The Mathematics of the Models of Reference

Intro: The Notion of MoR

Artificial Intelligence (AI) is the idea that the fuzzy aggregate of abilities we collectively call "intelligence" can be realized artificially – specifically, as an algorithm. It's no surprise, then, that AI faces two historical difficulties: (1) how to build an information processing device to implement such a conjectured algorithm; (2) how to fine-tune that fuzzy notion: intelligence. Nowadays' computers only capture in a rough way some aspects of that notion – typically, the "non-logical", analogical, associative, and contextual. But ordinary computing machines sometimes don't do well also when they play at home, in the fields of logic and mathematics (think about factoring large integers and non-polynomial problems).
To address the two problems, the iLabs research laboratory focuses, respectively, on two basic ideas: (1) a theoretical isomorphism between physical and informational reality; and (2) an extensive formal development of the notion of *model of reference* (MoR). Such a formal development is the *mathematics of the models of reference.*

As for (1): at iLabs, we believe information to be rooted in physical reality from its very bottom. We believe that what goes on in the world can be tersely, but correctly, described by claiming: what the universe *does* at any instant of time is a computation its overall state at the following instant. Therefore, Artificial Intelligence is not that artificial: the good information processing devices are out there! And this is so, because universal computation is embedded in the ultimate essence of reality.

As for (2): formally, a model of reference (MoR) is an ordered triple <*perception*, *thought*, *action*>. Using a standard operational notion, f: A $\Rightarrow$ B is a MoR defined on a set A of perceptions, whose proper thought consists in effectively mapping them to actions or outputs in a set B. Thus, we can write:

$f(x_1, x_2, \ldots, x_n) = y$

with $x_1, x_2, \ldots, x_n \in$ A, and $y \in$ B. At the bottom of reality, a MoR works as a simple change in the informational patterns of the smallest physical items constituting our world (more on this soon). At the human level, it is what explains our behaviour in terms of how we perceive and elaborate information. Generalizing: models of reference (MsoR) are fractal items displaying isomorphic structure at any level of reality. Thus, they are our most promising candidate to the theoretical role of functional interface between matter and computation. You can read this page because the relevant MsoR are activated in your eye, in order for it to process visual stimuli and send certain signals to the area of your brain taking care of vision. A frog can grab a fly by activating several hundreds of MsoR. Dually, we are phobic, or sick, or develop cancer, because our mind, our body, our cells, activate the "wrong" MsoR. Arithmetic operations, business software, or the proof of a theorem in universal algebra, are MsoR as well. What follows is an overview of the results spelled out with the due formal details in the [forthcoming] iLabs book, where the theoretical options to be exposed are also philosophically defended.

1. A Discrete Universe

How is a universe in which matter and information are two sides of the same coin to be conceived? Here's the iLabs proposal. The universe (call it U) has to be *discrete* and *finite*, to begin with, with *minimal space-time units* at its bottom (how minimal? 10^{-35} m for the minimal space unit, and 10^{-44} seconds for the minimal time unit, might be good guesses, but the exact sizes are irrelevant).

We call these atoms *cells*. We expect the space to be entirely occupied by morphologically identical cells. So there exists a finite number *w* of cells, that is, of minimal space-time units. Likewise, time is divided into discrete minimal units, the instants: t_0, t_1, ... (speaking algebraically: time is a discrete linear order). Our intuitive, everyday Euclidean space is three-dimensional. But a discrete universe can be shaped for computational purposes in 1, 2, 3, ..., *n*-dimensional spaces. To model our universe, we have chosen a two-dimensional, hexagonal grid (but our results can be obtained also by implementing our MsoR in a more traditional grid of squares, and in a three-dimensional environment with the three-dimensional analogue of hexagon: *rhombic dodecahedron*).

Hexagon and rhombic dodecahedron have various topological advantages in the representation of physical movement – specifically, the distance between cells can be approximated in terms of radius:

