Vector Parser

Cognition a compression or expansion of the world?

The fundamental conjecture of the Vector Parser was that cognition may be an expansion of the world, not a compression.

Not totally crazy anymore

The central idea that cognitive structure is an expansion not a compression **might seem odd, or implausible**.

But not totally crazy anymore. OpenCog embracing it?

The Vector Parser was an early attempt to implement this idea.

This fuller technical description, and an earlier paper I presented in 2000, are now old.

But the central idea is still not widely appreciated.

Short history of Vector Parser

The Vector Parser came out of work done initially **between 1994-96 attempting to formalize grammatical errors** for a project in Hong Kong.

Unique environment: tension between machine learning and distributed representation.

Key insight in 1996 was that **machine learning of grammar generated** something strikingly resembling **a kind of bottom up uncertainty principle.**

Still new in 2021.

Grammar can’t be learned

This remains the central conjecture of the Vector Parser.

Grammar cannot be learned. **There is a tension between different groupings which produces a kind of uncertainty principle**.

c.f. Paul Hopper. BIG history in linguistics dating back to Chomsky schism. But not understood mathematically in linguistics.

Flipside - Expansion

But there’s a **flipside** to that. Grammar can be put together in new ways all the time.

And all those new ways are **actually far more productive, even creative**. It’s a feature, not a bug.

Patterns can grow without bound

We know there are patterns which grow without bound.

Gosper’s Glider Gun an existence proof.

But finite **“learning” not “expansion” still dominates AI in 2021**. Even 10 years after deep learning.

Overview

I’ll try to split the presentation into four parts:

1. Quick worked example
2. Relationship to deep learning/neural nets/transformers
3. Relationship to more recent work on mathematical representations of indeterminacy and non-finite representation, OpenCog, Coecke.
4. Something fresh and fun, “one more thing”, looking to the future, at the end. Don’t go to sleep!

1) Quick overview of Vector Parser

The vector parser works much the **same way machine learning** of grammar does.

Zellig Harris, "notion of interchangeability": “constituents of the same type can be replaced by each other”

Generally machine learning attempts to **compress language structure based on substitutions**.

Same as Machine Learning. Just upside down

By contrast **the vector parser attempts to expand structure based on substitutions. Upside down.**

Algorithm (substitutes, rather than combining as in ML):

1. Substitutes for pairs of words with single words (Principle of Interchangeability)
2. Iterate and repeat.

Structure appears fairly easily. There’s a preferred order of substitution of the pairs ((A B) C) or (A (B C)).

It forms trees.

Example: “The White House news conference Monday”

“White House” can be expanded with single word substitutes {senate, ... jsp,...}

New vector - sum of components of vectors for observed pairs

word	similarity score
white house	1.4e+02
white sand beach	22
the white house	19
white sand beaches	17
soviet union	17
emerald city	17
same time	17
oval office	17
senate	13
main house	13
white sandy beach	13
united states	13
earl	13
philippines	13
jsp	13
seller	13

<--

Observed pairs between components of vectors for logical halves of: "(white house)"

headword	tailword	pair frequency
white	house	558
the white	house	377
white	sand beach	222
white	sand	625
white	sand beaches	225
white	water rafting	57
white	house job	19
white	mountains	120
white	house that	9
the upper	house	81
white	rhino	37
white	house on	16
a white	house	24
stone	house	99
white	walls	43

“News conference” can be substituted by {...meeting,… report,... committee,...}

New vector - sum of components of vectors for observed pairs

word	similarity score
news conference	3.1e+02
press conference	1e+02
month	11
rule	5.3
news stories	5.2
meeting	4.8
peace conference	4.7
child	4.6
man	3.3
walk	3.1
report	0.5
emerald city	0.5
bank	0.5
committee	0.44
emperor	0.43
peace prize	0.43
seller	0.43

<--

Observed pairs between components of vectors for logical halves of: "(news conference)"

headword	tailword	pair frequency
news	conference	35
press	conference	80
litigation	reform act	35
news	agency	15
story	girl	303
dispute	settlement	87
news	release	8
diet	session	49
customer	complaints	106
collection	capitalization	18
task	force	146
entrance	hallway	14
reference	portfolio	16
analysts	forecast	30
field	mouse	18
dance	lesson	13

These single word substitutes then combine to generate new single word substitutes for whole phrase {jsp, diet, ldp…} (Japanese corpus?)

New vector - sum of components of vectors for observed pairs

word	similarity score
loan loss provision	4.3e+04
nyse rule	4.3e+04
embassy row	3.8e+04
loan loss provisions	2.6e+04
jsp	2e+03
instrument	2e+03
panel	2e+03
evenings theme evenings	1.9e+03
securities industry	1.9e+03
chief executive officers	1.9e+03
auditor general	1.9e+03
diet	2.2e+03
proposal	2.2e+03
ldp	2.2e+03
foreign firms	2.2e+03

<--

Observed pairs between components of vectors for logical halves of: "((white house) (news conference))"

headword	tailword	pair frequency
loan	loss provision	73
nyse	rule	73
embassy	row	21
white house	complex	7
senate	report	21
directors	meeting	6
waves	records	14
last	year	916
senate	committee	16
wall	street	166
british	army officers	8
fairy	queen	9
united states	senate	10
ground	floor	1443

Those single word substitutes are found to combine with “the” (most things do.)

New vector - sum of components of vectors for observed pairs

word	similarity score
the soviet union	2.6e+05
the legislative assembly	2.4e+05
the united states	2.3e+05
the province	2.2e+05
the body	1.8e+05
the project	1.8e+05
the university	1.6e+05
the lake	1.5e+05
the floor	1.3e+05
the report	1.3e+05
the manager	1.3e+05
the room	1.3e+05

<--

Observed pairs between components of vectors for logical halves of: "(the ((white house) (news conference)))"

headword	tailword	pair frequency
the	story girl	301
the	past decade	48
the	mto	261
the	conference meeting	23
the	llp	30
the	jsp	92
the	federal government	120
the	rear	403
the	main road	206
the	diet	192
the	matter	547

And, as a unit, enough of these combinations are found to occur in the context of substitutes which expand “Monday”.

New vector - sum of components of vectors for observed pairs

word	similarity score
las vegas monday	6e+06
toronto monday	5.5e+06
tokyo monday	3.7e+06
london heathrow	2.8e+06
toronto niagara falls	2.6e+06
nairobi mombasa	2.3e+06
the boat landing	2.2e+06
alcohol tobacco	2.2e+06
september DIGIT october	2e+06
cairo luxor	1.8e+06
the diet session	1.8e+06
toronto tuesday	1.8e+06
DIGITpm monday	1.8e+06
july DIGIT august	1.7e+06

<--

Observed pairs between components of vectors for logical halves of: "((the ((white house) (news conference))) monday)"

headword	tailword	pair frequency
las vegas	monday	6
toronto	monday	6
tokyo	monday	7
london	heathrow	436
toronto	niagara falls	34
nairobi	mombasa	35
the boat	landing	8
alcohol	tobacco	14
cairo	luxor	18
the diet	session	6
toronto	tuesday	8
DIGITpm	monday	10

The path of largest pairwise substitution generates a tree.

```
                ((the ((white house) (news conference))) monday)
                                                              / \
                (the ((white house) (news conference)))                monday
   / \
the                    ((white house) (news conference))
                                  / \
                (white house)                      (news conference)
                   / \                                / \
              white        house            news           conference
```

Very much like Grammatical Induction/Machine Learning

It uses Harris's Principle of Interchangeability. But **substitutes to expand**, rather than **combining to compress**.

Expanded classes don't have to be global. They can be specific to a given context.

Everything else is the same.

Demo online (intermittently)

You can play with this yourself online (when the server is up!)

http://demo.chaoticlanguage.com

Remember it’s 20 years old. More than one simultaneous user will probably crash it!

English Examples

Where this system is shown to best advantage is with lexically idiosyncratic modification decisions.

