



**HAL**  
open science

# Conceptual Vectors, Lexical Networks, Morphosyntactic Trees and Ants: a bestiary for Semantic Analysis

Mathieu Lafourcade

► **To cite this version:**

Mathieu Lafourcade. Conceptual Vectors, Lexical Networks, Morphosyntactic Trees and Ants: a bestiary for Semantic Analysis. SNLP'07: The 7th International Symposium on Natural Language Processing, Dec 2007, Pattaya, Chonburi, Thailand, Thailand. lirmm-00202648

**HAL Id: lirmm-00202648**

**<https://hal-lirmm.ccsd.cnrs.fr/lirmm-00202648>**

Submitted on 7 Jan 2008

**HAL** is a multi-disciplinary open access archive for the deposit and dissemination of scientific research documents, whether they are published or not. The documents may come from teaching and research institutions in France or abroad, or from public or private research centers.

L'archive ouverte pluridisciplinaire **HAL**, est destinée au dépôt et à la diffusion de documents scientifiques de niveau recherche, publiés ou non, émanant des établissements d'enseignement et de recherche français ou étrangers, des laboratoires publics ou privés.

# Conceptual Vectors, Lexical Networks, Morphosyntactic Trees, and Ants

a bestiary for Semantic Analysis



Mathieu LAFOURCADE

LIRMM-INFO  
Equipe TALN

# Outline

- Semantic Analysis      What to look at? – Applications
- Casting                  Conceptual Vectors – Analysis Trees – Lexical Network
- Ant algorithm          Principles - Without lexical network - With lexical network
- Ressource Production      Lexical Network - Conceptual Vectors
- Future directions          Toward holistic analysis: syntax + semantic

# Outline

- Semantic Analysis      What to look at? – Applications
- Casting                  Conceptual Vectors – Analysis Trees – Lexical Network
- Ant algorithm          Principles - Without lexical network - With lexical network
- Ressource Production      Lexical Network - Conceptual Vectors
- Future directions          Toward holistic analysis: syntax + semantic

# Semantic Analysis

What do we look at?

## Selection/weighting of acceptions (WSD)

“L’avocat plaide”

The lawyer pleads

avocat/fruit or avocat/justice ?

## Prepositional group attachment

“He saw the girl with a telescope”

The man sees (the girl with a telescope)  
/ The man (sees [the girl] with a telescope)

“Il (voit avec un télescope ) la fille” or

“Il voit la (fille avec un télescope)” ?

## Interpretation trails

“L’avocat est véreux” 2 trails but not 4

The avocado is rotten  
/ The lawyer is corrupted

# Semantic Analysis

What do we look at?

## Anaphora/reference resolution

“The lawyer defended well his client. He was acquitted.”

he = lawyer or he = client ?

## Recognition of occurrences of lexical function [meltchuk] [schwab]

“Il a une forte fièvre”

Magn(fièvre) = forte ?

He has a high fever

Magn(fever) = high ?

### Indexing for information retrieval [jaillet, prince, chauché, teissere]

WSD (mouse)  
increase in precision

Synonymy (cat/true cat/feline) / semantic fields (horse/saddle)  
increase in recall

### Machine Translation [prince, delorme]

Reference resolution (il => he/she/it ? Son => his/her/its ?)

Contrastive phenomena (river <= rivière/fleuve ?)  
(abats => ofals/giblets ?)

Lexical functions (forte fièvre <=> high fever) but (forte <=> high)

# Outline

Semantic Analysis      What to look at? – Applications

Casting      Conceptual Vectors – Analysis Trees – Lexical Network

Ant algorithm      Principles - Without lexical network - With lexical network

Ressource Production      Lexical Network - Conceptual Vectors

Future directions      Toward holistic analysis: syntax + semantic



# Casting Conceptual Vectors

Thematic representation [chauché, lafourcade]

Lexical item = set of ideas => vector

An idea => a concept

For example, 873 concepts in Larousse thesaurus (French)  
more than 1000 in Roget

(1)existence, (2)inexistence, (3)matérialité, ..., (516)liberté, ...,  
... (872)jeux, (873)jouets

A component of the vector is the activation of the related concept

A concept is itself a vector on the same space (neighbourhood of concepts)

Concept are interdependent => « Conceptual » vectors

Vector combinations : addition, contextualisation...

[lafourcade, prince, schwab]

# Conceptual Vectors

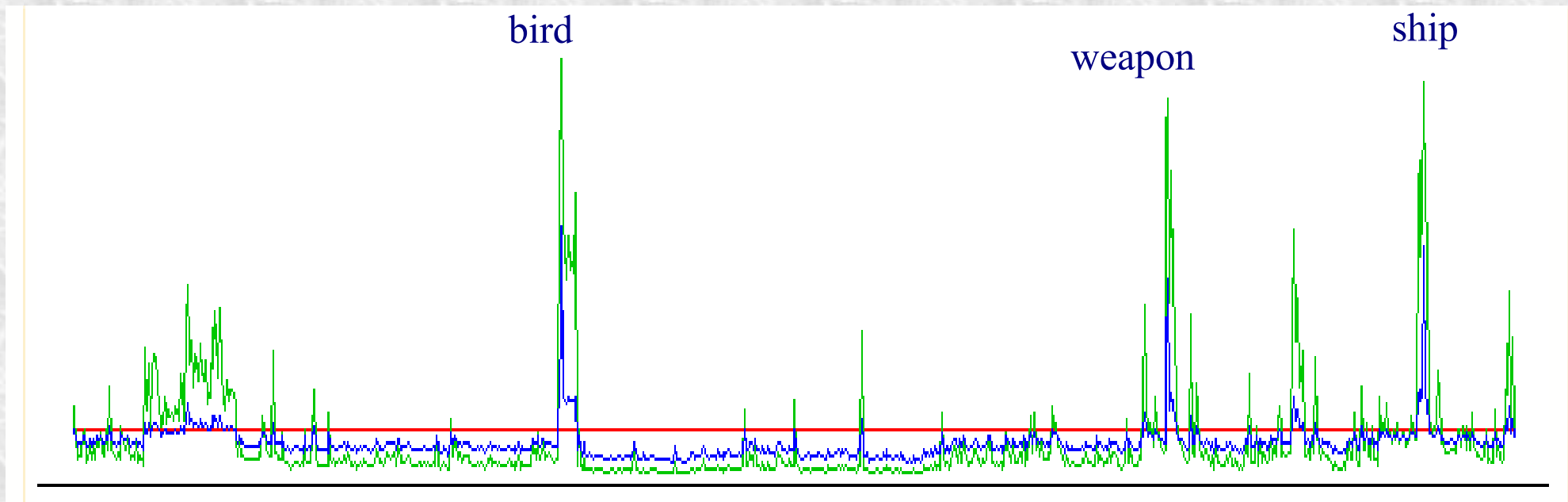
## Example

### Frigate

Ancient ship

Modern ship

Bird

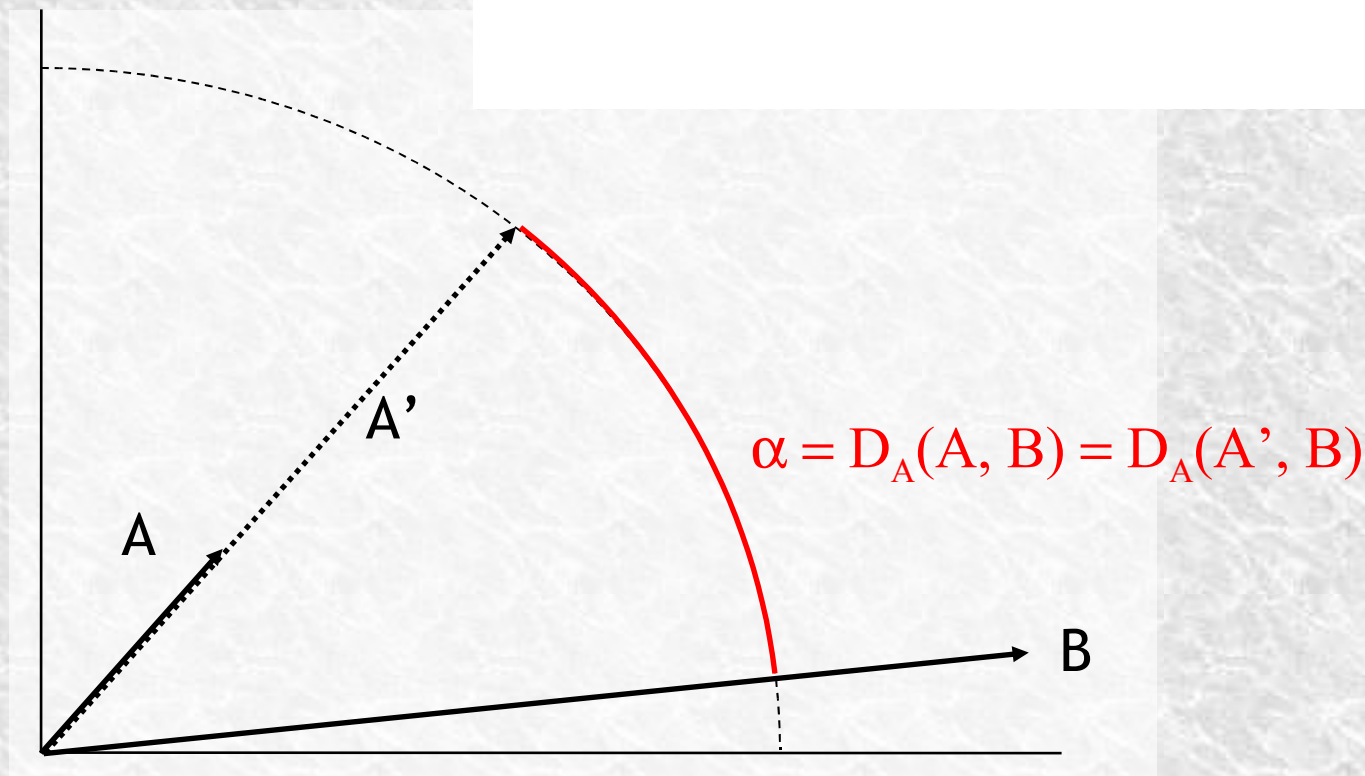


# Conceptual vectors

## Angular distance $D_A$

$$D_A(A, B) = \arccos(\text{Sim}(A, B))$$

$$\text{Sim}(A, B) = \cos(\widehat{A, B}) = \frac{A \cdot B}{\|A\| \times \|B\|}$$



# Conceptual Vectors

$$D_A(\textit{pelican}, \textit{pelican}) = 0 \text{ (} 0^\circ \text{)}$$



$$D_A(\textit{pelican}, \textit{white pelican}) = 0,2 \text{ (} 11^\circ \text{)}$$



$$D_A(\textit{pelican}, \textit{train}) = 1,22 \text{ (} 70^\circ \text{)}$$



$$D_A(\textit{pelican}, \textit{bird}) = 0,46 \text{ (} 26^\circ \text{)}$$



$$D_A(\textit{pelican}, \textit{seagull}) = 0,4 \text{ (} 23^\circ \text{)}$$



$$D_A(\textit{pelican}, \textit{fish}) = 0,35 \text{ (} 20^\circ \text{)}$$

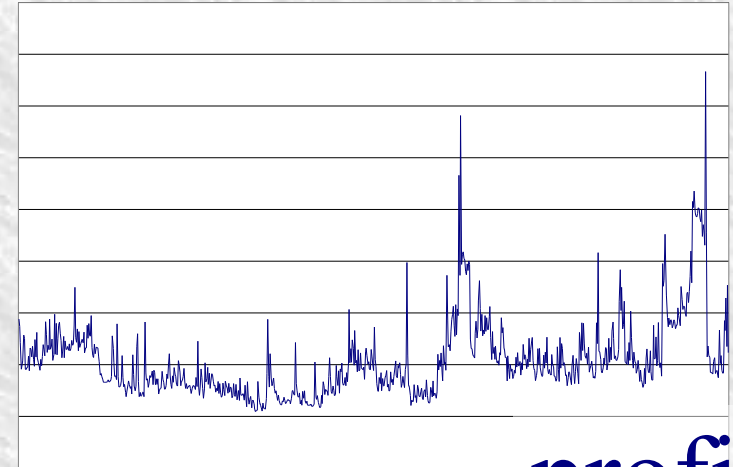
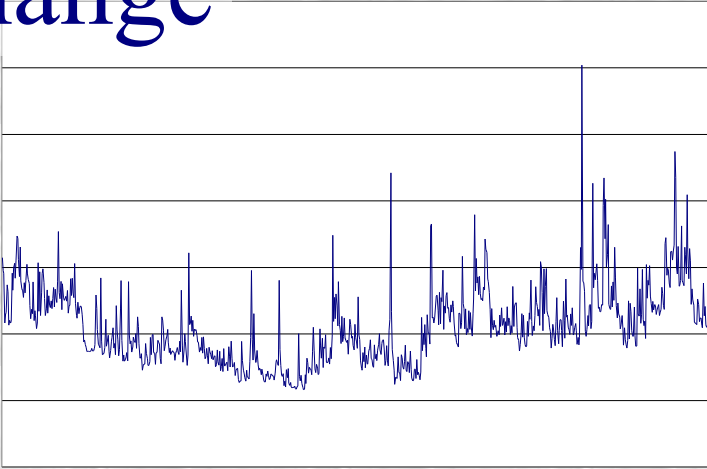


Thematic distance  $\neq$  Ontologic distance (of type *is-a*)  
but  
Thematic distance  $\supset$  Ontologic distance

