



# Hommage à Jacques Pitrat

Jean-Luc Dormoy



# Etait-ce une bonne idée de financer « l'équipe Pitrat » ?



1983-87 : Equipe IA @ EDF



1989 : Le séminaire EDF-  
IBM-Renault démarre

Jean-Marc David, Philippe  
Dague, Jean-Luc Dormoy



Il est remplacé par un séminaire EDF-LAFORIA  
puis LIP6

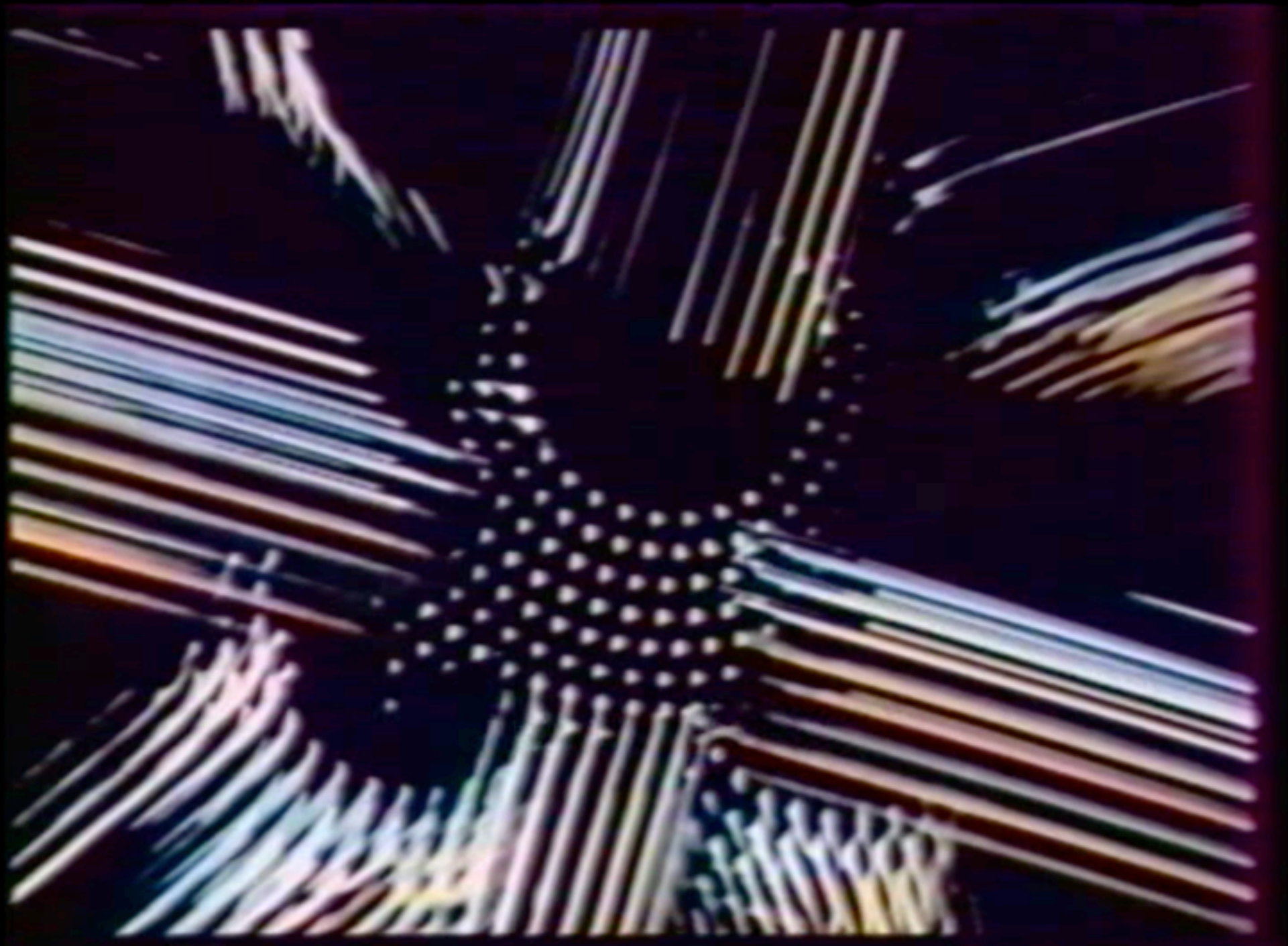


Est également pris en charge le séminaire annuel  
de trois jours « Métaconnaissance », devenu le  
séminaire « Berder » début septembre



