

Introduction aux Neurosciences Computationnelles

Jonas Ranft
jonas.ranft@ens.psl.eu
Institut de Biologie de l'ENS

Nov-Dec 2023

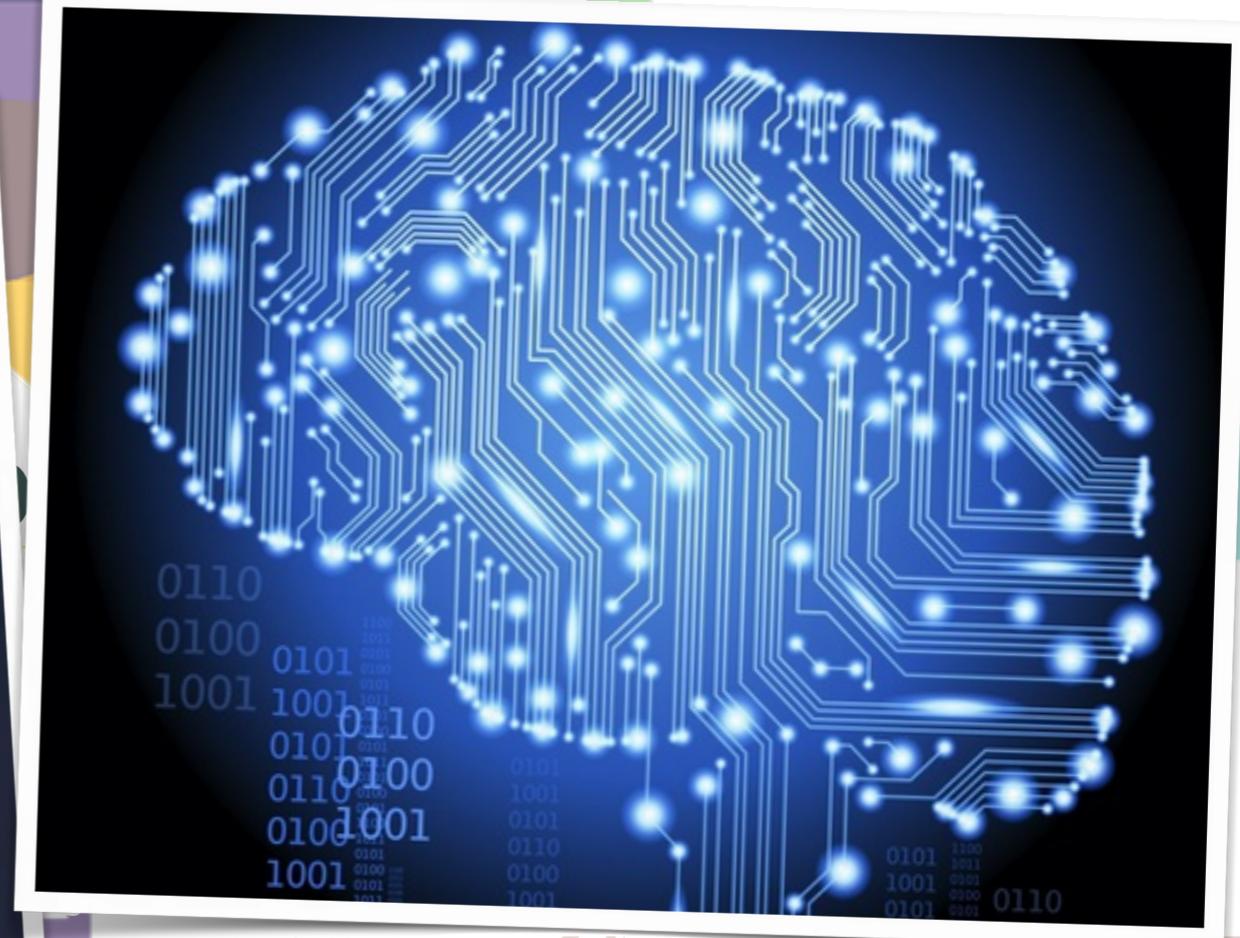


Préparation aux expos professionnelles



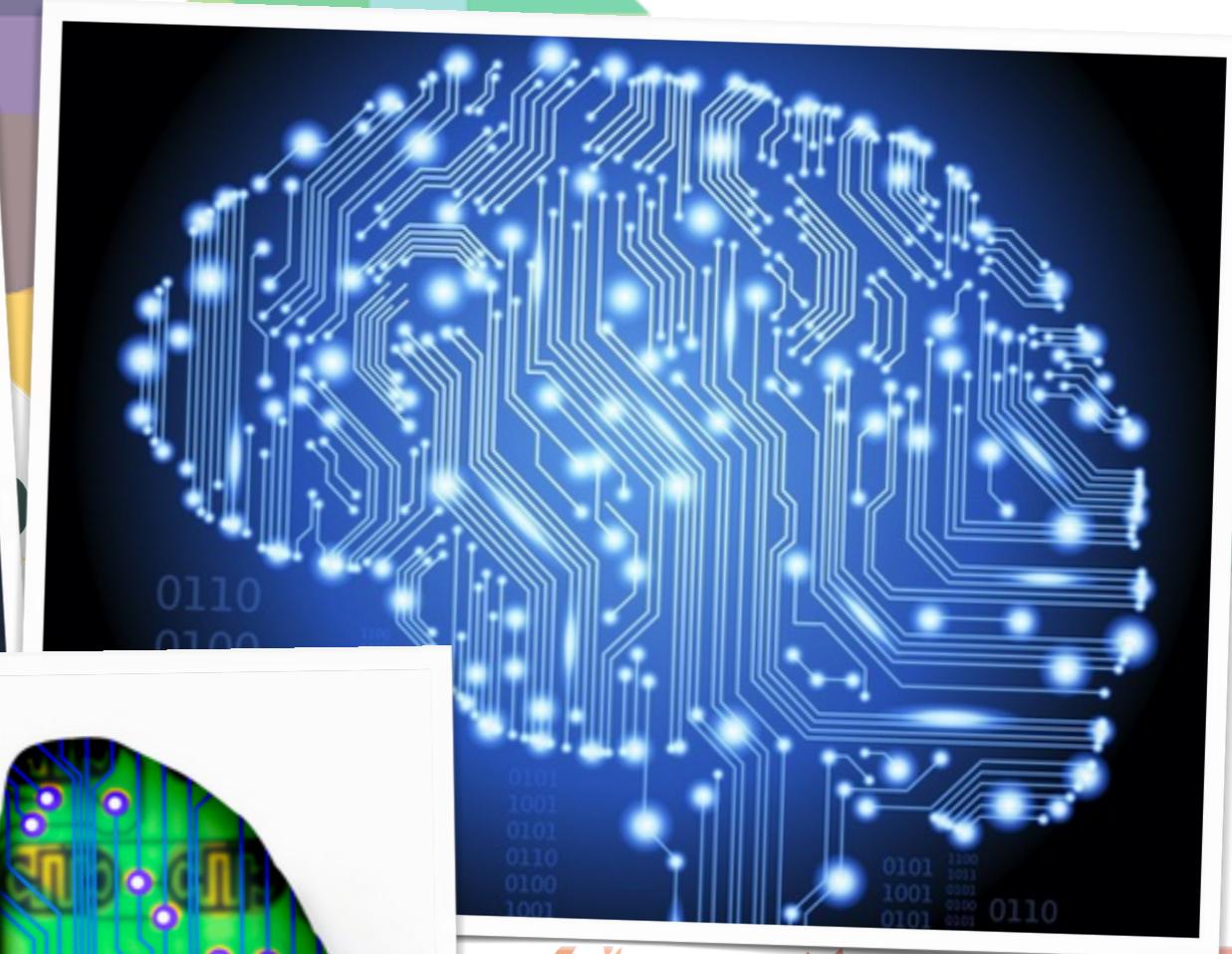
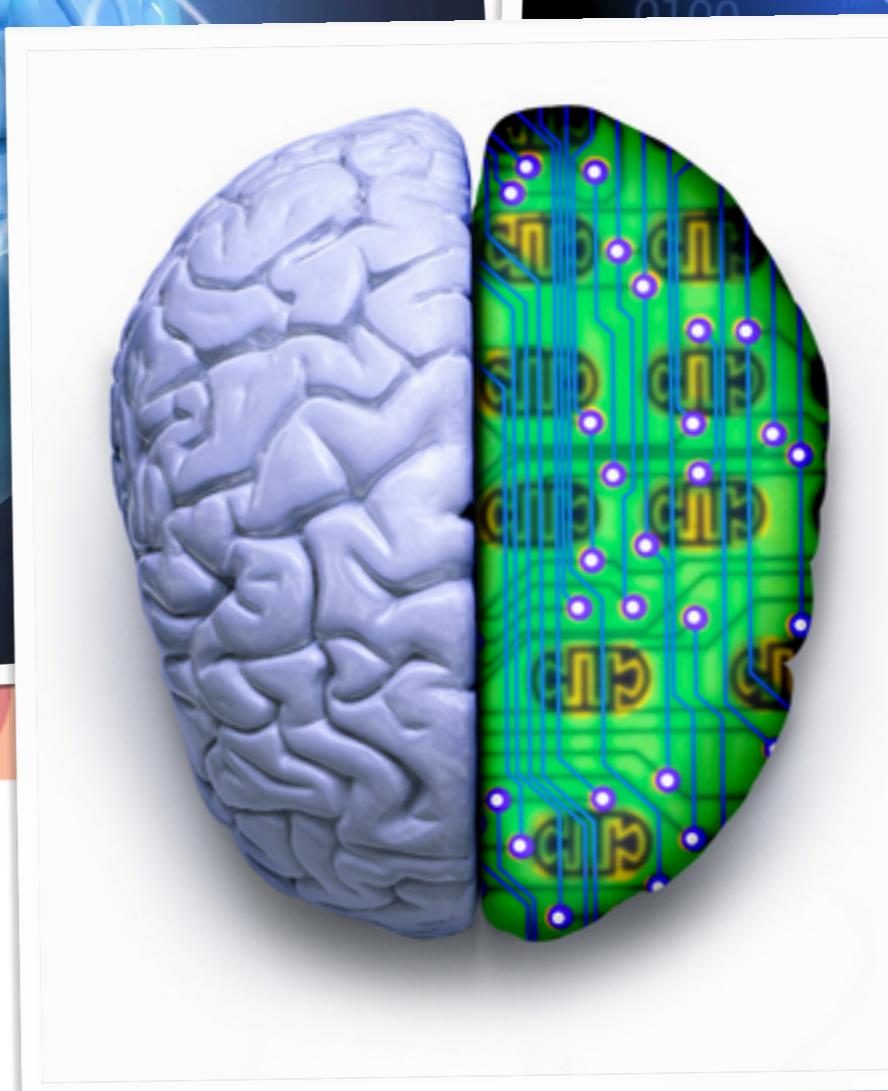
Jonas Ranft
jonas.ranft@ens.psl.eu
Institut de Biologie de l'ENS

Nov-Dec 2023



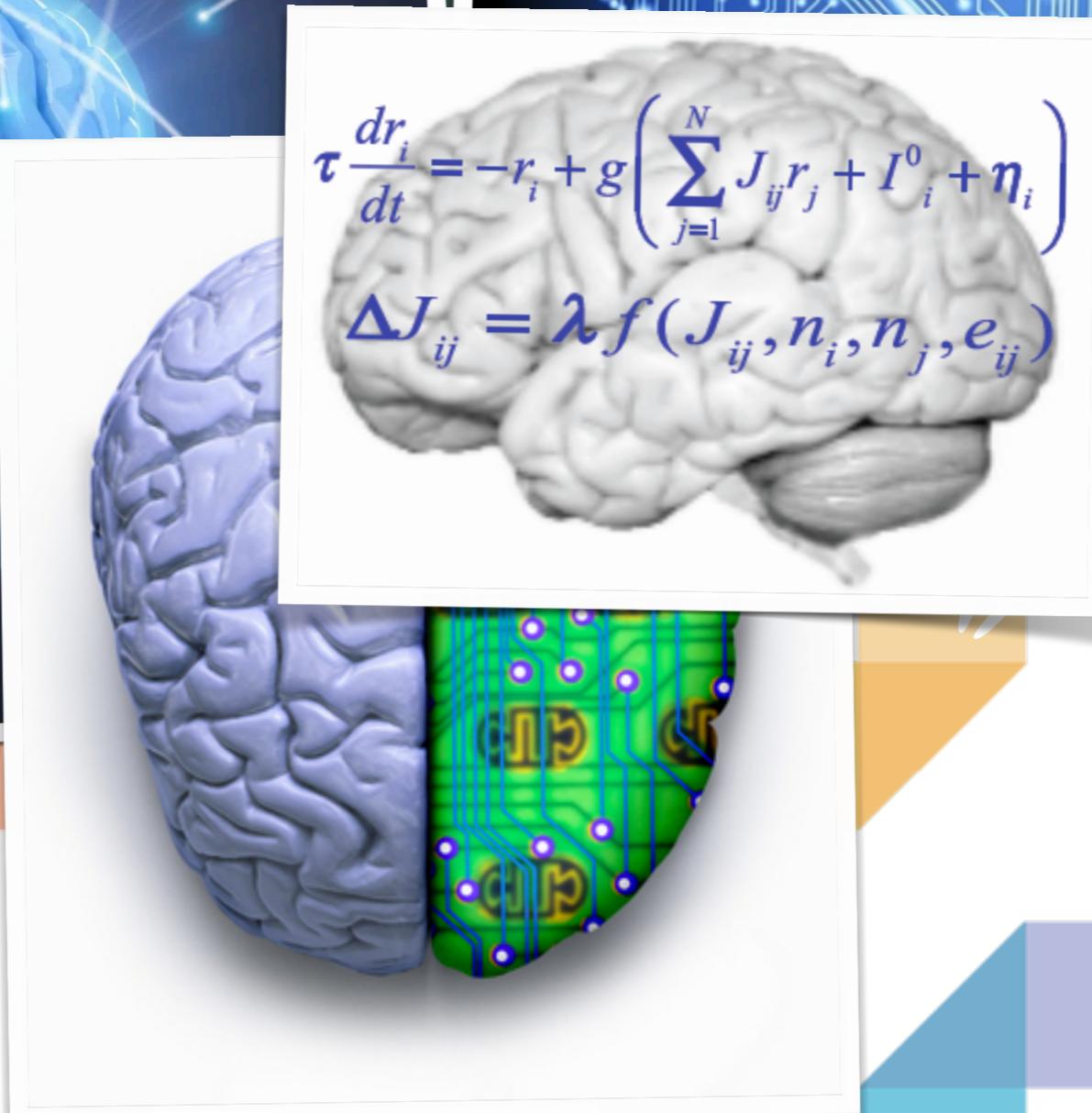
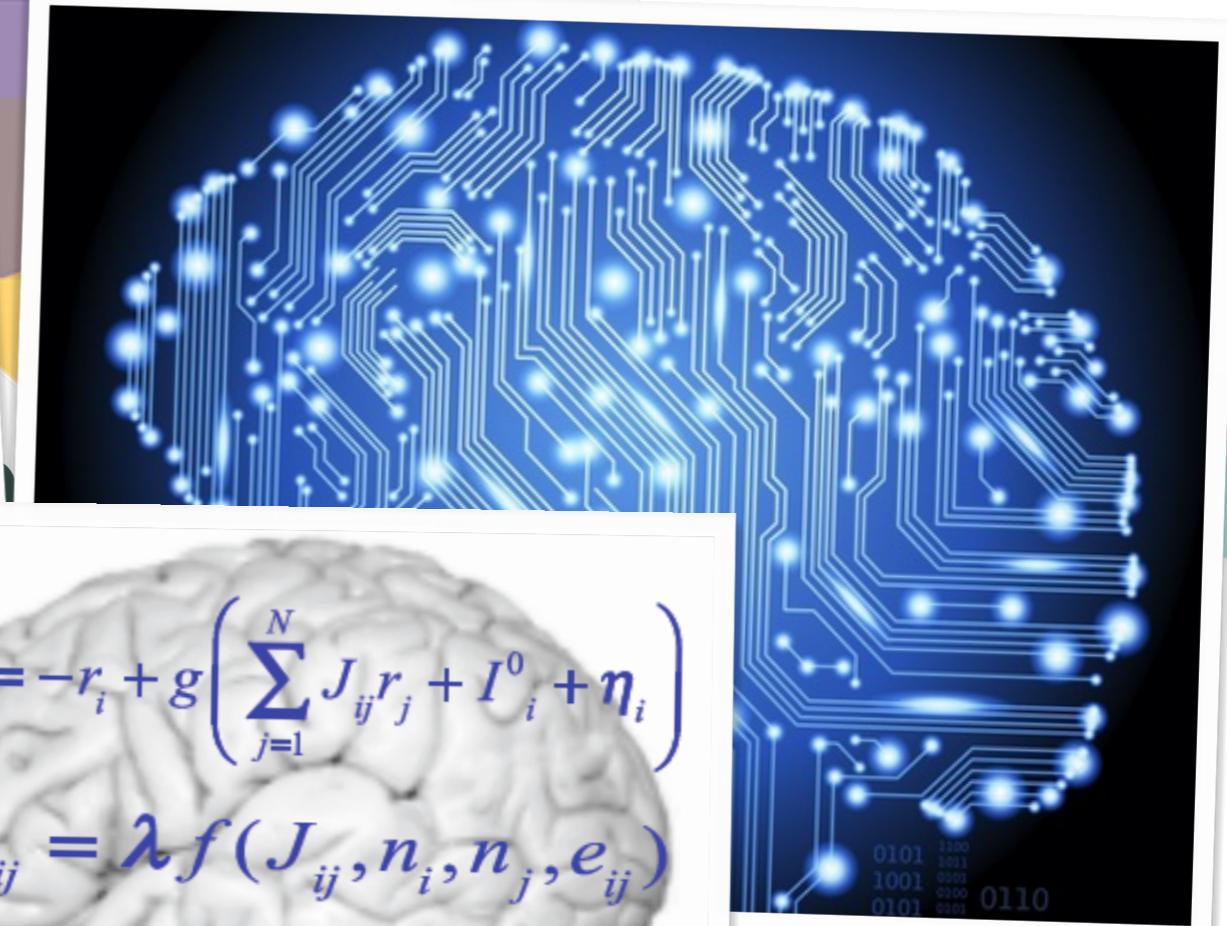
Jonas Ranft
jonas.ranft@ens.psl.eu
Institut de Biologie de l'ENS

Nov–Dec 2023



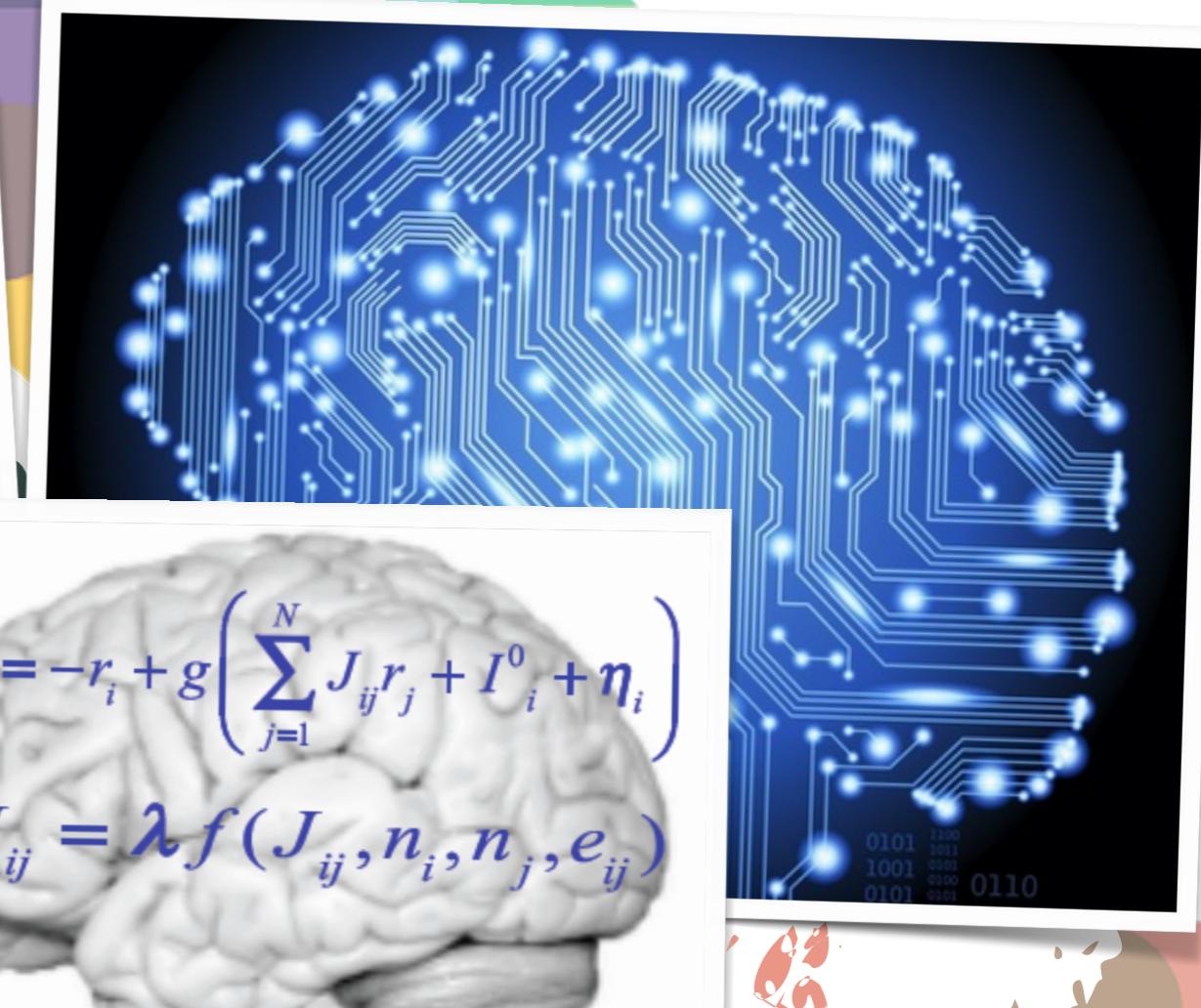
Jonas Ranft
jonas.ranft@ens.psl.eu
Institut de Biologie de l'ENS

Nov–Dec 2023



Jonas Ranft
jonas.ranft@ens.psl.eu
Institut de Biologie de l'ENS

Nov-Dec 2023



$$\tau \frac{dr_i}{dt} = -r_i + g \left(\sum_{j=1}^N J_{ij} r_j + I^0_i + \eta_i \right)$$
$$\Delta J_{ij} = \lambda f(J_{ij}, n_i, n_j, e_{ij})$$



Jonas Ranft
jonas.ranft@ens.psl.eu
Institut de Biologie de l'ENS

Nov-Dec 2023

But de ce cours

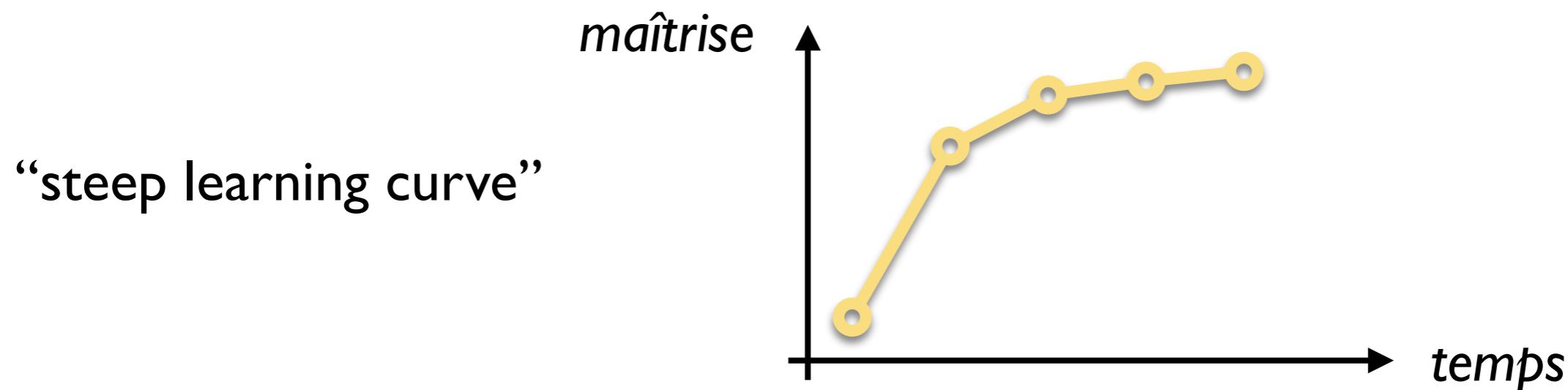
Premier contact avec les neurosciences computationnelles !

- Apprécier que les *modèles mathématiques*, *l'analyse théorique* et *les abstractions* sont nécessaires pour comprendre le cerveau.
- La description théorique des neurones et des réseaux biologiques *nous renseigne sur leur fonction et leur dynamique*.
- Les modèles utiles doivent saisir les *caractéristiques pertinentes* du système biologique à l'échelle spatio-temporelle étudiée.

But de ce cours (bis)

Python : un langage de programmation facile et polyvalent !

- Apprentissage d'un langage simple mais puissant.
`In [1]: print("hello world!")`
`Out: hello world!`
- De nombreuses bibliothèques pour *l'analyse des données, le calcul scientifique, l'apprentissage machine*, ... entièrement libres et mises à disposition par une grande communauté de développeur.e.s et scientifiques.
- Beaucoup de documentation et aide disponible en ligne.