- Cheap weekend return ("weekend return" identified as a strong unit.)
- English country house ("country house" identified as unit.)
- Electric sun roof ("sun roof" identified as a unit.)
- Car with electric sun roof
- Family station wagon ("station wagon" identified as a unit.)
- Sun terrace's garden
- Charming country house
- Stock market crash
- Open plan living
- Open plan kitchenette
- Lower house election
- United States' senate
- Senate committee meeting
- Indian ocean coastline
- Utility holding company
- Public service employee
- Two bedroom cottage
- Air conditioned cottage
- One bedroom suites
  Note "bedroom suites" wins out for plural: two bedroom suites (your call).
- Scuba diving lesson
- Ferry landing stage
- Tokyo stock exchange
- Air passenger bookings
- Complete holiday service
- Late morning departure
- Public sector organizations
- Microsoft Office applications
- Summer season holiday
- Future health care
- Human resource management
- Popular holiday resort
- Federal copyright law
- Evening walking tour. (Identifies the "walking tour" class.)
- Short stroll away (A "short stroll", not a short "stroll away".)
- White sandy beaches ("sandy beaches" identified as a unit.)
- Relaxing holiday away("relaxing holiday" stronger unit?)
- Relaxing week away("week away" stronger unit?)
- spectacular desert park (Identifies the "desert park" class, and prefers it to the equally plausible "spectacular desert" class.)
- the White House news conference Monday (Lots of things

"Vector" Parser - Engine

Enter a sentence fragment here (pref. travel or politics): Submit
Reset

Estimated Parse Tree

```
           ((the ((white house) (news conference))) monday)
                                                          /  \
            (the ((white house) (news conference)))                monday
  /  \
the                 ((white house) (news conference))
                                  /  \
        (white house)                     (news conference)
          /  \                               /  \
  white          house                news          conference
```

2) Contrast with deep learning/transformers

Remember, the **motivation for the Vector Parser** was that **learning resulted in a kind of "macro quantum", "uncertainty principle"**, which made global grammar impossible.

How does deep learning deal with this?

By ignoring it. (Ignoring? c.f. Chomsky. Ignores his schism.)

Why has deep learning crushed symbolic AI?

Success since 2010. Why? Usual answer is machine learning. But **why not symbolic machine learning?** Why distributed representation?

Stanford: "**we currently lack a clear understanding of how they work**"

Deep learning lacks theory.

Is the indeterminacy the (hidden) theory of deep learning?

From point-of-view of indeterminacy they **work better than learned symbolic representations, exactly because a lot is left distributed** and not “learned”?

Deep learning moving forwards, backwards?

If **indeterminacy** is the **missing theory of deep learning**, then “learning” is the wrong way to go.

They use back-prop like a talisman, thinking they are trying to compress. But what works is always more layers, bigger representations!

It’s like, oh wow, bigger representations does stuff we hadn’t imagined. Let’s make them “Foundational”. But "**we currently lack a clear understanding of how they work**"!

Simply embrace expansion and it’s easy.

With the vector parser we do know. Follows same principles of substitution.

The only difference we accept **they expand**.

You **don’t have to burn a hole in the planet** trying to learn infinity. Just expand it as you need it.

You can have structure. Just recognize it’s ephemeral, growing.

Semantic glosses.

Transformers **don’t give structure**.

Expansion explains this is because you **can’t learn infinity** (especially contradictions.)

What you have with transformers are **semantic glosses**.

The Vector Parser also generates semantic “glosses”

It comes from successive substitutions. Same as transformers?

It’s very **crude**. I think my database was a **40Mb corpus, not 800GB!** (EleutherAI?) But some amusing semantic equivalents do come up:

Well, that’s all right then -> Aye, aye, Sir!

```
         ((well (thats (all right))) then)
                                          /  \
         (well (thats (all right)))                then
   /  \
well              (thats (all right))
              /  \
      thats               (all right)
                          /  \
                       all          right
```

New vector - sum of components of vectors for observed pairs

word	similarity score
aye aye sir	5.5e+04
aye aye	5.1e+03
oh oh oh	4.2e+03
rachel lynde	2.6e+03
lynde says	1.2e+03
aye sir	8.6e+02
ah yes	7.7e+02
unless otherwise stated	5.9e+02
i wonder if	5.7e+02
medlock	5.6e+02
havisham	5.6e+02
oh marilla	5.1e+02
oh yes	4.8e+02
ah	4.8e+02

Observed pairs between components of vector for logical halves of: "((well (thats (all right))) then)"

headword	tailword	pair frequency
aye aye	sir	9
aye	aye	14
oh oh	oh	8
rachel	lynde	36
lynde	says	50
aye	sir	9
ah	yes	7
unless otherwise	stated	30
i hope	youll	25
maybe	one day	15
oh	yes	62
oh	marilla	70
im sure	hell	6

How does it create these semantic equivalents?

A succession of pairwise substitutions, selected by context:

(all right) -> yes -> yes -> aye aye -> aye aye sir

“All right” substitutes to {okay, yes,...}

New vector - sum of components of vectors for observed pairs

word	similarity score
all right	13
all tastes	1.6
all ages	1.6
okay	1.3
all season	1.2
all the facilities	1.2
all inclusive	1.2
very difficult	1.1
easier	1.1
all the comforts	1.1
wrong	1.1
in trouble	1
all at once	1
thank you	0.91
justified	0.9
all kinds	0.89
yes	0.87
i said	0.84

<--

Observed pairs between components of vectors for logical halves of: "(all right)"

headword	tailword	pair frequency
all	right	242
all	tastes	125
all	ages	200
all	season	496
all	inclusive	404
all	at once	63
all	the comforts	76
all	the facilities	135
all	the time	187
all	the way	144
all	directions	50
its all	right	17
all	kinds	109
all	my fault	6
all	year round	107
all	abilities	27
all	the attractions	55

Add “that’s” & (thats (all right)) substitutes to “yes, yes”

i think that	16
i thought	16
do you think	15
eh	15
i think	15
in the meantime	15
that is why	14
by the time	14
anyhow	14
the first thing	14
anyway	14
soon after	14
i know	14
no wonder	14
yes	14
oh yes	14
maybe	14

its all	right	17
are you	ready	16
he got	married	6
yes	yes	24
identify	all documents	9
im	going	252
thats	really	15
it was	impossible	26
theres	nothing	81
identify	all persons	18
youre	tired	7
hes	dead	13
are you	sure	36
its not	difficult	6
it is	impossible	65
thats	better	7
there was	nothing	83

Add “well” & (well (thats (all right))) substitutes to aye aye

New vector - sum of components of vectors for observed pairs

word	similarity score
well as for	8.1e+02
well i guess	6.1e+02
aye aye	6.1e+02
well i hope	5.7e+02
well weve plenty	3.6e+02
well i suppose	3.6e+02
rachel lynde	3.3e+02
oh yes	2e+02
impeccably appointed space	1.9e+02
oh oh oh	1.8e+02
well i think	1.7e+02
well maybe	1.3e+02
yes i suppose	1.3e+02
i hope	1.1e+02
oh i know	99
i wonder if	95
[illegible]	[illegible]

<--

Observed pairs between components of vectors for logical halves of: "(well (thats (all right)))"

headword	tailword	pair frequency
well	as for	63
well	i guess	7
aye	aye	14
well	i hope	11
well	weve plenty	9
rachel	lynde	36
well	i suppose	7
oh	yes	62
impeccably	appointed space	20
oh	oh oh	8
well	i think	17
well	maybe	6
oh	i know	16
yes	i suppose	6
ah	yes	7
[illegible]	[illegible]	[illegible]

Add then & ((well (thats (all right))) then) subs to “aye aye sir”

New vector - sum of components of vectors for observed pairs

word	similarity score
aye aye sir	5.5e+04
aye aye	5.1e+03
oh oh oh	4.2e+03
rachel lynde	2.6e+03
lynde says	1.2e+03
aye sir	8.6e+02
ah yes	7.7e+02
unless otherwise stated	5.9e+02
i wonder if	5.7e+02
medlock	5.6e+02
havisham	5.6e+02
oh marilla	5.1e+02
oh yes	4.8e+02
ah	4.8e+02
currie	4.7e+02

Observed pairs between components of vectors for logical halves of: "((well (thats (all right))) then)"

headword	tailword	pair frequency
aye aye	sir	9
aye	aye	14
oh oh	oh	8
rachel	lynde	36
lynde	says	50
aye	sir	9
ah	yes	7
unless otherwise	stated	30
i hope	youll	25
maybe	one day	15
oh	yes	62
oh	marilla	70
im sure	hell	6
must surely	rank	10

Possibly this generation of semantic glosses is the same

The context filter for substitutions is very primitive for the Vector Parser.

