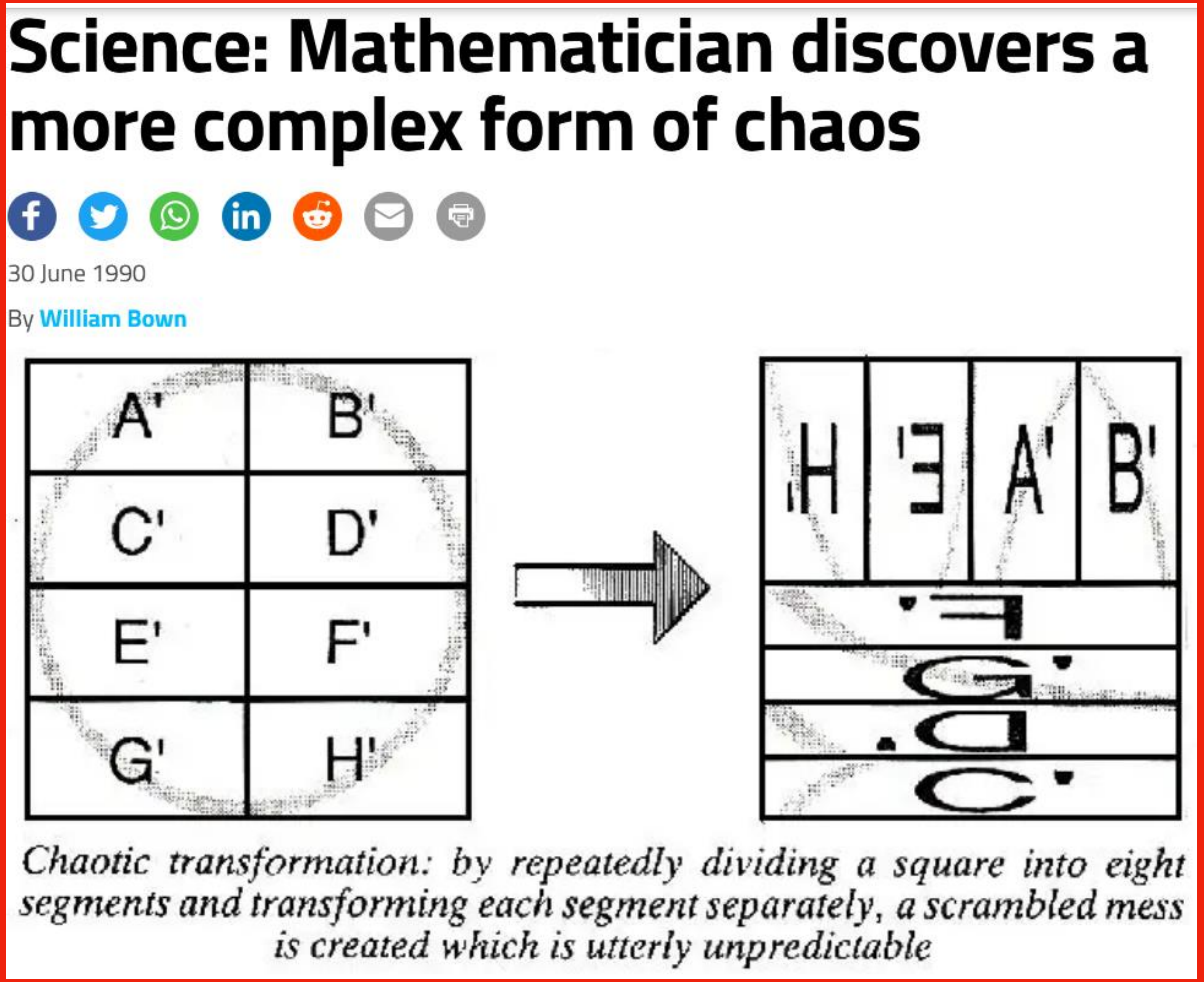
The system is chaotic for ρ= 28 but exhibits periodic orbits for other values of ρ.

$$\frac{\mathrm{d}x}{\mathrm{d}t} = \sigma y - \sigma x,$$
$$\frac{\mathrm{d}y}{\mathrm{d}t} = \rho x - xz - y,$$
$$\frac{\mathrm{d}z}{\mathrm{d}t} = xy - \beta z.$$

The Cantor set



Moore, a new form of chaos

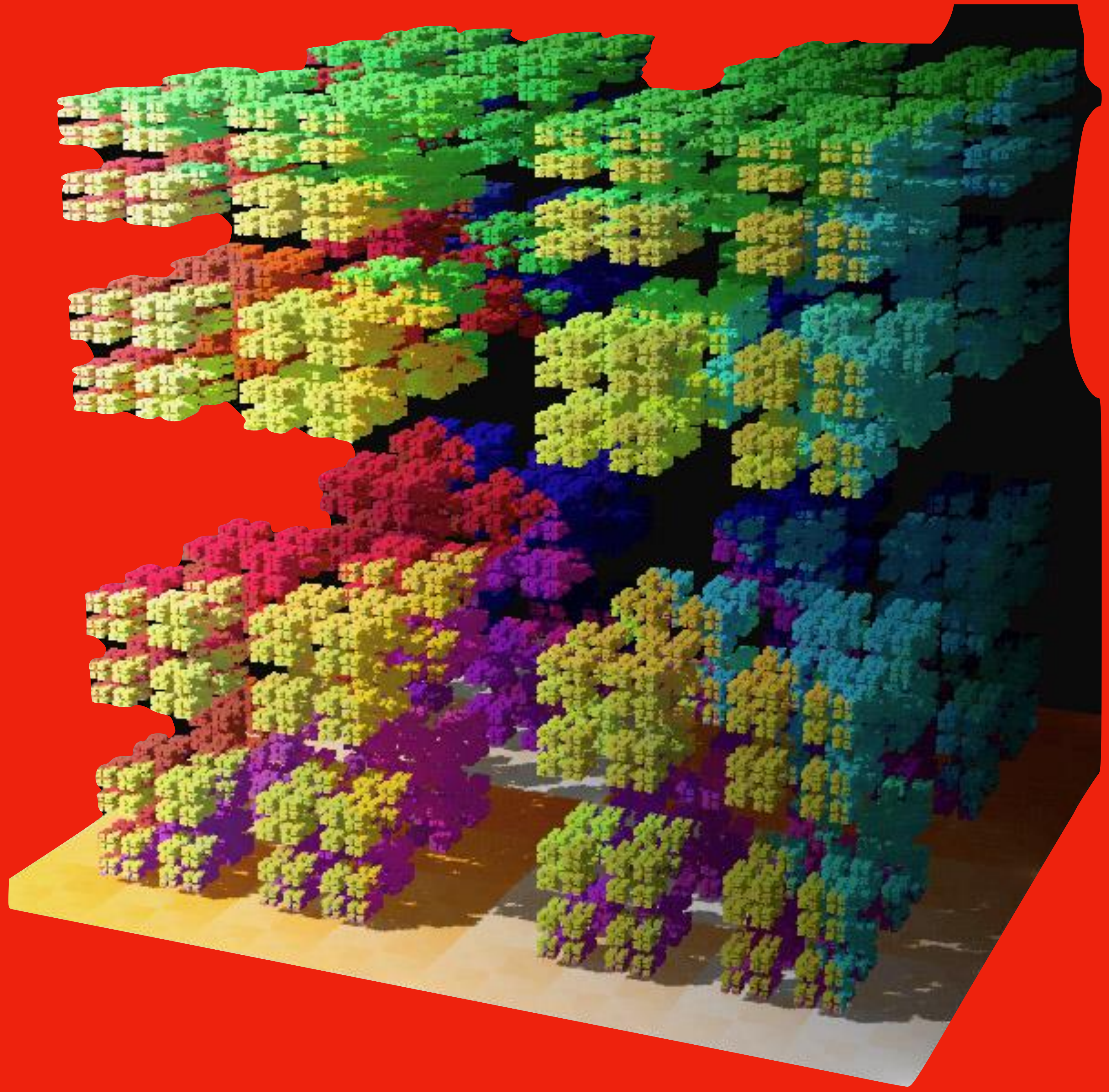


Science: Mathematician discovers a more complex form of chaos

30 June 1990

By William Bown

Chaotic transformation: by repeatedly dividing a square into eight segments and transforming each segment separately, a scrambled mess is created which is utterly unpredictable



Moore generalized the notion of shift in dynamical systems and was able to simulate any Turing machine (generalized shifts). They are conjugated to maps of the square Cantor set.

A symbolic dynamics tool: generalized shifts

Generalized shifts

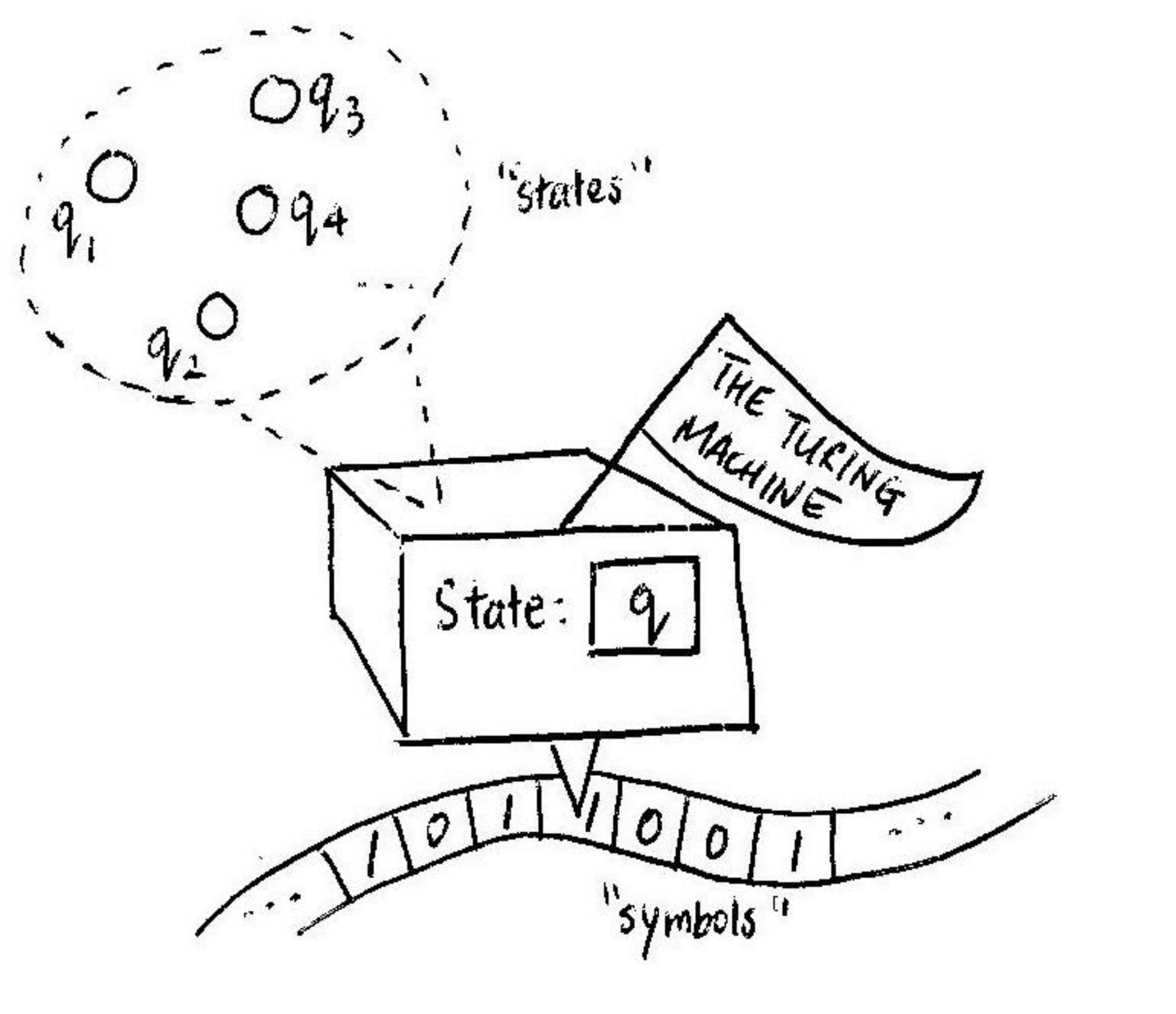
Let A be an alphabet. A generalized shift $\phi : A^{\mathbb{Z}} \to A^{\mathbb{Z}}$ is specified by two maps F and G. Denote by $D_F = \{i, ..., i+r-1\}$ and $D_G = \{j, ..., j+\ell-1\}$ the sets of positions on which F and G depend. The function G modifies the sequence only at the positions indicated by D_G:

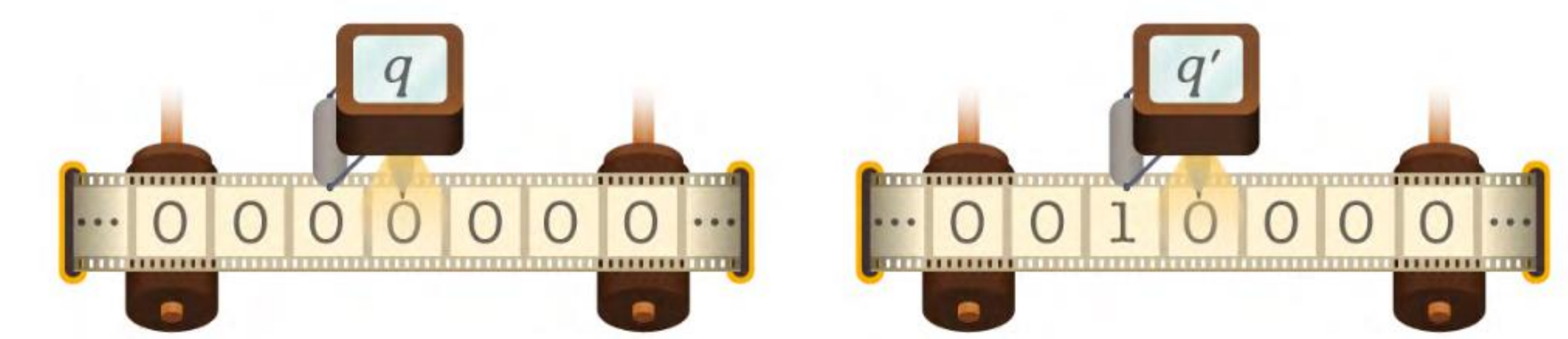
$$G : A^{\ell} \longrightarrow A^{\ell}$$
$$(s_j ... s_{j+\ell-1}) \longmapsto (s'_j ... s'_{j+\ell-1})$$

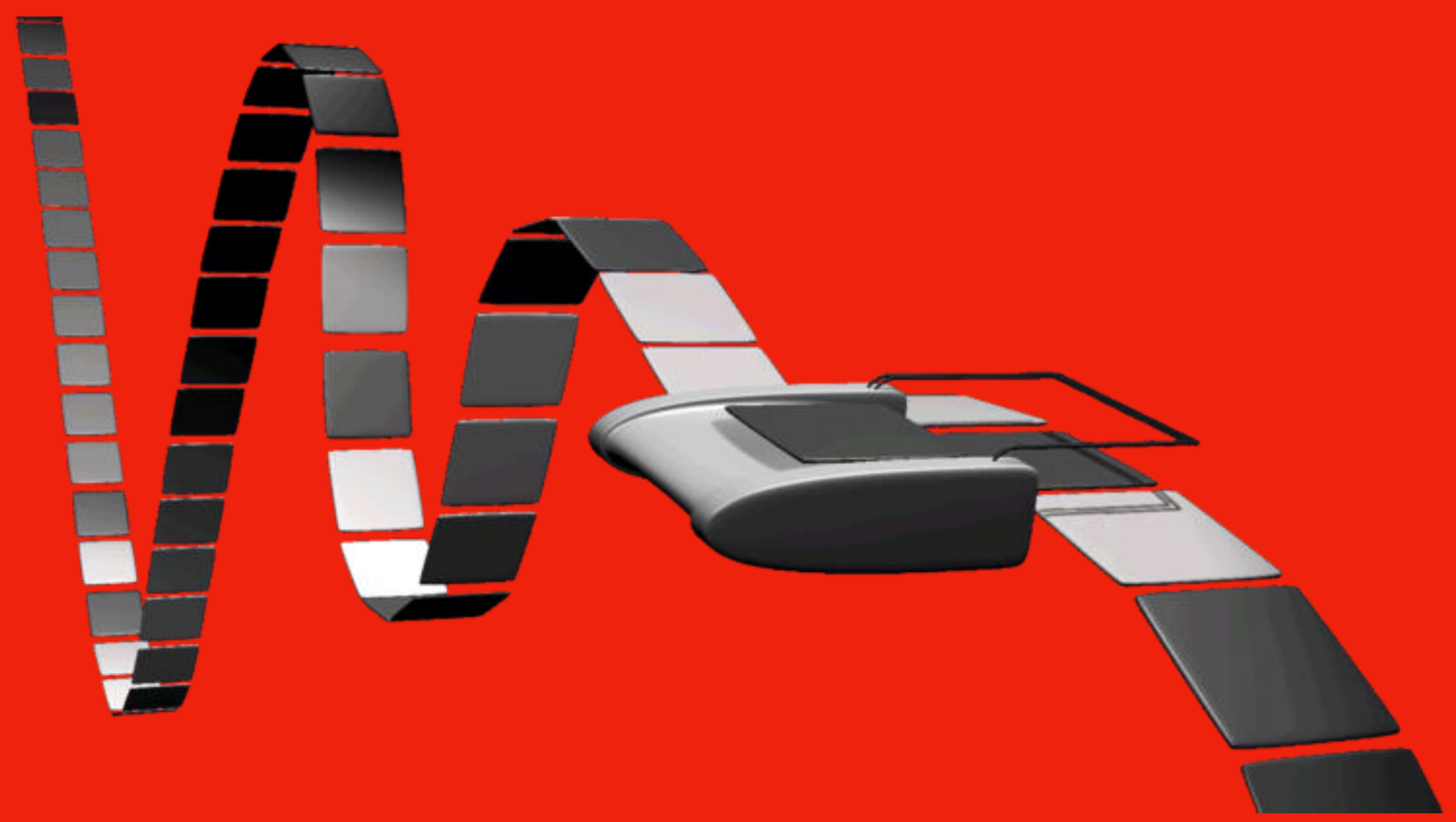
On the other hand, the function F assigns to the finite subsequence $(s_i, ..., s_{i+r-1})$ of the infinite sequence $S \in A^{\mathbb{Z}}$ an integer $F : A^r \longrightarrow \mathbb{Z}$. ϕ is defined as:

- Compute $F(S)$ and $G(S)$.
- Modify S changing the positions in D_G by the function $G(S)$, obtaining a new sequence S'.
- Shift S' by $F(S)$ positions.