# Conceptual Vectors

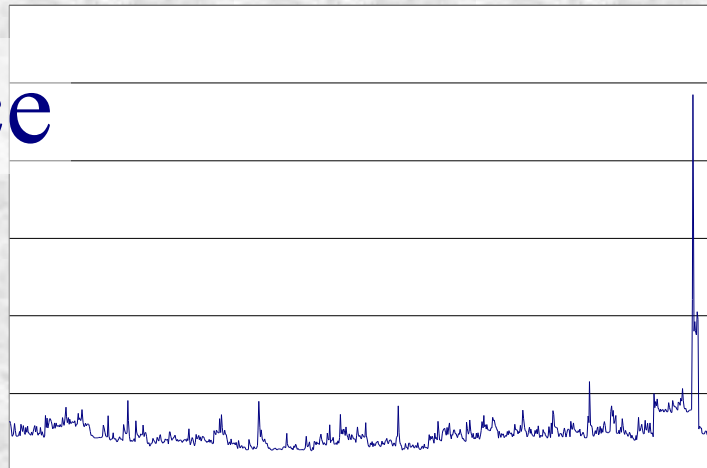
## Examples

exchange



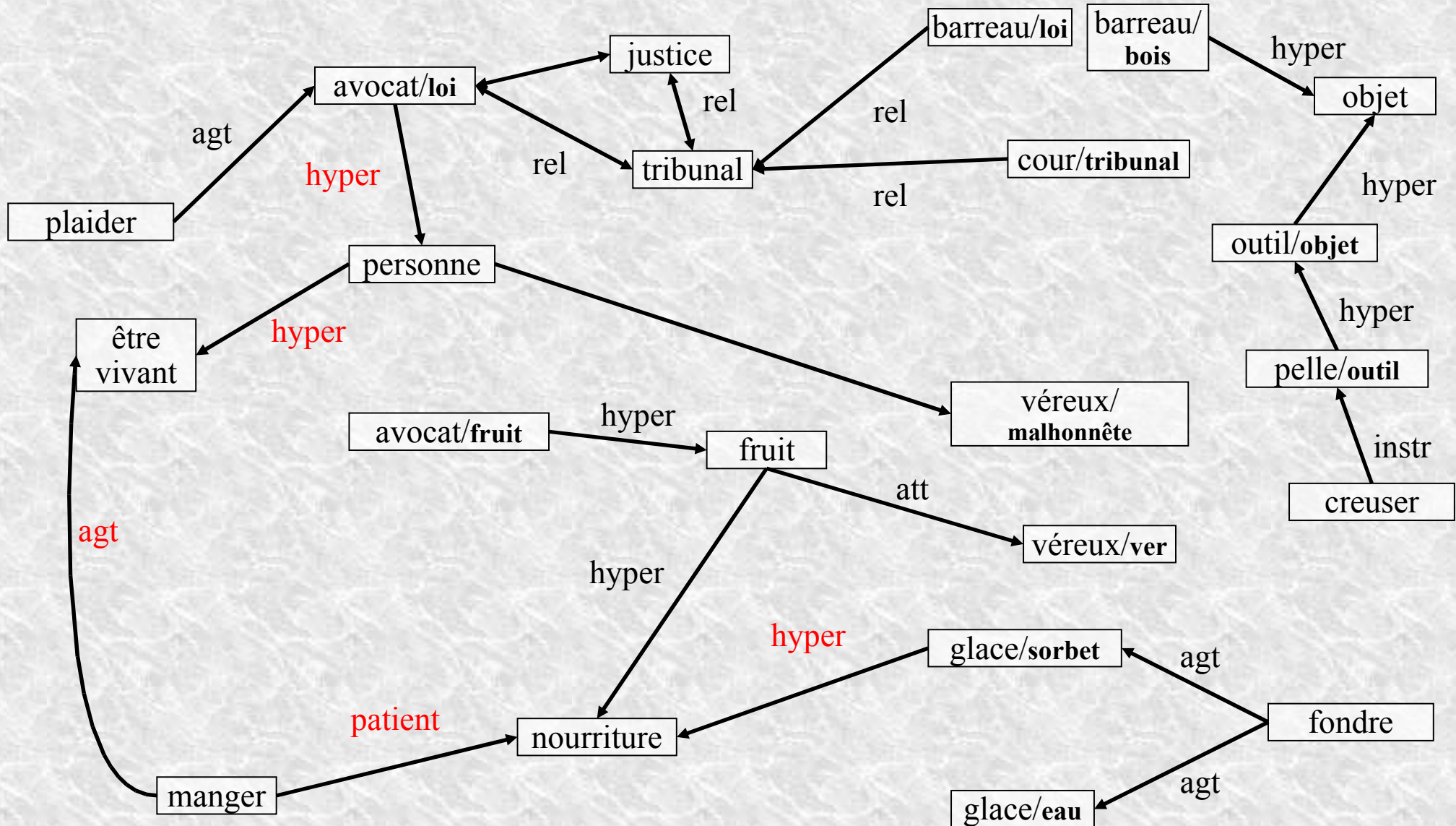
profit

finance



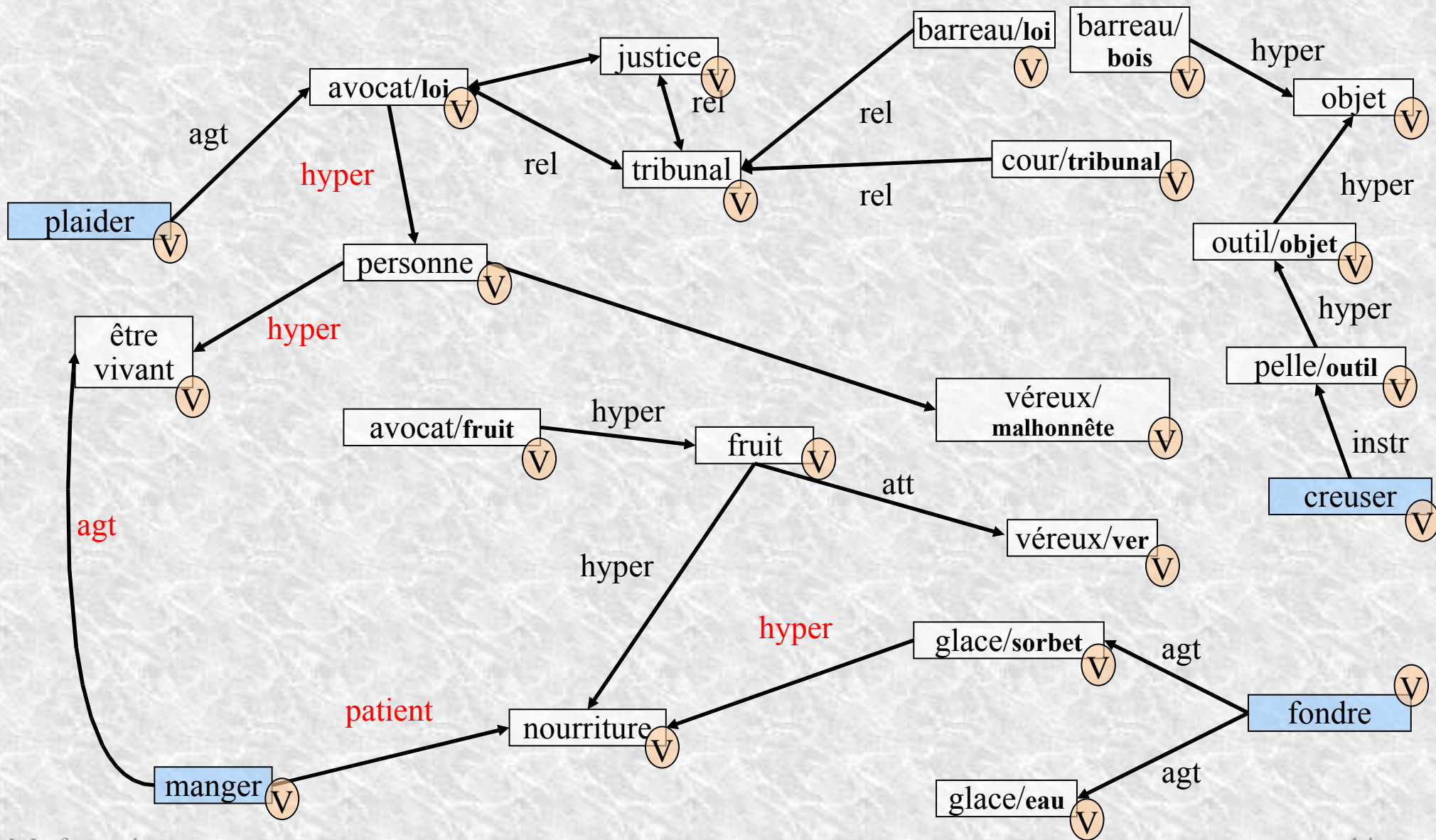
# Casting

## Lexical network



# Casting

## Lexical network with conceptual vectors



### Adding vectors to graph nodes

=> a shortcut for term proximity

graph searching algorithm cost  $> D_A$  function cost

=> vectors can be computed according to points of view

=> can be done incrementally

Graphs tend to be a **precision** oriented data structure

Vectors tend to be a **recall** oriented data structure

**We need both!**



### Constituent tree structure

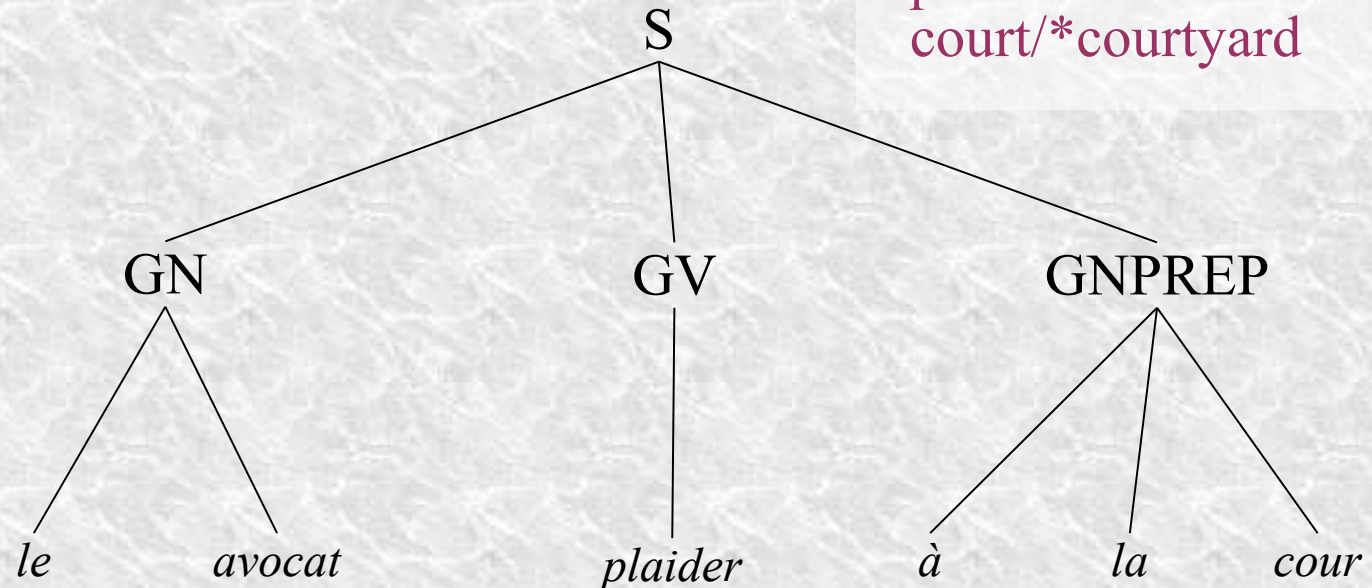
geometry + set of instantiated variables

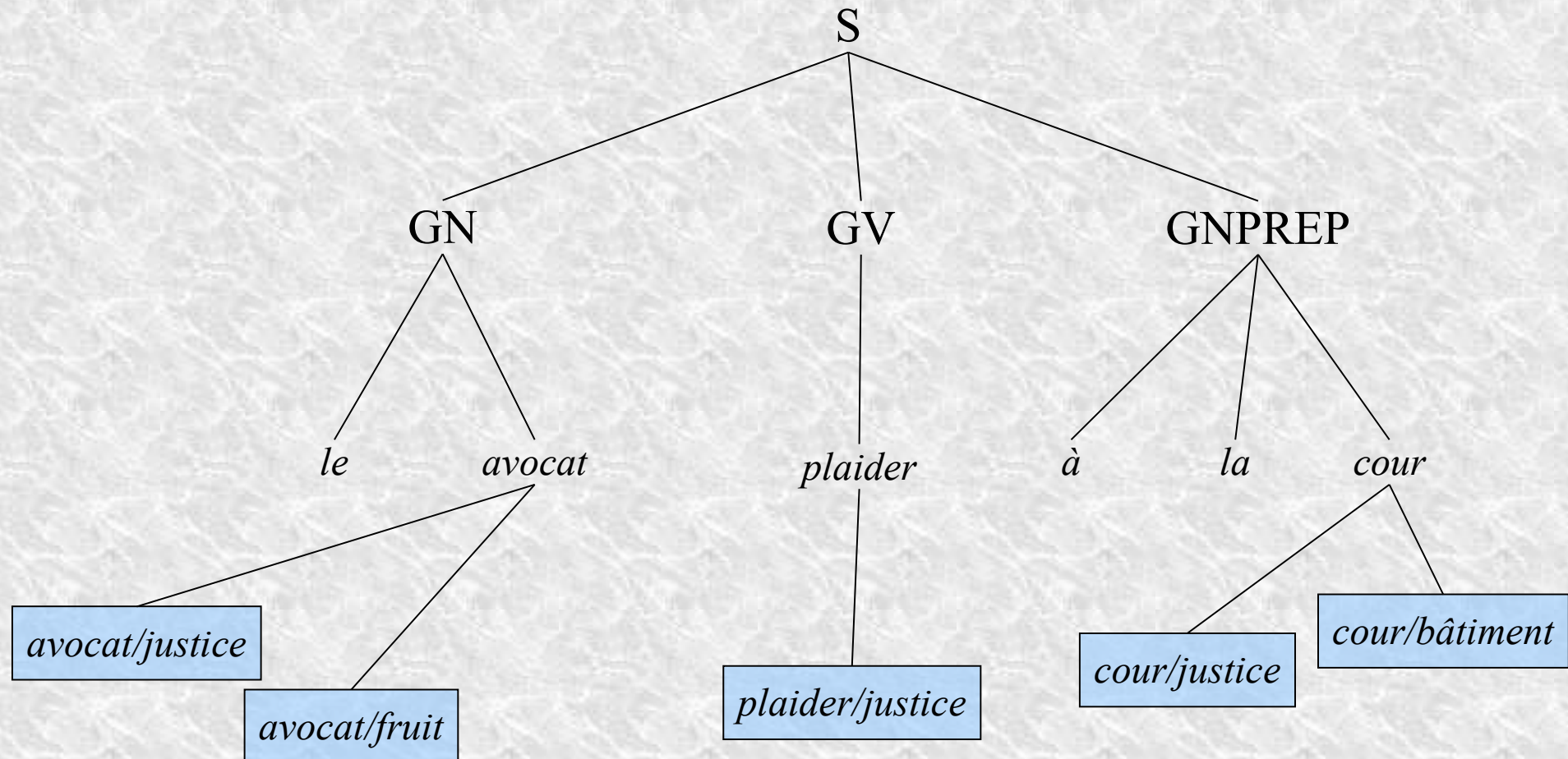
SYGMART application to French (SYGFRAN)

[chauché]

“L’avocat plaide à la cour”

The lawyer/\*avocado  
pleads at the  
court/\*courtyard

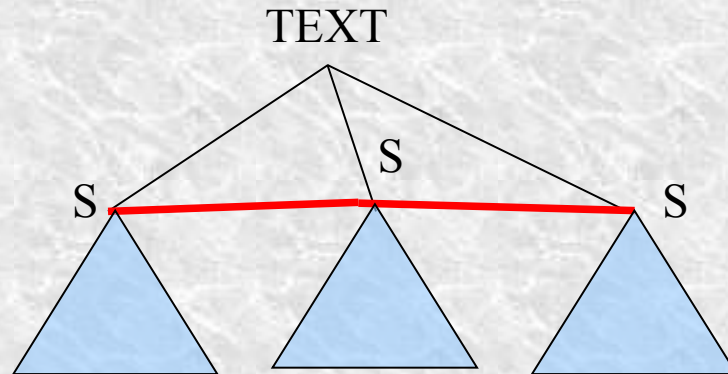




# Morphosyntactic trees Adding links => Graph

## Transforming the tree into a graph

Links between sentence to reconstitute text flow



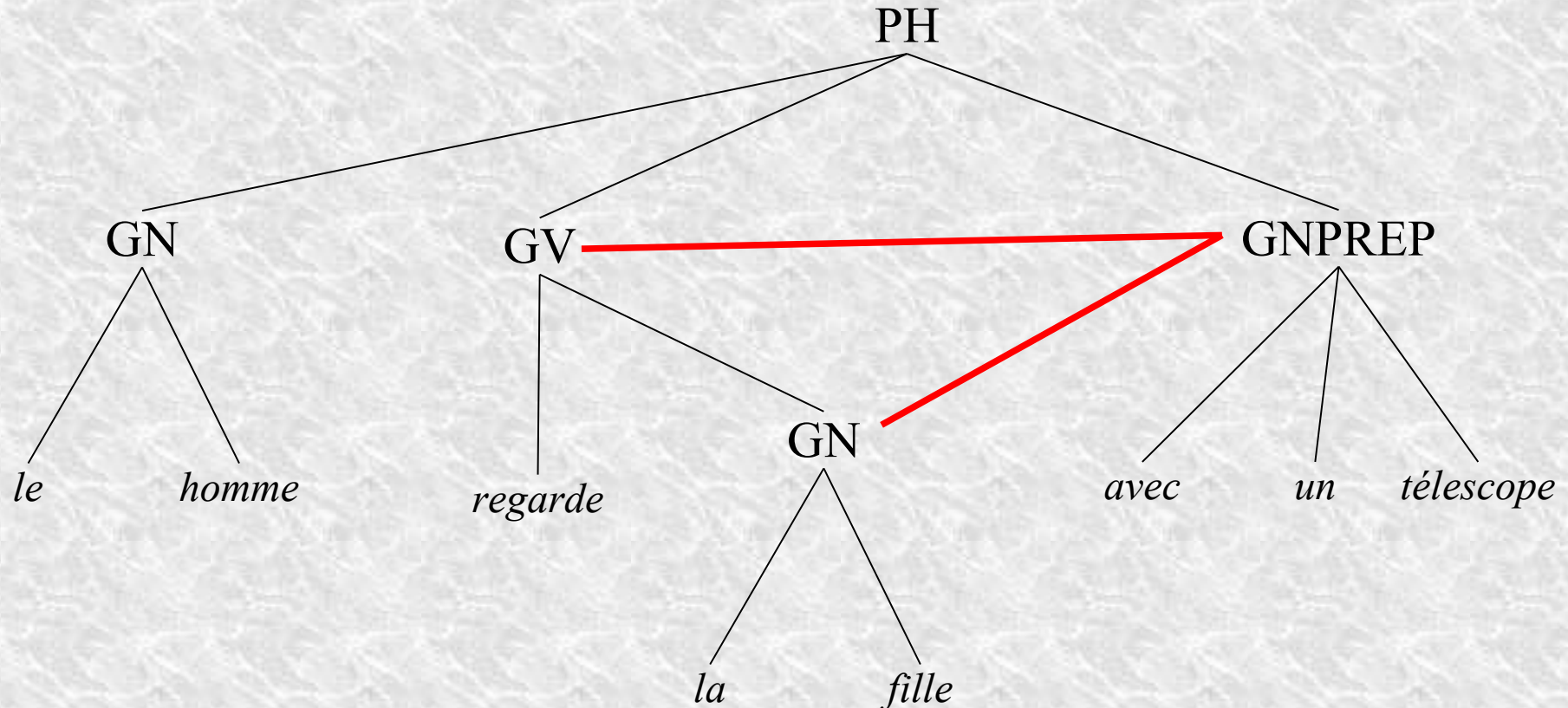
=> allows the computation of distance between sentences

Combination between structural and surface information

# Morphosyntactic trees Adding links => Graph

## Transforming the tree into a graph

Enumeration of syntactically possible attachments (GNPREP)



# Ant algorithms

## Multi-agent system – reactive agents

TSP – operational research - network – bio informatic

[dorigo] [bertelle] [bonabeau, théraulaz] [bruten] [costa, hertz]

## Principle

Stygmergy = indirect communication  
through modification of the environment

Pheromones drops – Slow decay over time  
=> handling of an ever-changing environment

