Abstract models of cognitive flexibility

Kai Krueger – Peter Dayan

Gatsby Computational Neuroscience Unit, UCL, UK

Princeton, November 30, 2009

Introduction

- Cognitive flexibility
  - perform large varieties of actions
  - rapidly adapt to new situations

Introduction

- Cognitive flexibility
  - perform large varieties of actions
  - rapidly adapt to new situations

- Full state-space is immense
  - in general, need all current and previous percepts
  - impossible for learning by simple trial and error

Introduction

- Cognitive flexibility
  - perform large varieties of actions
  - rapidly adapt to new situations
- Full state-space is immense
  - in general, need all current and previous percepts
  - impossible for learning by simple trial and error
- Need to exploit available structure
  - reduce state-space to a manageable size (independecies)
  - basis functions (task competencies)
  - hierarchies

Beneficial structure is available...

- Learning through instruction
  - full structure provided by environment

Beneficial structure is available...

- Learning through instruction
  - full structure provided by environment

- Coping with complexity through sequential learning
  - tackle progressively harder tasks (shaping)
  - initial simplification through neglect (auto-shaping)

Beneficial structure is available...

- Learning through instruction
  - full structure provided by environment
- Coping with complexity through sequential learning
  - tackle progressively harder tasks (shaping)
  - initial simplification through neglect (auto-shaping)
- Flexibility by abstraction
  - orthogonalize rules and symbols
  - variable instantiations

Beneficial structure is available...

- Learning through instruction
  - full structure provided by environment
- Coping with complexity through sequential learning
  - tackle progressively harder tasks (shaping)
  - initial simplification through neglect (auto-shaping)
- Flexibility by abstraction
  - orthogonalize rules and symbols
  - variable instantiations

Beneficial structure is available...

- Learning through instruction
  - full structure provided by environment
- Coping with complexity through sequential learning
  - tackle progressively harder tasks (shaping)
  - initial simplification through neglect (auto-shaping)
- Flexibility by abstraction
  - orthogonalize rules and symbols
  - variable instantiations

Outline

Outline

Shaping

Something

"A method of successive approximations" (Skinner 1938)

Ubiquitously in all animal experiments

Shaping

Something

"A method of successive approximations" (Skinner 1938)
Ubiquitously in all animal experiments

Properties of shaping:

- external alteration of stimuli and rewards
- withdrawal of intermittent rewards

Shaping

Something

"A method of successive approximations" (Skinner 1938)
Ubiquitously in all animal experiments

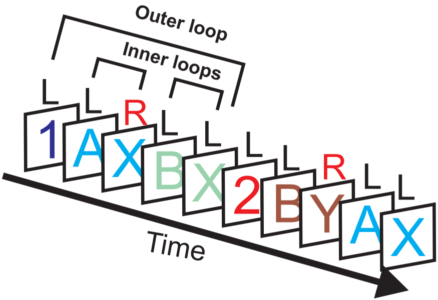
Properties of shaping:

- external alteration of stimuli and rewards
- withdrawal of intermittent rewards

Effects of shaping:

- Creates behavioural units
  - e.g. lever pressing of a rat
- Separate out time scales / branching points
- Create state spaces representations by learning independencies

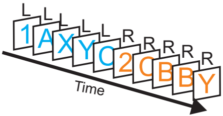
The 12-AX task



- Sequential hierarchical decision making task
- Outer loop: present 1 or 2, followed by 1 – 4 inner loops
- Inner loop: `A`, `B`, `C` followed by `X`, `Y`, `Z`
- Target sequence `AX` in context `1`, `BY` in context `2`
- Introduced by Frank et al. 2001 (to test gating)

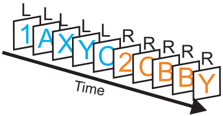
Defining a sequence for shaping

Remembering "1" and "2"
Left response following "1", Right response following "2"

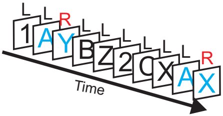


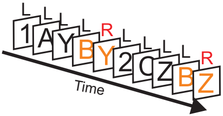
Defining a sequence for shaping

Remembering "1" and "2"
Left response following "1", Right response following "2"