Turing machines…





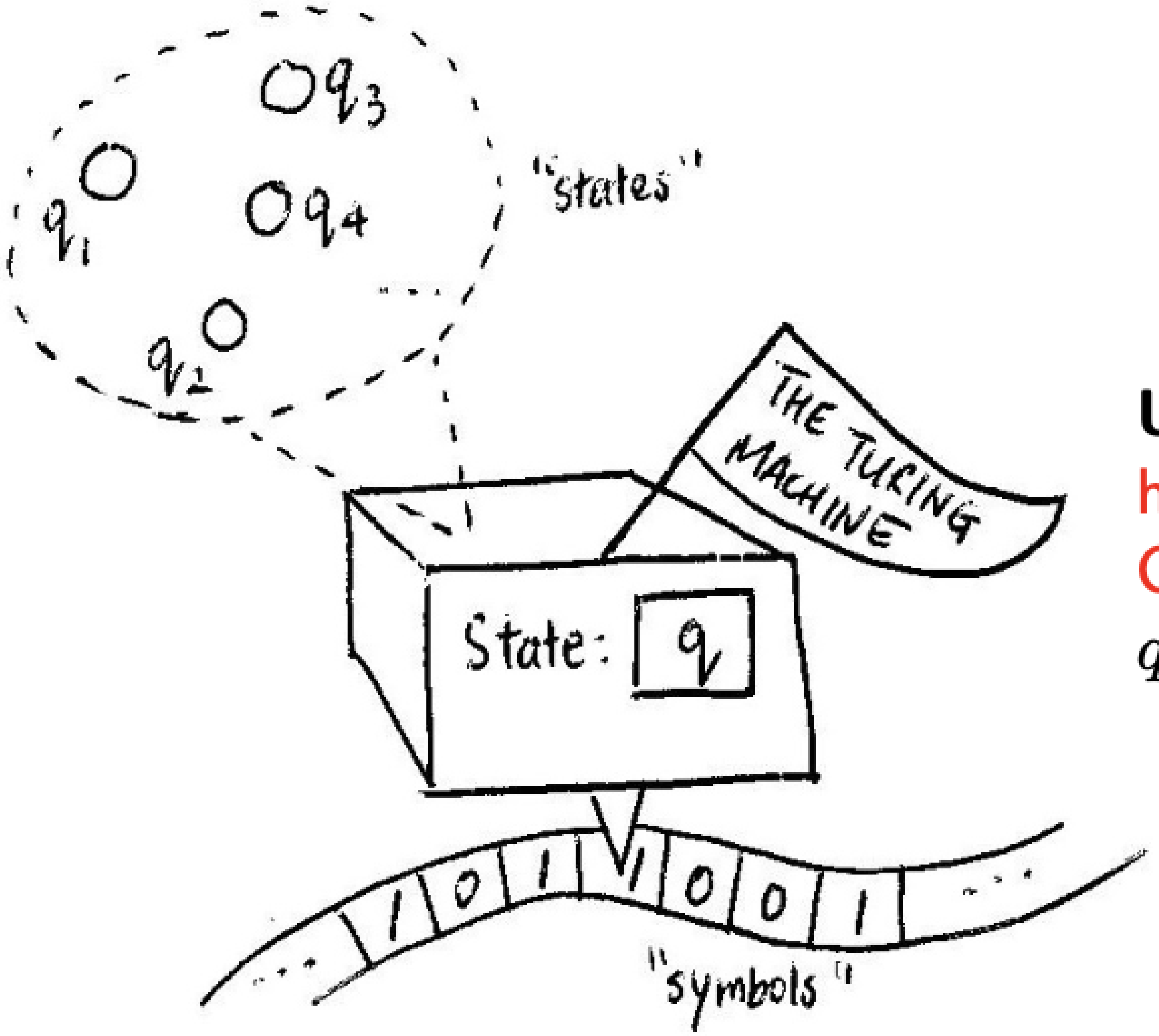


A Turing machine is a "printer" of states on a long tape. When it reaches the "halting" state, the machine stops.

Turing machines

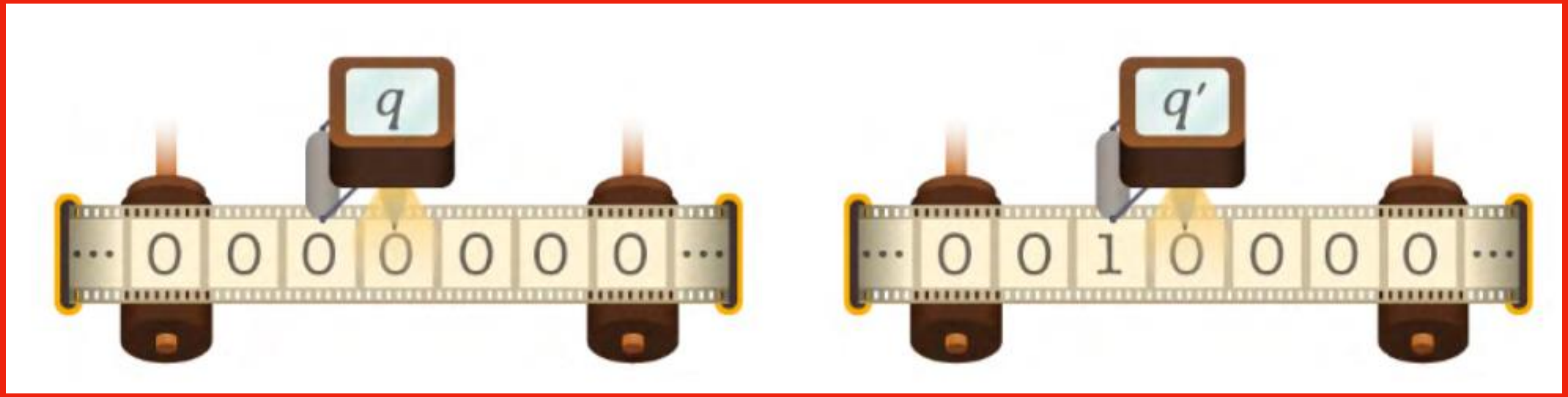
Turing machine

A Turing machine is defined as $T = (Q, q_0, q_{halt}, \Sigma, \delta)$, where Q is a finite set of states, including an initial state q_0 and a halting state q_{halt}, Σ is the alphabet, and $\delta : (Q \times \Sigma) \longrightarrow (Q \times \Sigma \times \{-1, 0, 1\})$ is the transition function. The input of a Turing machine is the current state $q \in Q$ and the current tape $t = (t_n)_{n\in\mathbb{Z}} \in \Sigma^{\mathbb{Z}}$.

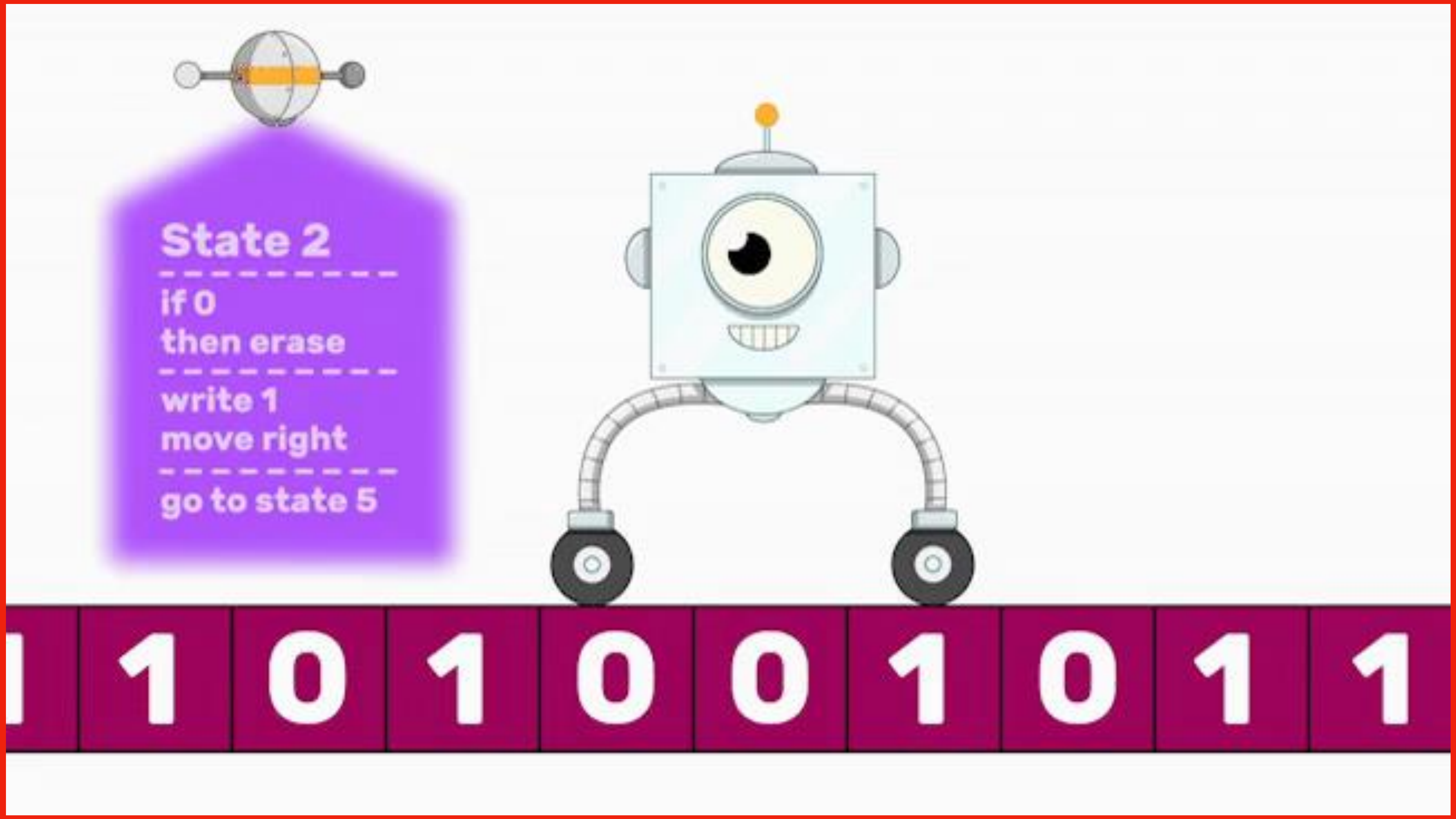


User's guide: If the current state is q_{halt} then halt the algorithm and return t as output. Otherwise compute $\delta(q, t_0) = (q', t'_0, \varepsilon)$, replace q with q', t_0 with t'_0 and t by the ε-shifted tape.

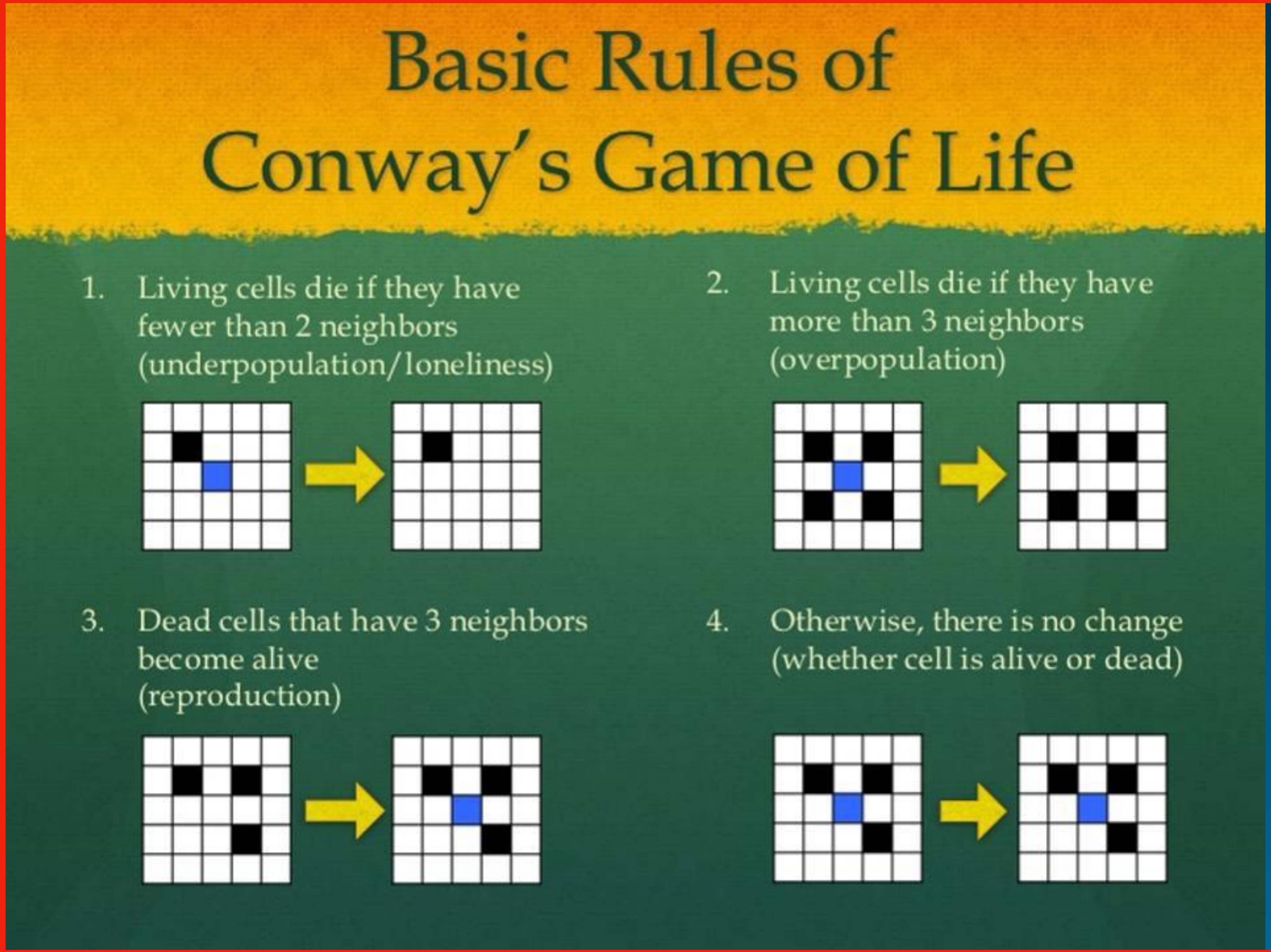
Turing machines…

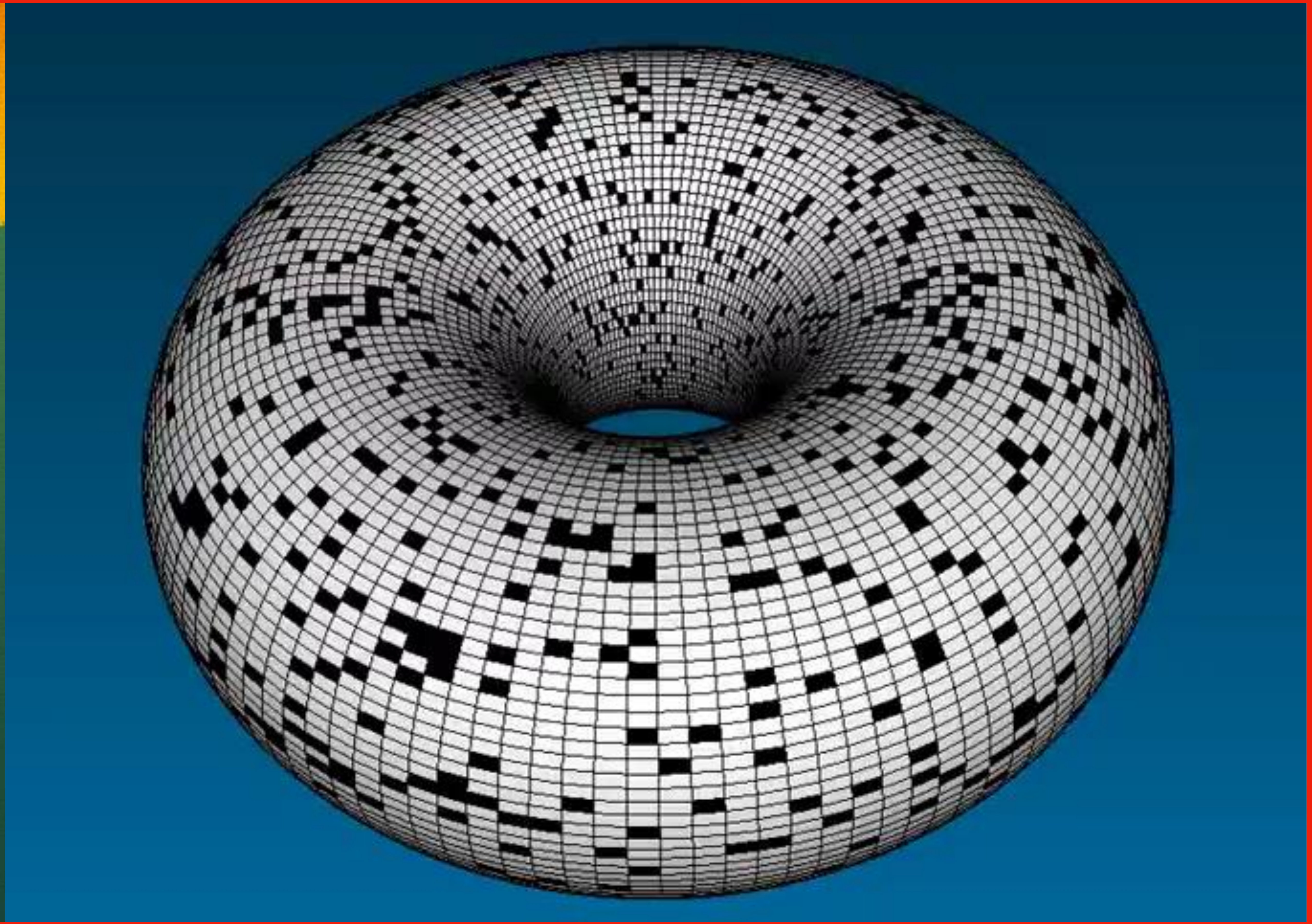


If δ(q, 0) = (q′, 1, +1), we replace 0 by 1, the new state is q’ and we shift the tape to the left



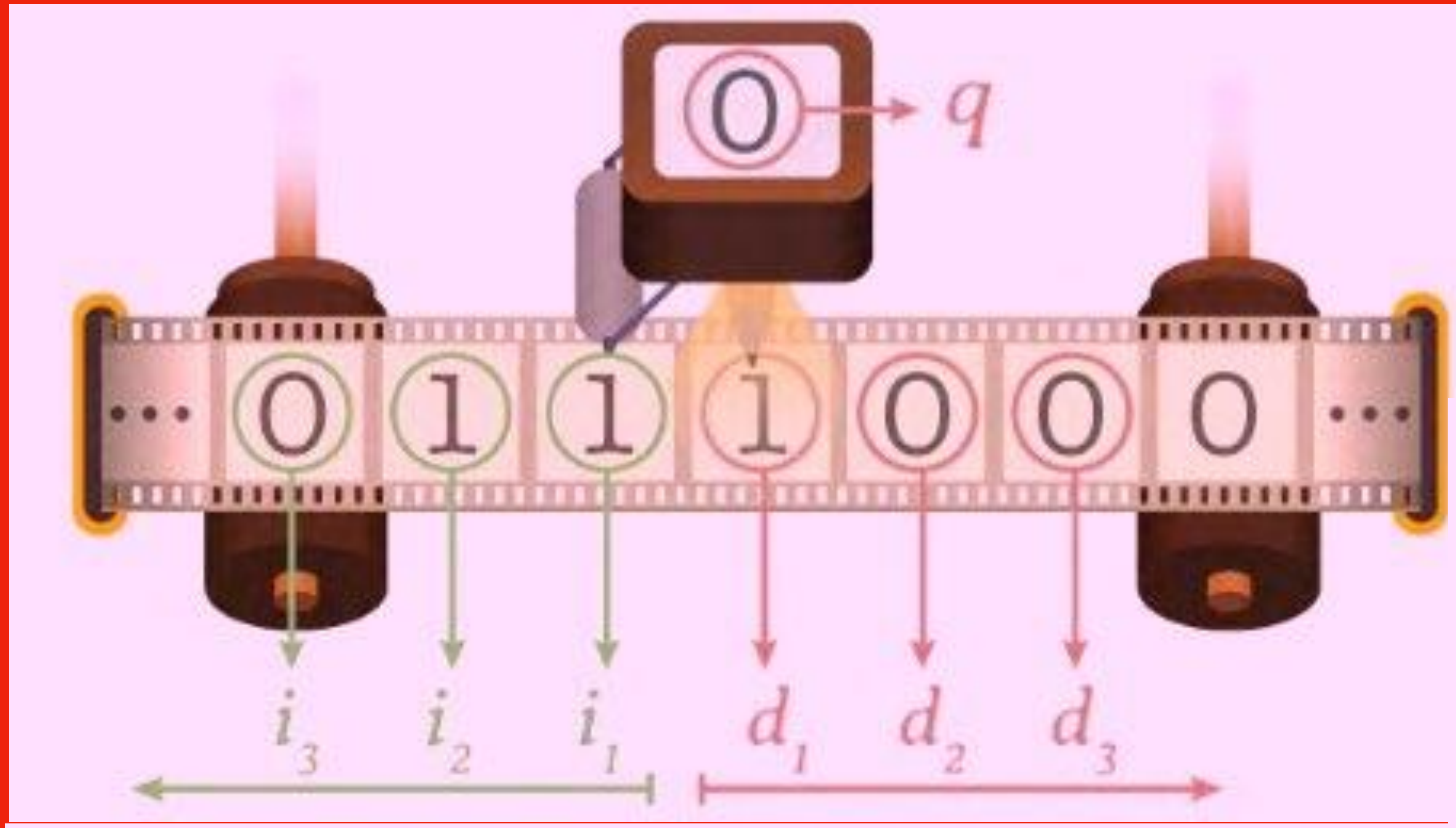
Turing machines and Conway's game...





John von Neumann: every Turing machine has a cellular automaton which simulates it.