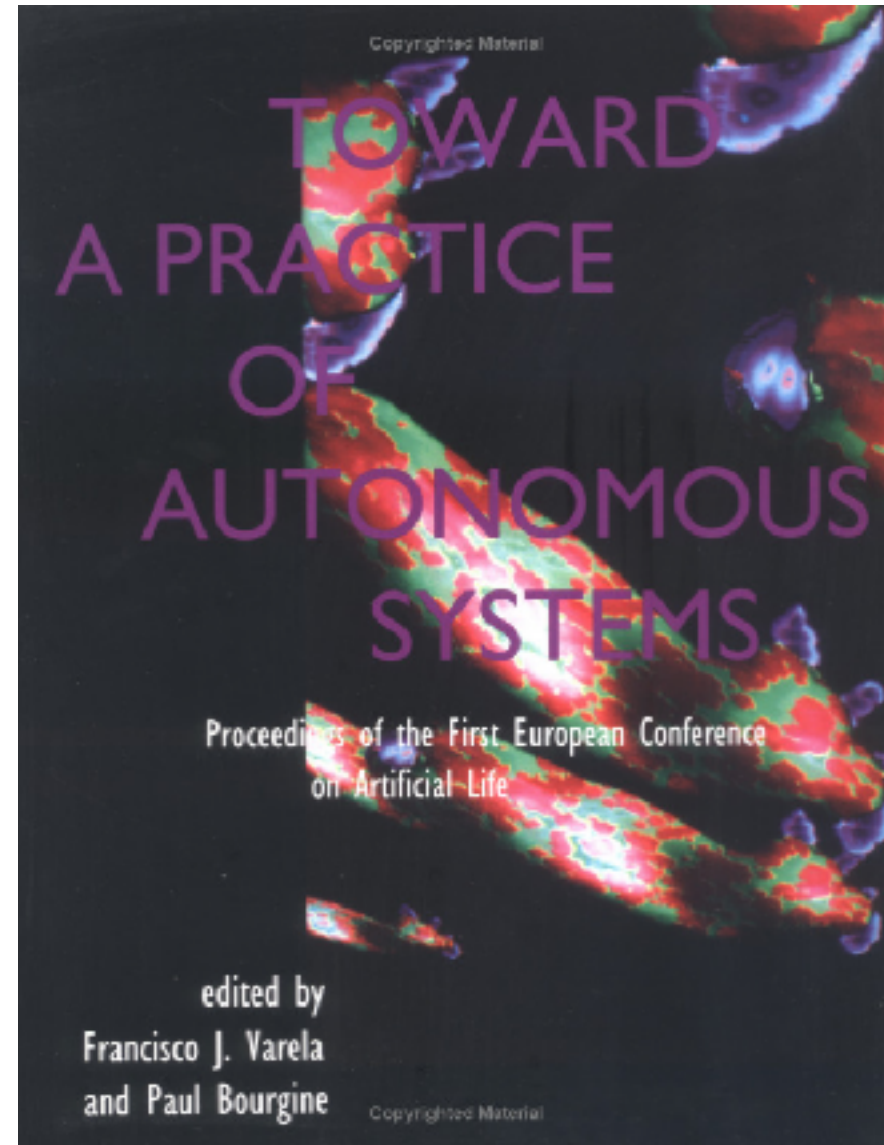
1983-1987







1991



## EPISTEMOLOGICAL ISSUES AND CONCEPTUAL FOUNDATIONS

- The Historical and Epistemological Ground of von Neumann's Theory of Self-Reproducing Automata and Theory of Games 385  
*Stefan Helmreich*
- Meta-Knowledge, Autonomy, and (Artificial) Evolution: Some Lessons Learnt So Far 392  
*Jean-Luc Dormoy and Sylvie Kornman*
- Feedforward: The Ethological Basis of Animal Learning 399  
*R. Allen Gardner and Beatrix T. Gardner*
- Computing at the Tissue/Organ Level (with Particular Reference to the Liver) 411  
*R. C. Paton, H. S. Nwana, M. J. R. Shave, and T. J. M. Bench-Capon*
- The Connection between AI and Biology in the Study of Behaviour 421  
*Barbara Webb and Tim Smithers*
- Amoeba, Planaria, and Dreaming Machines 429  
*Bruno Marchal*
- Selectionist Systems as Cognitive Systems 441  
*Bernard Manderick*

## Meta-Knowledge, Autonomy, and (Artificial) Evolution: Some Lessons Learnt So Far

Jean-Luc Dormoy  
EDF R&D Center  
IMA-TIEM  
1, avenue du Général De Gaulle  
92141 Clamart Cedex, France

Sylvie Kornman  
LAFORA  
Paris 6 University  
4, place Jussieu  
75251 Paris Cedex 05, France

### Abstract

We claim in this paper that an extensive use of meta-mechanisms is a very powerful tool for building autonomous AI & AL systems. We support that claim by examples of *knowledge-based systems* exhibiting unexpected and partly autonomous behaviors. They show that autonomy, as well as viability, could be achieved in the future by means of meta-mechanisms. In particular, meta-mechanisms require simpler mechanisms than multi-agent-like or emergent mechanisms, though they can achieve more sophisticated behaviors.

We also present the perspective of bootstrapping as a basic methodology for building AI & AL systems. Its motivation is that building AI or AL systems is probably too difficult a problem to be tackled by just finding out the right components and their assembling them. These systems must go through a sequence of evolutionary steps, and we must be aware of that fact.