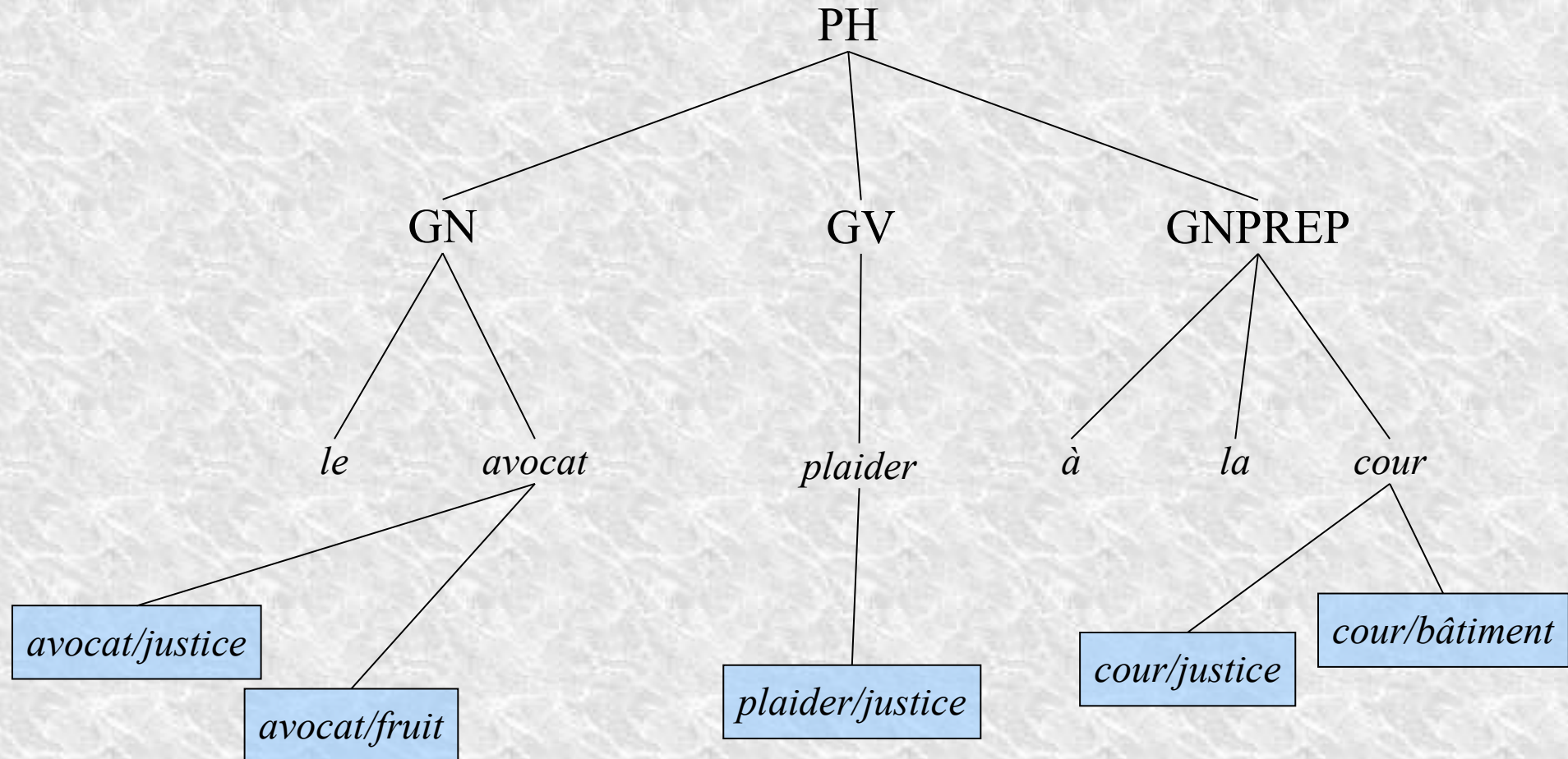
## NLP

Cognitive agents [stéphanini] [sabah]

otherwise reactive [lafourcade, guinand] [+cunningham] [zamora]

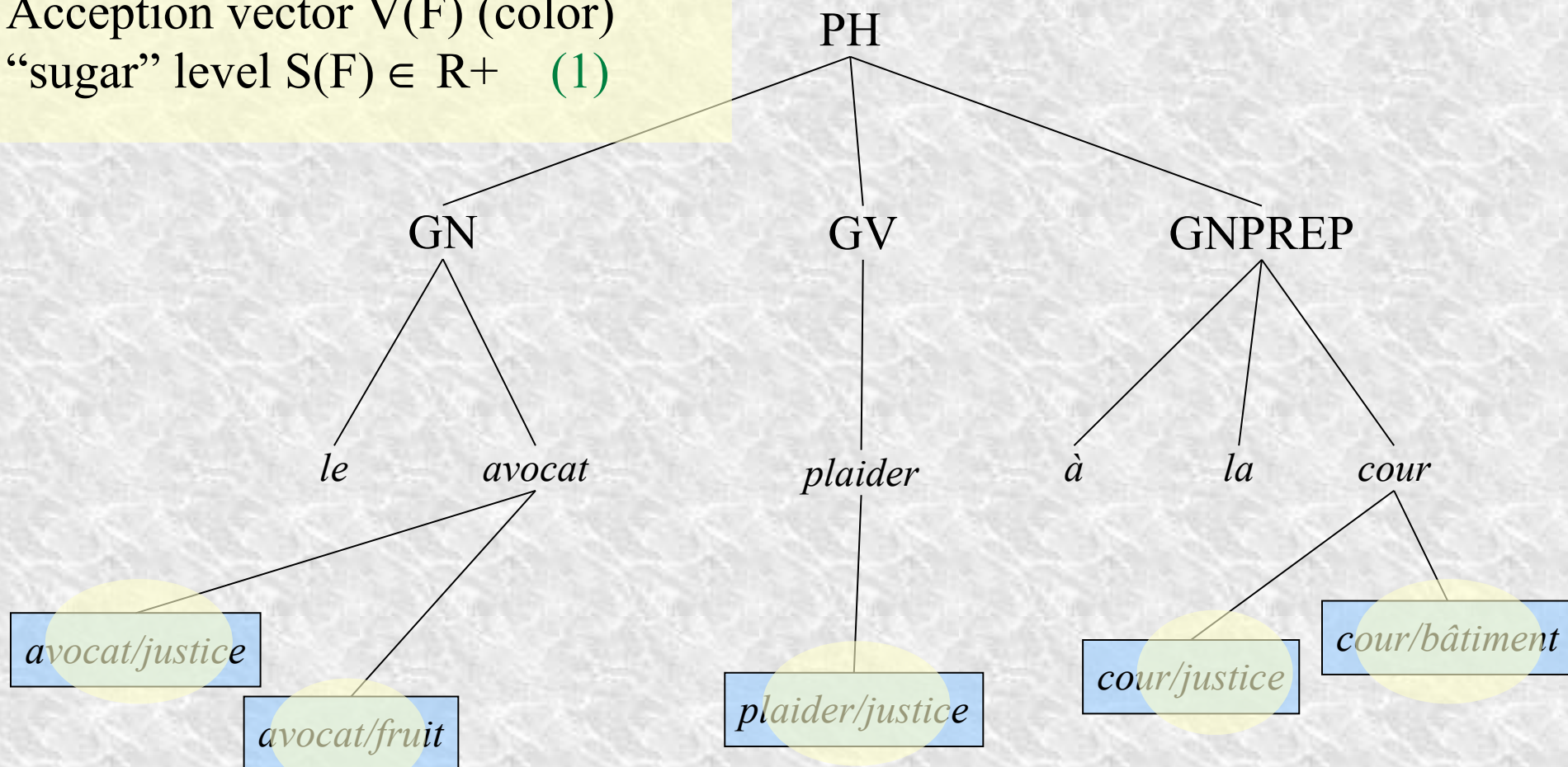
# Outline

- Semantic Analysis      What to look at? – Applications
- Casting                  Conceptual Vectors – Analysis Trees – Lexical Network
- Ant algorithm          Principles - Without lexical network - With lexical network
- Ressource Production      Lexical Network - Conceptual Vectors
- Future directions          Toward holistic analysis: syntax + semantic



### Anthill / Acception nodes

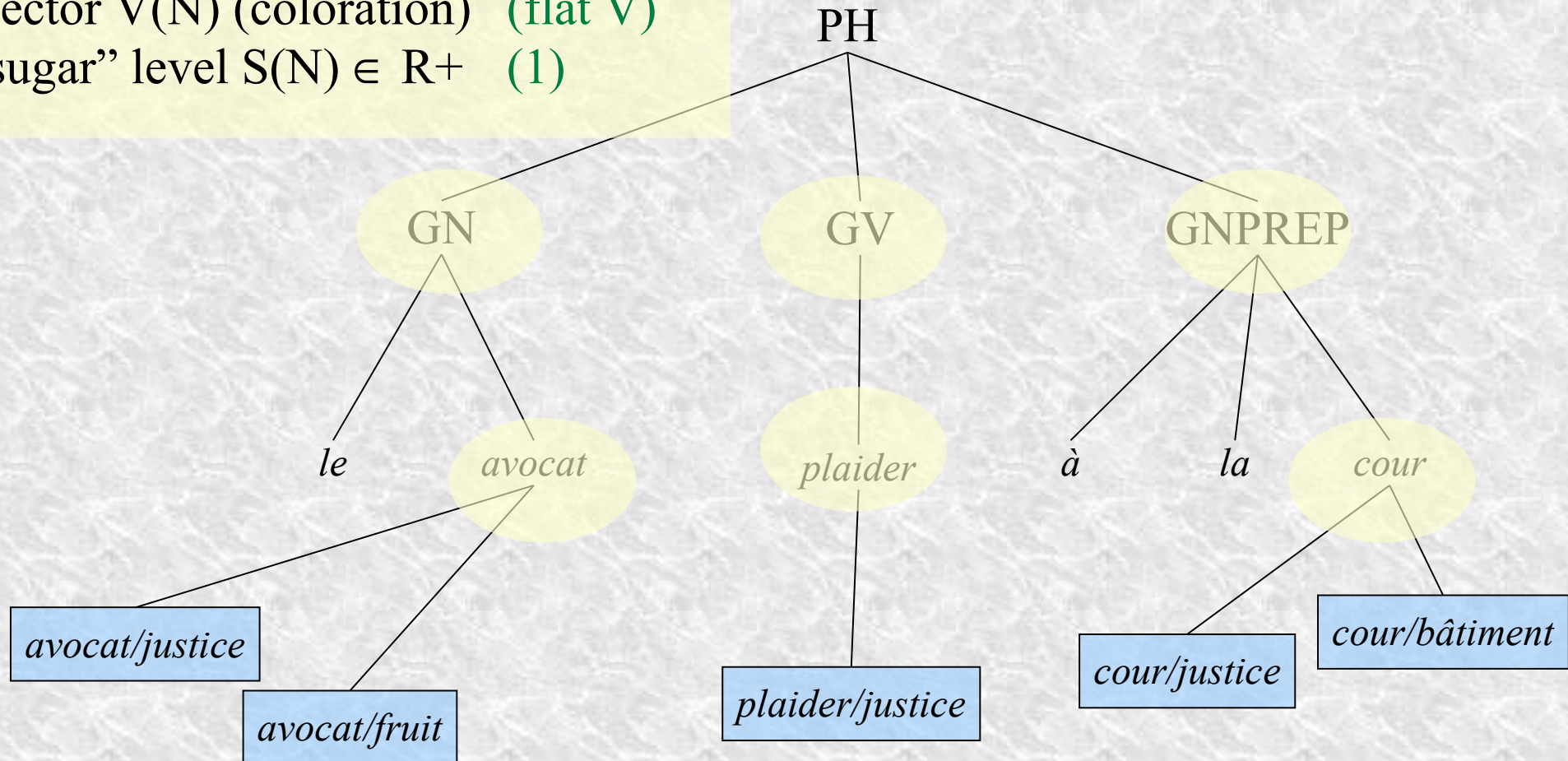
Acception vector  $V(F)$  (color)  
“sugar” level  $S(F) \in \mathbb{R}^+$  (1)





### Standard node

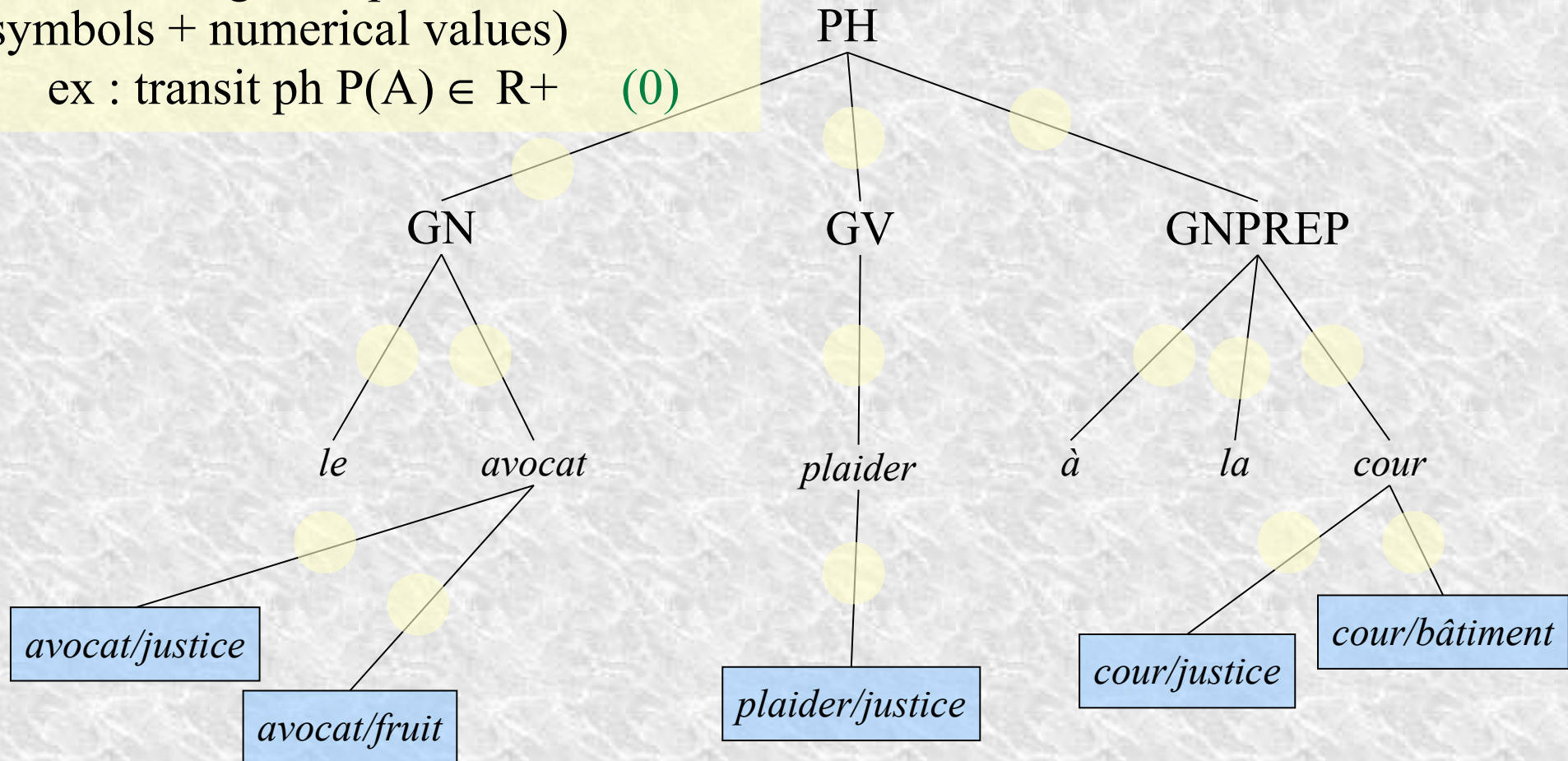
Vector  $V(N)$  (coloration) (flat  $V$ )  
“sugar” level  $S(N) \in \mathbb{R}^+$  (1)



### Arc A

- type
- Valuated signal = pheromones (symbols + numerical values)

ex : transit ph  $P(A) \in \mathbb{R}^+$  (0)



# Ant algorithms

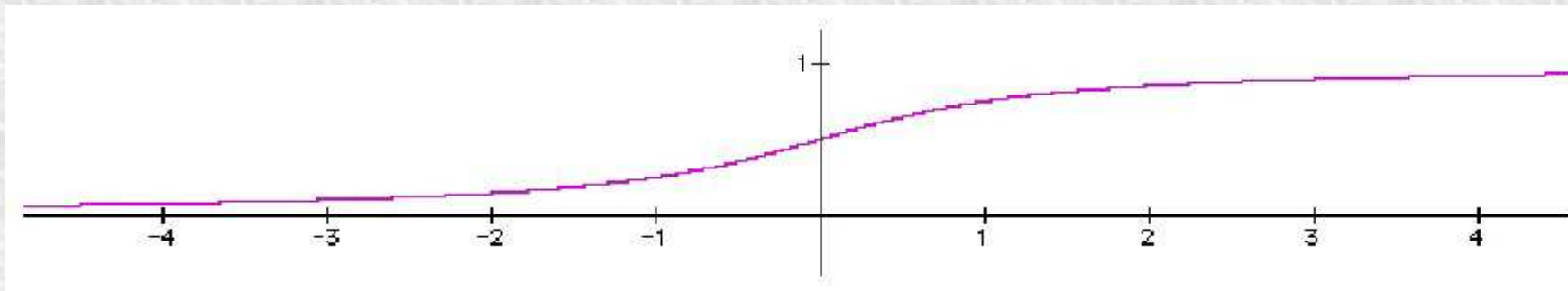
## Environment

### Ant production

one or several ants at each cycle

with a probability equal to the sugar level  $S(F)$

“sigmoïde” function



Production cost (experiment with 1/10)

### Attributes of an ANT

« sugar » load

Q real in [0,1]

sugar dropped at death

Life span

20 cycles

Reference to its home F

$V(f) = V(F)$

Current moving strategy

Prob = Q

### Two moving strategies

Looking for « sugar » - **search mode**

Bringing the « sugar » back home – **return mode**

# Ant algorithms

## Ant moves

### Pseudo-random moves

Evaluation of possible destinations + pseudo taboo list ( $k = 1$ )

