COSC421: Neural Models of Language

Lecture 5: Working memory for episodes and phonological material

Alistair Knott

Dept. of Computer Science, University of Otago

Recap

In Lectures 1–3, I introduced a model of the SM processes involved in 'experiencing' an agent grabbing a cup.

In Lecture 4, I argued:

- *Perception* of a reach-to-grasp action involves a sequence of three sensorimotor operations.
- *Execution* of a reach-to-grasp action involves a very similar sequence.

The sequence for perception of a reach-to-grasp

The sequence for perception of a reach-to-grasp

Operation 1	*O* attends to an external agent, configuring his mirror system circuit for action perception.

The sequence for perception of a reach-to-grasp

Operation 1	*O* attends to an external agent, configuring his mirror system circuit for action perception.
State 1	*O* receives reafferent feedback from this operation; the percept 'man'.

The sequence for perception of a reach-to-grasp

Operation 1	*O* attends to an external agent, configuring his mirror system circuit for action perception.
State 1	*O* receives reafferent feedback from this operation; the percept 'man'.
Operation 2	*O* establishes joint attention with the agent, and attends to another object (the cup).

The sequence for perception of a reach-to-grasp

Operation 1	*O* attends to an external agent, configuring his mirror system circuit for action perception.
State 1	*O* receives reafferent feedback from this operation; the percept 'man'.
Operation 2	*O* establishes joint attention with the agent, and attends to another object (the cup).
State 2	*O* receives feedback from this operation: the percept 'cup'.

The sequence for perception of a reach-to-grasp

Operation 1	*O* attends to an external agent, configuring his mirror system circuit for action perception.
State 1	*O* receives reafferent feedback from this operation; the percept 'man'.
Operation 2	*O* establishes joint attention with the agent, and attends to another object (the cup).
State 2	*O* receives feedback from this operation: the percept 'cup'.
Operation 3	*O* initiates a process of biological motion classification, which results in the action 'grab' being activated in *O*'s premotor cortex.

The sequence for perception of a reach-to-grasp

Operation 1	*O* attends to an external agent, configuring his mirror system circuit for action perception.
State 1	*O* receives reafferent feedback from this operation; the percept 'man'.
Operation 2	*O* establishes joint attention with the agent, and attends to another object (the cup).
State 2	*O* receives feedback from this operation: the percept 'cup'.
Operation 3	*O* initiates a process of biological motion classification, which results in the action 'grab' being activated in *O*'s premotor cortex.
State 3	As a corollary of this process, *O* re-attends to the agent as an animate agent.

The sequence for perception of a reach-to-grasp

Operation 1	*O* attends to an external agent, configuring his mirror system circuit for action perception.
State 1	*O* receives reafferent feedback from this operation; the percept ‘man’.
Operation 2	*O* establishes joint attention with the agent, and attends to another object (the cup).
State 2	*O* receives feedback from this operation: the percept ‘cup’.
Operation 3	*O* initiates a process of biological motion classification, which results in the action ‘grab’ being activated in *O*’s premotor cortex.
State 3	As a corollary of this process, *O* re-attends to the agent as an animate agent.
State 4	*O* sees the agent achieve a stable grasp on the cup, thereby re-attending to it in the haptic modality.

The sequence for execution of a reach-to-grasp

The sequence for execution of a reach-to-grasp

Operation 1	*A* attends to himself, configuring his mirror system circuit for action execution.

The sequence for execution of a reach-to-grasp

Operation 1	*A* attends to himself, configuring his mirror system circuit for action execution.
State 1	*A* receives reafferent feedback that this operation succeeded.

The sequence for execution of a reach-to-grasp

Operation 1	*A* attends to himself, configuring his mirror system circuit for action execution.
State 1	*A* receives reafferent feedback that this operation succeeded.
Operation 2	*A* selects an object to reach for (the cup), and hence executes an action of attention to the cup.

The sequence for execution of a reach-to-grasp

Operation 1	*A* attends to himself, configuring his mirror system circuit for action execution.
State 1	*A* receives reafferent feedback that this operation succeeded.
Operation 2	*A* selects an object to reach for (the cup), and hence executes an action of attention to the cup.
State 2	*A* receives feedback from this operation: the percept 'cup'.

The sequence for execution of a reach-to-grasp

Operation 1	*A* attends to himself, configuring his mirror system circuit for action execution.
State 1	*A* receives reafferent feedback that this operation succeeded.
Operation 2	*A* selects an object to reach for (the cup), and hence executes an action of attention to the cup.
State 2	*A* receives feedback from this operation: the percept 'cup'.
Operation 3	*A* selects an action category ('grab') and begins to execute the grab action.

The sequence for execution of a reach-to-grasp

Operation 1	*A* attends to himself, configuring his mirror system circuit for action execution.
State 1	*A* receives reafferent feedback that this operation succeeded.
Operation 2	*A* selects an object to reach for (the cup), and hence executes an action of attention to the cup.
State 2	*A* receives feedback from this operation: the percept 'cup'.
Operation 3	*A* selects an action category ('grab') and begins to execute the grab action.
State 3	*A* receives feedback from the process of action execution, which includes an action gestalt which he can associate with himself 'as an agent'.

The sequence for execution of a reach-to-grasp

Operation 1	*A* attends to himself, configuring his mirror system circuit for action execution.
State 1	*A* receives reafferent feedback that this operation succeeded.
Operation 2	*A* selects an object to reach for (the cup), and hence executes an action of attention to the cup.
State 2	*A* receives feedback from this operation: the percept 'cup'.
Operation 3	*A* selects an action category ('grab') and begins to execute the grab action.
State 3	*A* receives feedback from the process of action execution, which includes an action gestalt which he can associate with himself 'as an agent'.
State 4	*A* re-establishes the cup in the haptic modality.