On one hand, bootstrapping means that any new component should be made *applicable to the system as a whole* - and not only *participate* to the system's behavior - in order to help further extensions. On the other hand, the strategy used when building a system should *not* be mainly concerned by immediate performance, *but* by further extensions. This means that new functions or components should be added to the system *in order to* make further extensions possible or easier. We describe some experiments in building AI systems according to this strategy, and its constraints and difficulties. Indeed, we have no clean theory of bootstrapping yet, instead mere intuitions. So, these experiments mainly aim at figuring out how the design of a sophisticated system through bootstrapping steps should be conducted. We think that, eventually, a theory will be required.

### 1 Introduction

A common view of AL and AI system architectures is based on *multi-agent architectures*. In this view, a set of

non-intelligent small agents interact with each other, and from this interaction emerges a global behavior. In a practical way, the physical implementation of the agents share and act through a common substrate - e.g. data structures in 'symbolic' multi-agent systems, or neurons in neural networks, but they are of a different nature. Their own substrate, i.e. their physical implementation, does not intersect the substrate they act on.

The aim of this paper is to discuss the meta view, i.e. a view where agents can also *act on* agents. This view is not new in any way, but we think it can be useful to reconsider it with respect to refurbished problems stressed by the AL community, such as autonomy, viability and evolution. We make various claims, and we show how they are justified by some examples of previous AI systems, and by the systems we are building, namely the Sual and Sade/Meta-Hari systems.

We first claim that *meta is powerful*. Indeed, a simple mechanism achieves more when properly used at the meta-level rather than at the basic level. Acting at the meta-level means that the components implementing the mechanism act on the physical implementation of other components, whereas acting at the basic level means merely interacting with them. Conversely, the designer who intends to build a system exhibiting a given behavior would better choose a meta-level architecture.