### Environment modification

Drop of pheromones of the arc

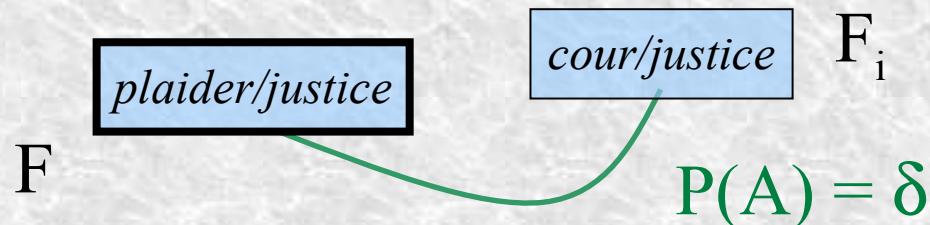
$$P(A) = P(A) + \delta$$

Modification of the vector of the reached node

$$V(N) = V(N) + \alpha V(f)$$

normed sum with  $\alpha$  and  $\delta$  small

### Creation of a bridge towards F



### General principle

Current position = a node  $N$  with  $k$  neighbors  $N_i$  through arcs  $A_i$

Evaluation function for each  $N_i = Eval(N_i)$

Evaluation function for each  $A_i = Eval(A_i)$

Global evaluation function of destination  $dest_i$

$$Eval(dest_i) = \max \begin{cases} Eval(N_i) + Eval(A_i) \\ \epsilon \end{cases}$$

$$Prob(dest_i) = \frac{Eval(dest_i)}{\sum_{k=1}^n Eval(dest_k)}$$

# Ant algorithms

## Sugar foraging and sharing

### Search mode

Specific action: taking sugar from the node

$$Eval(N_i) = S(N_i)$$

$$Eval(A_i) = 1 - P(A_i)$$

### Return mode

Specific action: dropping sugar on the node if its looks like home

$$Eval(N_i) = 1 - \frac{2}{\pi} D_A(V(N_i), V(f))$$

$$Eval(A_i) = P(A_i)$$

At each simulation cycle, pheromone signals decay slightly

$$P(A) = \text{Min} (P(A) - d * \eta, 0)$$

$$d = \text{dist}(N_a, N_b) \quad \text{and} \quad \eta \text{ a decay factor}$$

Arc of the tree between  $N_a$  and  $N_b$

$$P(A) = \text{Min} (P(A) - \eta, 0) \quad \text{because} \quad \text{dist}(N_a, N_b) = 1$$

Bridge between  $N_a$  and  $N_b$

$$\text{Deleted if } P(A) = 0$$

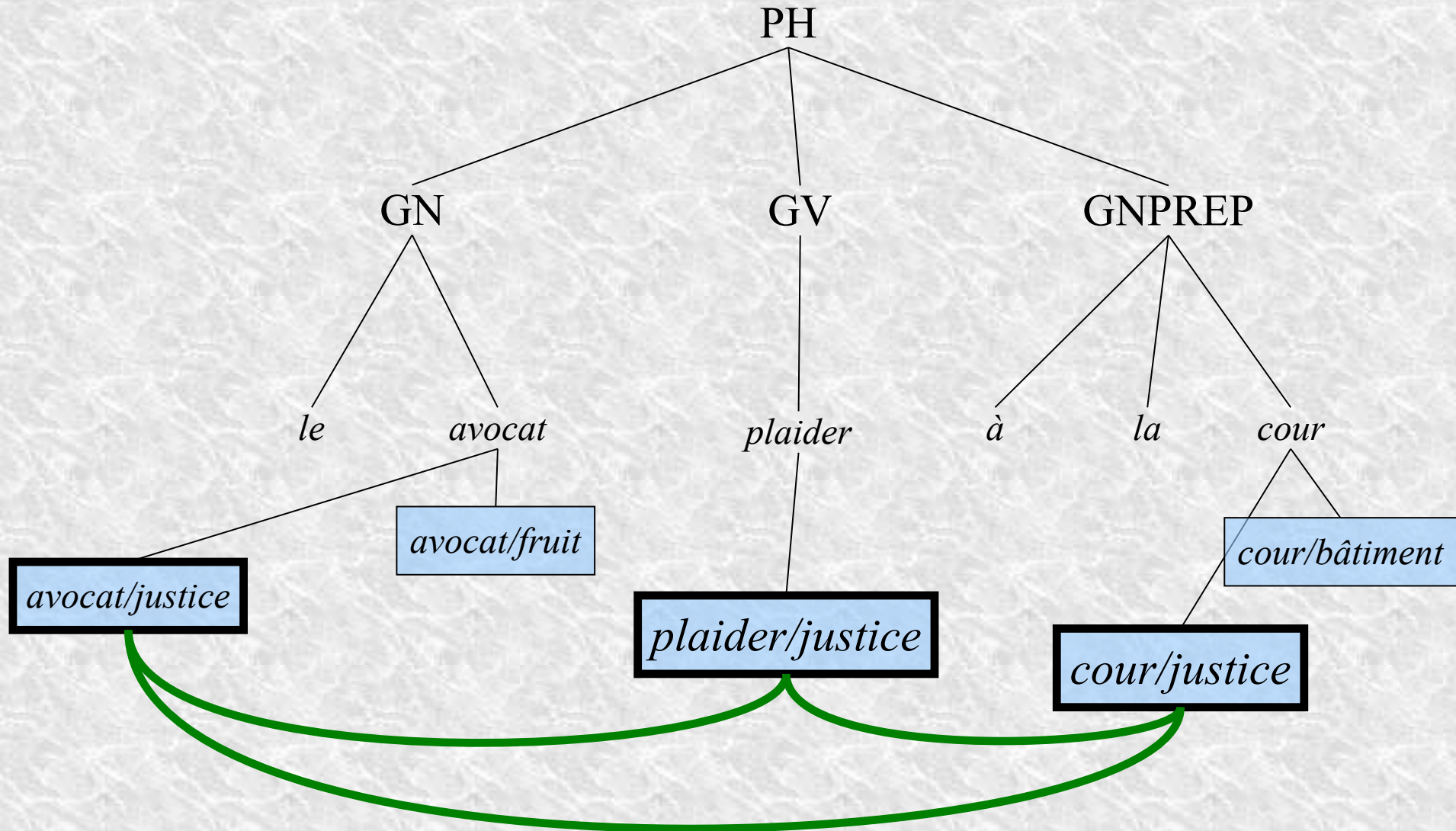
The longer the bridge, the harder it is to keep



# Ant algorithms

## Simple example

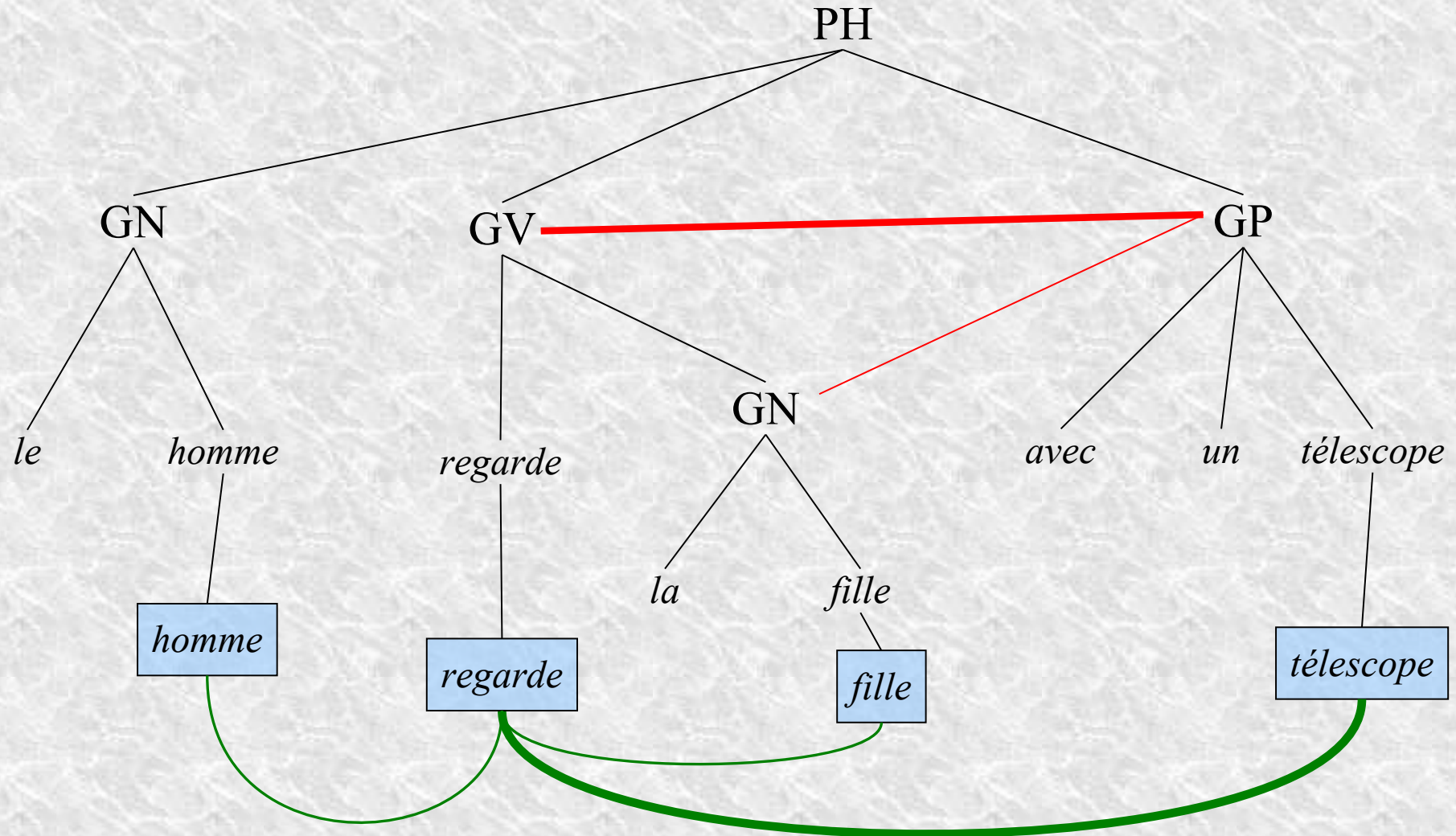
The lawyer/\*avocado pleads at the court/\*courtyard



# Ant algorithms

## Simple example for attachment

The man sees (the girl with a telescope)  
/ The man (sees [the girl] with a telescope)



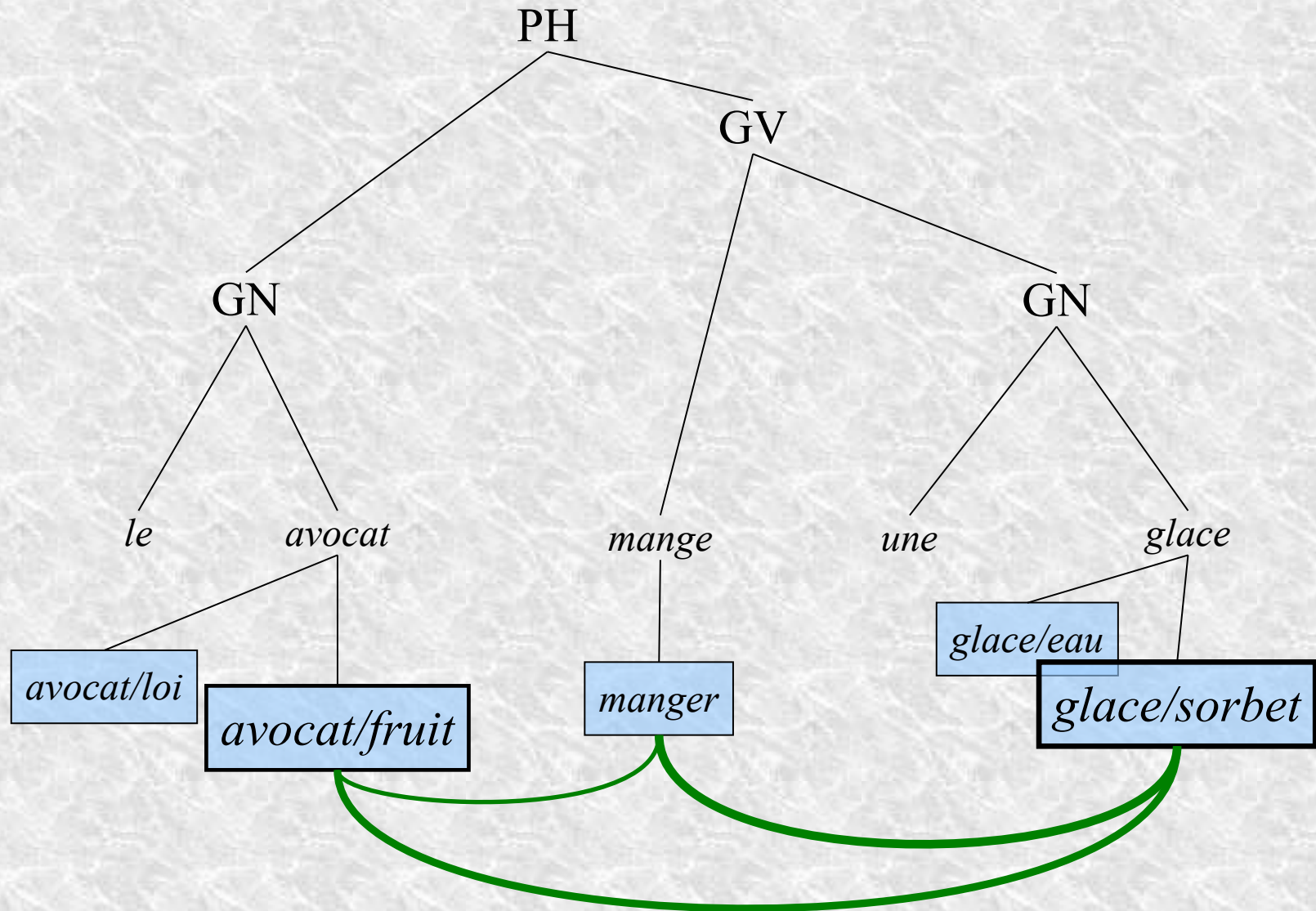
Thematic information is enough for WSD  
in 73% of the cases

Scoring scheme	All terms	Nouns	Adjectives	Verbs	Adverbs
Fine Grain Scoring	0.73	0.76	0.78	0.61	0.85
Coarse Grain Scoring	0.85	0.88	0.9	0.72	0.95
Base Line with POS	0.33	0.30	0.4	0.17	0.60
Base Line no POS	0.25	0.24	0.25	0.12	0.52

Room for improvement

# Ant algorithms

\* Simple example



### Ant castes

Ants with some constraints on the evaluation functions

### Castes exploiting the lexical network

Predicate  $\Rightarrow$  agent

Prédicate  $\Rightarrow$  patient

Prédicate  $\Rightarrow$  instrument ...

Local copy            Acceptions and relations of the networks

Local fusion            of nodes that seems to be identical

# Ant algorithms caste Predicate => Patient

Ant caste created by a verb (predicate)

Search mode

$$Eval(N_i) = S(N_i)$$

$$Eval(A_i) = 1 - P(A_i) + \max \begin{cases} 2 & \text{si } type(A_i) = patient \\ 0 & \text{sinon} \end{cases}$$

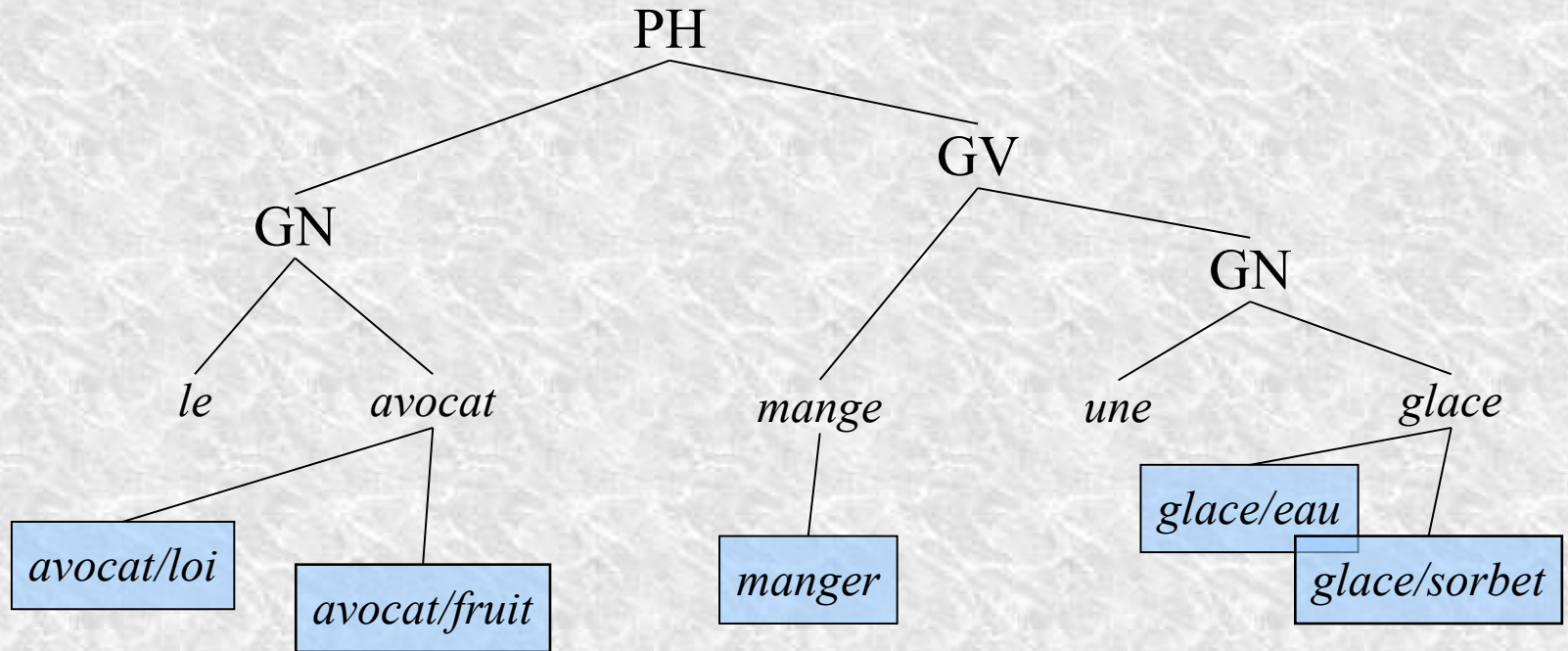
Return mode

$$Eval(N_i) = 1 - \frac{2}{\pi} D_A(V(N_i), V(f))$$

$$Eval(A_i) = P(A_i) + \max \begin{cases} 2 & \text{si } type(A_i) = patient \\ 0 & \text{sinon} \end{cases}$$