Square Cantor set and Turing machines

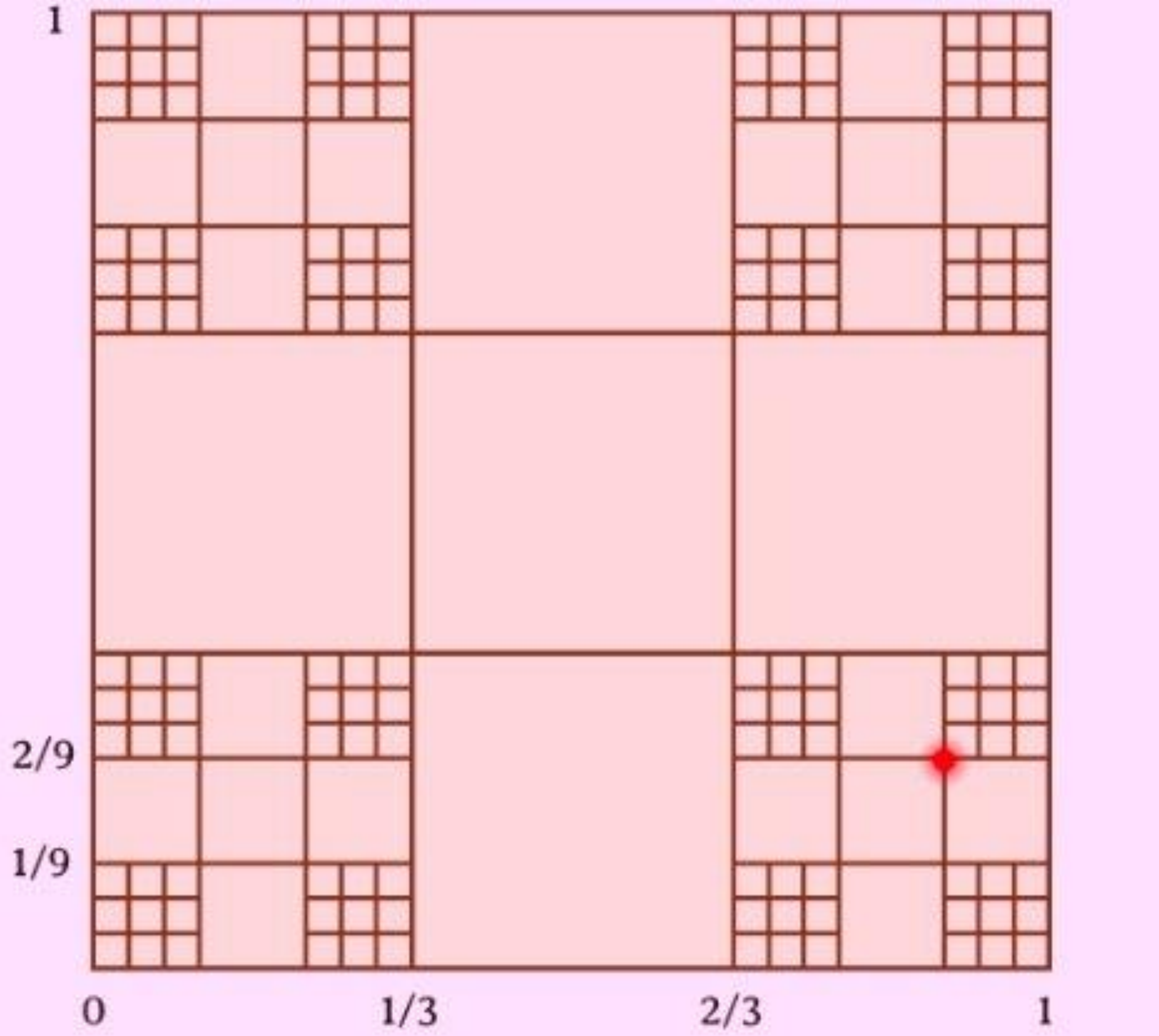


$$i = (i_1, i_2, i_3, \ldots) = (1, 1, 0, \ldots),$$

$$d = (d_1, d_2, d_3, \ldots) = (1, 0, 0, \ldots).$$

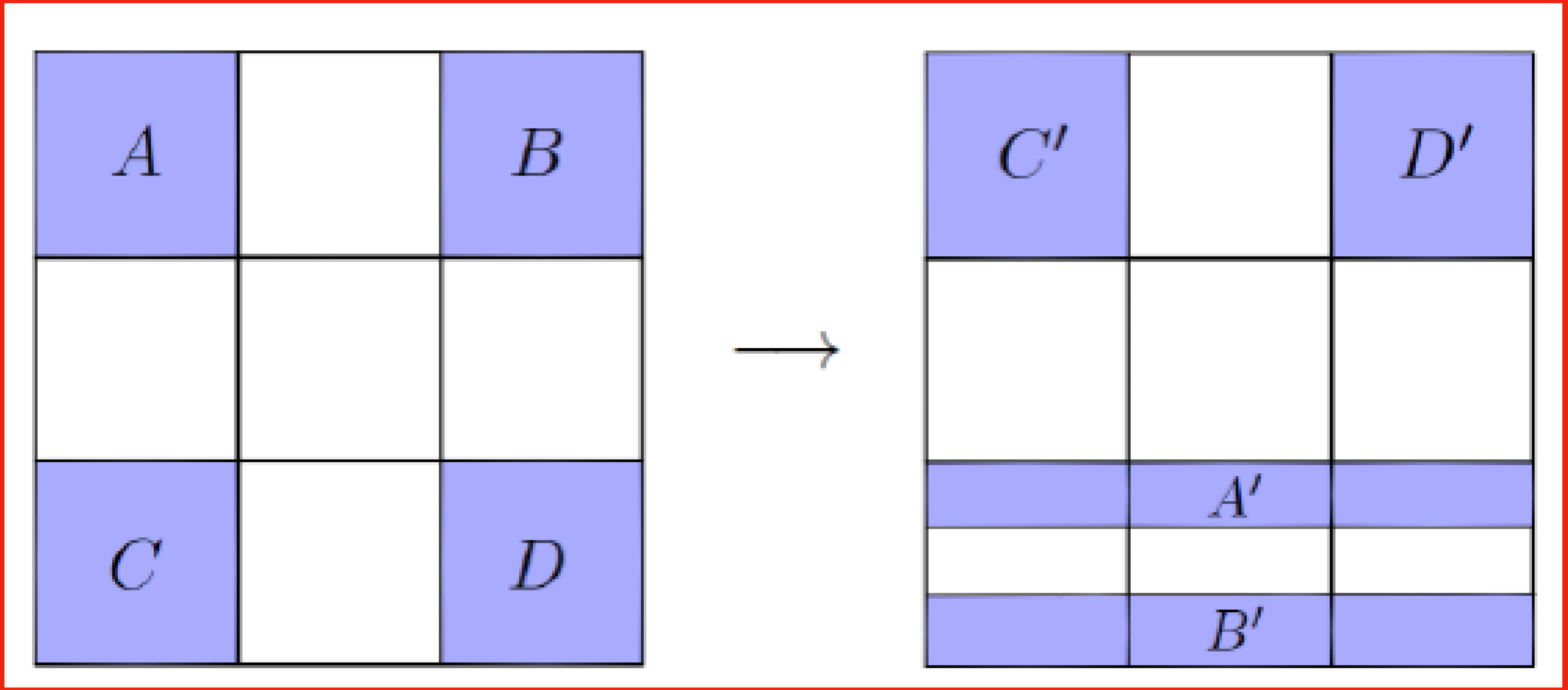
$$x = i_1 \cdot \frac{2}{3} + i_2 \cdot \frac{2}{3^2} + i_3 \cdot \frac{2}{3^3} + \cdots = \frac{2}{3} + \frac{2}{9},$$

$$y = q \cdot \frac{2}{3} + d_1 \cdot \frac{2}{3^2} + d_2 \cdot \frac{2}{3^3} + \cdots = \frac{2}{9}.$$



Each configuration of a Turing machine can be associated to a point in the square Cantor set.

Key point in Moore's construction



Any universal Turing machine is associated to transformations of the square Cantor set (a dynamical system).

Moore's theory

Moore's fundamental lemma (1991)

Given a reversible Turing machine there is a bijective map ϕ of the square Cantor set (a generalized shift) that is conjugated to it. This map is the restriction to the square Cantor set of a piecewise linear map which consists of finitely many area-preserving linear components defined on blocks.

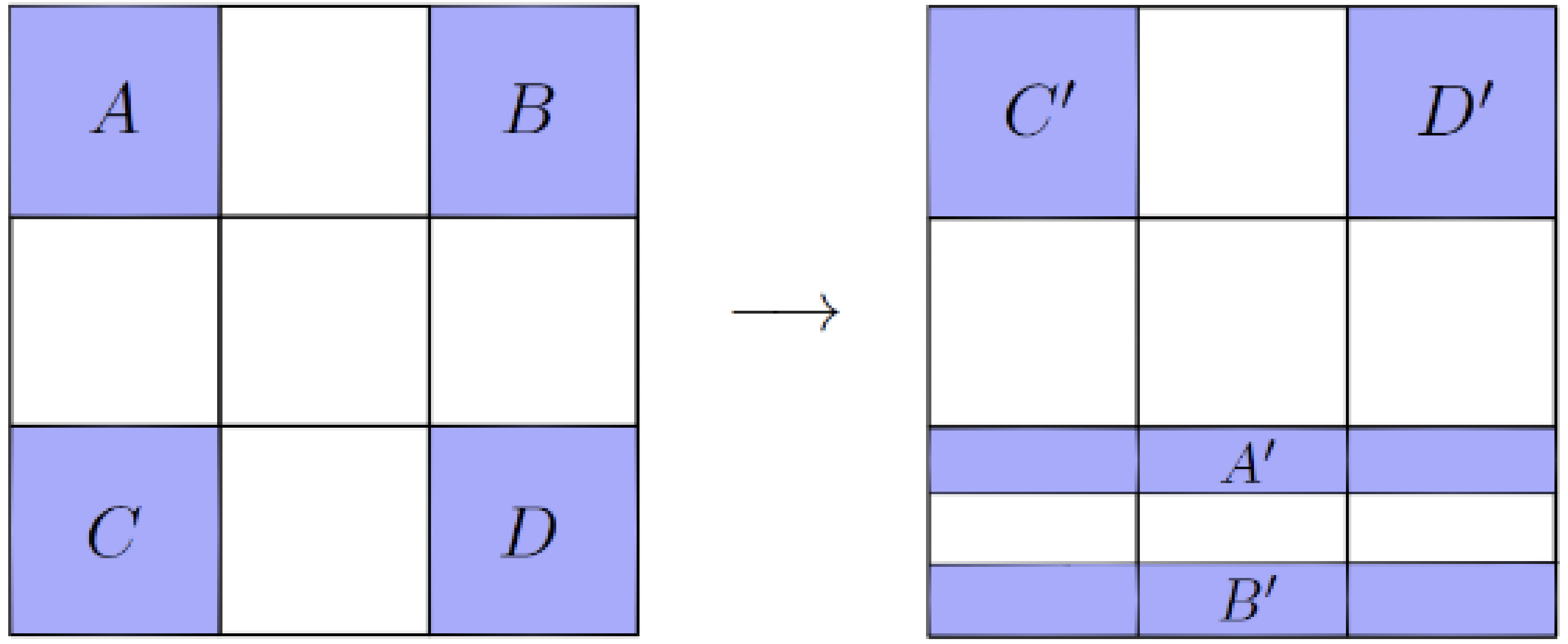


Figure: On A and B this basically acts as a translation of the baker's map, and on C and D this is simply a vertical translation

Turing machines associated to dynamical systems

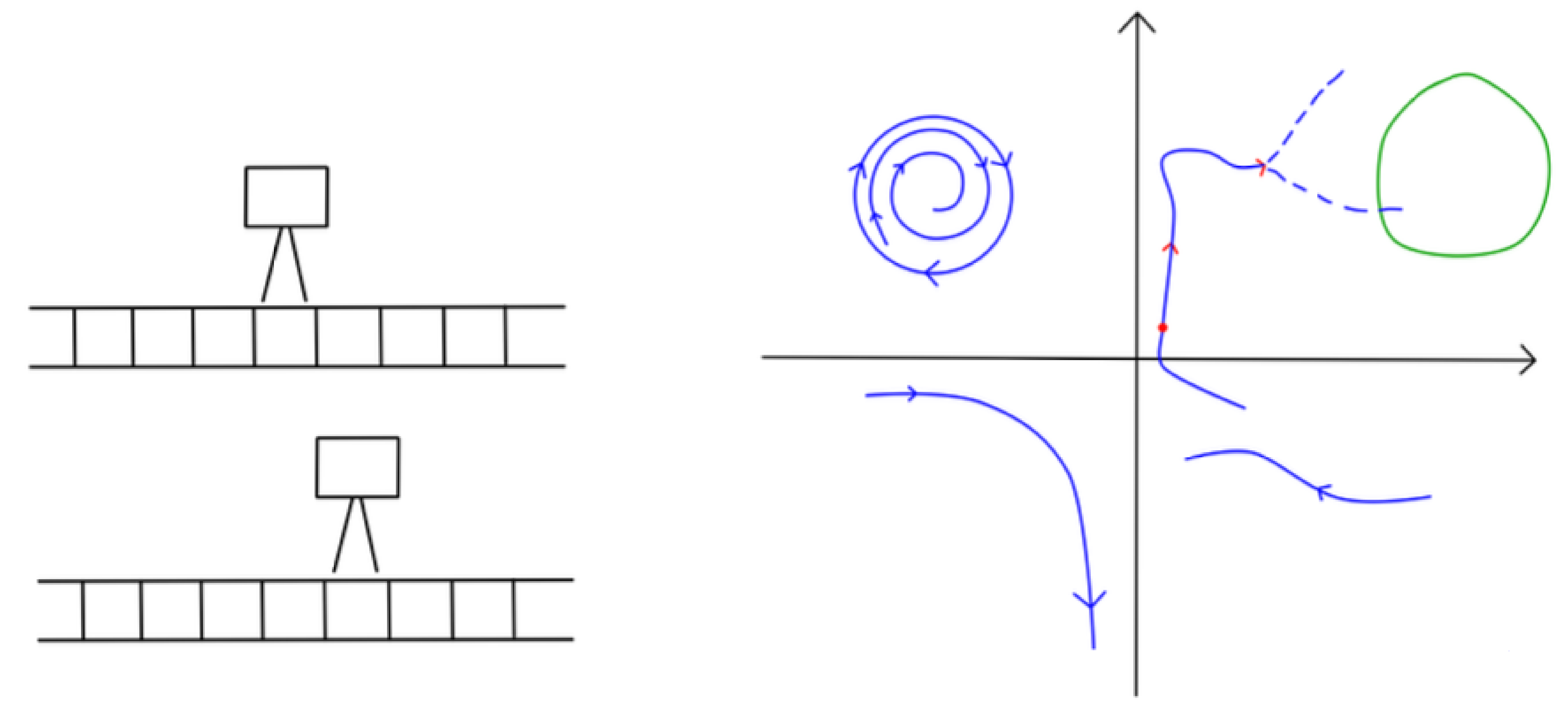


Figure: Turing machine and Turing complete vector field associated to a point and an open set.

A vector field is said to be Turing complete if it can simulate any Turing machine. In other words, the halting of any Turing machine with a certain input is equivalent to a certain trajectory of the field entering a certain open set in M.

Turing machines associated to dynamical systems

A vector field simulating a Turing machine

Let T be a Turing machine. A vector field X on $\mathbb{R}^3$ simulates T if: for any integer $k \geqslant 0$, an input tape t, and a finite output string $(t^*_{-k}, ..., t^*_k)$, there exist a computable point $p \in \mathbb{R}^3$ and a computable open set $U \subset \mathbb{R}^3$ such that the (forward)-orbit of X through p intersects U if and only if T halts with an output tape whose positions $-k, ..., k$ correspond to the symbols $t^*_{-k}, ..., t^*_k$.

If T_{un} is a universal Turing machine (i.e., a Turing machine that can simulate any other Turing machine), then any vector field that simulates T_{un} is called Turing complete.

Undecidability

Since the halting problem is undecidable, a Turing complete vector field X exhibits undecidable trajectories: no general algorithm to check whether the trajectories of X starting at certain initial points will intersect a certain open set (certain $\equiv$ computable).

Computation with stationary fluids

An inviscid and incompressible fluid flow in equilibrium on a Riemannian manifold (M, g) (of dimension 3) is described by the stationary Euler equations:

$$\nabla_X X = -\nabla P\,, \qquad \operatorname{div} X = 0$$

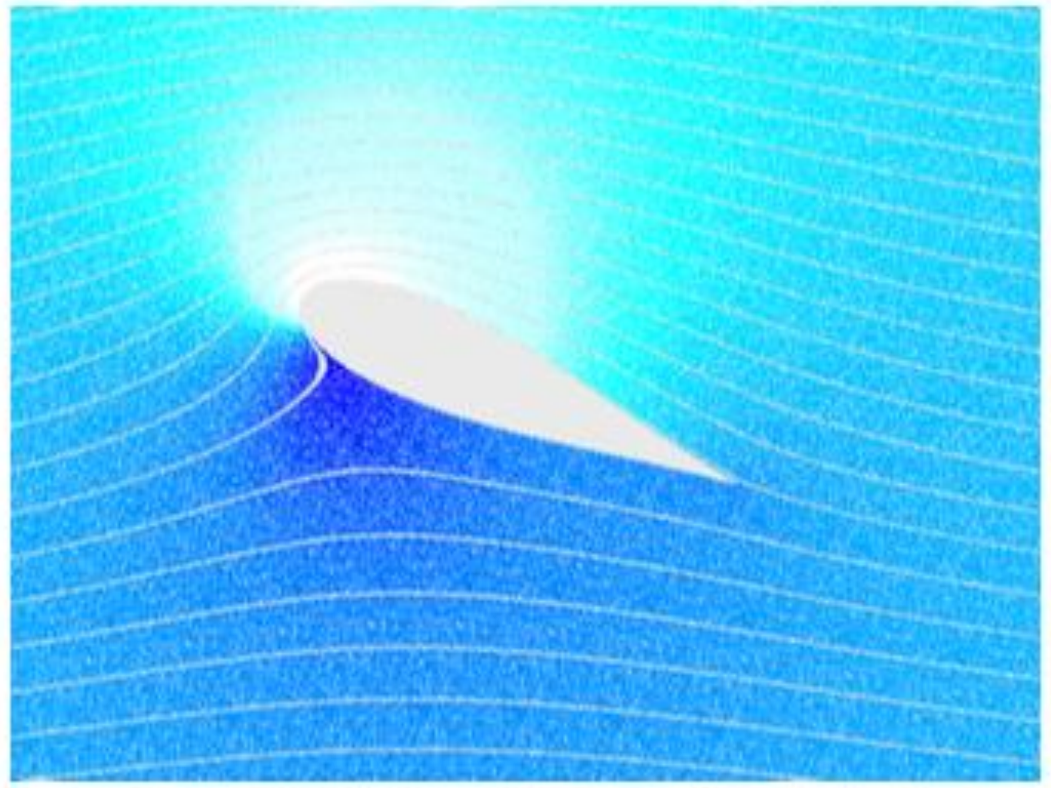
- X is the velocity field of the fluid: a vector field on M.
- P is the hydrodynamic pressure of the fluid: a scalar function on M.

Objective

We want to construct a stationary solution of the Euler equations whose velocity field X is Turing complete: any computer algorithm can be simulated using the orbits of X (the fluid particle paths).

$\Longrightarrow$ to this end, we shall use the most flexible class of steady Euler flows: the Beltrami fields.

Incompressible fluids on Riemannian manifolds


Classical Euler equations on $\mathbb{R}^3$:

$$\begin{cases} \dfrac{\partial X}{\partial t} + (X \cdot \nabla) X = -\nabla P \\ \operatorname{div} X = 0 \end{cases}$$

The evolution of an **inviscid and incompressible fluid flow** on a Riemannian n-dimensional manifold (M, g) is described by the Euler equations:

$$\frac{\partial X}{\partial t} + \nabla_X X = -\nabla P \,, \qquad \operatorname{div} X = 0$$

- X is the velocity field of the fluid: a non-autonomous vector field on M.
- P is the inner pressure of the fluid: a time-dependent scalar function on M.