Plan des cours et TD : des neurones aux réseaux

- Quand : Le mercredi de 13h30 à 17h30
- Où : Cordier E
- Programmation interactive : <http://tiny.cc/introcompneuro>
- Matériel : <https://github.com/j-ranft/IntroNeuroComp2023>
- Évaluation : devoirs d'un cours à l'autre, petit examen à la fin

	Cours	TD
8 nov.	Introduction	Premiers pas avec Python
15 nov.	Le neurone I	Analyse de trains de spikes
22 nov.	Le neurone 2	Modèle Intégrer-et-Tirer
6 déc.	Réseaux neuronaux	Modèles de taux de décharge
13 déc.	Apprentissage et mémoire	Modèle de Hopfield
20 déc.	Apprentissage automatique	Perceptron

Informations pratiques

■ Intervenant :

Jonas Ranft

Institut de Biologie de l'ENS (IBENS)

46 rue d'Ulm, 75005 Paris

jonas.ranft@ens.psl.eu

■ Manuels :

- Dayan & Abbott, *Theoretical Neuroscience*, Cambridge 2001
(<http://www.gatsby.ucl.ac.uk/~lmate/biblio/dayanabbott.pdf>)
- Gerstner et al., *Neuronal dynamics*, Cambridge 2014
(<https://neuronaldynamics.epfl.ch/online/index.html>)

■ Distribution de Python :

<https://www.anaconda.com/products/individual>
(dernière version Python 3.x)



Leçon 1: Introduction

Jonas Ranft
Neurosciences computationnelles

Nov–Dec 2023

A quoi sert le cerveau ?

A quoi sert le cerveau ?



Arbre
pas de neurones

A quoi sert le cerveau ?



Arbre
pas de neurones



C. elegans
302 neurones

Le cerveau génère des mouvements
(= comportement).

A quoi sert le cerveau ?



Arbre
pas de neurones



C. elegans
302 neurones



Mouche
1 000 000 neurones

Le cerveau génère des mouvements
(= comportement).

Les cerveaux plus complexes
génèrent une plus grande variété
de comportements.

A quoi sert le cerveau ?



Arbre
pas de neurones



C. elegans
302 neurones



Mouche
1 000 000 neurones



Rat
1 000 000 000 n.



L'homme
10 000 000 000 000 n.

Le cerveau génère des mouvements
(= comportement).

Les cerveaux plus complexes
génèrent une plus grande variété
de comportements.

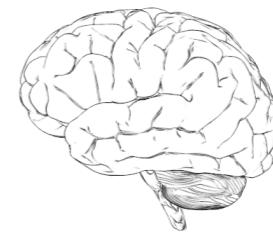


Les cerveaux plus
complexes peuvent
apprendre plus de
comportements.

Traitement cognitif



Qu'est-ce que le cerveau sait bien faire ?



échecs	1 : 0
scrabble	1 : 0
Jeopardy!	1 : 0
jeux vidéo	1 : 0
Go	1 : 0
réconnaissance d'objets	1 : 1

Qu'est-ce que le cerveau sait bien faire ?



échecs	1 : 0
scrabble	1 : 0
Jeopardy!	1 : 0
jeux vidéo	1 : 0
Go	1 : 0
réconnaissance d'objets	1 : 1

Les ordinateurs surpassent les cerveaux dans les tâches impliquant l'exploration de bases de données et d'algorithmes.

Qu'est-ce que le cerveau sait bien faire ?

Lionel Messi - Barcelona vs. Getafe, 2007



Qu'est-ce que le cerveau sait bien faire ?

Lionel Messi - Barcelona vs. Getafe, 2007



Qu'est-ce que le cerveau sait bien faire ?

RoboCup 2019



Qu'est-ce que le cerveau sait bien faire ?

RoboCup 2019



Qu'est-ce que le cerveau sait bien faire ?



foot



0 : 1



**des tâches
suffisamment
complexes**

0 : 1

Qu'est-ce que le cerveau sait bien faire ?



foot



0 : 1

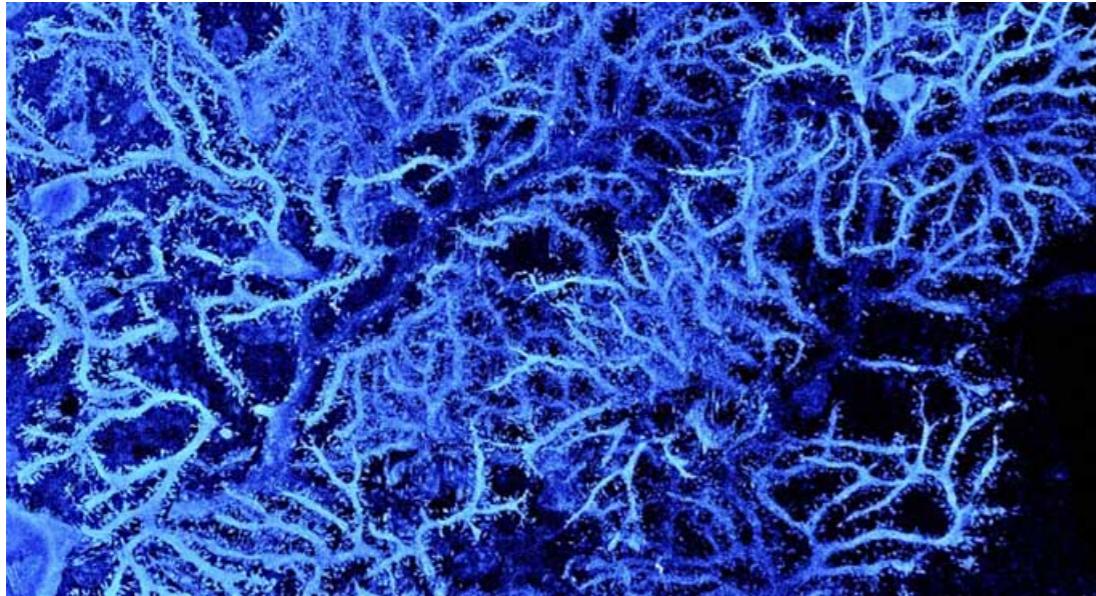


**des tâches
suffisamment
complexes**

0 : 1

**Les cerveaux sont meilleurs dans les tâches
impliquant une interaction avec le monde réel.**

Cerveaux : des dispositifs informatiques remarquables



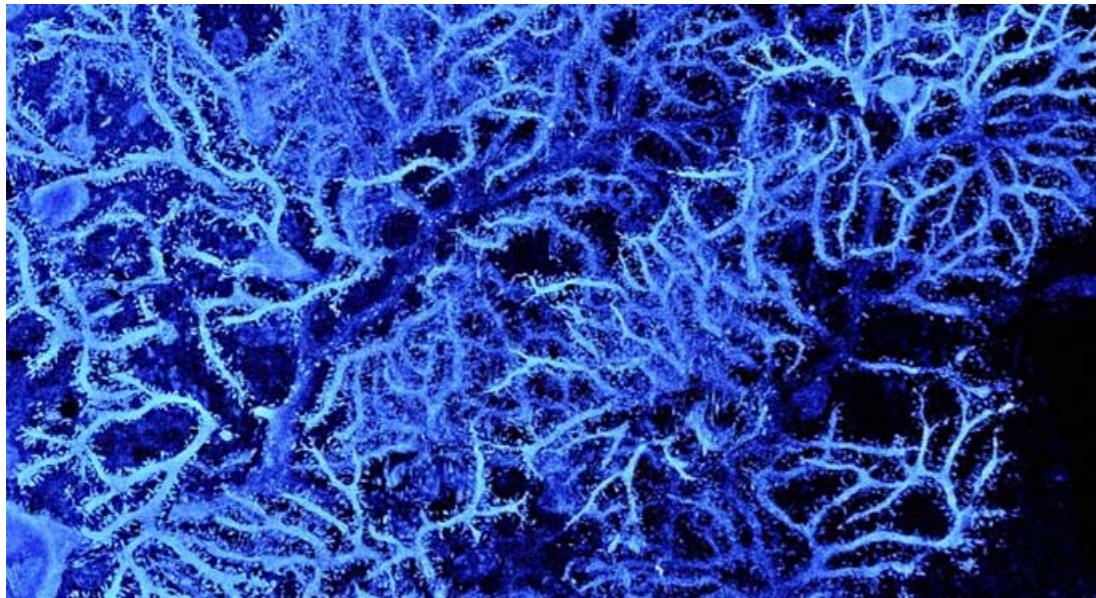
neurone : 10 kB de stockage

cerveau : 10^{10} neurones

cerveau : 100 TB de stockage

cerveau : 12 W de puissance

Cerveaux : des dispositifs informatiques remarquables



neurone : 10 kB de stockage

cerveau : 10^{10} neurones

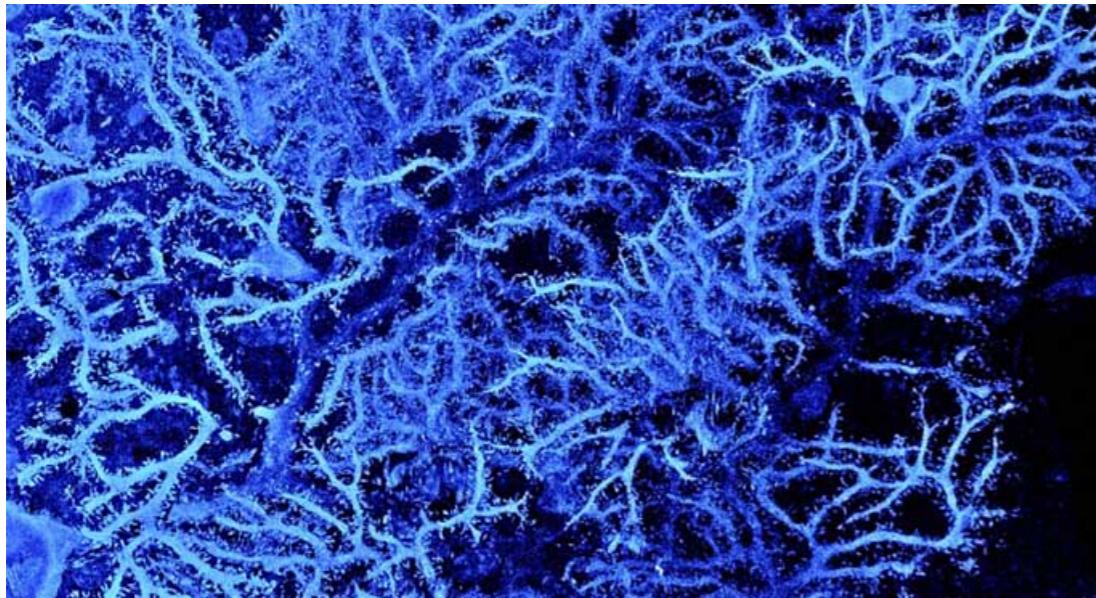
cerveau : 100 TB de stockage

cerveau : 12 W de puissance

ordinateur : 5 mW/GB

ordinateur @ 100 TB : 500 W

Cerveaux : des dispositifs informatiques remarquables



neurone : 10 kB de stockage

cerveau : 10^{10} neurones

cerveau : 100 TB de stockage

cerveau : 12 W de puissance

ordinateur : 5 mW/GB

ordinateur @ 100 TB : 500 W

Les cerveaux sont ~ 40 fois plus efficace sur le plan énergétique !

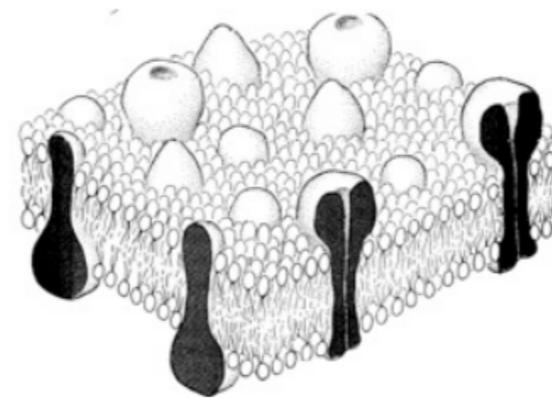
Pourquoi modéliser le cerveau ?

- ▶ Pour le comprendre
- ▶ Pour s'en inspirer
- ▶ Pour le soigner (pour l'améliorer ?)

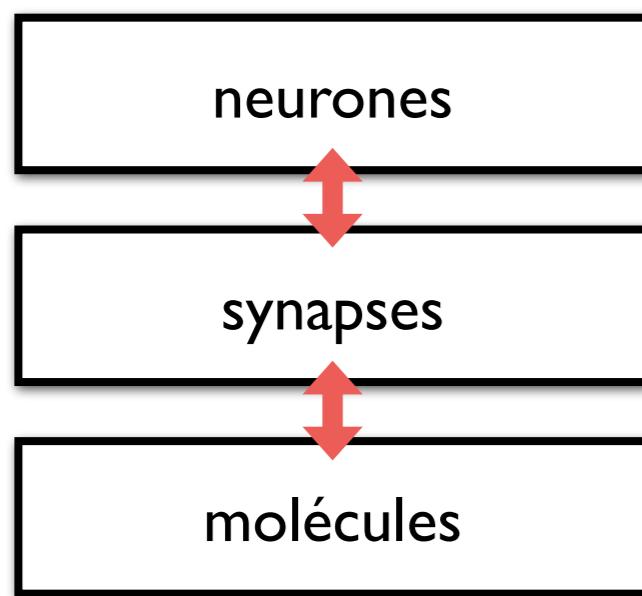
De quoi le cerveau est-il fait ?

molécules

1 nm



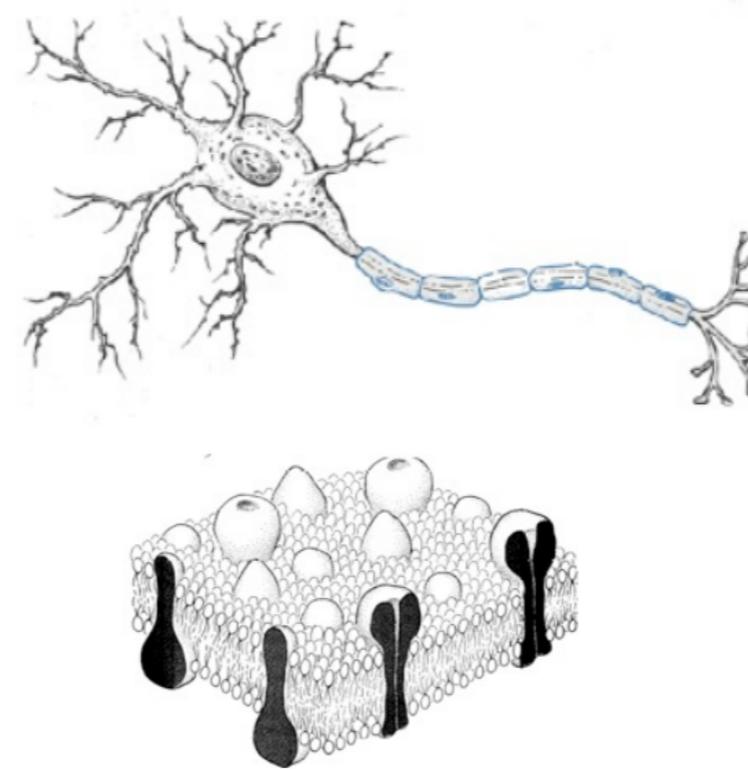
De quoi le cerveau est-il fait ?



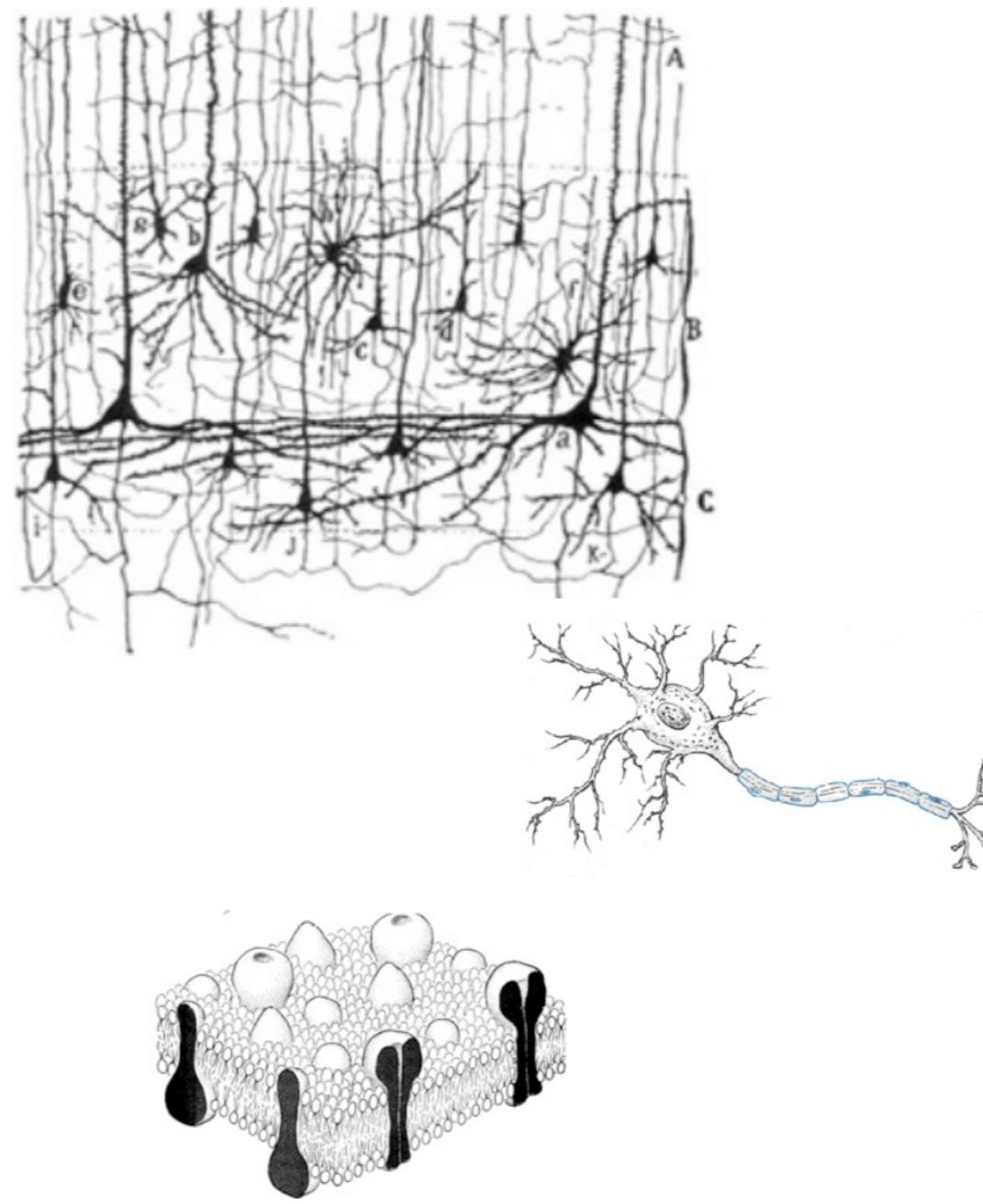
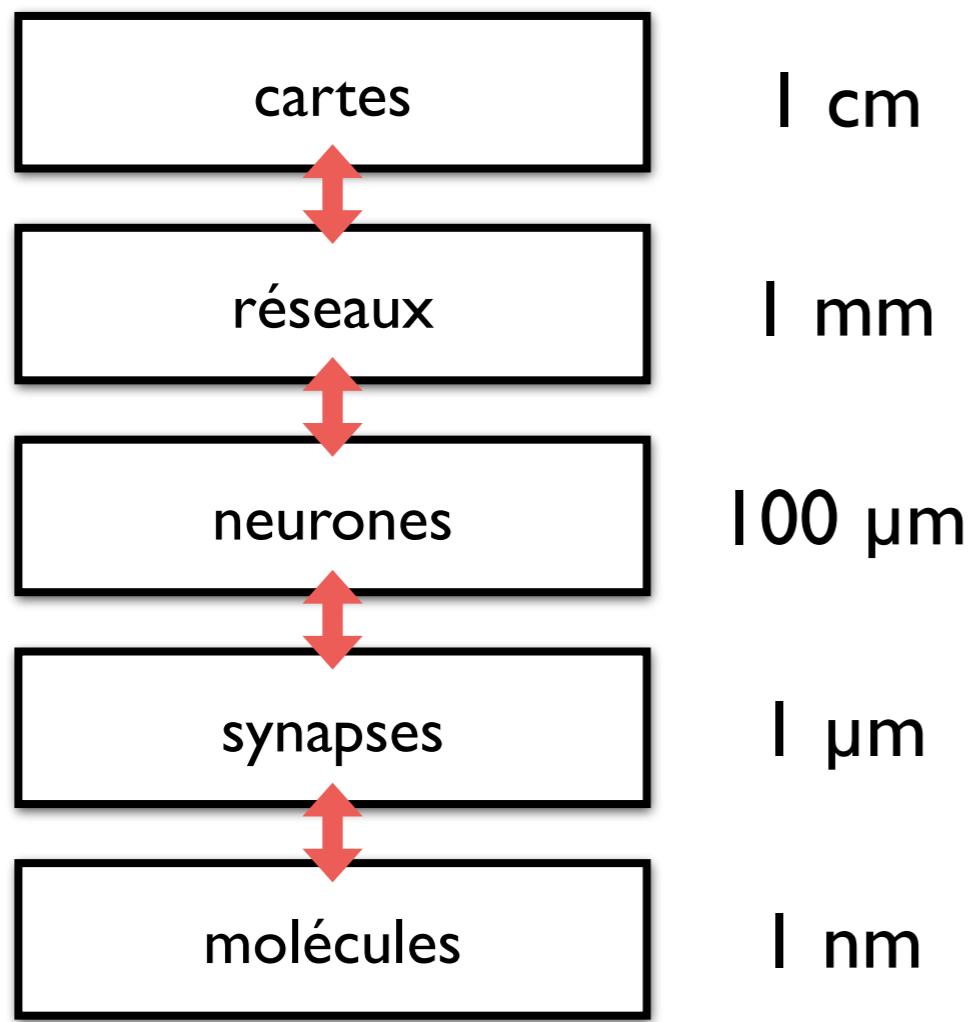
100 μm

