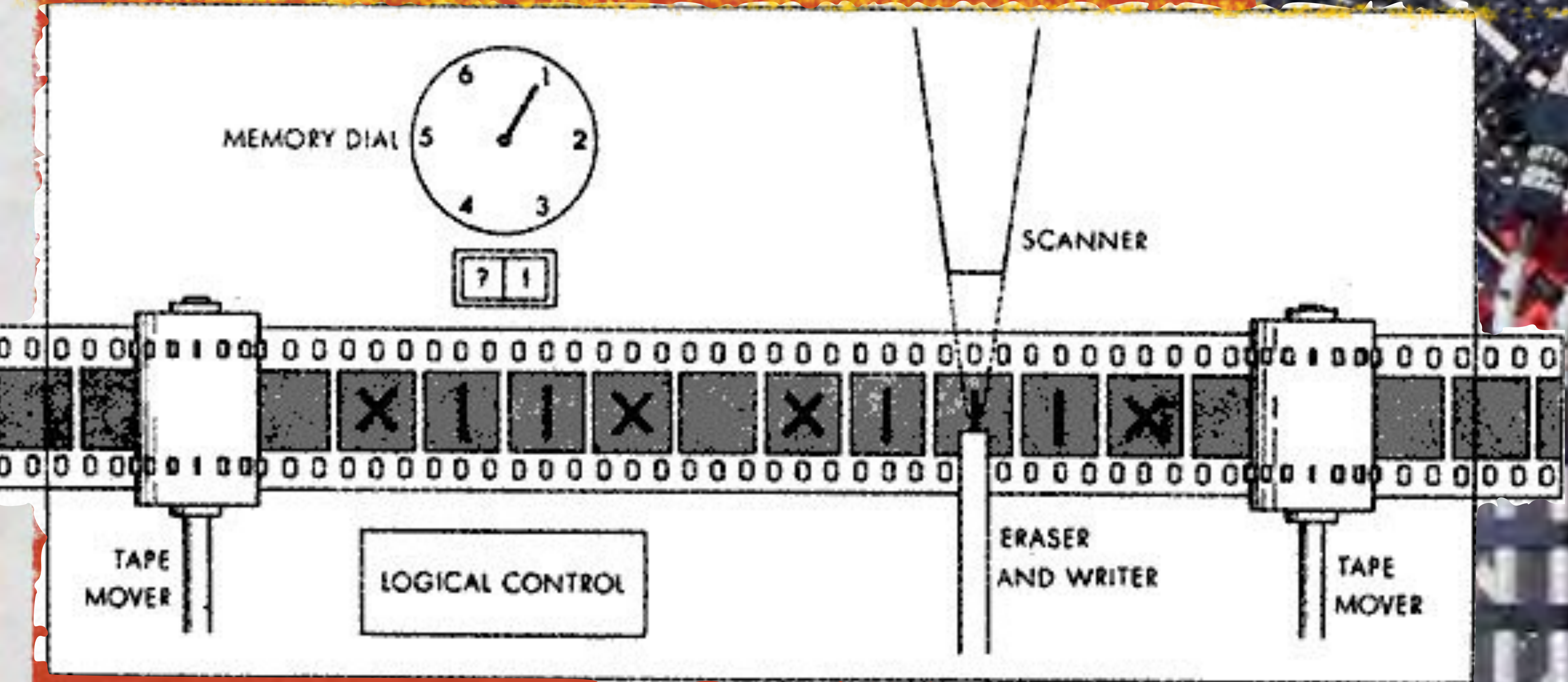
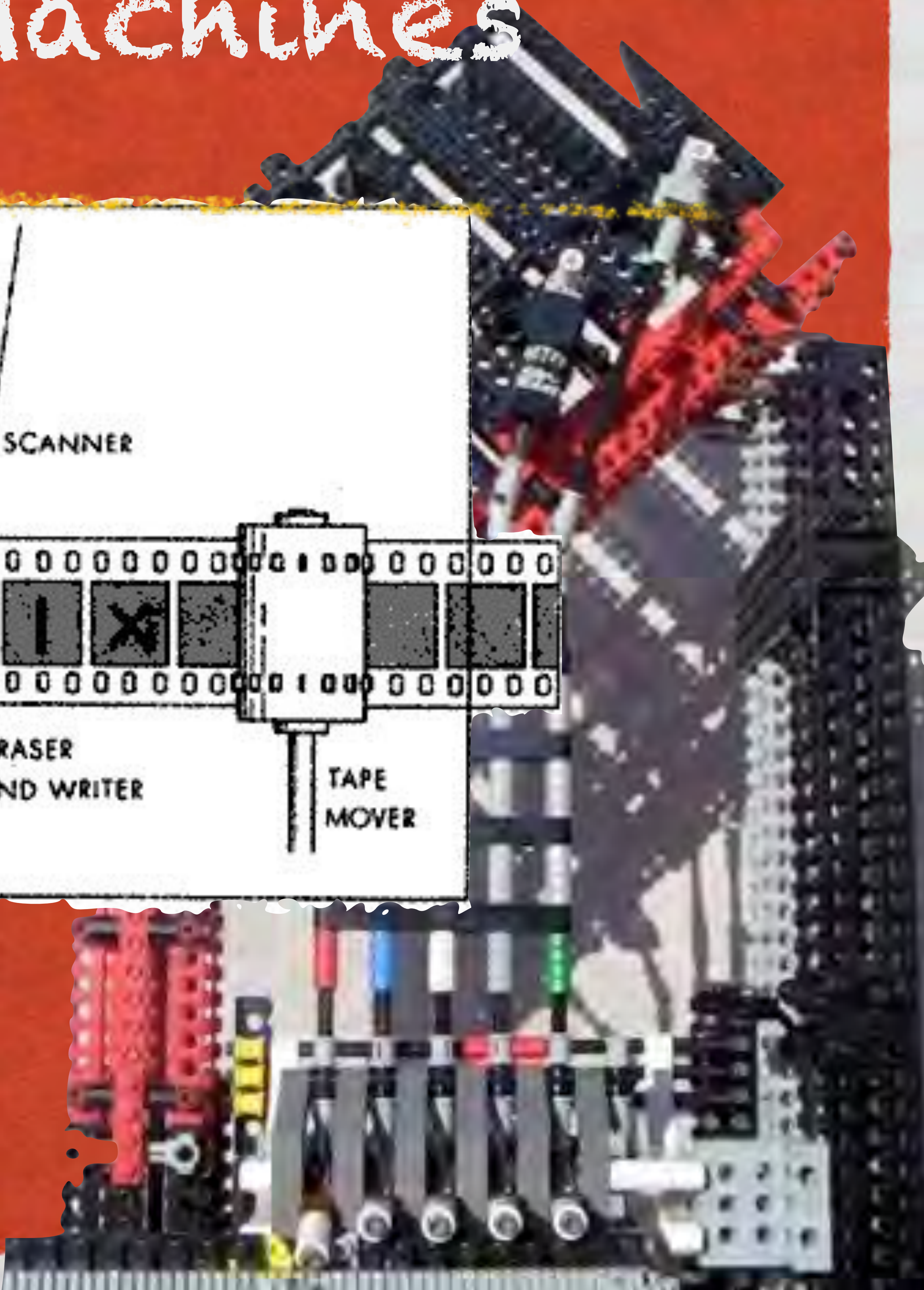
# Analog Models

## Alan Turing (1936)

A man provided with paper, pencil, and rubber, and subject to strict discipline, is in effect a universal machine

# Turing's Machines





## Turing [1948]

- The property of being ‘discrete’ is only an advantage for the theoretical investigator, and serves no evolutionary purpose, so we could not expect Nature to assist us by producing truly ‘discrete’ brains.

## Turing [1948]

- All machinery can be regarded as continuous, but when it is possible to regard it as discrete it is usually best to do so.

## Turing's Premises

- Sequential (discrete) symbol manipulation
- Deterministic
- Finite internal states
- Finite symbol space
- Finite observability and local action
- Linear external memory suffices

# Digital vs. Analog

## Turing [1950]

- It is true that a discrete-state machine must be different from a continuous machine. But if we adhere to the conditions of the imitation game, the interrogator will not be able to take any advantage of this difference.

# Analog

- Analog space

- Analog time

## Analog Space

- Uncountably many possible values

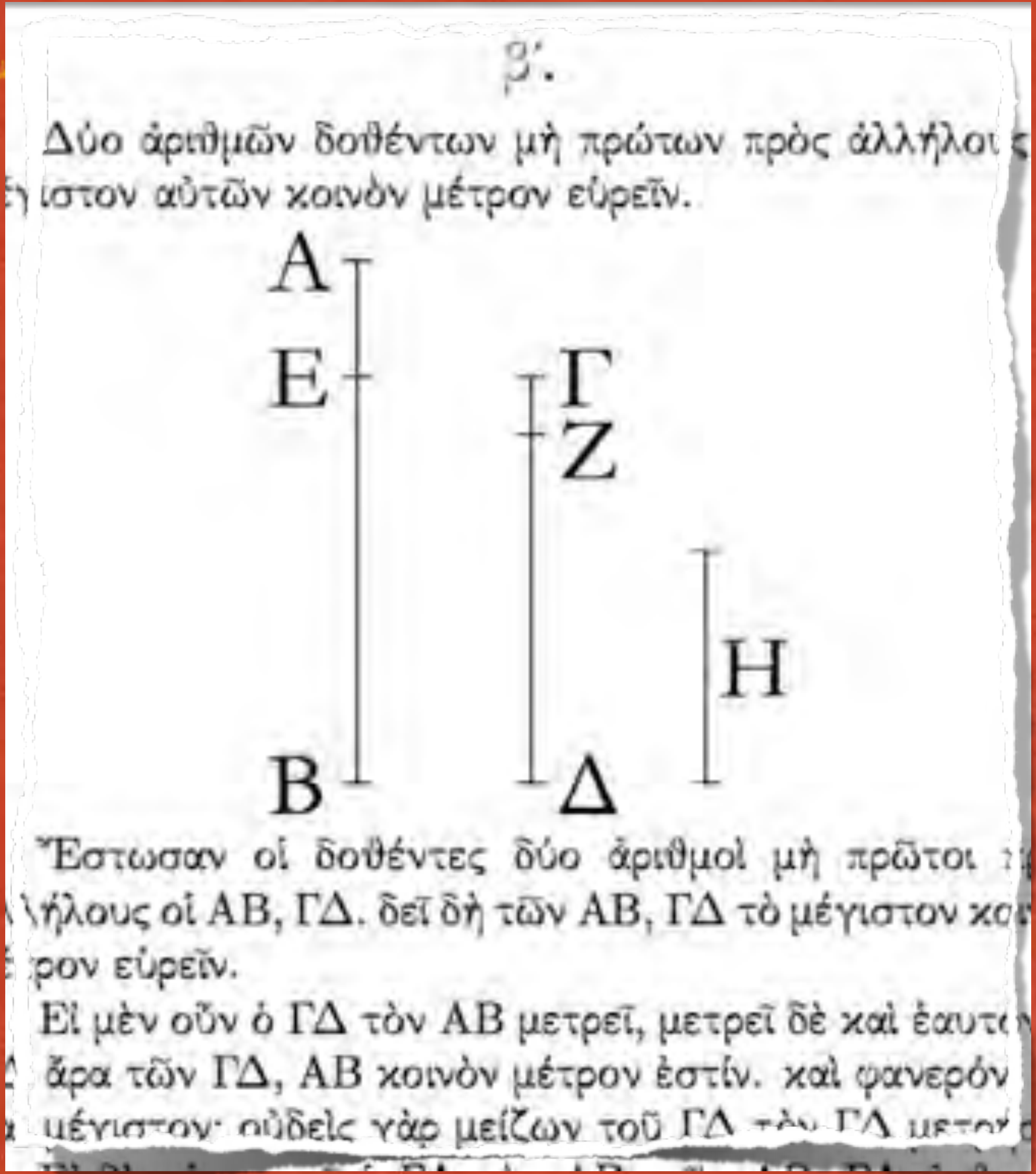
# Euclid (c. -300)

Euclid's GCD algorithm was formulated geometrically: Find common measure for 2 lines.

Used repeated subtraction of the shorter segment from the longer.

# Euclid's Elements

- Finitely describable – in terms of basic compass operations



β'.

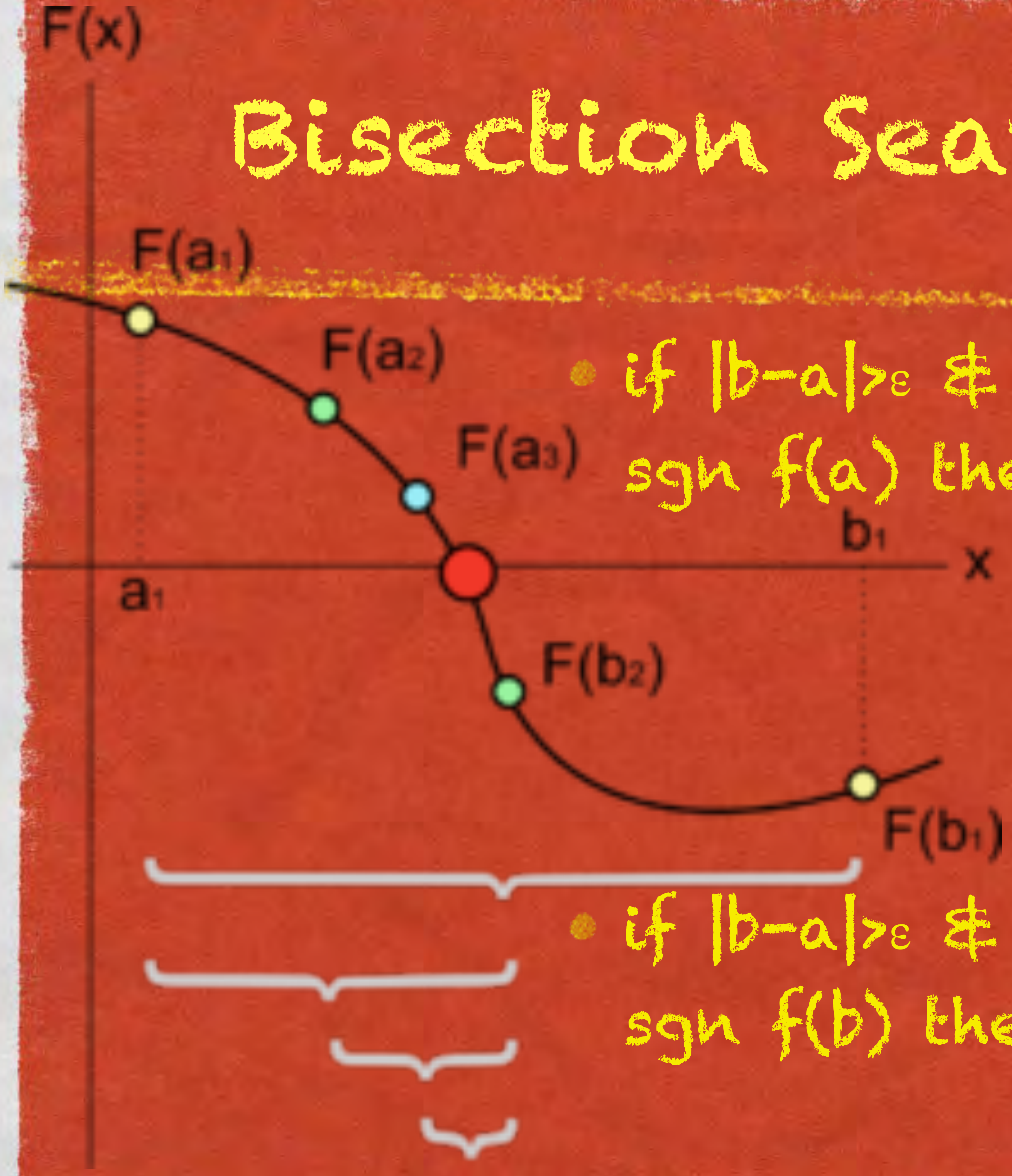
Δύο ἀριθμῶν δοθέντων μὴ πρώτων πρὸς ἀλλήλους
ιστον αὐτῶν κοινὸν μέτρον εὑρεῖν.

Ἔστωσαν οἱ δοθέντες δύο ἀριθμοὶ μὴ πρῶτοι
λήλους οἱ ΑΒ, ΓΔ. δεῖ δὴ τῶν ΑΒ, ΓΔ τὸ μέγιστον κοι
ρον εὑρεῖν.
Εἰ μὲν οὖν ὁ ΓΔ τὸν ΑΒ μετρεῖ, μετρεῖ δὲ καὶ ἑαυτ
ἄρα τῶν ΓΔ, ΑΒ κοινὸν μέτρον ἐστίν. καὶ φανερόν
μέγιστον· οὐδεὶς γὰρ μείζων τοῦ ΓΔ τὸν ΓΔ μετρ

## Turing [1950]

- A small error in the information about the size of a nervous impulse impinging on a neuron, may make a large difference to the size of the outgoing impulse.