Incompressible fluids on Riemannian manifolds

If X does not depend on time, it is a steady or stationary Euler flow: it models a fluid flow in equilibrium. The equations can be written as:

$$\nabla_X X = -\nabla P\,, \qquad \operatorname{div} X = 0\,,$$

$$\iff \quad \iota_X d\alpha = -dB\,, \qquad d\iota_X \mu = 0\,, \qquad \alpha(\cdot) := g(X,\cdot)$$

where $B := P + \frac{1}{2}||X||^2$ is the Bernoulli function.

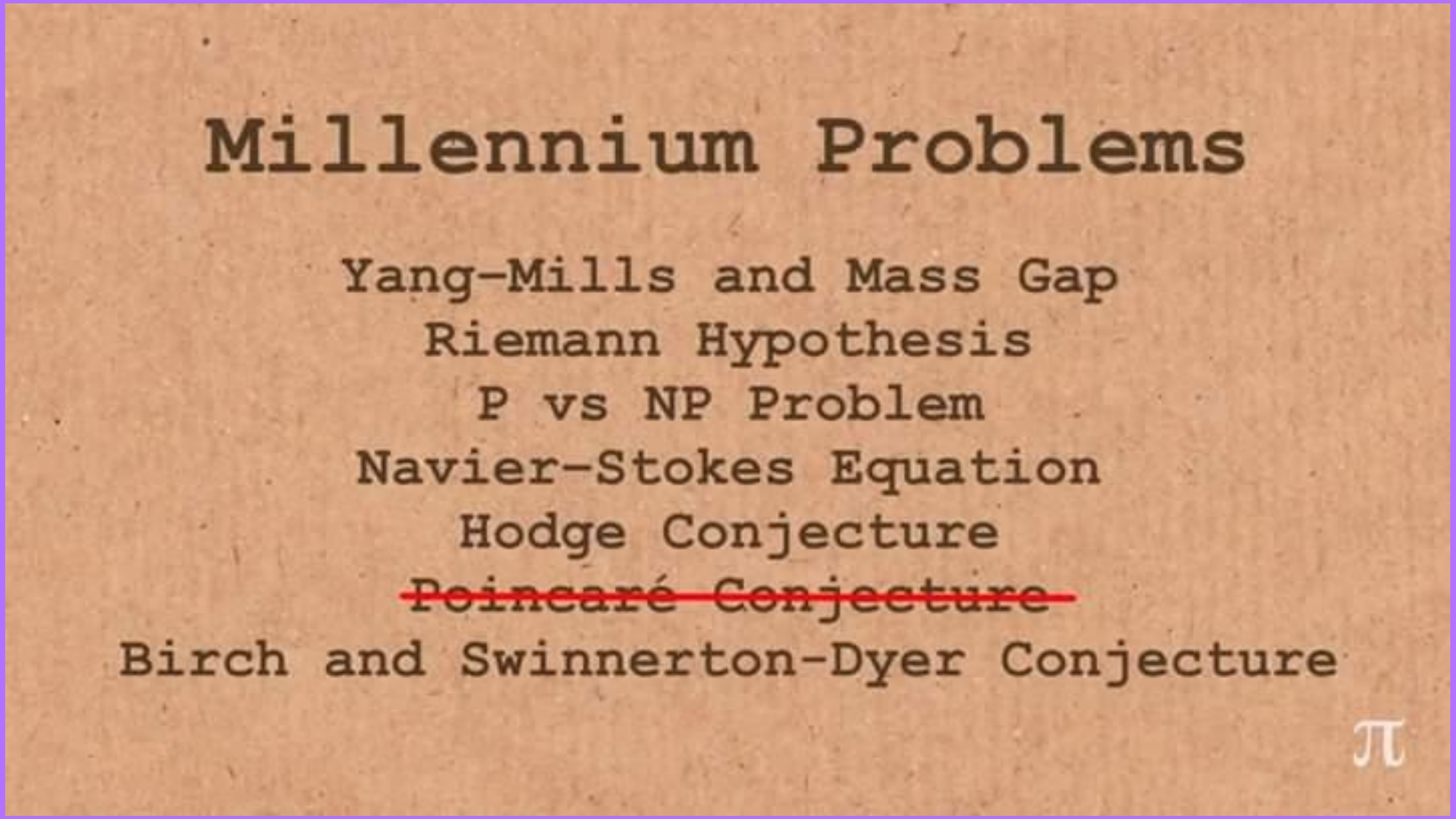
Beltrami fields:

$\operatorname{curl} X = fX$, with $f \in C^\infty(M) \qquad \operatorname{div} X = 0.$

Example (Hopf fields on S^3 and ABC fields on T^3)

- The Hopf fields $u_1 = (-y, x, \xi, -z)$ and $u_2 = (-y, x, -\xi, z)$ are Beltrami fields on S^3.
- The ABC flows $(\dot{x}, \dot{y}, \dot{z}) = (A\sin z + C\cos y, B\sin x + A\cos z, C\sin y + B\cos x)$, $((x, y, z) \in (\mathbb{R}/2\pi\mathbb{Z})^3)$ are Beltrami.

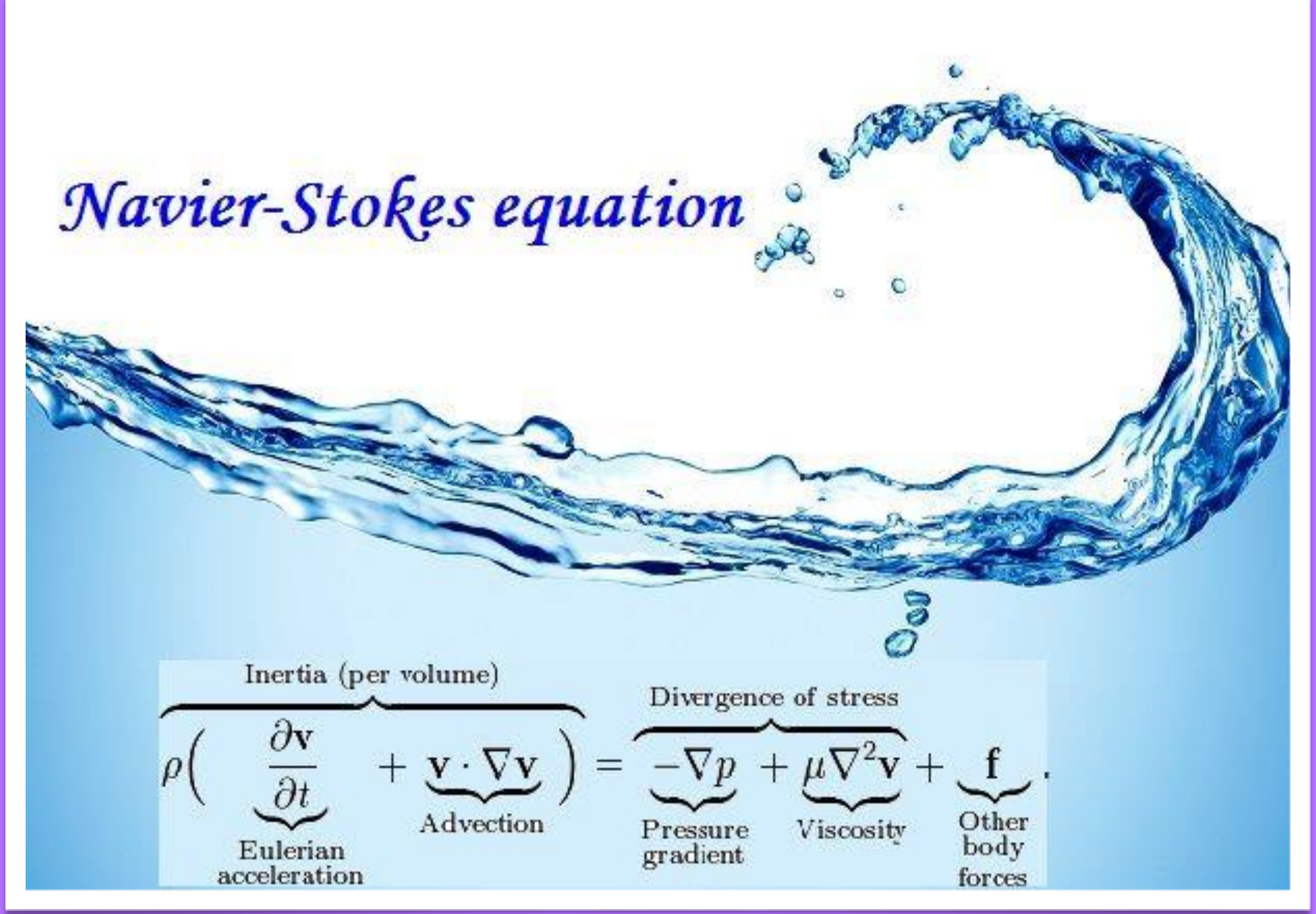
A million dollars for a correct answer



Navier-Stokes problem

Existence of smooth solutions

The Navier-Stokes equations model the motion of an incompressible and viscous fluid.



The inviscid case corresponds to the Euler equations.

Navier-Stokes in a nutshell

Quick formulation of the problem

The problem is to determine whether all initial conditions - starting configurations of the fluid - give rise to smooth solutions that evolve indefinitely, or whether, in certain circumstances, solutions degenerate and "blow up" after a certain time. This explosion corresponds to the appearance of singularities, regions of space where the energy of the fluid becomes concentrated to the point of becoming infinite.

The Navier-Stokes regularity problem

Formulation of the problem

The *Navier–Stokes* equations are then given by

$$(1) \qquad \frac{\partial}{\partial t}u_i + \sum_{j=1}^{n} u_j \frac{\partial u_i}{\partial x_j} = \nu \Delta u_i - \frac{\partial p}{\partial x_i} + f_i(x,t) \qquad (x \in \mathbb{R}^n, t \geq 0),$$

$$(2) \qquad \operatorname{div} u = \sum_{i=1}^{n} \frac{\partial u_i}{\partial x_i} = 0 \qquad (x \in \mathbb{R}^n, t \geq 0)$$

with initial conditions

$$(3) \qquad u(x,0) = u^\circ(x) \qquad (x \in \mathbb{R}^n).$$

Here, $u^\circ(x)$ is a given, C^∞ divergence-free vector field on $\mathbb{R}^n$, $f_i(x,t)$ are the components of a given, externally applied force (e.g. gravity), ν is a positive coefficient (the viscosity), and $\Delta = \sum_{i=1}^{n} \frac{\partial^2}{\partial x_i^2}$ is the Laplacian in the space variables. The *Euler equations* are equations (1), (2), (3) with ν set equal to zero.

The Navier-Stokes regularity problem

Formulation of the problem

$$|\partial_x^\alpha u^\circ(x)| \le C_{\alpha K}(1+|x|)^{-K} \quad \text{on } \mathbb{R}^n, \text{ for any } \alpha \text{ and } K \tag{4}$$

and

$$|\partial_x^\alpha \partial_t^m f(x,t)| \le C_{\alpha m K}(1+|x|+t)^{-K} \quad \text{on } \mathbb{R}^n \times [0,\infty), \text{ for any } \alpha, m, K. \tag{5}$$

We accept a solution of (1), (2), (3) as physically reasonable only if it satisfies

$$p, u \in C^\infty(\mathbb{R}^n \times [0,\infty)) \tag{6}$$

and

$$\int_{\mathbb{R}^n} |u(x,t)|^2 dx < C \quad \text{for all } t \ge 0 \quad \text{(bounded energy)}. \tag{7}$$

Alternatively, to rule out problems at infinity, we may look for spatially periodic solutions of (1), (2), (3). Thus, we assume that $u^\circ(x)$, $f(x,t)$ satisfy

$$u^\circ(x+e_j) = u^\circ(x), \quad f(x+e_j,t) = f(x,t) \quad \text{for } 1 \le j \le n \tag{8}$$

The Navier-Stokes regularity problem

Formulation of the problem

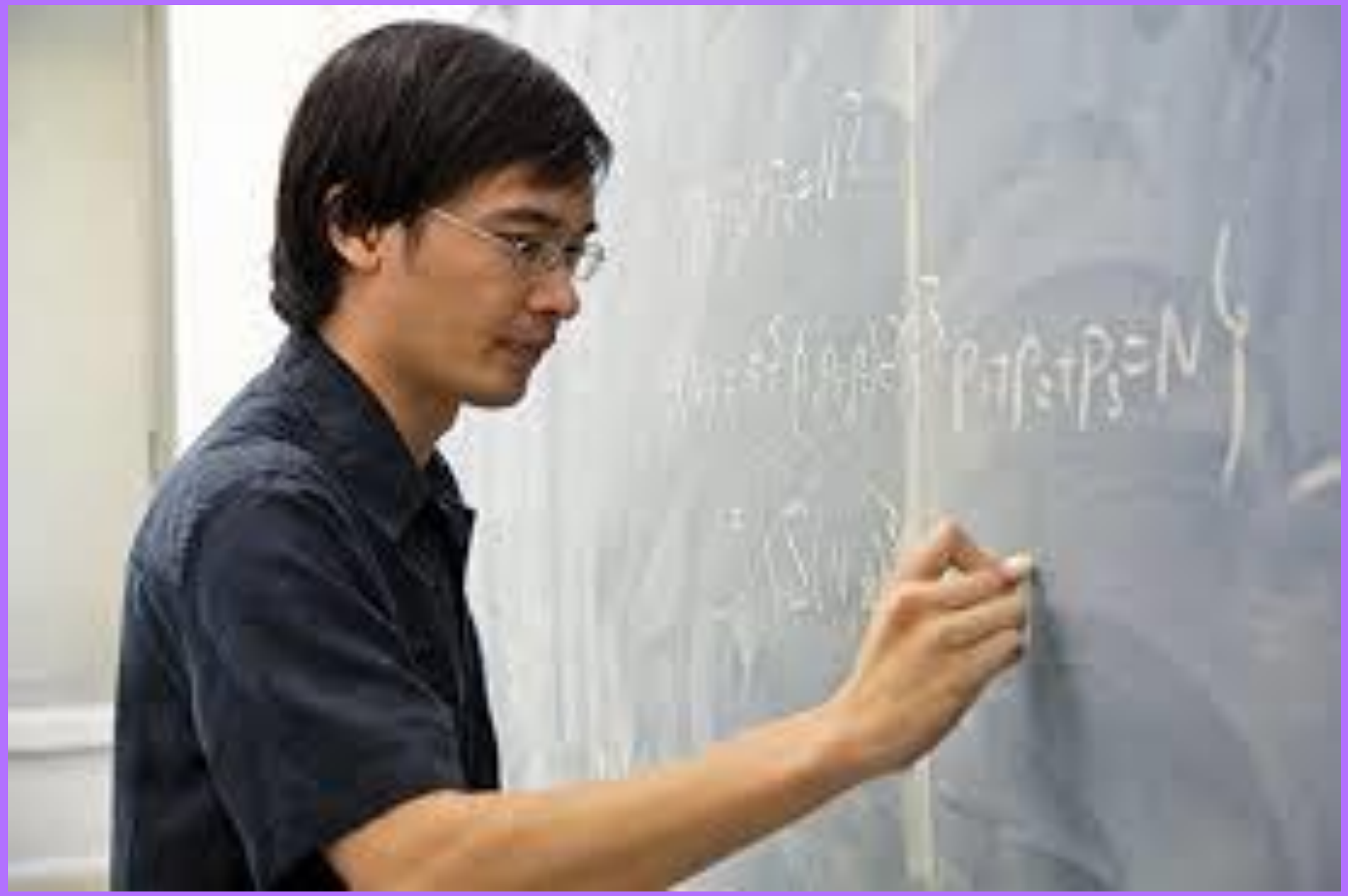
(A) Existence and smoothness of Navier–Stokes solutions on $\mathbb{R}^3$. Take $\nu > 0$ and $n = 3$. Let $u^\circ(x)$ be any smooth, divergence-free vector field satisfying (4). Take $f(x,t)$ to be identically zero. Then there exist smooth functions $p(x,t), u_i(x,t)$ on $\mathbb{R}^3 \times [0,\infty)$ that satisfy (1), (2), (3), (6), (7).

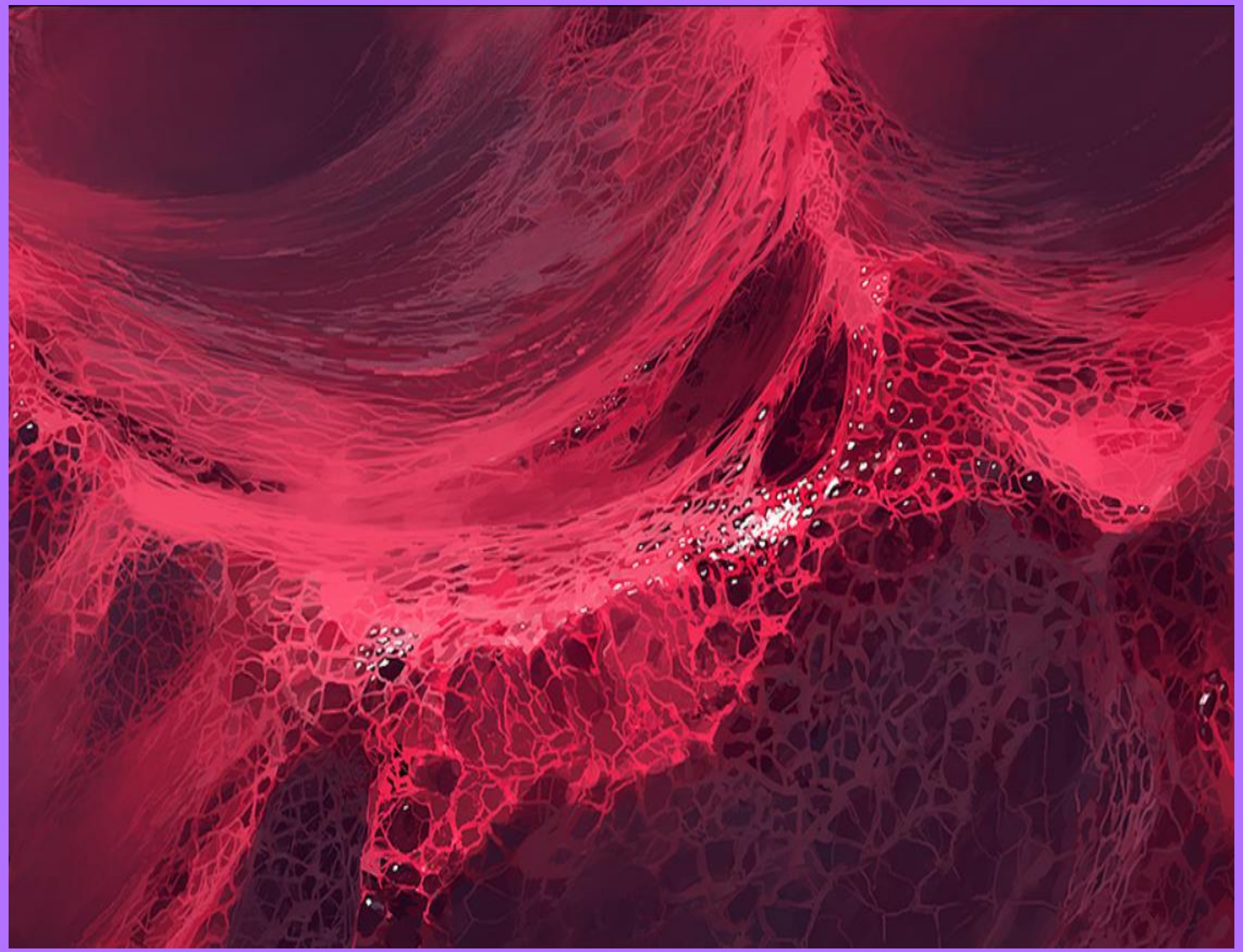
(C) Breakdown of Navier–Stokes solutions on $\mathbb{R}^3$. Take $\nu > 0$ and $n = 3$. Then there exist a smooth, divergence-free vector field $u^\circ(x)$ on $\mathbb{R}^3$ and a smooth $f(x,t)$ on $\mathbb{R}^3 \times [0,\infty)$, satisfying (4), (5), for which there exist no solutions (p,u) of (1), (2), (3), (6), (7) on $\mathbb{R}^3 \times [0,\infty)$.

From dimension 2 to 3

The 2-dimensional case was solved by Olga Ladyzhenskaya in 1958.
The 3-dimensional case is still open.