1 μm

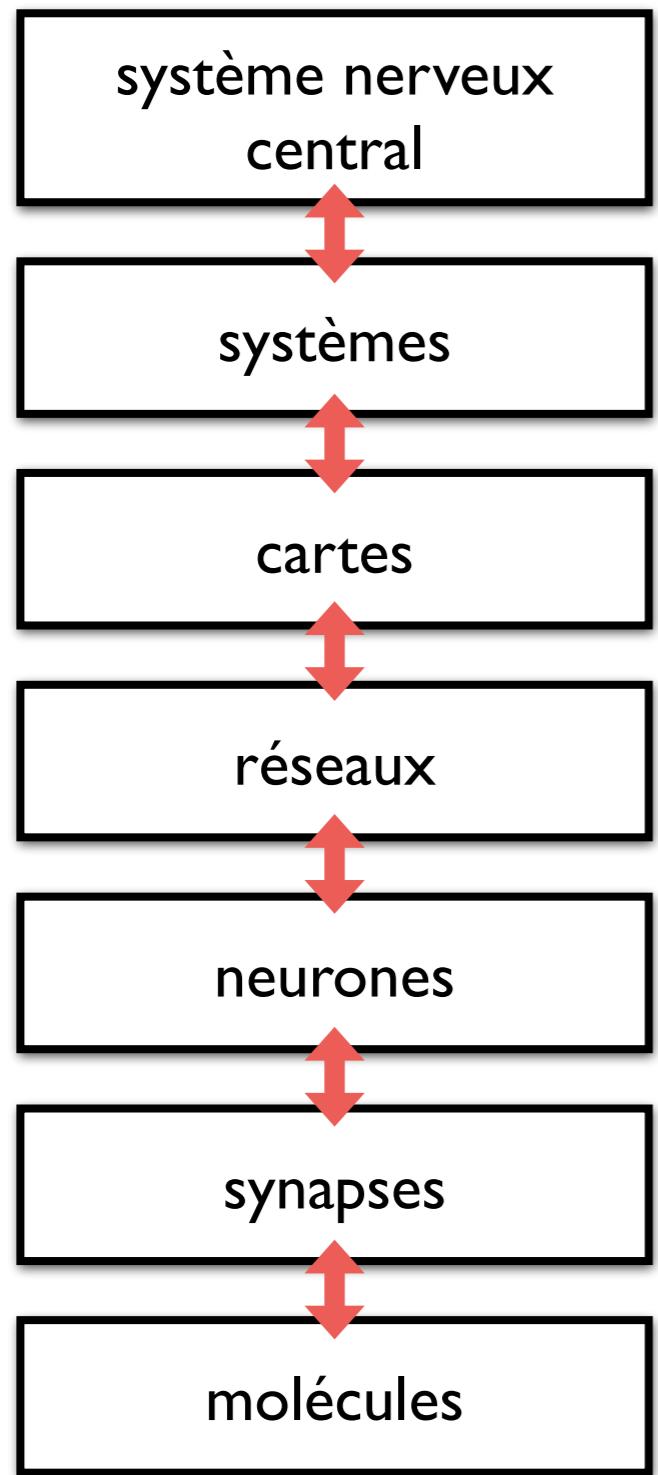
1 nm



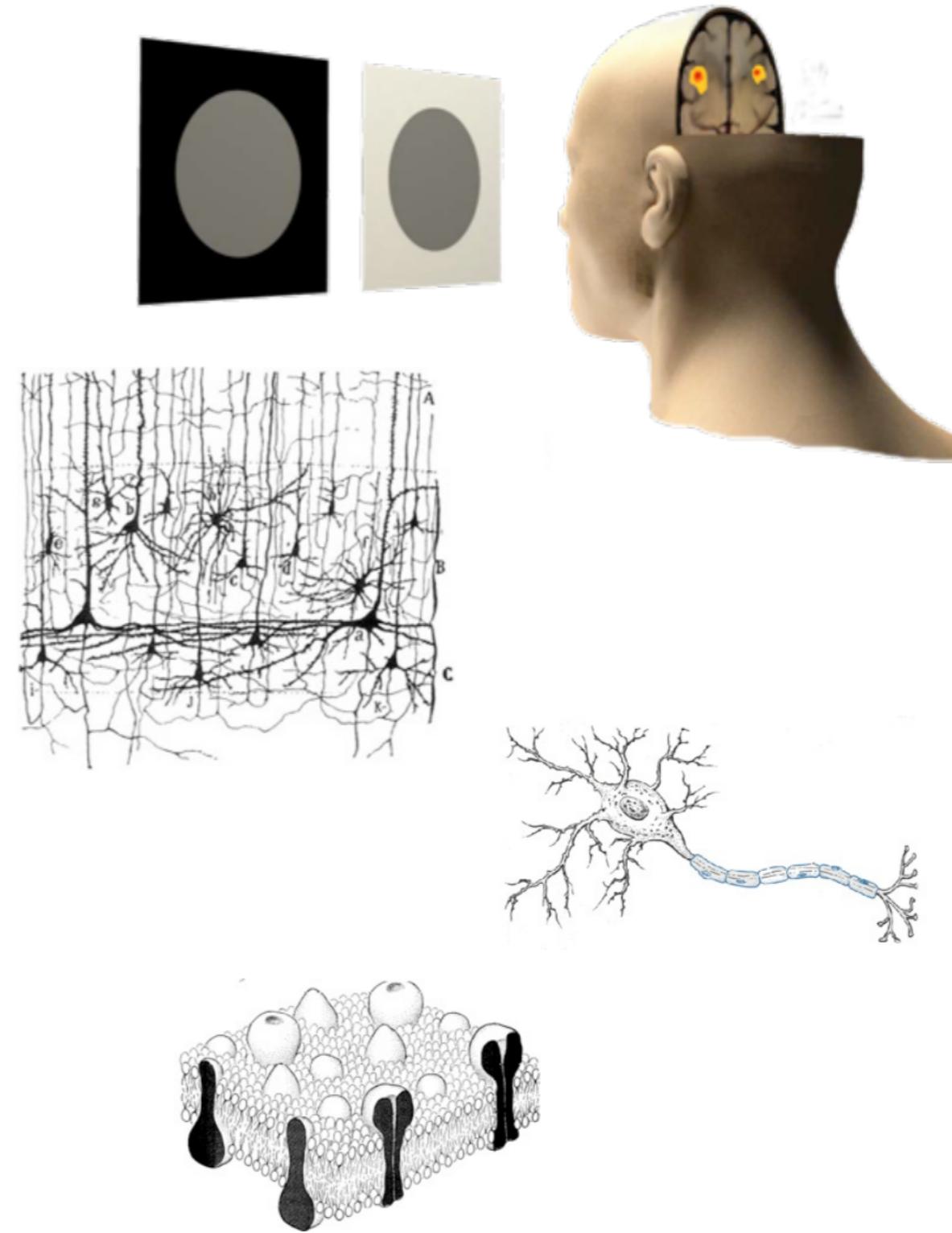
De quoi le cerveau est-il fait ?



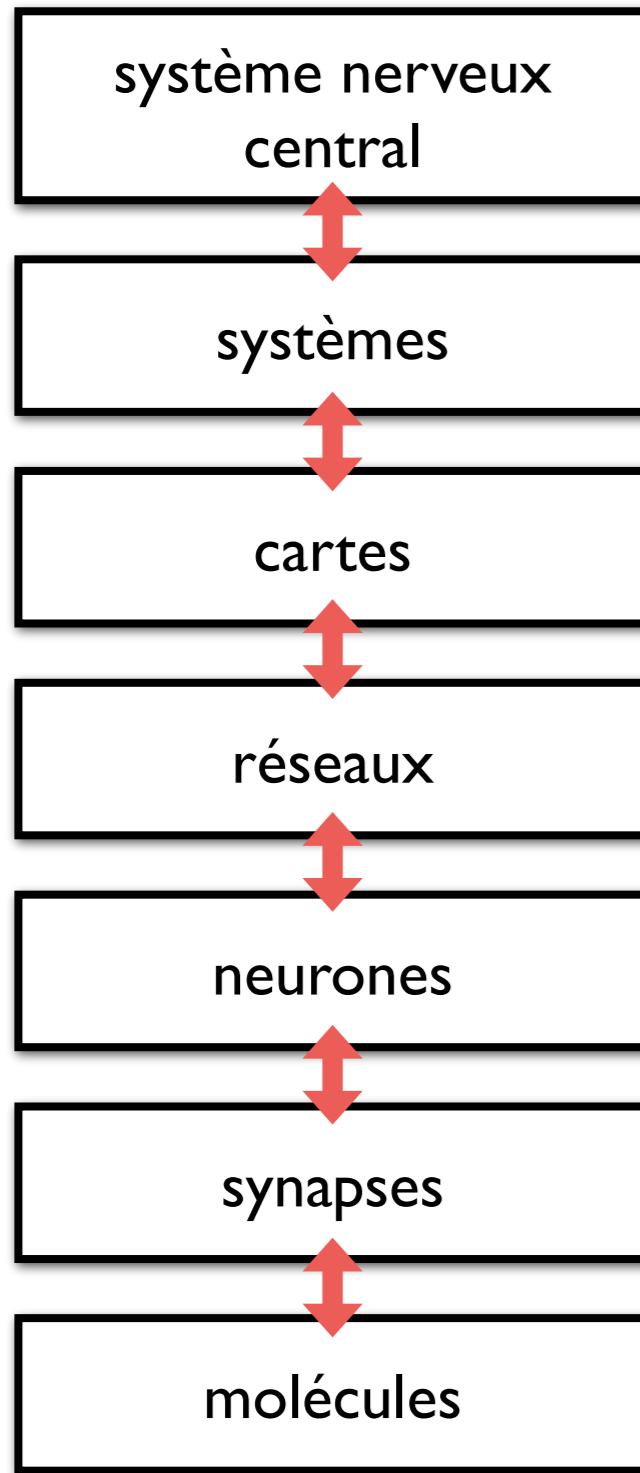
De quoi le cerveau est-il fait ?



1 m
10 cm
1 cm
1 mm
100 μ m
1 μ m
1 nm



De quoi le cerveau est-il fait ?



1 m

10 cm

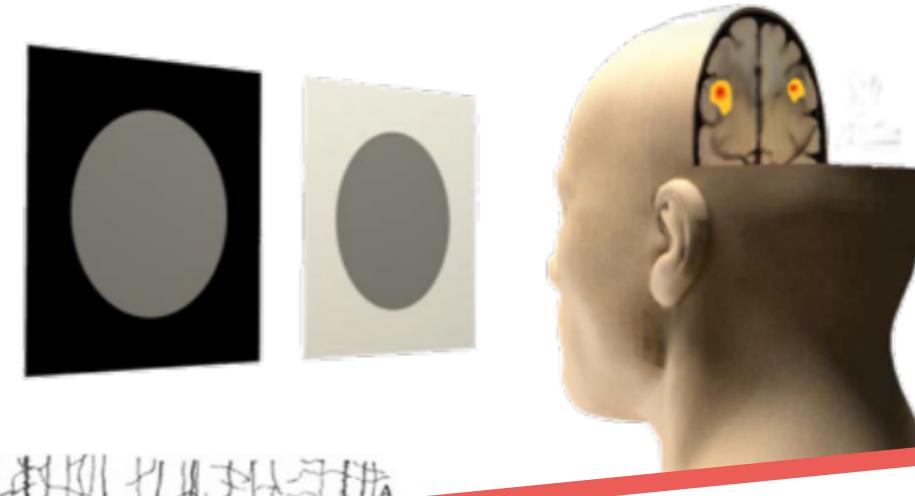
1 cm

1 mm

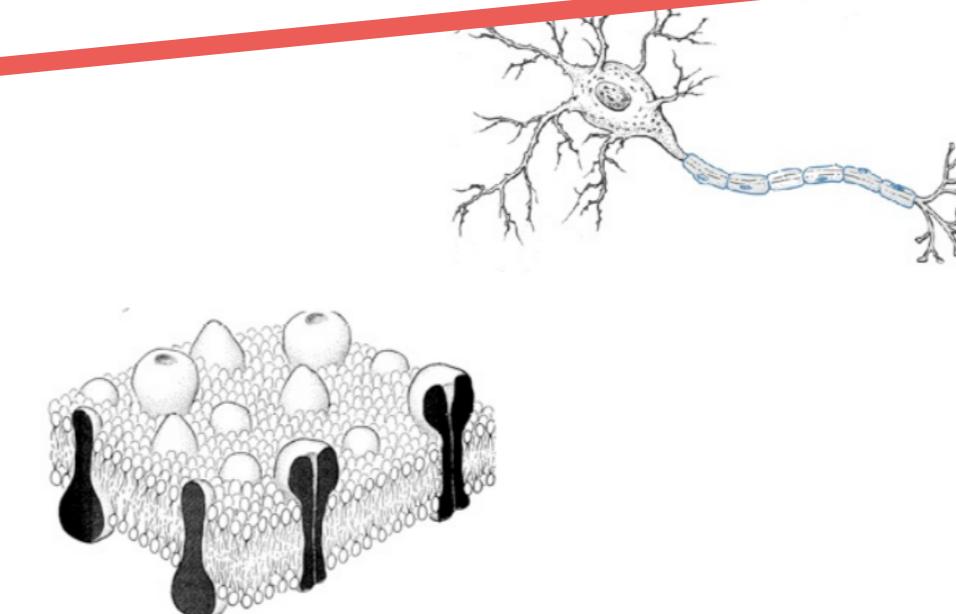
100 µm

1 µm

1 nm



**Comment est-ce
qu'il fonctionne ?**



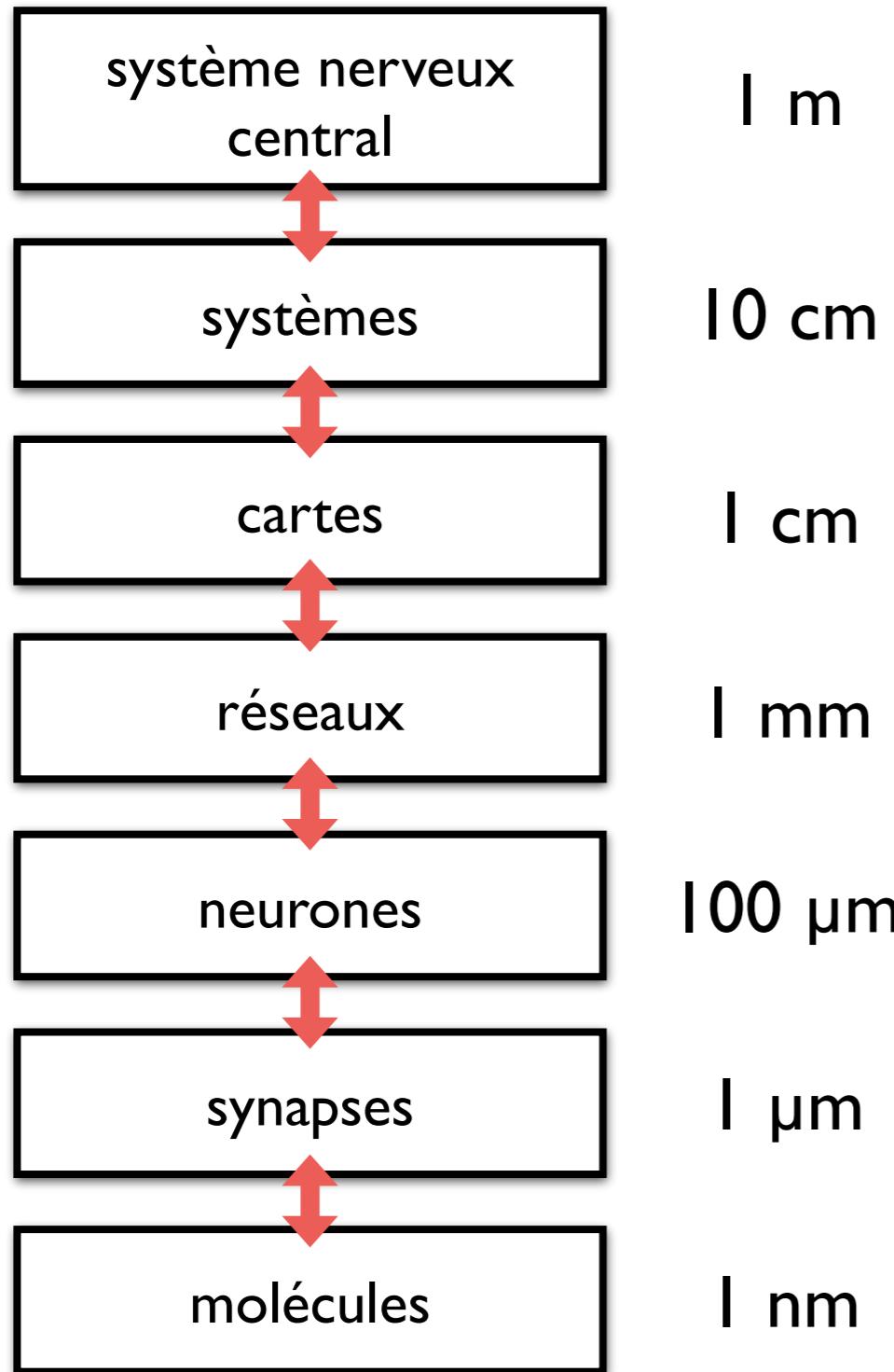
1. L'approche “bottom up”

Tout simplement tout reconstruire !

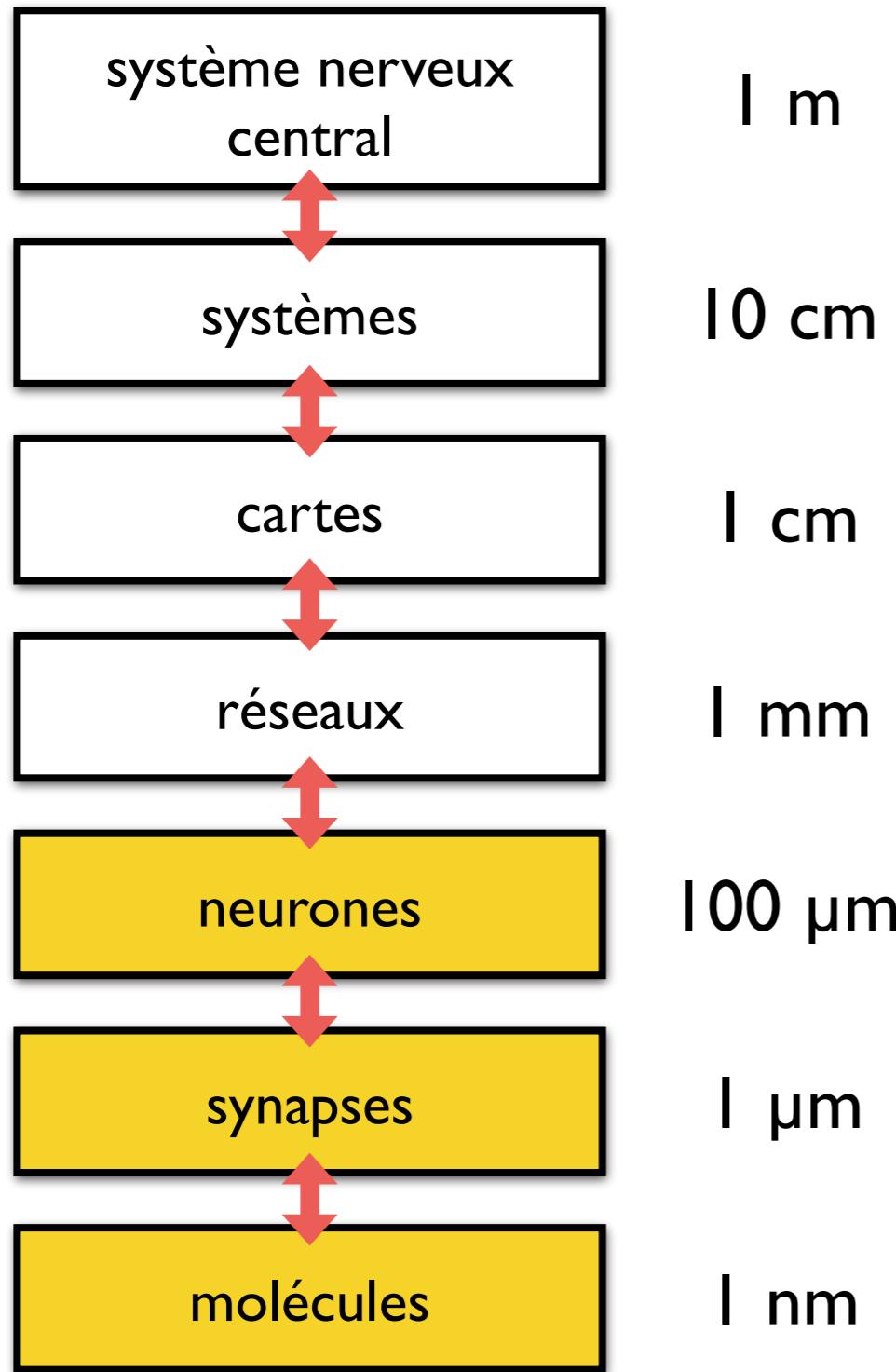
- Ingénierie inverse, ou “reverse engineering,” du cerveau : Désassembler et examiner/analyser *les ingrédients au niveau microscopique* pour en découvrir les principes de bases.
- Utiliser ces éléments afin de reproduire quelque chose qui ressemble aux *observations à une échelle supérieure*.
- Approche s'appuyant sur la physique et l'ingénierie.



A la recherche du mécanisme : Construire un système à partir des pièces

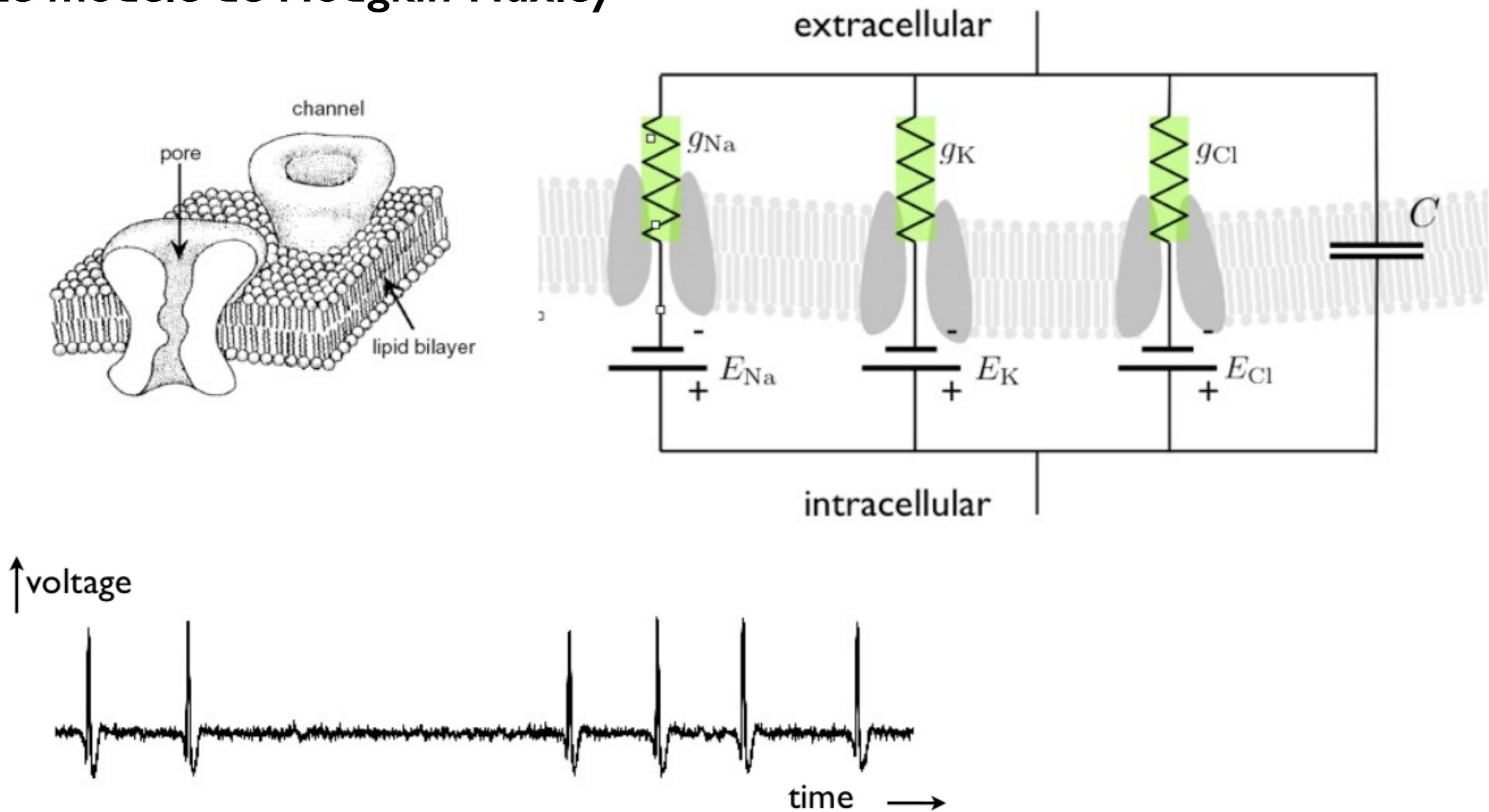


A la recherche du mécanisme : Construire un système à partir des pièces



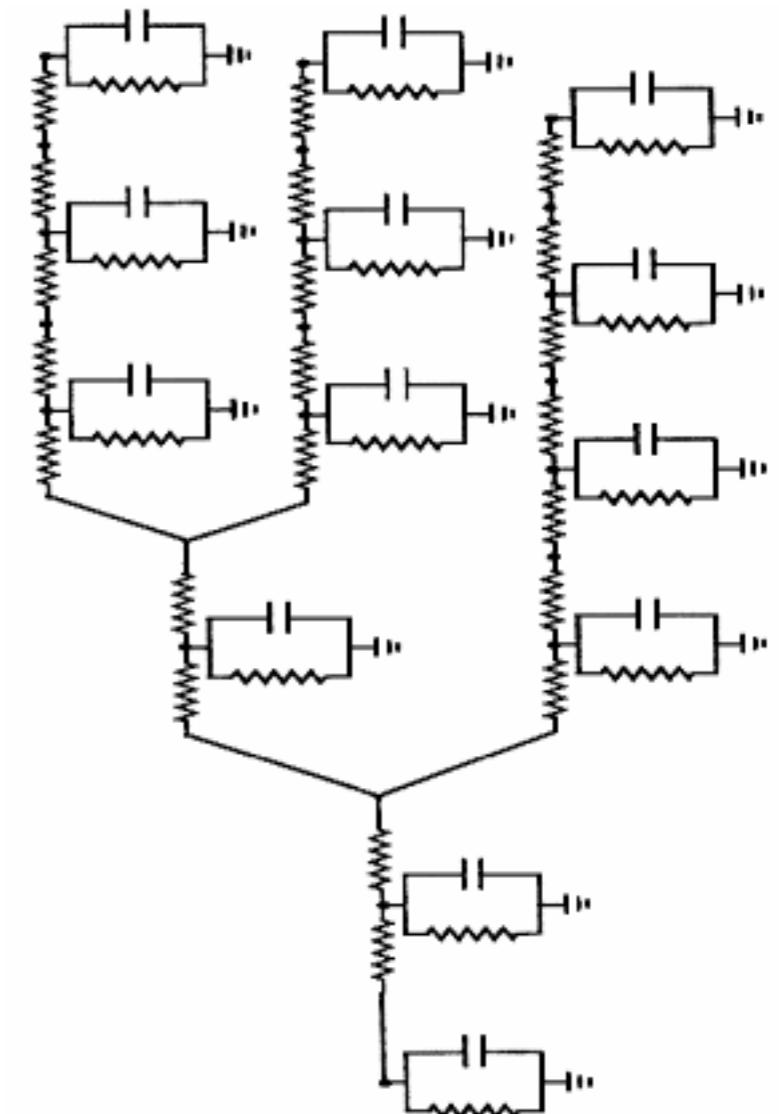
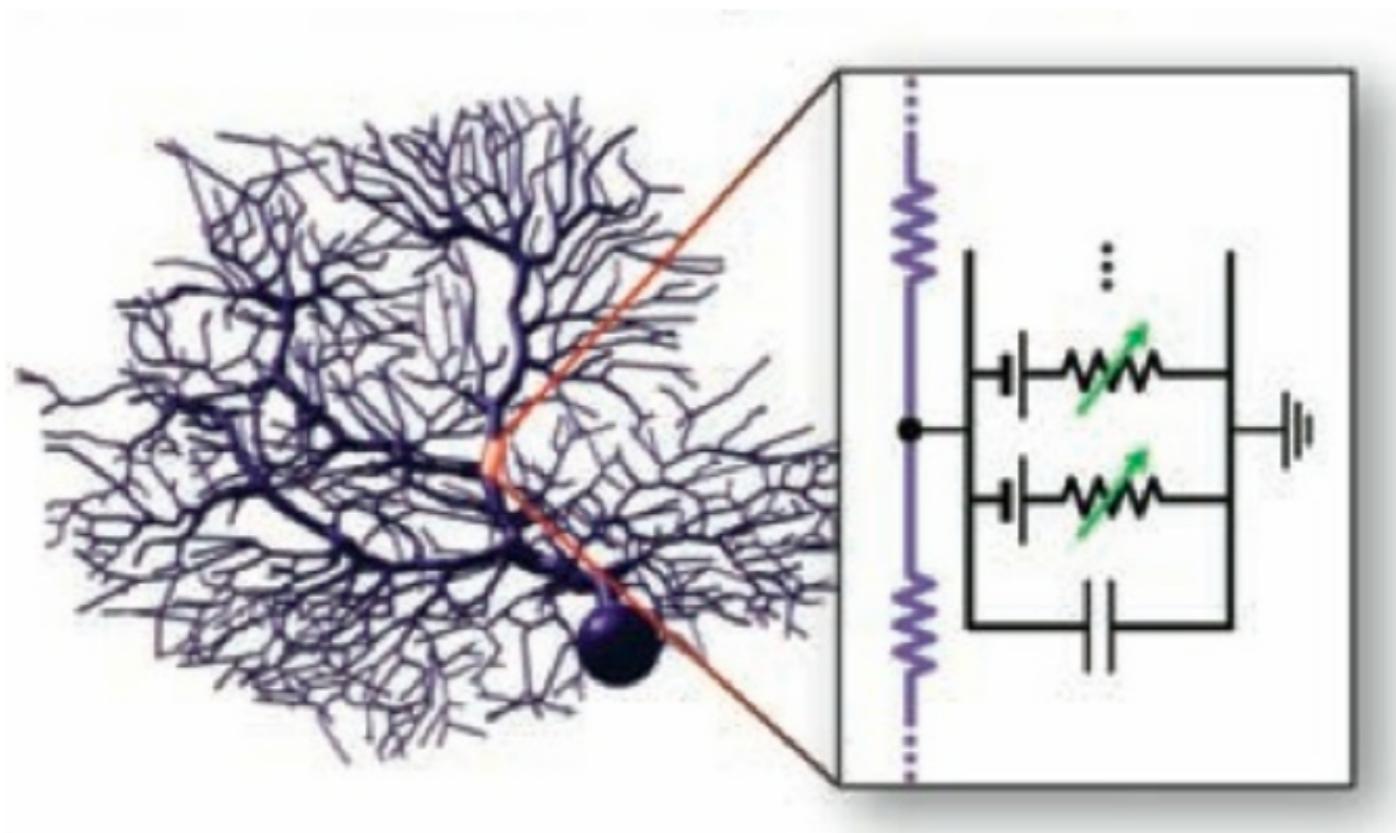
Biophysique du potentiel de la membrane