# Bisection Search



- if $|b-a|>\varepsilon$ & sgn $f((a+b)/2)$ = sgn $f(a)$ then $a := (a+b)/2$
- if $|b-a|>\varepsilon$ & sgn $f((a+b)/2)$ = sgn $f(b)$ then $b := (a+b)/2$

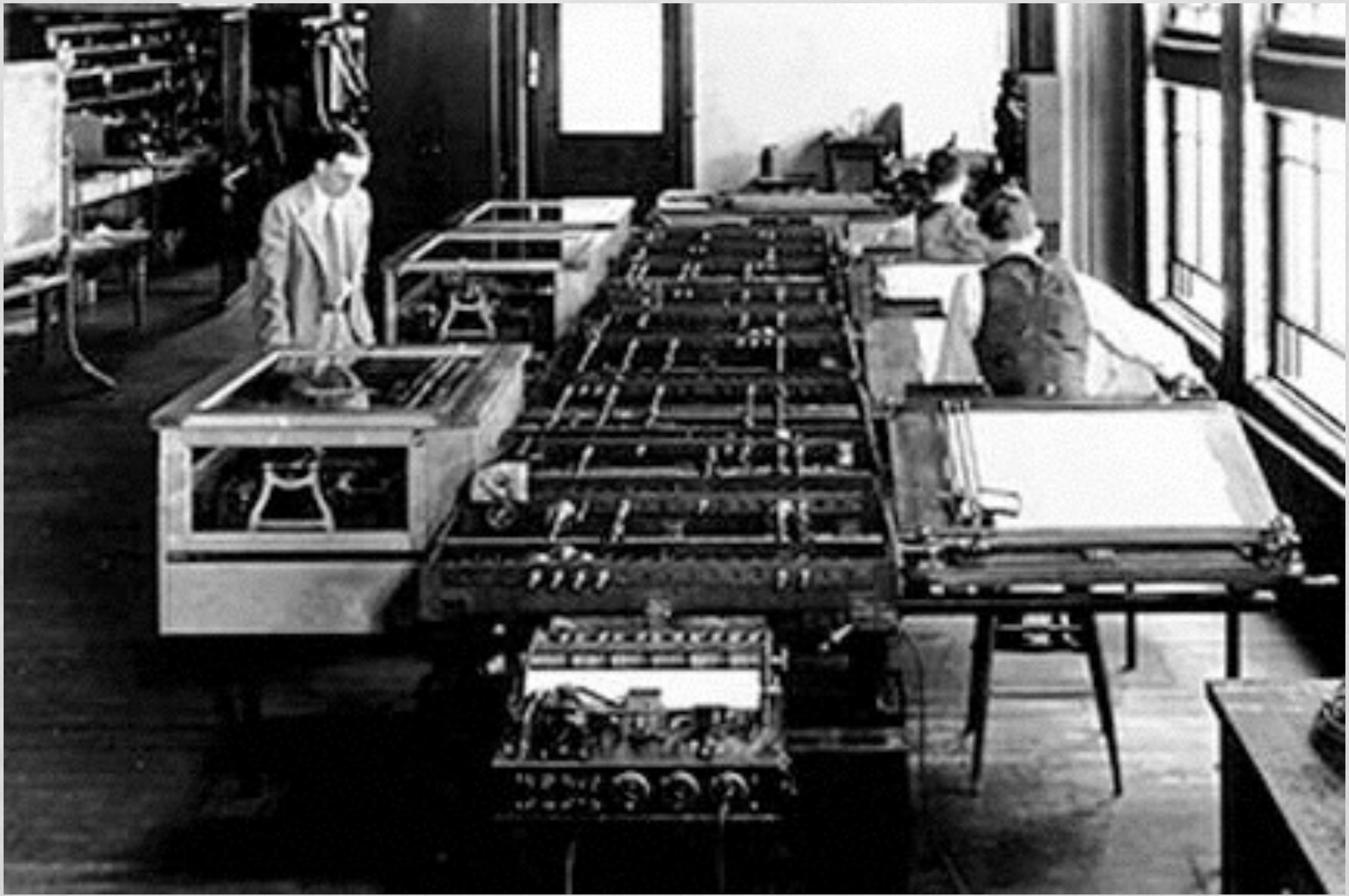
# Bisection Search (Saul Gorn)

- Although [this procedure is] among the slowest, it is applicable to any continuous function.  The fact that no differentiability conditions have to be checked makes it ...  an 'old work-horse'.

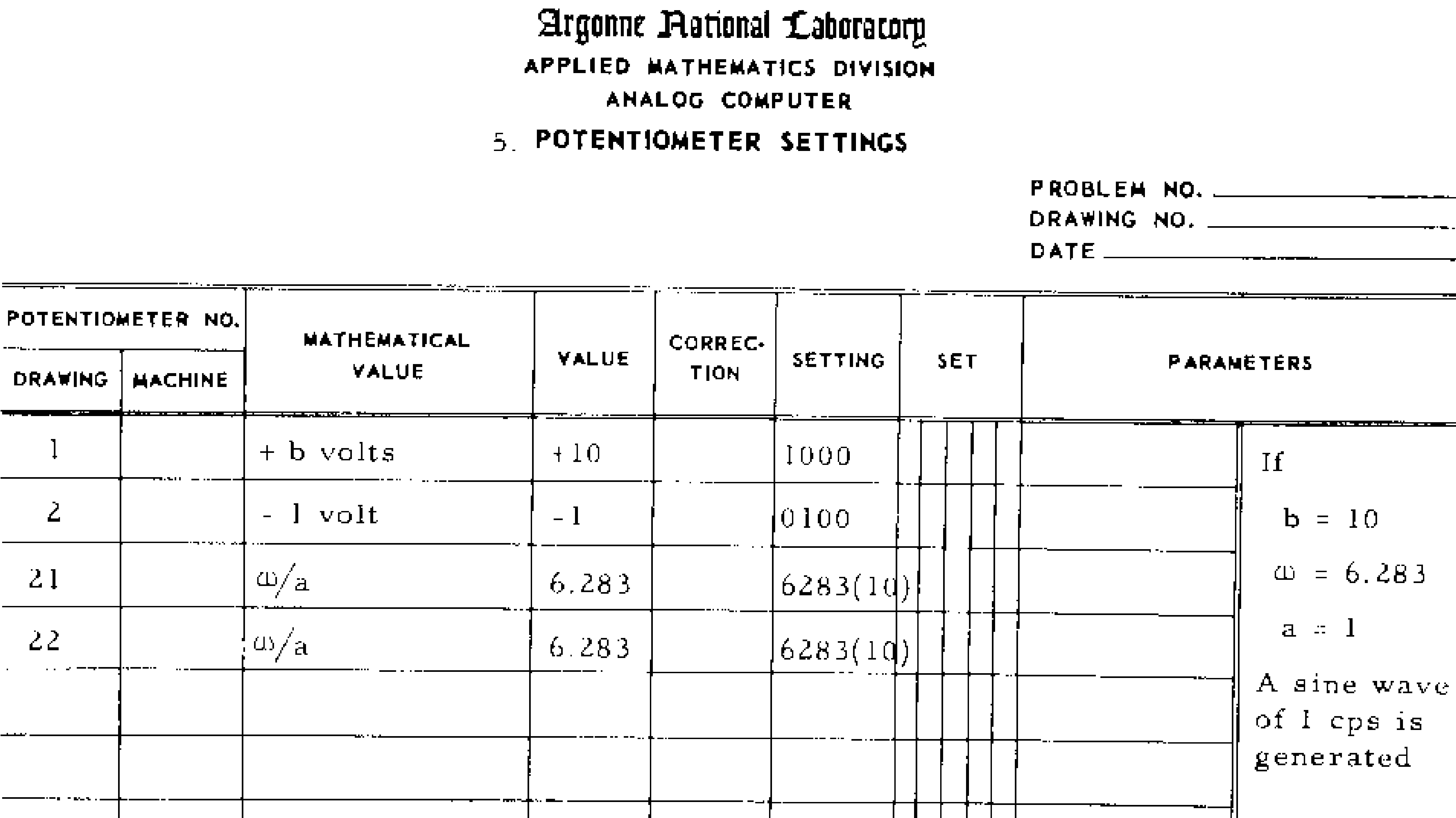
## Continuous Space

- Idealized
- Computable reals
- Intervals
- Arbitrary precision

## Vannevar Bush's Differential Analyser



# Analog Programs



Argonne National Laboratory

APPLIED MATHEMATICS DIVISION

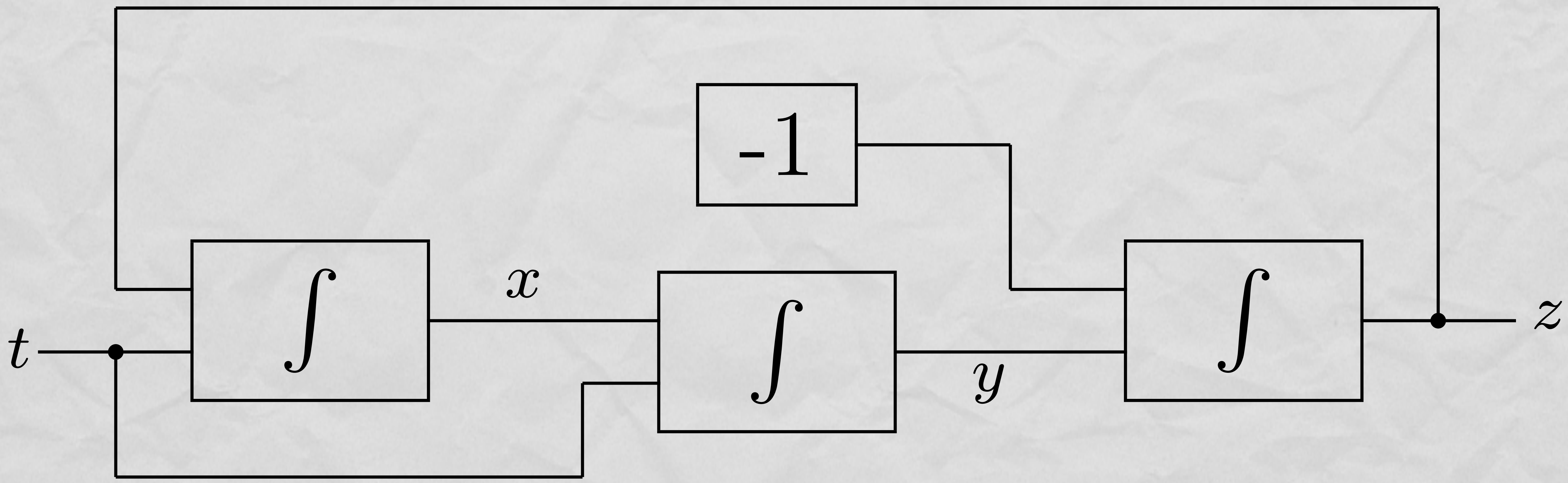
ANALOG COMPUTER

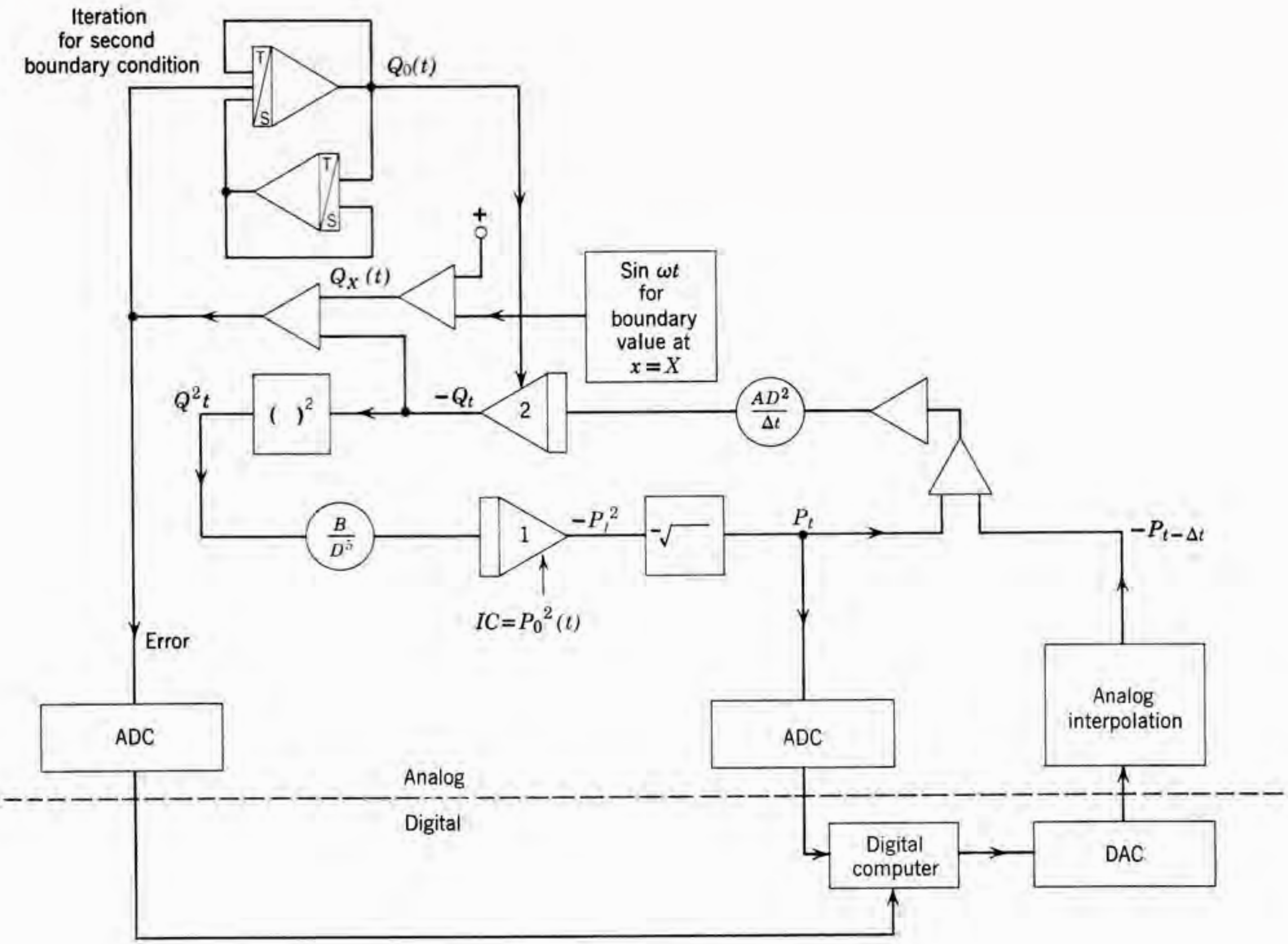
5. POTENTIOMETER SETTINGS

PROBLEM NO. __________
DRAWING NO. __________
DATE __________

| POTENTIOMETER NO. | | MATHEMATICAL VALUE | VALUE | CORREC-TION | SETTING | SET | PARAMETERS |
|---|---|---|---|---|---|---|---|
| DRAWING | MACHINE | | | | | | |
| 1 | | + b volts | +10 | | 1000 | | If |
| 2 | | - 1 volt | -1 | | 0100 | | b = 10 |
| 21 | | $\omega/a$ | 6.283 | | 6283(10) | | $\omega$ = 6.283 |
| 22 | | $\omega/a$ | 6.283 | | 6283(10) | | a = 1 |
| | | | | | | | A sine wave of 1 cps is generated |
| | | | | | | | |

# General-Purpose Analog Computer




Iteration for second boundary condition
T
S
$Q_0(t)$
T
S
+
$Q_X(t)$
Sin $\omega t$ for boundary value at $x = X$
$Q^2 t$
$(\ )^2$
$-Q_t$
2
$\frac{AD^2}{\Delta t}$
$\frac{B}{D^5}$
1
$-P_t^2$
$-\sqrt{\ }$
$P_t$
$-P_{t-\Delta t}$
$IC = P_0^2(t)$
Error
ADC
ADC
Analog interpolation
Analog
Digital
Digital computer
DAC

## Hybrid Computers

# Analog Time

- State evolves as time progresses
- Time is dense or continuous

## Turing [1948]

- The states of `continuous' machinery ... form a continuous manifold, and the behaviour of the machine is described by a curve on this manifold.