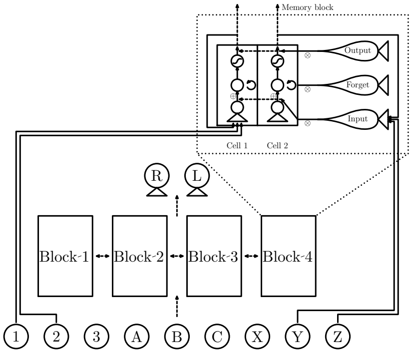


1-back for targets A / B



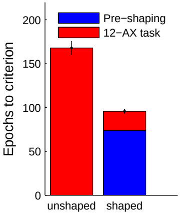


Long-Short Term Memory (LSTM) Network



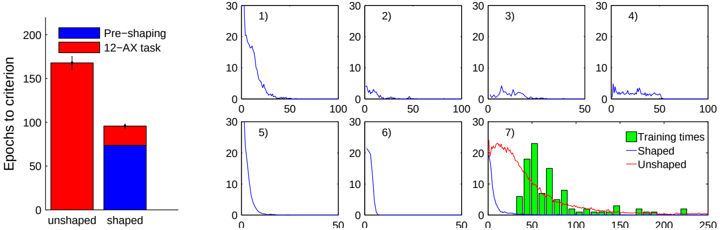
- Abstraction of PFC: Provides working memory, gating
- Easily modularized by adding / enabling memory blocks
- Hochreiter and Schmidhuber (1999)

Simple shaping



- Improvement in learning times:
  - 8 fold decrease for 12-AX alone
  - still better overall including training

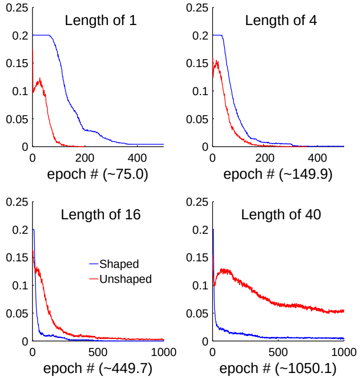
Simple shaping



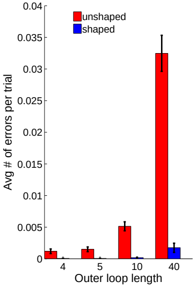
- Improvement in learning times:
  - 8 fold decrease for 12-AX alone
  - still better overall including training
- Need 4 stages for training 1 and 2
- High variance in shaping times

Shaping: when is it useful?

- Can shaping reduce scaling of learning time with task complexity?
- One aspect of complexity: temporal credit assignment
  - increase the outer loop length
    ⇒ higher complexity
- Results:
  - training time still increases, but scales slower
  - increasing complexity ⇒ shaping more importantly



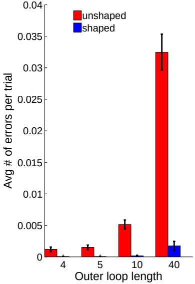
Rule abstraction

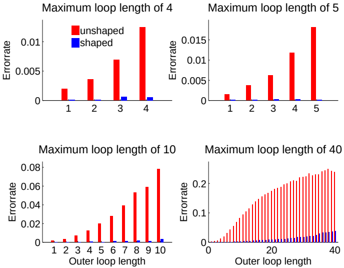


After training on 12-AX with loop length 4

- Flexibility to cope with change in statistics
- Should perform perfectly as rules haven't changed
- Shaping made network more robust to changes

Rule abstraction



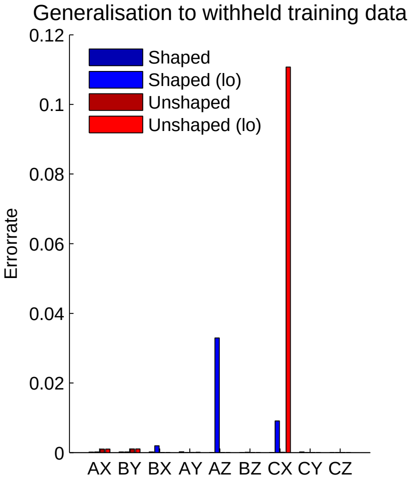


After training on 12-AX with loop length 4

- Flexibility to cope with change in statistics
- Should perform perfectly as rules haven't changed
- Shaping made network more robust to changes

Generalising to unseen data

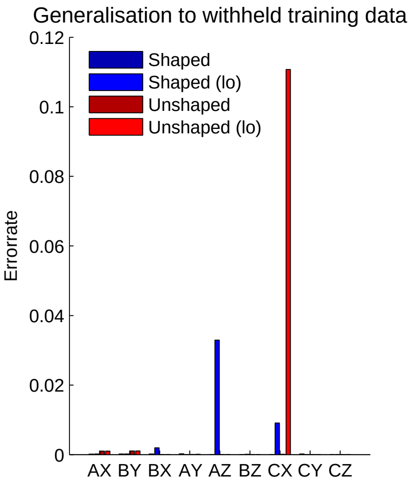
- During training never present `AZ` or `CX`
- Analyze error patterns to left out data