Le modèle de Hodgkin-Huxley



Reconstruction de neurones

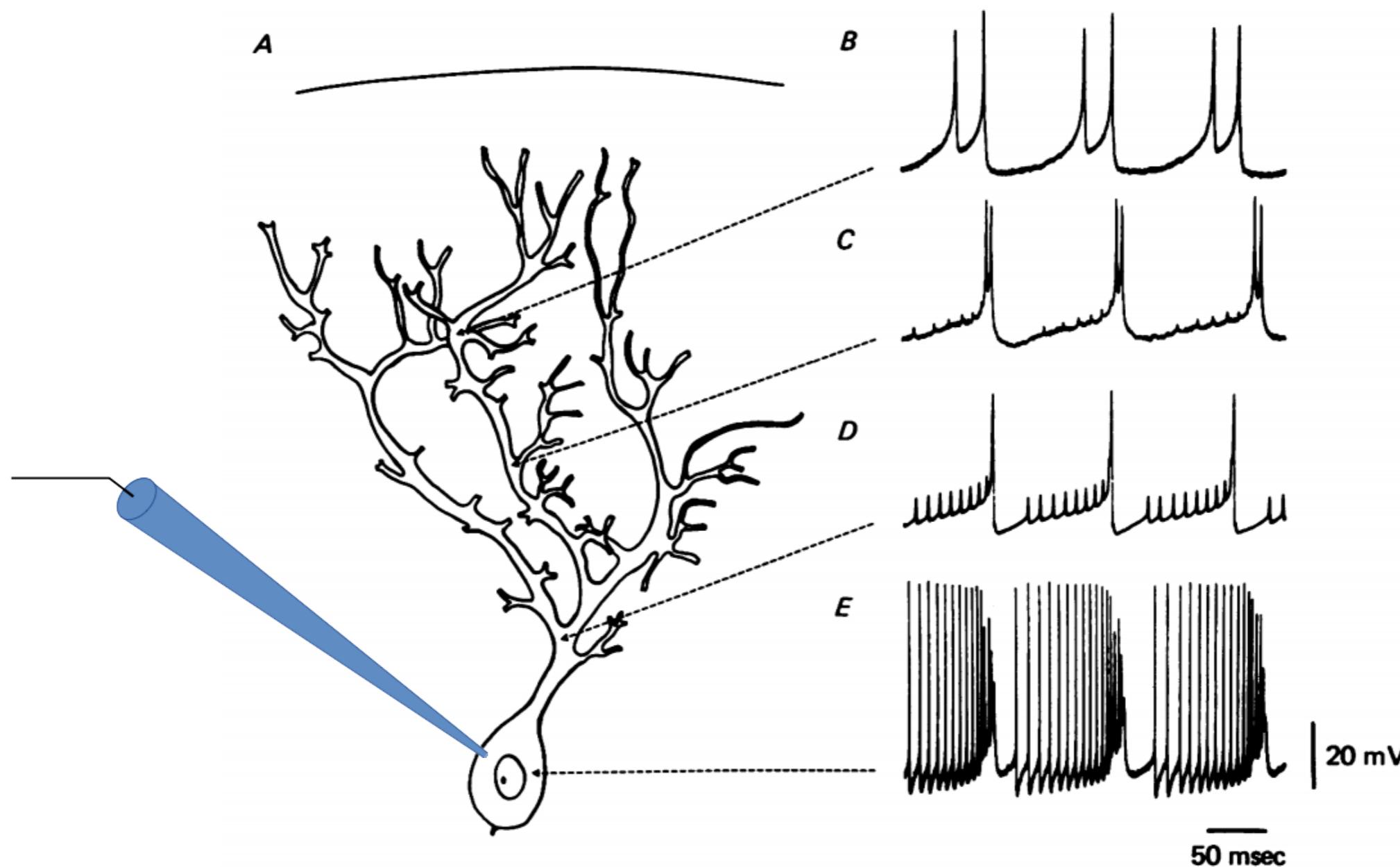
La théorie des câbles et la modélisation compartimentée



Modèles compartimentées détaillés
de neurones individuels.

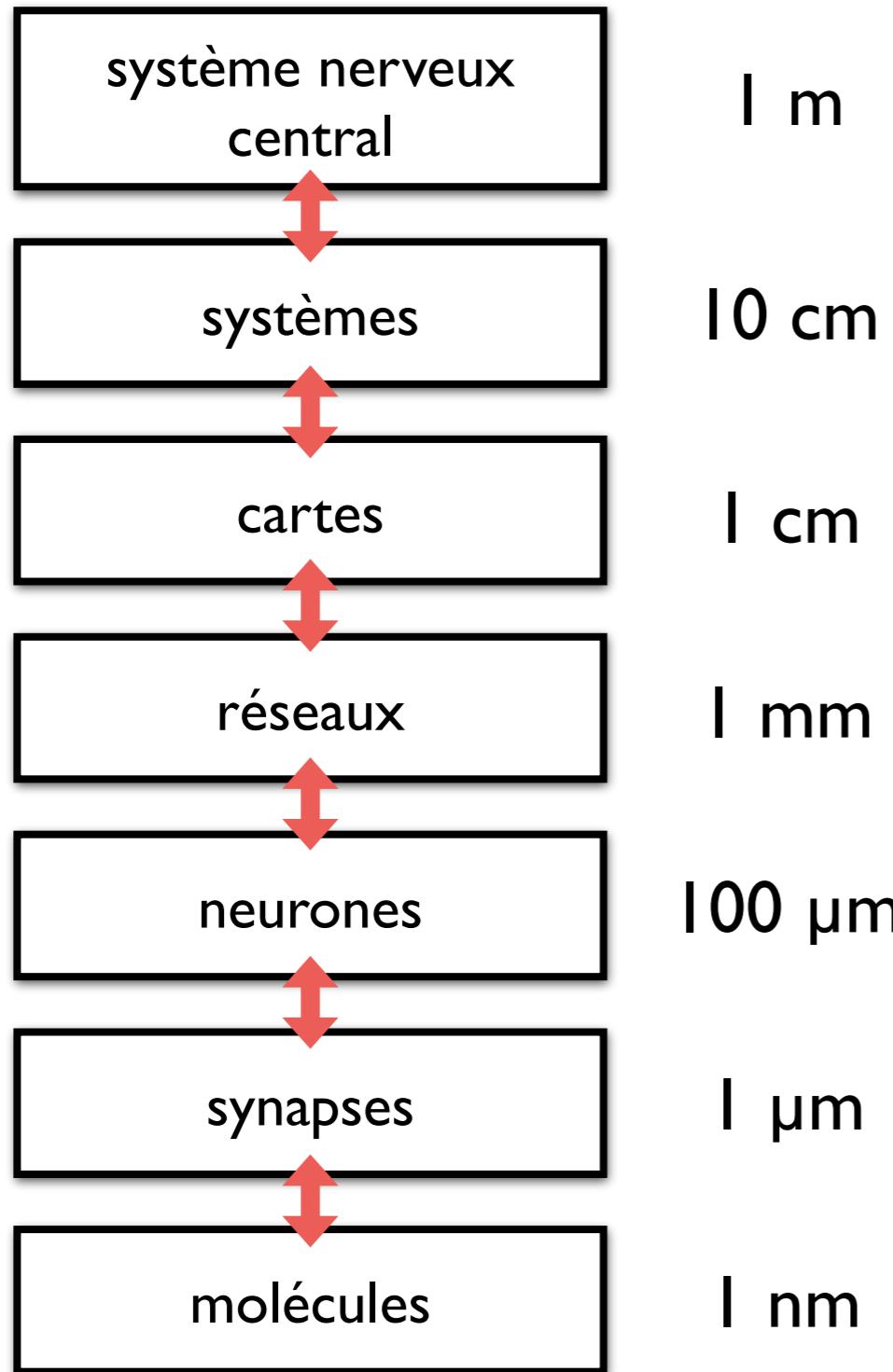
Reconstruction de neurones

Simulation de la propagation spatiale du potentiel de membrane

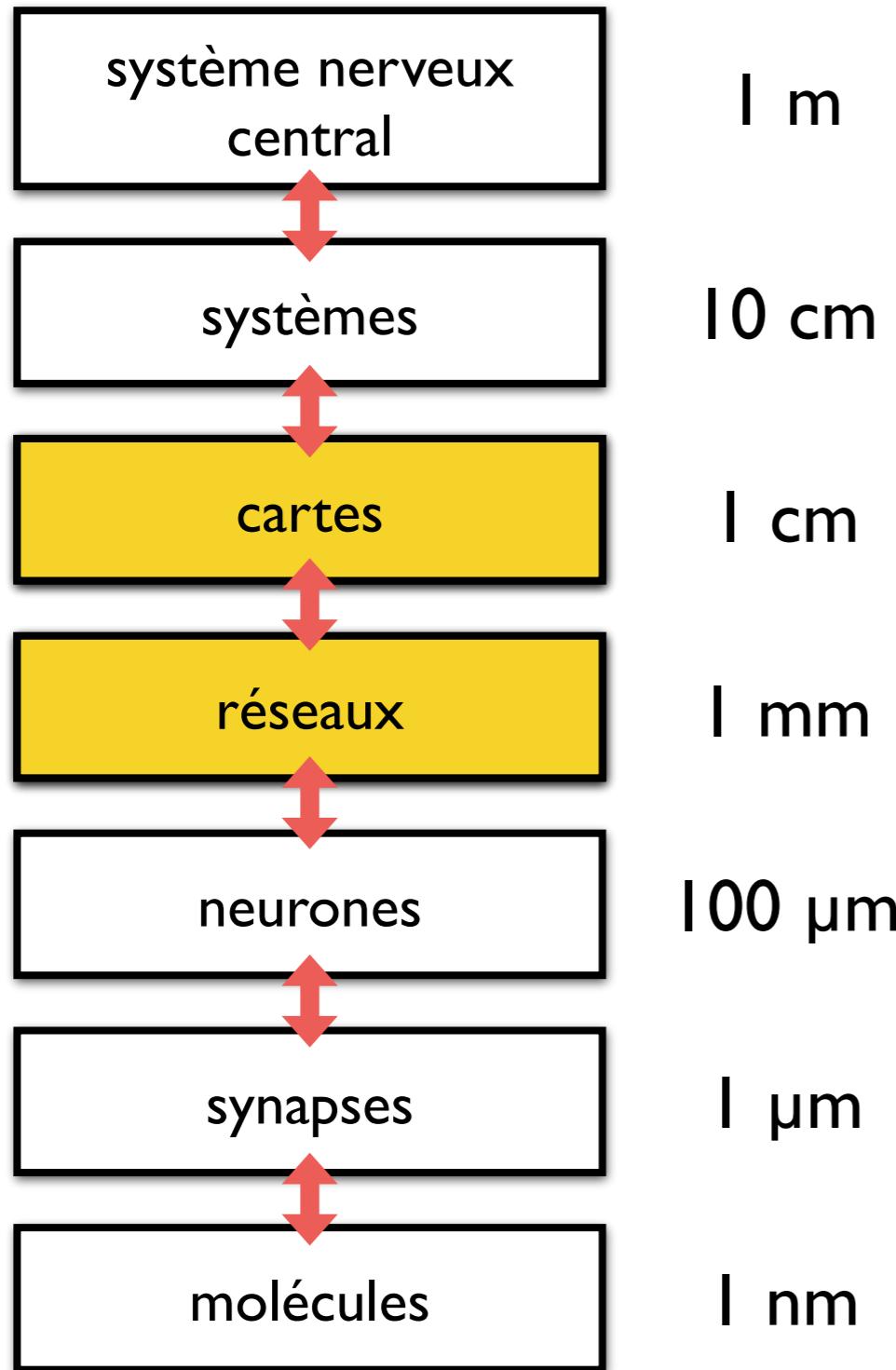


Llinas & Sugimori, 1980

A la recherche du mécanisme : Construire un système à partir des pièces

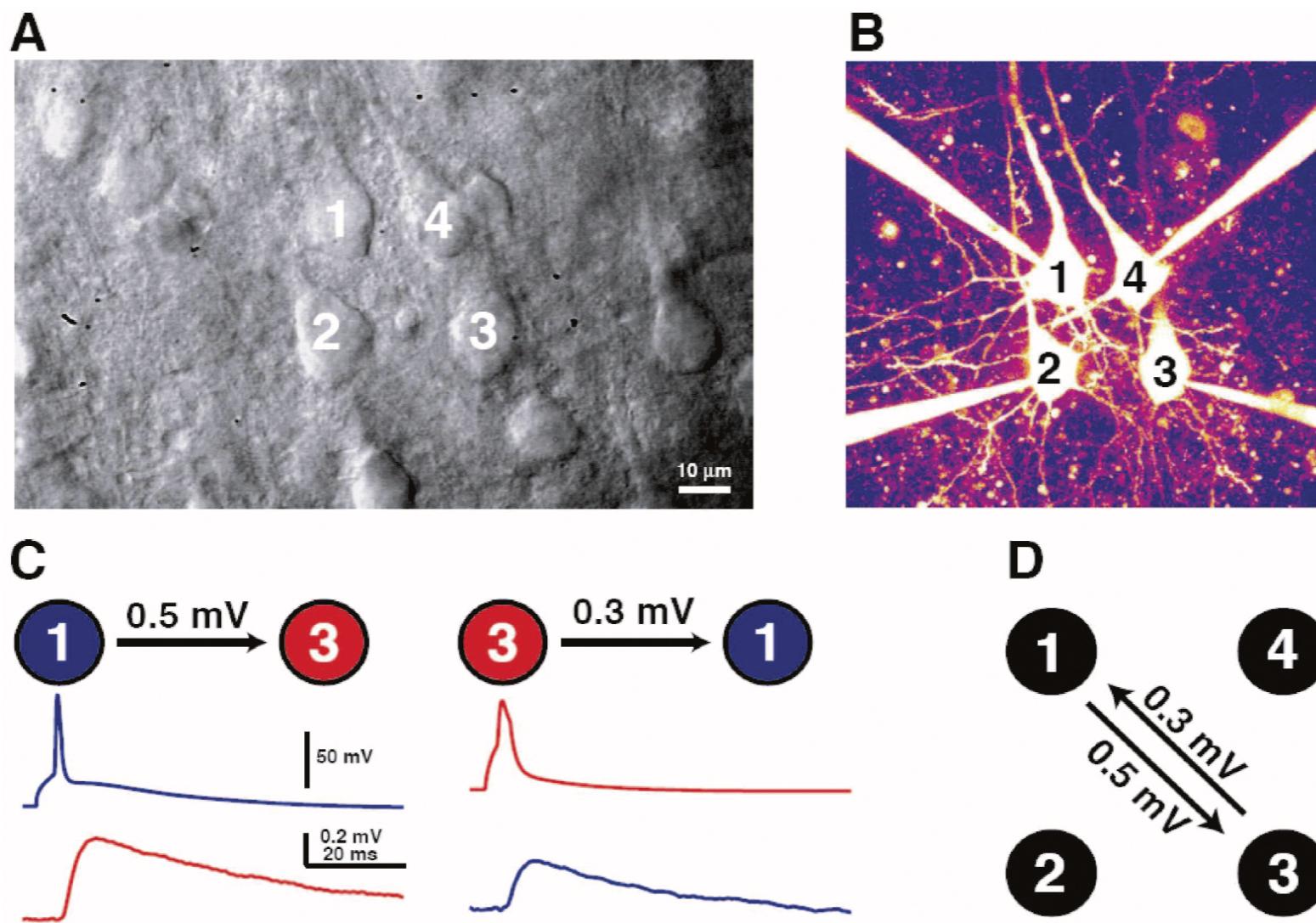


A la recherche du mécanisme : Construire un système à partir des pièces



Reconstruction des réseaux

- Manipulation et enregistrement électrique de deux neurones (ou plus) en même temps:

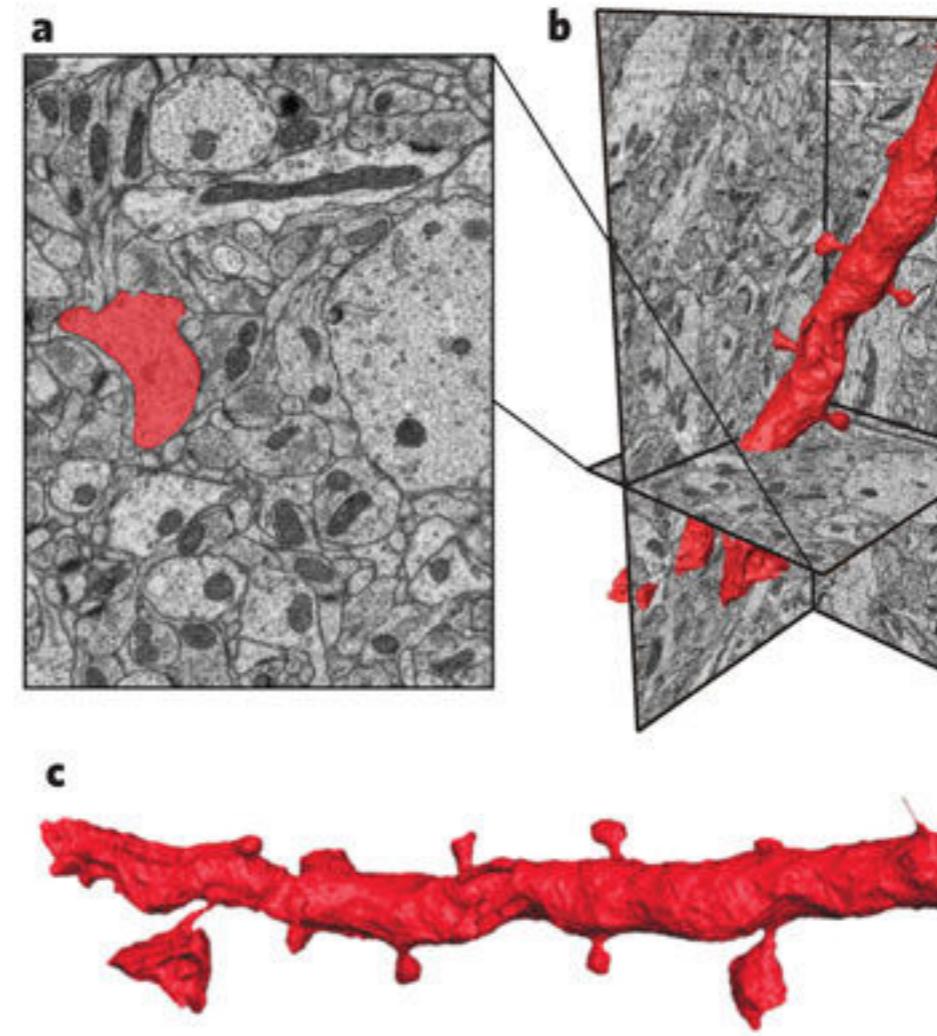
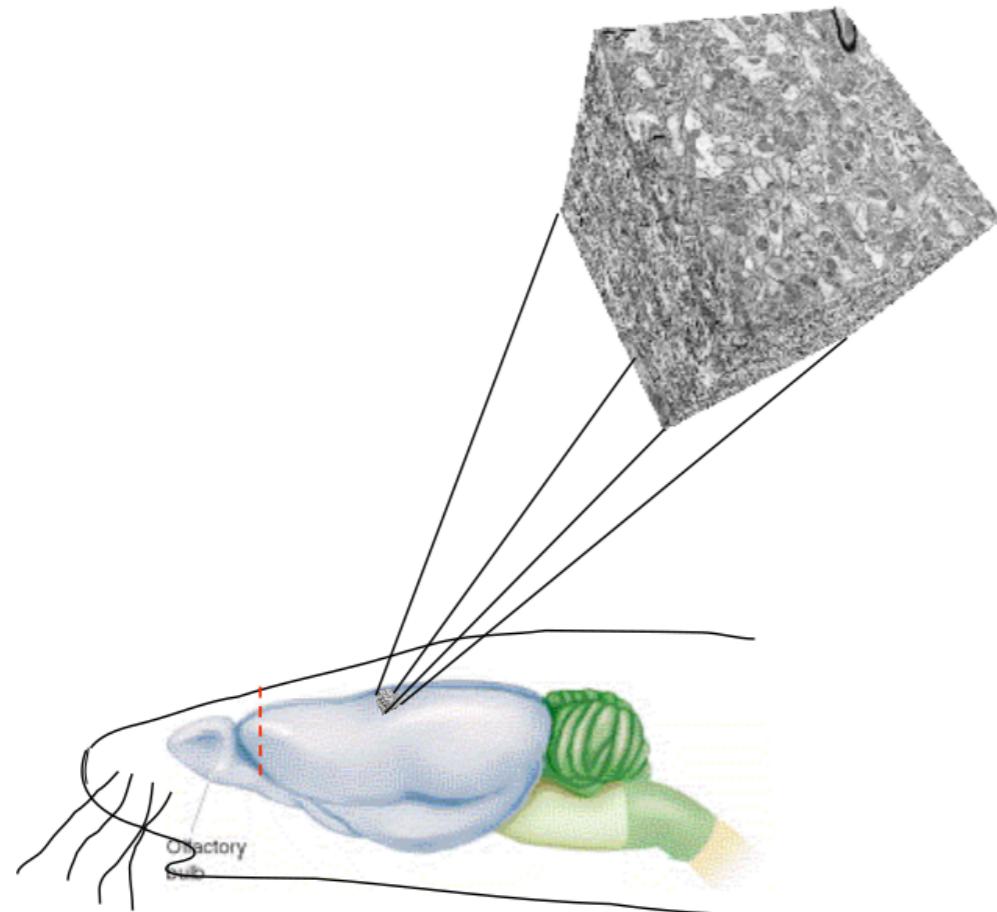


Song et al., 2005

Résultats : probabilités de connexion et propriétés électriques

Reconstruction des réseaux

- Serial Block-Face Scanning Electron Microscopy (SEM):
scanner des tranches du cerveau et reconstruire le circuit...