Tao's approach…





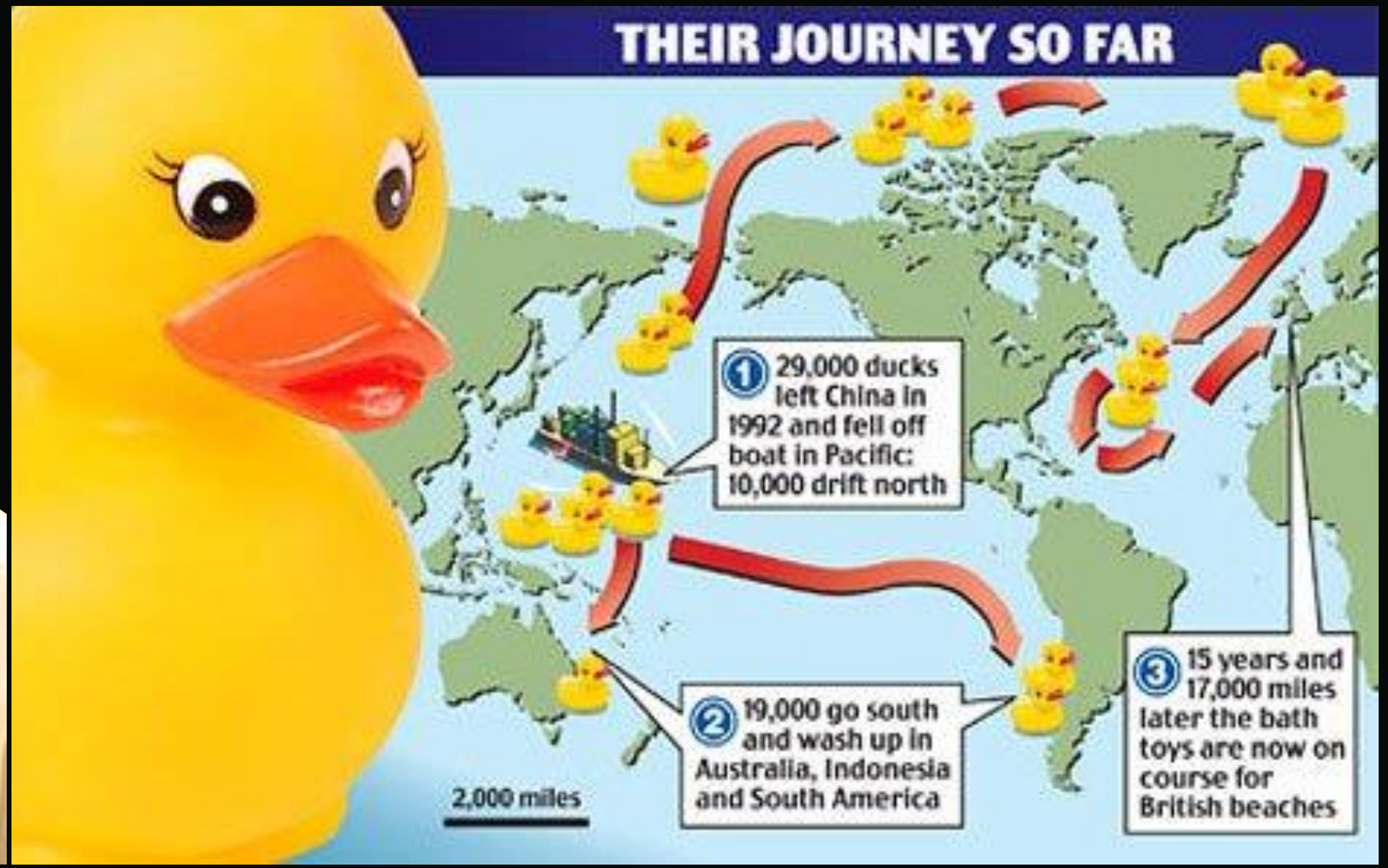
"One could hope to design logic gates entirely out of ideal fluid. If these gates were sufficiently "Turing complete", and also "noise-tolerant" one could then hope to combine enough of these gates together to "program" a self-replicating von Neumann machine"

Tao, JAMS 2016.

Tao's dream: To create an initial entry programmed to evolve as a rescaled version of itself (like a Von Neumann self-replicating machine). Can this idea be applied to achieve a blow-up in the Navier-Stokes equations?

“A Fluid computer”

From the friendly floatees to the Fluid Computer

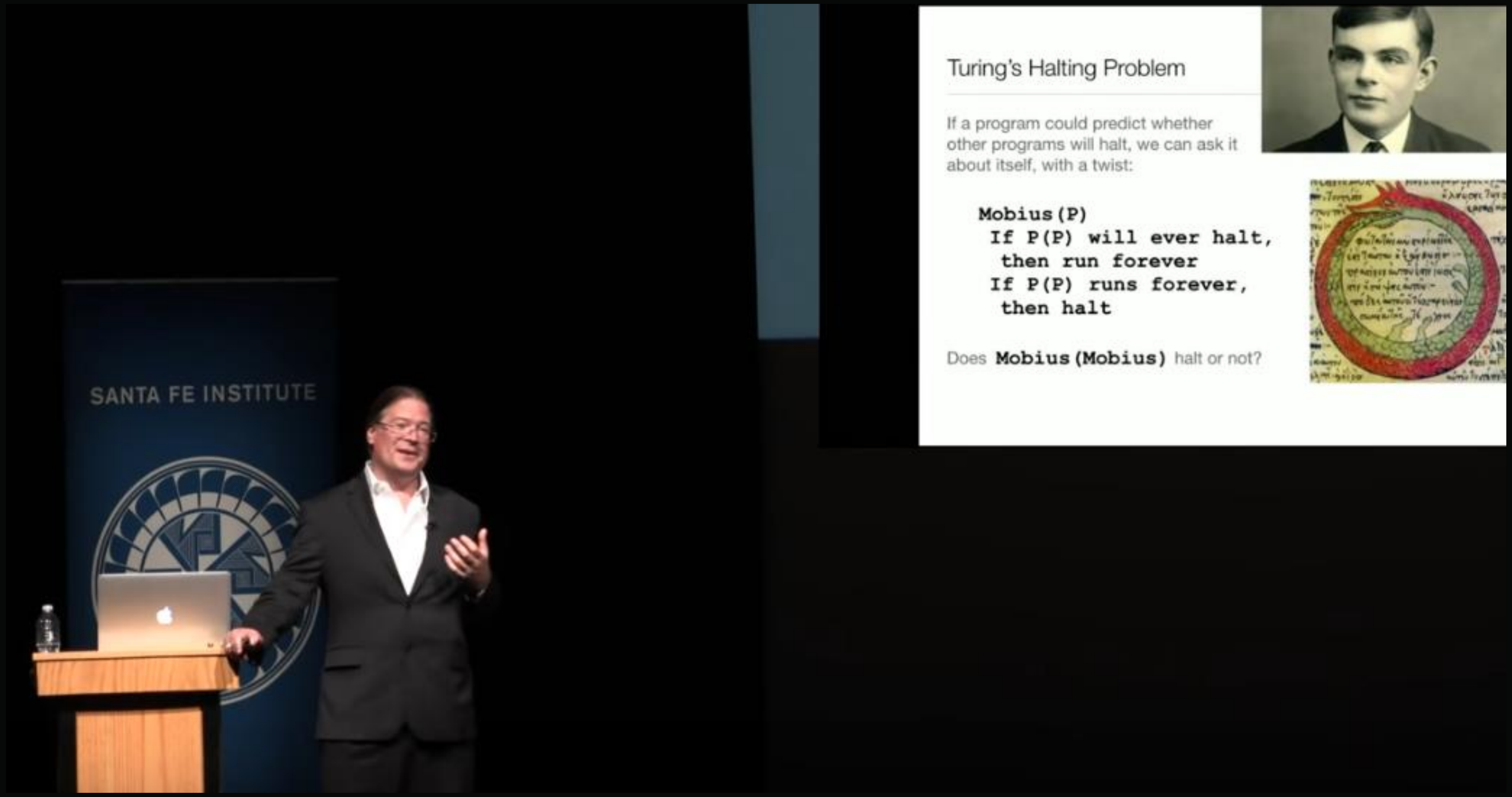


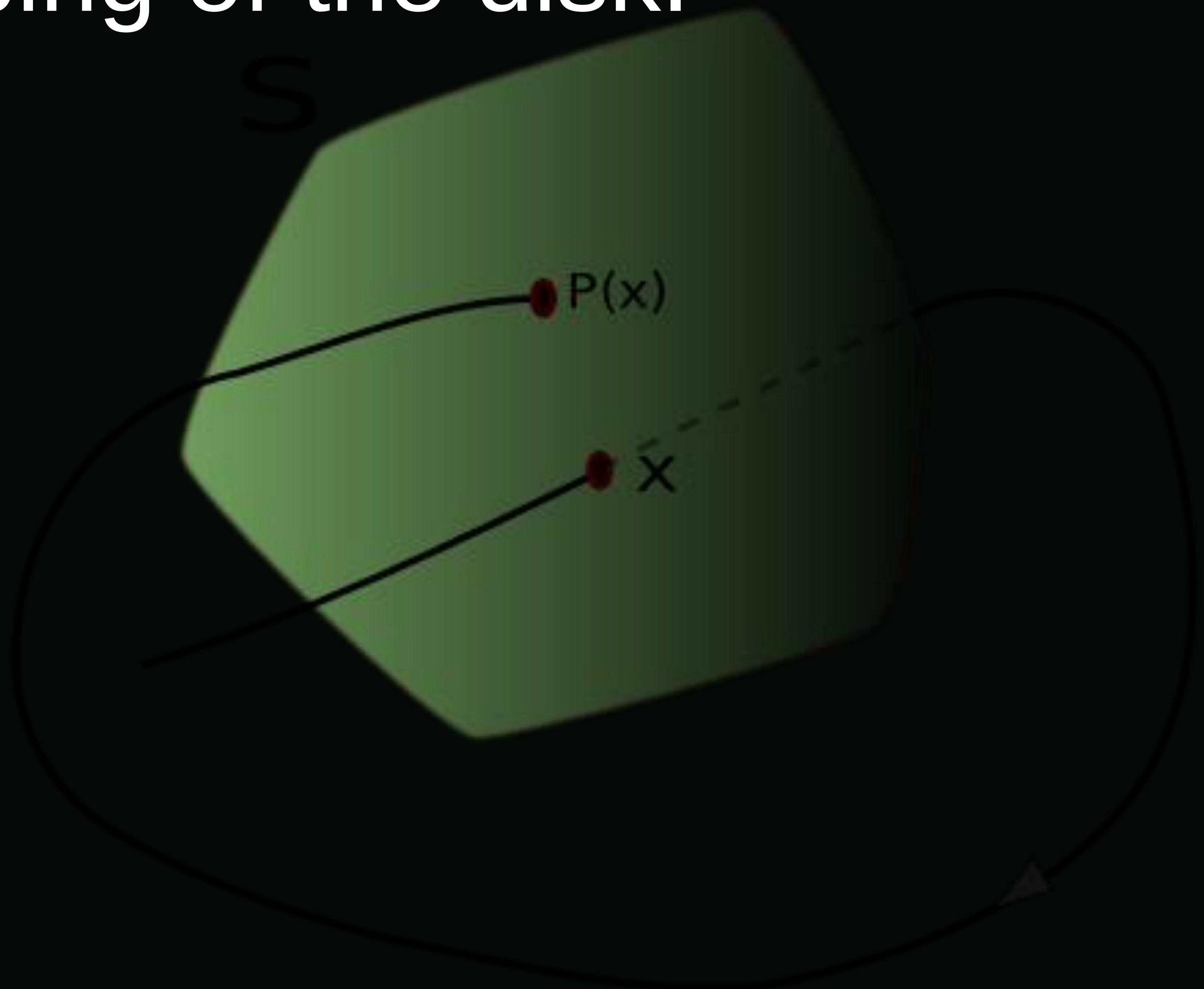
- **1991, Moore:** Is hydrodynamics capable of performing computations?
- **January 10, 1992:** 29000 rubber ducks are lost in the ocean.
- **July 2007:** A rubber duck shows up in Scotland.
- **December 2020:** (Cardona-M.-Peralta-Presas, PNAS 2021) There exist stationary Euler flows in dimension 3 which are Turing complete, i.e., they can simulate any Turing machine (Fluid computer).

Our construction

Logical chaos from 2D to 3D

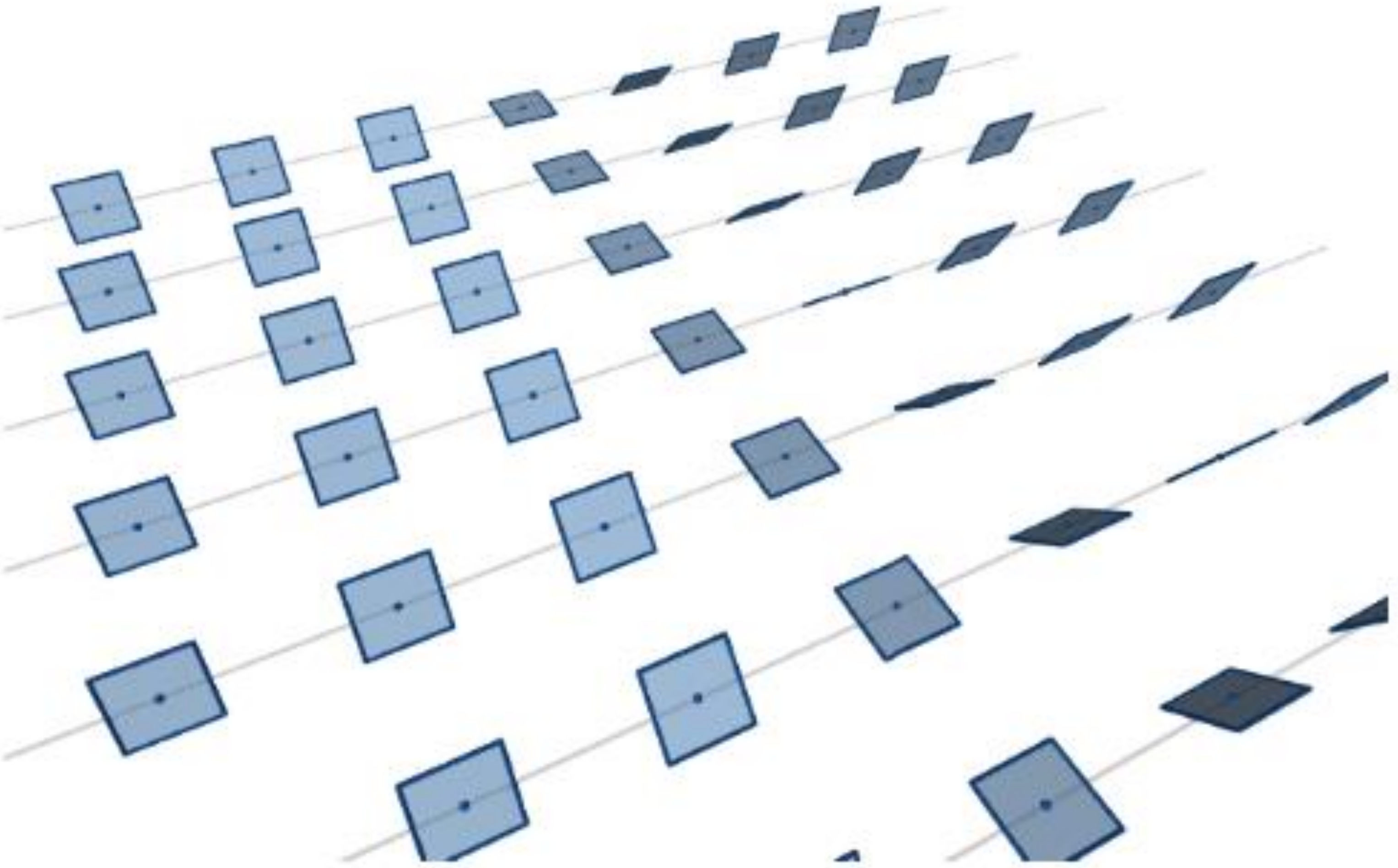
- Present Moore’s transformation as a Poincaré section of a trajectory of a 3-dimensional vector field. First extend mapping to smooth mapping of the disk.





This vector field has a special geometric features “Reeb". What is the relation to Euler equations? And to Navier-Stokes?

New tools: Geometries of forms



Symplectic	Contact
$\dim M = 2n$	$\dim M = 2n+1$
2-form ω, non-degenerate $d\omega = 0$	1-form α, $\alpha \wedge (d\alpha)^n \neq 0$
Darboux theorem $\omega = \sum_{i=1}^n dx_i \wedge dy_i$	$\alpha = dx_0 - \sum_{i=1}^n x_i dy_i$
Hamiltonian $\iota_{X_H}\omega = -dH$	Reeb $\alpha(R) = 1$, $\iota_R d\alpha = 0$ Ham. $\begin{cases} \iota_{X_H}\alpha = H \\ \iota_{X_H} d\alpha = -dH + R(H)\alpha. \end{cases}$

An example of contact structure

The kernel of a 1-form α on M^{2n+1} is a contact structure whenever $\alpha \wedge (d\alpha)^n$ is a volume form $\Leftrightarrow d\alpha|_\xi$ is non-degenerate.