Secondly we re-consider the autonomy and viability problems with respect to meta-level architectures. In particular, an operational closure of the system<sup>1</sup> can be achieved by simpler mechanisms in a meta architecture. Autonomy can be enhanced in meta-level architectures by the fact that components can be used at any level. Indeed, from the designer's point of view, this is a saving principle: let components act on components, and you will need less, because a component can participate to various functions at various levels. Viability can be tackled by specific-purpose components, which for example observe other components, and try to fix improper behaviors.

Thirdly, we discuss the bootstrapping problem of AI or AL systems. Bootstrapping has been a long-known technique for building compilers, for example. It consists of using an already-existing version of a system for running a new version. When systems have the capability of processing themselves, the old version can

<sup>1</sup> In the sense of Varela

1999











## INTRODUCTION

Par Jacques Pflrat

CNRS-LIP6

Université Pierre et Marie Curie

L'équipe Métacognition du LIP6 et plusieurs chercheurs d'EDF se sont réunis dans l'île de Berder du 22 au 24 septembre 1999. Ce rapport contient les textes de quelques-unes des interventions qui y ont été faites.

Un concept "méta" important est apparu directement ou indirectement dans de nombreux papiers : c'est le concept de monitoring où le système examine ce qu'il fait pour mieux le faire ensuite. Après une présentation générale de l'intérêt et des problèmes soulevés par le monitoring, nous allons le retrouver plusieurs fois. Un outil indispensable pour mettre en œuvre un monitoring est d'avoir un métamoteur : il permet de savoir ce qui se passe et de changer éventuellement la stratégie d'utilisation des connaissances. Le métamoteur de SEPTARZ peut même faire des expériences en cours de résolution afin de déterminer dynamiquement les meilleurs paramètres pour traiter l'expertise de base utilisés ; les résultats obtenus montrent effectivement une amélioration des performances. La structure de métamoteur également adoptée pour IDREB permet aussi d'avoir des métarègles d'introspection qui permet au système de savoir ce qu'il fait et pourquoi il le fait, donc de changer dynamiquement de stratégie de résolution. Pour le système MARECHAL, le monitoring est au cœur du projet, puisqu'il s'agit, dans une application à des wagons, d'arbitrer les conflits qui peuvent venir de ce que plusieurs méthodes peuvent être utilisables au même moment ; il faut analyser ce qu'elles donnent pour en déduire une solution globale.

Le monitoring est apparu de façon moins centrale dans plusieurs autres papiers. Par exemple, GENSAM doit généraliser à partir de petits échantillons. Il existe de nombreuses méthodes qui permettent de le faire, chacune ayant ses avantages et ses inconvénients qu'il faut balancer dans chaque situation. La métacognition de contrôle donnée dans la section 5.2 est un excellent exemple d'expertise de monitoring. Dans l'annexe 6 de son article, D. Pastre donne un exemple de monitoring qu'elle a fait au cours de sa recherche de la démonstration d'un théorème. Par ailleurs, l'analyse des résultats du tournoi de Football montre des erreurs des programmes dues à une insuffisance du monitoring, par exemple quand ils ne voient pas qu'ils se retrouvent après chaque coup dans une position quasi-identique à la précédente, mais un peu plus défavorable.

Plusieurs papiers présentent des travaux en cours de réalisation dans l'équipe. D'abord deux applications à la démonstration de théorèmes. Nous avons une analyse des résultats obtenus par MUSCADET à la compétition de démonstrateurs organisée dans le cadre des conférences CADE ; nous voyons aussi l'enrichissement du système pour mieux résoudre les problèmes de la TPT Problem Library. A mi-chemin entre la démonstration de théorèmes et l'ETAO, le système de J.-P. Spagnol devra être capable de produire des démonstrations adaptées au niveau des élèves. Enfin, un système d'ETAO, comme COMADEN ?, se doit d'avoir des interfaces bien adaptées aux divers dialogues que le système aura avec l'élève.