Fig. 1: rhombic dodecahedron

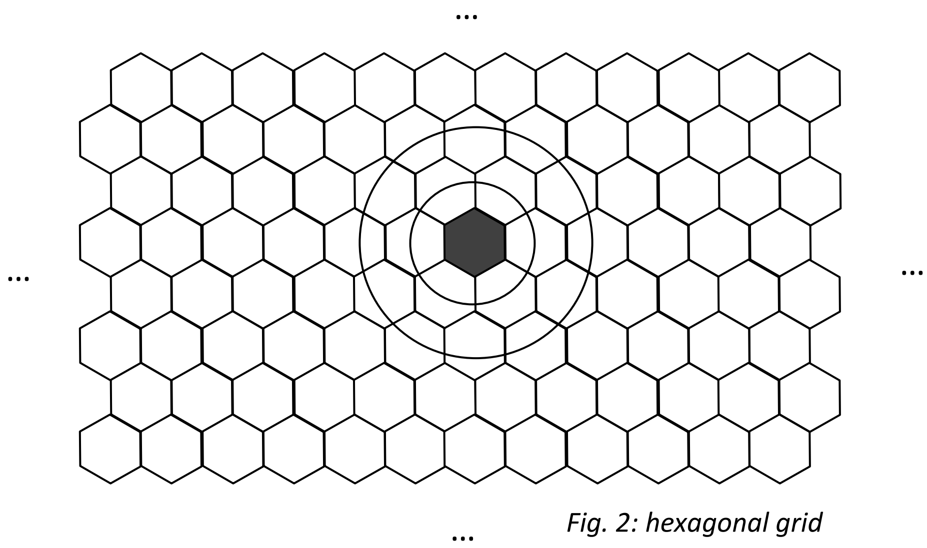

Fig. 2: hexagonal grid

Once a spatial basis has been fixed, each cell in a bi-dimensional frame is univocally individuated as a point in a lattice by an ordered couple of integers, <*i*, *j*>. Next, at each instant *t*, each cell, <*i*, *j*> instantiates exactly one state $\sigma \in \Sigma$, with Σ a finite set of states of cardinality $|\Sigma| = k$. Let "$\sigma_{i, j, t}$" denote the state of cell <*i*, *j*> at time *t*.

This is our strictly conventionalist perspective: first, we believe that the huge variety of worldly objects with their properties, qualities, and features surrounding us emerges as a high-level by-product of these simple ingredients: atomic cells and their few basic states. Second, anything whatsoever is ultimately an aggregate of cells. Call *system* any such aggregate. Then any system is just as legitimate as any other, our ordinary objects being just the aggregates that match with our ways of carving reality – and these depend on our cognitive apparatus, our learning capacities, and our practical interests.

The universe does not work randomly: rules determine how each point in the lattice *updates* its state. We don't know what the basic rules are, but we know for sure that they have to be *models of reference*: deterministic sequences of inputs, elaborations, and outputs. Next, our bet is that, at the bottom, rules must be few and simple: complexity and variety should *emerge* at higher levels, and depend upon the underlying simplicity: *simplex sigillum veri*.

There are no mysterious "actions at a distance" in the universe, but just local interactions: each point <*i*, *j*> interacts only with the six adjacent cells, called its neighbourhood:

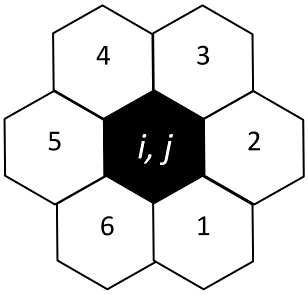

Fig. 3: a cell and its neighbourhood

Let us label "[*i*, *j*]" point <*i*, *j*>'s neighborhood. Then, some deterministic dynamic MoR-rule governs the atoms – and thus, the world: at each instanct *t*, each point <*i*, *j*> synchronically updates its state with respect to instant *t*-1, following the unique MoR $\phi: \Sigma \Rightarrow \Sigma$, such that for each σ, <*i, j*>, and *t*:

$$\sigma_{i,\, j,\, t+1} = \phi(\sigma_{[i,\, j],\, t})$$

Our world hosts a *globally finite amount of information*: given $|\Sigma| = k$ and a number *w* of points in the lattice, we have at most k^w global configurations for U. Therefore, the entire evolution of our universe U is a finite global transition graph, G_Φ - the graph of the global transition function $\Phi: \Gamma \Rightarrow \Gamma$ (with Γ the phase space or set of global configurations of U) induced by the MoR ϕ.

2. The iLabs World

2.1. Perfect Reversibility for Hi-Tech Computation

The dynamical laws of physics are reversible at micro-level: distinct initial states in a microphysical system always lead to distinct final states. It is likely that any formal model aiming at capturing computations that actually go on down there, at the bottom of reality, must host a *reversible* dynamics.