Representation of the cup-grabbing event in memory

In this lecture I'll talk about how a cup-grabbing event is represented in **working memory**.

I'll also discuss how *words* are held in working memory—which will be an introduction to linguistic representations in the brain.

Why do we need an account of memory?

Why do we need an account of memory?

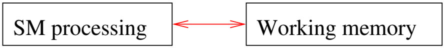
1. The relation between SM experience and language is not *direct*.
   - We don't *have to* talk about SM experiences.
   - We can talk about things *other than* SM experiences.

Why do we need an account of memory?

1. The relation between SM experience and language is not *direct*.
   - We don't *have to* talk about SM experiences.
   - We can talk about things *other than* SM experiences.

Language interfaces with **working memory** (**WM**) representations.

- A SM experience can be stored in working memory.

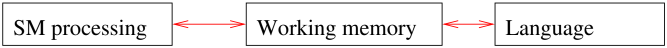


Why do we need an account of memory?

1. The relation between SM experience and language is not *direct*.
   - We don't *have to* talk about SM experiences.
   - We can talk about things *other than* SM experiences.

Language interfaces with **working memory** (**WM**) representations.

- A SM experience can be stored in working memory.
- WM event representations can be 'read out' linguistically.

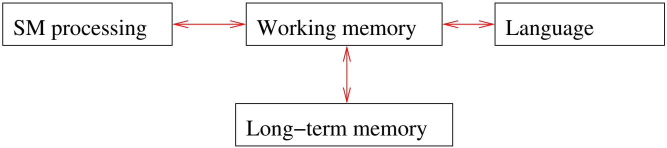


Why do we need an account of memory?

1. The relation between SM experience and language is not *direct*.
   - We don't *have to* talk about SM experiences.
   - We can talk about things *other than* SM experiences.

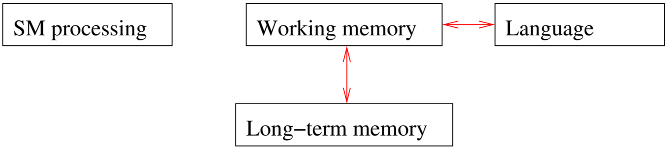
Language interfaces with **working memory** (**WM**) representations.

- A SM experience can be stored in working memory.
- WM event representations can be 'read out' linguistically.
- We can also retrieve events from **long-term memory** into WM.



Why do we need an account of memory?

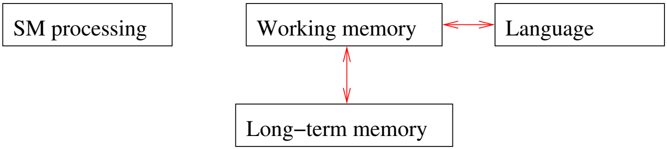
2. We must make reference to memory when describing particular syntactic constructions.



Why do we need an account of memory?

2. We must make reference to memory when describing particular syntactic constructions.

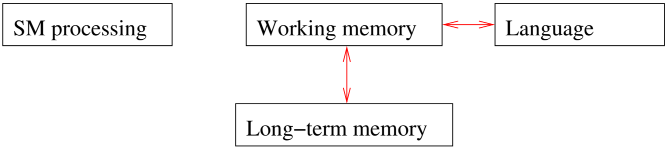
- E.g. **assertions** encode something in LTM.



Why do we need an account of memory?

2. We must make reference to memory when describing particular syntactic constructions.

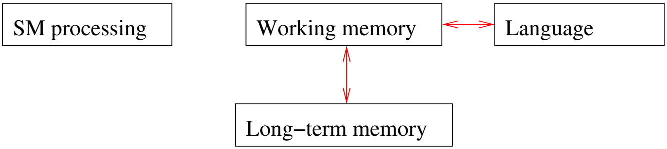
- E.g. **assertions** encode something in LTM.
- E.g. **questions** retrieve something from LTM.



Why do we need an account of memory?

2. We must make reference to memory when describing particular syntactic constructions.

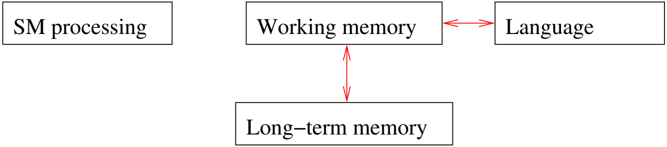
- E.g. **assertions** encode something in LTM.
- E.g. **questions** retrieve something from LTM.
- E.g. **indefinite NPs** (*a cat*) encode something in WM.



Why do we need an account of memory?

2. We must make reference to memory when describing particular syntactic constructions.

- E.g. **assertions** encode something in LTM.
- E.g. **questions** retrieve something from LTM.
- E.g. **indefinite NPs** (*a cat*) encode something in WM.
- E.g. **definite NPs** (*the cat*) retrieve something in WM.



Lecture overview

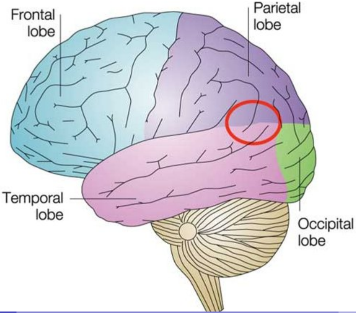
- Working memory (WM) and long-term memory (LTM)
- The different forms of WM: visuospatial, phonological and semantic
- A simple model of phonological WM by Burgess and Hitch (1999)
- Semantic WM representations
  - Baddeley's **episodic buffer**
  - Representations of **planned action sequences** in WM

Working memory and long-term memory

There is a well-established distinction between working memory (WM) and long-term memory (LTM) in psychology.

The basic idea:

- WM involves **frontal cortex** and the temporoparietal junction.