## Baron Kelvin's Tide Predictor

# Britt Phillips` Water Computer

# Toy Problem: Mortar

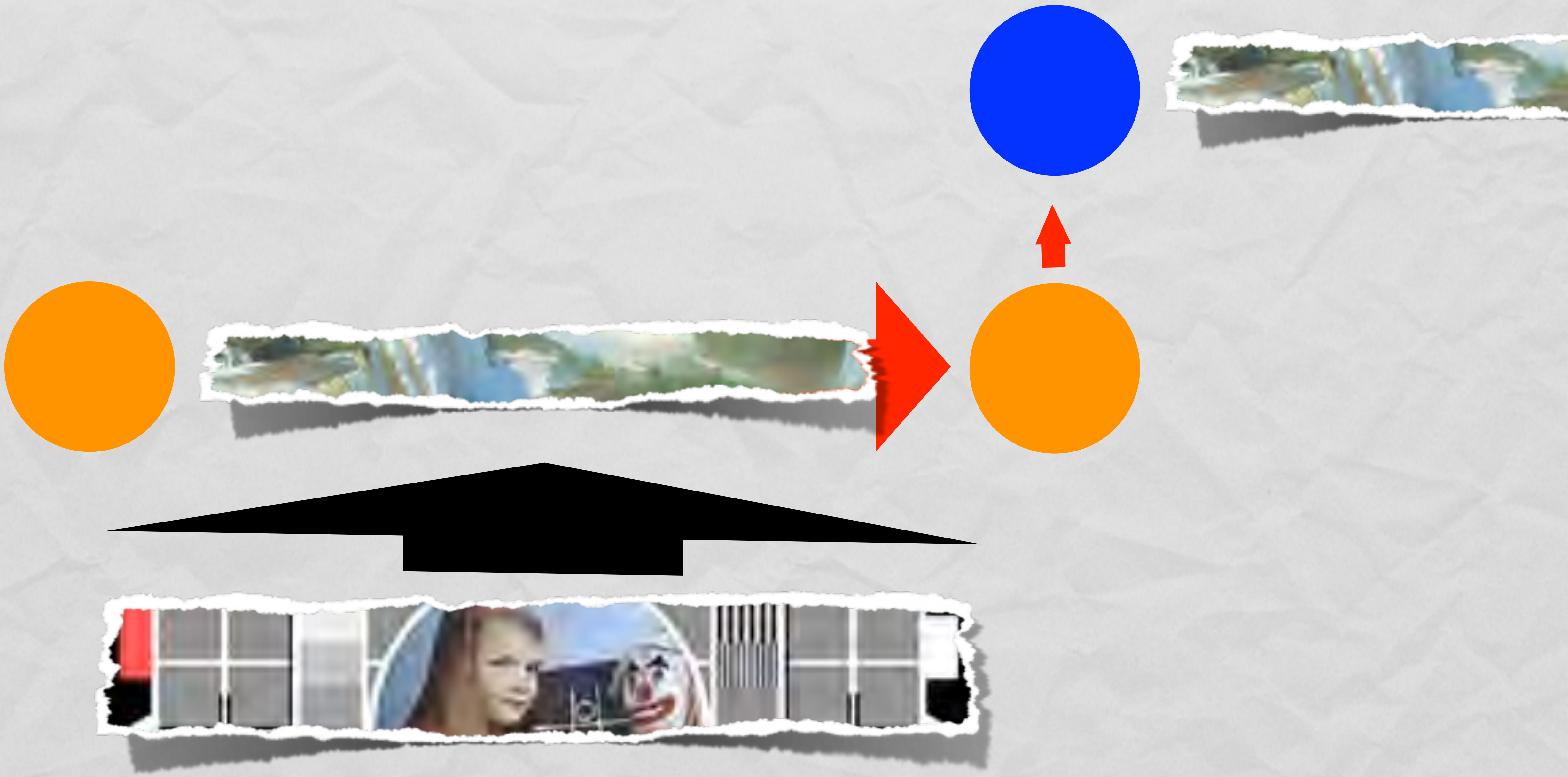
- $t$ time signal
- $g, a, s$ inputs

$x := t \cdot s \cdot \cos a$

$y := t \cdot s \cdot \sin a$

$- \frac{1}{2} \cdot g \cdot t^2$

# Flow & Jump



## Flows

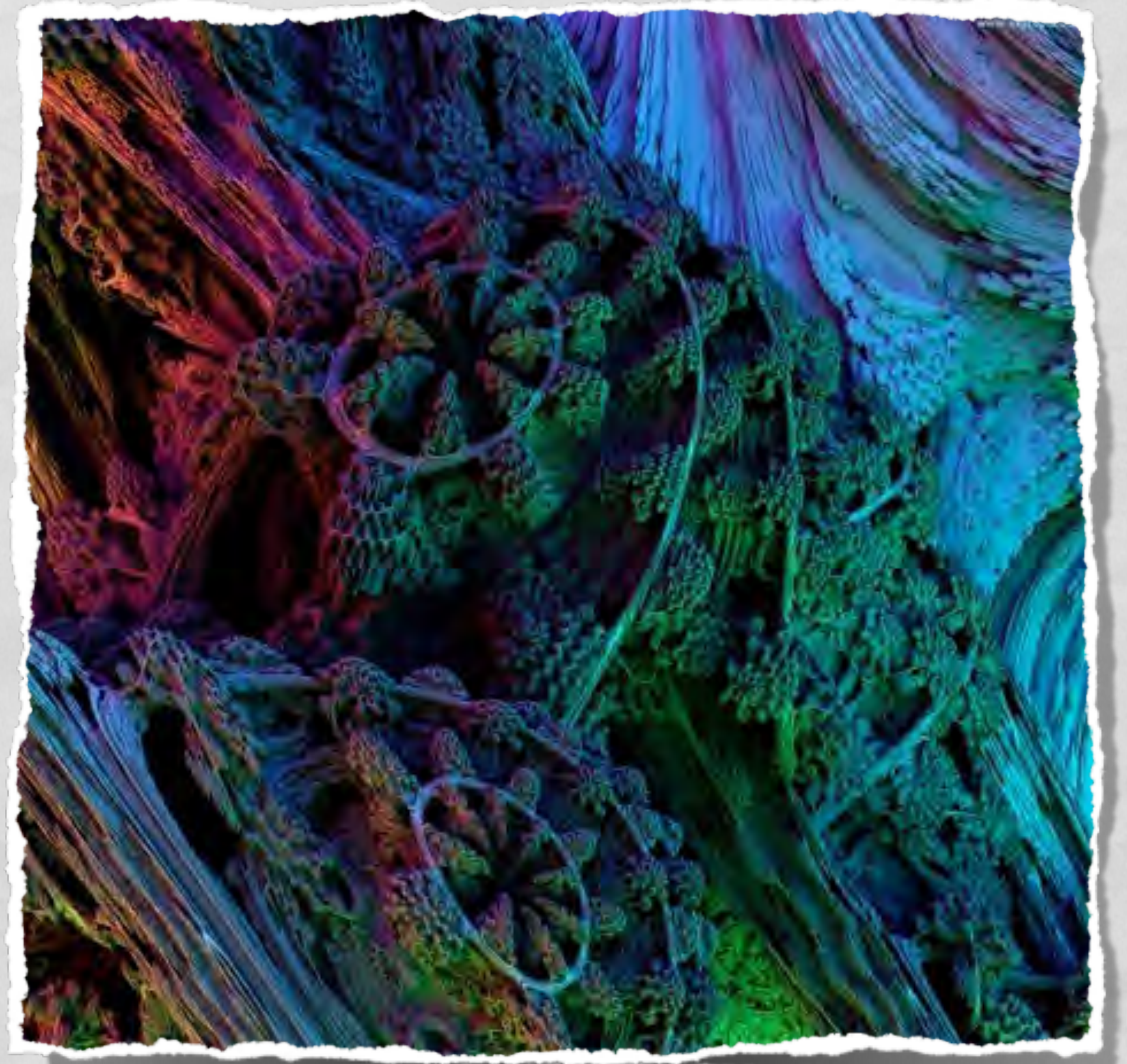
- Fixed dynamics over stretch of time

# Jumps

- Change of dynamics
- Shouldnt happen too often

## Sparse Jumps

- Dynamics change only finitely often in any finite trajectory



## Flows

- Fixed dynamics over stretch of time
- If input wouldnt change, nothing would
- Critical equalities unchanging

A discrete algorithm is a discrete process whose evolution has a finite description

An analog algorithm is a continuous process whose evolution has a finite description

## Algorithm

I. An algorithm is a state-transition system

II. Logical structures capture salient aspects of states

III. The transition relation can be described finitely

## Simple

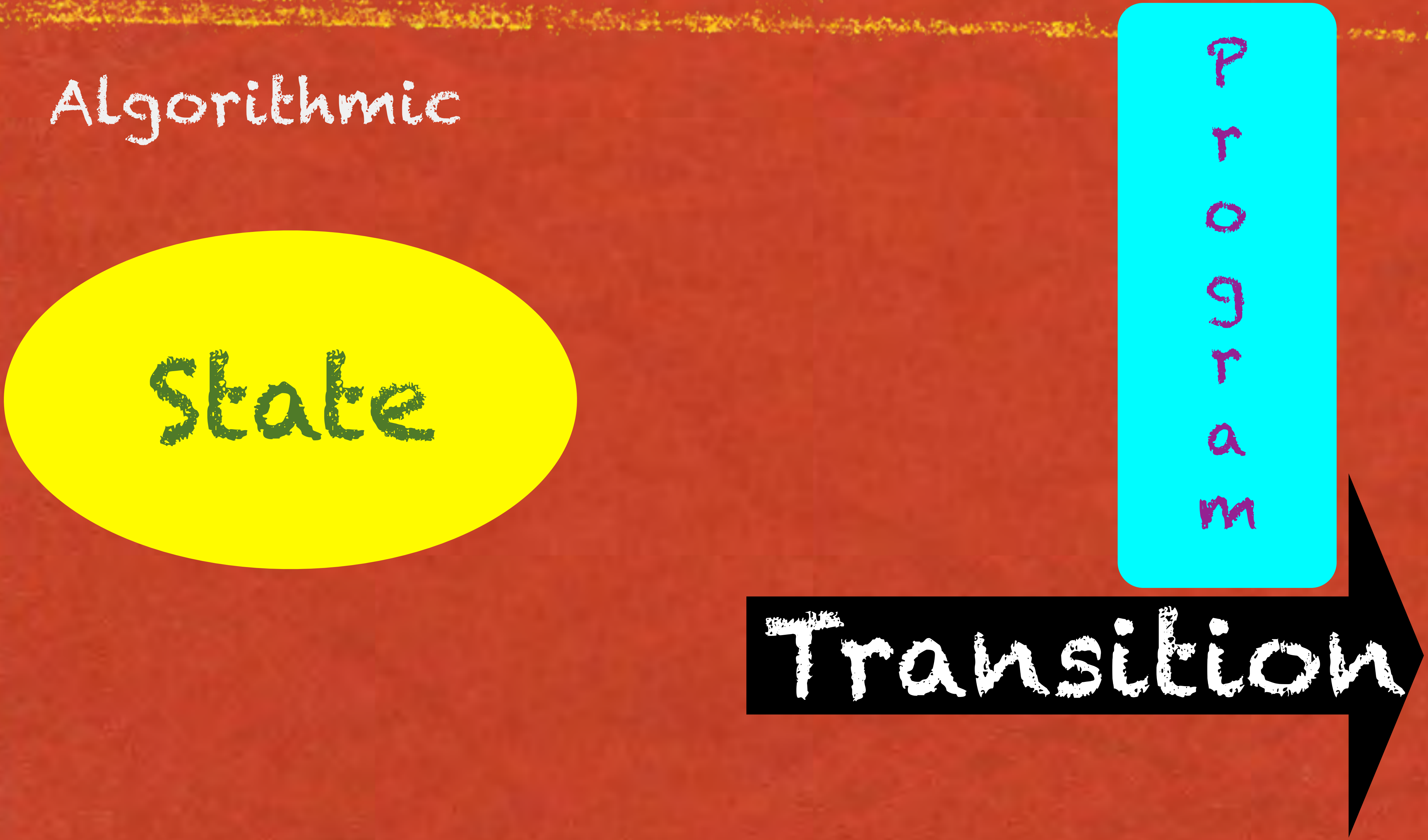
- No input signal other than time
- Explicit (solved) equations
- Ignore output or interaction

## Transition System

State
Transition

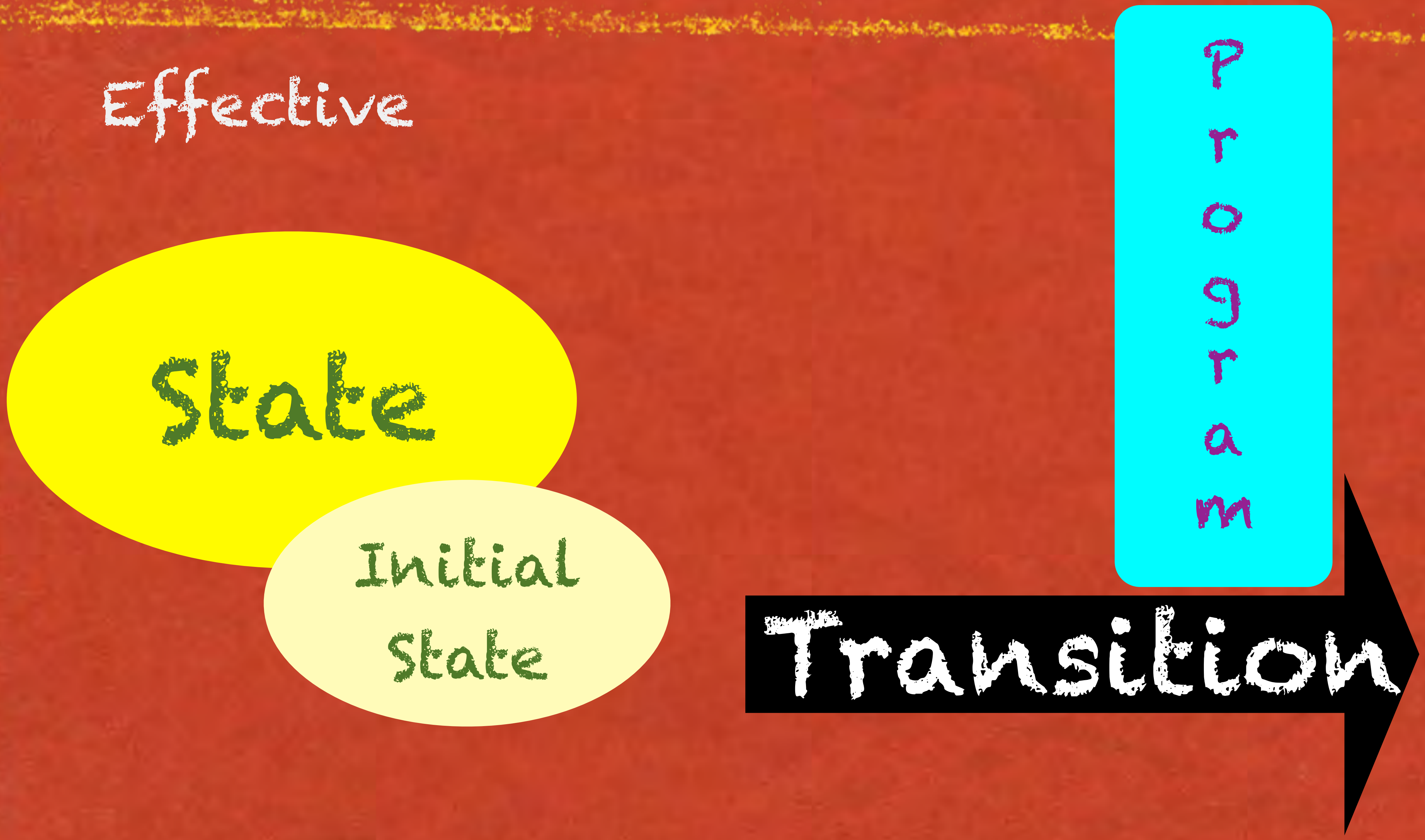
# Transition System

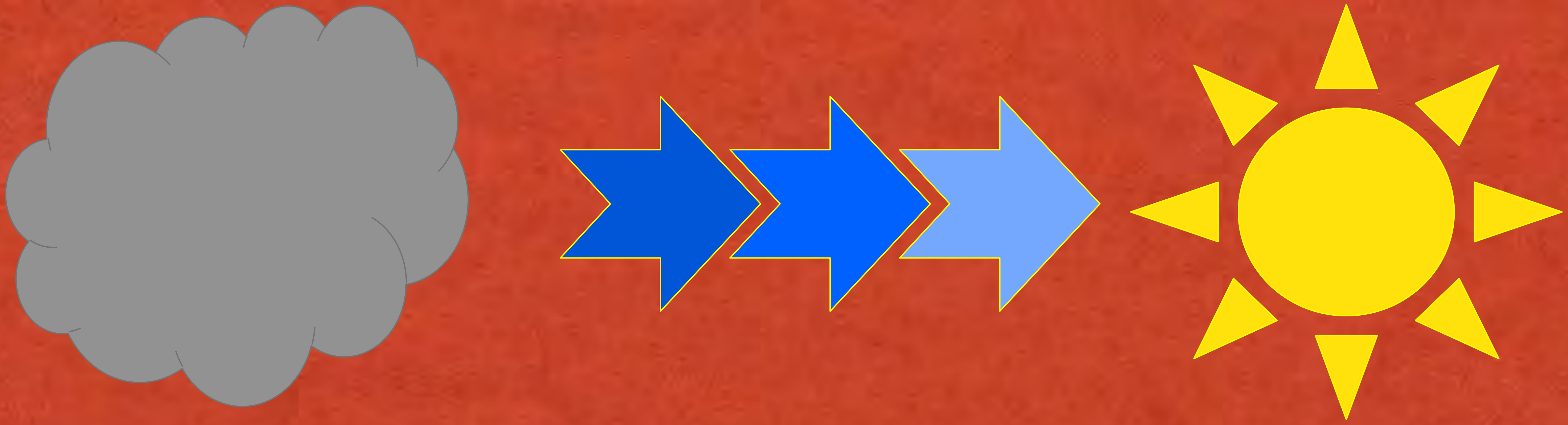
Algorithmic



# Transition System

Effective

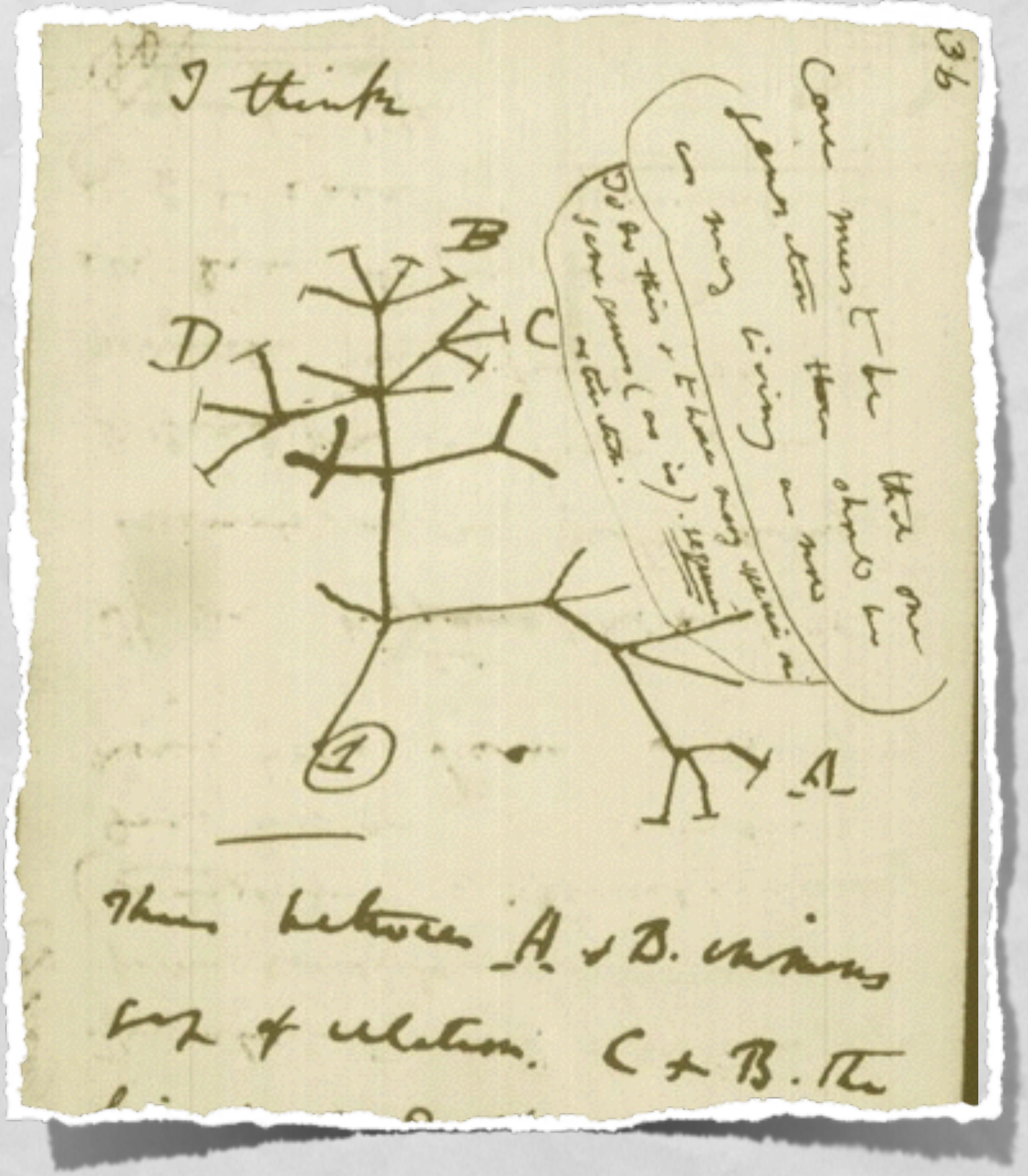




I

# Timelines

- Sequential/ Branching
- Discrete/ Continuous
- Dense/Sparse
- Finite/Infinite/ Transfinite



Case must be that one generation then should be as many living as now

To do this & to have many species in same genus (as is). requires extinction.

