Binons

Version 2.0
2nd April, 2022

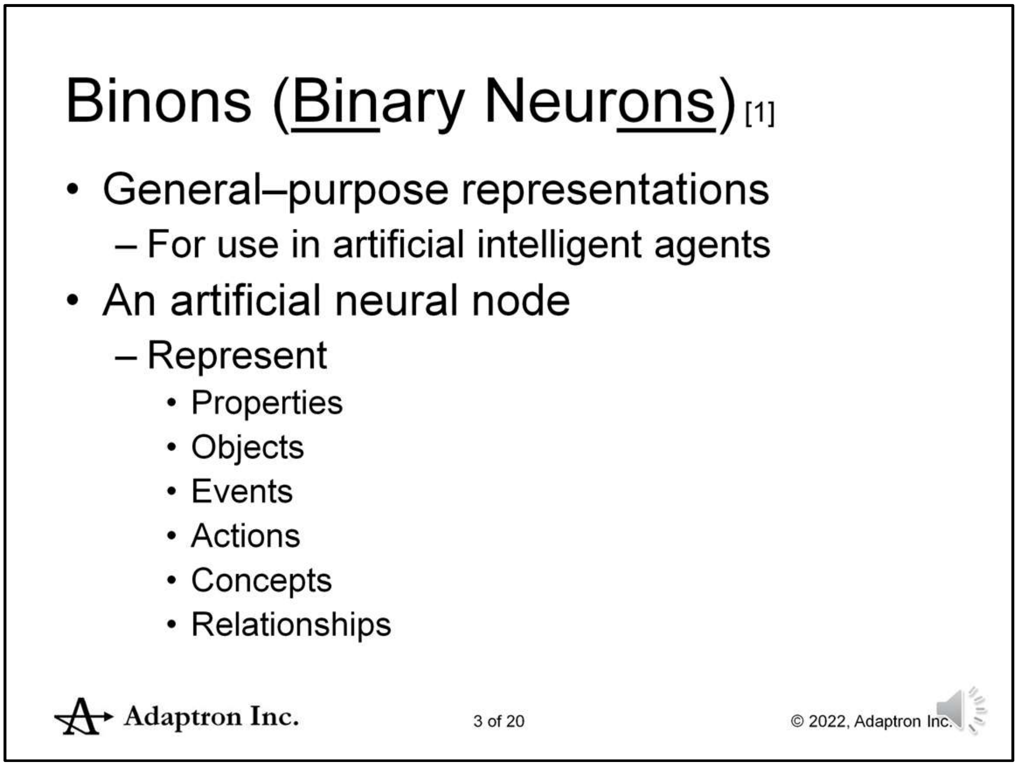
Binons are the general-purpose representations used in Adaptron.
They are learnt and can be performed.
That means they are not only used to represent spatial and temporal patterns for object and event recognition, but also for performing actions.

Contents

- Binon's structure
- Multiplicity
- Binon functionality and states
- Spatial and temporal binons
- Property, entity and control binons
- Calculating ratio values
  - Using logarithms based on JND*
- Symbolic patterns