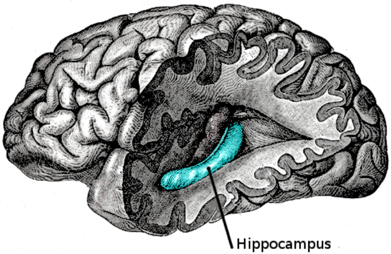


Working memory and long-term memory

There is a well-established distinction between working memory (WM) and long-term memory (LTM) in psychology.

The basic idea:

- WM involves **frontal cortex** and the temporoparietal junction.
- LTM involves the **hippocampus**, and adjacent regions of temporal cortex.



Working memory and long-term memory

Neuropsychology.

- Patients with damage to the hippocampus (& associated cortical areas) show impaired LTM with normal WM.
  Patient **HM** had his hippocampal region removed bilaterally.
  - He had **anterograde amnesia**: couldn't form new memories.
  - But he could still carry on a conversation, play chess, repeat phone numbers.
- Patients with damage to the frontal lobes (sometimes) have the reverse pattern.
  Patient **KF** (Shallice and Warrington, 1970) had damage to his left temporoparietal cortex.
  - He had very bad phonological STM. (E.g. a digit span of 1.)
  - But his ability to store events in LTM was intact.

Working memory and long-term memory

Behavioural experiments on normal subjects.

A classic WM task: **immediate serial recall** (**ISR**) of a list of stimuli.

There's good evidence subjects encode stimuli in ISR experiments *phonologically*—i.e. as sounds, rather than as meanings.

- **Phonological similarity effects**: *ba ga da* is harder than *bo ga di*
- **Word length effects**: *kitten collie rooster* is harder than *cat dog chick*

In LTM experiments, stimuli are encoded as meanings, not sounds.

- *cat* is not confused with *bat*
- *cat* is confused with *kitten*

Working memory and long-term memory

Neural mechanisms.

WM is implemented in two ways.
1. The *activity levels* of cells.

- To store a pattern of activation in a group of neurons, you *keep it active*.

2. Short-term synaptic connections between neurons.
(Connections that fade over 10/20 seconds.)

- To store a pattern of activation in a group of neurons, you strengthen the synaptic connections between these neurons.

LTM is implemented in 'long-term' synaptic connections between neurons.

Working memory and long-term memory

Summary:

	WM	LTM
Duration	seconds/minutes	days/years
Stored in	frontal cortex/TPJ	hippocampus ($\rightarrow$cortex)
Represented as	sounds **&. . .**	meanings
Implemented as	neural activity / short-term synaptic connection strengths	long-term synaptic connection strengths

Episodic and semantic LTM

There are actually two forms of LTM.

Episodic memory: memory for specific episodes in an agent's life.

- E.g. 'Yesterday, John grabbed this cup here'.

Semantic memory: memory for generic facts.

- E.g. 'Cups tend to have handles'.

I'll focus on episodic memory, because that's how a (single) cup-grabbing event would be represented.

(There's also **procedural memory**, which is LTM for *skills*. A lot of learning in the motor system is of this type.)

Definitions of ‘working memory’

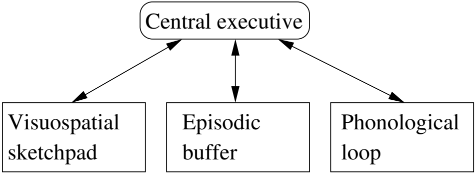
The best known model of WM: Baddeley and Hitch (1974) (updated by Baddeley, 2000).

Baddeley’s definition of WM: a short-term store which subserves ‘cognitive’ operations: language processing, reasoning, learning.

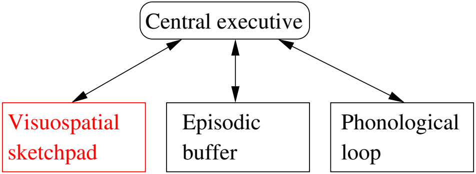
The term ‘working memory’ is also used in sensorimotor psychology, to refer to an animal’s **prepared actions** or **task set**.

Baddeley wants to keep these two senses of WM separate.
I’ll look at both types of WM.

Baddeley's model of WM



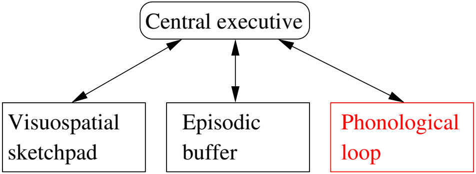
Baddeley's model of WM



The visuospatial sketchpad: a working memory for **visual patterns.**

- E.g. remembering a shape, so you can recognise it later. (Probably involves IT and its interface with PFC)
- E.g. remembering where you saw something. (We won't be looking at this.)
- Patterns can be spatially complex, but not temporally complex.

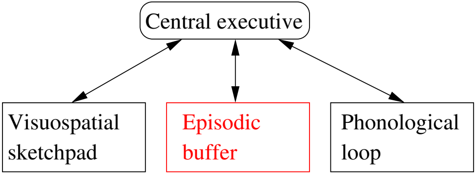
Baddeley's model of WM



The phonological loop: holds a short sequence of **words or phonemes**.

- Items in the phonological loop are stored as *sounds*.
  - Phonological similarity / word length effects (already mentioned)
- Phonological sequences need to be *rehearsed* to be retained, as shown by studies of **articulatory suppression**.
  - It's harder to recall a phon. sequence if you have to say *the the the* during the delay period. (Try it. . .)

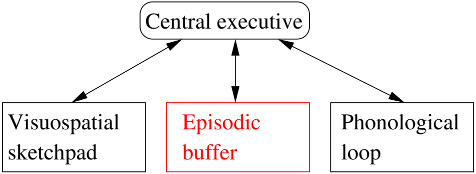
Baddeley's model of WM



The episodic buffer: a special form of storage for 'episodes'.