Thus between A & B. immens gap of relation. C & B. The

## Hartley Rogers, Jr.

For any given input, the computation
is carried out in a discrete
stepwise fashion, without use
of continuous methods or
analogue devices.

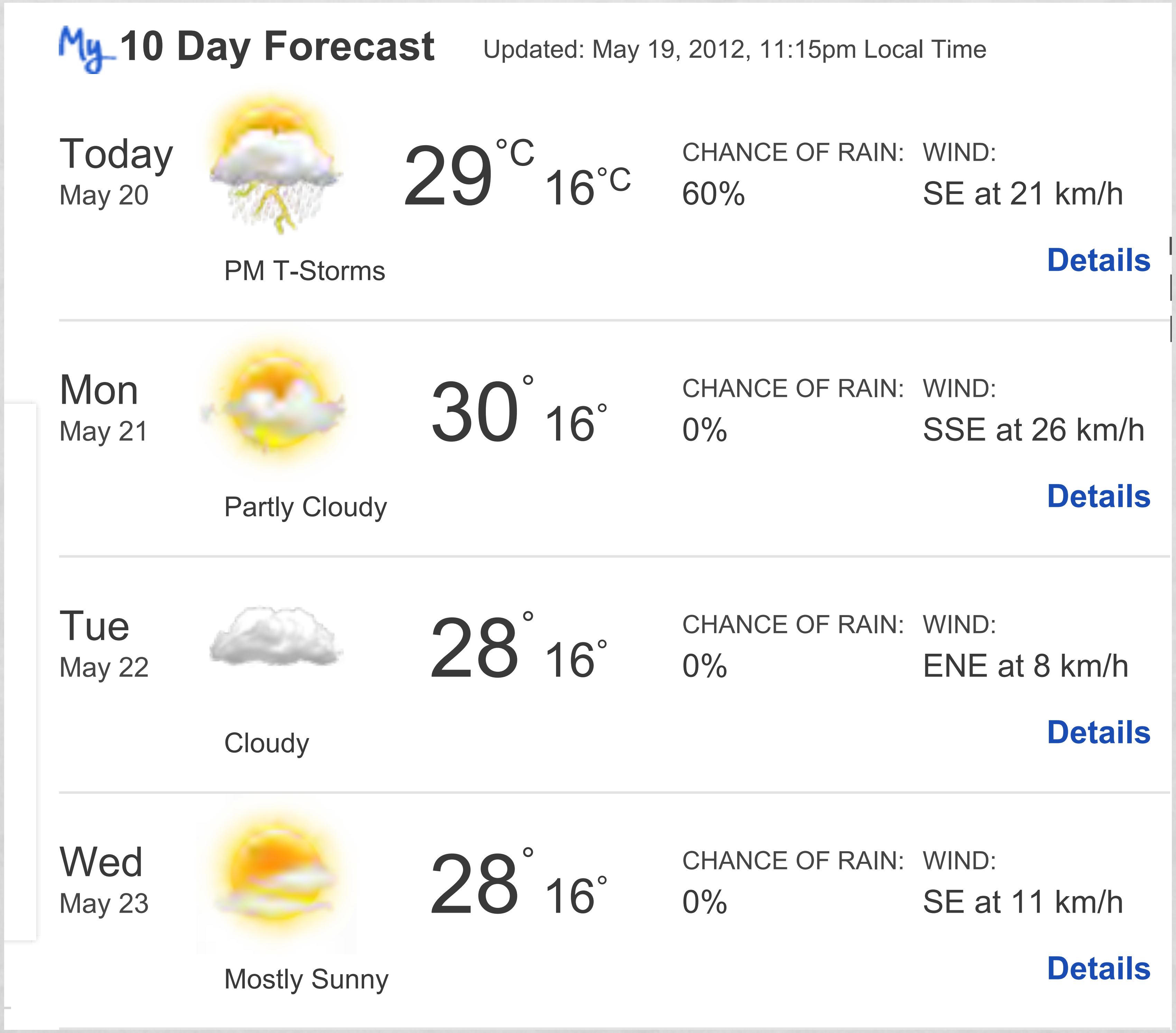
## Discrete Algorithm

- A discrete state-transition system

## Analog Algorithm

- A continuous state-transition system.

# Discrete Transitions



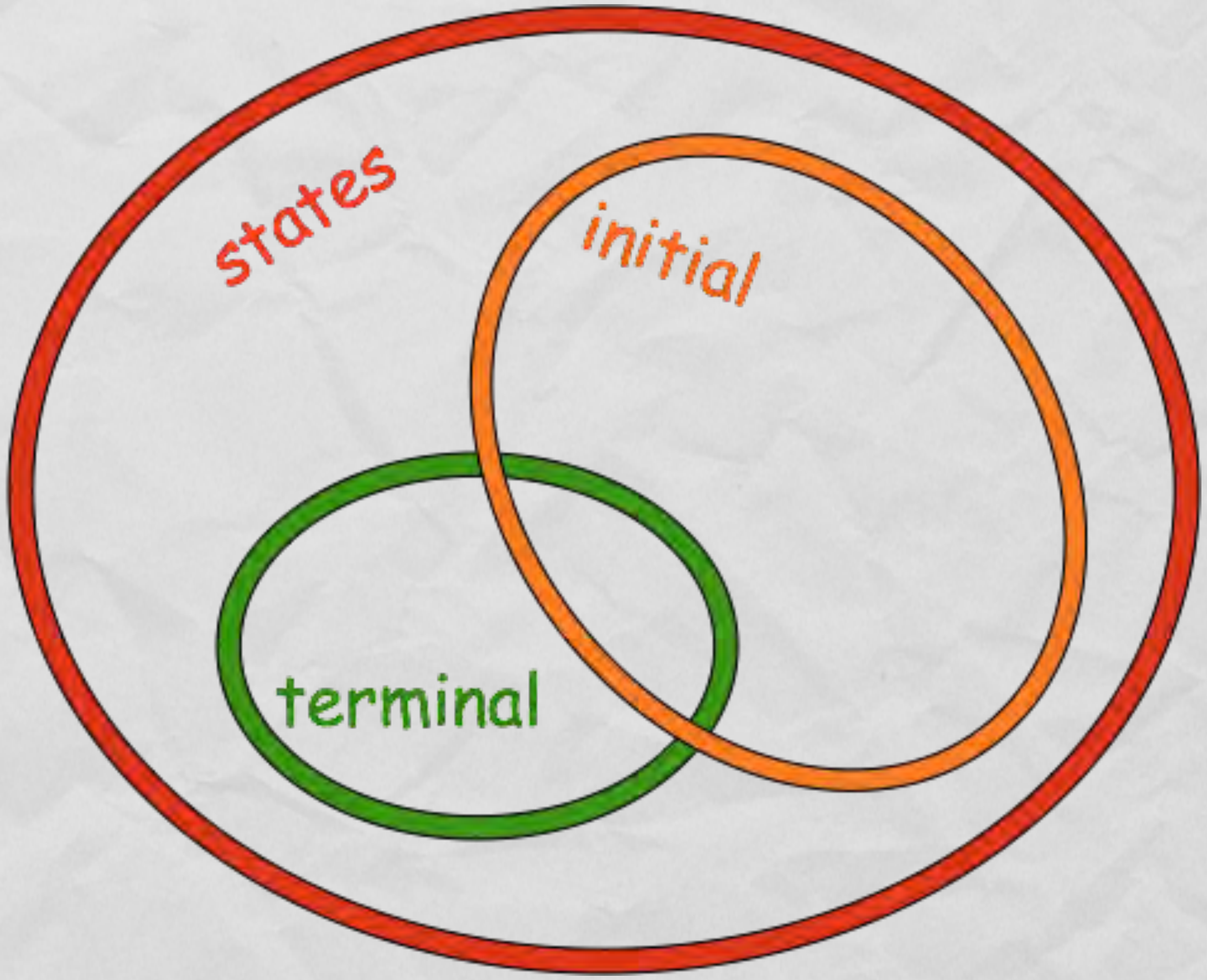
My 10 Day Forecast    Updated: May 19, 2012, 11:15pm Local Time

| | | | | | |
|---|---|---|---|---|---|
| Today<br>May 20 | PM T-Storms | 29°C 16°C | CHANCE OF RAIN: 60% | WIND: SE at 21 km/h | Details |
| Mon<br>May 21 | Partly Cloudy | 30° 16° | CHANCE OF RAIN: 0% | WIND: SSE at 26 km/h | Details |
| Tue<br>May 22 | Cloudy | 28° 16° | CHANCE OF RAIN: 0% | WIND: ENE at 8 km/h | Details |
| Wed<br>May 23 | Mostly Sunny | 28° 16° | CHANCE OF RAIN: 0% | WIND: SE at 11 km/h | Details |

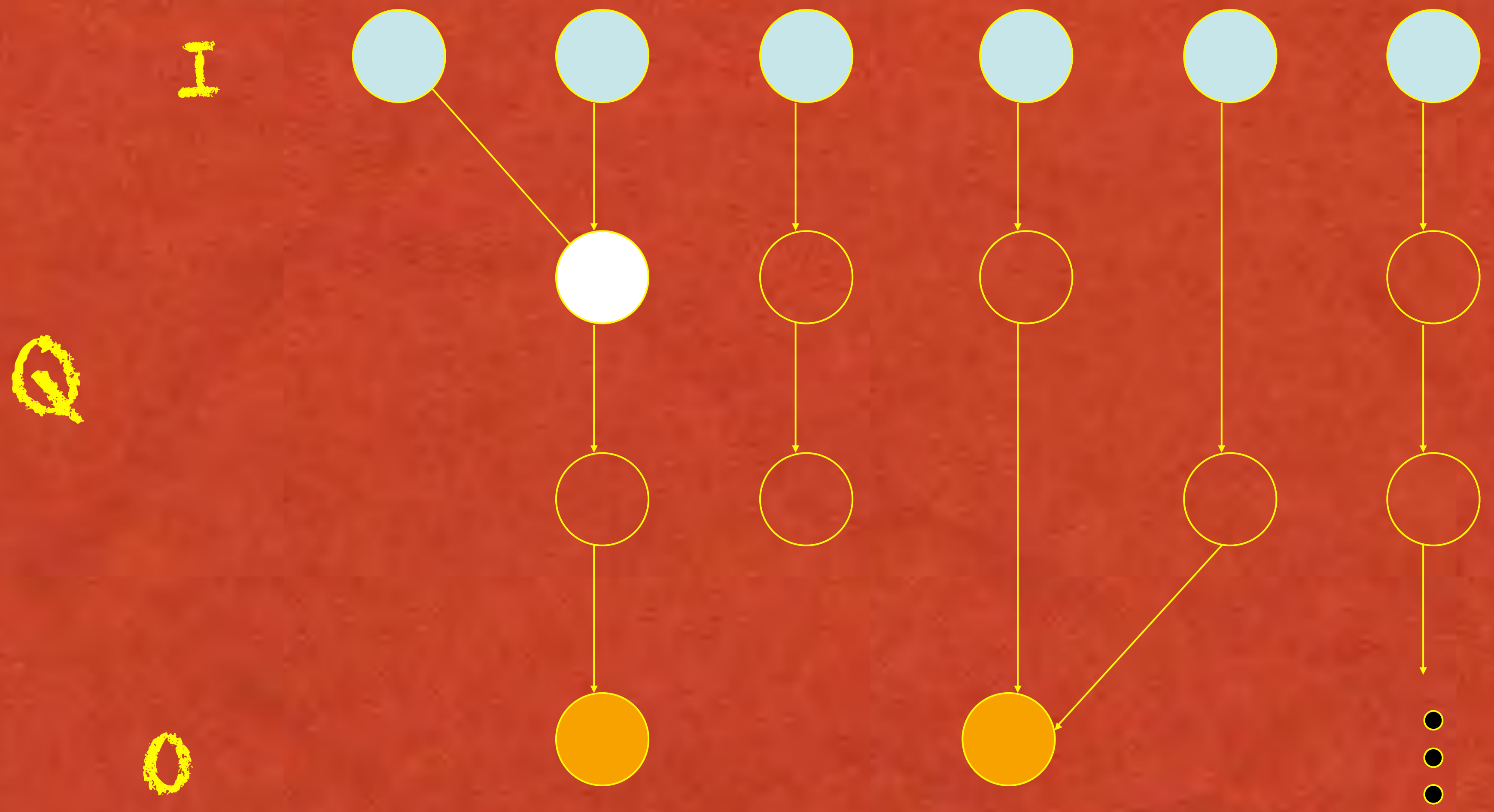
11
12
1
10
2
9
3
8
4
7
6
5

## States

- Everything needed
- Initial & terminal states



# Discrete Transition System



# Continuous System

## Infinite Change

- Zap f
  - $f(t) := 0$
- Uncountably many changes

## I. Evolution

- An algorithm is a state-transition system.

- Its transitions are partial functions.

## Intermediate States

Ce genre de Peinture ... de pouvoir être interrompu quand on veut & repris de même

Paul Romain Chaperon, Traité de la peinture au pastel (1788)