But I would guess that the generation of “semantic glosses” works in much the same way as transformers.

Potential benefits of expanding model over transformers

1. **Structure**
2. “One-shot learning” no different to any other input. “Corner cases”, “long tail”? **Creativity**..” (Freewill, consciousness? Creates “time”?)
3. **Doesn’t need enormous data** to list all possibilities beforehand.
4. **Doesn’t need $millions on compute time** to compute everything beforehand. Only generates structures it meets.
5. **Finally a theory** for distributed representation. The theory is **undecidability**.

Basically it provides a **theory for distributed models**. An argument why they work, and how they need to change.

So that’s the contrast with deep learning.

Probably the same thing

But they **ignore undecidability**.

They work backwards the whole time. Thinking they’re compressing, but actually expanding ever more structure.

But because they ignore it, working backwards, they don’t know how to move forwards.

3) Contrast with approaches based on indeterminate structure

No longer totally crazy.

Unlike 20 years ago, there is now some other AI work which is starting to come to the same conclusions.

Recent work on decidability in OpenCog Atomspace. Or Bob Coecke “togetherness”.

History recapped

That **linguistic category** might be **undecidable** was the **initial motivation** for the vector parser.

Though **initially the link to mathematical dedicability was not known,** only an analogy to quantum type behaviour.

Freeman R. J., Example-based Complexity--Syntax and Semantics as the Production of Ad-hoc Arrangements of Examples, Proceedings of the ANLP/NAACL 2000 Workshop on Syntactic and Semantic Complexity in NL Processing Systems, pp. 47-50.

Categories are represented as vectors. Irreducibly distributed.

Analogues found in other fields: physics, maths, and linguistics.

Over the years I found more an more analogues in different fields.

My initial analogy was between assemblies of elements and quantum uncertainty, c.f. **Robert Lachlan** “maco quantum” properties of assemblies. (Reinventing Physics from the Bottom Down - Nice “upside down” feel to it.)

Hofstadter’s GEB. Led to a closer study of **Goedel incompleteness**, which led to **category theory**.

I also stumbled on the observation of indeterminacy creating **Chomsky’s schism in linguistics**. Lamb, Newmeyer.

It was amusing that both **Goedel** and **Chomsky** thought indeterminacy pointed to the existence of external oracles. God for Goedel, and Universal Grammar for Chomsky.

Now, finally some signs of this AI.

- Bob Coecke’s “**Togetherness**” with a quantum/category theoretic formulation.
- OpenCog also now embracing **category theory** notions in their **atomspace**?

Progress! In 2010 decidability was not a factor for OpenCog

OpenCog dev to author, 2010: “Clearly there's a lot of work being done in natural language processing ... but this is primarily an engineering task; **I don't see how it relates to questions of decidability**"

Vepstas, “Mereology”, 2020: "In the remaining chapters, the sheaf construction will be used as a tool to create A(G)I representations of reality. Whether the constructed network is an accurate representation of reality is **undecidable**, and this is true even in a narrow, formal, sense."

Vector categories are sets, and elements link to other sets.

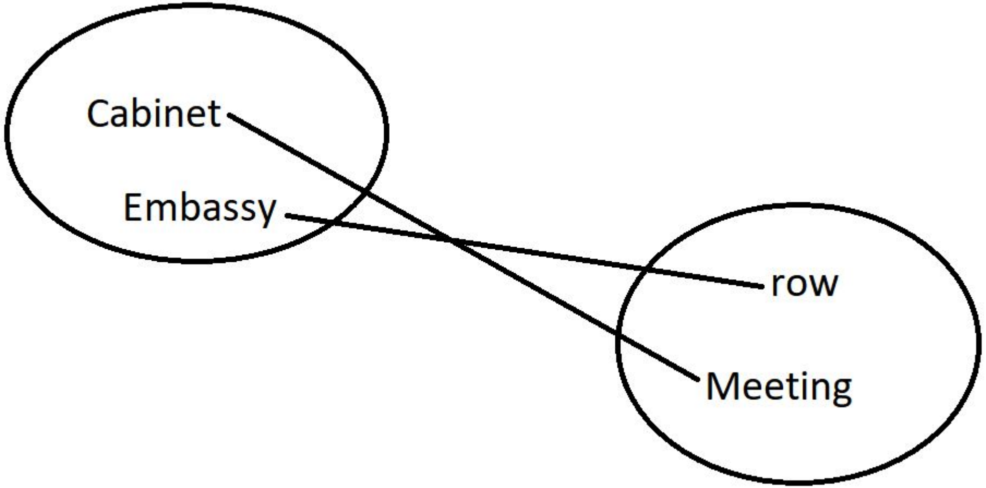
New vector - sum of components of vectors for observed pairs

word	similarity score
loan loss provision	4.3e+04
nyse rule	4.3e+04
embassy row	3.8e+04
loan loss provisions	2.6e+04
cse rule	2.4e+04
cabinet meeting	2.3e+04
litigation reform act	2.2e+04
loan loss reserve	2e+04
fourth quarter loan	1.8e+04
crown corporations secretariat	1.8e+04
white house job	1.7e+04
glass bottom boat	1.5e+04
motion picture arts	1.2e+04
dsb meeting	1.2e+04
story girl	1.1e+04
nasd rule	1e+04
trade winds sweep	9.5e+03

Observed pairs between components of vectors for logical halves of: "((white house) (news conference))"

headword	tailword	pair frequency
loan	loss provision	73
nyse	rule	73
embassy	row	21
loan	loss provisions	176
cse	rule	21
cabinet	meeting	51
litigation	reform act	35
loan	loss reserve	39
fourth	quarter loan	30
crown	corporations secretariat	25
white house	job	19
glass	bottom boat	93
motion	picture arts	10
dsb	meeting	6
story	girl	303
nasd	rule	10

Correspondence to a hypergraph formalism?



The vector parser’s generated categories just graph ops?

That the vector parser generates categories dynamically is perhaps only equivalent to set operations.

How product categories are generated is perhaps specific

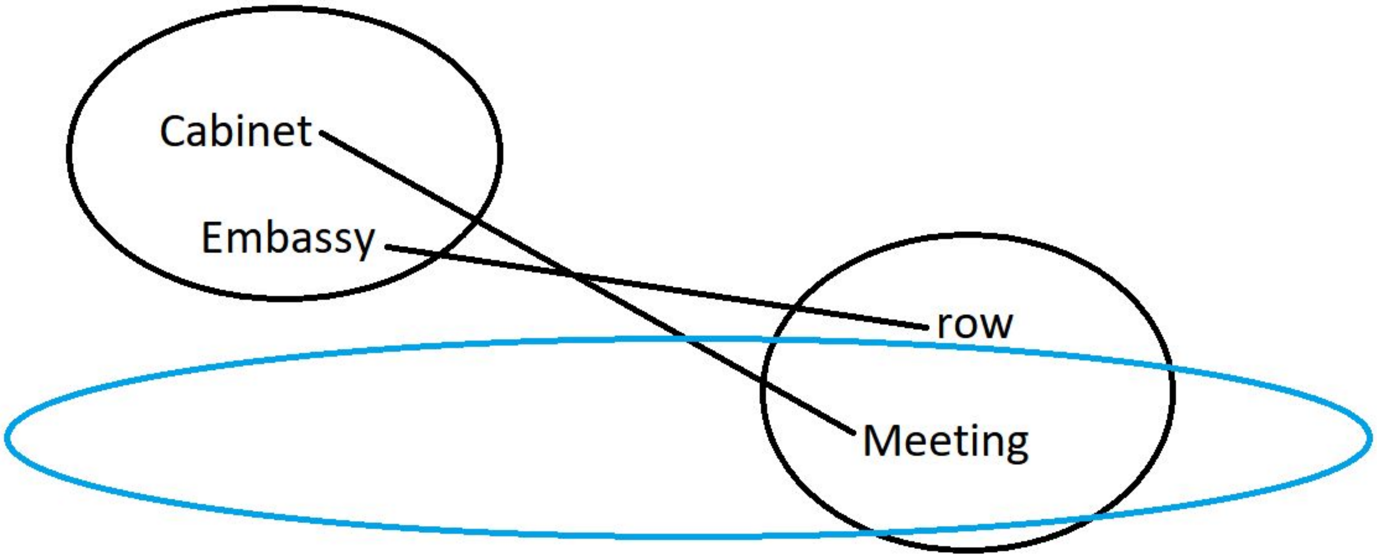
New vector - sum of components of vectors for observed pairs