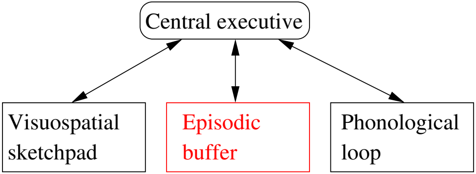
1. This form of storage is **semantic**. Some evidence for it:
   - Sentences are easier to recall than sequences of unrelated words (Baddeley *et al.*, 1987).
   - Amnesic patients can retain the 'gist' of a paragraph of around 15 propositions for a short period (Wilson & Baddeley, 1988).
   - Amnesic patients can reason, solve problems, play bridge…

Baddeley's model of WM



2. The episodic buffer can interface with the phonological loop.
   - A simple example: **chunking**.
     - Four two-syllable words are easier to recall than eight one-syllable words (Hulme *et al.*, 1991)
     - Model: the phonological buffer stores a sequence of **pointers** to semantic items held 'in a separate WM buffer'.
   - Baddeley: our improved WM for sentences / paragraph gist is due to items held in this same buffer.

Baddeley's model of WM

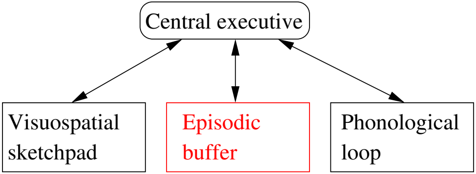


3. The episodic buffer holds representations which integrate **sensory**, **semantic** and **phonological** information, and which are maintained through rehearsal.

Baddeley and Andrade (2000): subjects shown stimuli varying in meaningfulness (high/low) and in modality (visual/phonological) and asked to rate their ‘vividness’ after an interim distractor task.

- Meaningful stimuli were rated as more vivid.
- There was a modality-specific effect of distractor task on rated vividness, which didn't interact with meaningfulness.

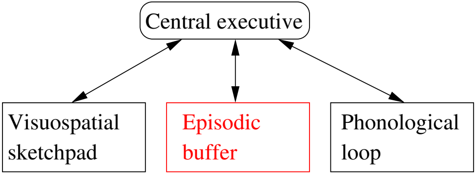
Baddeley's model of WM



3. The episodic buffer holds representations which integrate **sensory**, **semantic** and **phonological** information, and which are maintained through rehearsal.

- Low/high meaning visual stimuli ((jumbled) photos)
  - Visual distractor task: shapes *lose* vividness
  - Phonological distractor task: shapes *retain* vividness
- Low/high meaning phonological stimuli ((scrambled) sentences)
  - Visual distractor task: tones *retain* vividness
  - Phonological distractor task: tones *lose* vividness

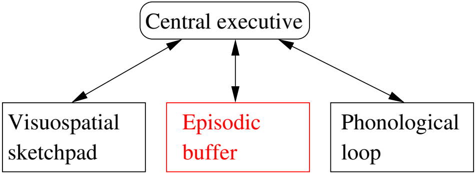
Baddeley's model of WM



3. The episodic buffer holds representations which integrate **sensory**, **semantic** and **phonological** information, and which are maintained through rehearsal.

- There must be a WM medium where we represent stimuli *semantically*. (Because meaning affects vividness.)
- The medium must be maintained through rehearsal. (Because distractor tasks affect storage.)
- The medium must interface with phonological/visual modalities. (Because distractor effects are modality specific.)

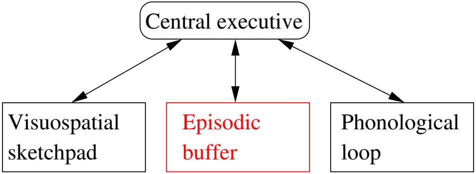
Baddeley's model of WM



4. The episodic buffer plays a role in the storage of **episodic memories.**

- Experienced episodes are initially stored in the episodic buffer.
- From there they are relayed to longer-term storage.

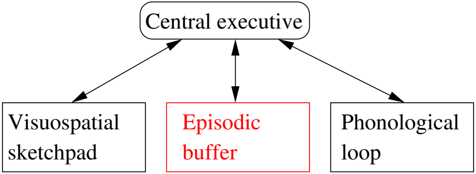
Baddeley's model of WM



5. Material in the episodic buffer is rehearsed by a process of 'sequential attention'.

- Rehearsal in the phonological buffer involves 'producing' each item in the sequence.
- There must be a way in which items in the episodic buffer are sequentially 'produced'.

Baddeley's model of WM

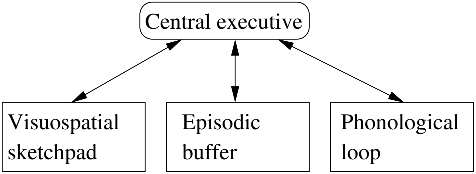


6. The episodic buffer is probably implemented in a network involving **frontal cortex.**

fMRI study by Prabhakaran *et al.* (2000):

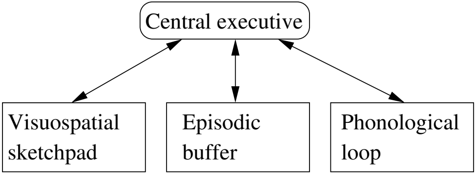
- A task requiring retention of integrated verbal and spatial information activates a right frontal area.
- More posterior areas are activated by tasks requiring retention of unintegrated material.

Baddeley's model of WM



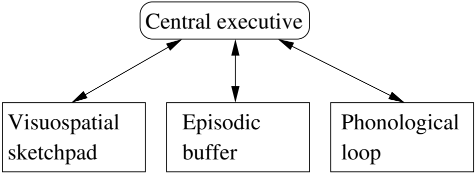
- There are good models of phonological WM.

Baddeley's model of WM



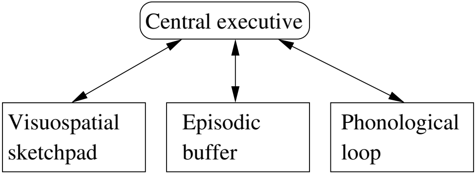
- There are good models of phonological WM.
- There are good models of prepared action sequences in WM.