On the other hand, irreversible *computation* is an impractical waste of energy. An AND gate gives us 0 as its output at *t*+1. What was the input at *t*? 0 and 1, or *vice versa*, or two zeros? As von Neumann already conjectured, this informational entropy costs ~$3 \cdot 10^{-21}$ joules per elementary computational step at room temperature. The loss of information has a thermodynamic cost, to be paid in terms of a loss of non-computational energy. This does not depend on inefficient circuit design: it follows directly from the existence of irreversible calculation. As technology advances, this informational entropy problem is to put pressure on us: we may need reversible computing devices sooner than we thought.

iLabs have developed a mathematical model (a finite state cellular automaton) displaying a perfectly *reversible* dynamics and capable of conserving the *totality* of the information stored at the beginning of the universe. Next, we have effectively proved that our automaton is capable of *universal computation*: our discrete, computational universe can host universal Turing machines, capable of computing (given Turing's Thesis) anything which can be computed. This hinges on each single cell of our universe's being capable of instantiating a single MoR as a computational primitive. Our model is thus a good candidate for the realization of high performance computational devices: systems capable of hosting logical circuits that perform computations with internal energy dissipation virtually close to *zero*.

2.2. The iLabs Rule

Each point of the lattice of the iLabs model instantiates states wich are *sextuples* of bits in the set {1, 0}. Intuitively, think of them as implemented in the sides of an hexagonal cell:

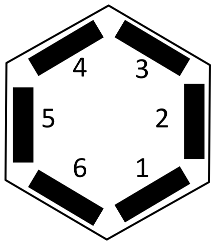

Fig. 4: the six sides of a cell instantiating a sextuple

Each side gets value 1 ("active") or 0 ("inactive") at each point of time: <*i*, *j*> at *t* has state $\sigma_{i, j, t} = \langle x_1, x_2, x_3, x_4, x_5, x_6 \rangle$, x_1 corresponding to side 1, x_2 to side 2, etc., each x having value $v \in V = \{1, 0\}$. Since $|V| = 2$, $|\Sigma| = 64$: each cell has $2^6 = 64$ possible states.
The neighbourhood [*i*, *j*] of point <*i*, *j*> is fixed by the value $v \in V$ of the six adjacent sides:

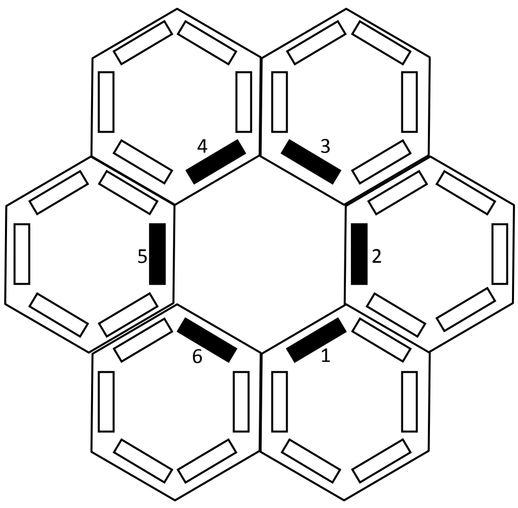

Fig. 5: the neighbours' adjacent sides

Still intuitively: think of each atom or cell as *perceiving* (input) the output exhibited by the neighbourhood, and *acting* (output) by exposing the result of its processing, as fixed by the following MoR ϕ: $\{1, 0\}^6 \Rightarrow \{1, 0\}^6$, which is but a conditional routing of signals:

$$\sigma_{i,j,t+1} = \begin{cases} Perm(\sigma_{[i,j],t}), \text{ if } \sum v_{[i,j],t} (\text{mod } 2) = 1 \\ Id(\sigma_{[i,j],t}) \text{ otherwise} \end{cases}$$

"$\sum v_{[i,j],t}$ (mod 2)" being the sum modulo 2 of the $v \in \{1, 0\}$ of each member of the input sextuple at t. *Id* is just *identity*, mapping each sextuple $\sigma \in \Sigma$ to itself:

$Id(<x_1, x_2, x_3, x_4, x_5, x_6>) = <x_1, x_2, x_3, x_4, x_5, x_6>$.

Perm is a permutation on Σ, that is, an operator $\Sigma \Rightarrow \Sigma$ exchanging the first three items with the last three ones:

$Perm(<x_1, x_2, x_3, x_4, x_5, x_6>) = <x_4, x_5, x_6, x_1, x_2, x_3>$.