* JND: Just Noticeable Difference

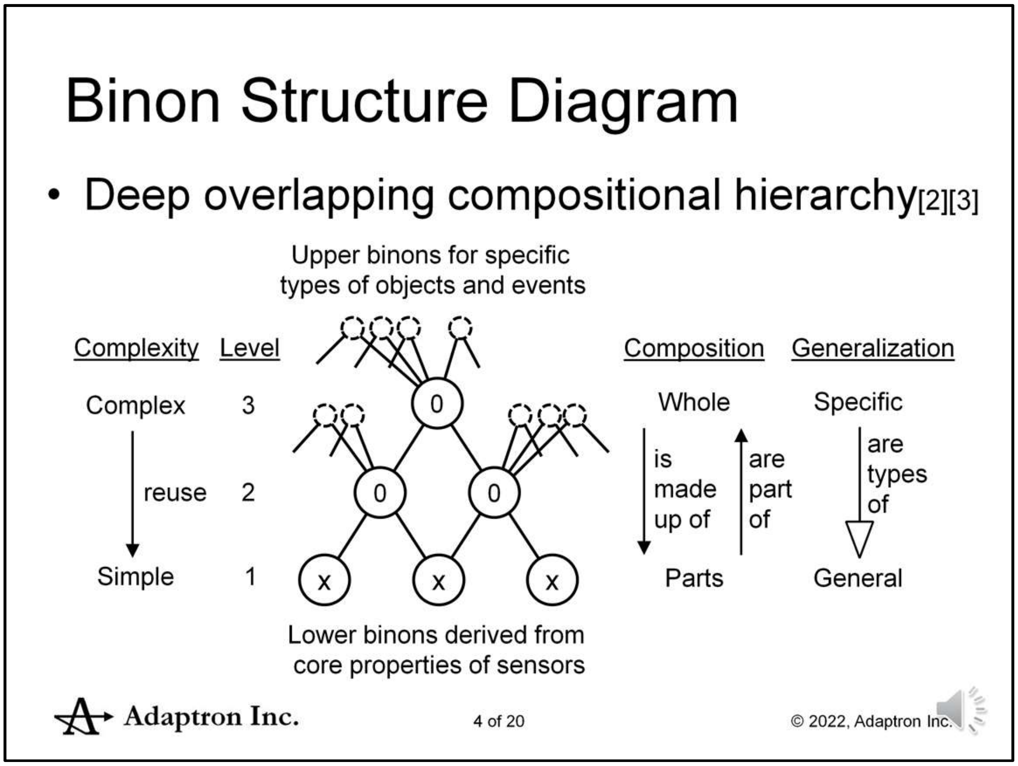
This presentation describes the following:
The structure of binons and their multiplicity property.
The states of binons and how they function
Dividing binons into spatial and temporal ones and the types of binons such as property, entity and control binons.
Calculating binon values to represent ratios and how logarithms and the Just Noticeable Difference is used in their derivation.
And finally how binons can be used to represent symbolic patterns.
More details of how binons are learnt and composed into an artificial neural network are explained in subsequent presentation.



The word binons comes from binary neurons. They are general-purpose representations for artificially intelligent agents such as Adaptron.

They can be used for learning and representing objects and events as perceived in the world and actions performed by an agent.

They can be used to represent concepts and properties of things. They can also associate all these things together as relationships.



This a binon structure diagram. Binons are combined into deep (that is a multi layer) overlapping compositional hierarchies.

Being compositional means that upper binons are made up of the lower ones.

Two lower binons are combined to form an upper binon so the structure is a binary hierarchy from the top down.

The lower binons are closer to sensors and are simple.

But a binon can be shared and reused in the composition of many upper binons. This results in a lattice network from a bottom up perspective.

Upper binons are more complex. They become more specific at higher levels and are more general at lower levels.

Binons

- Lower to upper binons
  - Simple to complex
  - General to specific
- Deterministic
  - No weights on links
  - No backpropagation [4]
- Values are integers [5]
- Overlapping
  - Only one representation per combination