Baddeley's model of WM



- There are good models of phonological WM.
- There are good models of prepared action sequences in WM.
- There are *no* good models of the episodic buffer.

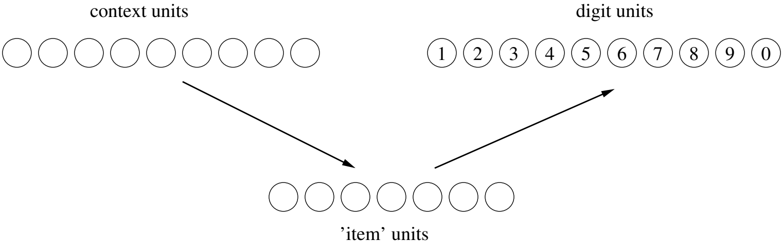
Baddeley's model of WM



- There are good models of phonological WM.
- There are good models of prepared action sequences in WM.
- There are *no* good models of the episodic buffer.
- My proposal: if episodes are experienced as SM sequences, then they can be stored in the episodic buffer as *prepared SM sequences*.

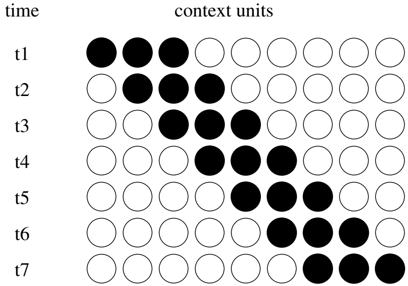
Burgess and Hitch's model of the phonological loop

Burgess and Hitch (1999) proposed a model of phonological working memory.This model is what you're looking at in Assignment 1. Here's a simplified version:



Burgess and Hitch's model of the phonological loop

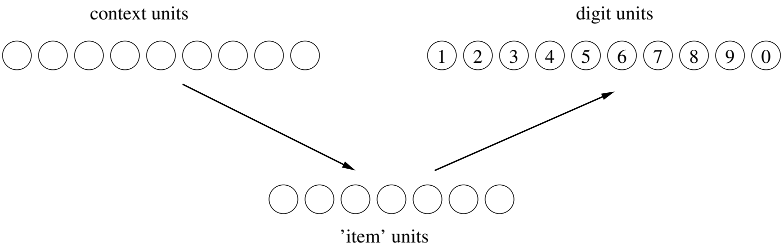
The context units are hardwired to step through a sequence of patterns, as shown here:



Note: context representations use **population coding.**
(Successive contexts are represented by overlapping patterns.)

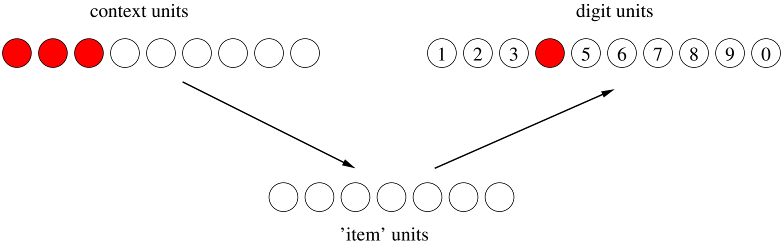
Burgess and Hitch's model of the phonological loop

Here's how it works.



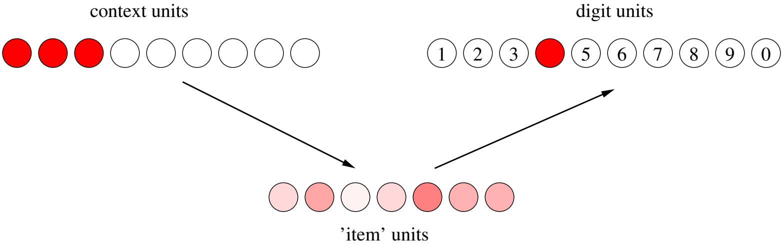
Burgess and Hitch's model of the phonological loop

We present the first digit, in the first context.



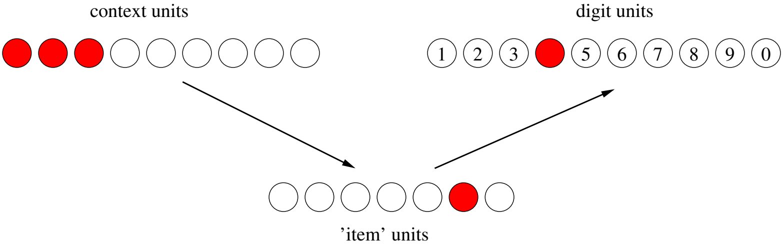
Burgess and Hitch's model of the phonological loop

We evoke a pattern of activity in the 'item' units.



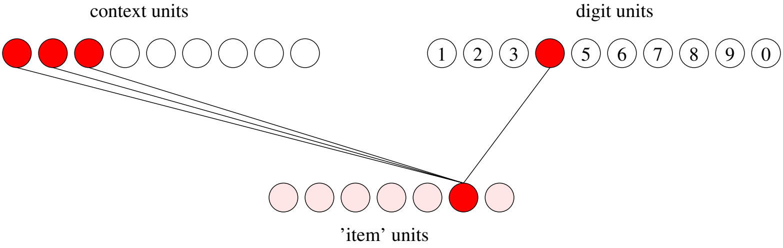
Burgess and Hitch's model of the phonological loop

Competition in the **item** units sharpens this pattern, so there's a clear winner.