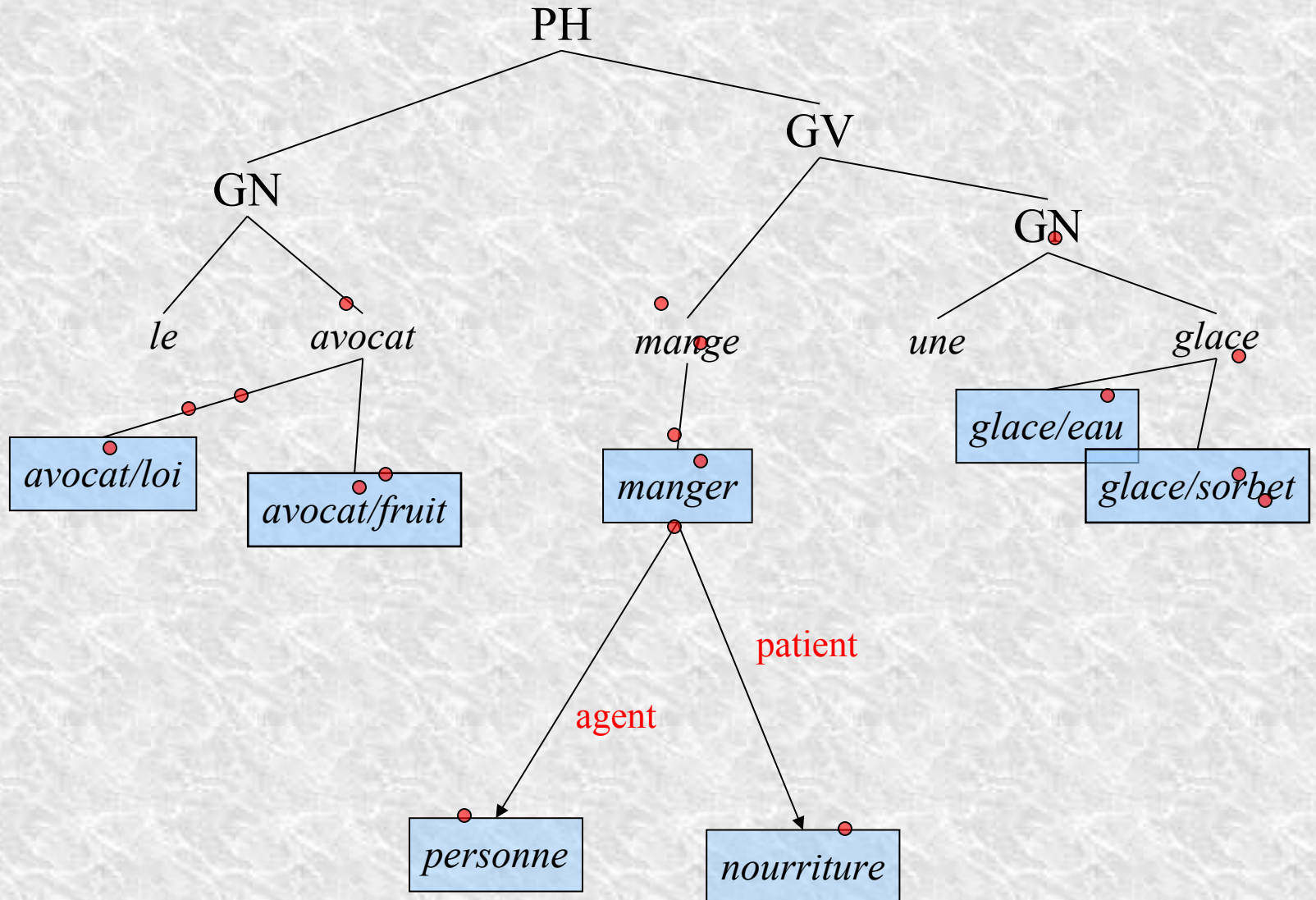
# Ant algorithms

## Example with lexical network



# Ant algorithms

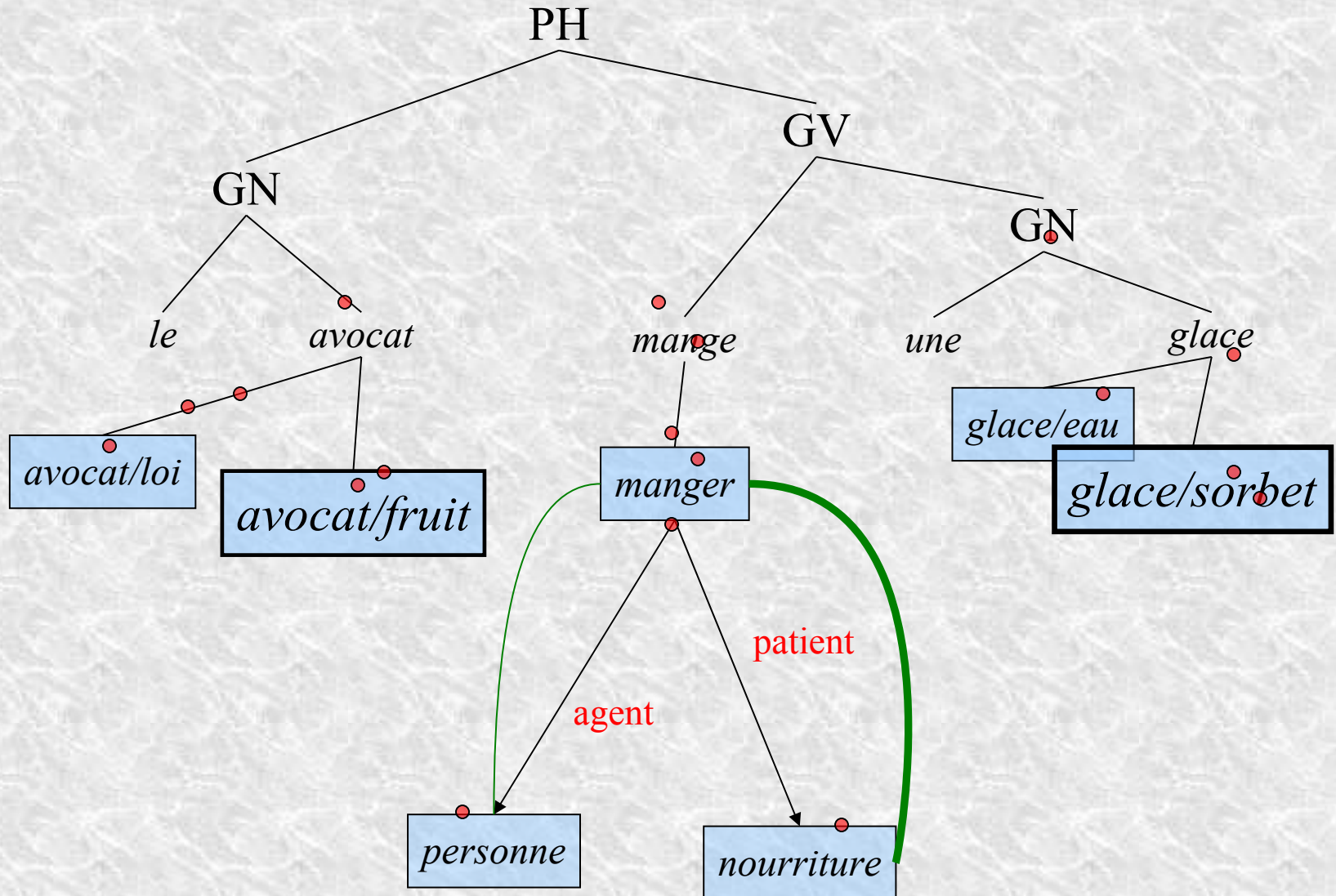
## Example with lexical network





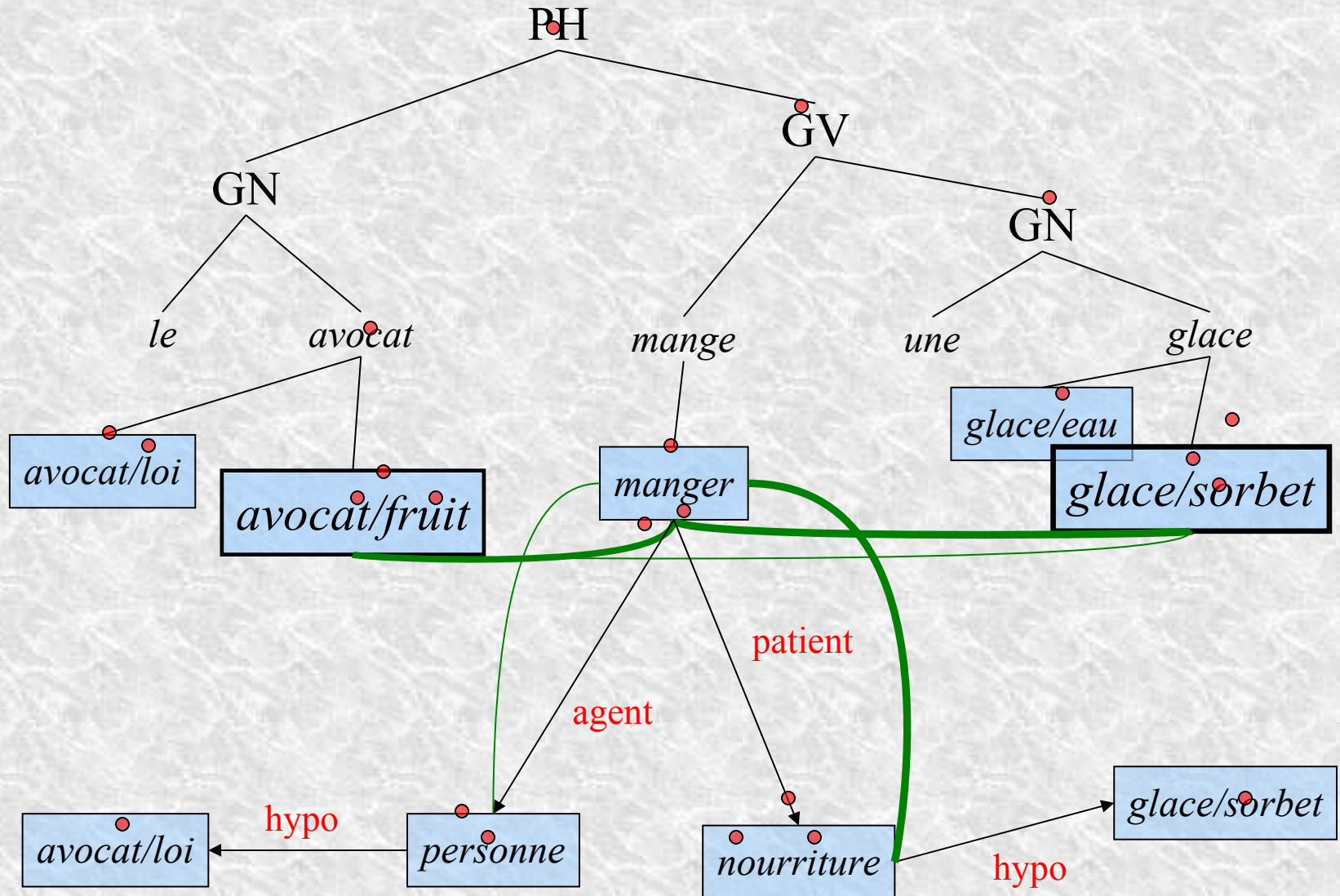
# Ant algorithms

## Example with lexical network



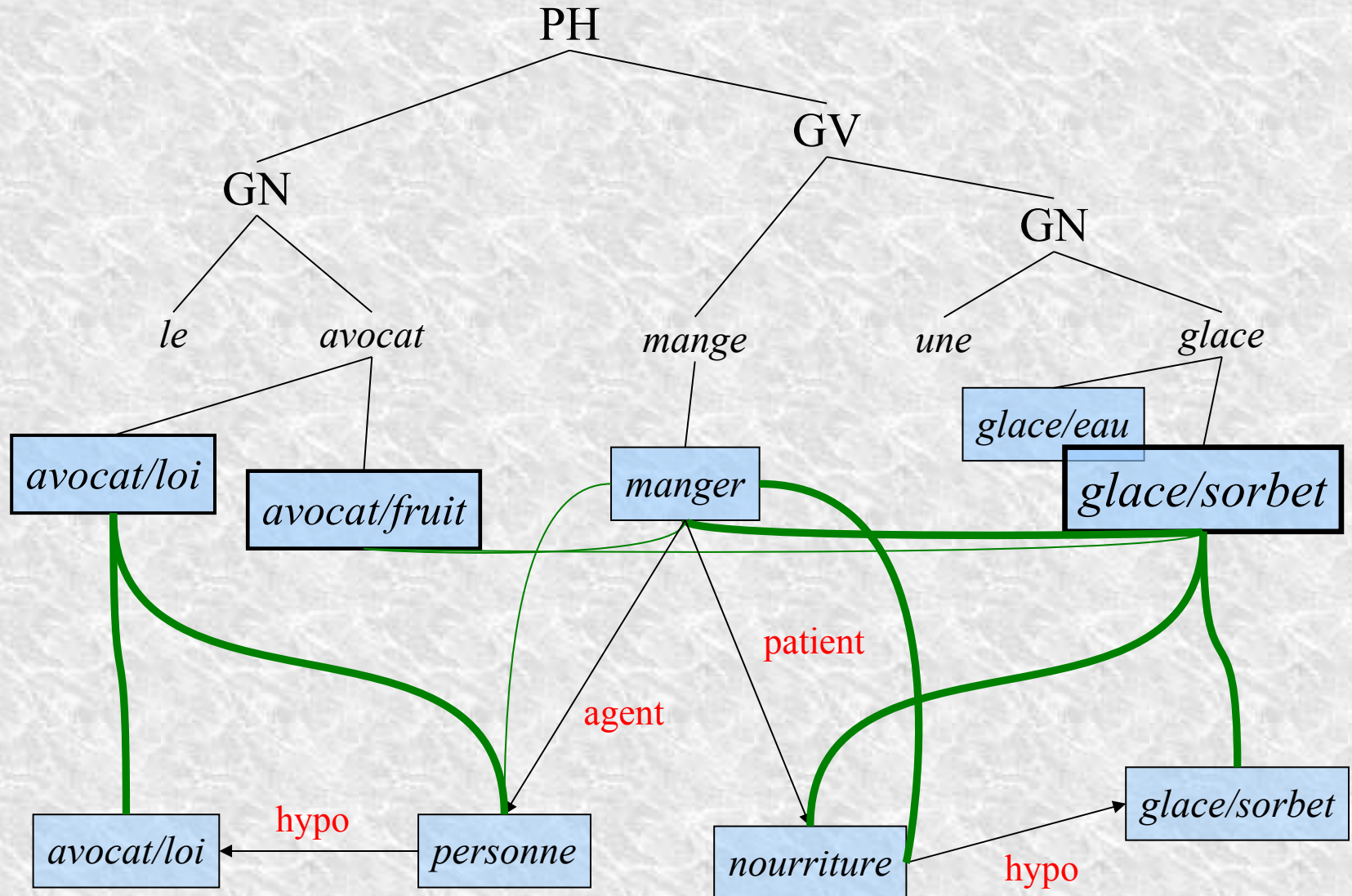
# Ant algorithms

## Example with lexical network



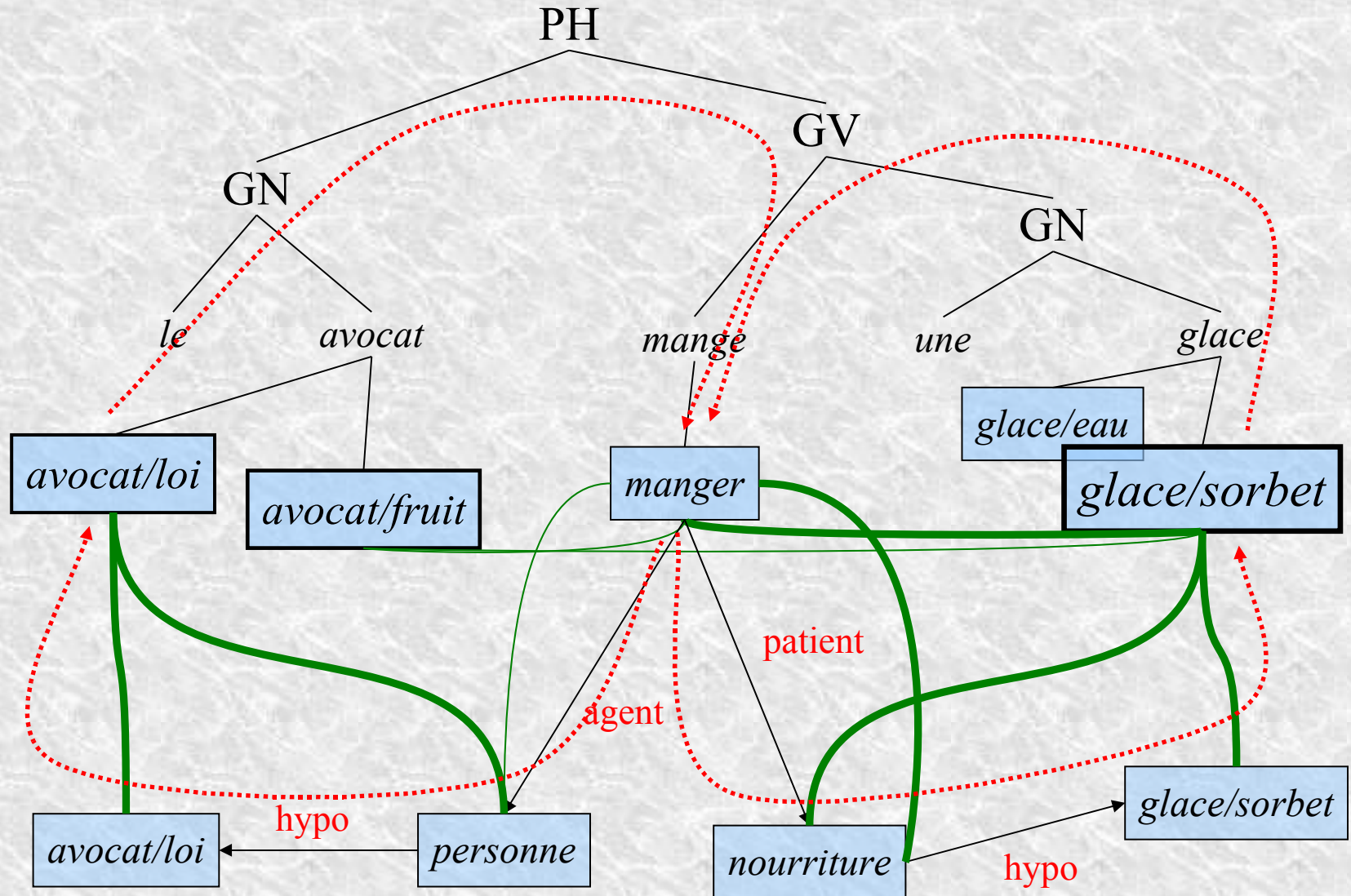
# Ant algorithms

## Example with lexical network



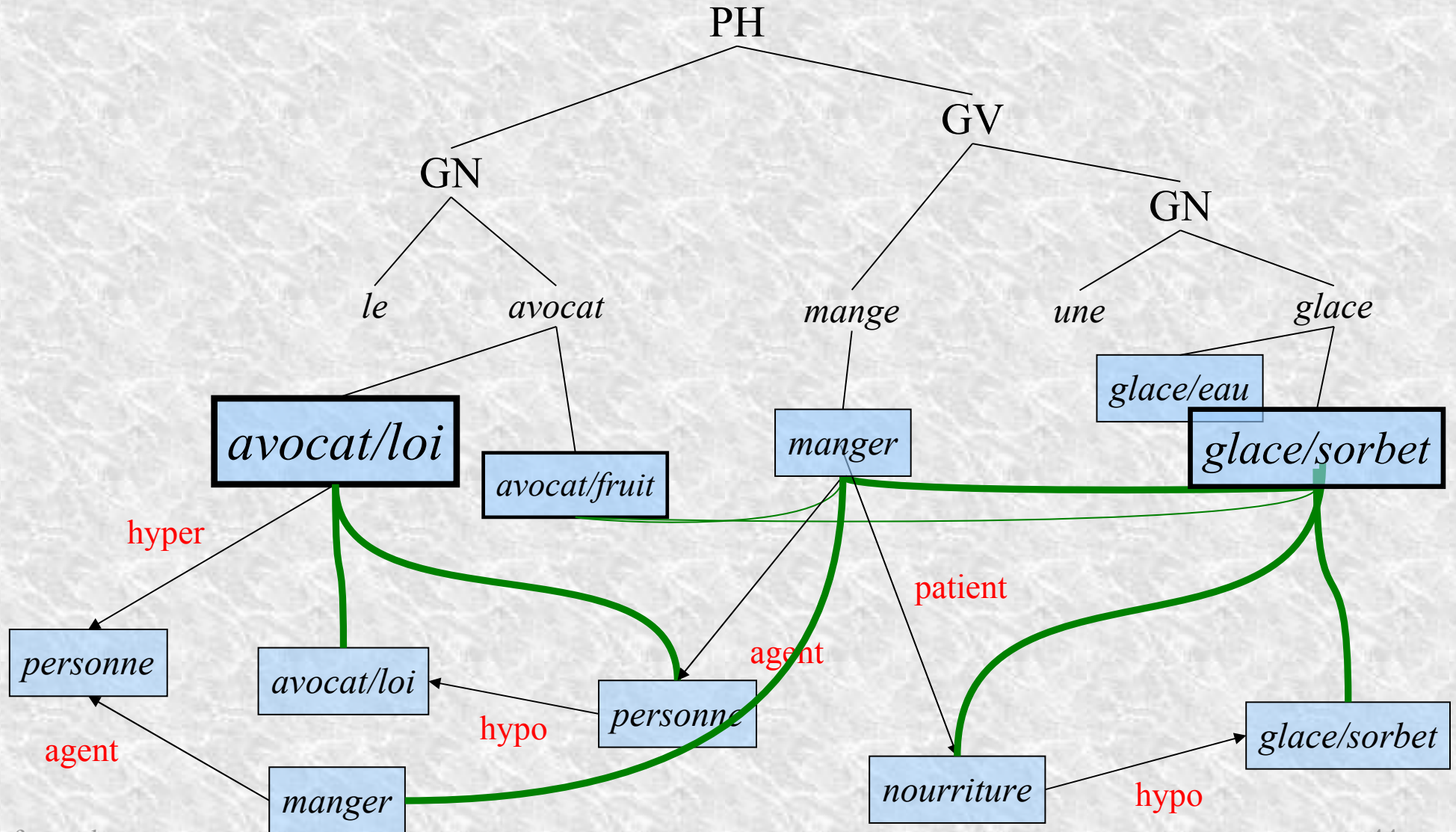