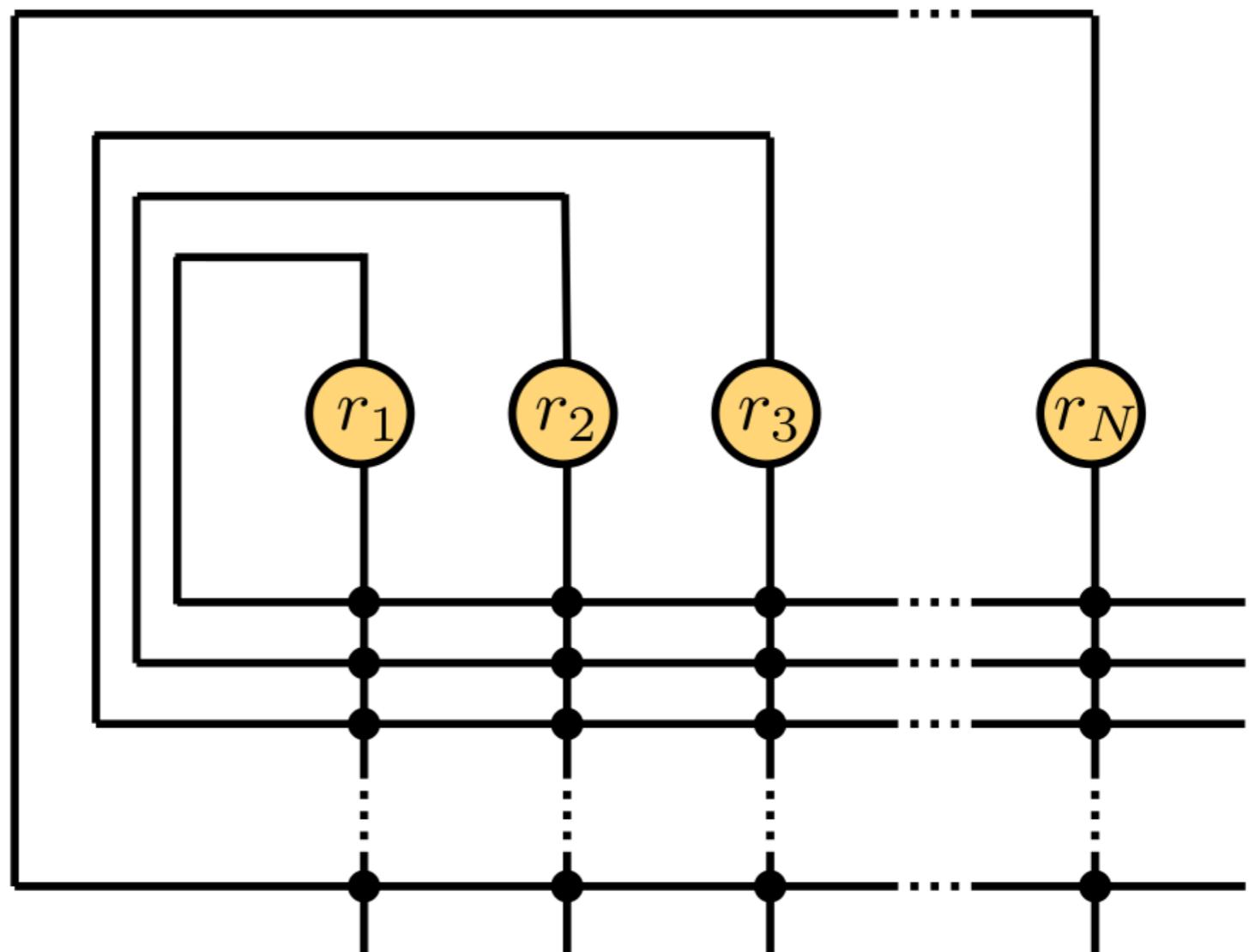
courtesy of Winfried Denk

Résultats : le “connectome” – une carte complète des connexions neurales dans le cerveau

Théorie des réseaux neuronaux

Neurones, synapses, connexions → activité du réseau

$$\dot{r}_i = -r_i + f\left(\sum_{j=1}^N w_{ij}r_j + I_i\right)$$

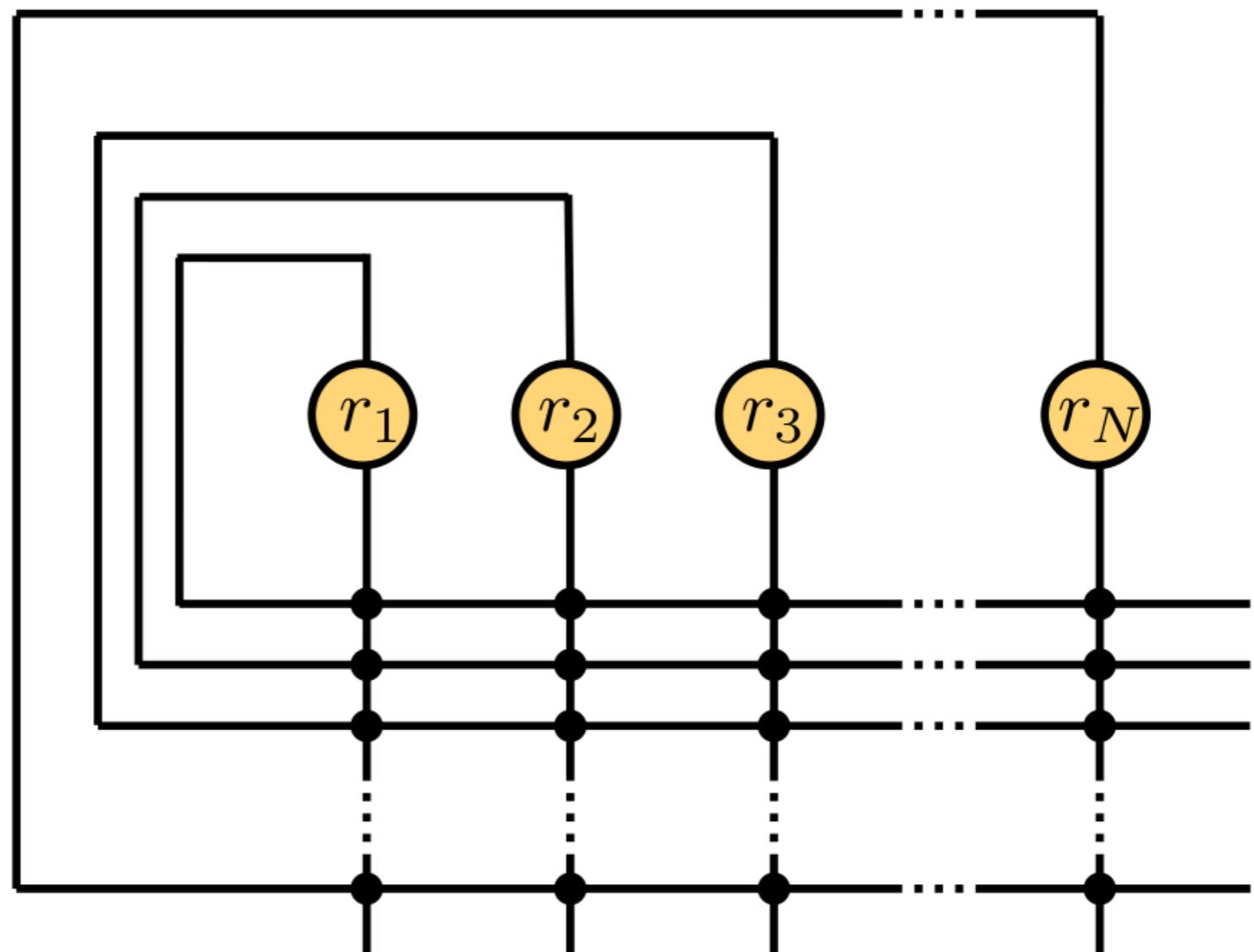


Théorie des réseaux neuronaux

Neurones, synapses, connexions → activité du réseau

taux de décharge

$$\dot{r}_i = -r_i + f\left(\sum_{j=1}^N w_{ij} r_j + I_i\right)$$



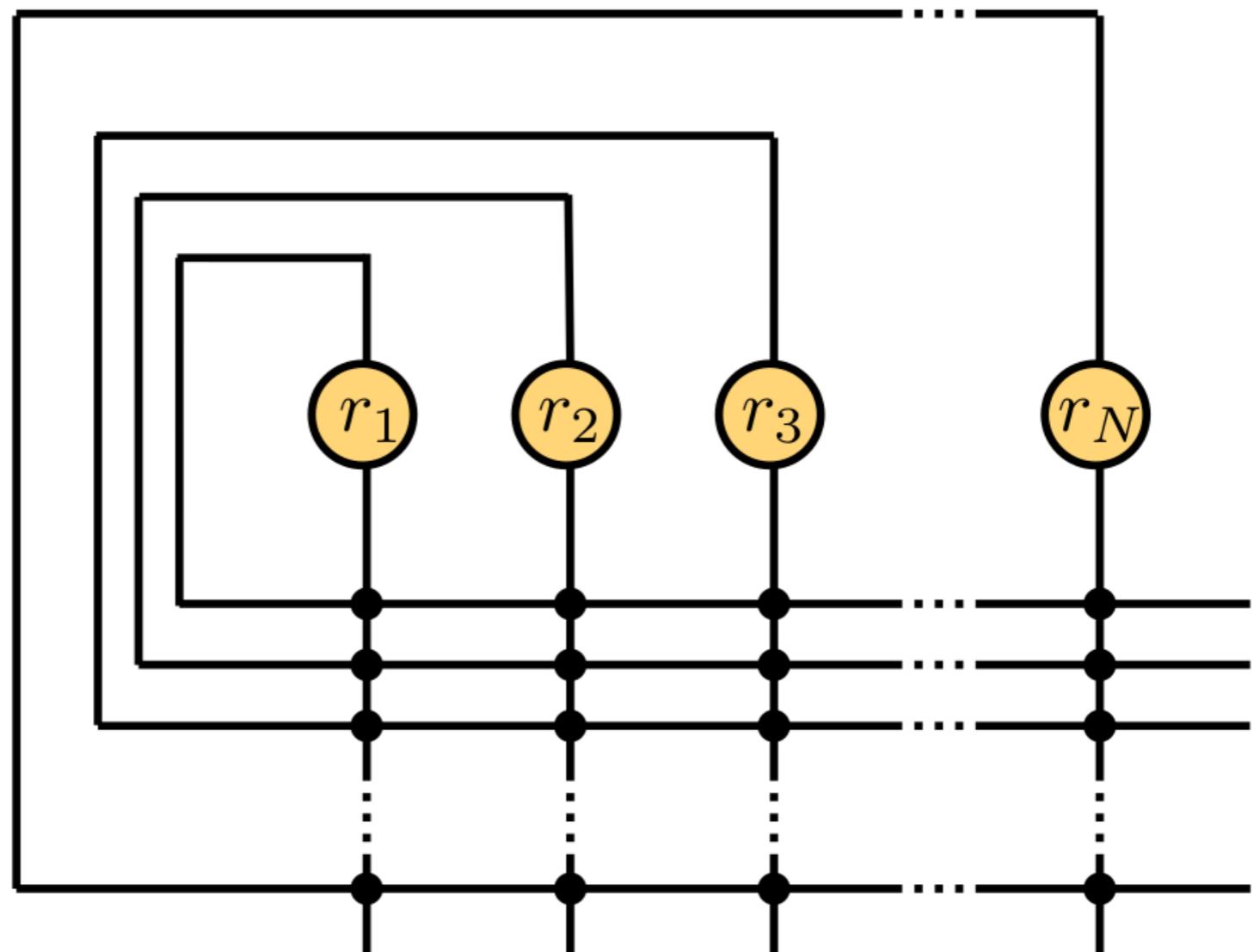
Théorie des réseaux neuronaux

Neurones, synapses, connexions → activité du réseau

taux de décharge

$$\dot{r}_i = -r_i + f\left(\sum_{j=1}^N w_{ij} r_j + I_i\right)$$

w_{ij} ... matrice de connexion



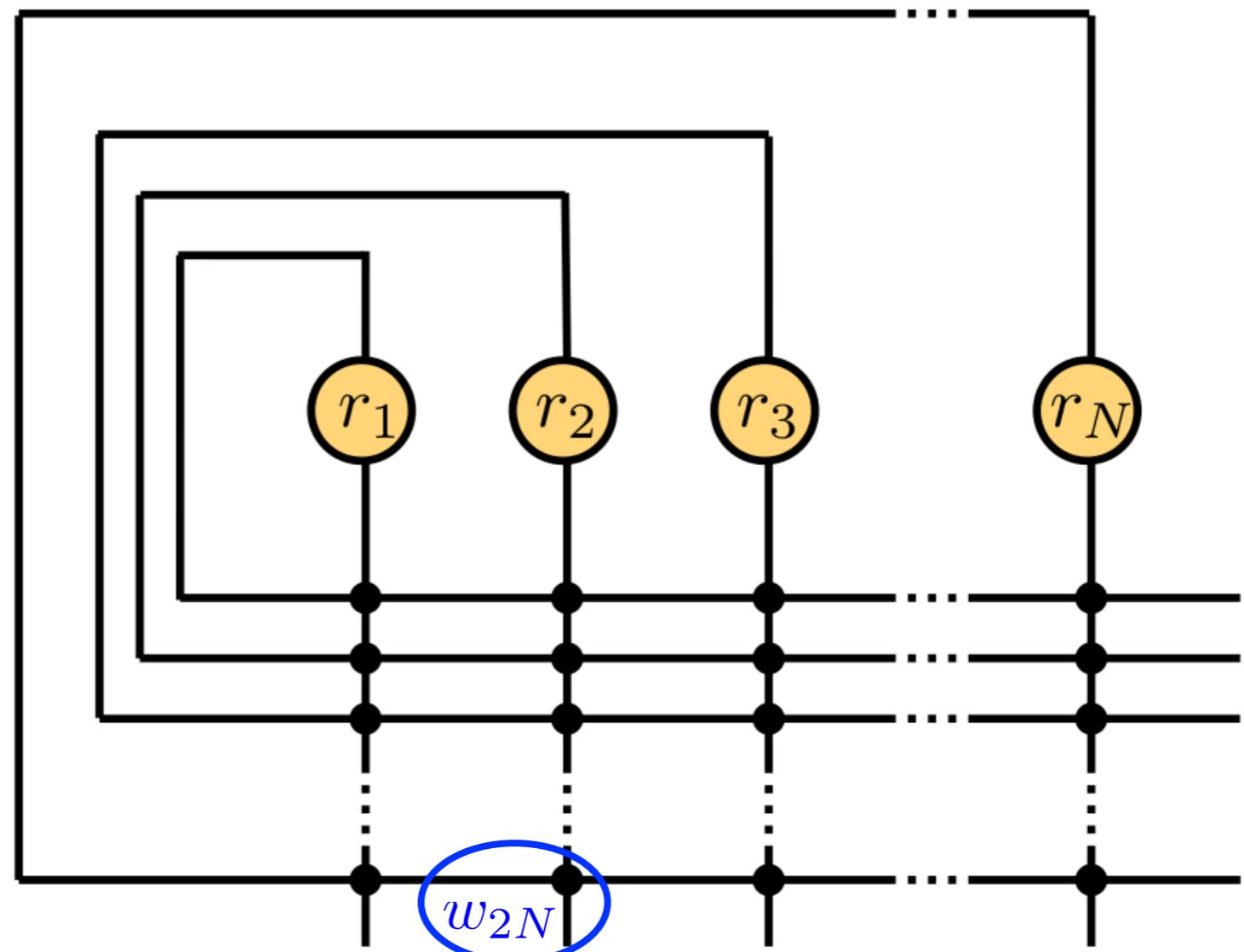
Théorie des réseaux neuronaux

Neurones, synapses, connexions → activité du réseau

taux de décharge

$$\dot{r}_i = -r_i + f\left(\sum_{j=1}^N w_{ij} r_j + I_i\right)$$

w_{ij} ... matrice de connexion



Dynamique du réseau largement déterminée par la connectivité

Régimes dynamiques :

- points fixes stables / instables
- cycles limites
- attracteurs chaotiques

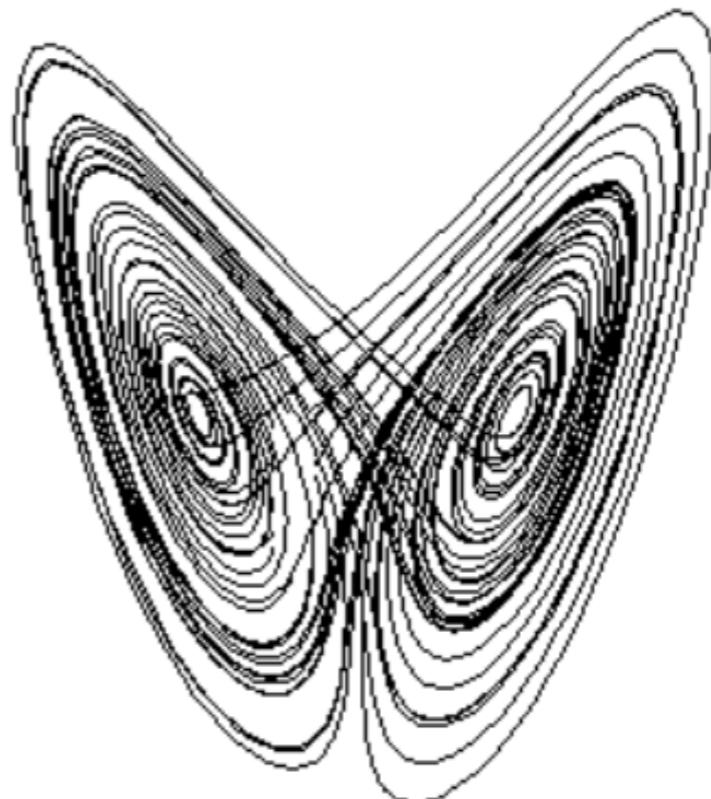
Rémarque :

- Différents attracteurs peuvent coexister dans différentes parties de l'espace de phase.

Dans la limite $N \rightarrow \infty$:

- Un réseaux neuronal peut tout calculer.

$$\dot{r}_i = -r_i + f\left(\sum_{j=1}^N w_{ij}r_j + I_i\right)$$

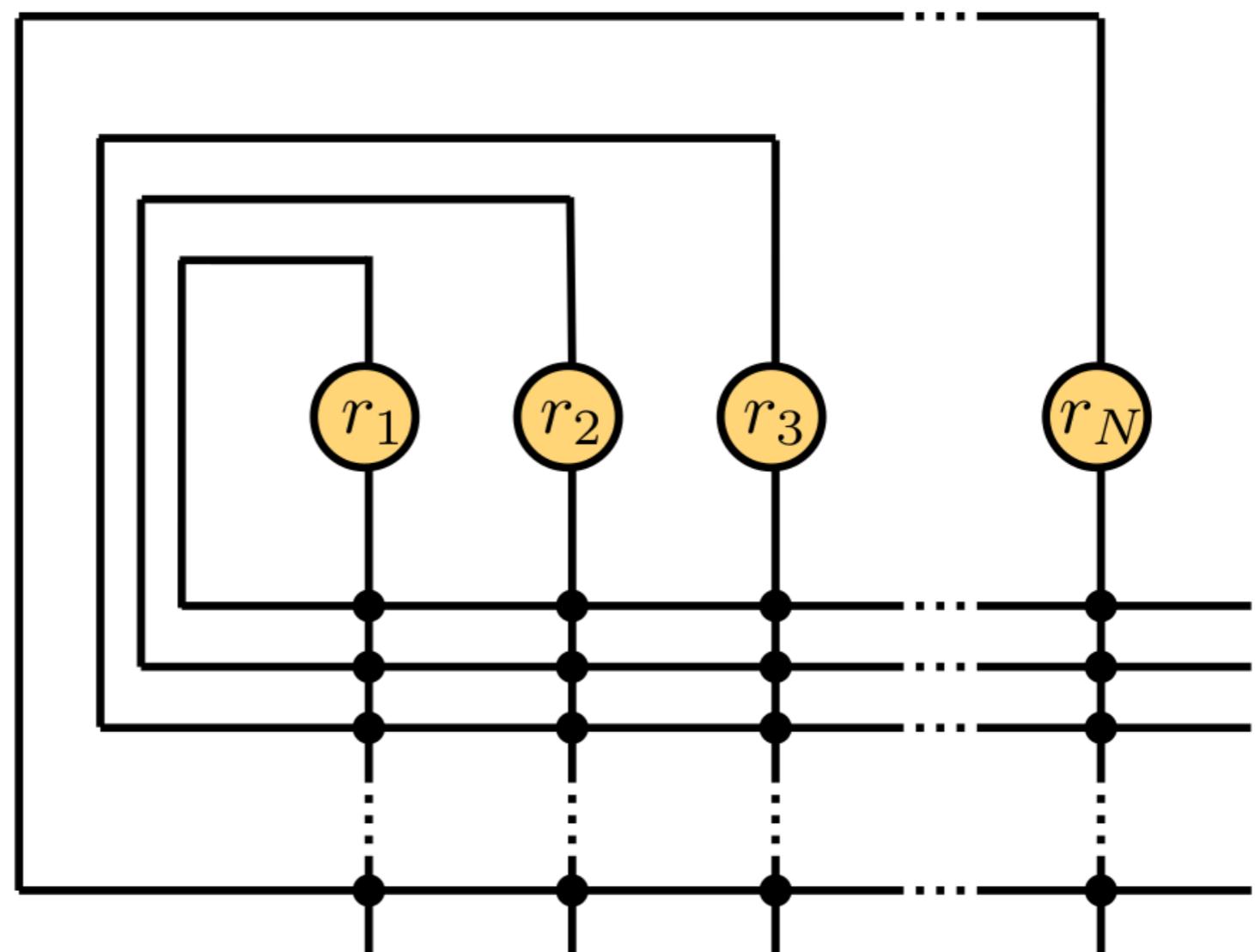


Théorie (statistique) des réseaux neuronaux

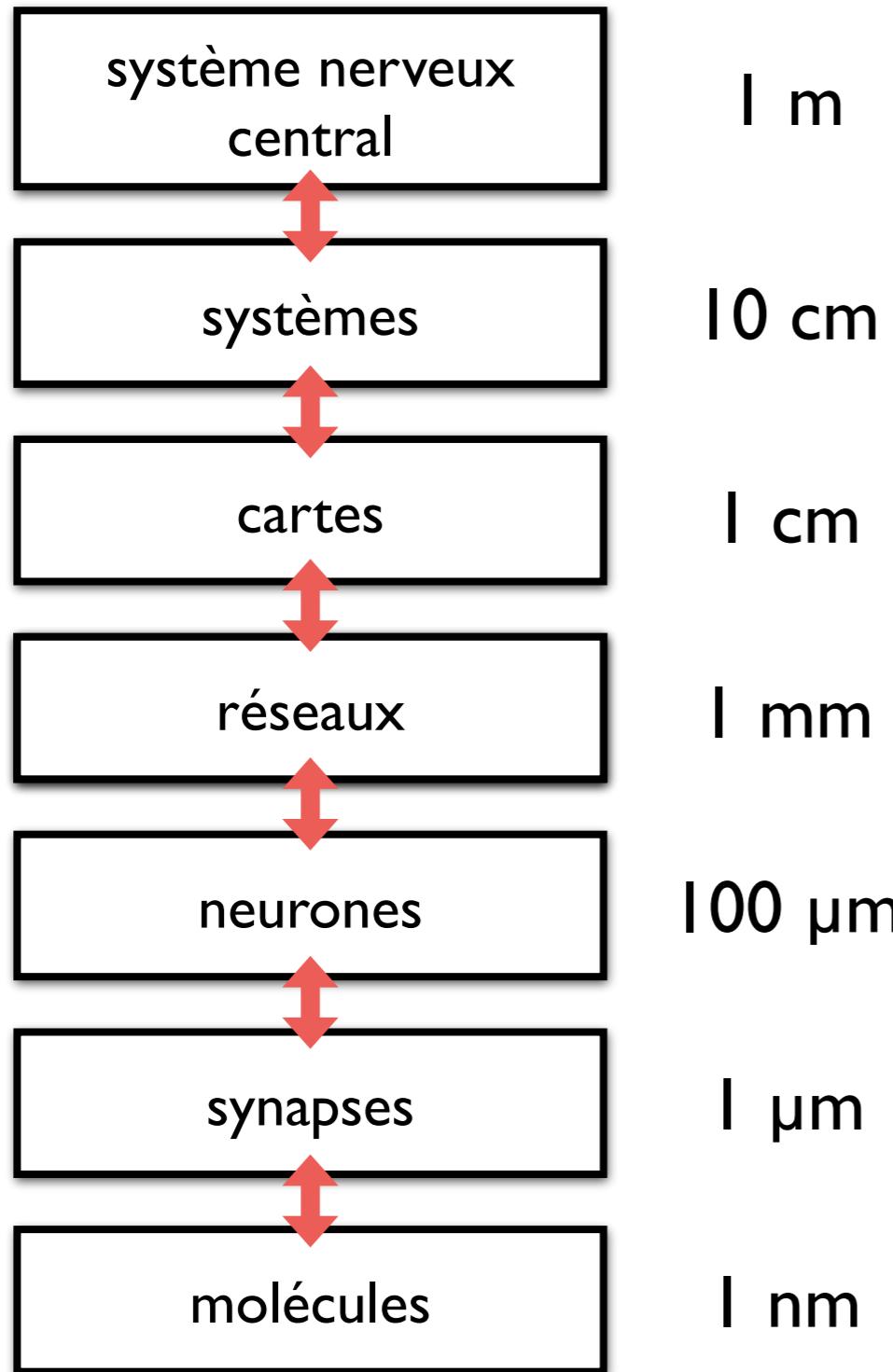
Neurones, synapses, connexions → activité du réseau