An odd number of 1s in the input sextuple gives their permutation as an output: any incoming signal is transmitted by point $<i, j>$ to its opposite side (this is just aid to intuition: the cell's "sides" are encoded in the computation). An even number of 1s turns on *Id*, and the effect is a kickback: signals (still speaking intuitively) have to go back where they came from.

ϕ is an (albeit peculiar) outer-totalistic cellular automaton rule, securing *a total conservation* of the amount of "active" and "inactive" bits the universe starts with: since *Perm* and (trivially) *Id* are permutations, the output of any input sextuple will retain the number of 0s and 1s in the input.

Furthermore, ϕ is a *reversible* rule: each input or perception is mapped by the two sub-rules of ϕ, *Id* and *Perm*, to a distinct output or action. So there exists an inverse MoR, ϕ^{-1}, mapping ϕ's outputs to the respective inputs.

But ϕ is also *strongly* reversible, that is, $\phi^{-1} = \phi$. For ϕ is time-reversal invariant: it is mapped to ϕ^{-1} by the time-reversing transformation: $t \mapsto -t$. This means that at any point in time t, we can recover any state of the universe U at t-n (up to the starting configuration of U – the beginning of the universe) by running the system backwards on the basis of ϕ itself.

2.3. Universal Computation

MoR ϕ is *computation-universal*: our discrete universe U can host universal Turing machines (UTM) capable of universal computation. A constructive proof of this can be found in the iLabs book [forthcoming], where it is shown that each primitive of computation of a conventional UTM, namely (a) the storage, (b) transmission, and (c) processing of signals via a functionally complete set of logical gates, can be emulated by patterns produced via ϕ in the relevant cellular automaton.

Specifically: by implementing ϕ, each point <*i*, *j*> in the lattice constituting our discrete universe performs the double function of signal storage and transmission, so that any chain of adjacent cells works as a wire along which bits of information can move.

Additionally, each cell *is* itself a universal logical gate, precisely insofar as it is a locus for the transmission and redirection of signals, by implementing the functionally complete set {AND, OR}, and also the FAN-OUT of signals (again: see our [forthcoming] book for details).

Calling $\bar{\sigma}_0$ the vector expressing the initial configuration of U (the ordered *w*-tuple of the states of each of the *w* atoms in the grid at the beginning of the universe), the evolution of U is prescribed by ϕ (with the corresponding global rule Φ). And such an evolution can produce UTMs, and so implement any finite algorithm and evaluate any computable claim.

Since our discrete universe can host UTMs, its evolution is *unpredictable* in an exact sense: given the Halting Theorem, there is no general algorithm capable of predicting whether a given UTM will halt after *n* steps given a certain input. So there is no computational short path to predict the evolution of U: you can only sit back and look.

3. MoR as concepts

To think of our MsoR as triples <perception, thought, action> can sound, at first, as a kind of "semantic animism". However, MsoR can recapture the general notion of *operator*, to which most mathematical concepts can be reduced. The mathematics of the MsoR is a "mathematics of thought" insofar as it provides a formal, mathematically respectable way to make many cognitive notions precise. The very notion of *concept* can be dealt with satisfactorily via the theory of MsoR.

The nature of concepts has been the subject of philosophical disputes for some two thousand years: concepts have been sometimes reduced to physical or mental entities, sometimes elevated to the status Platonic ideas, or Fregean *Sinne*, etc. iLabs believe that concepts just *are* MsoR: rules effectively mapping a given signal or perception or input to an output or action following an internal processing activity, or thought. Or, more modestly: by accepting to conventionally identify the traditional notion of concept with the one of MoR, and by formally developing the latter, we can gain insight on the former. MsoR are "concepts at work": they fulfil the theoretical roles assigned to concepts, and provide precise physical-informational realizers for them.

Treating MsoR as operators allowed iLabs to import in the mathematics of MsoR the traditional theory of computable functions. In the following, we provide some appetizers of the full-fledged theory.

3.1. Equivalent MoR

MsoR have clearly spelled *equivalence conditions*. Let $\approx$ be a generic equivalence relation determining a partition of the set of all MsoR into equivalence classes. We can introduce criteria or conditions of equivalence between MsoR via axioms instantiating the schematic form:

(EC) $(\alpha[x/f] \leftrightarrow \alpha[x/g]) \rightarrow f \approx g$

α being our chosen condition. Two given MsoR f and g can be taken as equivalent under the relevant $\approx$ if they both satisfy the chosen α.