word	similarity score
loan loss provision	4.3e+04
nyse rule	4.3e+04
embassy row	3.8e+04
loan loss provisions	2.6e+04
cse rule	2.4e+04
cabinet meeting	2.3e+04
litigation reform act	2.2e+04
[illegible]	[illegible]
scene	2.5e+03
defendant regents	2.5e+03
third quarter loan	2.5e+03
sec rule	2.5e+03
trade secret claim	2.4e+03
opposition parties	2.4e+03
nation	2.4e+03
meeting	2.4e+03
young man	2.4e+03
ruling party	2.4e+03

Observed pairs between components of vectors for logical halves of: "((white house) (news conference))"

headword	tailword	pair frequency
loan	loss provision	73
nyse	rule	73
embassy	row	21
loan	loss provisions	176
cse	rule	21
cabinet	meeting	51
litigation	reform act	35
public	accountants aicpa	17
seminar	participants	10
exchange	act release	47
web	site	182
fairy	story	12
young	man	117
international	film festival	12
rain	forest	95
ugc	project	116
target	audience	10

Note: product cat. doesn't necessarily include operand elems



2020: OpenCog is even hinting at the idea of expansion

Vepstas’: "Brutal Composition -- 'It's just the way things are." There is no true, nontrivial, and finitely long answer.'

Mereology: https://github.com/opencog/atomspace/blob/master/opencog/sheaf/docs/mereology.pd

So where can the vector parser add something?

I like this recent work.

But it does seem perhaps **trapped in formalism**. Also linguistic formalism. **Ignores “Chomsky” schism**.

Possibly being less steeped in formalism was an advantage. The Vector Parser had fewer qualms reducing representation to observation.

If all representations reduce to a network. **Why not just keep the original network of observations**?

All representations reduce to a network

If equivalence of graph operations is true, **why have an intermediate representation at all**? Any abstraction must always be partial. Or political.

You might capture some habitual forms, perhaps.

But generally the task of disambiguation will be comparable to the task of creation.

A simple **change of basis**. Doubling the work. **Laboriously coding into ambiguity, and then laboriously decoding out.**

Why struggle with atomspace at all?

Why not just go straight from networks of raw observation to contextual structure at run time?

Bob Coecke’s “Togetherness”

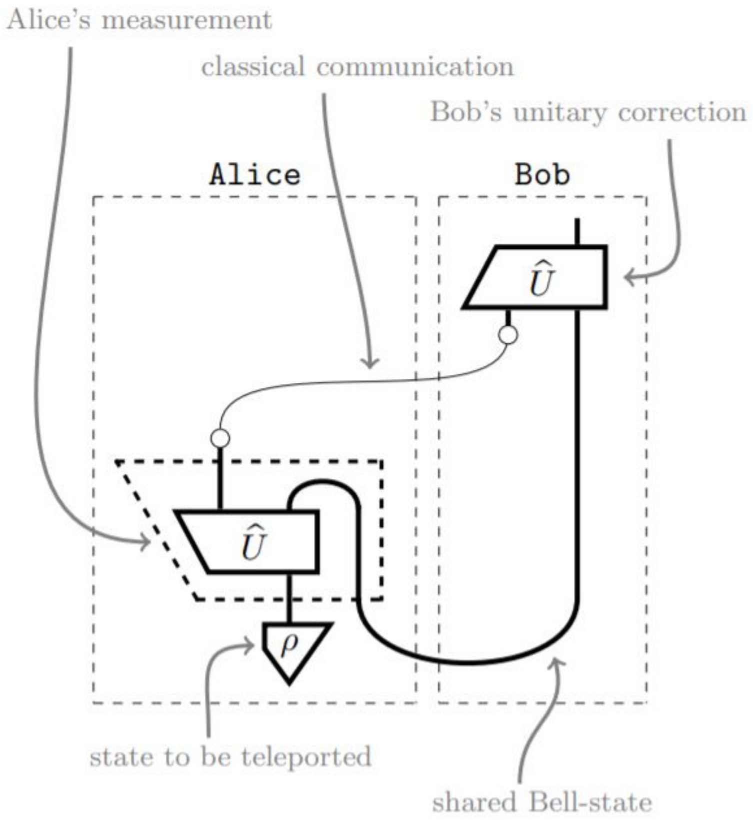
For Coecke's togetherness this criticism of **excessive preoccupation with formalism** might be **even more stark**.

He has a kind of quantum formalism. This matches my original insight that learned grammar displays a kind of "macro quantum" quality.

From quantum foundations via natural language meaning to a theory of everything, https://arxiv.org/abs/1602.07618

Coecke: “quantum theory naturally carries over to modelling how word meanings interact in natural language.”

Coecke “Togetherness”. Quantum formalism for meaning.



I’m sure it’s right. But why struggle with formalism at all?

Coecke's formalism embraces this quantum quality. I like it. I’m sure it’s the right formalism if you want to formalize structure globally.

But while for quantum mechanics we need such a formalism because we don't have direct access to any underlying assemblies (possibly Stephen Wolfram developing one??). **For cognition we do have access to underlying assemblies**. They are observations.

Isn't that what I said in my 2000 NAACL workshop paper. That the **key insight** is that observation is irreducibly complex, that cognitive structure only expands from that. That **you have to calculate everything from observation** anyway.

Summary: Three approaches to indeterminate structure in current tech

- NN’s ignore it.
    - Thus they are blind to the underlying indeterminacy, and can’t move forward: Stanford ““Foundation” paper: "**we currently lack a clear understanding of how they work**"
- Category theory, Coecke et al? Deal with indeterminate structure by laboriously encoding it into mathematical formalisms, and then laboriously decoding out of those mathematical formalisms.

  Also not clear they fully embrace expanding structure yet.

- Vector parser dealt with indeterminate structure by expanding new structure from context at run-time.

Conclusion:
The idea that cognition is an expansion of observation met sterile ground in 2000

The **vector parser was a first step** in the direction of pointing to an importance for expansion in cognitive representations.

But there was little interest in distributed models, period, in 2000.

Even in 2010 I met resistance to the idea **mathematical decidability** was relevant to language/cognitive structure. OpenCog flatly rejected it.

It was not immediately better on benchmarks.

Sample results - Hu Guoping, iFlytek, 2003

Compared Parser	Setting	P1	R2	P2	F-Score2
Symbolic Parser	基于整句解释的语法分析系统, 265 Sentence Single POS input	53.39%	53.48%	67.36%	59.62%
	基于整句解释的语法分析系统, 265 test Sentences All Possible POS input	56.74%	54.90%	69.43%	61.31%
Vector Parser	200M Corpus, -li 60, 2nd Indextable, 299 test Sentences	72.96%	63.98%	62.84%	63.41%

1.2 Compared with the ICT symbolic parser

Because the Symbolic from the ICT can just analyze 183 sentences of the 265 testing sentences, so following i sentences)

Compared Parser	Setting	P1	R2	P2	F-Score2
Symbolic Parser	基于整句解释的语法分析系统, 265 Sentence Single POS input	52.66%	54.27%	68.61%	60.61%
	基于整句解释的语法分析系统, 265 test Sentences All Possible POS input	56.40%	55.89%	70.25%	62.25%
	中科院计算所，pcfg-上下文无关183 sentences (82 failed)	64.03%	57.39%	60.32%	58.82%
	中科院计算所，prcfg-结构上下文相关,183 sentences (82 failed)	70.36%	64.78%	67.59%	66.16%
Vector Parser	200M Corpus, -li 60, 2nd Indextable, 299 test Sentences	72.66%	64.20%	62.90%	63.54%

What it got right

Hence the results for the vector parser as formulated were no better than symbolic parsers.