# Ant algorithms

## Example with lexical network



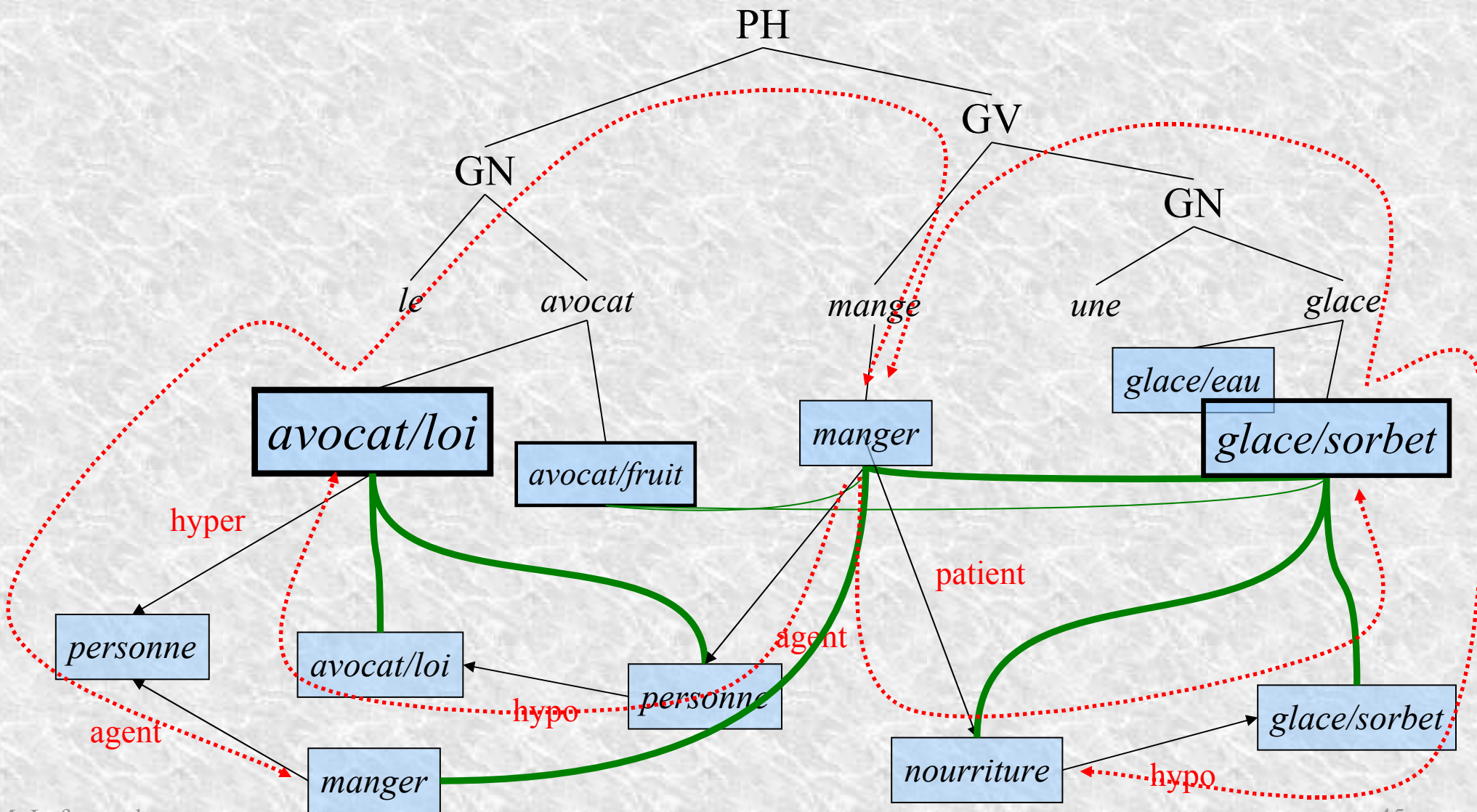
# Ant algorithms

## Example with lexical network



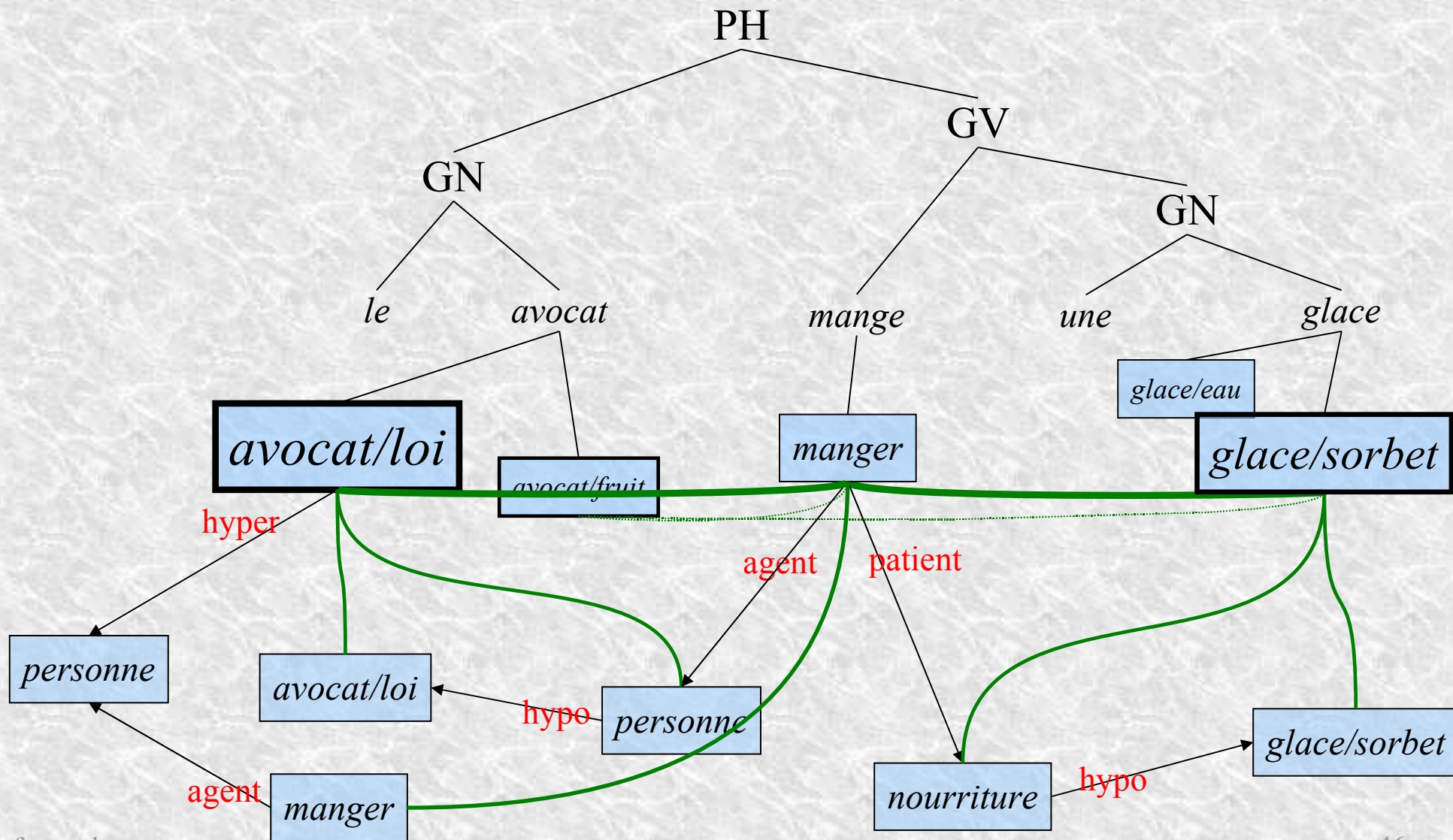
# Ant algorithms

## Example with lexical network



# Ant algorithms

## Example with lexical network

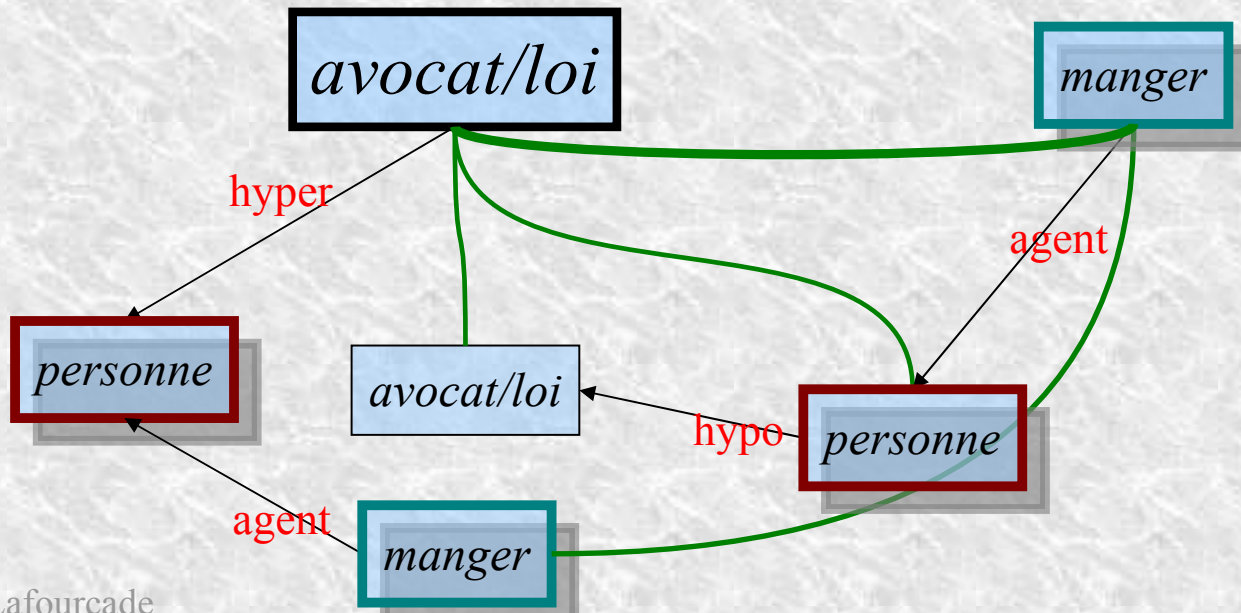


### Node fusion between identical nodes

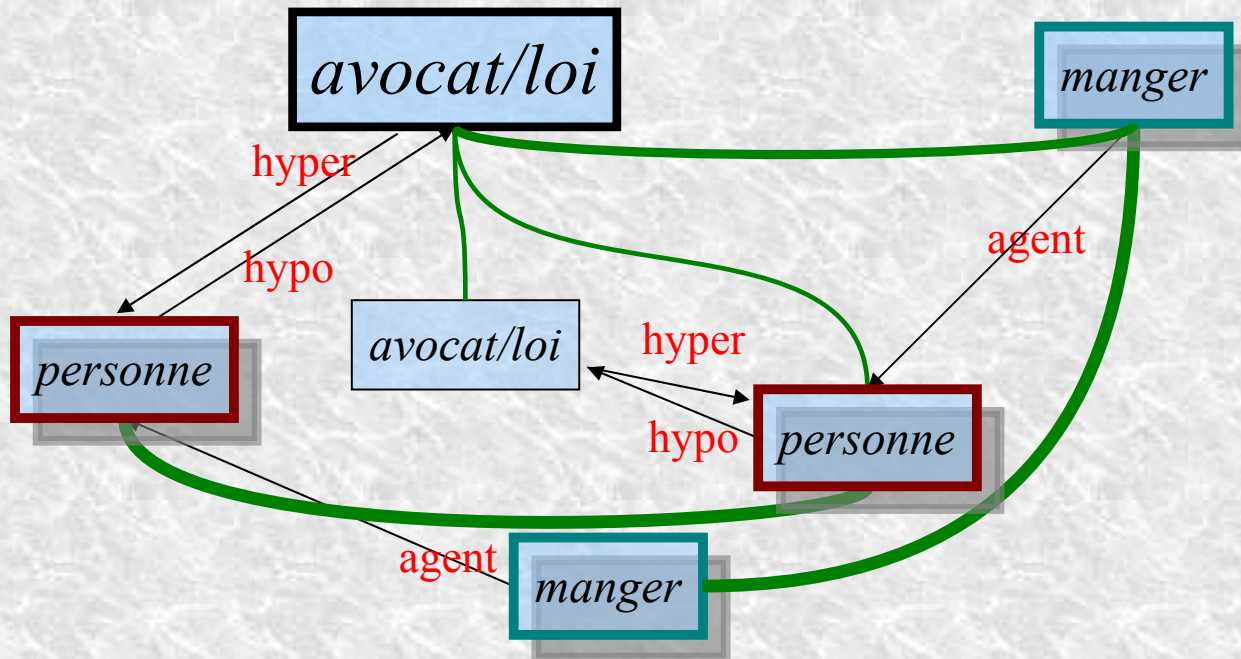
same label (simplified view here)