A minimal condition on any equivalence relation on MsoR (ME) is the following. In order for f and g to be minimally equivalent ($f \approx_0 g$), it is necessary that they share the same set of perceptions and actions. The sufficient condition for $f \approx_0 g$ is that they that map the same perceptions or inputs to the same actions or outputs:

Given any two f: A $\Rightarrow$ B, and g: C $\Rightarrow$ D, if:

(a) A = C and B = D,
(b) For any input $x_1, x_2, \ldots, x_n$, f and g give the same output y,

Then, $f \approx_0 g$.

(ME) is still a week criterion, not including the time factor. A temporally qualified equivalence, $\approx_1$, will add to (a) and (b) the following clause:

(c) … And output y is produced by f and g after the same number n of time units,

Then, $f \approx_1 g$.

Stricter equivalences can be obtained by adding further clauses. We can capture procedural isomorphisms between MsoR: now we want f and g to be equivalent ($f \approx_2 g$) iff they "do the same things", not only "by employing the same amount of time", but also "via isomorphic thoughts", that is, via computational procedures, P_f and P_g respectively, such that there exists an isomorphism i between P_f and P_g. Hence, we need a fourth clause with the following form:

(d) … And output y is produced by f and g via two computational sequences P_f and P_g, such that $i(P_f, P_g)$,

Then, $f \approx_2 g$.

A good mathematical characterization of i is provided in the iLabs [forthcoming] book by putting the computations performed by a set of recursive MsoR in canonical form, and by arithmetizing them via standard encoding procedures: the required i is then defined on the relevant numeric codes.

4. Recursive MoR, Meta-Models

Intuitively, a *recursive* MoR is one that "refers to itself" in its very definition. Slightly more precisely: a recursive MoR is such that its actions or outputs given certain perceptions or inputs are determined by the output of *that very* MoR with respect to simpler inputs or perceptions: what a recursive MoR does given some more complex perceptions depends on what *it* does, or would do, for simpler ones (and, as shown in detail in our book, also "simpler" can be characterized mathematically in a precise way). Recursive MsoR can therefore be introduced via standard recursive definitions: the operator in the *definiendum* recurs in the *definiens*.

By fully recapturing the theory of recursive operators, the mathematics of the MsoR allowed iLabs to define the key notion of *meta-model*. Informally, a meta-model is simply *a model of reference capable of perceiving other models of reference and of operating on them*. Technically, this has been achieved by having MsoR themselves encoded as states of the cells inhabiting our digital universe U: a meta-model can then take as inputs the codes of the MsoR it perceives.

By embedding Kleene's (Strong) Recursion Theorem in the theory of MsoR, it was then proved that there can be a *universal MoR*, capable of emulating the thoughts of any recursive MoR. This is a (partial) MoR with the following form:

$$univ(e, <x_1, \dots, x_n>).$$

Given a recursive n-ary MoR $f(x_1, \dots, x_n)$ with code e, that is, $[e]_n(x_1, \dots, x_n)$, *univ* takes as inputs the code of f and (the code for) its input, and provides as output the one f or $[e]_n$ would give:

$$f(x_1, \dots, x_n) \cong [e]_n(x_1, \dots, x_n) \cong univ(e, <x_1, \dots, x_n>).$$

(again, details can be found in our book). *univ* obviously corresponds to a UTM and, given the results presented in Section 2, it can be implemented in our digital universe U.

5. Recursive Self-Reference and Beyond

Recursive self-reference takes place when a MoR not only refers to itself, but *is aware* of such a self-reference. A system implementing a self-referentially recursive MoR can therefore be aware of what it does via that very MoR.

Again, this is not semantic animism. For such expressions as "being aware" can be given a precise mathematical meaning. Much research at iLabs is guided by the persuasion that recursive self-reference is at the basis of what people ordinarily call "consciousness": a key difference between conscious thoughts and any other computational procedure is that our mind, as (self-)conscious, can think about and have a viewpoint on itself (albeit with arguably limited powers to operate on its own source code). If Artificial Intelligence is to be real, this will be achieved by means of recursive self-reference – or this is our bet.

As shown in our book, the Recursion Theorems, applied to our recursive MsoR, guarantee that we can define partial MsoR which are recursively self-referential, for they *include their own code in their recursive definition*. These are simply classical fixed-point definitions. Since numeric codes are perceptions taken as inputs by (meta-)models of reference, which can also emulate the thought procedures performed by the encoded MsoR, recursive self-referential MsoR can perceive themselves in a precise mathematical fashion, and represent the computational procedure in which they consist within themselves.