# Intermediate States

## Inputs

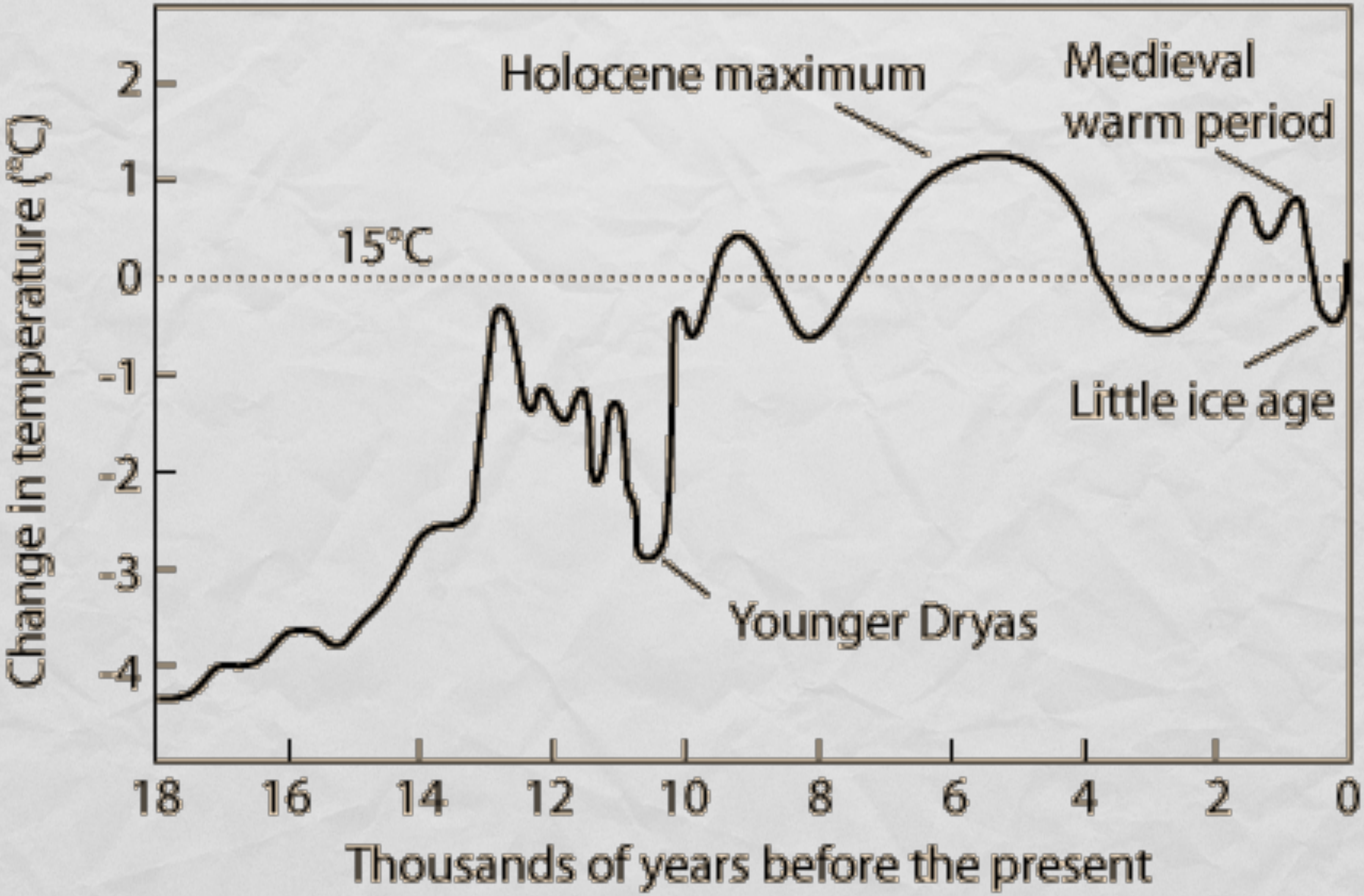
- Environment provides inputs

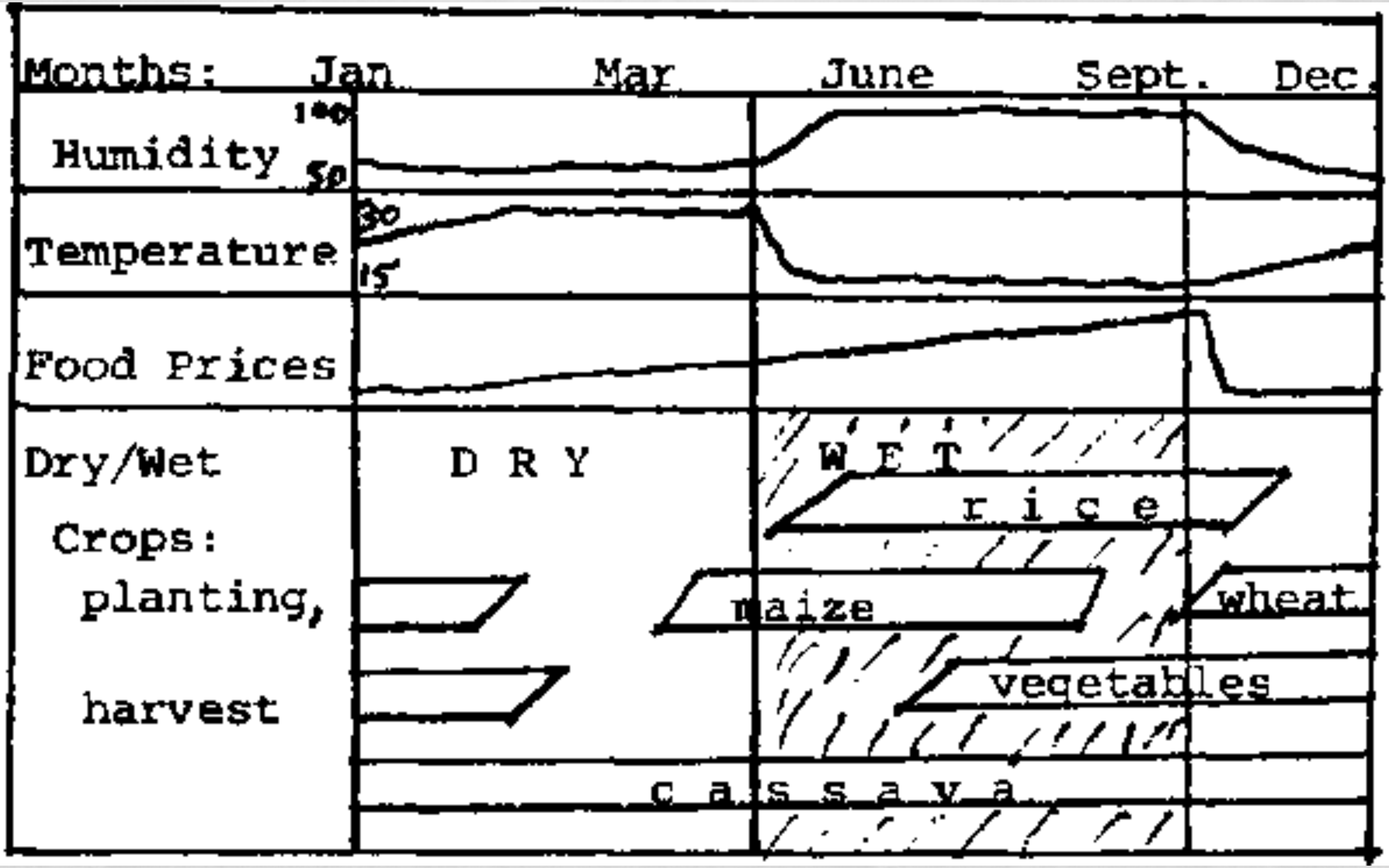
## Discrete Input

- Alternate environment steps and algorithm steps

## Signals

- Function from time to domain (closed under isomorphism)
- Concatenation is associative


Holocene maximum
Medieval warm period
15°C
Younger Dryas
Little ice age
Change in temperature (°C)
2
1
0
-1
-2
-3
-4
18
16
14
12
10
8
6
4
2
0
Thousands of years before the present


Months: Jan Mar June Sept. Dec.
Humidity
100
50
Temperature
30
15
Food Prices
Dry/Wet
D R Y
W E T
Crops:
planting,
harvest
rice
maize
wheat
vegetables
c a s s a v a

# II

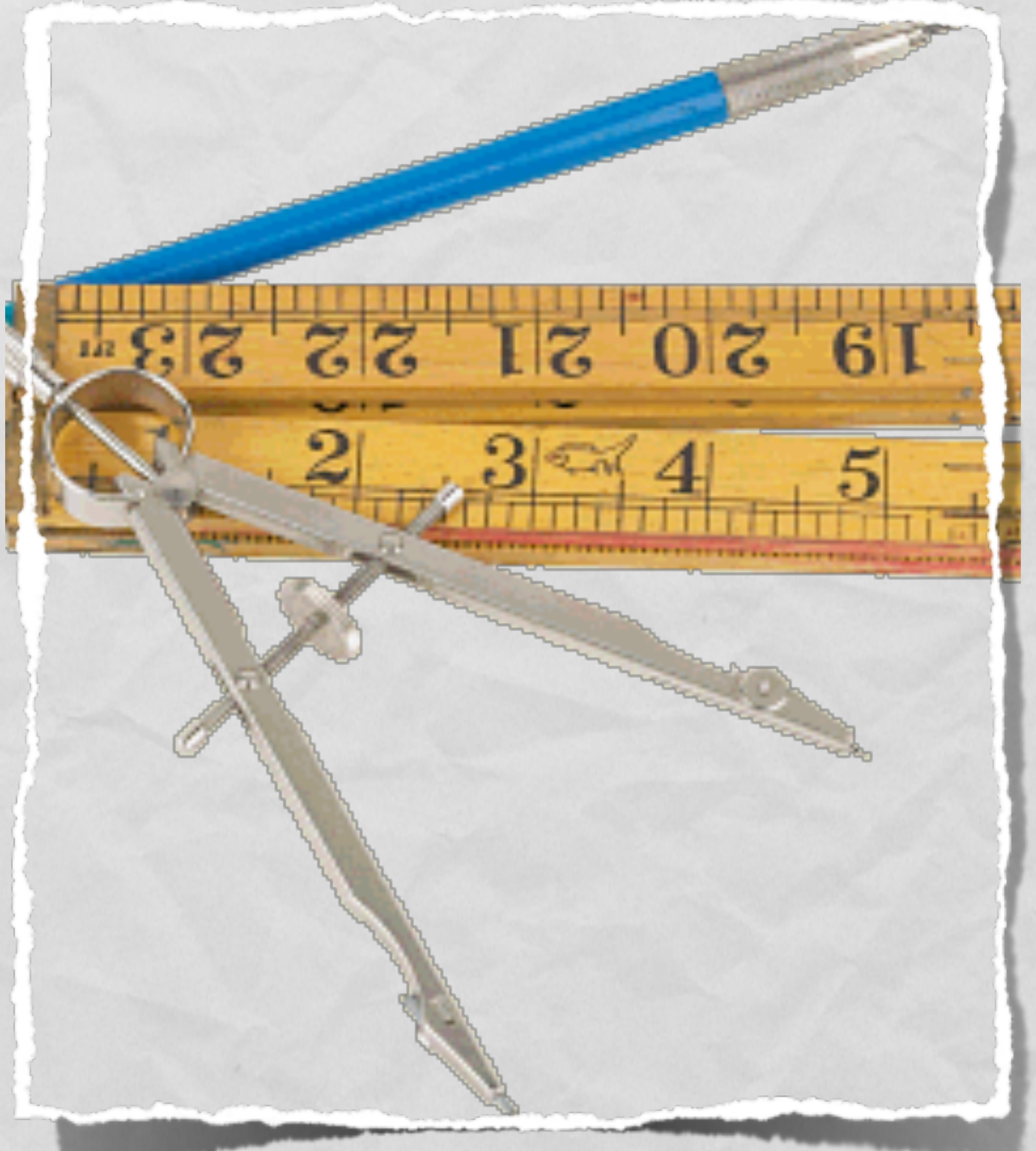
State encapsulates all relevant data!

## States

- Everything needed to proceed (besides the algorithm)

# Geometry

- Domain (underlying set): points, lines, rays, circles, tuples and small bags
- Vocabulary & Operations: Compass; Ruler; =; ∩; Tuple & Bag



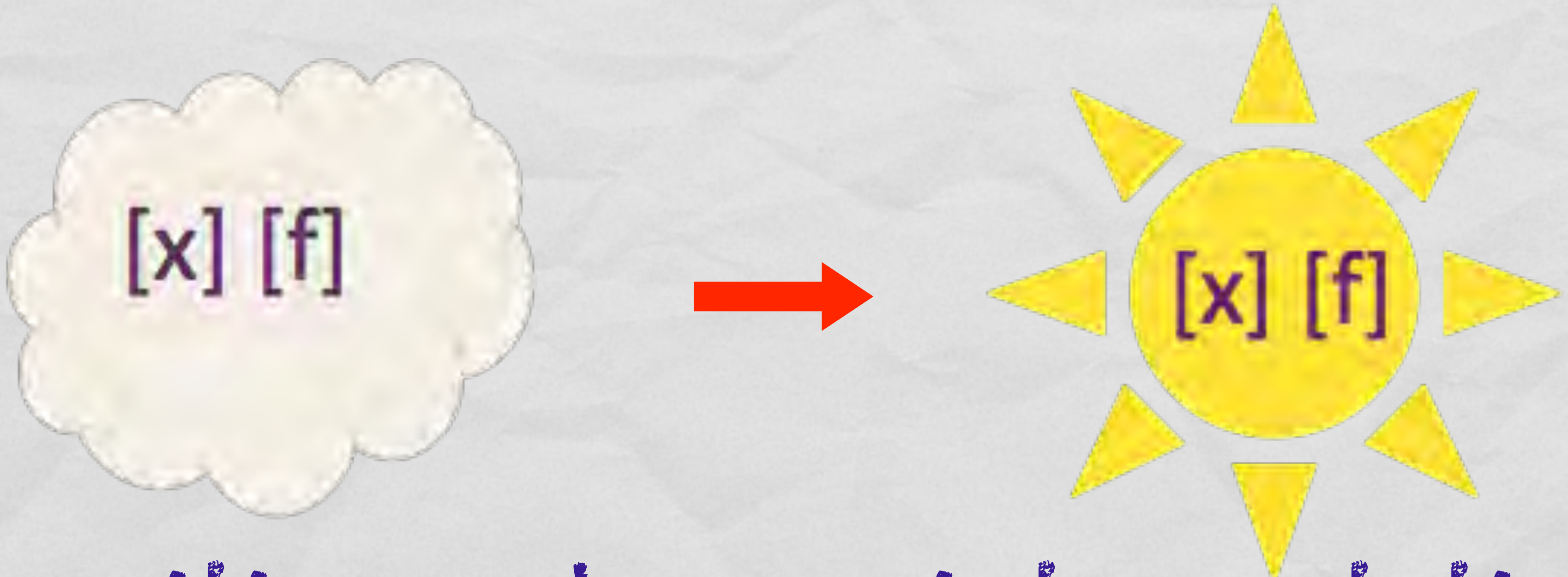
a b

f e

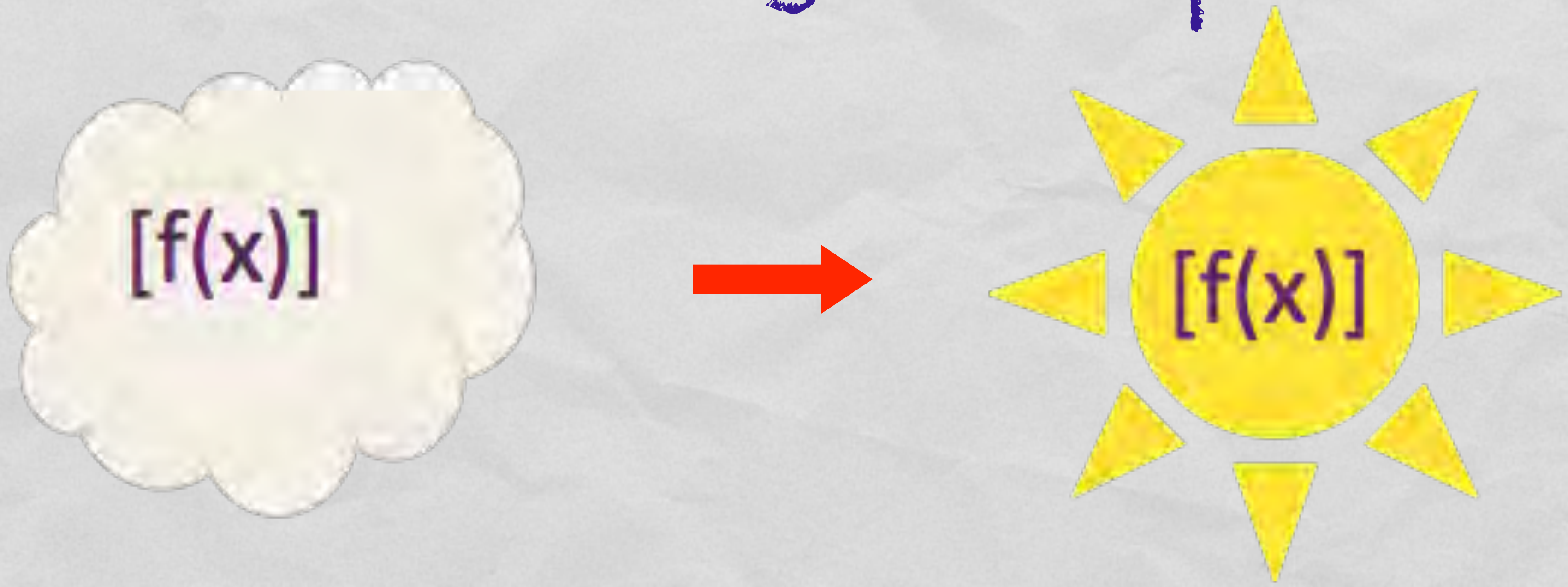
+ - / sgn

= &

# Algebras

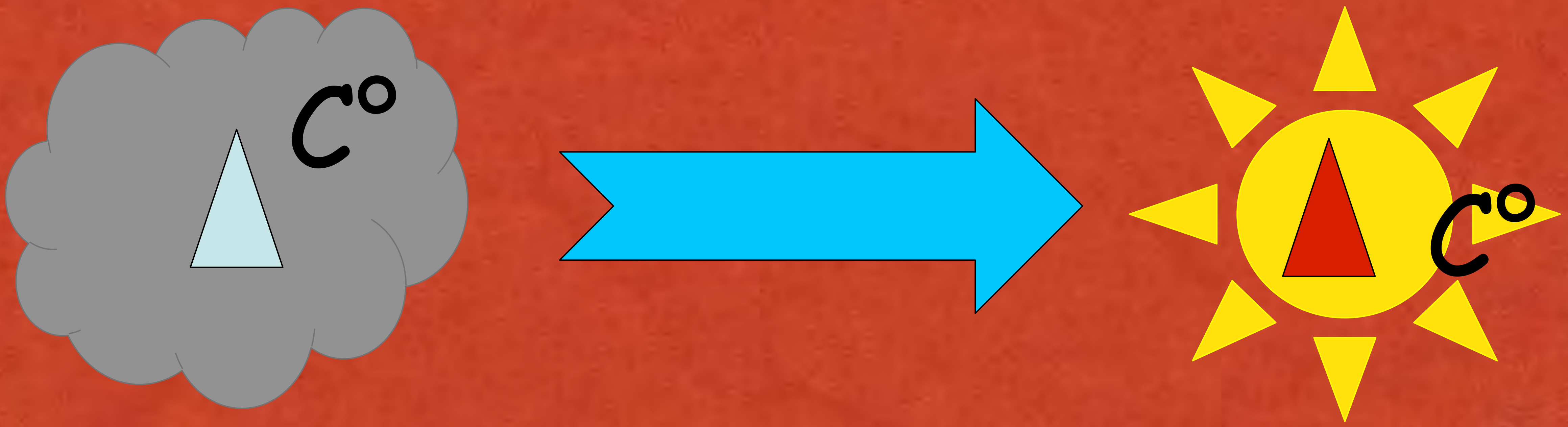


Transitions change interpretations



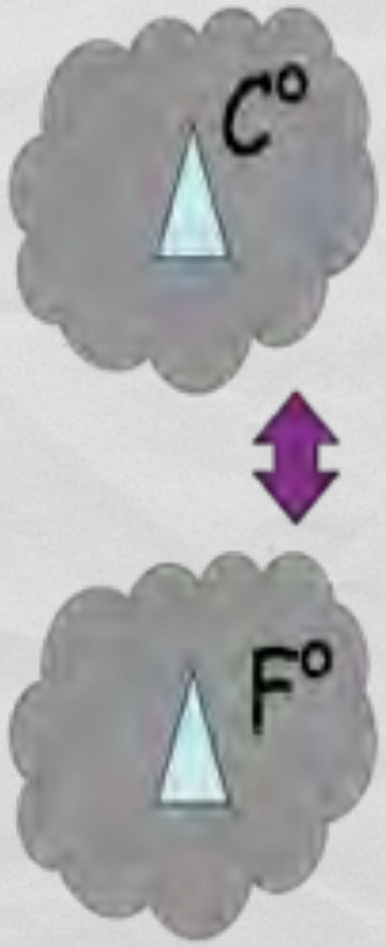
## II. Abstract State

- States are (first-order) structures.
- All states share the same (finite) vocabulary.
- Transitions preserve the domain (base set) of states.