... if they are not too far

... if they are similar regarding their relations

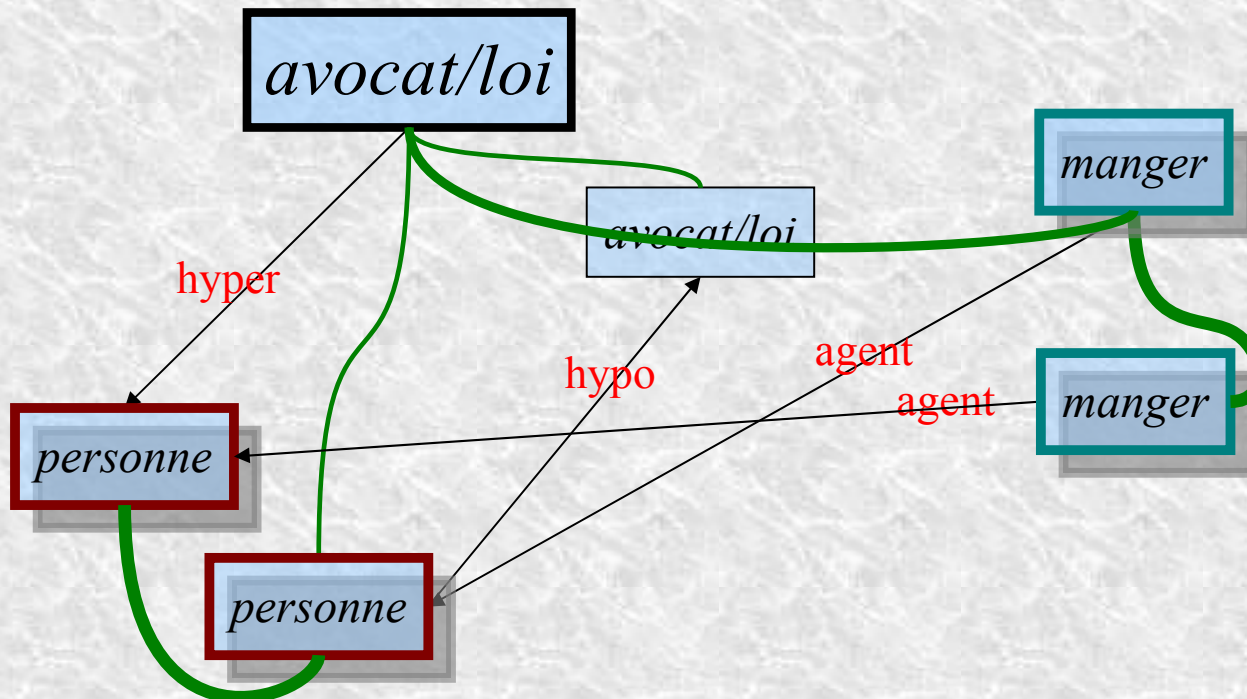


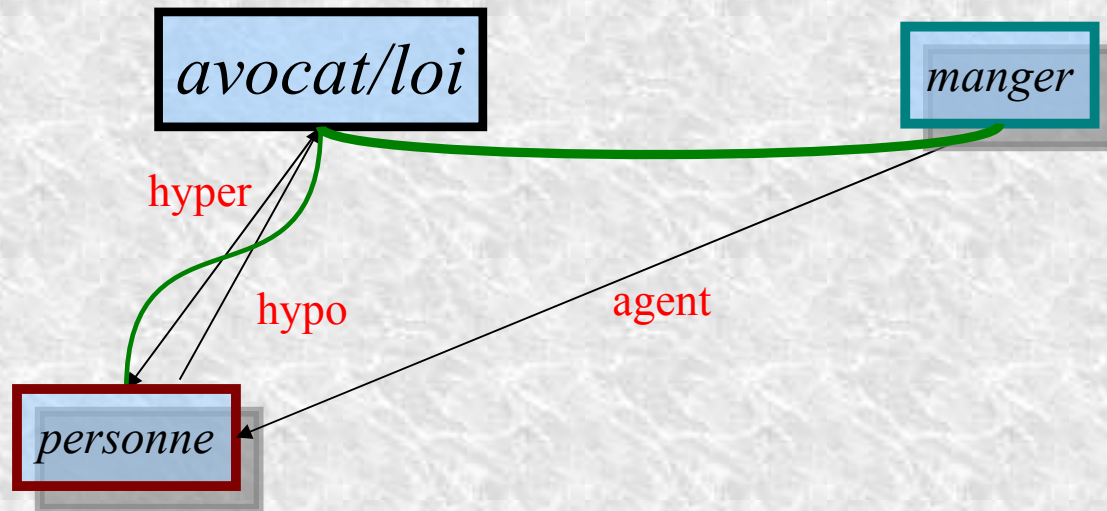




# Lexical Network

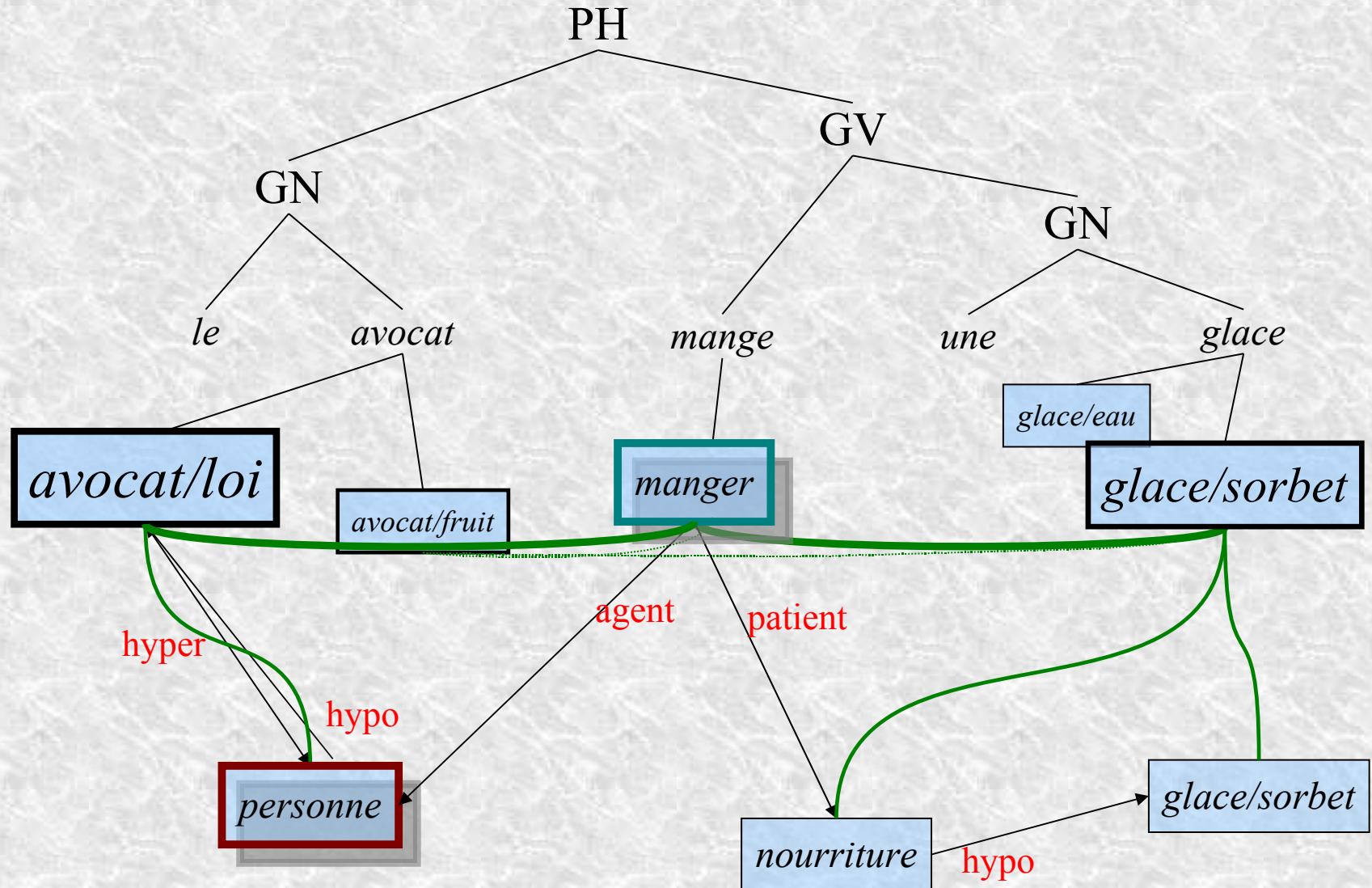
## Node fusion





# Ant algorithm

Finally...



# Adding a lexical network with

hyponym/hyperonym  
part-of/whole-of  
agent/patient/instrument  
typical location

# leads to proper WSD in 87% of the cases

Still room for improvement

# Outline

- Semantic Analysis      What to look at? – Applications
- Casting                  Conceptual Vectors – Analysis Trees – Lexical Network
- Ant algorithm          Principles - Without lexical network - With lexical network
- Ressource Production      Lexical Network - Conceptual Vectors
- Future directions          Toward holistic analysis: syntax + semantic

### Hand made

WordNet – EurowordNet – HowNet

=> precise but not much recall

=> normative

### Automatic Extraction

... (many)

=> many errors mainly due to polysemy  
and bad relation identification

There is room for improvement



<http://www.lirmm.fr/jeuxdemots>

There is room for education



# Conceptual Vectors Construction

With a given set of concepts (Thesaurus Larousse – Roget – ... )  
=> problem of lexical density

From human usage dictionary, thesaurii, ...

=> problem of WSD and word relation identification

=> problem of saliency

Give up set of concepts => ask only for the set size  
Use more explicit resources => lexical network