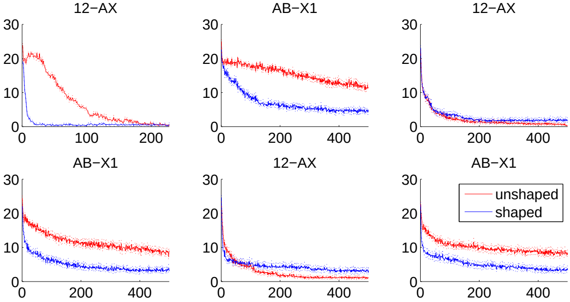
Generalising to unseen data

- During training never present `AZ` or `CX`
- Analyze error patterns to left out data

- Ambiguous generalisation
- Differences in emphasis:
  - shaped: positive response after `A`
  - unshaped: positive response to `X`



Reversal



- Reverse stimulus - rule association (repeat after 500 epochs)
  - shape all components needed
- Learn representations to rapidly switch
  - unshaped: mostly fails
  - shaped: succeeds more often

Making shaping work

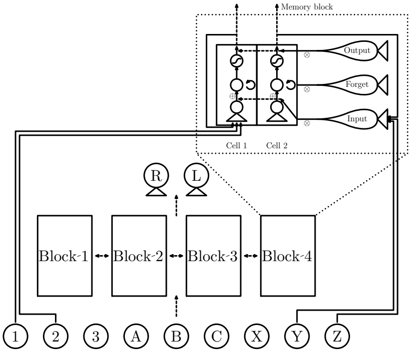
- Can any learning architecture benefit from shaping?
  - no! Need a mechanism of allocation

Making shaping work

- Can any learning architecture benefit from shaping?
  - no! Need a mechanism of allocation

Allocation

- For each new task: Create and activate a new memory module
- Disable learning in previous modules (until full 12-AX)
- Force separation of learning

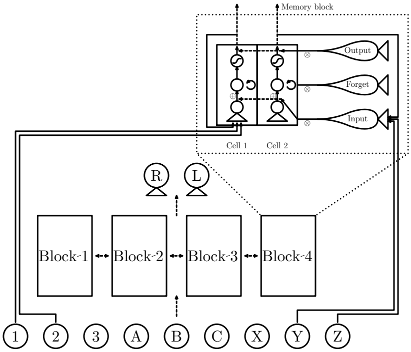


Making shaping work

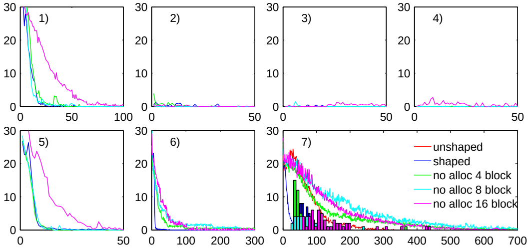
- Can any learning architecture benefit from shaping?
  - no! Need a mechanism of allocation

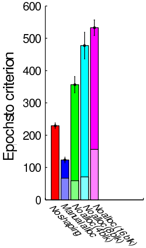
Allocation

- For each new task: Create and activate a new memory module
- Disable learning in previous modules (until full 12-AX)
- Force separation of learning
- Allocation initially done by hand. (Finger of God)



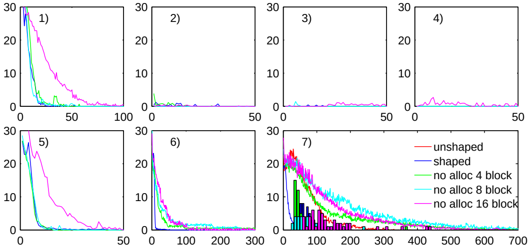
Without memory allocation:

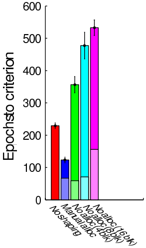




- No allocation harms performance
- Critical interference

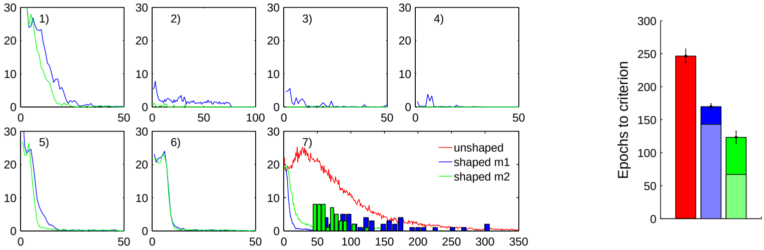
Without memory allocation:





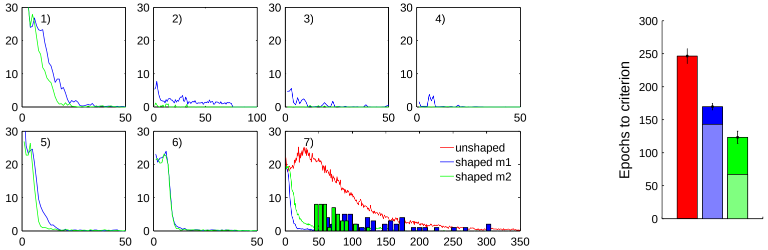
- No allocation harms performance
- Critical interference
- Increasing network capacity doesn't help

A more realistic procedure for allocation



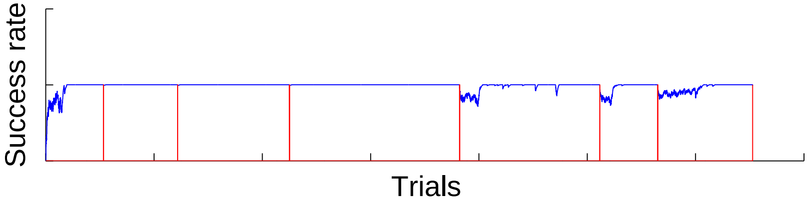
- For each new task:
  - Create a new fully plastic memory unit
  - Reduce plasticity to minimum of previous units.
- Treat all task equal.
  No special treatment of the 12-AX task stage

A more realistic procedure for allocation



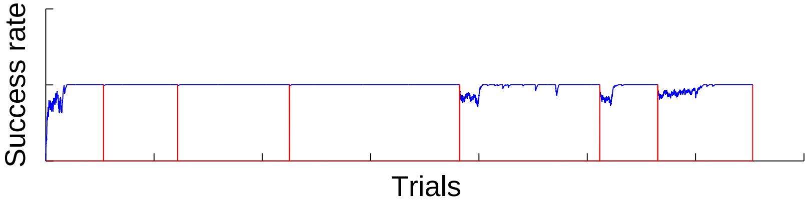
- For each new task:
  - Create a new fully plastic memory unit
  - Reduce plasticity to minimum of previous units.
- Treat all task equal.
  No special treatment of the 12-AX task stage
- Meta plasticity

Errors and uncertainty



- Shaping ⇒ occasional sudden changes of task structure
- Large degree of unexpected uncertainty
  - Perhaps signalled by norepinephrine (Yu 2005)

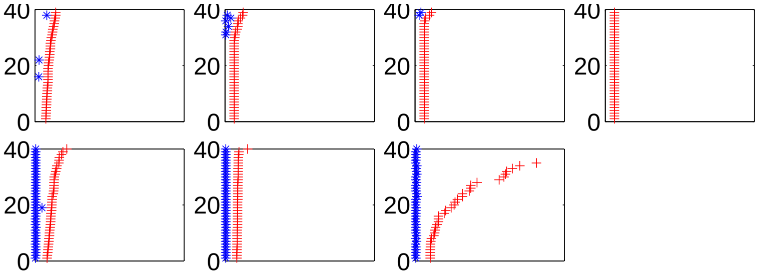
Errors and uncertainty



- Shaping ⇒ occasional sudden changes of task structure
- Large degree of unexpected uncertainty
  - Perhaps signalled by norepinephrine (Yu 2005)

- A full model of uncertainty ⇒ expected vs unexpected uncertainty
- Dirichlet process prior on allocation?

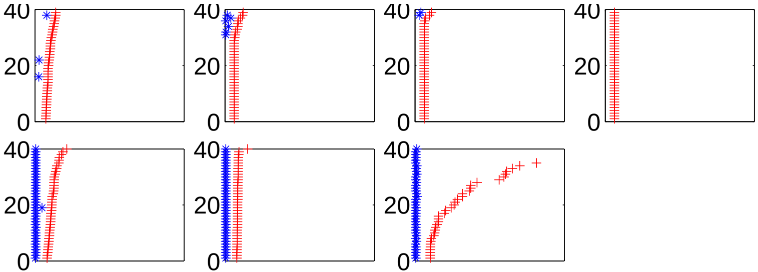
Allocation points



How effective is auto allocation?

- Mostly detects correct task boundaries
- A few Spurious points
- Can help get out of local minima

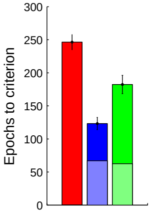
Allocation points



How effective is auto allocation?