Un problème délicat, mais qui se présente souvent, est la généralisation à partir d'un nombre restreint d'observations : le système GENSAM en expérimente des solutions dans le domaine médical. Le projet CASSICE reconnaît automatiquement les manœuvres exécutées par le conducteur d'un véhicule. Naturellement, les jeux sont toujours un terrain de prédilection pour l'IA. Trop souvent, on a étudié les jeux à information complète où de nombreuses difficultés n'apparaissent pas. C'est pourquoi F. Kacik prend comme sujet d'étude les jeux de carte, mais naturellement dans le cadre de systèmes généraux qui reçoivent comme données les règles du jeu particulier auquel ils doivent jouer. Enfin, T. Casanova fait le compte-rendu de l'expérience qu'il a organisée ; nous y avons tous pris part en créant dans un temps limité un programme pour jouer à un jeu que nous ne connaissions pas : le football des philosophes, appelé aussi Phutball.

Nous remercions Fabrice Kacik et Tristan Pennéret pour avoir si parfaitement préparé l'organisation matérielle de ce colloque, qui a pu avoir lieu grâce à un contrat EDF. Ils ont également pris en charge l'édition de ce rapport.



# SOMMAIRE

<b>Monitorer la recherche d'une solution</b> _____	<b>6</b>
<b>Gestion implicite de la concurrence dans un système à base de tableau noir</b> ____	<b>45</b>
<b>Le nouveau MUSCADET et la TPTP Problem Library</b> _____	<b>58</b>
<b>Automatisation du raisonnement et de la rédaction de preuves en géométrie</b> _	<b>108</b>
<b>Spécification de dialogues et construction d'interfaces modulaires et réutilisables</b> _____	<b>127</b>
<b>GENSAM: un système généralisant les petits échantillons</b> _____	<b>140</b>
<b>Un Système Expert et un Outil pour la Reconnaissance Automatique des Manœuvres Automobiles</b> _____	<b>148</b>
<b>Un système général de jeu de cartes</b> _____	<b>170</b>
<b>Un domaine intéressant pour l'IA</b> _____	<b>170</b>
<b>Un tournoi de programmes de Phutball</b> _____	<b>180</b>

Automatic programming:  
 JYLucas, CJimenez-Domingue

LE BUISSON ARDENT



# Le bootstrap

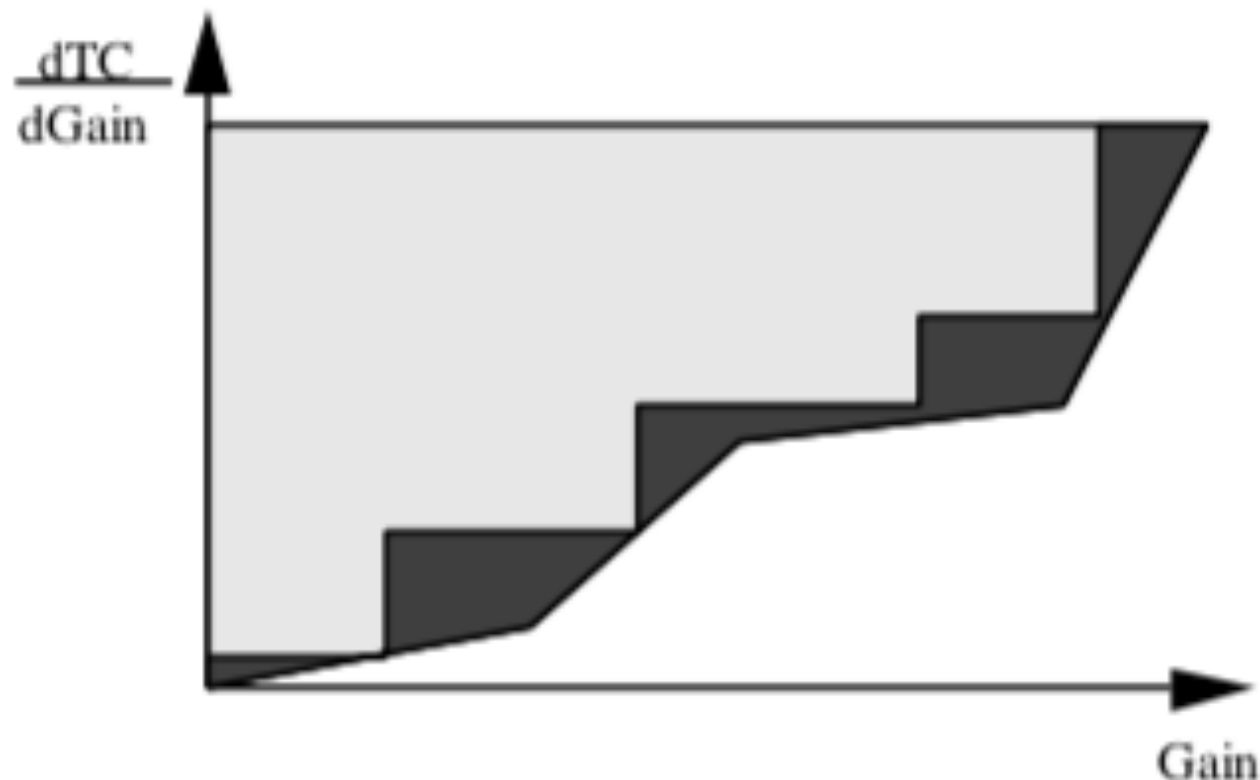
L'homme est trop limité pour construire une AGI seul