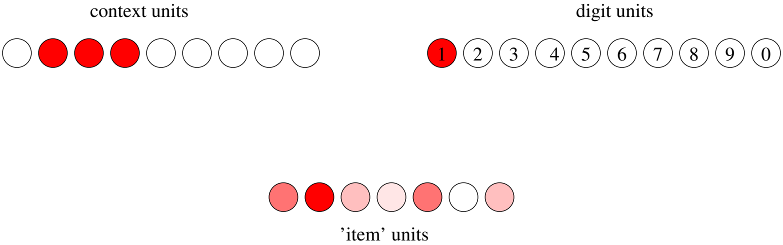
Burgess and Hitch’s model of the phonological loop

We strengthen associations between all active units. (Now the winning **item** unit maps the current context onto the currrent digit.)



Burgess and Hitch's model of the phonological loop

Then we *inbibit* the winning **item** unit, and move to the next context.



WM representations of prepared actions

Behavioural psychologists often use the term WM to refer to the place where an animal holds its **current task set.**

- In many circumstances, an animal maintains a set of prepared actions, or a set of prepared responses to stimuli.
- These are assumed to be held in 'working memory'.

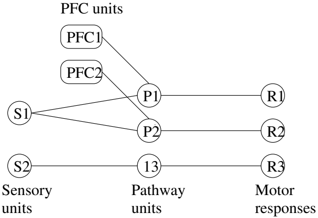
Recall from Lecture 3: Task set and action preparation involve **PFC**.

- PFC imposes top-down biases on *attentional operations*
- PFC imposes top-down biases on *motor actions*.

Evidence includes the Wisconsin card-sorting task, the Stroop task. . .

Miller and Cohen's model of PFC

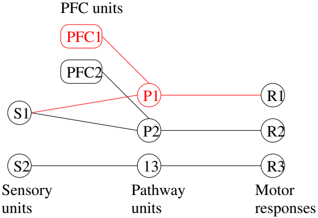
Here's a well-known model of how PFC implements cognitive set, by Miller and Cohen (2001).



- There are many different **pathways** from stimuli to motor responses. (Visual→parietal→premotor→motor cortices.)
- **Intermediate units** in these pathways compete with one another.
- PFC units can bias this competition towards one pathway or another.

Miller and Cohen's model of PFC

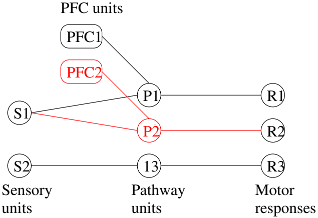
Here's a well-known model of how PFC implements cognitive set, by Miller and Cohen (2001).



If the PFC1 assembly is active, this biases the agent towards responding to S1 with R1.

Miller and Cohen's model of PFC

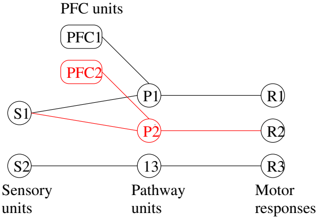
Here's a well-known model of how PFC implements cognitive set, by Miller and Cohen (2001).



If the PFC2 assembly is active, this biases the agent towards responding to S1 with R2.

Miller and Cohen's model of PFC

Here's a well-known model of how PFC implements cognitive set, by Miller and Cohen (2001).



If the PFC2 assembly is active, this biases the agent towards responding to S1 with R2.

- Since pathway units compete with each other, either bias makes the agent tend to *ignore* S2.

WM representations of prepared action sequences

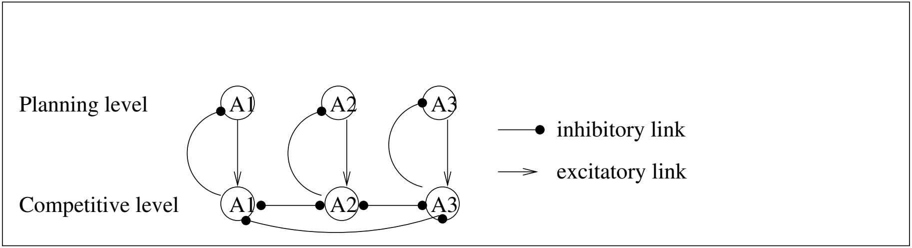
Recall: our cup-grabbing action is actually a *sequence* of actions. How can an agent prepare a sequence?

There is good evidence that PFC is involved in planning action sequences.

- Lesion studies in monkeys (e.g. Petrides, 1991) and humans (Petrides and Milner, 1982)
- PFC cells in monkeys sensitive to specific prepared sequences (e.g. Barone and Joseph, 1989)

An influential model of sequence preparation in PFC is called **competitive queueing** (see Grossberg, 1978; Houghton, 1995; Rhodes *et al.*, 2004).

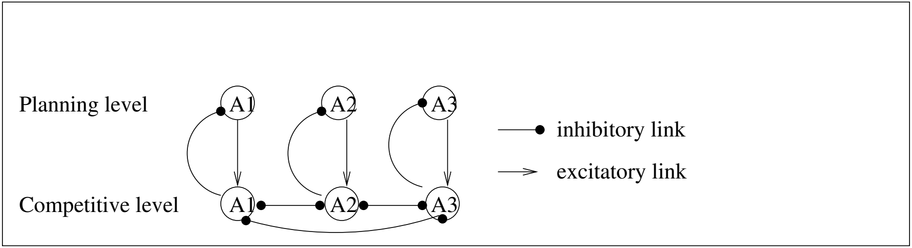
Competitive queueing



Each action is represented at a **planning level** and a **competitive level**.

- A sequence plan is a gradient of activation in the planning level.
- This gradient is passed to the competitive level, where the most active action is selected and executed.
- The winning action inhibits its counterpart in the planning level, and the next-most-active action is the next to win. (And so on.)

Competitive queueing



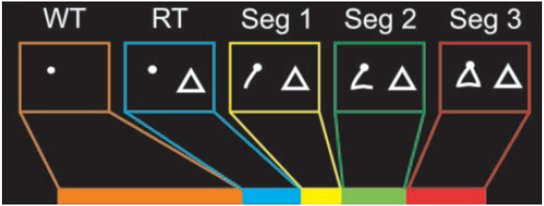
There is good evidence for competitive queueing.