# Conceptual Vectors Construction

Emergence - infinite iteration through the lexical network nodes

## Step 1 – Agglomeration

Spring attraction model

$V(N) =$  Weighted means of neighbour vectors  $v_i$

If  $v_i = 0$  then  $v_i =$  random

## Step 2 – Separation

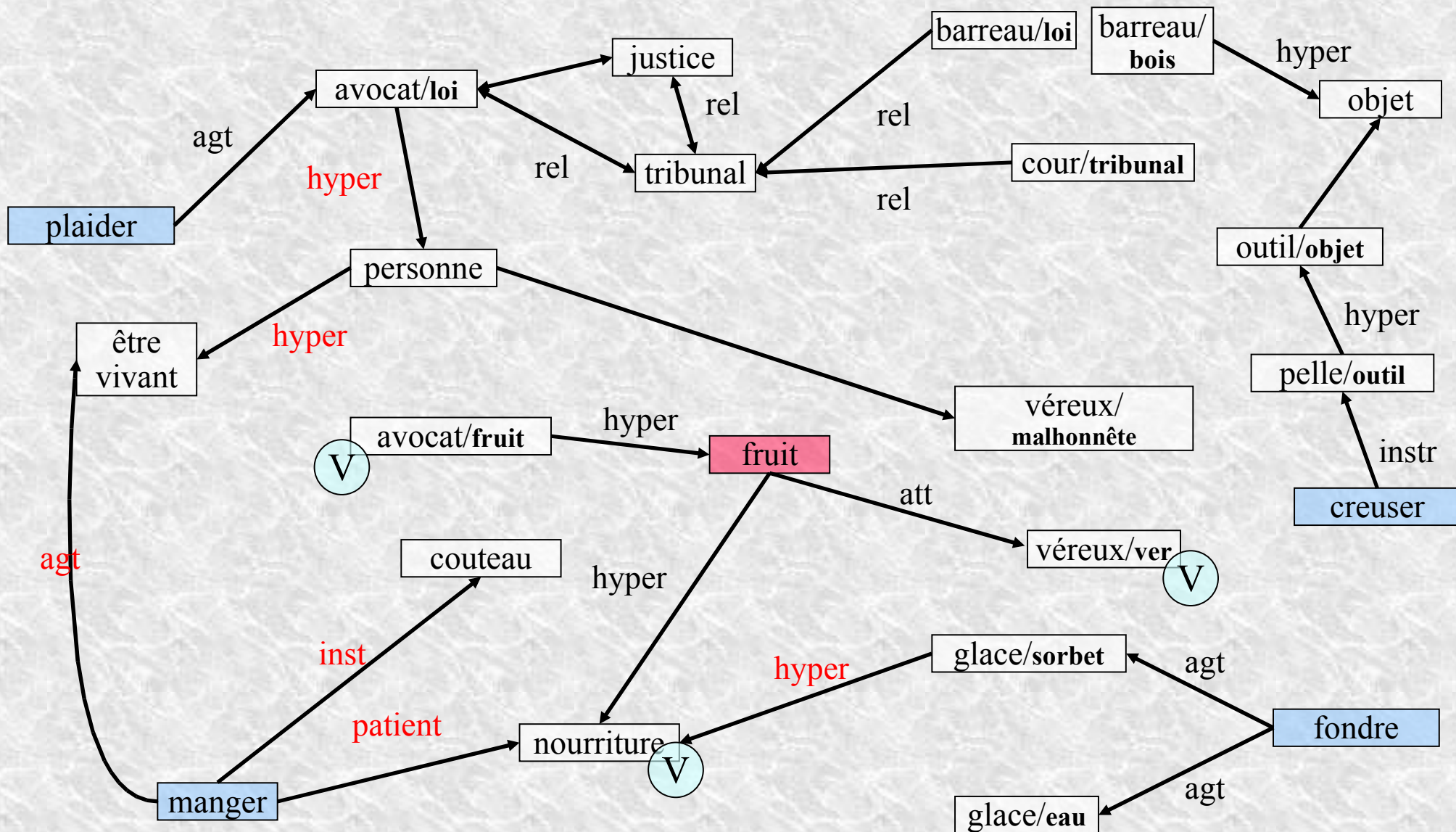
Repulsive particular model

Select  $k$  closest neighbours according to  $D_A$

Compute and apply a repulsive vector with a  $1/d^2$  rule

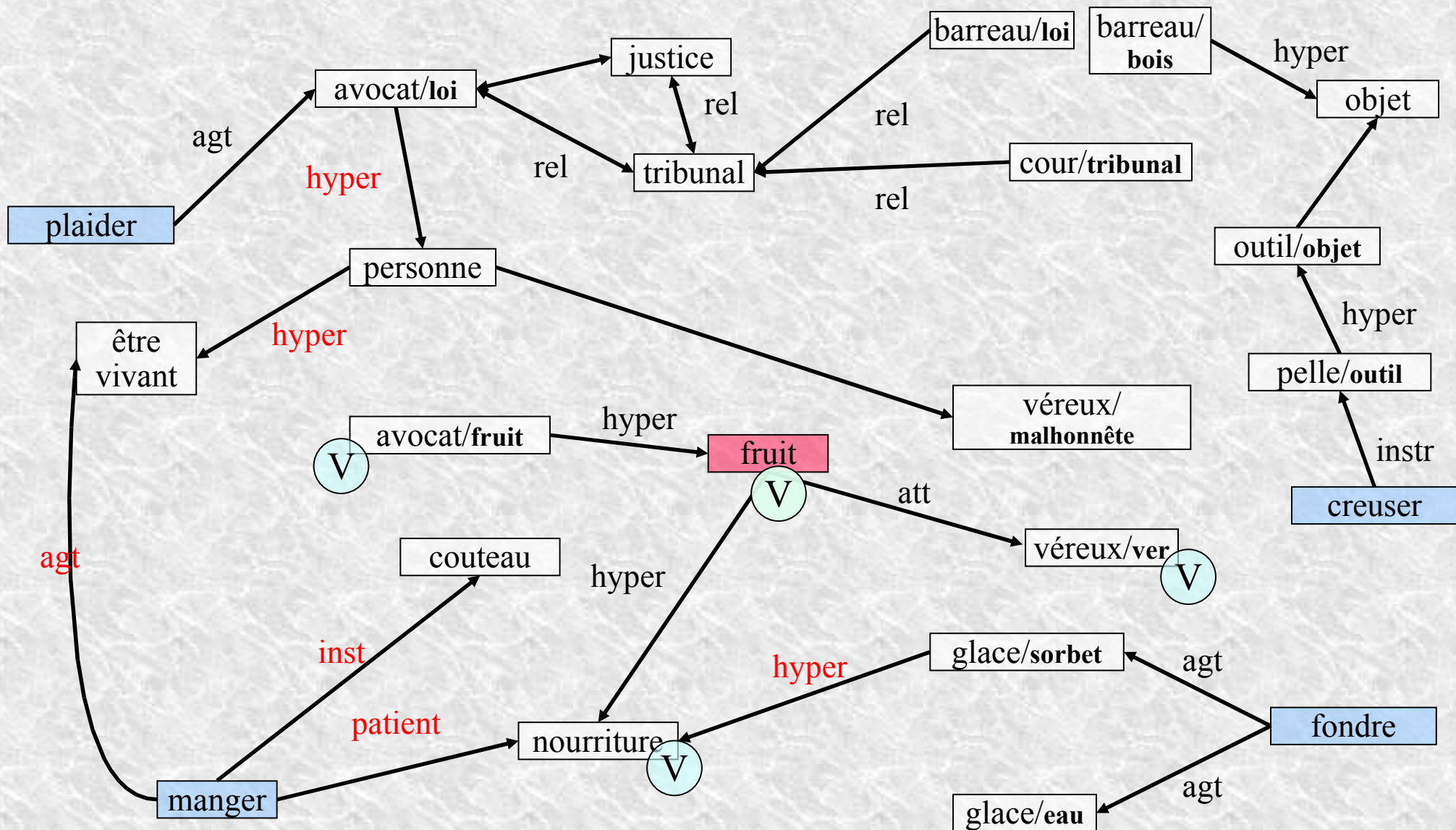
# Conceptual Vectors

## Construction



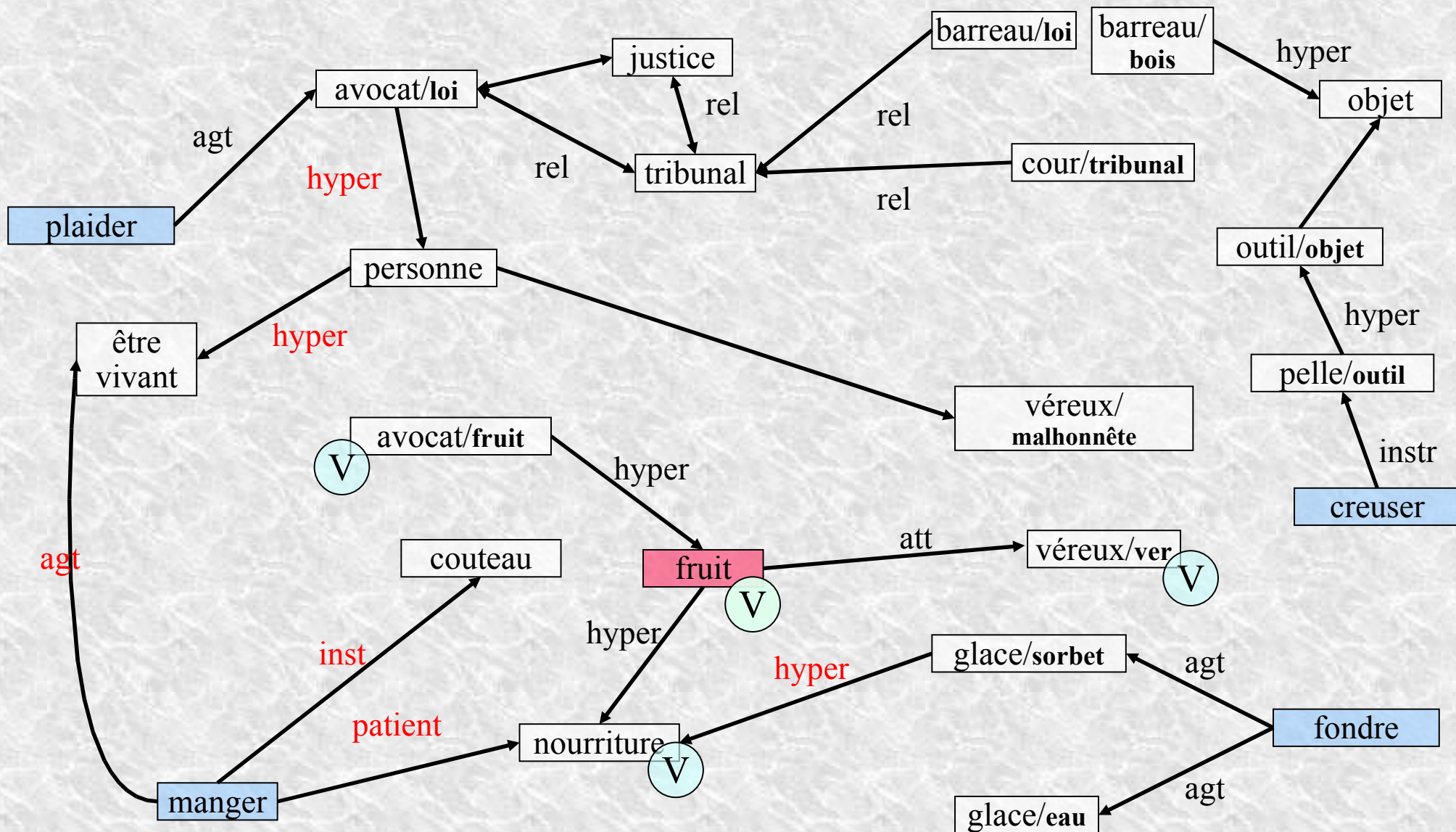
# Conceptual Vectors

## Construction

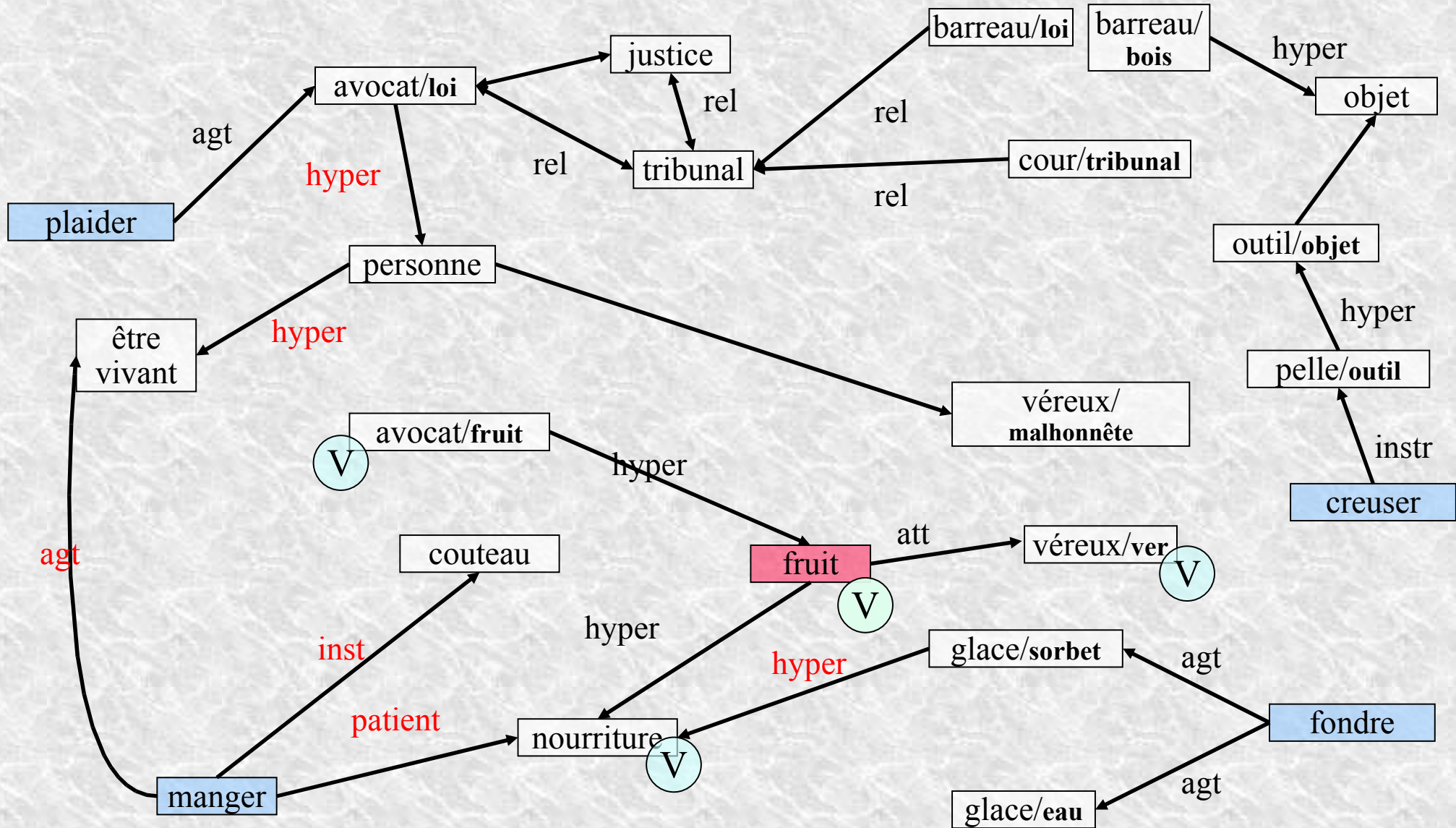


# Conceptual Vectors

## Construction



# Conceptual Vectors Construction



# Ant algorithm

More...

## Production ratio between ant castes

Adjustment through rewards  $\Rightarrow$  increase if successfully  
 $\Rightarrow$  decrease toward equiprobability with time

## Self termination

Internal factors ( $\neq$  simulated annealing)

«freezing ants» - narrow the gap between min and max of pheromones on a given arc

Produced when a stability is perceived locally  $\Rightarrow$  contamination

## Inhibition

«killing ants» - alert pheromone –  
kill foe ant, compromising the durability of their anthill

# Outline

- Semantic Analysis      What to look at? – Applications
- Casting                  Conceptual Vectors – Analysis Trees – Lexical Network
- Ant algorithm      Principles - Without lexical network - With lexical network
- Ressource Production      Lexical Network - Conceptual Vectors
- Future directions      Toward holistic analysis: syntax + semantic



# Future directions

Taking into account frequent schemas

Prepositional phrase attachment [gala, lafourcade]

Acceptation distribution

over the general domain

over a specific domain

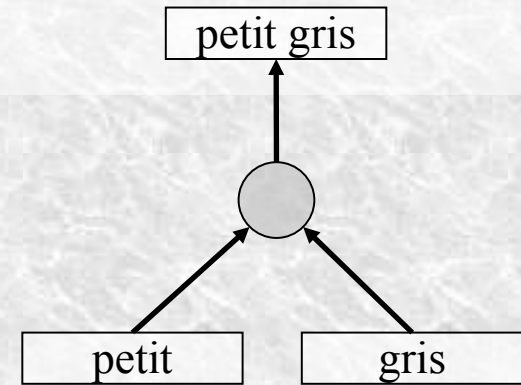
Frequency for compound terms meanings

petit gris - serpent de mer - ...

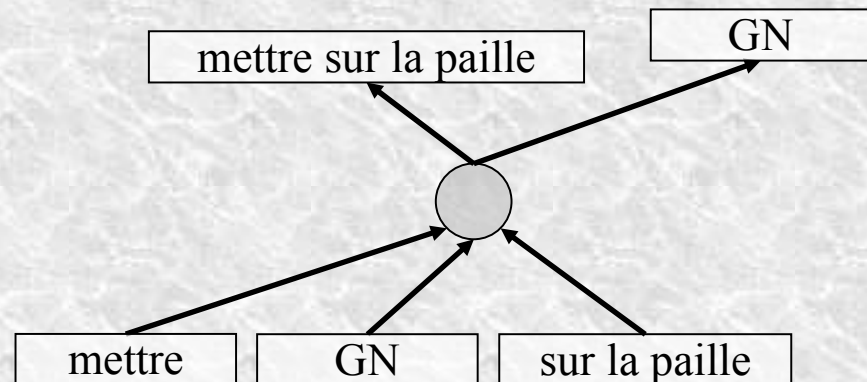
Frequency for locution meanings

mettre x sur la paille

to ruin/bankrupt x  
to put x on the hay

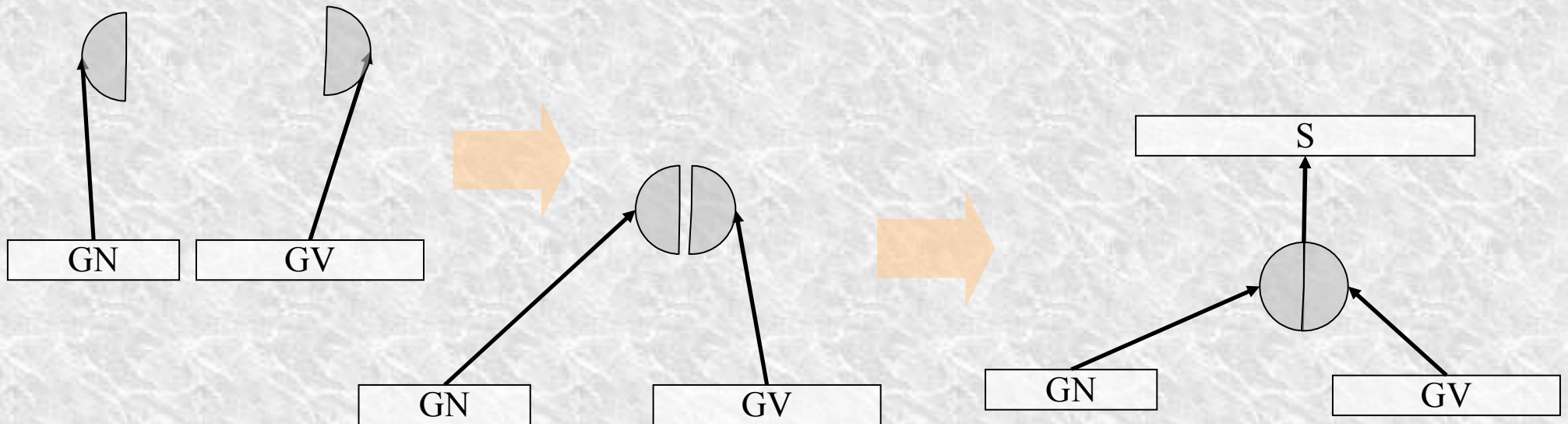
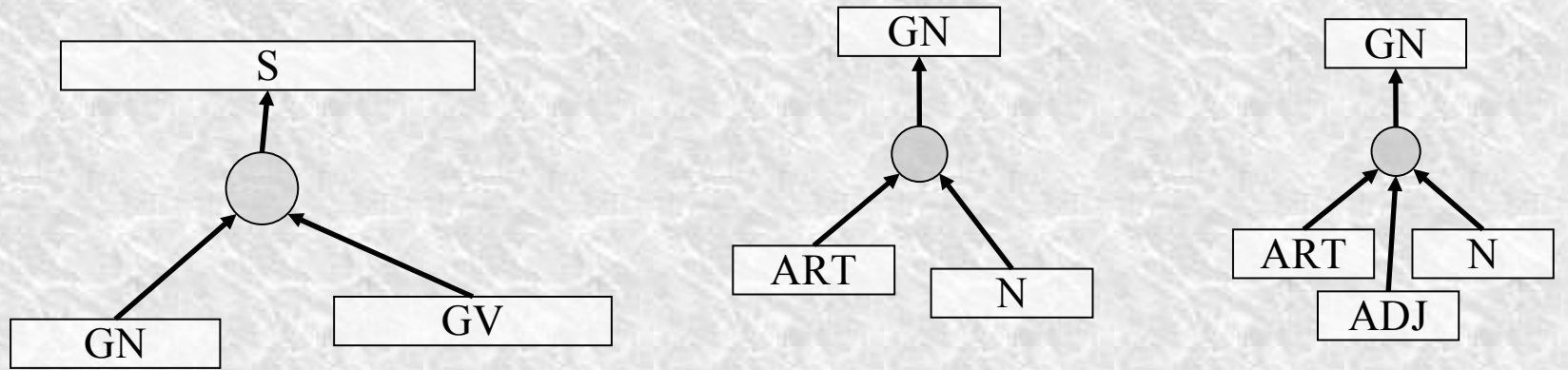


a squirrel – a paintbrush  
a mushroom  
a snail  
a commuter train



# Future directions holistic analysis

## Syntactic schemas in the lexical network



### All nodes

- produce information

- request information

- Produce agents for these tasks

### Information

- Deduced from the lexical network

- Symbolic (MAS, SING, V, GV, ...)

- Vectors (different types according to network relation labels)

### Ants as agents

- Information carriers

- Structure builders

# Conclusion

Holistic analysis

<= what is important and when?

Emergence of some solutions through errors

<= perfect ants would produce nothing

Coordination problem facilitated

<= no central control but population control

Generic and extensible architecture

<= adding new information type is (relatively) easy

Tuning is difficult => can be assessed through experiment