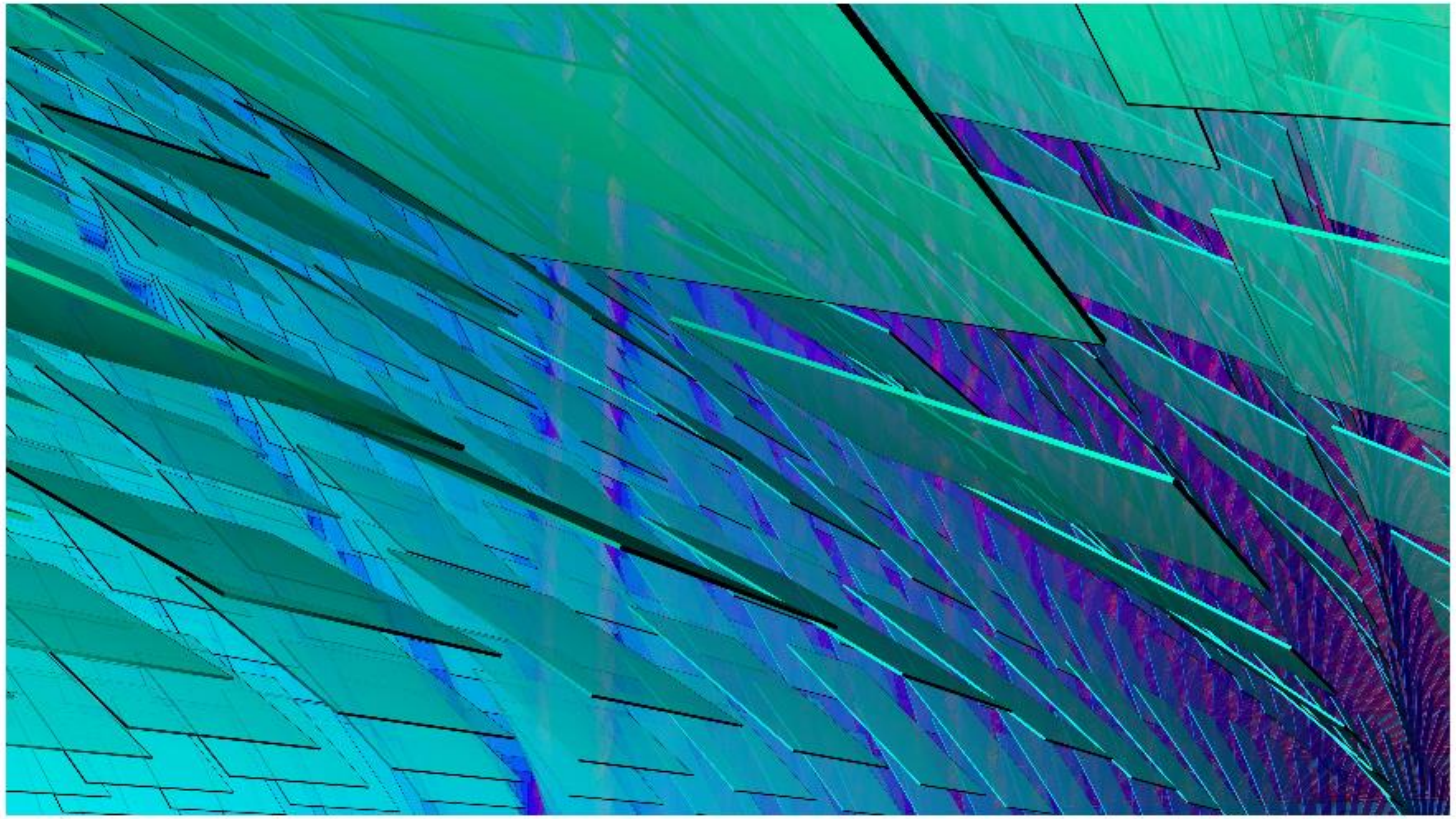
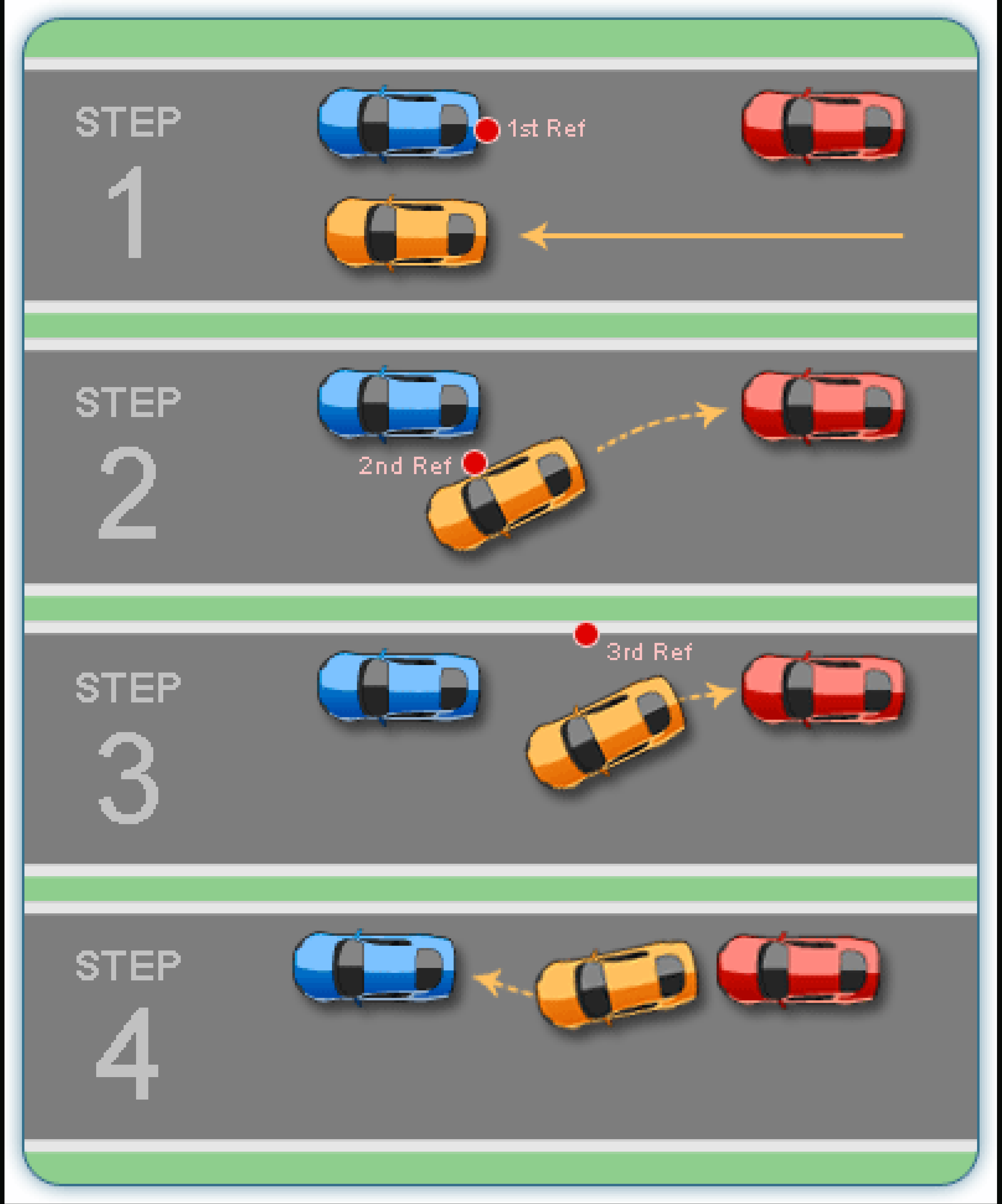


Figure: Standard contact structure on $\mathbb{R}^3$ by Robert Ghrist

$$\alpha = dz - y dx \quad \xi = \ker \alpha = \left\langle \frac{\partial}{\partial y}, y\frac{\partial}{\partial z} + \frac{\partial}{\partial x} \right\rangle d\alpha = -dy \wedge dx = dx \wedge dy$$

$$\Rightarrow \ \alpha \wedge d\alpha \ = \ dx \wedge dy \wedge dz$$

Contact geometry explains why it is difficult to park a car



Contact geometry and parallel parking

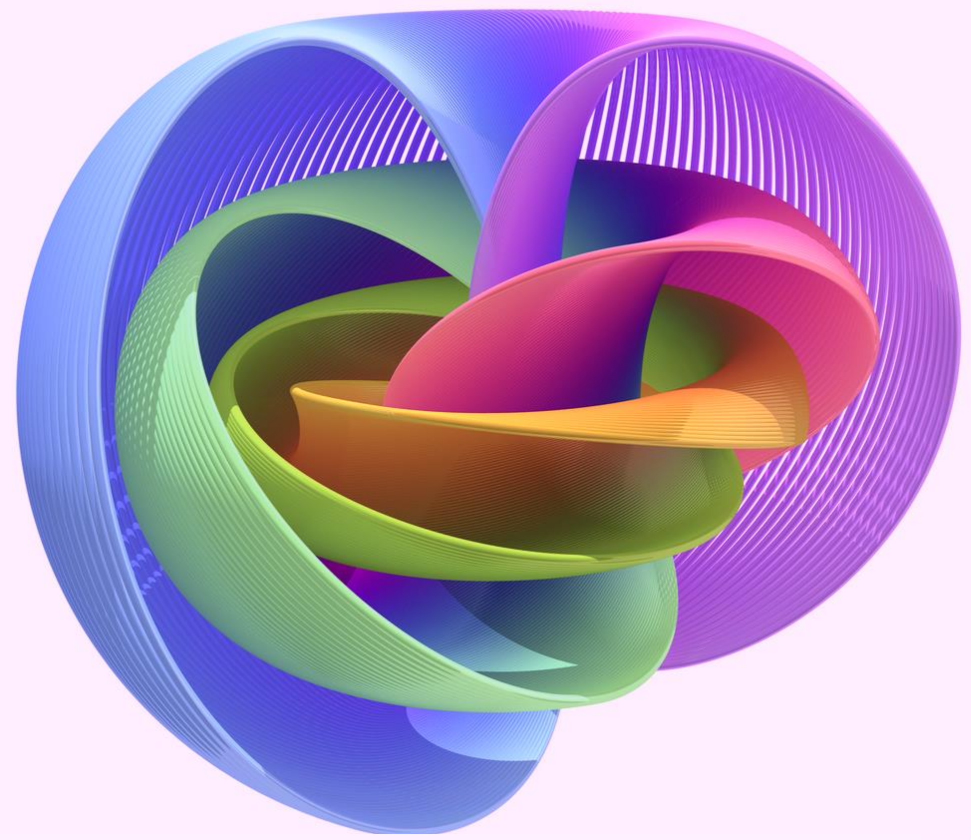
Theorem 17. *A car of length L can be parallel parked in any space of length $L+\epsilon$, $\epsilon > 0$.*

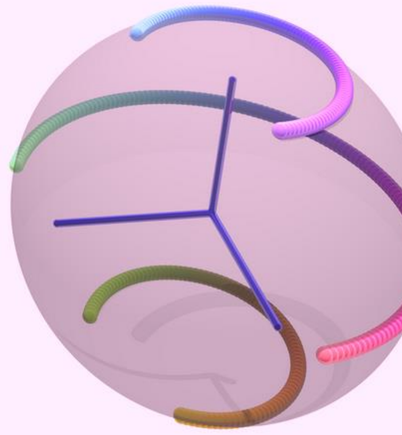
Proof. Let us assume that the car is on the plane $\mathbb{R}^2$. Its position can be described by a single coordinate (x, y) and the angle $\theta \in S^1$ its tires are facing, or equivalently a point in the configuration space $\mathbb{R}^2 \times S^1$, which has contact form

$$\alpha = \sin\theta dx - \cos\theta dy.$$

(Note that $\alpha \wedge d\alpha = -dx \wedge dy \wedge d\theta$, so we will reverse the usual orientation of S^1.) The car's path $\gamma(t) = (x(t), y(t), \theta(t))$ will satisfy $\frac{dy}{dx} = \tan\theta$, or equivalently $\frac{dx}{dt}\sin\theta - \frac{dy}{dt}\cos\theta = 0$: thus $\gamma(t)$ must be Legendrian. We now take a path through configuration space which pulls the car up parallel to the parking spot and then slides it horizontally into place without turning the wheel; this is physically impossible, but an arbitrarily close Legendrian approximation will successfully park the car. □

Hopf fields as Reeb and Beltrami fields





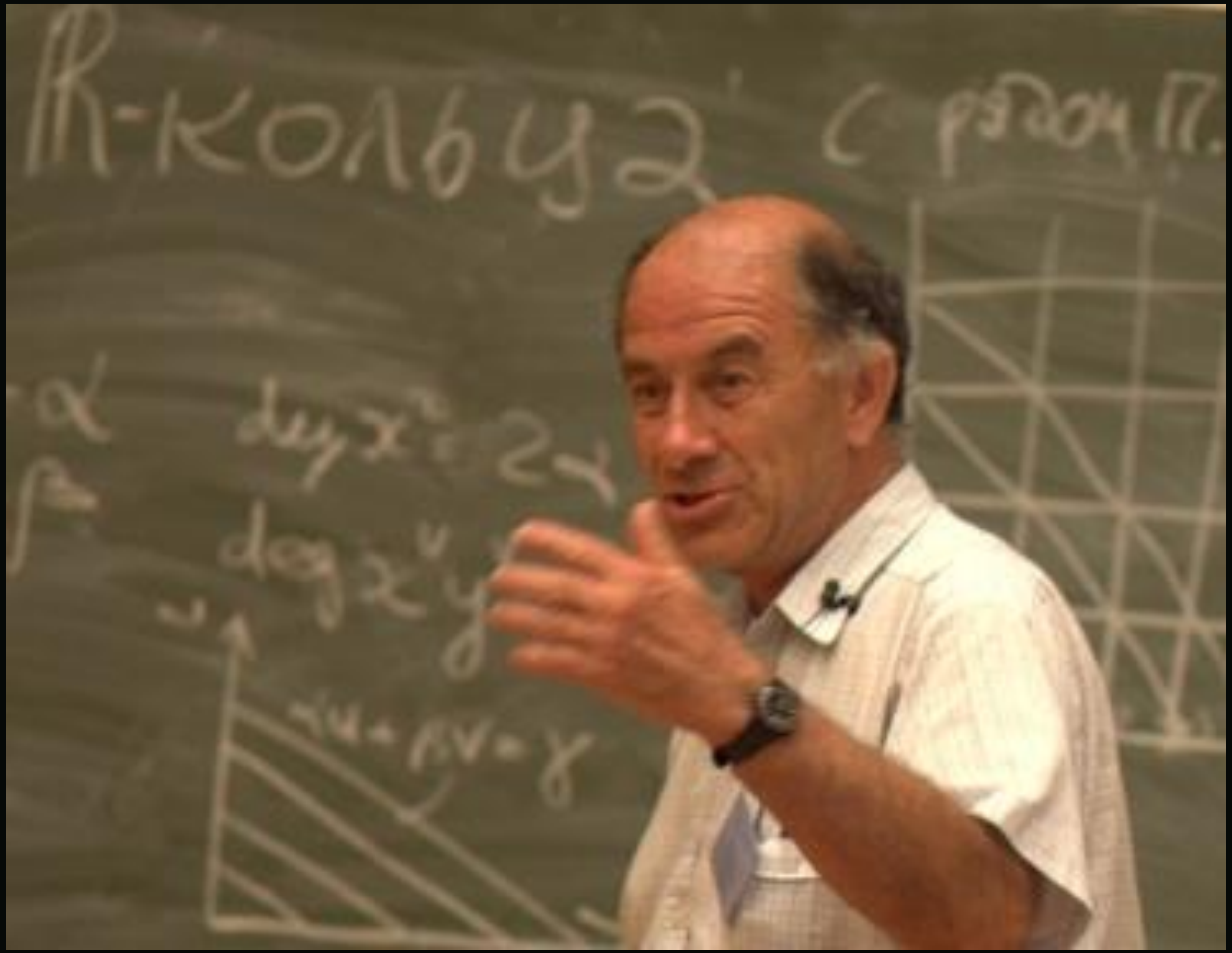
- $S^3 := \{(u,v) \in \mathbb{C}^2 \mid |u|^2 + |v|^2 = 1\}, \alpha = \frac{1}{2}(ud\overline{u} - \overline{u}du + vd\overline{v} - \overline{v}dv)$.

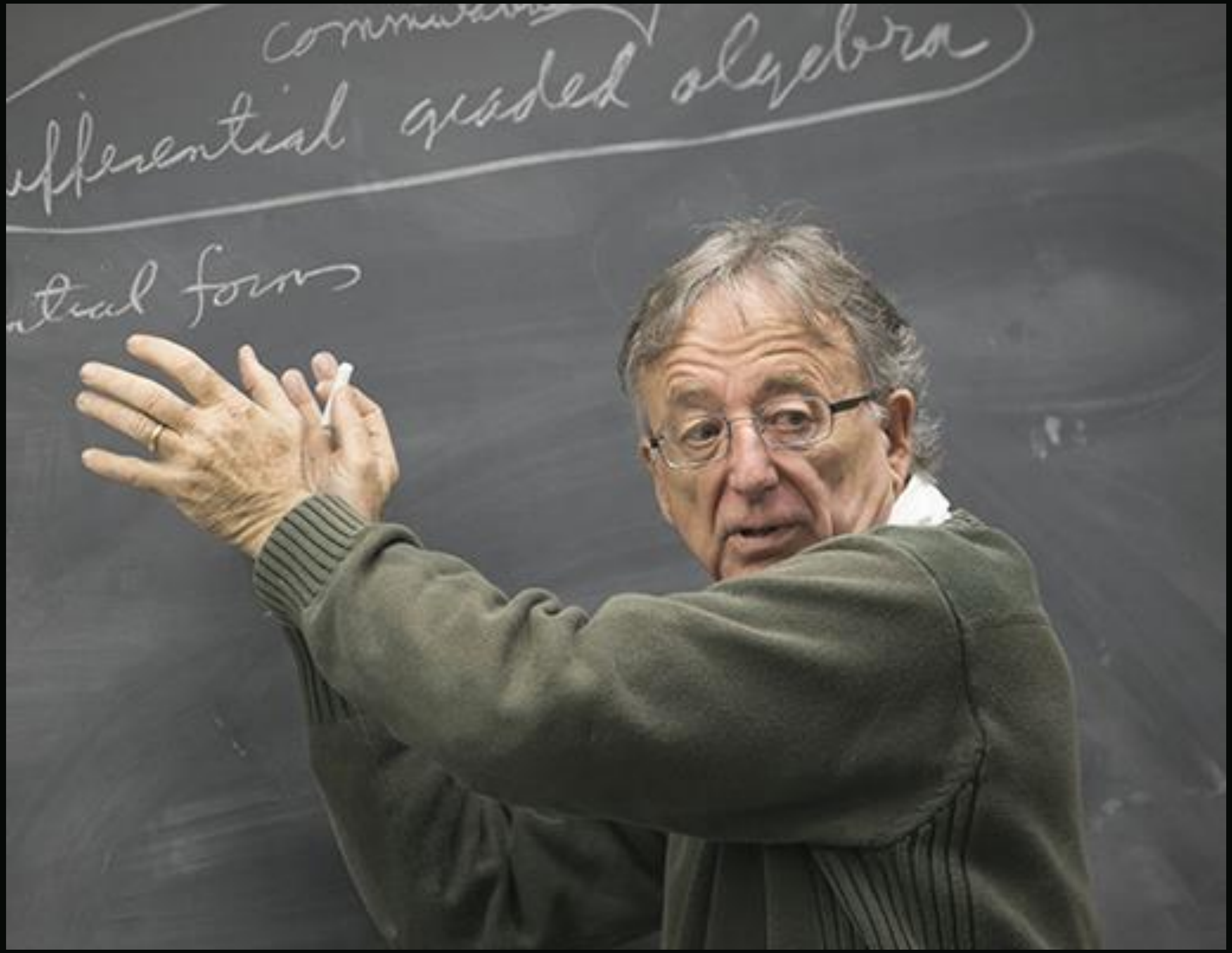
  The orbits of the Reeb vector field form the Hopf fibration!

$$R_\alpha = iu\frac{\partial}{\partial u} - i\overline{u}\frac{\partial}{\partial \overline{u}} + iv\frac{\partial}{\partial v} - i\overline{v}\frac{\partial}{\partial \overline{v}}$$

- $\mathbb{S}^3 = \{(z_1, z_2) \in \mathbb{C}^2 : |z_1|^2 + |z_2|^2 = 1\}$ can be endowed with Hopf coordinates $(z_1, z_2) = (\cos s \exp i\phi_1, \sin s \exp i\phi_2)$, $s \in [0, \pi/2]$, $\phi_{1,2} \in [0, 2\pi)$. The Hopf field $R := \partial_{\phi_1} + \partial_{\phi_2}$ is a steady Euler flow (Beltrami) with respect to the round metric.