- It provides a good account of **reversal errors**, where the order of two successive actions is swapped.
- Averbeck *et al.*, 2002 found PFC cells that behave exactly like actions in the planning level.

Averbeck *et al.*'s experiment

Averbeck *et al.* (2002) trained monkeys to draw a number of different shapes in response to cue stimuli.

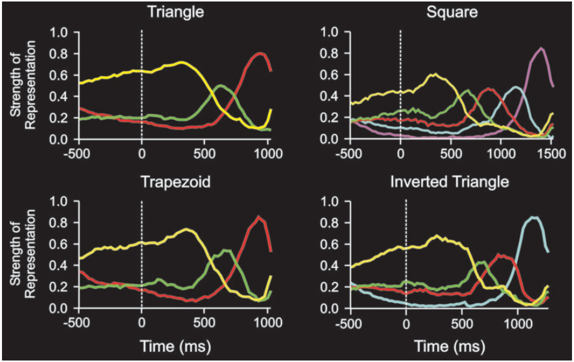
- Drawing each shape involved a sequence of motor movements.
- After the cue appeared, there was a delay before the monkey could begin to draw.



- PFC cells were recorded during the wait and drawing periods.

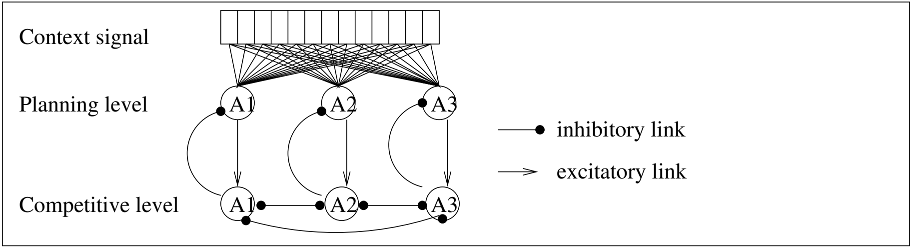
Averbeck *et al.*'s experiment

Different PFC cells were sensitive to different movements.



- Note: the *activity level* of cells during the delay period encodes the *order* in which movements will occur.
- That's exactly what competitive queueing predicts.

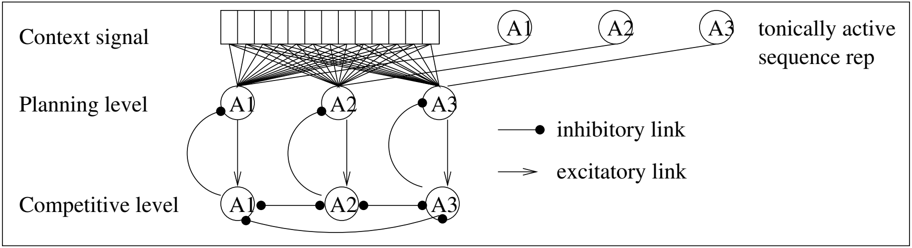
Extensions to competitive queueing



CQ models cannot deal with **repeated actions** (e.g. A1, A1, A2).

- To allow repeated actions, CQ models are often augmented with an extra **context signal**, which evolves independently in time (see e.g. Burgess and Hitch, 1999).

Extensions to competitive queueing



CQ models cannot deal with **repeated sequences**, since plans are destructively updated as a plan is executed.

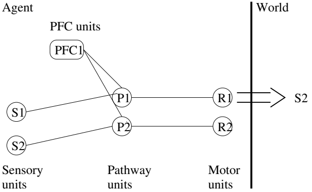
- To fix this, we can introduce a **tonically active** version of the planned sequence, which can restore the planning level pattern.
- Averbeck and Lee (2007) find cells in PFC which encode a planned sequence in the intervals between repeated executions.

Varieties of context signal

The context signal described above evolves as a function of time. But some actions take an unpredictable amount of time.

Another model of sequence preparation is based on Miller and Cohen's pathway-biasing model of PFC.

- Assume some stimuli are **reafferent consequences** of the agent's own actions.
- Biasing pathways from these stimuli to other actions effectively prepares action sequences.



Executing a planned action sequence: summary

It is likely that planned sequences involve a mixture of CQ and pathway-biasing mechanisms.

In either case, when an agent is executing a planned sequence:

- There will be a **sustained signal**, representing all the actions in the selected plan simultaneously.
- There will be a sequence of **transient signals**, interleaving actions and their reafferent consequences.

Action1-then-Action2 plan	*Action1*
↓	*SensoryConsequence*
↓	*Action2*
↓	*SensoryConsequence*

Replaying PFC plans: simulation mode

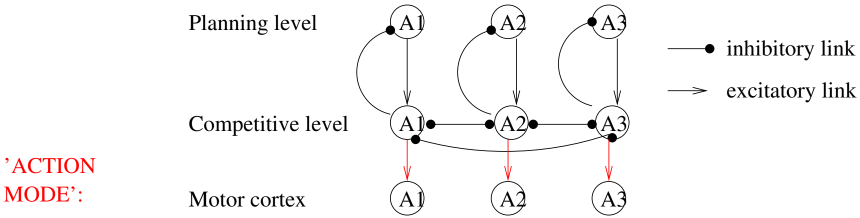
There is some evidence that agents can ‘internally replay’ stored sensorimotor sequences.

- This comes from work on **place cells** in the hippocampus.

Imagine a **simulation mode**, in which a prepared SM sequence can be internally replayed, with no external effects.