Dans *quelles conditions*
peux-t-on observer

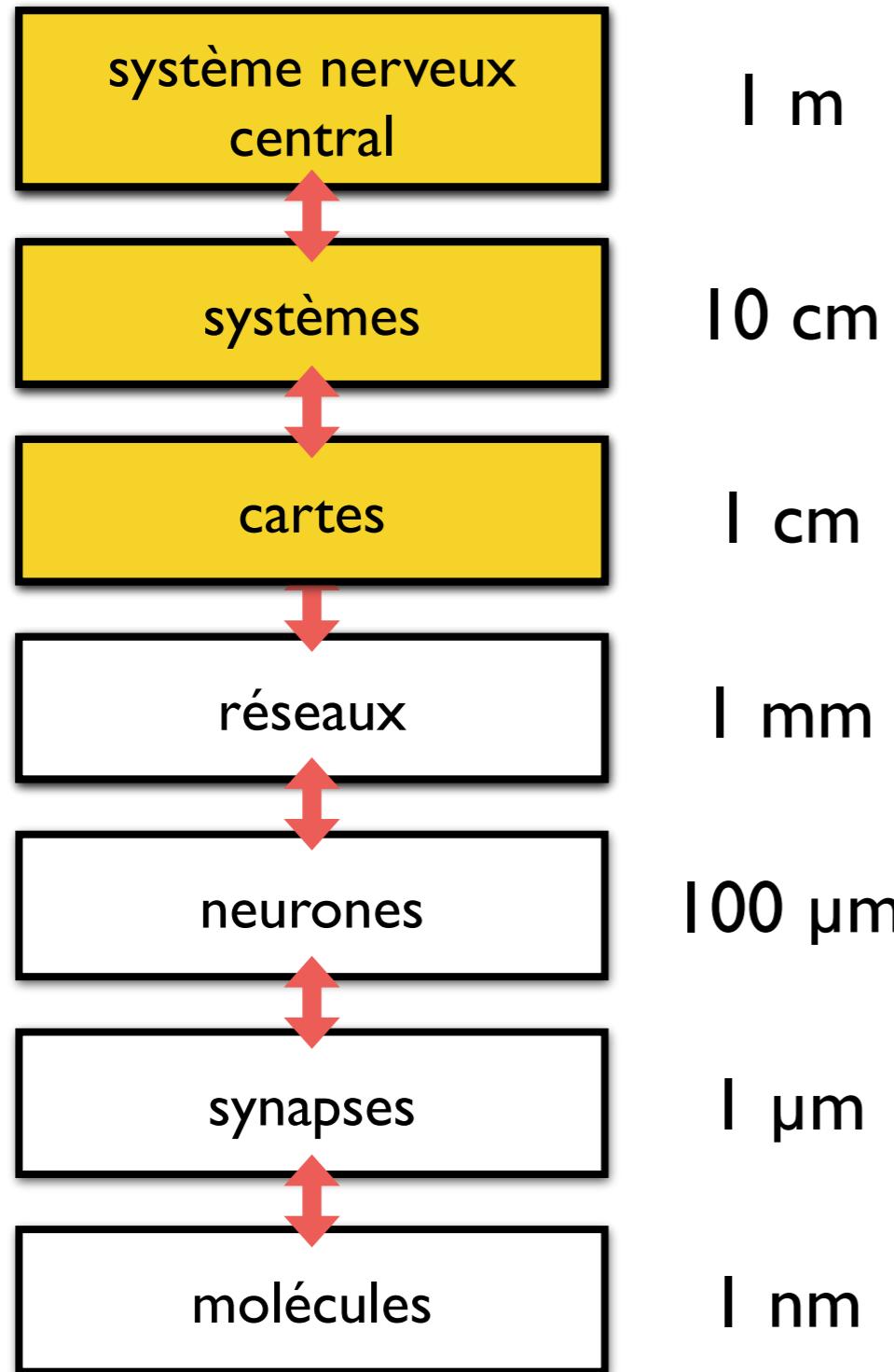
- des points fixes ?
- une activité synchrone ?
- ... asynchrone ?
- des trains de spike irréguliers (Poisson) ?
- des oscillations ?
- des motifs spatiaux ?
- ...



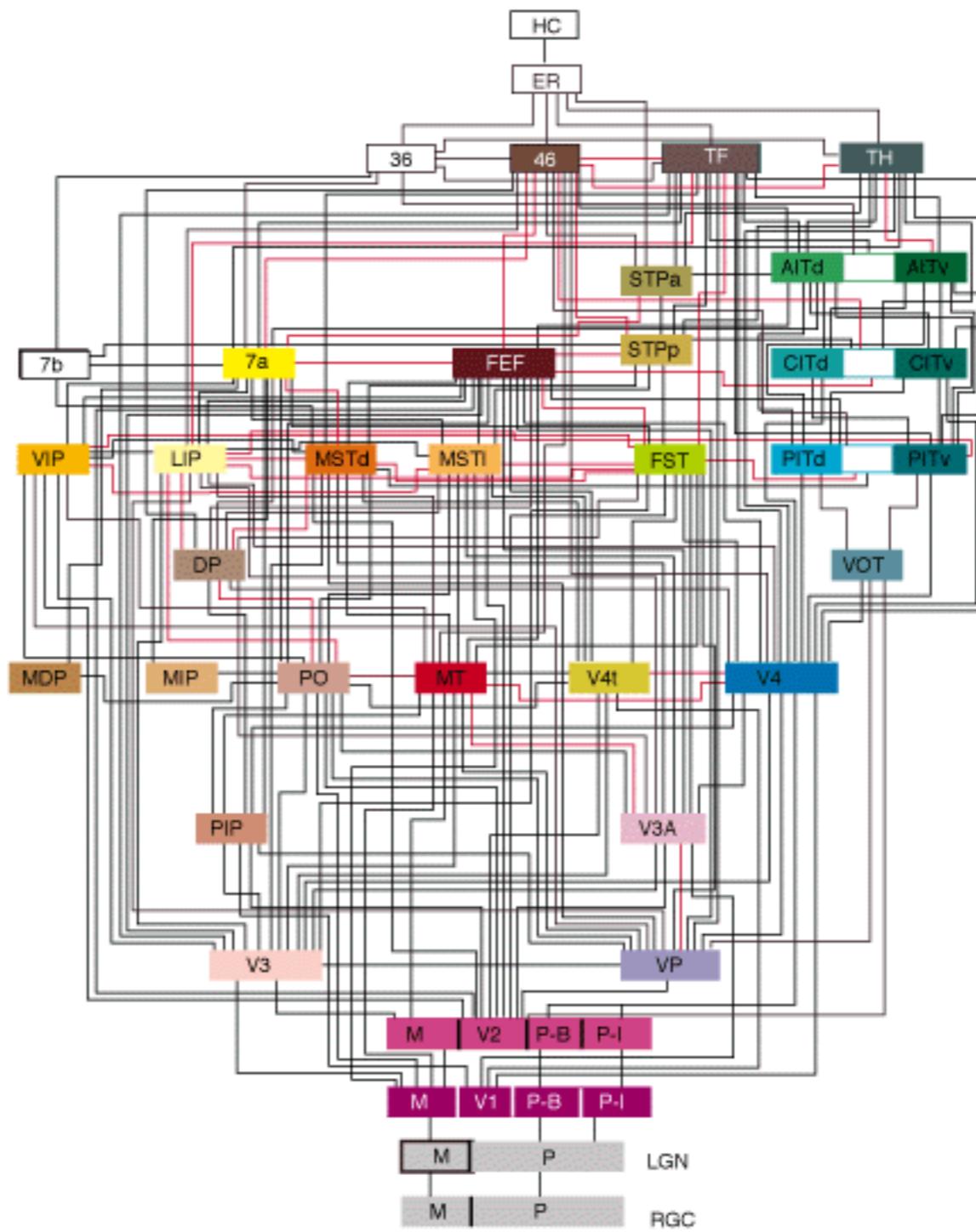
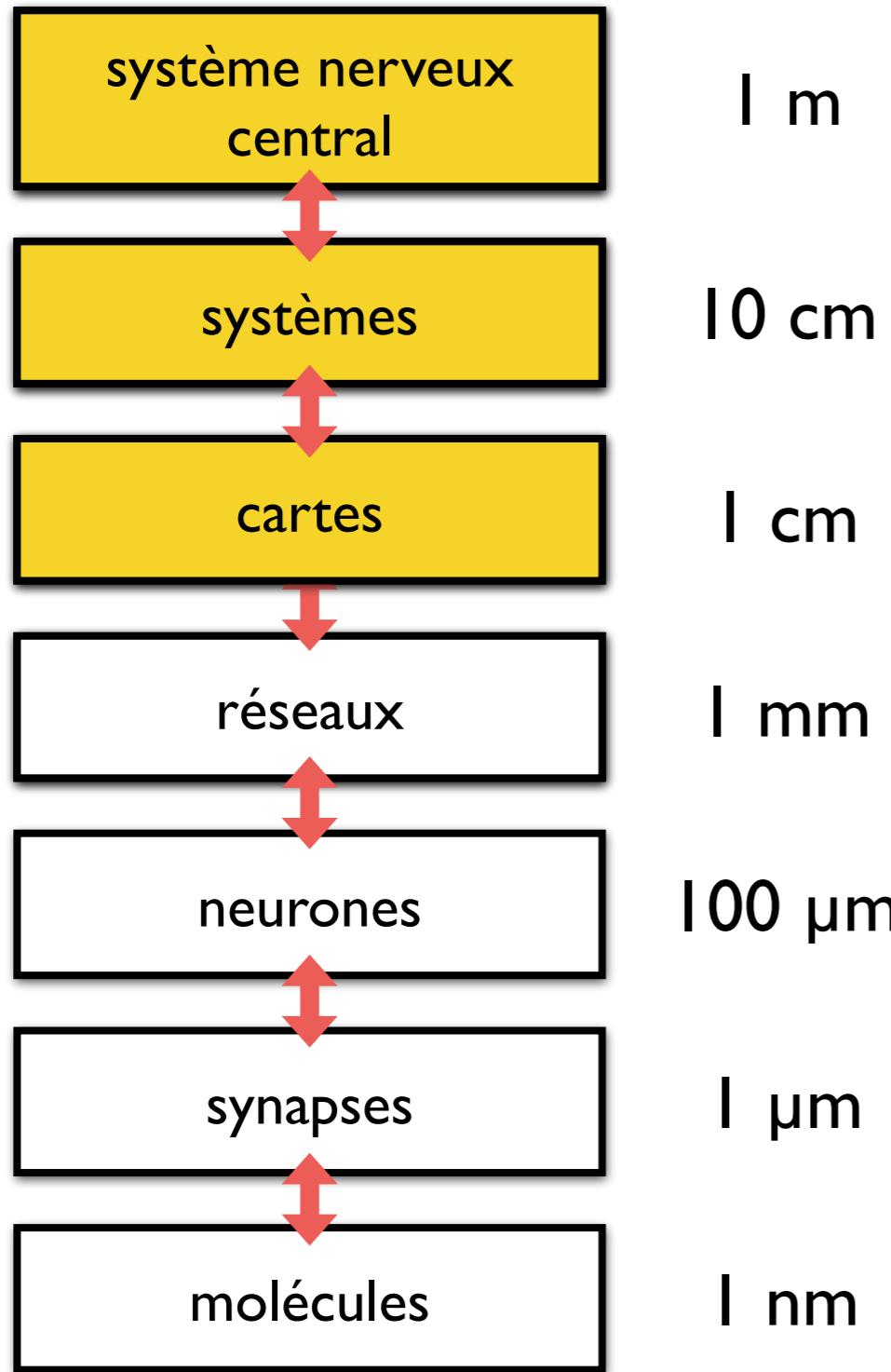
A la recherche du mécanisme : Construire un système à partir des pièces



A la recherche du mécanisme : Construire un système à partir des pièces

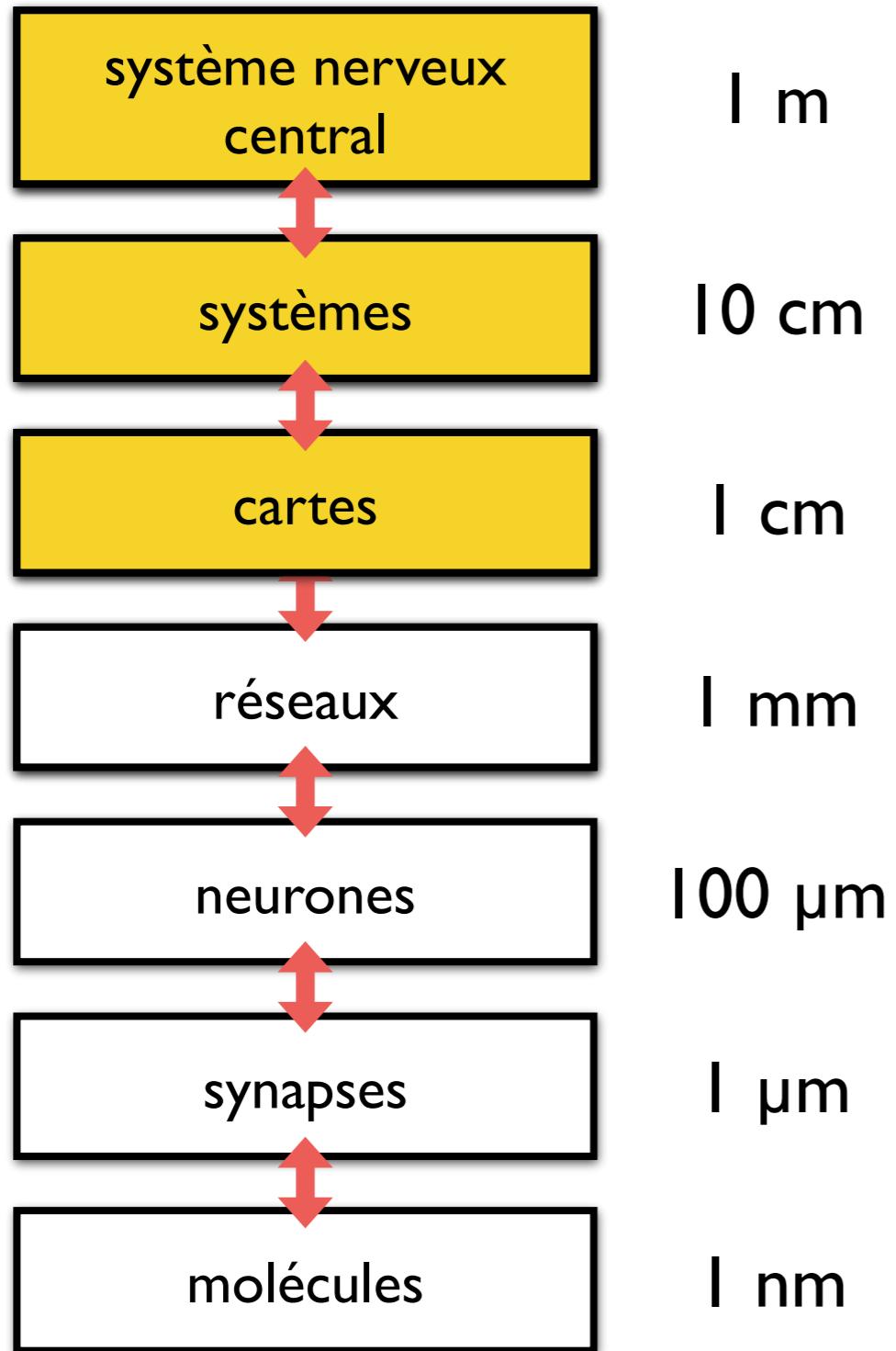


A la recherche du mécanisme : Construire un système à partir des pièces



Felleman & van Essen, 1991

A la recherche du mécanisme : Construire un système à partir des pièces



?

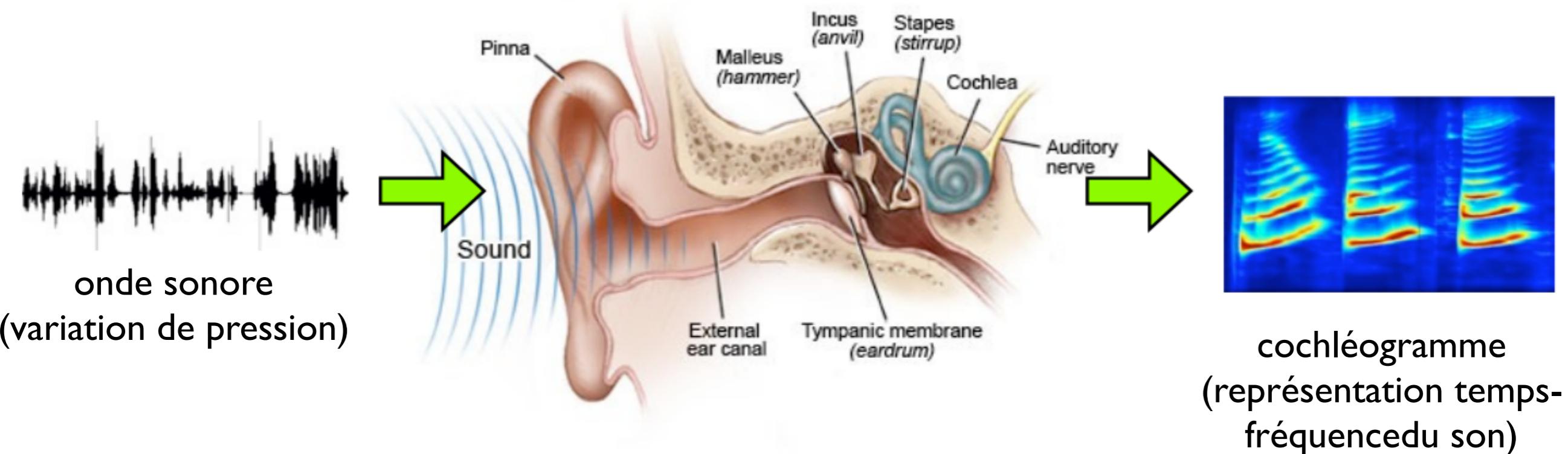
Human Brain Project

2. Une approche informatique

Étudier des problèmes de calcul

Calcul : manipuler des informations

L'ouïe



Représentation de l'information

Exemple musicale : Art Blakey - Mayreh

Représentation de l'information

Exemple musicale : Art Blakey - Mayreh

Représentation de l'information

Exemple musicale : Art Blakey - Mayreh

Comment transmettre un concert de jazz ?

Représentation de l'information

Exemple musicale : Art Blakey - Mayreh

Comment transmettre un concert de jazz ?

- Partition :



Représentation de l'information

Exemple musicale : Art Blakey - Mayreh

Comment transmettre un concert de jazz ?

- Partition :



Représentation de l'information

Exemple musicale : Art Blakey - Mayreh

Comment transmettre un concert de jazz ?

- Partition :



- Son :

Représentation de l'information

Exemple musicale : Art Blakey - Mayreh

Comment transmettre un concert de jazz ?

- Partition :



- Son :



- Enregistrement :

Représentation de l'information

Exemple musicale : Art Blakey - Mayreh

Comment transmettre un concert de jazz ?

- Partition :



- Son :



- Enregistrement :



- Langage :

“L'autre jour, je suis allé à ce concert avec ce super batteur, et puis ...”

Représentation de l'information

Exemple musicale : Art Blakey - Mayreh

Comment transmettre un concert de jazz ?

- Partition :



- Son :



- Enregistrement :



- Langage :

“L'autre jour, je suis allé à ce concert avec ce super batteur, et puis ...”

PERTE !

Représentation de l'information

Exemple musicale : Art Blakey - Mayreh

Comment transmettre un concert de jazz ?

- Partition :



- Son :



- Enregistrement :



- Langage :

“L'autre jour, je suis allé à ce concert avec ce super batteur, et puis ...”

PERTE !

On peut représenter (“coder”) l'information avec plus ou moins de perte.

Pourquoi représenter l'information autrement ?

Exemple : le nombre 23

XXIII système romain

23 système décimal

10111 système binaire

Pourquoi représenter l'information autrement ?

Exemple : le nombre 23

XXIII	système romain	?
23	système décimal	en multiples de 10
10111	système binaire	en multiples de 2

Pourquoi représenter l'information autrement ?

Exemple : le nombre 23

XXIII	système romain	?
23	système décimal	en multiples de 10
10111	système binaire	en multiples de 2

Pouvez-vous additionner les nombres suivants ?

$$\begin{array}{r} 29 \\ + 33 \\ \hline \end{array}$$

$$\begin{array}{r} 11101 \\ + 100001 \\ \hline \end{array}$$

$$\begin{array}{r} XXIX \\ + XXXIII \\ \hline \end{array}$$

Pourquoi représenter l'information autrement ?

Exemple : le nombre 23

XXIII	système romain	?
23	système décimal	en multiples de 10
10111	système binaire	en multiples de 2

Pouvez-vous additionner les nombres suivants ?

$$\begin{array}{r} 29 \\ + 33 \\ \hline 62 \end{array}$$

$$\begin{array}{r} 11101 \\ + 100001 \\ \hline 111110 \end{array}$$

$$\begin{array}{r} \text{XXIX} \\ + \text{XXXIII} \\ \hline \text{LXII} \end{array}$$

Pourquoi représenter l'information autrement ?

Exemple : le nombre 23

XXIII

système romain

?

23

système décimal

en multiples de 10

10111

système binaire

en multiples de 2

Pouvez-vous additionner les nombres suivants ?

$$\begin{array}{r} 29 \\ + 33 \\ \hline 62 \end{array}$$

SIMPLE

$$\begin{array}{r} 11101 \\ + 100001 \\ \hline 111100 \end{array}$$

SIMPLE

$$\begin{array}{r} \cancel{XXIX} \\ + \cancel{XXXIII} \\ \hline LXII \end{array}$$

Difficile

Pourquoi représenter l'information autrement ?

Exemple : le nombre 23

XXIII	système romain	?
23	système décimal	en multiples de 10
10111	système binaire	en multiples de 2

Pouvez-vous additionner les nombres suivants ?

$$\begin{array}{r} 29 \\ + 33 \\ \hline 62 \end{array}$$

SIMPLE

$$\begin{array}{r} 11101 \\ + 100001 \\ \hline 111100 \end{array}$$

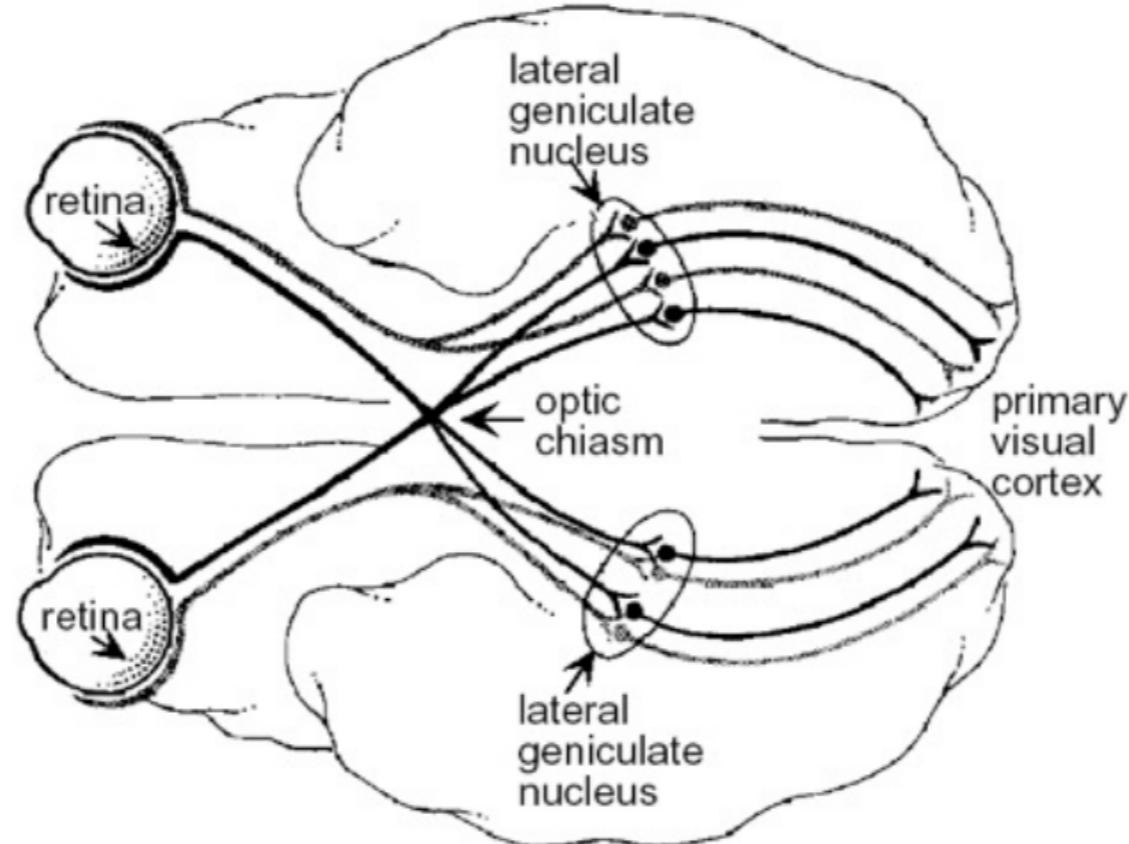
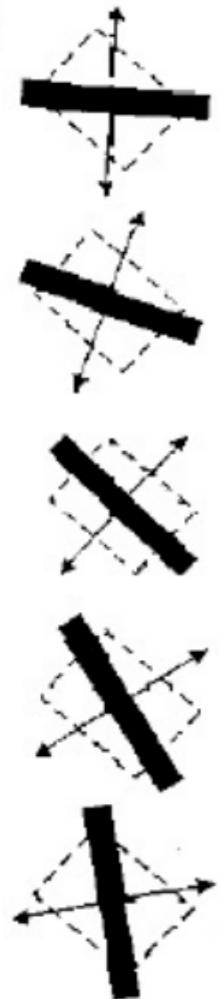
SIMPLE

$$\begin{array}{r} \text{XXIX} \\ + \text{XXXIII} \\ \hline \text{LXII} \end{array}$$

Difficile

Certaines représentations peuvent permettre des algorithmes plus faciles.