# Programmer c'est trop dur...

- Machine Learning (par exemple Go)
    - Les limitations de l'homme dans la résolution de problèmes sont visibles
    - Les limitations de l'homme dans la programmation sont encore plus visibles
  - Conception générale de systèmes
    - Organe et fonction : appliquer l'un à l'autre et l'autre à l'un
-

La raison de cet échec est que nous avons voulu faire une étape de bootstrap trop grande et trop directe. Pour schématiser ce point de vue, on peut considérer le graphique de la figure 3.



*Gain = gain dû à l'ajout de connaissances*

*TC = temps de conception*

*Figure 3 : Le tunnel du bootstrapping*

Organe



Fonction





# La conscience

Vous êtes l'objet de mes pensées...

Méta-Pensée



Pensée



# Vérifier c'est trop dur...

- Machine Learning (par exemple Voiture autonome)
    - Les limitations de l'homme dans la conduite de systèmes sûrs sont visibles
    - Les limitations de l'homme dans la conception de systèmes sûrs encore plus visibles
  - Conception générale de systèmes
    - Laisser la vérification au fonctionnement
    - (Apprendre la vérification)
-

# Prochaines étapes en IA

Que faut-il financer ?



Artificial Intelligence  
Artificial General Intelligence  
Artificial Super Intelligence?

NOKIA Bell Labs



# DARPA Just Announced a \$2B Research Program on Artificial Intelligence

## First \$200M on Common Sense

**TA2: Test Environment**

Video & Simulation Experiences

Predictions, Expectations, & Simulation Actions

Simulated Commonsense Agent



**TA1: Foundations of Human Common Sense**

**AI Beginner**

Learns the foundations of common sense from simulated experiences

**AI2 Benchmarks for Common Sense**

Natural Language & Image-based Questions

True/False & Multiple Choice Answers

Broad Commonsense QA Service



**TA3: Broad Common Knowledge**

**AI Librarian**

Learns common sense by reading and extracting knowledge from the Web

1. The object recognition capabilities of a two year old
2. The language understanding capabilities of a four year old
3. The manual dexterity of a six year old
4. The social understanding of an eight year old





# Explorer...

- Améliorations du Machine Learning
  - Peu d'exemples
  - Non supervisé
- « Common sense »
- Ok...
  
- Sujets de fond
  - Bootstrap
  - Conscience
  - Vie...