Simulation mode

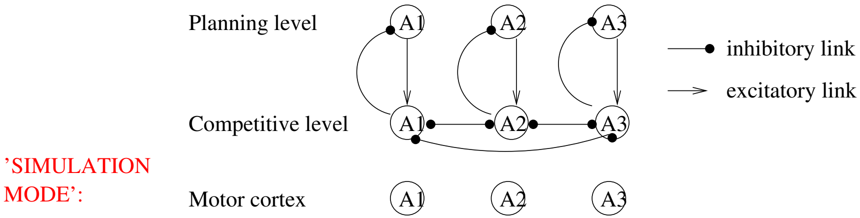
Simulation is easy in a competitive queueing model.



- If you **execute** a prepared action sequence, each winning action activates an action in the motor cortex.

Simulation mode

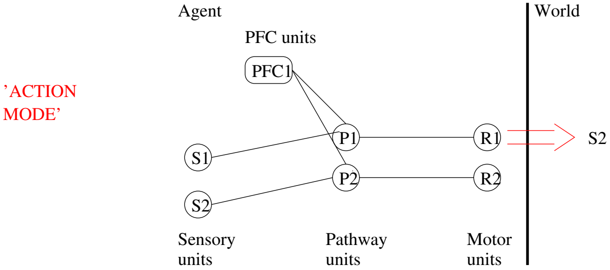
Simulation is easy in a competitive queueing model.



- If you **execute** a prepared action sequence, each winning action activates an action in the motor cortex.
- If you **simulate** a prepared action sequence, you just need to switch off the links to the motor cortex.

Simulation mode

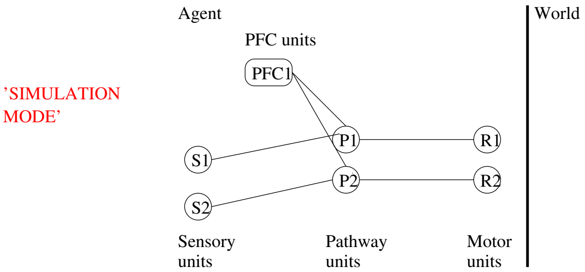
In an associative chaining model, simulation is a bit more tricky.



- We have to switch off links to overt actions (as for CQ).

Simulation mode

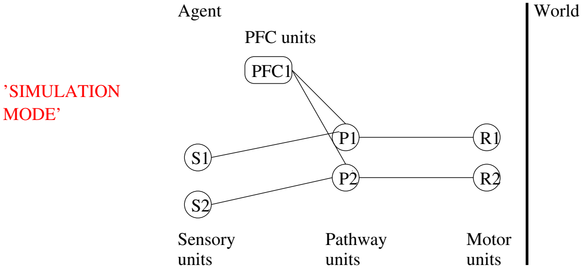
In an associative chaining model, simulation is a bit more tricky.



- We have to switch off links to overt actions (as for CQ).

Simulation mode

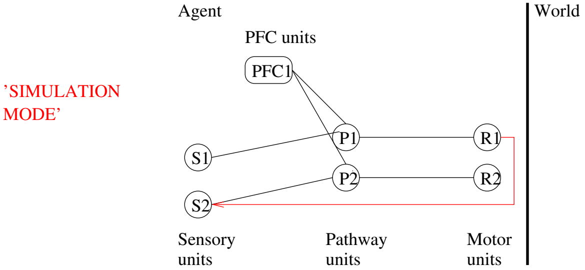
In an associative chaining model, simulation is a bit more tricky.



- We have to switch off links to overt actions (as for CQ).
- We must also find a way for a simulated action to trigger its *own reafferent consequence*.

Simulation mode

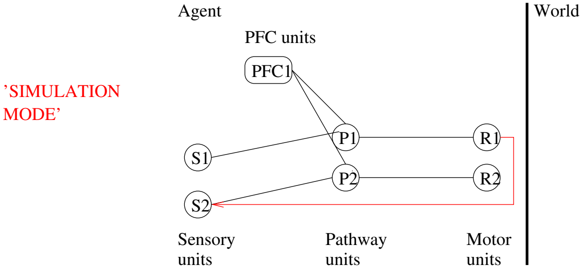
In an associative chaining model, simulation is a bit more tricky.



- We have to switch off links to overt actions (as for CQ).
- We must also find a way for a simulated action to trigger its *own reafferent consequence*.

Simulation mode

In an associative chaining model, simulation is a bit more tricky.



- We have to switch off links to overt actions (as for CQ).
- We must also find a way for a simulated action to trigger its *own reafferent consequence*.
  There are ways this could be done, both for motor actions and for attentional actions.

The episodic buffer revisited

A stored sensorimotor sequence plan which supports internal replay sounds a lot like Baddeley's episodic buffer.

- It accesses representations in several SM modalities
- It has the form of a sequence, which can be 'rehearsed'
- It buffers observed actions as well as executed actions
- It is implemented in frontal areas.

Proposal: **episodes are stored in WM as prepared SM sequences.**

Summary

- An event is *experienced* as a *SM sequence*.
- It is stored in *working memory* (in PFC) as a *planned SM sequence*.
- PFC storage probably involves a mixture of competitive queueing, context-based and associative chaining mechanisms.
- The planned SM sequence can be *internally replayed*.

I will argue that generating a sentence reporting a given episode involves replaying a planned SM sequence in working memory, in a special mode where SM signals can activate phonological signals.

Replaying a grasp episode: timecourse of SM signals

Sustained PFC signal	Transient signals		
	Context signals	Action signals	Reafferent signals
$plan_{attend\_agent\,/\,attend\_cup\,/\,grasp}$ ↓ ↓	C_1	*attend_agent*	*attending_to_agent*
$plan_{attend\_agent\,/\,attend\_cup\,/\,grasp}$ ↓ ↓	C_2	*attend_cup*	*attending_to_cup*
$plan_{attend\_agent\,/\,attend\_cup\,/\,grasp}$ ↓ ↓	C_3	*grasp*	*attending_to_agent*
$plan_{attend\_agent\,/\,attend\_cup\,/\,grasp}$ ↓ ↓	C_4		*attending_to_cup*