Exemple le plus célèbre : “DéTECTEURS de bord” dans le système visuel



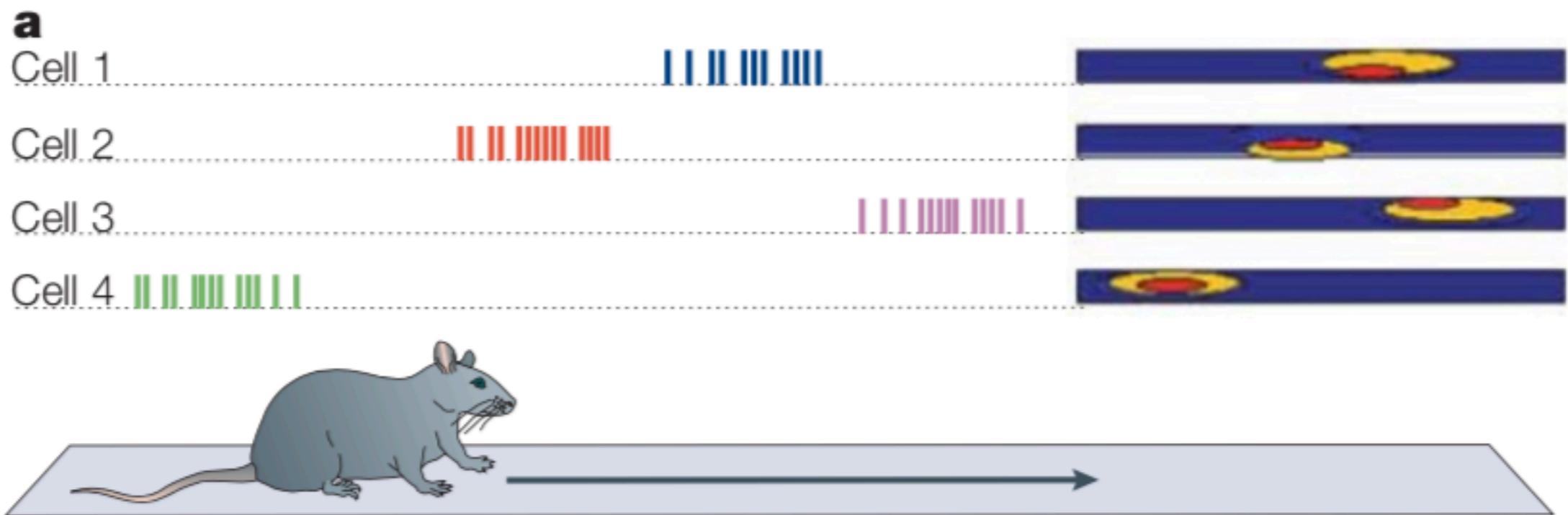
stimulus :
barre noire

activité dans le
cortex visuel (VI)



Autre exemple célèbre : “Place cells” dans l’hippocampe

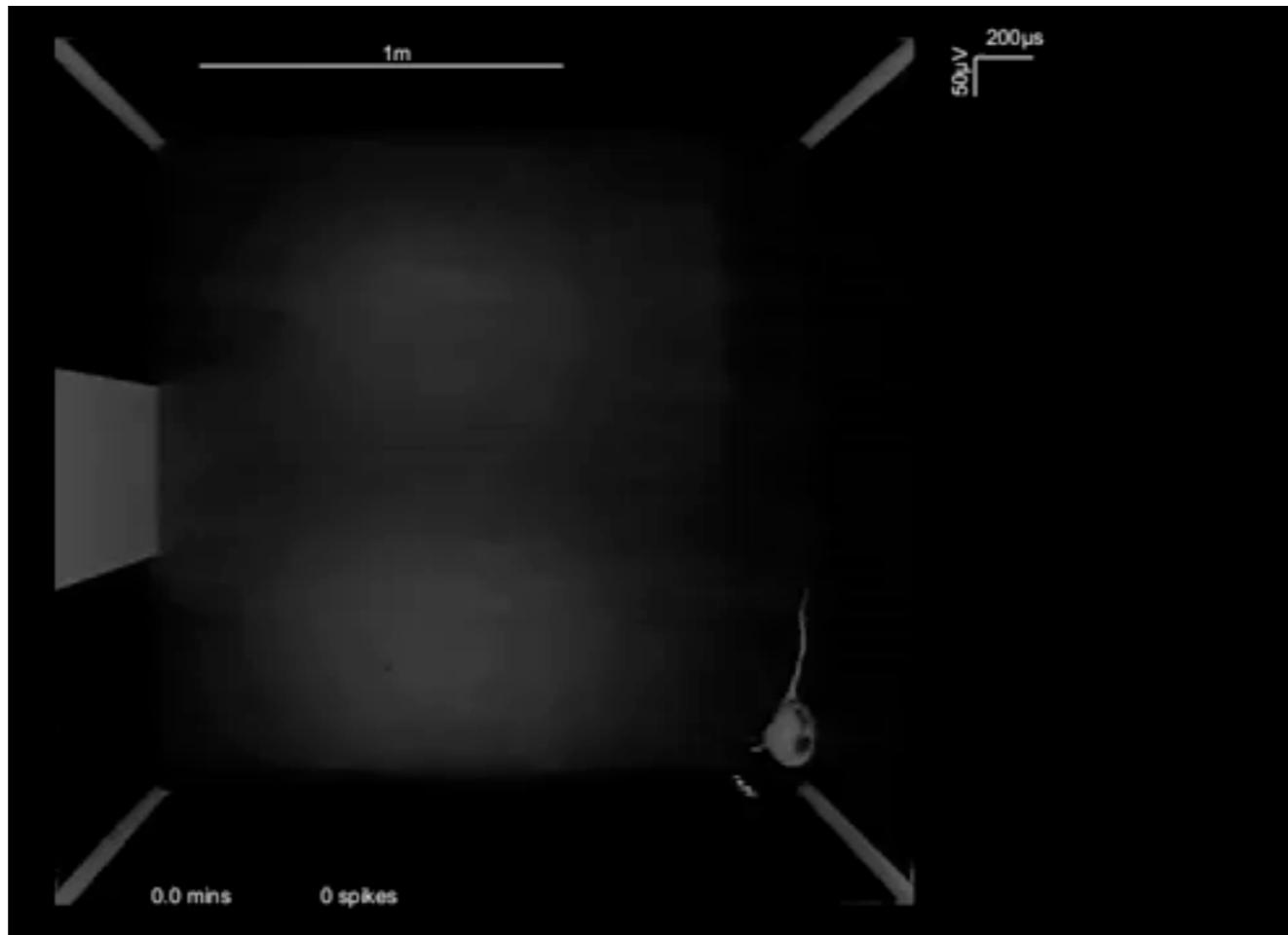
Rongeur sur une piste linéaire



Nakazawa et al., 2004

Autre exemple célèbre : “Place cells” dans l’hippocampe

Enregistrement de neurones dans l’hippocampe
d’un rat en “champ libre”.



Courtesy Roddy Grieves

Autre exemple célèbre : “Place cells” dans l’hippocampe

Enregistrement de neurones dans l’hippocampe
d’un rat en “champ libre”.



Courtesy Roddy Grieves

Étudier les représentations dans le cerveau

Travail expérimental :

- représentations *perceptives* : vision, audition, odorat etc.
- représentation des *variables motrices*
- représentations *d'ordre supérieur (cognitif)* :
 - décisions
 - mémoire à court terme
 - récompenses
 - rêves
 - incertitudes
 - ...

Travail théorique :

- quantifier le contenu d'information
- essayer de caractériser le *code neuronal*
- comprendre les problèmes de calcul (caractériser *les algorithmes*) :
 - reconnaissance d'objet
 - reconnaissance sonore
 - maximisation de récompense
 - ...

Ce que nous savons déjà

Ce que nous savons déjà

Très peu.

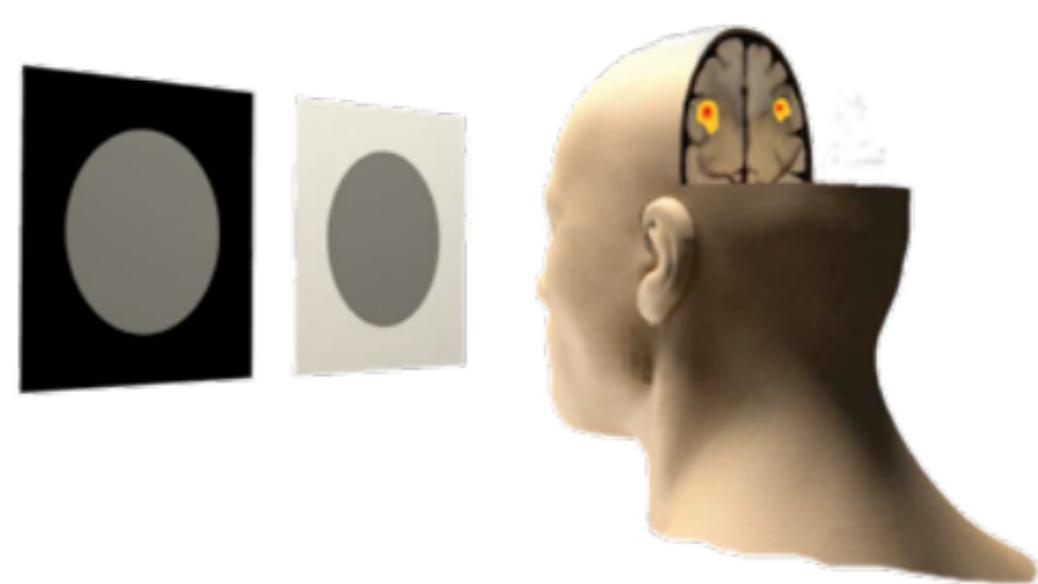
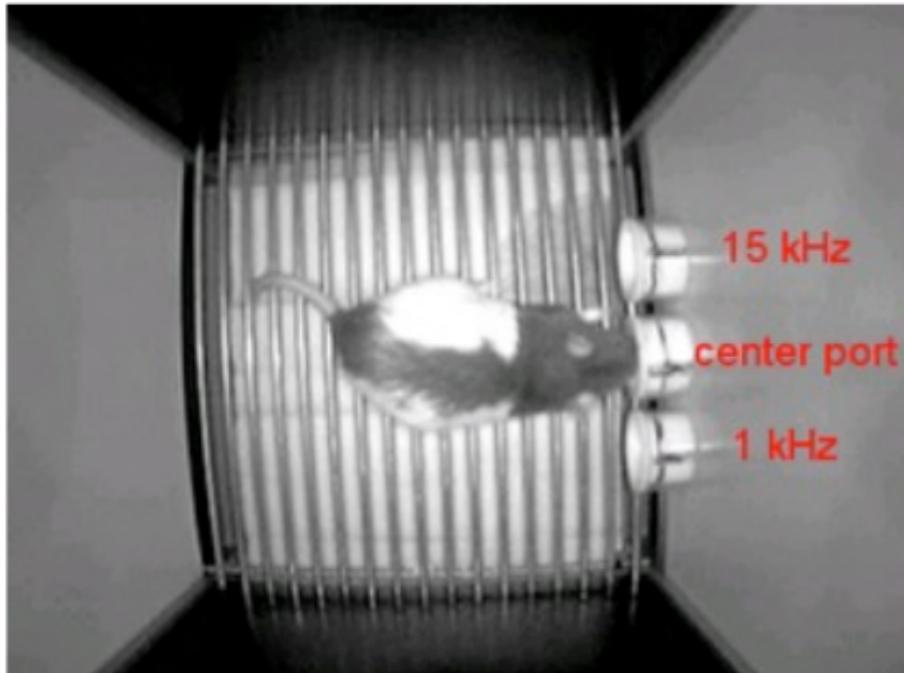
Ce que nous savons déjà

Très peu.



Ce dont nous avons besoin

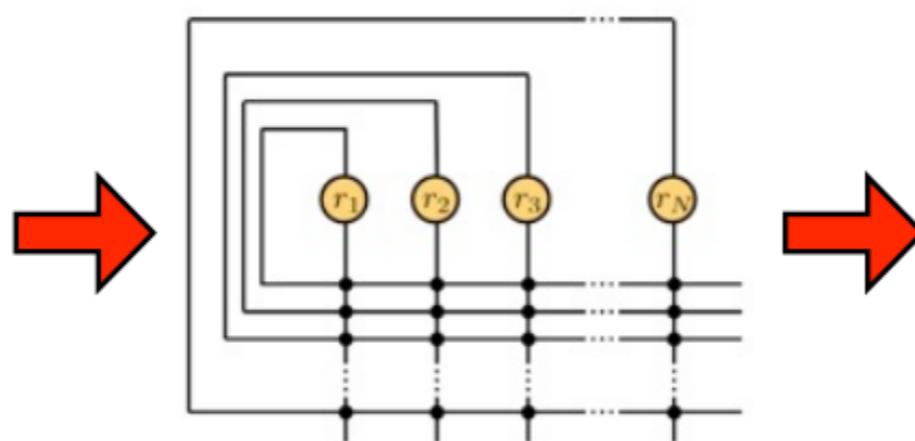
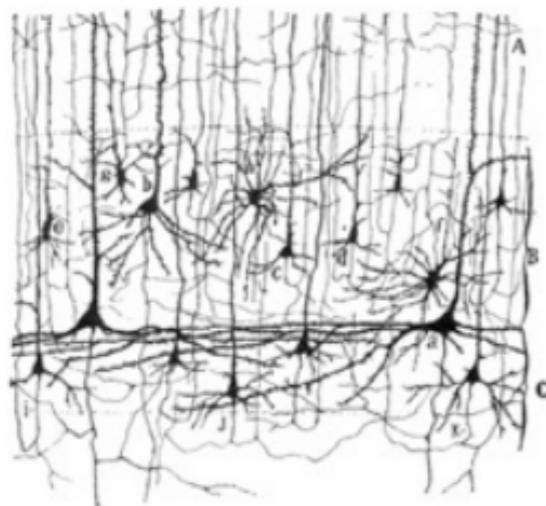
- Biologistes et psychologues :



- Pour sonder les cerveaux des animaux et des humains
- Concevoir et réaliser des expériences pertinentes
- Analyser et quantifier le comportement humain et animal

Ce dont nous avons besoin

- Physicien.ne.s, informaticien.ne.s, et mathématicien.ne.s :

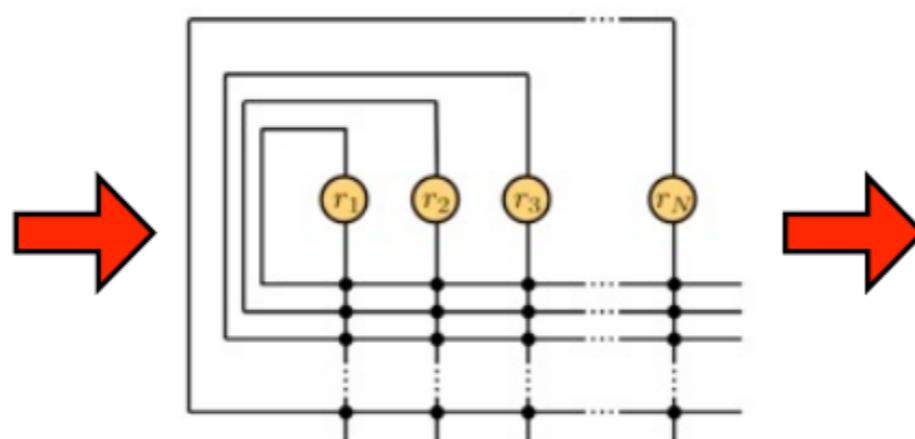
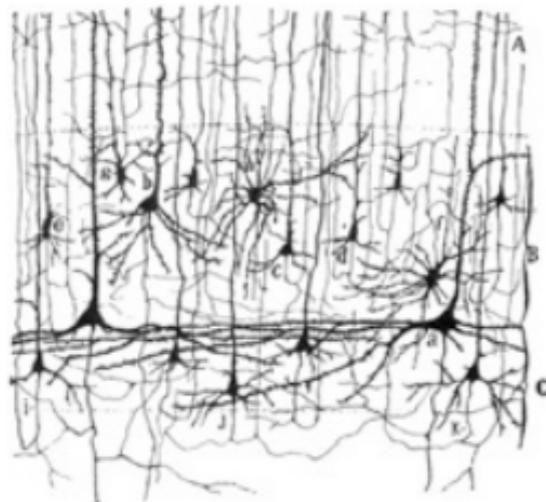


$$\dot{r}_1 = -r_1 + f\left(\sum_{j=1}^N w_{1j}r_j + E_1\right)$$
$$\dot{r}_2 = -r_2 + f\left(\sum_{j=1}^N w_{2j}r_j + E_2\right)$$

- Pour créer des modèles biophysiques des neurones et des réseaux.
- Formuler des théories mathématiques du traitement de l'information.

Ce dont nous avons besoin

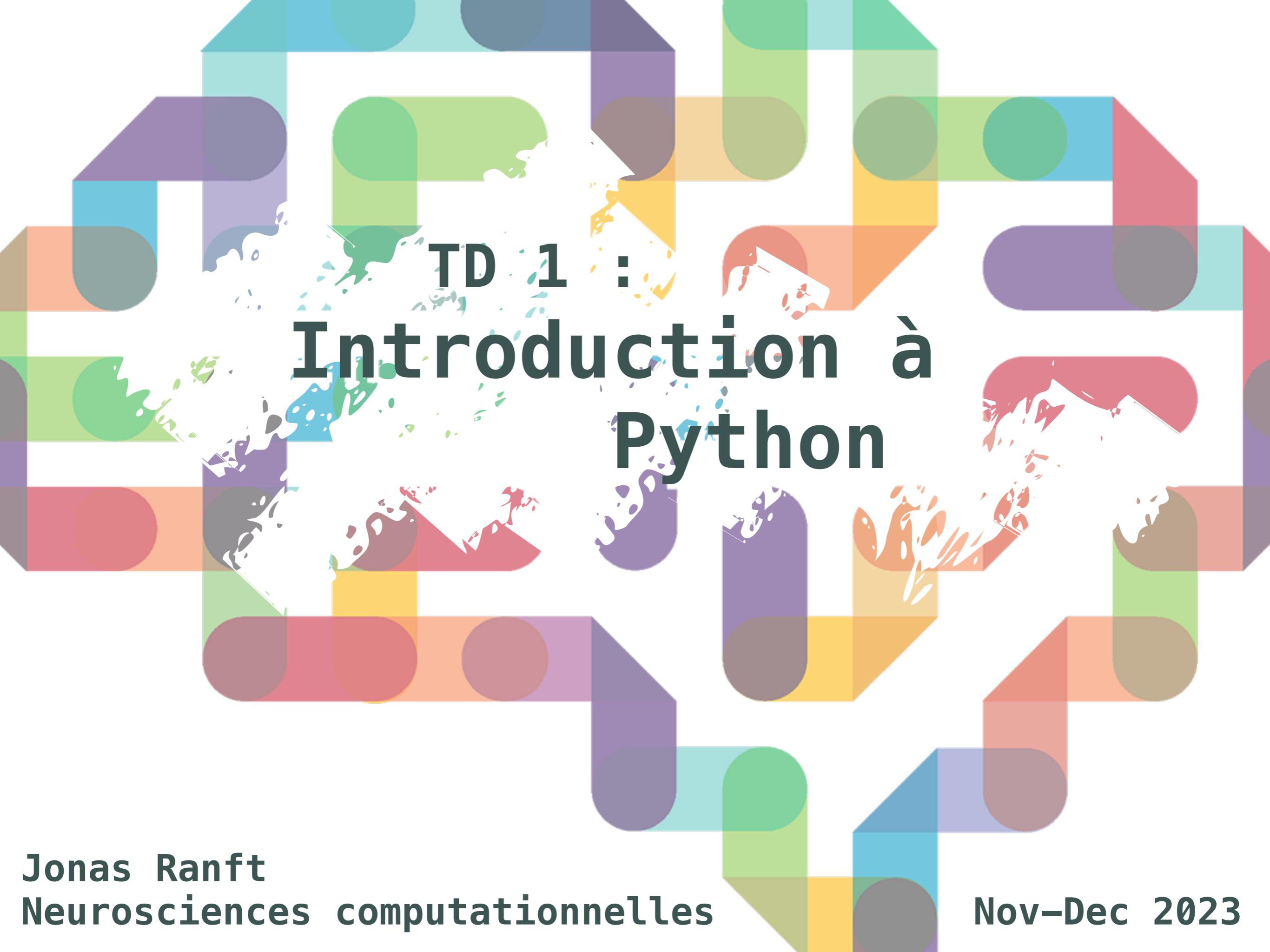
- Physicien.ne.s, informaticien.ne.s, et mathématicien.ne.s :



$$\dot{r}_1 = -r_1 + f\left(\sum_{j=1}^N w_{1j} r_j + E_1\right)$$
$$\dot{r}_2 = -r_2 + f\left(\sum_{j=1}^N w_{2j} r_j + E_2\right)$$

- Pour créer des modèles biophysiques des neurones et des réseaux.
- Formuler des théories mathématiques du traitement de l'information.

Les neurosciences, c'est une entreprise *interdisciplinaire* !



TD 1 : Introduction à Python

Jonas Ranft
Neurosciences computationnelles

Nov–Dec 2023

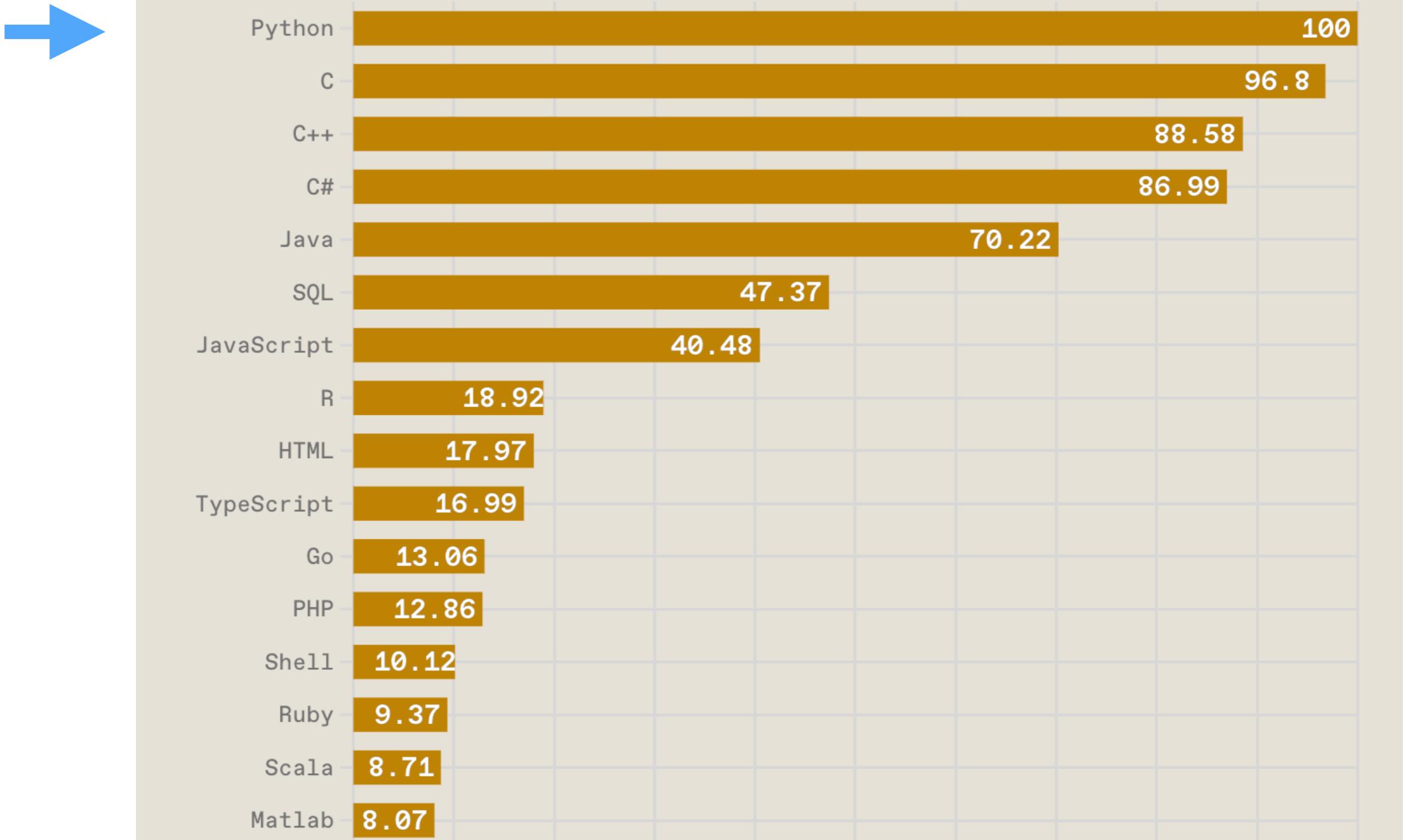
Qu'est-ce que Python ?