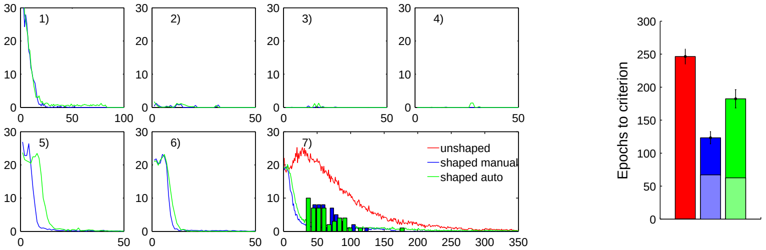
- Mostly detects correct task boundaries
- A few Spurious points
- Can help get out of local minima
- More difficult in a non deterministic task

Shaping in its full form



- Fully automated allocation: still significantly better than unshaped networks
- Performs worse than manual allocation though

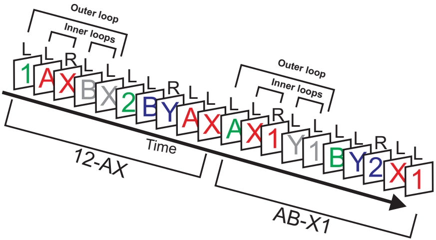
Shaping in its full form



- Fully automated allocation: still significantly better than unshaped networks
- Performs worse than manual allocation though
- Looks very similar in the learning curve. Worse performance on the tails

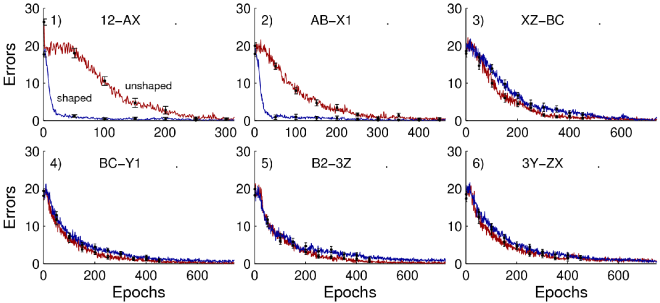
Outline

Generalisation of the 12-AX task



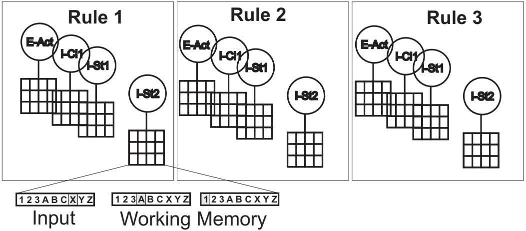
- Another form of generalisation:
  Separate abstract rules from concrete symbols
- Keep rules fixed but switch stimulus identity
  - 12-AX ... AB-X1 ... XZ-BC task
- Need to be able to rapidly update stimulus mapping without losing rules

LSTM and the generalised 12-AX task



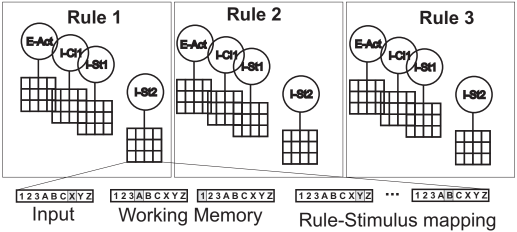
- No improvement of switching time over repeated reversals
- Shaping alone can not provide this type of generalisation
- Rules and stimulus identity are linked in weight matrix

A bilinear model of rule execution



- Internal / External actions
  - gating: store / memory clear
- $P(Act) = \sigma(\vec{x}W\vec{x} + \vec{w}\vec{x} + b)$
- Multiple simultaneous rules (Or operator)

A bilinear model of rule execution

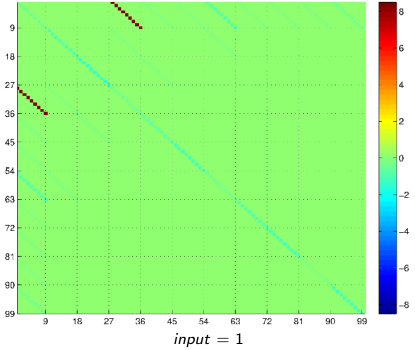


- Internal / External actions
  - gating: store / memory clear
- $P(Act) = \sigma(\vec{x}W\vec{x} + \vec{w}\vec{x} + b)$
- Multiple simultaneous rules (Or operator)
- Extend state space by rule - stimulus mapping