Geometry of Fluids

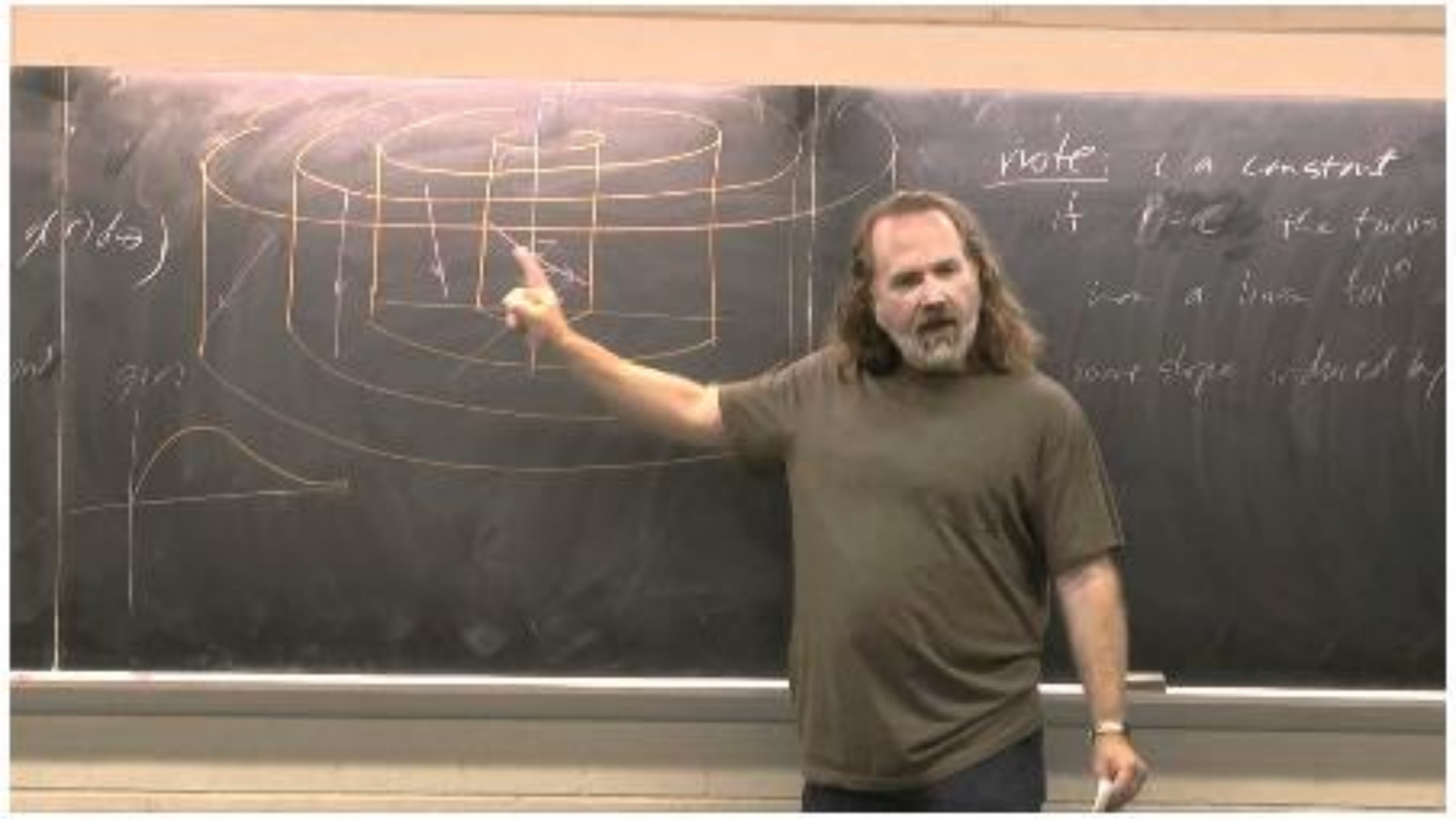


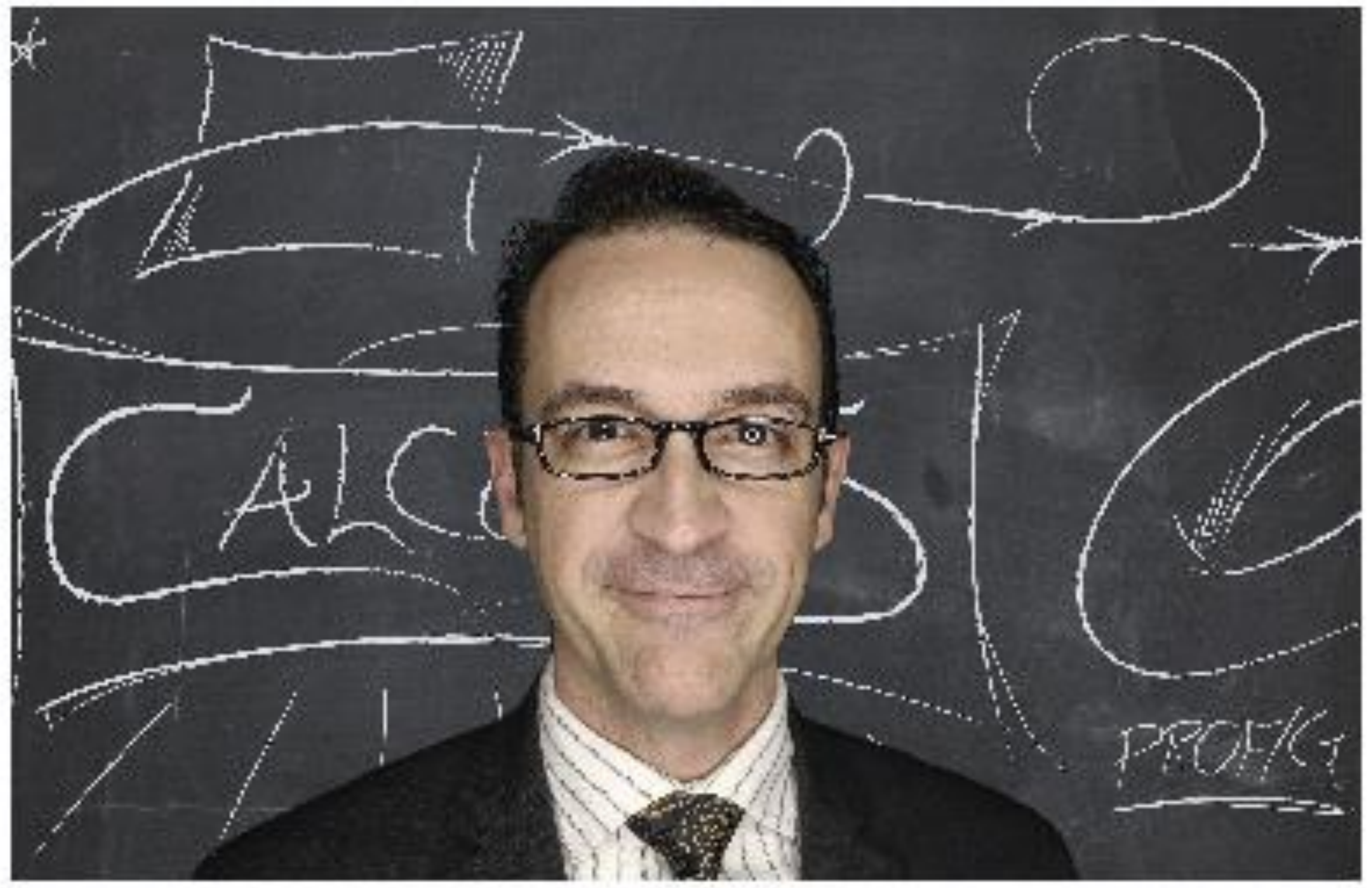


The magic mirror

In terms of $\alpha = \iota_X g$ and μ (volume form) the **stationary Euler equations** read

$$\begin{cases} \iota_X d\alpha = -dB \\ d\iota_X \mu = 0 \end{cases}$$





- Etnyre-Ghrist:
  {Rotational non singular Beltrami v.f.} $\rightleftarrows$ {Reeb v.f. reparametrized}

Main theorem 1

Theorem

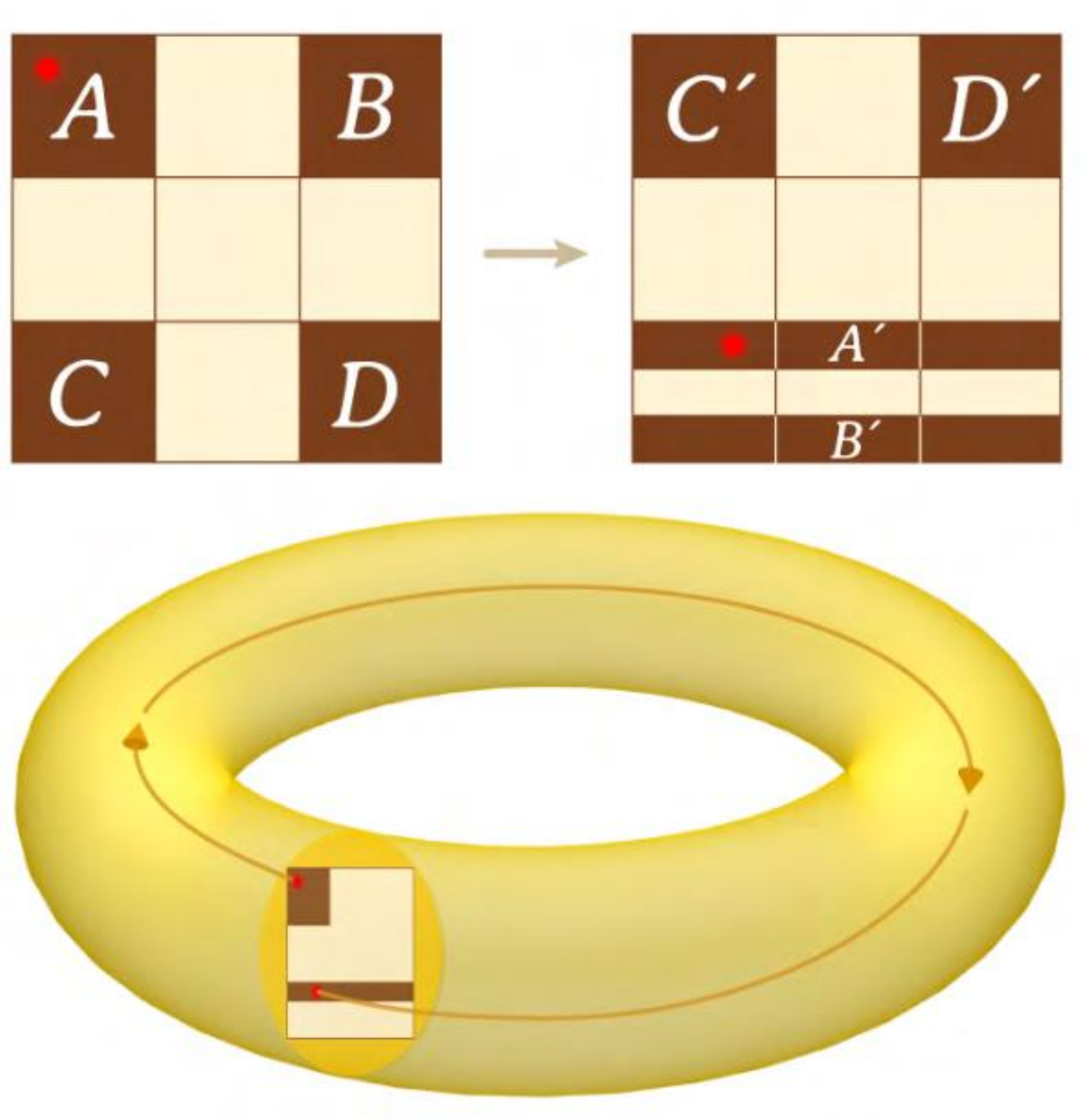
Any non-vanishing Beltrami field with positive proportionality factor is a reparametrization of a Reeb flow for some contact form. Conversely, any reparametrization of a Reeb vector field of a contact structure is a non-vanishing Beltrami field for some Riemannian metric.

A magic mirror

- **Weinstein conjecture for Reeb vector fields** ⇝ **periodic orbits for Beltrami vector fields**(Etnyre-Ghrist)
- **h-principle** ⇝ Reeb embeddings ⇝ **universality of Euler flows** (Cardona–M–Peralta-Salas–Presas)
- **Reeb suspension of area preserving diffeomorphism of the disc** ⇝ **Construction of universal 3D Turing machine** (Cardona–M–Peralta-Salas–Presas)
- Uhlenbeck's genericity properties of eigenfunctions of Laplacian ⇝ existence of escape trajectories (M–Oms–Peralta-Salas)

Constructing the Fluid Computer

From Moore to 3D



Moore: There is a block transformation of the Cantor square set onto itself that sends the red point on the left to the one on the right.

CMPP: Using this idea, one can construct a flow on a solid torus (below) in such a way that every time a particle passes through a transversal, it follows this block transformation.

This particle follows the trajectory of a Reeb field, and through the mirror, one can associate it with a solution of the Euler equation (fluid).

A fluid computer in dimension 3

Theorem 2 (Cardona, M., Peralta-Salas & Presas)

There exists an Eulerisable flow X in $\mathbb{S}^3$ that is Turing complete. The metric g that makes X a stationary solution of the Euler equations can be assumed to be the round metric in the complement of an embedded solid torus.

Turing, 1936: The halting problem is **undecidable**.

Corollary

There exist undecidable fluid particle paths: there is no algorithm to decide whether a trajectory will enter an open set or not in finite time.

Does this give finite-time blow-up for Navier-Stokes?

Short answer: No
Long answer: On a Riemannian 3-manifold (M, g) the Navier-Stokes read as

$$\begin{cases} \frac{\partial u}{\partial t} + \nabla_u u - \nu \Delta u = -\nabla p\,, \\ \operatorname{div} u = 0\,, \\ u(t=0) = u_0\,, \end{cases} \tag{1}$$

where $\nu > 0$ is the viscosity.

- Δ is the Hodge Laplacian (whose action on a vector field is defined as $\Delta u := (\Delta u^\flat)^\sharp$).
- The vector field X is of Beltrami type (with constant factor 1). When considered as an initial datum of NS, we obtain:

  $$X(t) = X\, e^{-\nu t}$$

  $\Longrightarrow$ it exists for all time.
- The exponential decay implies that it can simulate just a finite number of steps of any Turing machine.

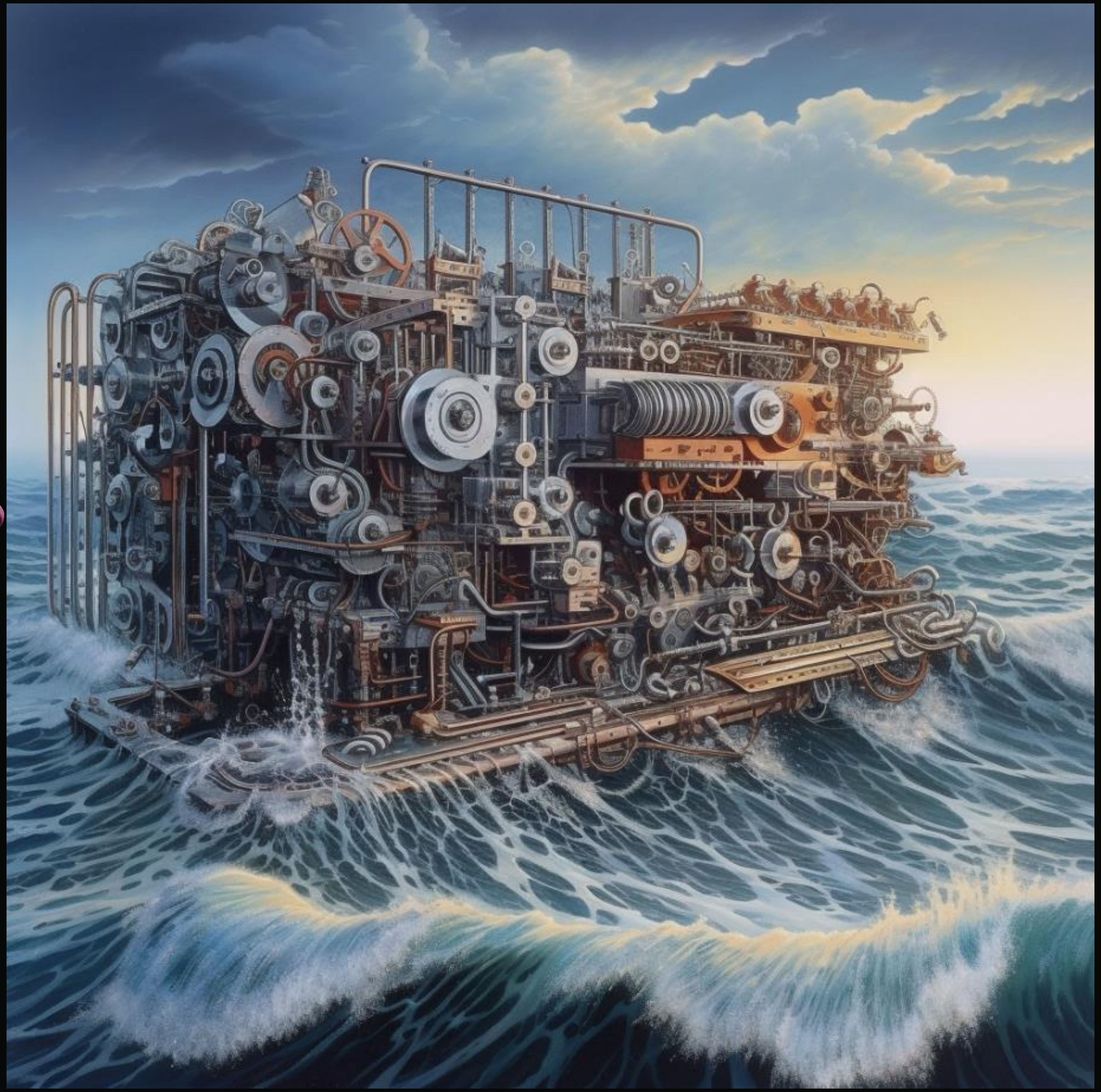
Conclusions

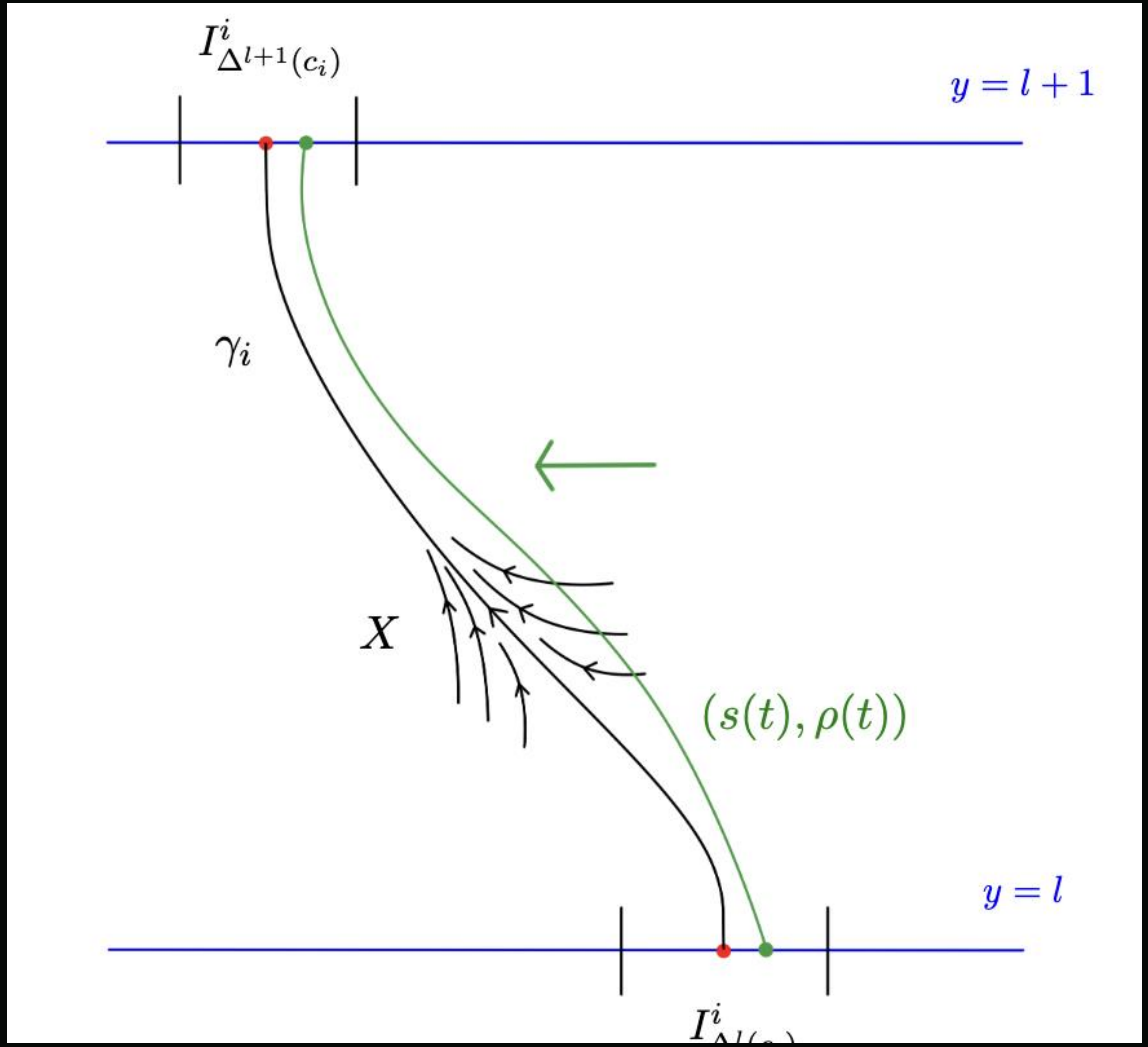
Can such techniques be applied to Navier-Stokes?

- Our result is enigmatic: For some systems, it is not possible to decide if particles will reach certain regions in space, no matter how potent the computational problem is. In other words, **the problem is not computable.**
- Our construction only works when the fluid does not have viscosity.
- A theorem in Computer Science by Olivier Bournez, Daniel Graça and Emmanuel Hainry show that it is not possible to construct Turing complete systems with finite energy which are robust by perturbations. In other words, **adding viscosity to the system can destruct the computational power.**

Can we do this better?

In our construction which uses the mirror the Euler equations depend strongly on the metric which is not the Euclidean metric inside a small solid torus on the 3-sphere. Can we choose an Euclidean metric everywhere?