But convenient in that it could be applied to any language.

No initial, enormous, “learning” burden. Eliminates **corner cases**. Structure generated in context...

New structure meaningful. Creativity? Freewill, consciousness....

What it got wrong

I sought to make all structure dynamic so no expanding structure was lost.

But I missed that **abstraction into vectors was also abstraction**. Similarity is subjective/context dependent.

Also “attention” was only to immediate context. So in some ways worse than symbolic parsers.

To retain all information, **you need the network.**

One more thing...

3) Addendum. Recent work: Network and physical analogue?

For a long time I was looking for a physical analogue or visualization of a cross-product to demonstrate how the vector parser was generating new structure.

Then I read this paper:

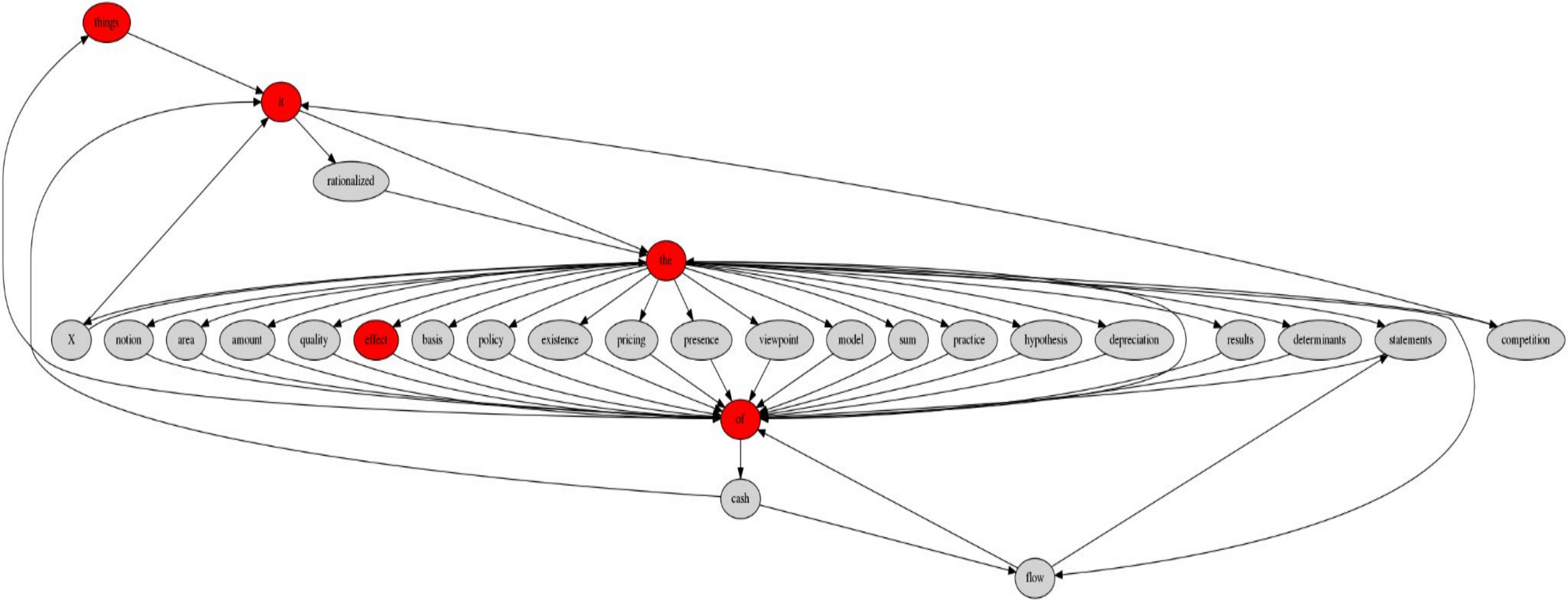
A Network of Integrate and Fire Neurons for Community Detection in Complex Networks, Marcos G. Quiles, Liang Zhao, Fabricio A. Breve, Roseli A. F. Romero

Synchronous oscillations may correspond to substitution groupings.

This is obvious in retrospect. Looking at graphs of substitution groupings.

Substitution classes depend on context. So do oscillations.

Network activated by “The effect of things it…” with substitution class for the_of



Language sequences naturally form “small world” networks

A **network of all sequences** in a corpus of language **will oscillate**.

Feedback is automatic because of the interconnectedness of language.

To get oscillation all you need to do is connect randomly inhibitory nodes to the right degree.

I have done some early experiments with this.

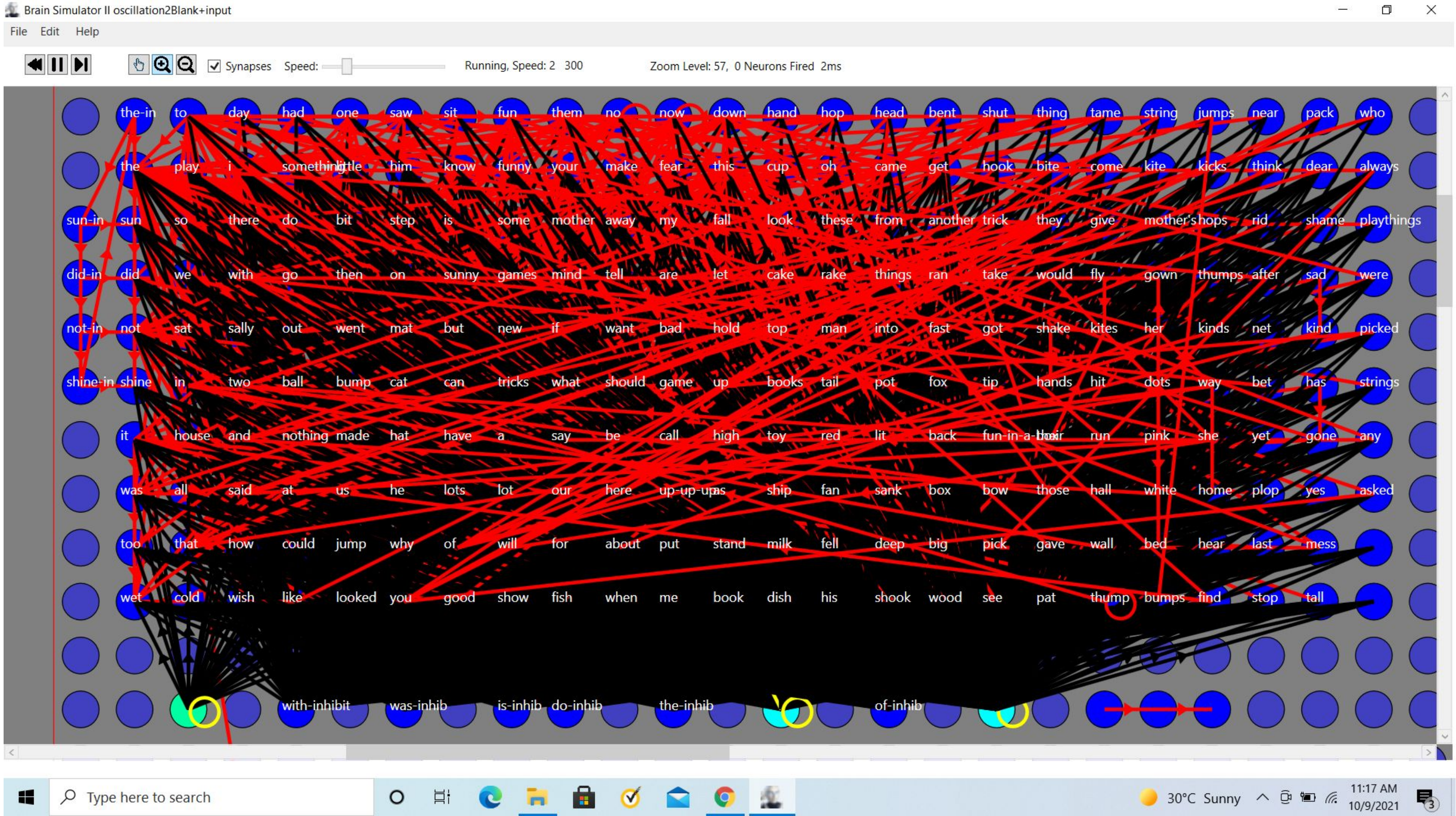
Experimental implementation using neurosimulator