Un langage de programmation simple, puissant, généraliste

- récente (depuis 1991)
- gratuite et “open source”
- interprété (pas de compilation nécessaire)
- l'accent est mis sur la lisibilité du code
- les concepts peuvent être exprimés en moins de lignes que C/C++ ou Java
- vastes bibliothèques, fonctions disponibles
- visualisation facile

Python est très utilisé

Les langages de programmation les plus populaires (2022)



Syntaxe très claire, lisible → facile à apprendre

```
Entrée [1]: print("Hello world!")
```

```
Hello world!
```

```
Entrée [2]: # comments are preceded by #s
```

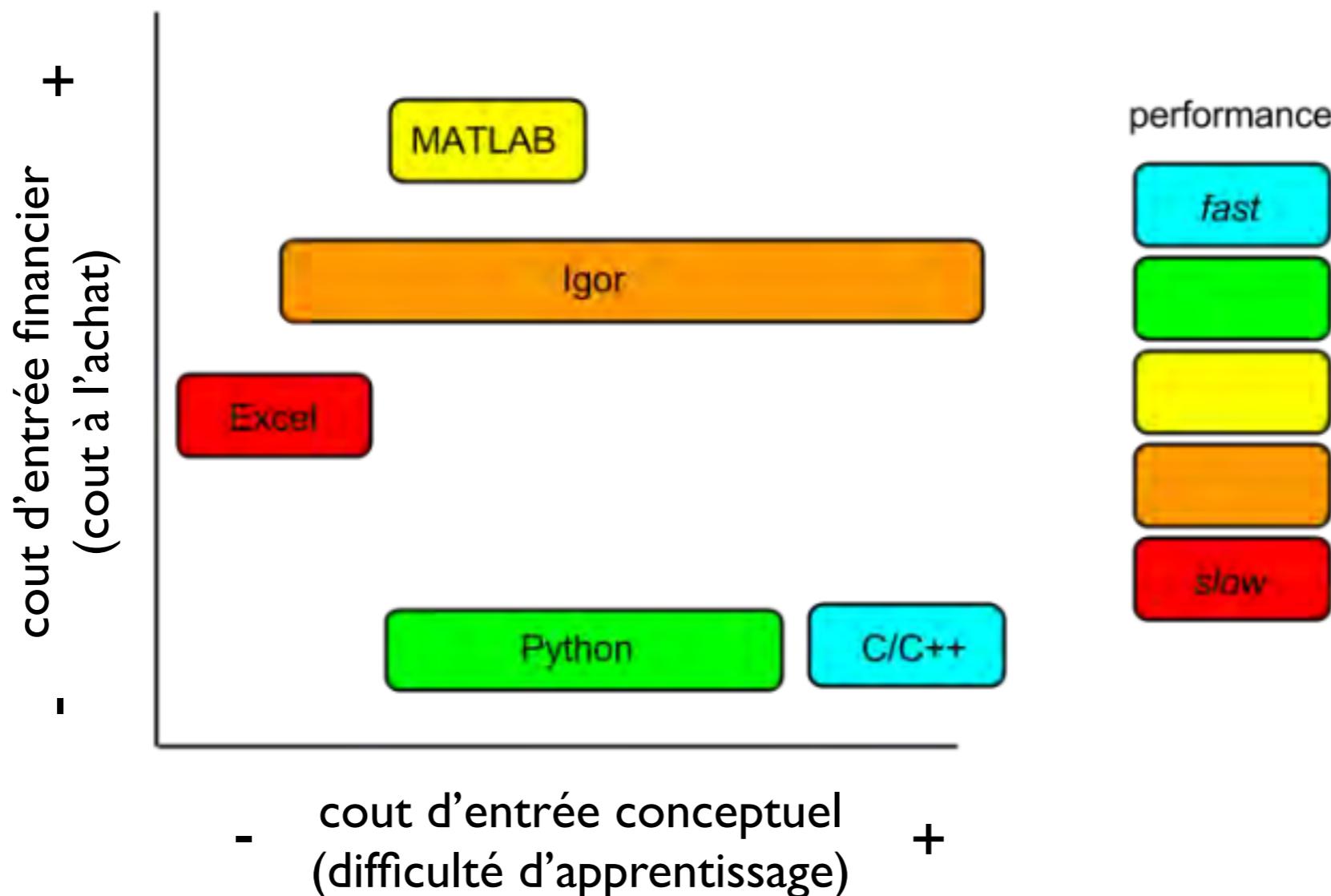
```
# we can easily define our own functions
def my_add_function(x,y):
    return x + y

result = my_add_function(3, 5)

print(result)
```

Excellent ratio retour sur investissement

Gratuit & peu de barrières conceptuelles



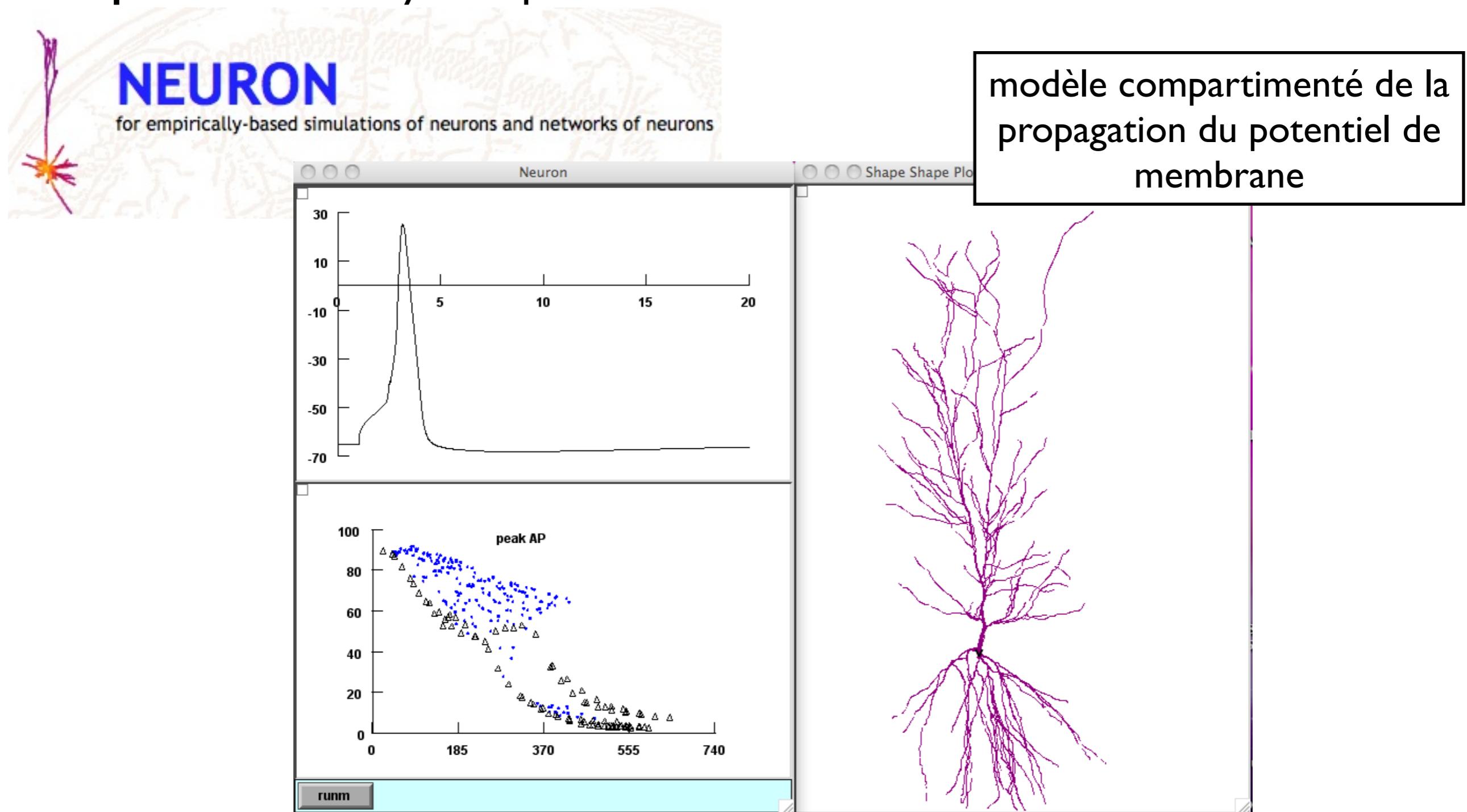
Bibliothèques et modules (standard et tierces)

Il existe beaucoup de **bibliothèques** (ou **modules**) avec des implémentations de fonctions de toute sorte:

- Analyse de données (*panda*, *numpy*, *scipy*, ...)
- Apprentissage machine (*scikit-learn*, *theano*, *TensorFlow*, ...)
- Calcul scientifique (*numpy*, *scipy*, *numba*, ...)
- Visualisation & génération de plots (*matplotlib*, *bokeh*, ...)
- Programmation web
- Création d'interface graphique
- Contrôle d'interfaces matérielles (Arduino, Raspberry Pi, ...)
- Simulateurs des neurones et de réseaux neuronaux (*Neuron*, *brian2*, ...)

Python pour neurosciences: Simulateur de neurone

Exemple : interface Python pour NEURON



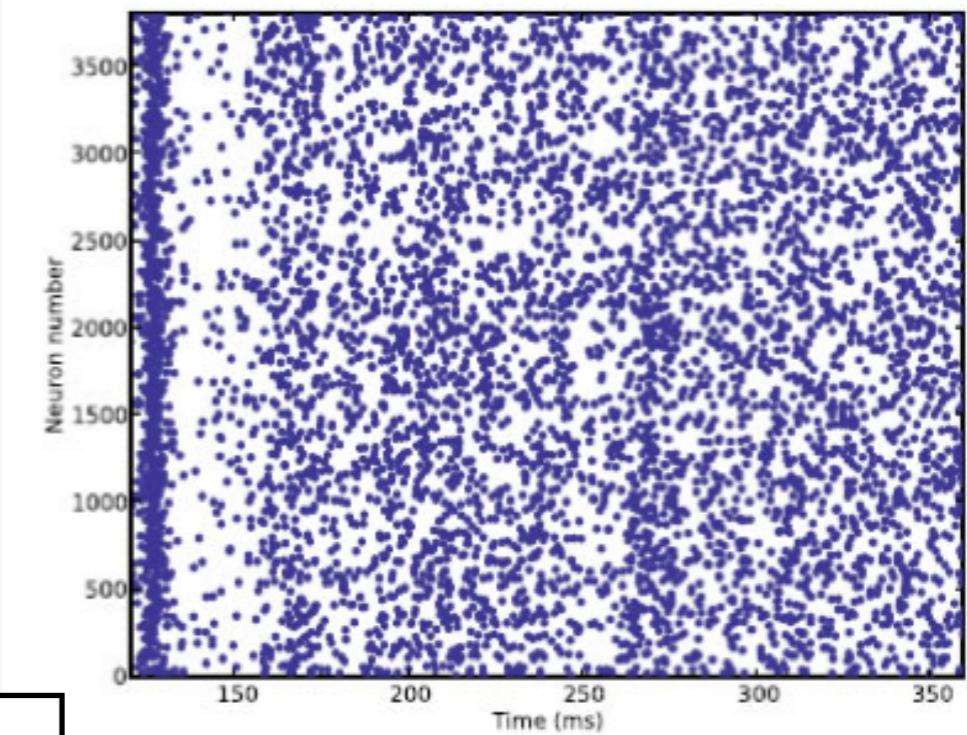
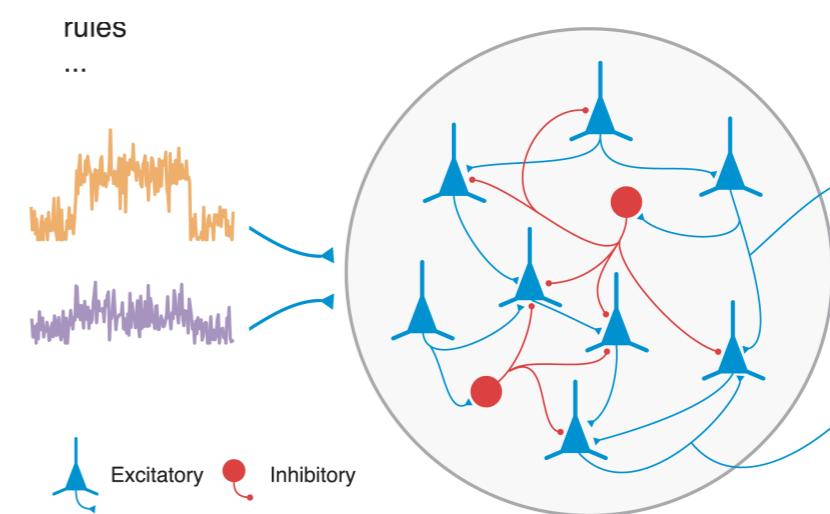
Python pour neurosciences : Simulateur de réseau

Exemple : Brian, un simulateur de réseau de neurones à spike

BRIAN

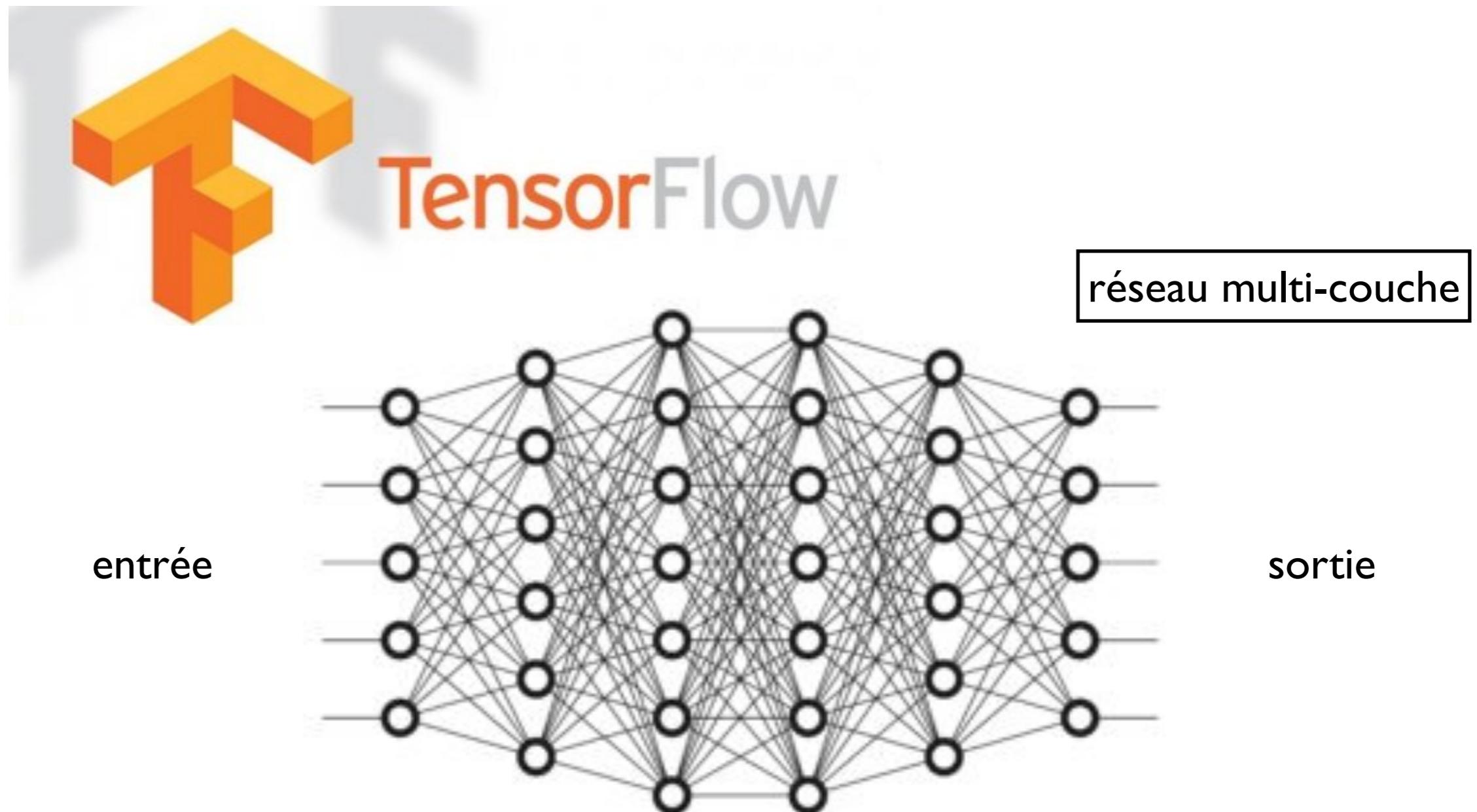
```
1 from brian import *
2 eqs = '''
3 dv/dt = (ge+gi-(v+49*mV)) / (20*ms) : volt
4 dge/dt = -ge/(5*ms) : volt
5 dgi/dt = -gi/(10*ms) : volt
6 '''
7 P = NeuronGroup(4000, eqs, threshold=-50*mV, reset=-60*mV)
8 P.v = -60*mV+10*mV*rand(len(P))
9 Pe = P.subgroup(3200)
10 Pi = P.subgroup(800)
11 Ce = Connection(Pe, P, 'ge', weight=1.62*mV, sparseness=0.02)
12 Ci = Connection(Pi, P, 'gi', weight=-9*mV, sparseness=0.02)
13 M = SpikeMonitor(P)
14 run(1*second)
15 raster_plot(M)
16 show()
```

réseau récurrent, connexions aléatoires



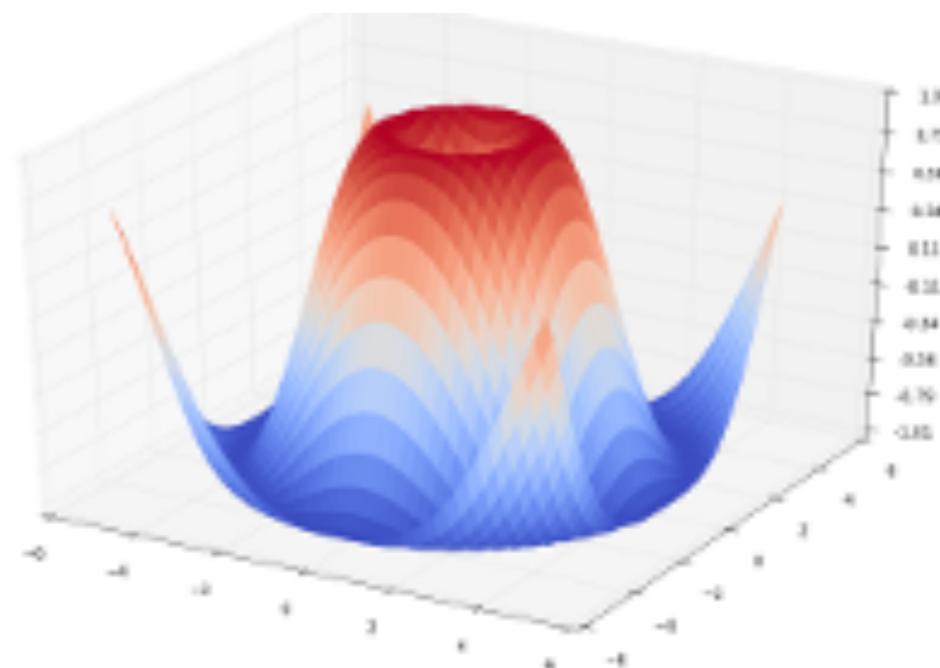
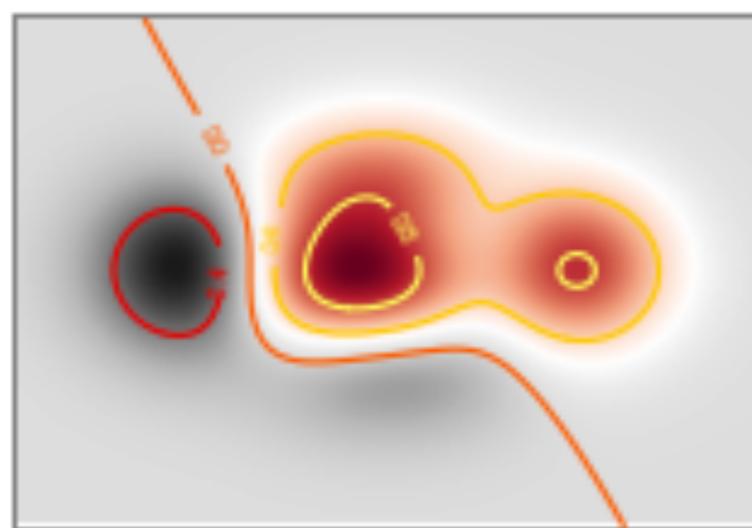
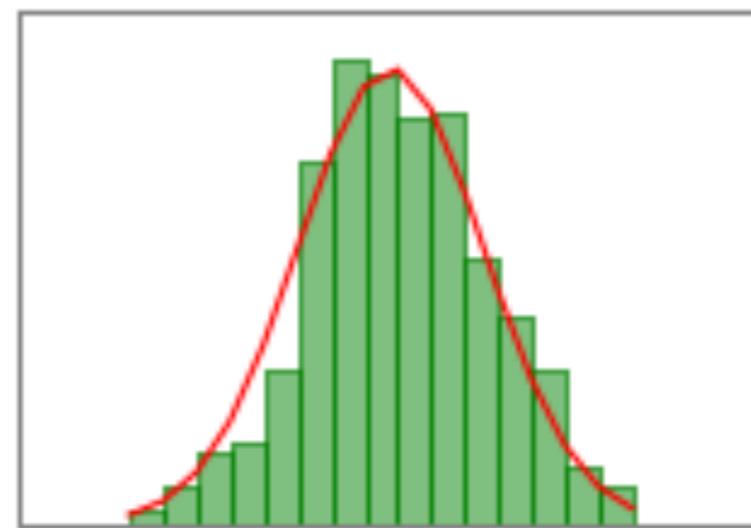
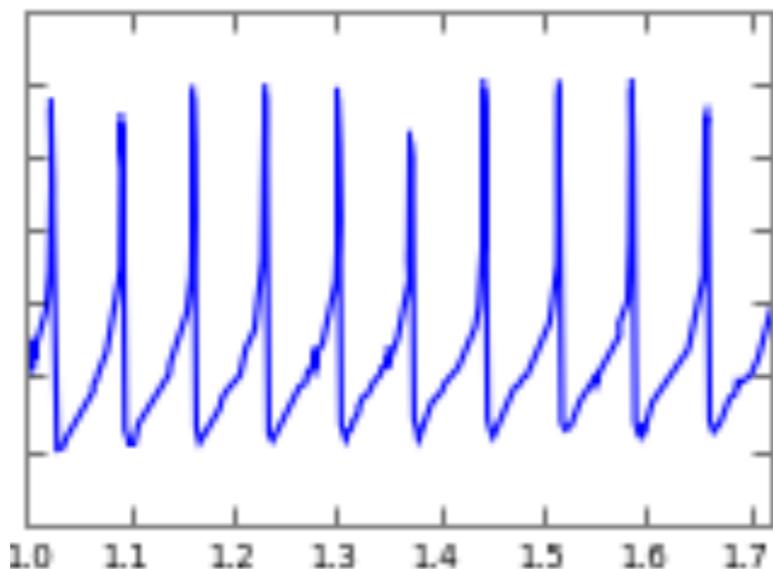
Python pour neurosciences : Apprentissage machine/profond

Exemple : TensorFlow - entraîner et utiliser des réseaux “profond” (“deep learning”)



Python pour neurosciences : Visualisation

Exemple : Matplotlib, une bibliothèque pour créer des plots et graphiques



Interpréteur et IDEs

- **IPython** : interpréteur de ligne de commande (shell) interactive ; shell & introspection améliorée : mise en évidence du code, auto-complétion, etc.
- **IPython/jupyter notebook** :
 - interpréteur de ligne de commande dans le navigateur
 - combine l'exécution de code, le texte riche, les équations mathématiques, les tracés et le rich media
- **Spyder** : Scientific PYthon Development EnviRonment
- **PyCharm** : environnement de développement professionnel
- **Atom, ..., Vim, Emacs**
- **Jython** : un autre interpréteur python écrit en java au lieu de c
- **IronPython** : une implémentation python pour le framework .NET

Interpréteur et IDEs

- **IPython** : interpréteur de ligne de commande (shell) interactive ; shell & introspection améliorée : mise en évidence du code, auto-complétion, etc.
- **IPython/jupyter notebook** :
 - interpréteur de ligne de commande dans le navigateur
 - combine l'exécution de code, le texte riche, les équations mathématiques, les tracés et le rich media
- **Spyder** : Scientific PYthon Development EnviRonment
- **PyCharm** : environnement de développement professionnel
- **Atom, ..., Vim, Emacs**
- **Jython** : un autre interpréteur python écrit en java au lieu de c
- **IronPython** : une implémentation python pour le framework .NET

dans ce cours