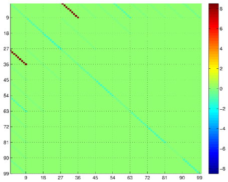
Training the model

- Supervised, non-sequential training
- Generate sets of training examples
(e.g. "X | 1 A | 1 A X 2 B Y Z 3" $\Rightarrow$ R)

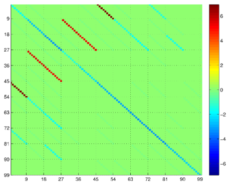
- Train each rule separately: approximate shaping
- Only possible operation: comparison to variable mapping
  - off-diagonal elements can't contribute
- Restrict to multi-diagonal weight matrix



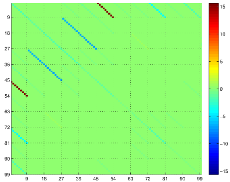
Learned weights



$input = 1$



$Input = X \wedge Mem1 = 1 \wedge Mem2 = A$



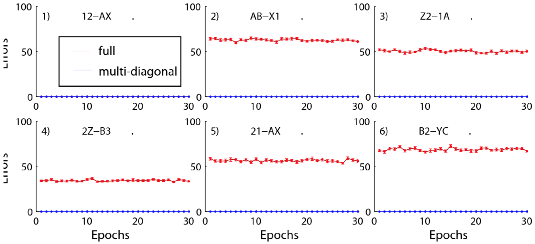
$input = X \wedge (Mem1 \neq 1 \vee Mem2 \neq A)$

- Performs task without errors if mappings are loaded correctly
- Reversals as easy as storing new memories
  - $[1AX2BYCZ3] \Rightarrow 12 - AX$
    $[AX1BY2CZ3] \Rightarrow AB - X1$

Automatic generalisation

- Initially weights trained on many rule-stimulus mappings
- Force ability to generalise

Automatic generalisation



- Initially weights trained on many rule-stimulus mappings
- Force ability to generalise

- Can generalisation occur naturally?
- Make learning generalisations easier than specifics
- Proof of concept: Multi-diagonal restriction can achieve this

Caveats

- A model of execution, not learning

Caveats

- A model of execution, not learning
  - learn stimulus - rule mapping on reversal

Caveats

- A model of execution, not learning
  - learn stimulus - rule mapping on reversal
  - learning in a sequential way

Future directions

- Learn stimulus mapping by destabilising representations on negative reward

Caveats

- A model of execution, not learning
  - learn stimulus - rule mapping on reversal
  - learning in a sequential way

Future directions

- Learn stimulus mapping by destabilising representations on negative reward
- Tackle sequential learning in a different approach:
  - the normative (Bayesian) way
  - representational learning in PFC (task grammars)

Outline

The need for behavioural data

- 12-task has been used in several modeling papers
  - simple enough to learn and analyze
  - rich enough to try various ideas

The need for behavioural data

- 12-task has been used in several modeling papers
  - simple enough to learn and analyze
  - rich enough to try various ideas
- No published behavioural learning data on the task so far
  - test some of the predictions from the models
  - coordinate with experiments done here in the lab

The need for behavioural data

- 12-task has been used in several modeling papers
  - simple enough to learn and analyze
  - rich enough to try various ideas
- No published behavioural learning data on the task so far
  - test some of the predictions from the models
  - coordinate with experiments done here in the lab
- Only preliminary data so far
  - can't draw any conclusions yet
  - would really like feedback and suggestions though

The need for behavioural data

- 12-task has been used in several modeling papers
  - simple enough to learn and analyze
  - rich enough to try various ideas
- No published behavioural learning data on the task so far
  - test some of the predictions from the models
  - coordinate with experiments done here in the lab
- Only preliminary data so far
  - can't draw any conclusions yet
  - would really like feedback and suggestions though
- Don't necessarily expect models to capture human behaviour
  - hypothesis testing?

Instructions to subjects

- Subjects are asked to learn the rules of a task
  - keep parameters close to those used during modelling
- Responses are self timed (20 seconds timeout)

Properties of the task

- The rules and feedback are deterministic and reliable
- The rules depend on the current and some of the previous stimuli.
- The rules don't depend on the subjects response or its correctness

The behavioural experiment

Subjects

- Subjects were recruited from the UCL psychology subject pool
- Declared when they learned the rules

Feedback

- Correct / Incorrect feedback after each trial
- Asymmetric reward: Cumulative reward after each epoch
- Performance related pay (calculated during testing trials)

The behavioural experiment

Subjects

- Subjects were recruited from the UCL psychology subject pool
- Declared when they learned the rules