Adaptron Inc. 5 of 20

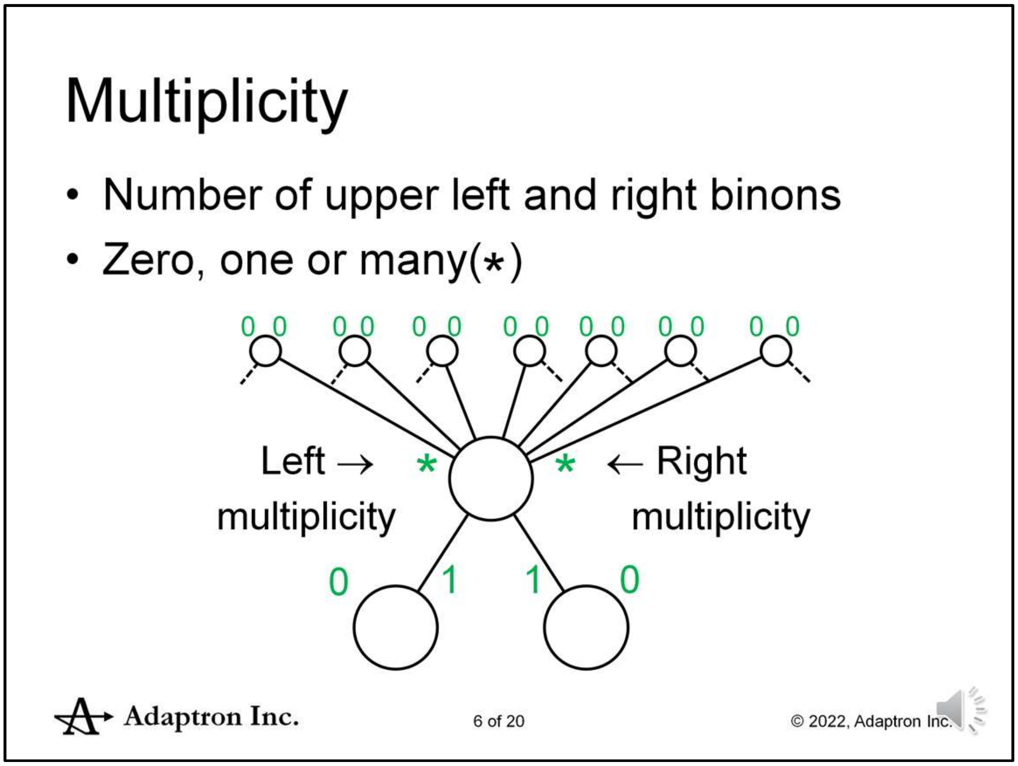
The lower to upper binon relationships were illustrated on the previous slide.

Binons are deterministic because there are no weights on the links or in ANN terms the activation function is a step function, the weight is 1 if the link exists and zero if there is no link. And there is no backpropagation process as found in modern deep learning networks.

The important values in the nodes are integers, (i.e., the Xs) and they represent properties such as ratios. Rather like McCulloch-Pitts neurons [6].

Ratio values are in level 1 binons and are the lowest level of symbolic representation.

The network structure is overlapping so that there is only one representation for any combination of binons.



Binons have two multiplicities. They are the number of links to the upper left and right binons.

That is they are the counts of how many binons are reusing a binon.

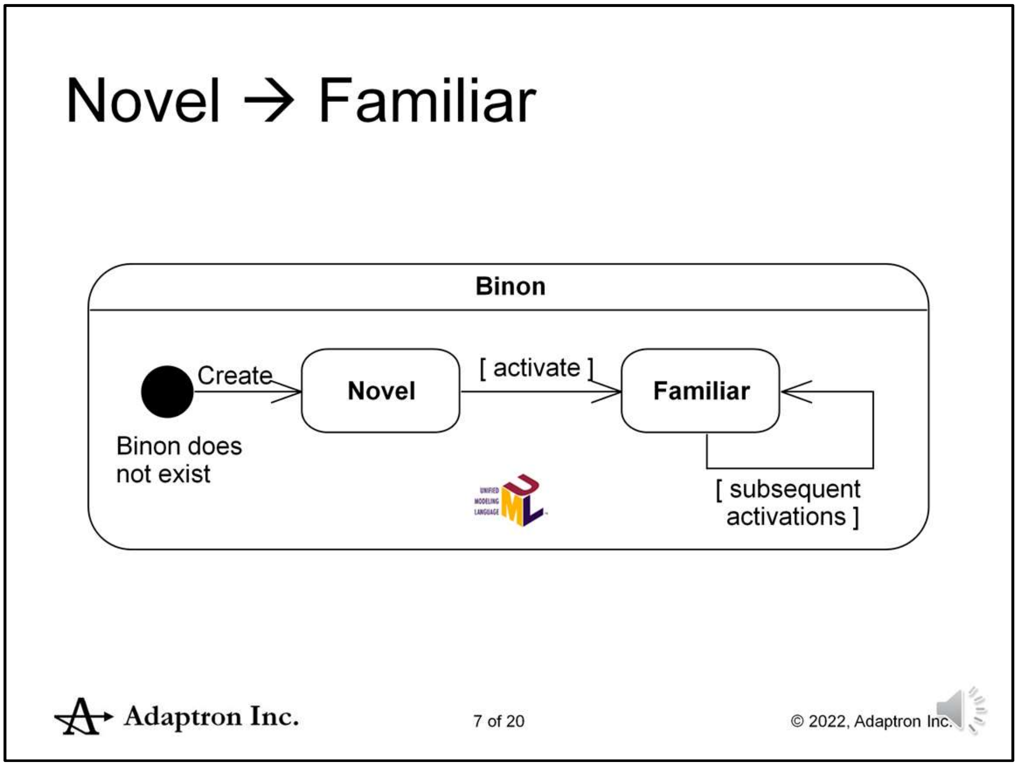
Multiplicity is either zero, one or many (i.e., two or more).