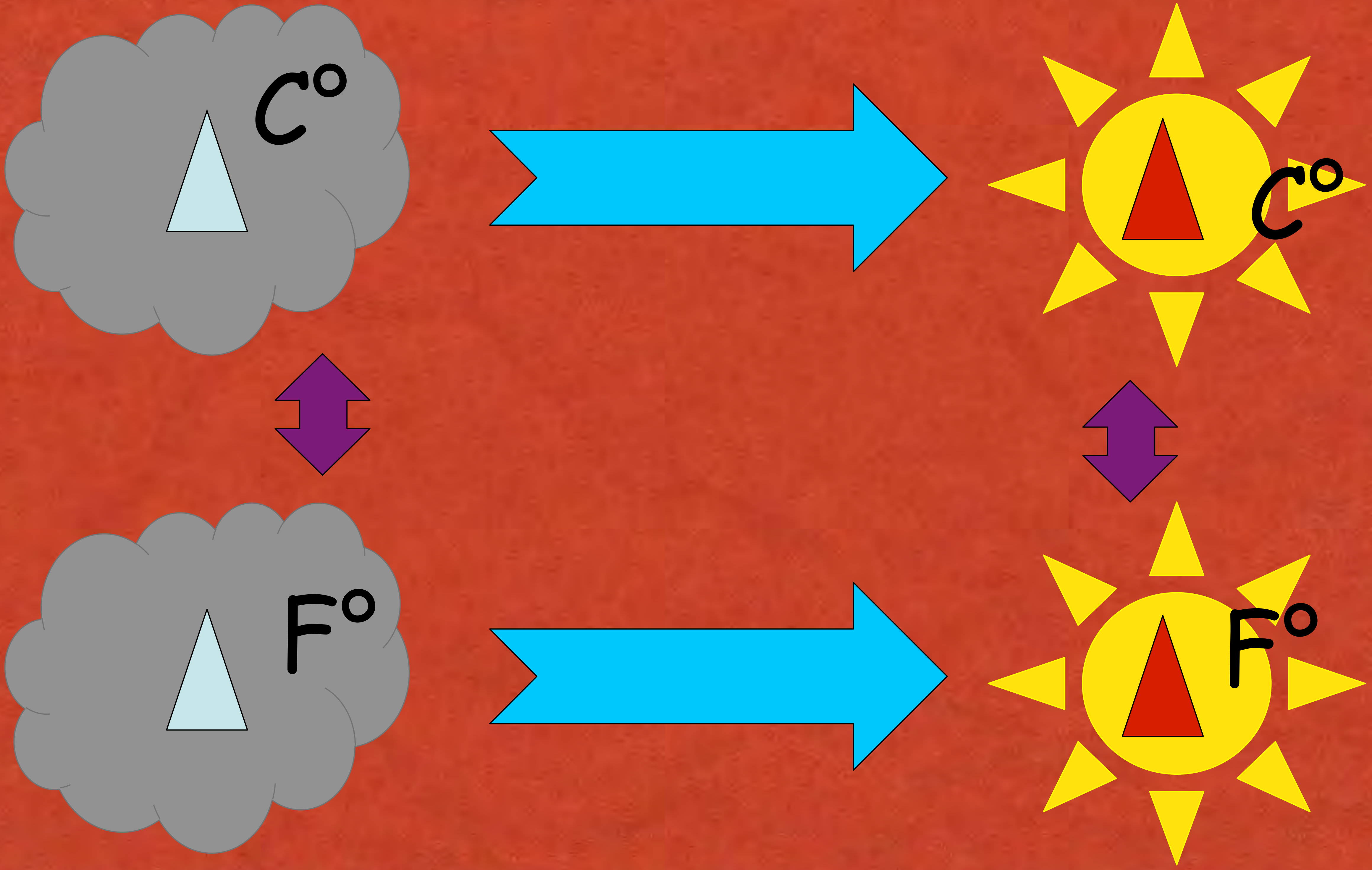

C°
C°

# States & Transitions

- States are abstract (closed under isomorphism)
- Behavior does not depend on internal representation



## Transitions respect isomorphisms



## Isomorphism

- Transitions respect isomorphisms
  - $X \cong Y \Rightarrow X' \cong Y'$

# Isomorphism

- Transitions respect isomorphisms
  - $X \cong Y \Rightarrow X_t \cong Y_t$

## Isomorphism

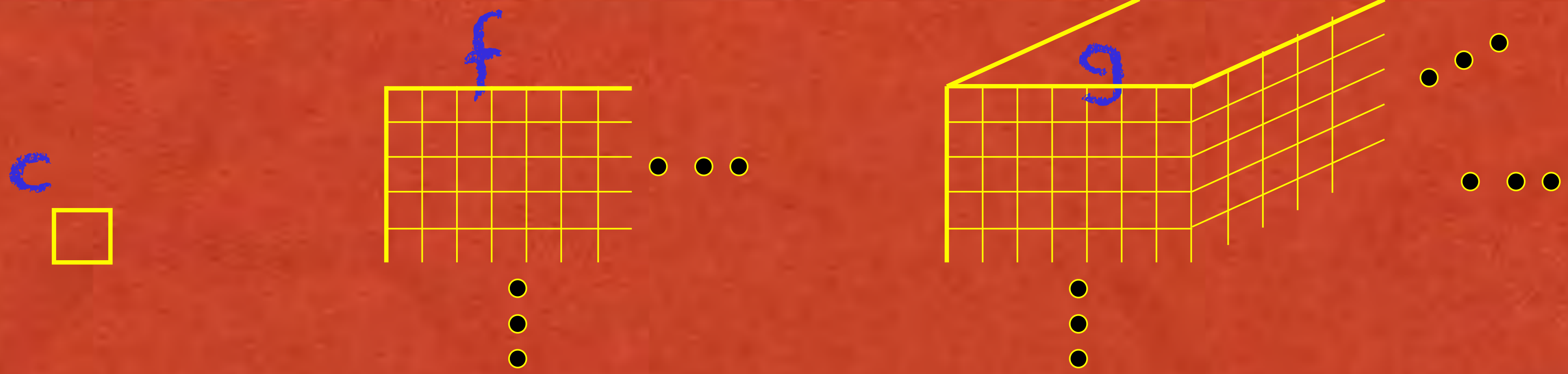
- Transitions respect isomorphisms
  - $X \cong Y$ & $u \cong v \Rightarrow X_u \cong Y_v$

# Inputs

- Environment provides inputs via ports
- $X_u$ state after input u

# States are Abstract

- Data is arranged in a structure of a finite signature.



- An algorithm is abstract, thus applicable to all isomorphic structures.

$$\begin{tikzcd}
\text{x="101"; s(x)="110";...} \arrow[d, "\tau"'] \arrow[r, phantom, "\cong"] & \text{x=5; s(x)=6;...} \arrow[d, "\tau"'] \\
\text{x="110"; s(x)="111";...} \arrow[r, phantom, "\cong"] & \text{x=6; s(x)=7;...}
\end{tikzcd}$$

## Compute by Analogy

## What is Analog Computation?

- Identifying the motivating problem.
- Specifying the problem by a system of differential equations.
- Designing a network to solve the equations.
- Calculating conditions on the data and parameters to ensure good experimental behaviour of the network.
- Constructing an analog machine using a particular technology.
- Using the machine for measurements/ experimental procedures.

# Models

$$\frac{\partial c_1}{\partial \theta} = \frac{1}{P_g} \frac{\partial^2 c_1}{\partial Z^2} - \frac{\partial c_1}{\partial Z} - \frac{N}{m} (c_1 - c_2) \tag{1}$$

$$\frac{\partial c_2}{\partial \theta} = \frac{1}{P_{gl}} \frac{\partial^2 c_2}{\partial Z^2} + N \varepsilon (c_1 - c_2) - M f (c_2 - c_3) \tag{2}$$

$$\frac{\partial c_3}{\partial \theta} = M f (c_2 - c_3) - Q f c_3 \tag{3}$$

$$f = \exp(-P_p Z) \tag{4}$$

## Signals

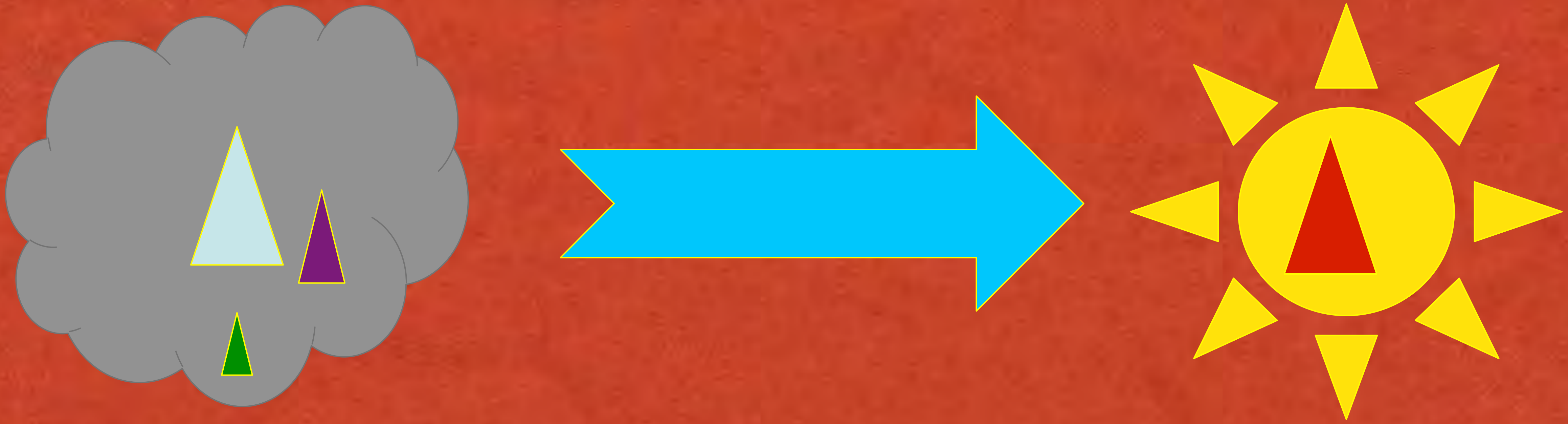
- Function from time to domain (closed under isomorphism)
- Concatenation is associative

## Retrospection

- Current state depends on past
- Intermediate states
  - $X_{uv} = (X_u)_v$

# II. Abstract State

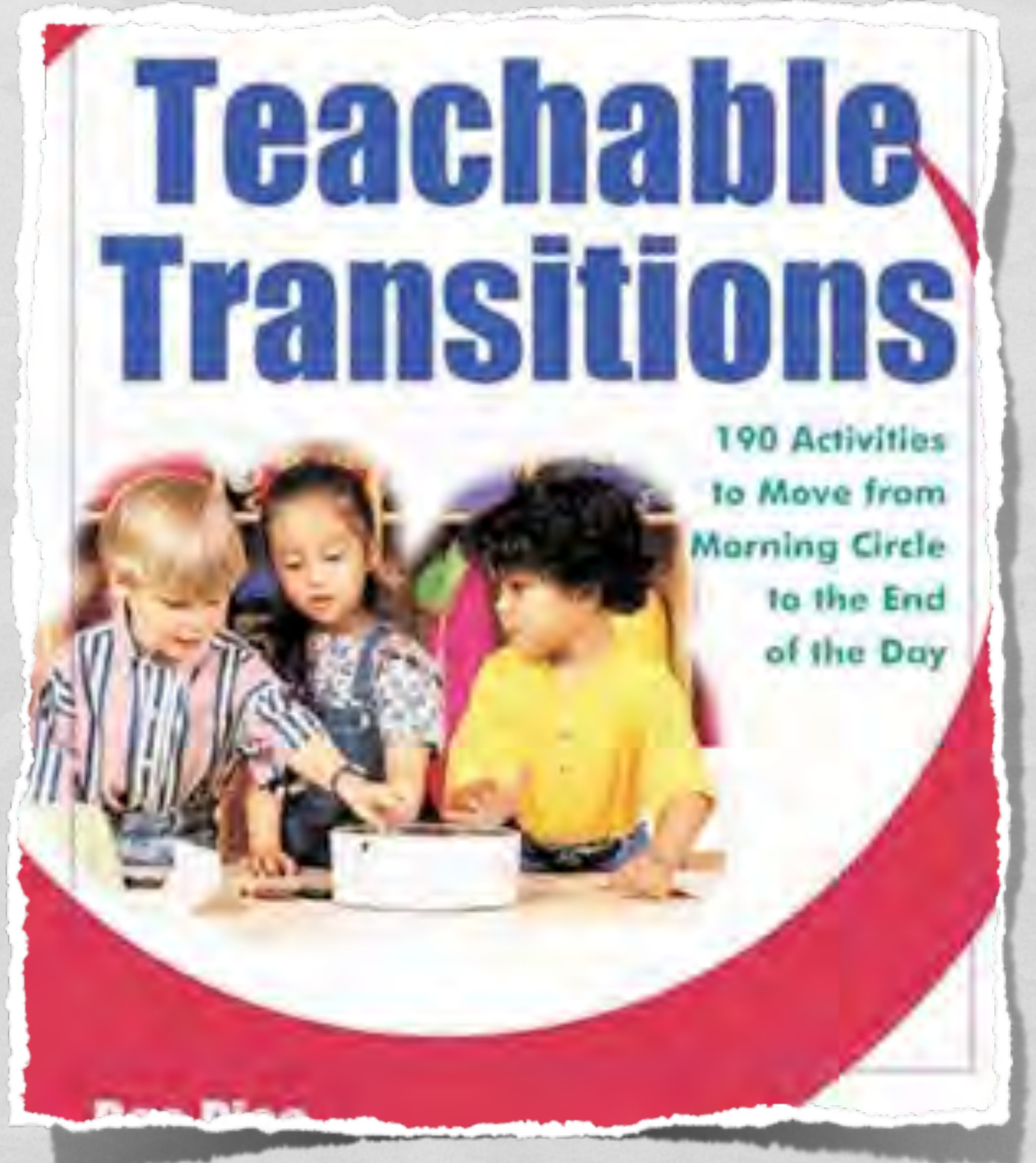
- States are (first-order) structures.
- All states share the same (finite) vocabulary.
- Transitions preserve the domain (base set) of states.
- States (and initial and terminal states) are closed under isomorphism.
- Transitions commute with isomorphisms.



III

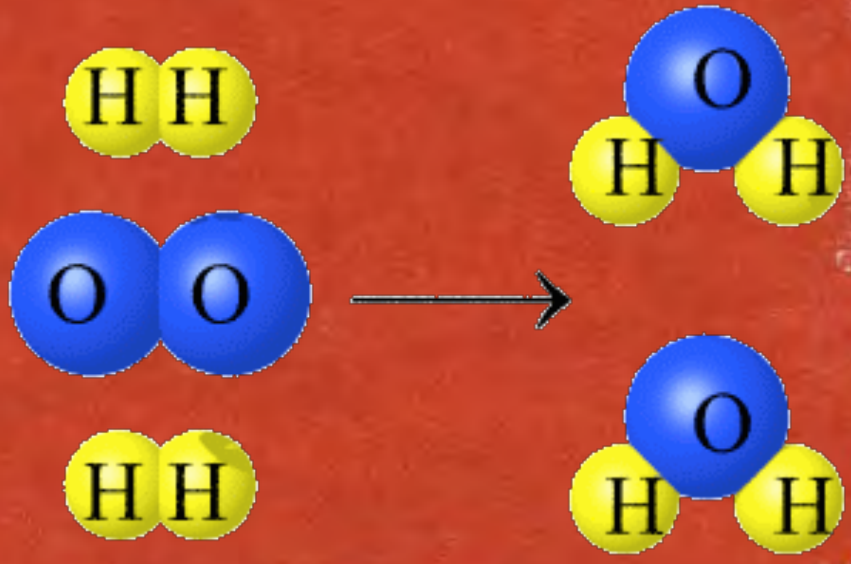
# Algorithmic Transitions

- States evolve
- Evolution described by finite program


Teachable Transitions
190 Activities to Move from Morning Circle to the End of the Day

## What is a transition?

- Transitions are algorithmic if they can all be described finitely (without presupposing any special knowledge).



## Kleene

An algorithm in our sense must be fully and finitely described before any particular question to which it is applied is selected....