This was done on neurosimulator, Brain Simulator II, by Charles Simon. A very nice GUI interface: https://futureai.guru/

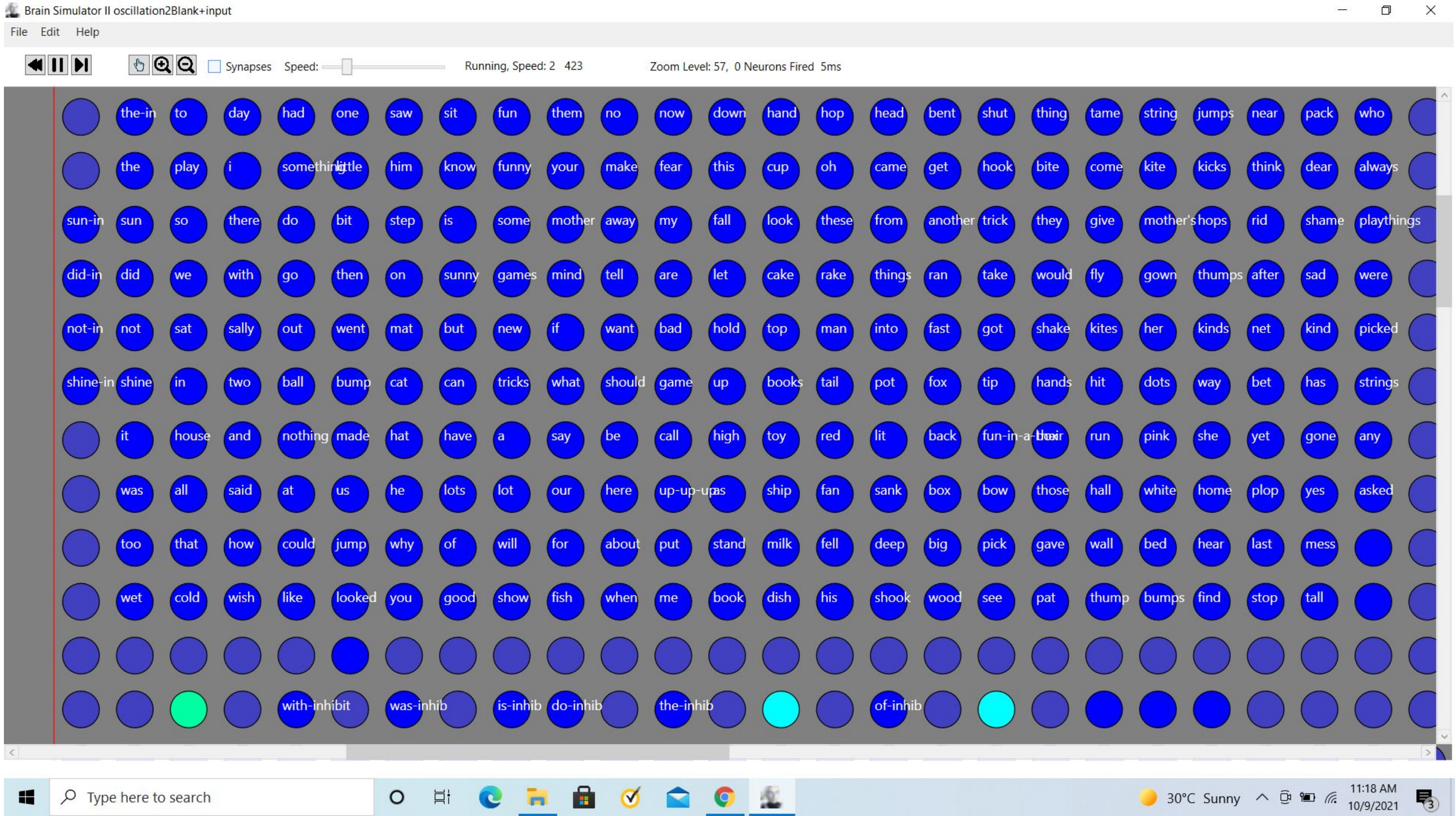
Charlie did some mods for me. Thanks Charlie!

In Brainsim II it is easy to jam language sequences into a network

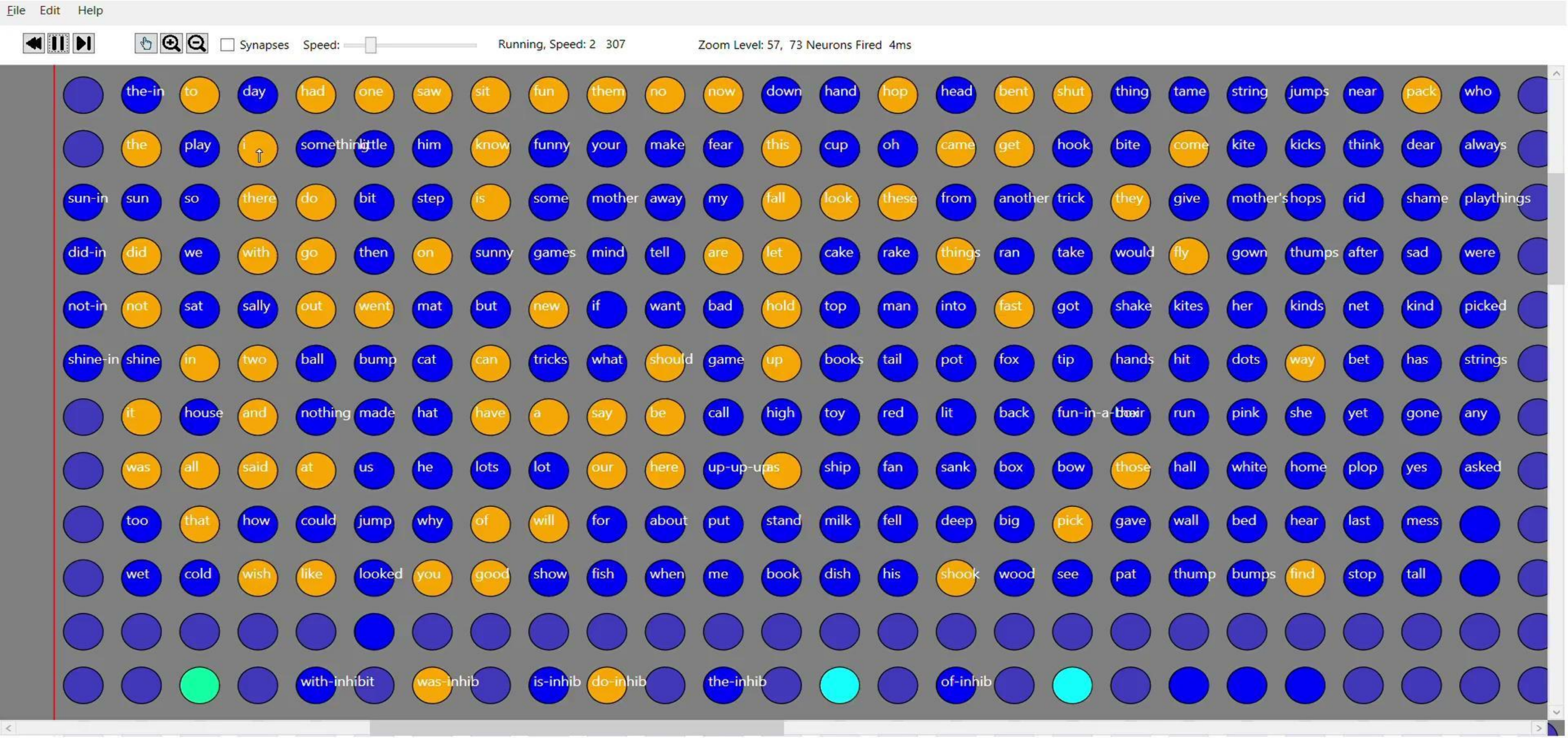
Though the GUI soon gets crowded with links displayed!