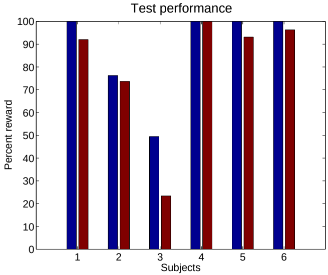
Feedback

- Correct / Incorrect feedback after each trial
- Asymmetric reward: Cumulative reward after each epoch
- Performance related pay (calculated during testing trials)

Testing trials

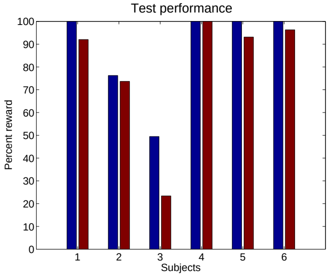
- Verify rules were learned correctly
- Explore to Exploit
- No feedback was given during testing epochs

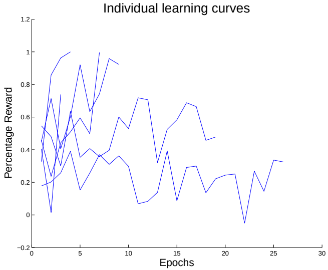
Condition 1: The plain 12-AX task



- The base 12-AX task is well learnable
    - 4 out of 6 subjects learned rules correctly

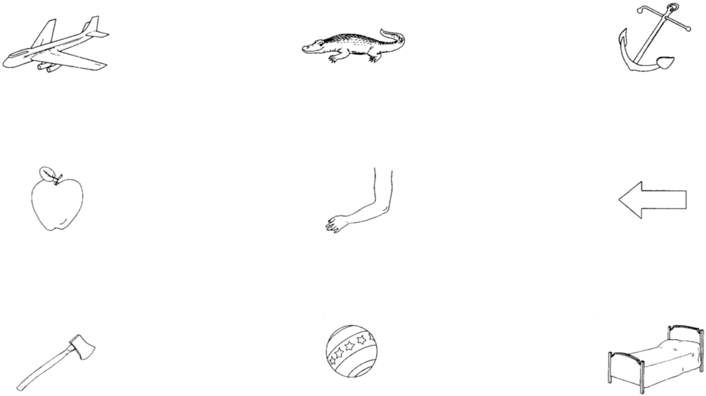
Condition 1: The plain 12-AX task





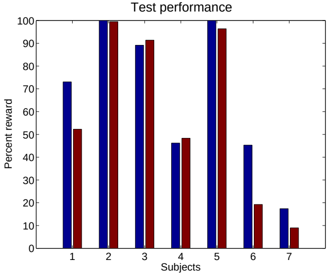
- The base 12-AX task is well learnable
  - 4 out of 6 subjects learned rules correctly
- Majority learn it in a short number of trials (3 - 8) epochs
  - wallclock time: 10 - 30 mins
  - one subject described it as too easy.

The Apple-Axt task



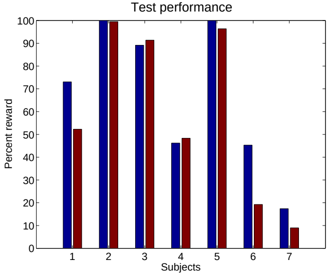
Taken from Snodgrass & Vanderwart (1980)

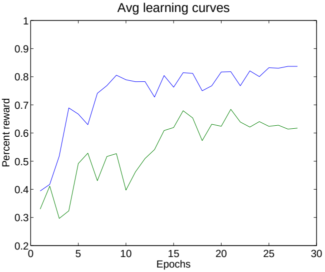
Making it harder: The Apple-Axt no pairs task



- Change in stimulus / change in task statistics (removed pair structure)
- Task just about still learnable
- 3 out of 6 subjects learned

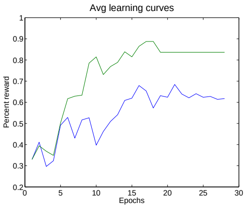
Making it harder: The Apple-Axt no pairs task





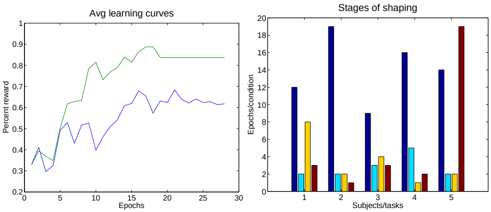
- Change in stimulus / change in task statistics (removed pair structure)
- Task just about still learnable
- 3 out of 6 subjects learned
- Looks like a difference might emerge