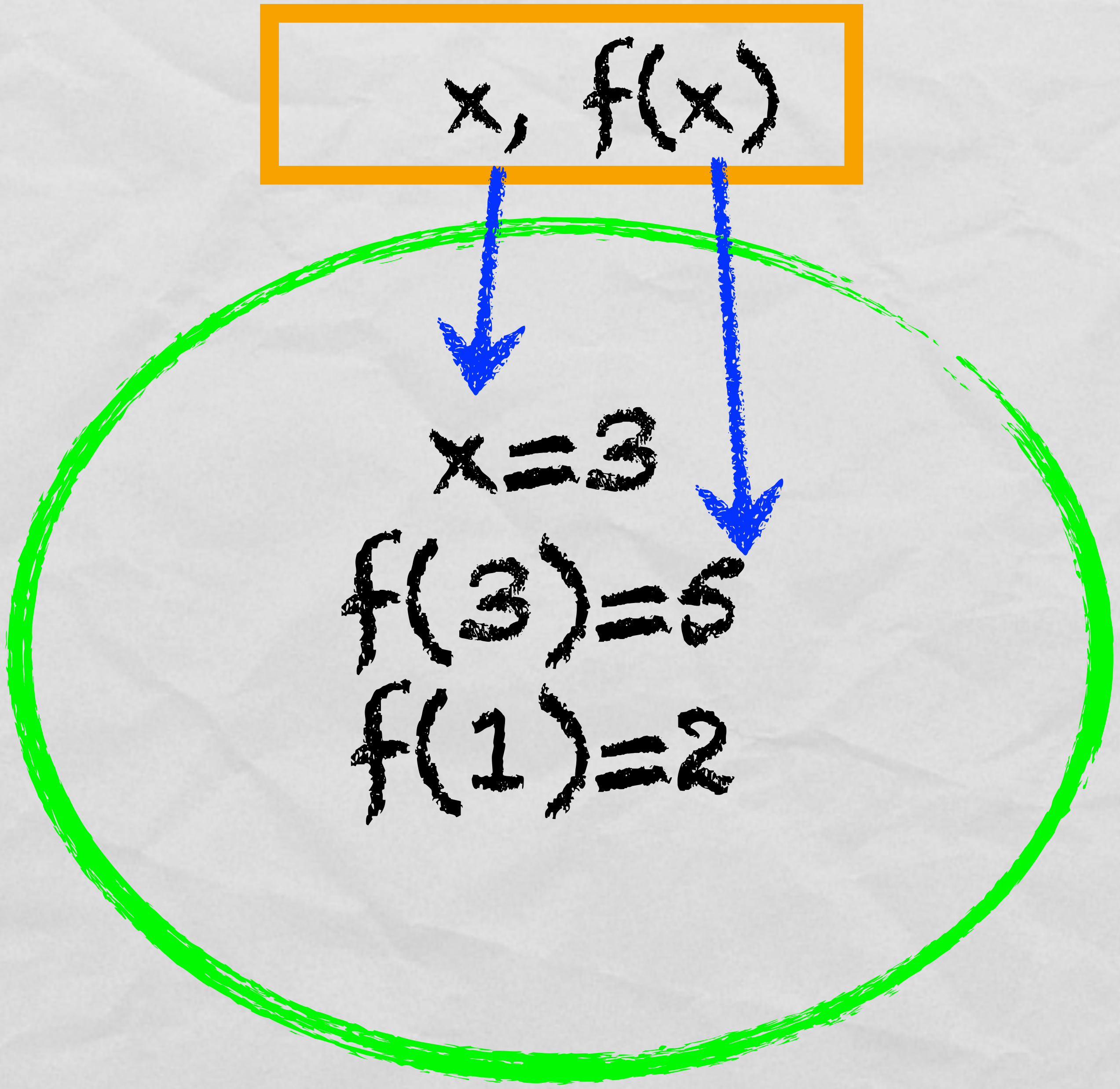
All steps must ... be predetermined and performable without any exercise of ingenuity or mathematical invention by the person doing the computing.

## III. Algorithmic Transitions

Yuri Gurevich

- Transitions are determined by a fixed finite set of terms, such that states that agree on the values of these terms, also agree on all state changes.

## Terms & Locations



## Critical Terms

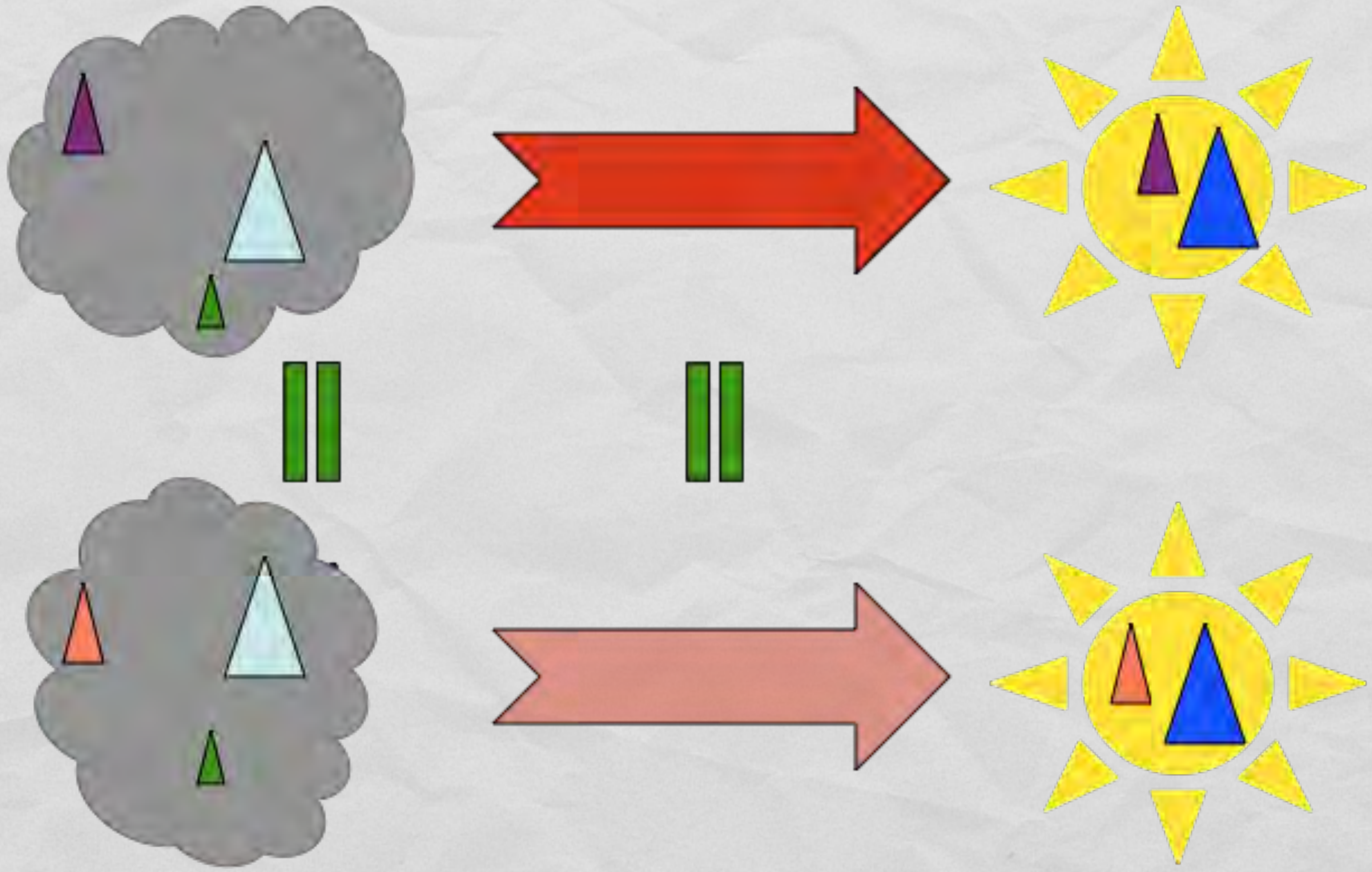
$\boxed{T\text{: } x,\ f(x)}$

$$\begin{tikzcd}
\begin{array}{c} x=3 \\ f(3)=5 \\ f(1)=7 \end{array} \arrow[d] & \begin{array}{c} x=3 \\ f(3)=5 \\ f(1)=2 \end{array} \arrow[d] & \begin{array}{c} x=3 \\ f(3)=5 \\ f(1)=1 \end{array} \arrow[d, "X" description] \\
\begin{array}{c} x=1 \\ f(3)=0 \\ f(1)=7 \end{array} & \begin{array}{c} x=1 \\ f(3)=0 \\ f(1)=2 \end{array} & \begin{array}{c} x=1 \\ f(3)=0 \\ f(1)=4 \end{array}
\end{tikzcd}$$

## Operations

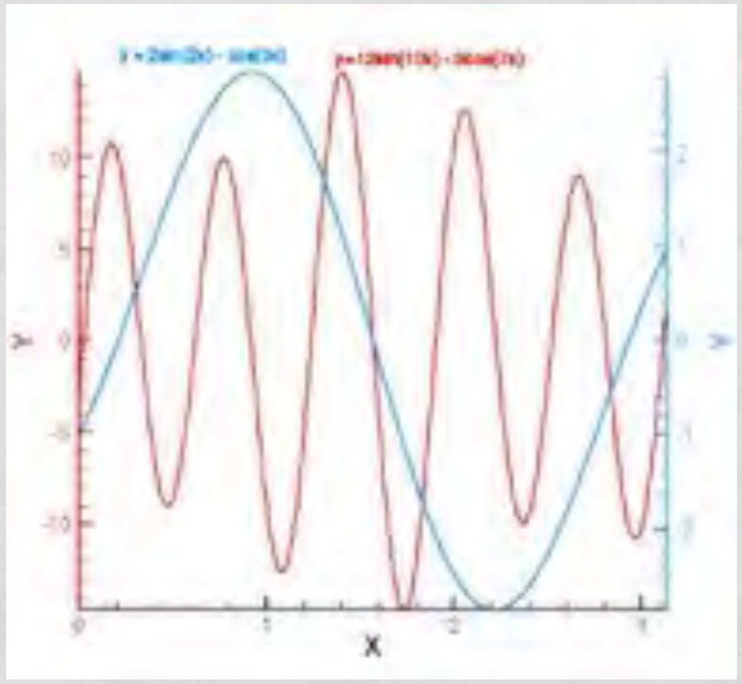
- Abstract algebraic operations
- May be partial (3/0 is undefined)
- Hangs when result is undefined

# Algorithmic Transitions



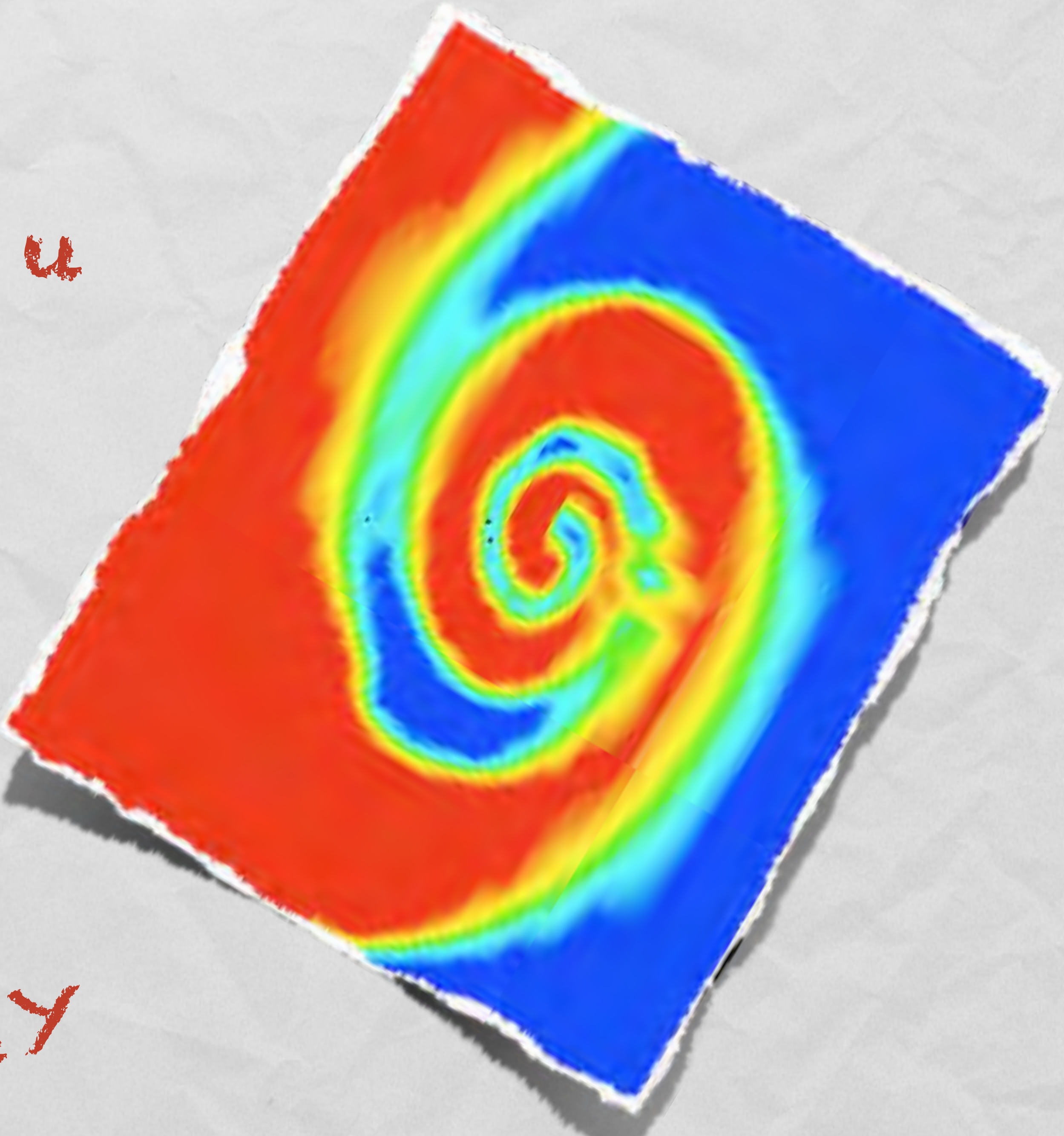
# Transitions

- View state as location-value pairs
    - $f(a,b,c) \mapsto d$
- Changes to state $X$
    - $\Delta X = X' \setminus X$
    - $\Delta_u X = X_u \setminus X$



## Evolution

- Transition to $X_u$ under input signal u
  - $\Delta_u X = X_u \setminus X$
- Evolution depends on critical terms and input port
  - $X =_T Y \Rightarrow \Delta_u X = \Delta_u Y$



## Flows

- Fixed dynamics over stretch of time
- If input wouldnt change, nothing would
- Equalities between critical terms maintained