~1000 word corpus. Links hidden. Inhibitory nodes green.



Turns out easy to generate oscillations. Only need inhibition.



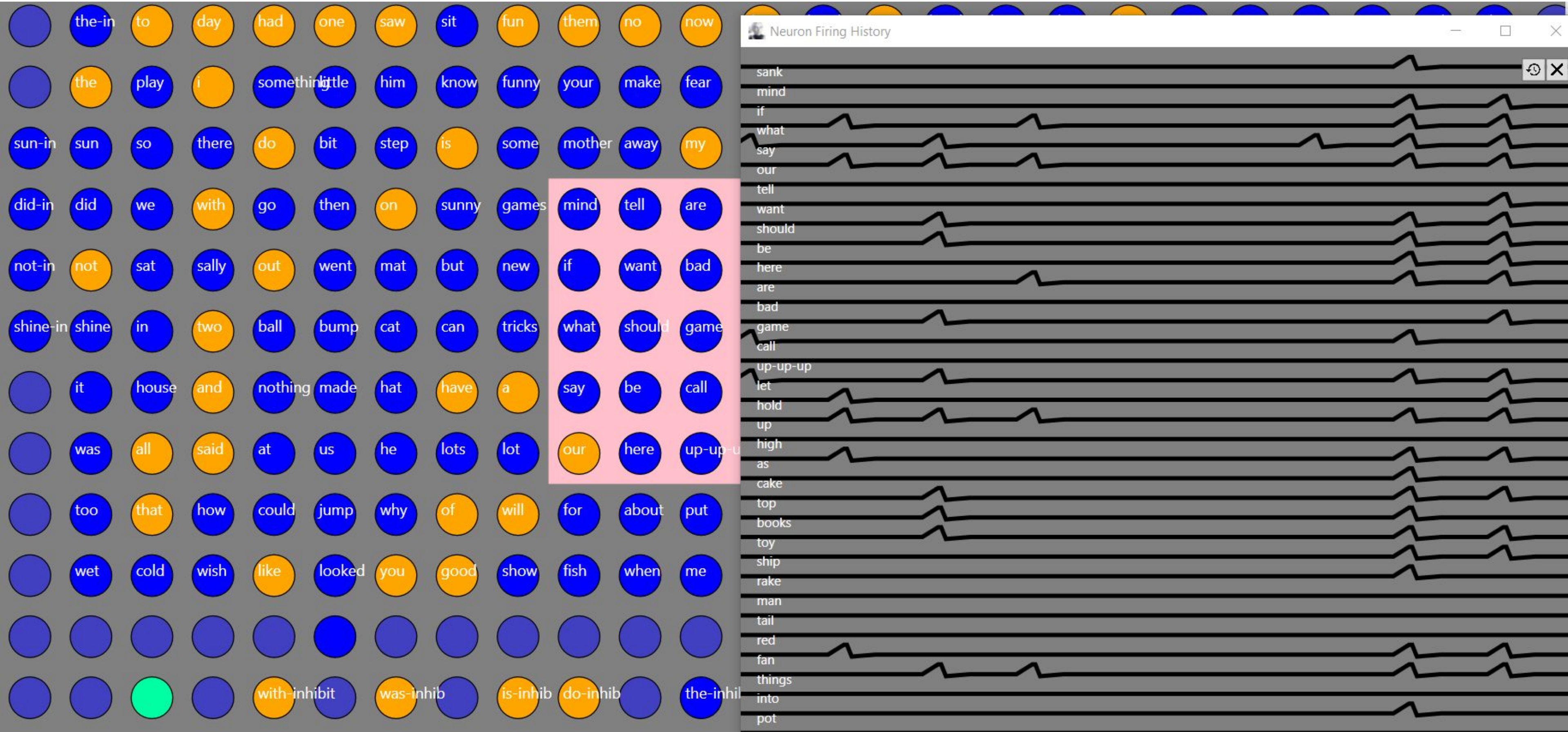
Substitution classes should synchronize

Elements oscillating in synchrony should be equivalent to substitution classes in the vector parser.

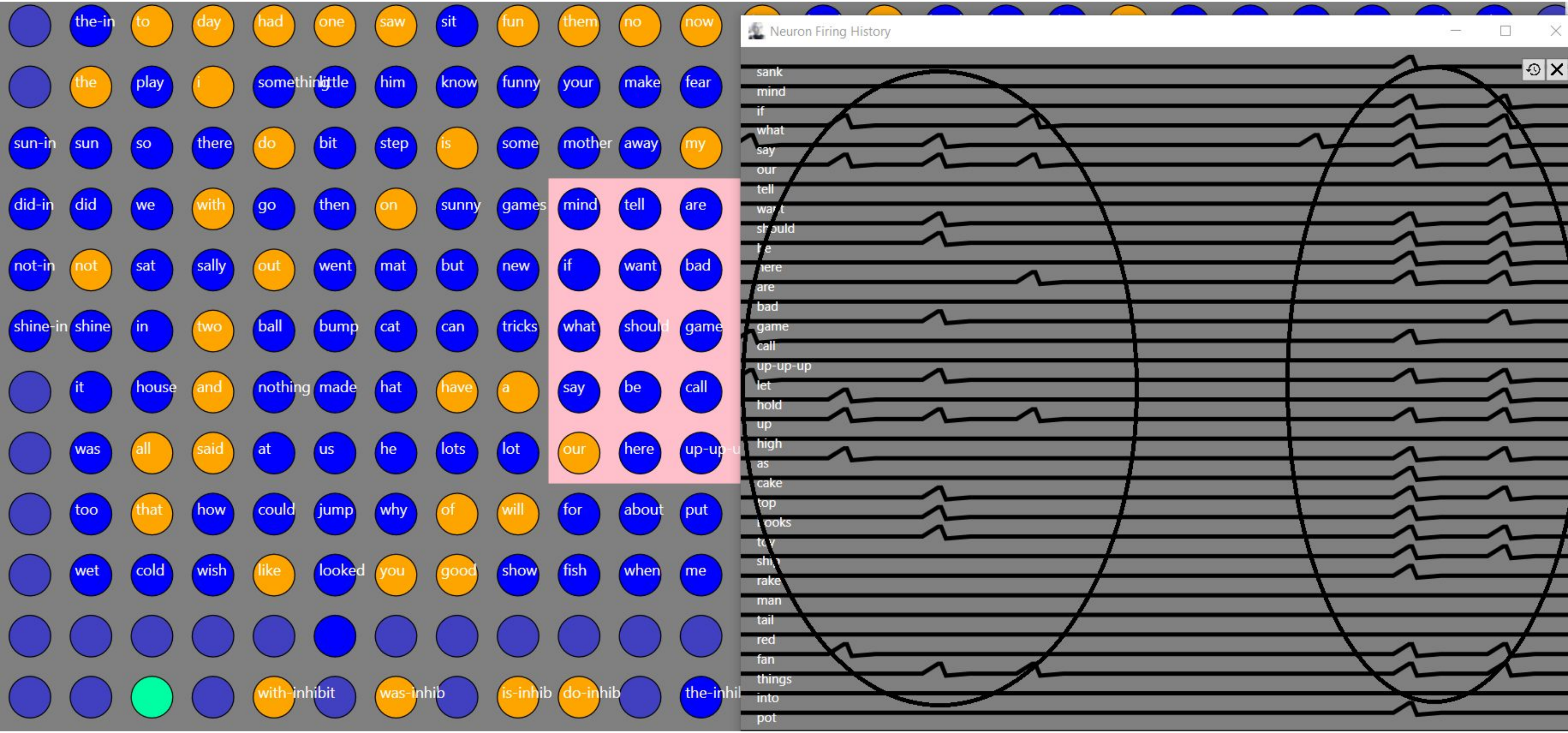
That is because both correspond to shared contexts.