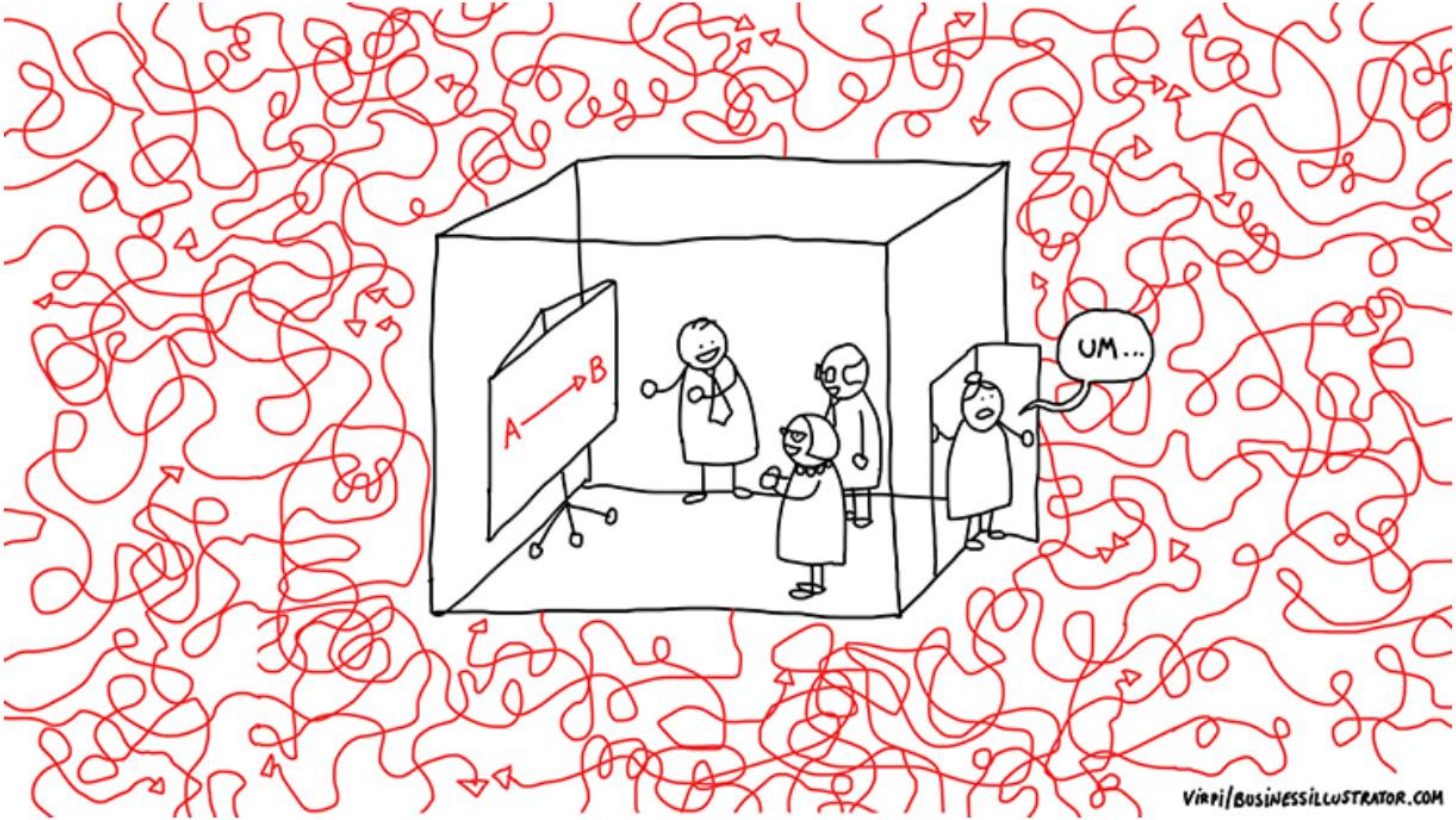
The Euclidean case

Theorem (Cardona, M., Peralta-Salas, 2021)

There exists a Beltrami vector field on $\mathbb{R}^3$ which is Turing complete.

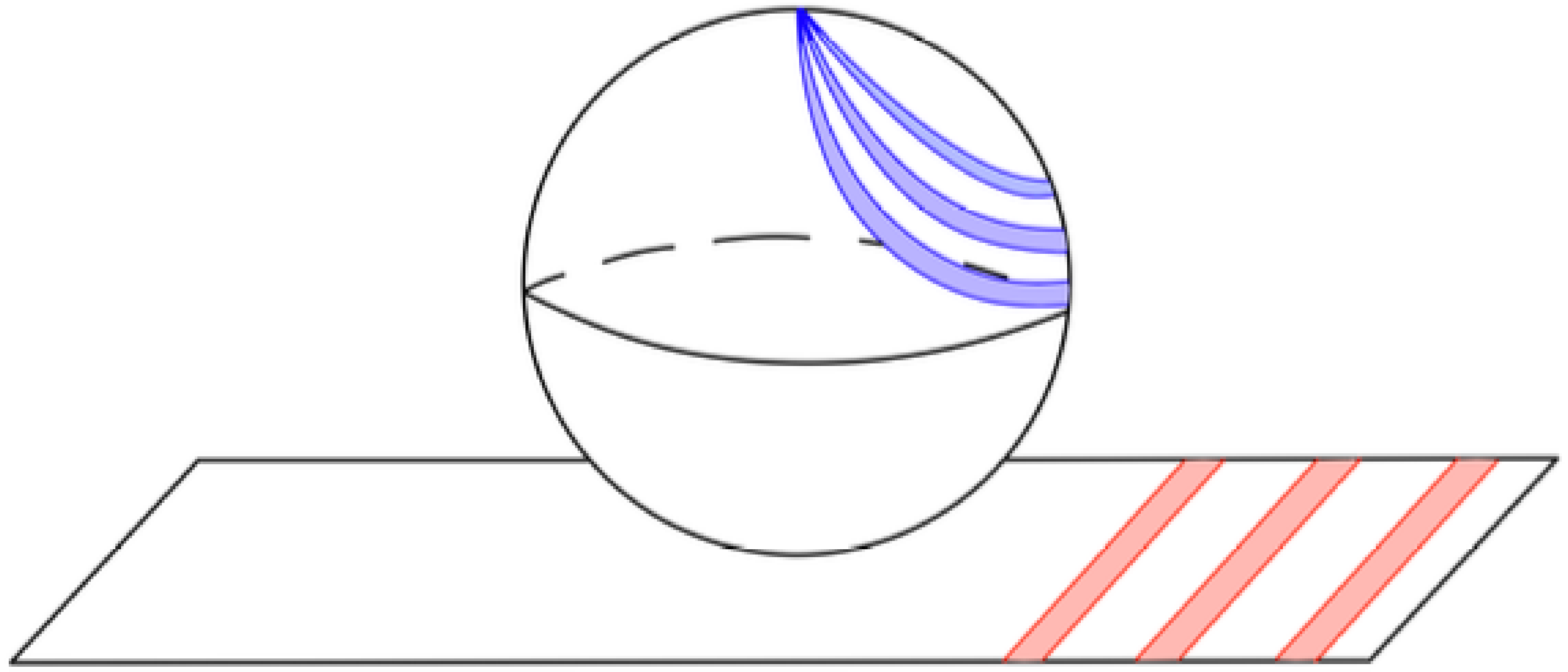
- This vector field does not have finite energy.
- The vector field has an invariant plane where all the computations of the machine take place.
- The computational power of this machine is weakly robust. It persists when the vector field is perturbed with an error with exponential decay.
- The proof is not geometrical: It requires a Cauchy-Kovalevskaya theorem and techniques of the theory of dynamical systems of gradient type.
- This construction has compact approximations which are Turing complete (at $\mathbb{T}^3$) with tapes of finite length.
- It is generic: The Turing completeness occurs with probability 1 for arbitrary Beltrami vector fields.

Outside the Beltrami box



Outside the Beltrami box

The Euler equations on (M, g) are Turing complete if: for any integer $k \geqslant 0$, given a Turing machine T, an input tape t, and a finite string $(t^*_{-k}, ..., t^*_k)$ of symbols of the alphabet, there exist an explicitly constructible vector field $X_0 \in \mathfrak{X}^\infty_{vol}(M)$ and an open set $U \subset \mathfrak{X}^\infty_{vol}(M)$ such that the solution to the Euler equations with initial datum X_0 is defined for all time and intersects U if and only if T halts with an output tape whose positions $-k, ..., k$ correspond to the symbols $t^*_{-k}, ..., t^*_k$.



The manifold

The manifold M is diffeomorphic to $SO(N) \times \mathbb{T}^{N+1}$ and $\dim(M) \lesssim 10^{35}$.

Theorem 4 (Cardona, M., & Peralta-Salas)

There exists a smooth compact Riemannian manifold (M, g) such that the Euler equations on (M, g) are Turing complete. In particular, the problem of whether the solution to the Euler equations with an initial datum X_0 will reach a certain open set $U \subset \mathfrak{X}^\infty_{vol}(M)$ or not is undecidable.

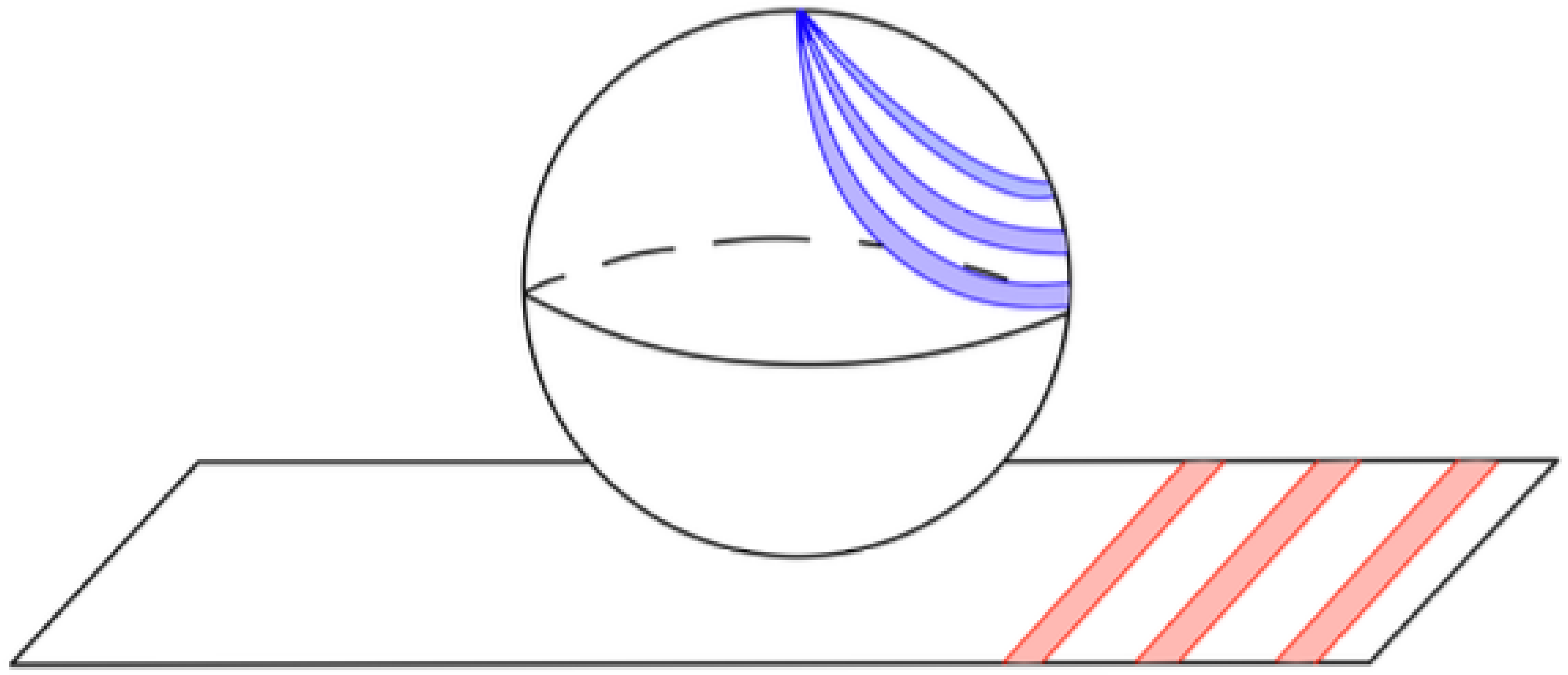
Proof

- There exist polynomial vector fields which are Turing complete on a sphere. Idea: *We compactify* a proof by *Graça et al* on $\mathbb{R}^n$ and we regularize it to get global smooth vector fields.
- Recall:

Theorem (Torres de Lizaur)

Given a polynomial vector field Y on $\mathbb{S}^n$. There exists a Riemannian manifold (M, g) such that $(\mathbb{S}^n, Y)$ can be embedded as Euler equations on (M, g).

- Combine to conclude.



The manifold

The manifold M is diffeomorphic to $SO(N) \times \mathbb{T}^{N+1}$ and $\dim(M) \lesssim 10^{35}$.

New ideas

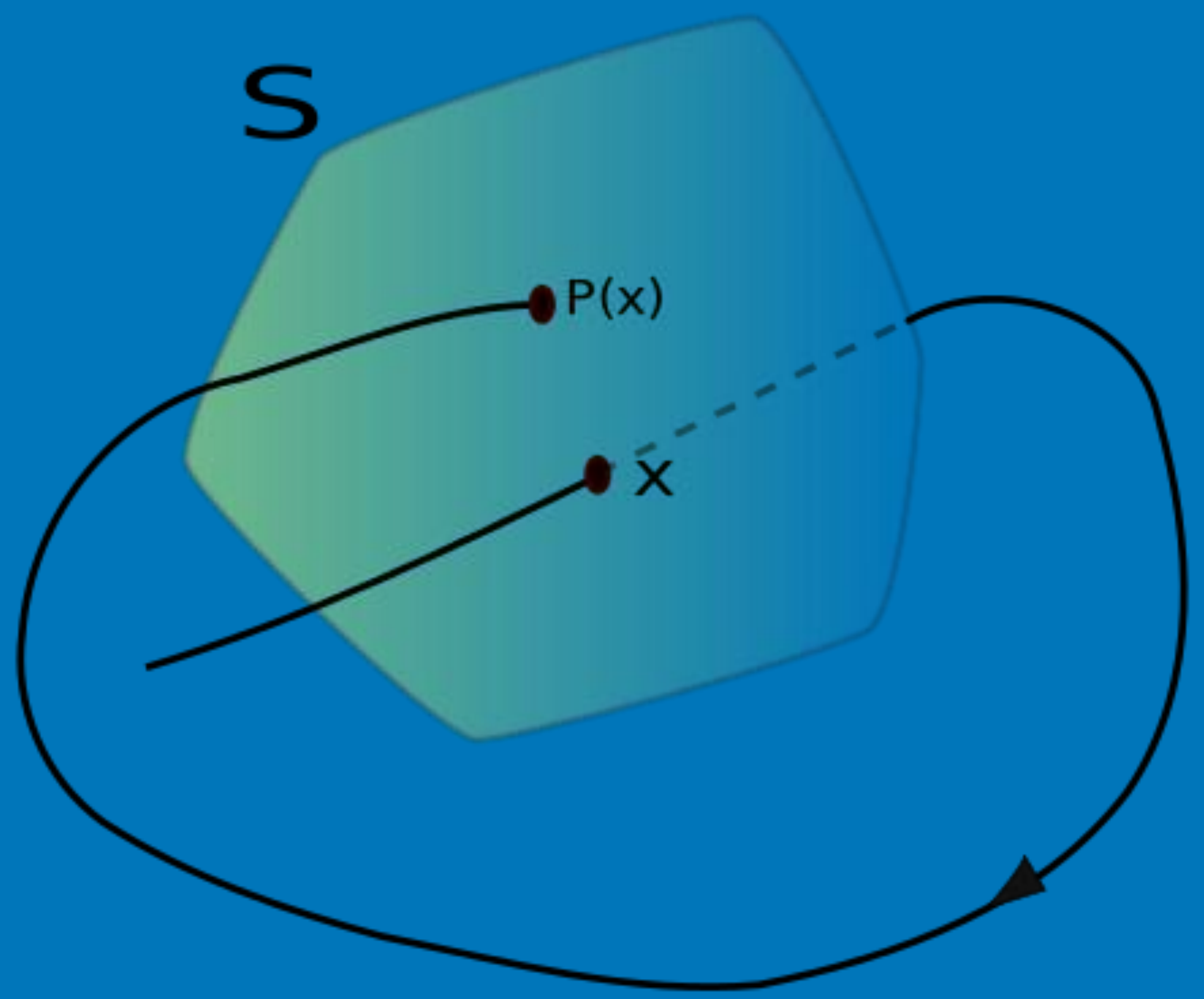
Reflecting the stars on the sea…

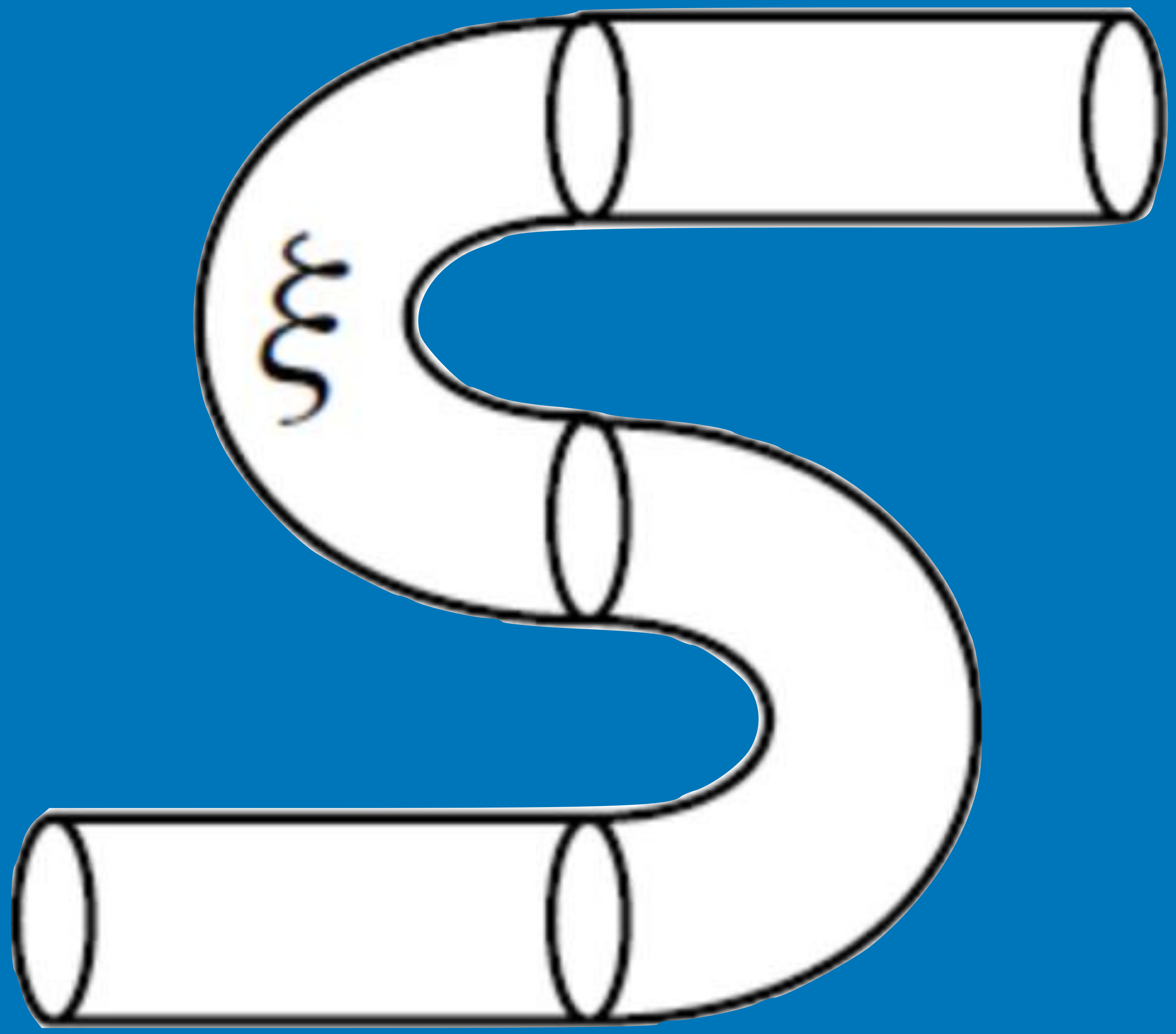
Arnold's dream of establishing a connection between the dynamical complexity of celestial mechanics and of stationary solutions of hydrodynamics:

"Car les écoulements avec **curl v = λ v** ***admettent, probablement, des lignes de courant avec une topologie aussi compliquée que celle des orbites en mécanique céleste."***

New ideas

Fluid computers à la Feynman





Thank you!

Merci!