## Jumps

- Change of dynamics
- Requires conditionals

## Partiality

- Locations still to be accessed are determined by locations already accessed

## Past Determines Future

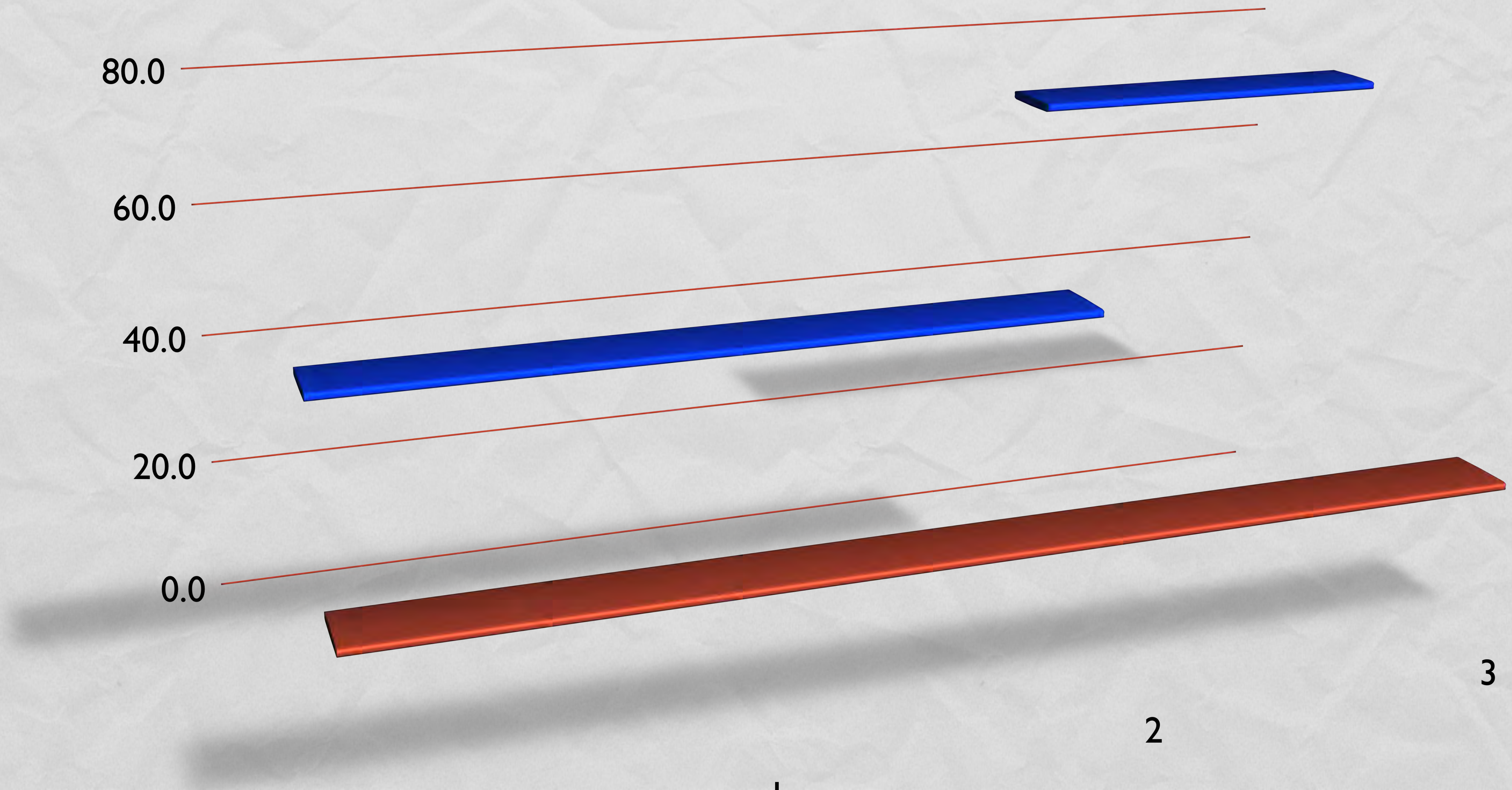
$$X_{\tilde{u}} =_T Y_{\tilde{v}}$$

$$u_{|u|} = v_{|v|}$$

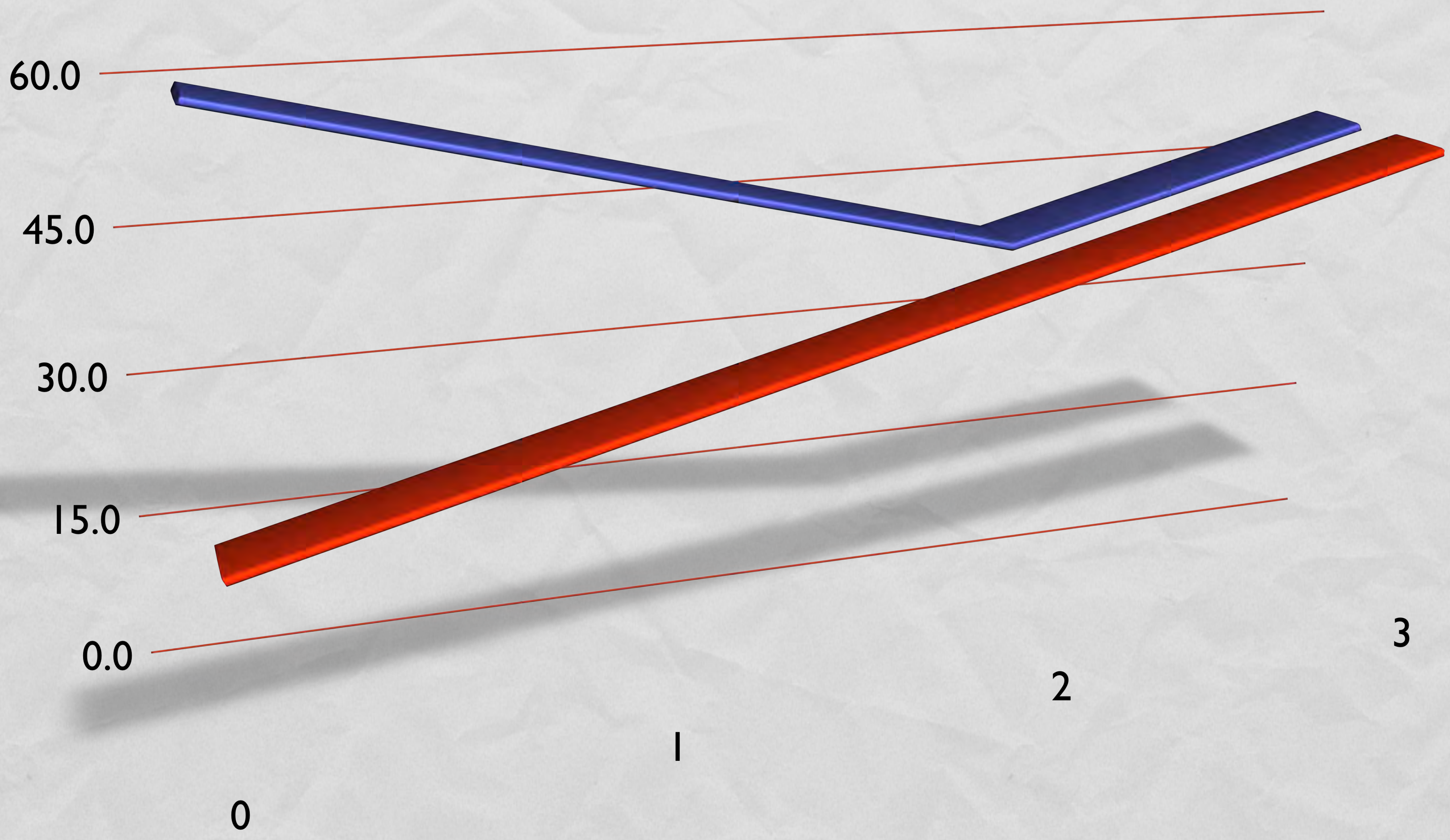
$$\Rightarrow$$

$$\Delta_u X = \Delta_v Y$$

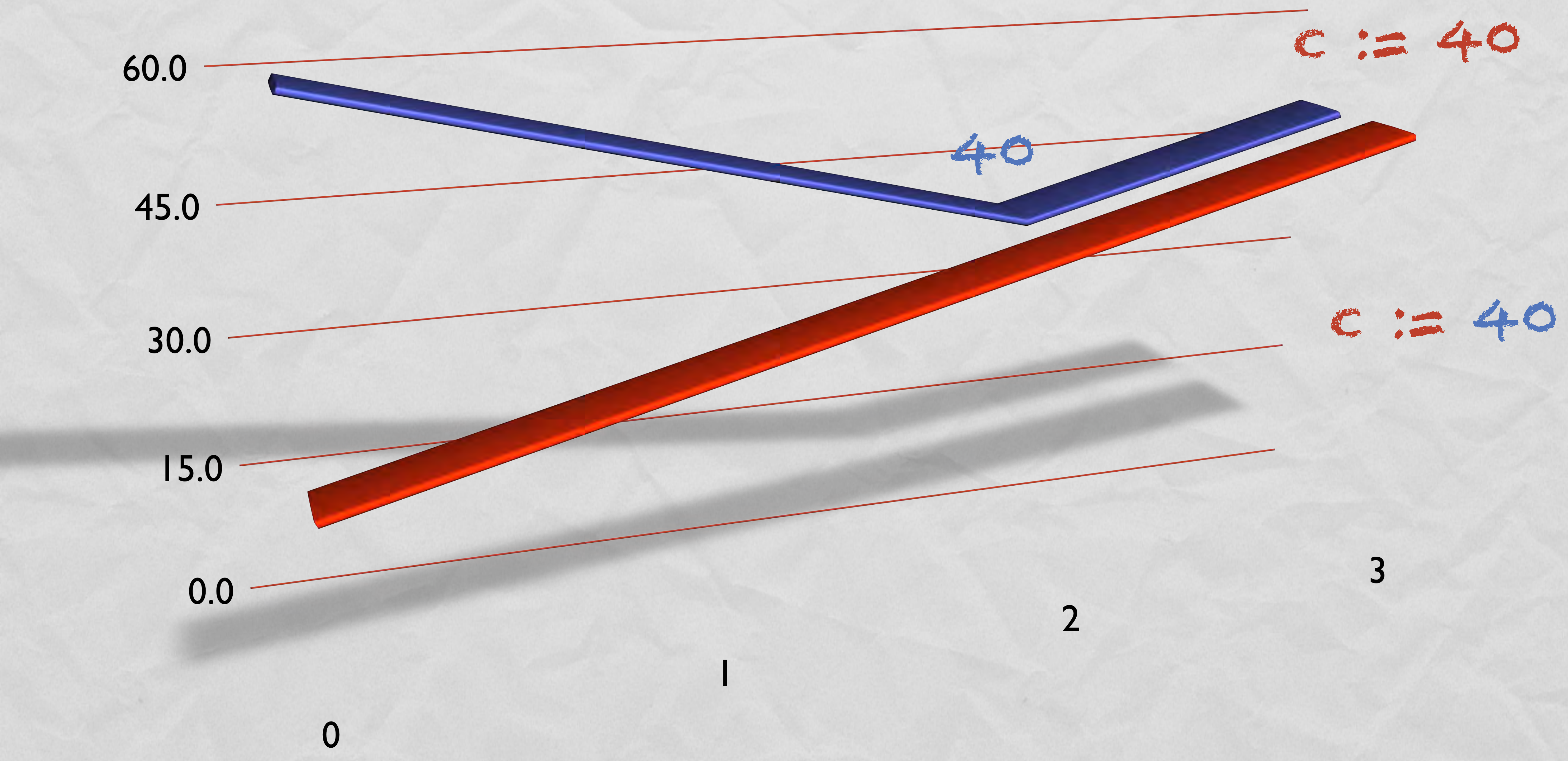
# Non-Deterministic



# Deterministic



# Critical Moment



## Non-Flows

- x := x+1
- if x=0 then y := -y
  - unless y=0
- x := x/2
  - unless x=0

## Flow vs. Jump

$z = \lfloor t \rfloor$

if $t<z$ then $z := z-1$ ||

if $t \geq z+1$ then $z := z+1$

## Before & After

CONCERT HALL
R. Goebel Photo.
60 Souvenir Postals of St. Charles and Vicinity
St. Charles, Mo.
BEFORE
AFTER
HURRICANE FEBRUARY 26TH, 1876.

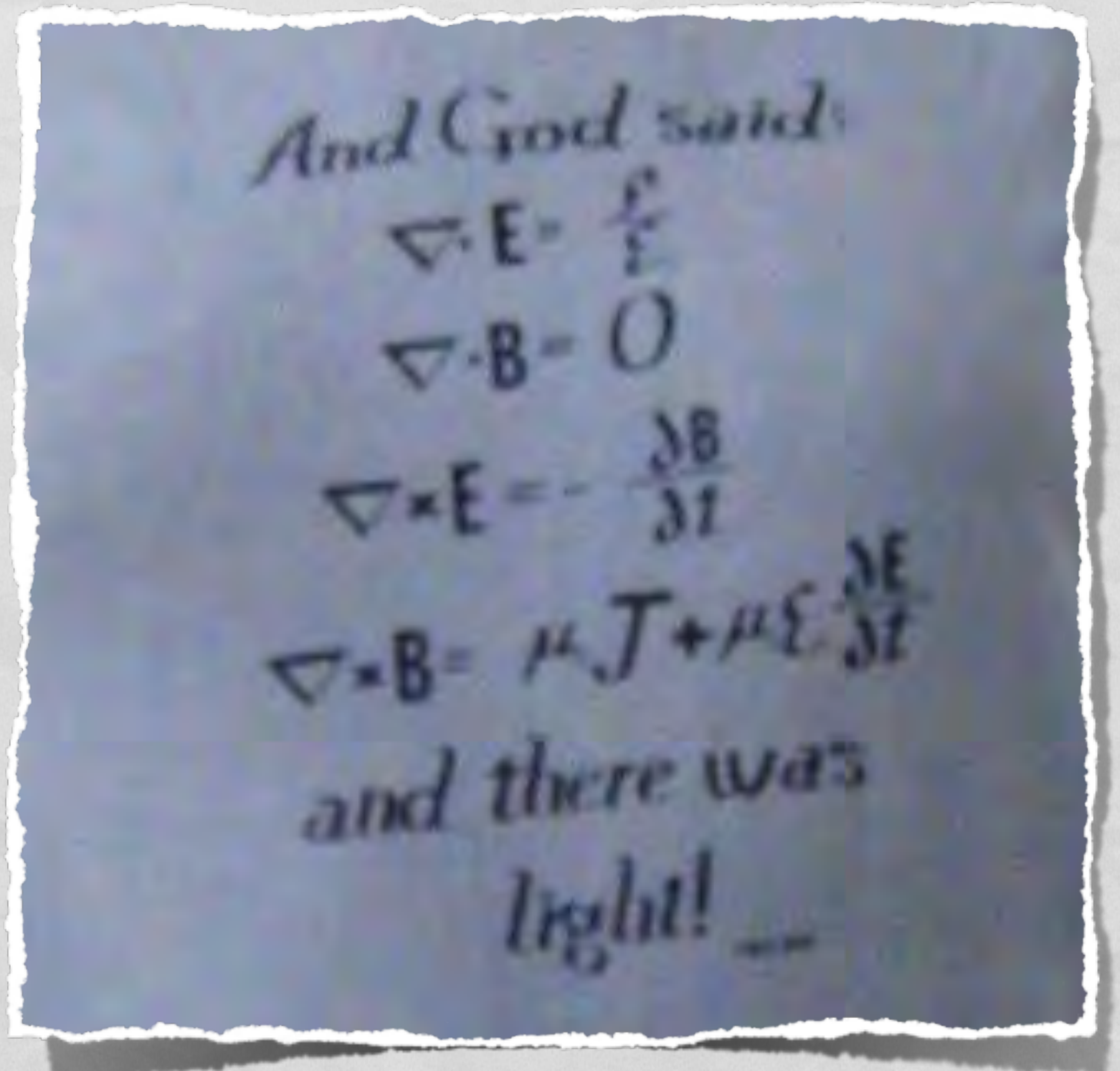
- At every jump, there are "before" and "after" states: X before algorithm makes changes and X' after

## Critical Moment

- Suppose $f(a,b,c) := d$ in $\Delta_u X$
- After some prefix $w$ of $u$, $d$ in $[\![T]\!]_{X_w}$
- If not, let $Y$ be $X$ with $d'$ instead of $d$
- And $v$ be $u$ with $d'$ instead
- By criticality $f(a,b,c) = d$ also in $Y_u$
- By isomorphism $f(a,b,c) = d'$ in $Y_v$
- So $u \neq v$, and signals must part ways

## Flow Equations

- Solved form
- Left-differentiation operator
- Solver for implicit equations
- Infinitesimals



And God said:

$$\nabla \cdot E = \frac{\rho}{\varepsilon}$$

$$\nabla \cdot B = 0$$

$$\nabla \times E = -\frac{\partial B}{\partial t}$$

$$\nabla \times B = \mu J + \mu \varepsilon \frac{\partial E}{\partial t}$$

and there was light! ...

# Bounce, Bounce

- $t$ time signal
- $dt$ infinitesimal
- $g,k$ constants
  - $s := s + g \cdot dt$
  - $x := x + s \cdot dt$
  - if $x=0$ then $s := -k \cdot s$

# Explicit Flow

- t time signal
- g,a,s inputs

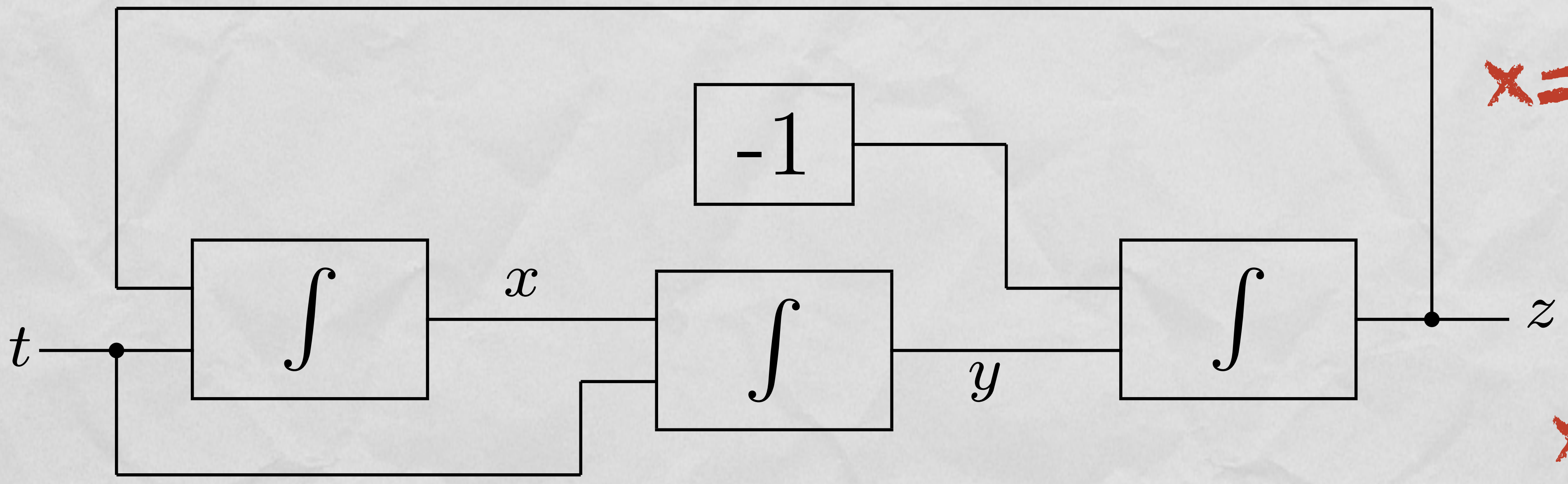
Interception
Acquisition
Staging
Booster Burn Out
Target Acquisition by Green Pine Radar
Incoming Target Missile ("Black Sparrow")
Enemy Missile Launched From H3
Launch of Arrow Interceptor
Arrow Weapon System

$x := t \cdot s \cdot \cos a$

$y := t \cdot s \cdot \sin a$

$- \frac{1}{2} \cdot g \cdot t^2$

## Implicit Flow



- t time signal
- initially
  $x=1;\ y=z=0$

$x' = z$

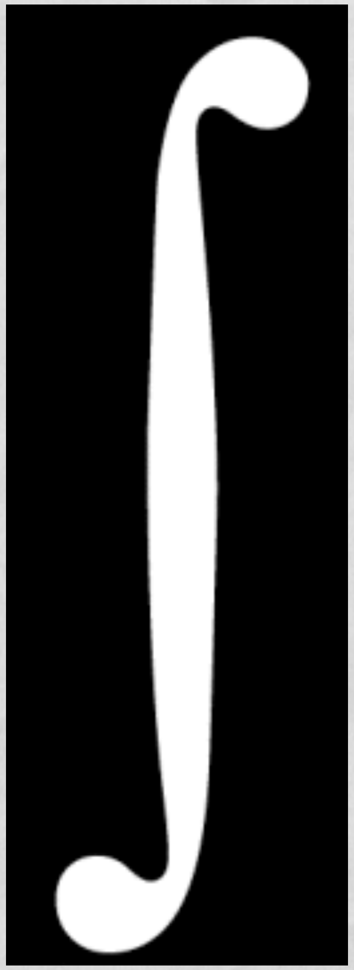
$y' = x$

$z' = -y$

## Signal of Signals

- Tiny pieces of history
- Left-differentiation operator
- $y := x'$

## Bruno Scarpellini 1963



- Compute the undecidable via infinite precision integral.
- It is conceivable that the mathematics of a collection of axons may lead to undecidable propositions. (2003)

## John Myhill

- To make Scarpellini's work a basis for constructing an actual computer which can solve problems which are not digitally (= recursively) solvable:
  - Assume perfect functioning.
  - Assume perfect sensor (zero test).

0

## Tide-Prediction Manual

- The machine to be described here, like almost every contrivance, apparatus, or machine in practical use, is based very largely upon what has been accomplished by others who previously labored in the same field.

U.S. Coast and Geodetic Survey (1915)

# IV

An algorithm is effective if its initial states have a finite description

If initial states can be described, then all states can be