We might seek these synchronously oscillating substitution classes using a raster plot.

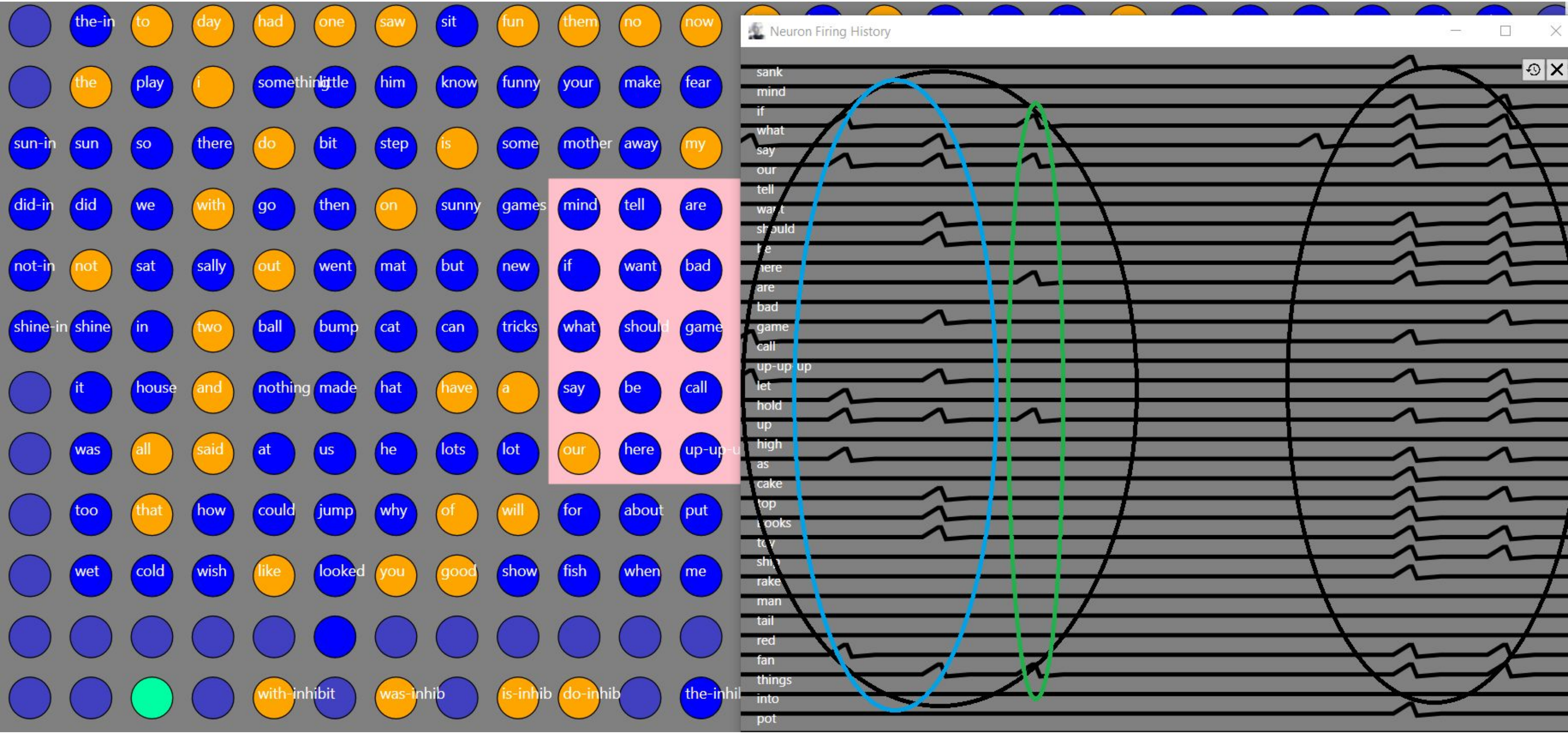
Raster plot shows synchrony = shared context = substitution?



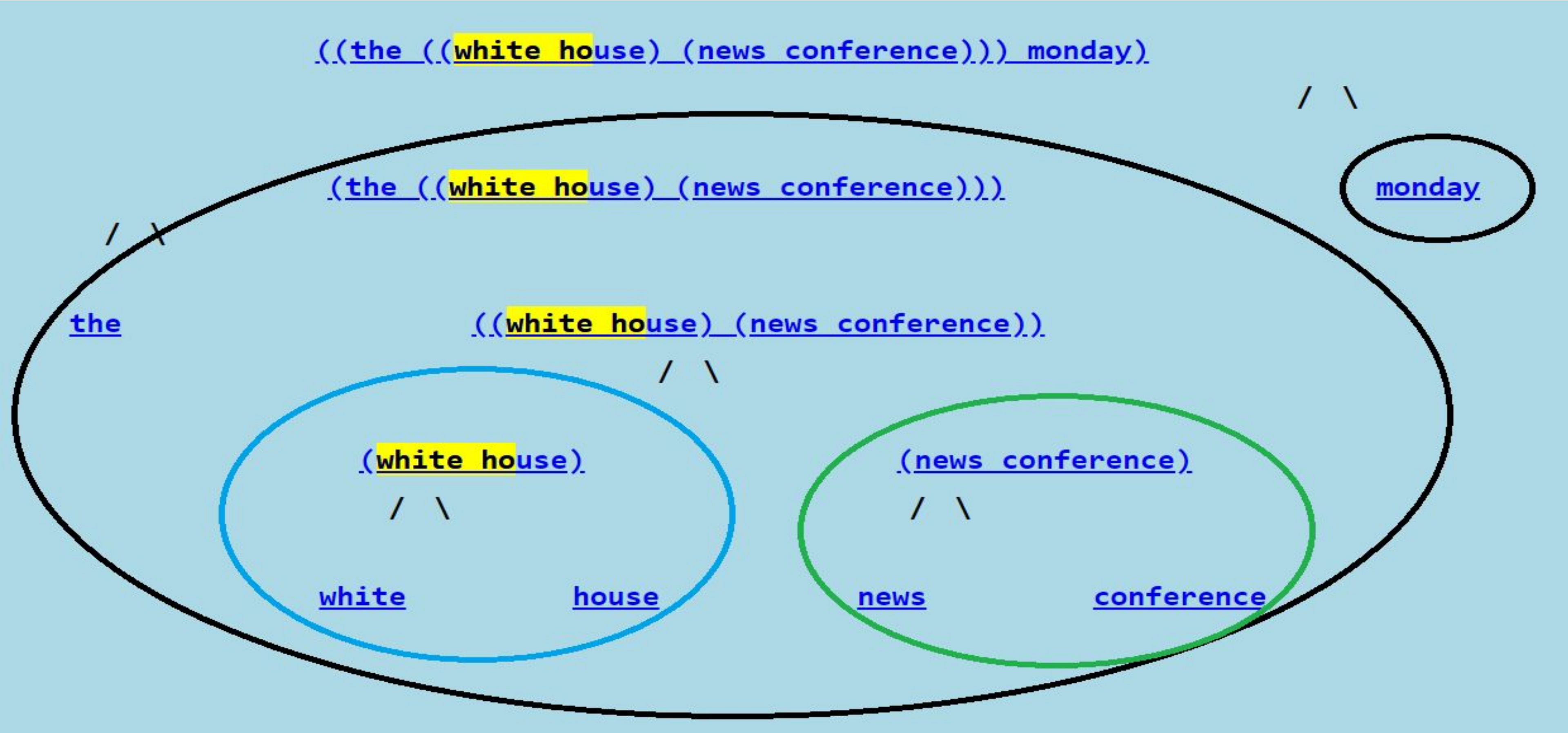
Different synchrony energies correspond to finer grain substitution groupings?



Different energy synchronies define trees?



Comparable to vector formulation substitution trees?



But oscillation substitution mapping is still speculation

Such substitution classes form parse trees in the vector formulation. So we might speculate they can form meaningful parse trees in a sequence network oscillation formulation.

An effective demonstrator will **require a simulator capable of many more nodes**. Preferably natively spiking. (**If anyone has access to large scale spiking hardware (Loihi?), let me know. I want to try this network oscillation structuring at scale.**)