Condition 3: Shaping



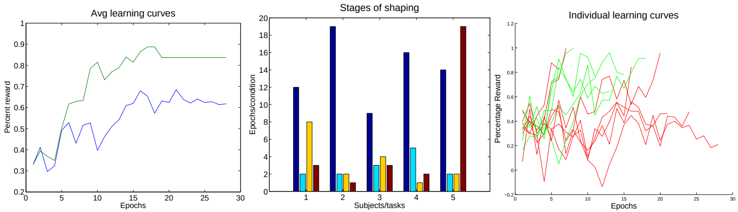
- Train on shaping tasks first.
- Less effect than hoped:
- Debriefing: Several subjects couldn't name relation between tasks

Condition 3: Shaping



- Train on shaping tasks first.
- Less effect than hoped:
- Debriefing: Several subjects couldn't name relation between tasks

Condition 3: Shaping

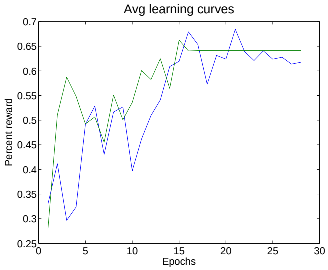


- Train on shaping tasks first.
- Less effect than hoped:
- Debriefing: Several subjects couldn't name relation between tasks

Condition 4: Shaping without knowing task boundaries

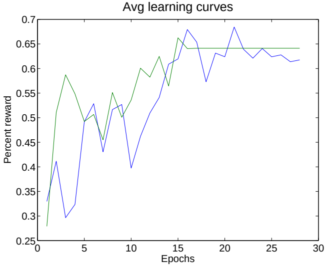
- No hint of task changes: more comparable to model

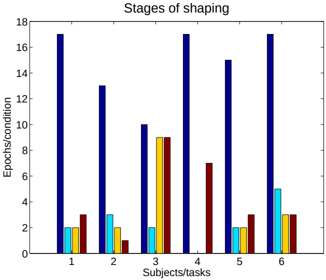
Condition 4: Shaping without knowing task boundaries



- No hint of task changes: more comparable to model
- No benefit for shaping?

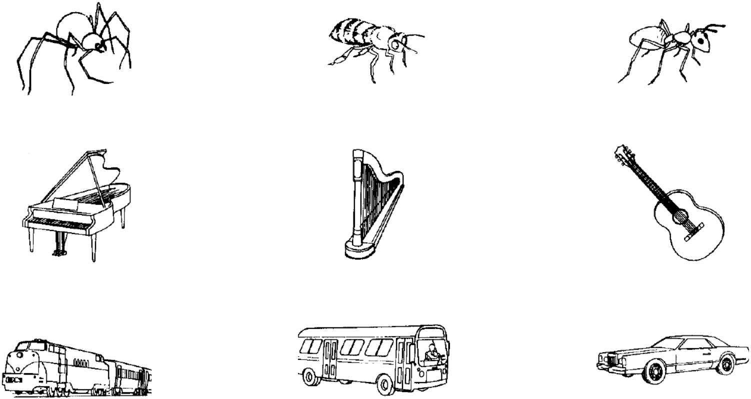
Condition 4: Shaping without knowing task boundaries





- No hint of task changes: more comparable to model
- No benefit for shaping?

Stimuli for the Piano-Train task



Taken from Snodgrass & Vanderwart (1980)

Further conditions to test

- Try to confirm differences in stimulus categorisation
    - not just result of increase working memory load?
- Try and understand more why no stronger benefits of shaping are seen
- Test ability for reversal and rule - stimulus abstraction (differences in shaping)

Conclusions

A need for modelling shaping

- Sequential learning: important in flexible behaviour
    - Shaping provides tools to study it methodically
- Need architectural support for full benefits

Conclusions

A need for modelling shaping

- Sequential learning: important in flexible behaviour
  - Shaping provides tools to study it methodically
- Need architectural support for full benefits

Allocation: a potential solution

- Viable solution to benefit from sequential learning
- Automatic allocation: fully integrate into learning architecture

Conclusions

A need for modelling shaping

- Sequential learning: important in flexible behaviour
  - Shaping provides tools to study it methodically
- Need architectural support for full benefits

Allocation: a potential solution

- Viable solution to benefit from sequential learning
- Automatic allocation: fully integrate into learning architecture

Behavioural data

- To early to draw conclusions: Useful hints for future models?

Thank you for listening!

Task grammars

A Bayesian model of cognitive flexibility

- Learning as inference in a graphical model of tasks
- Assumption: World "generates" task in a highly structured way
- Slow learning of parameters of inverse model
- Rapid inference of individual tasks

- Representational learning in PFC
  - Map neural hierarchies onto model hierarchies