If both multiplicities are zero then the binon is a brand new one and has not been reused. That is it is not part of any more complex binon.

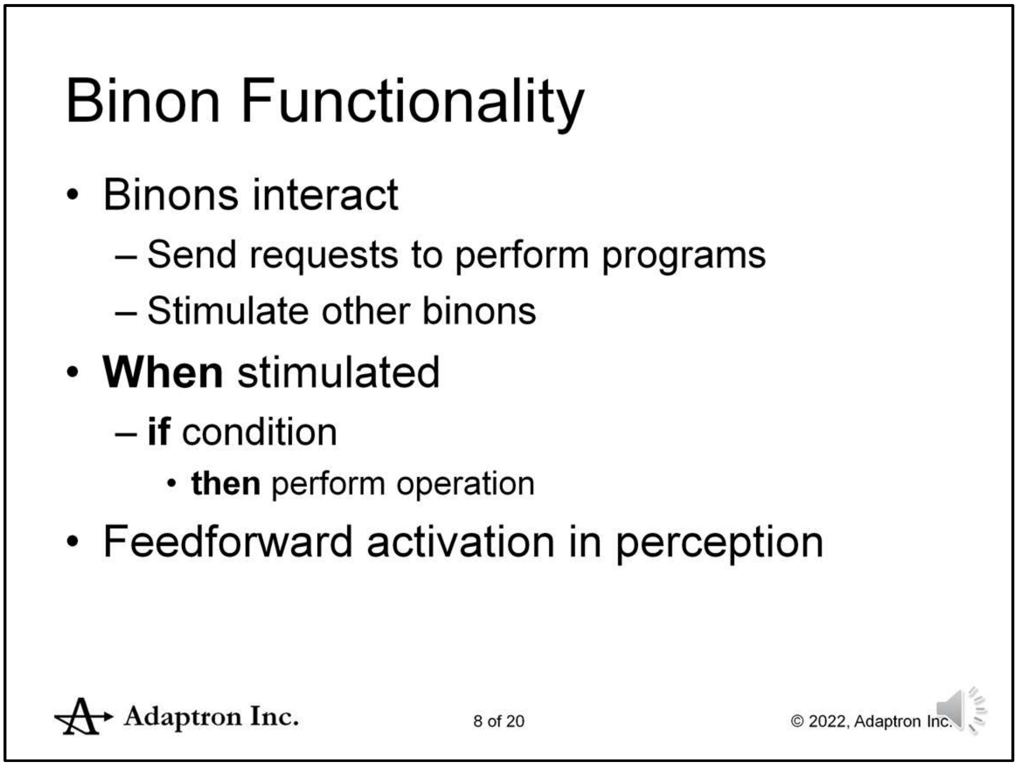
A multiplicity of one indicates you can predict the associated binon in that direction.

A good example in English words are the letters Q and U. The Q predicts the U.

A multiplicity of more than one indicates the binon is being reuses many times and this means there is uncertainty in what other binons may occur beside it.



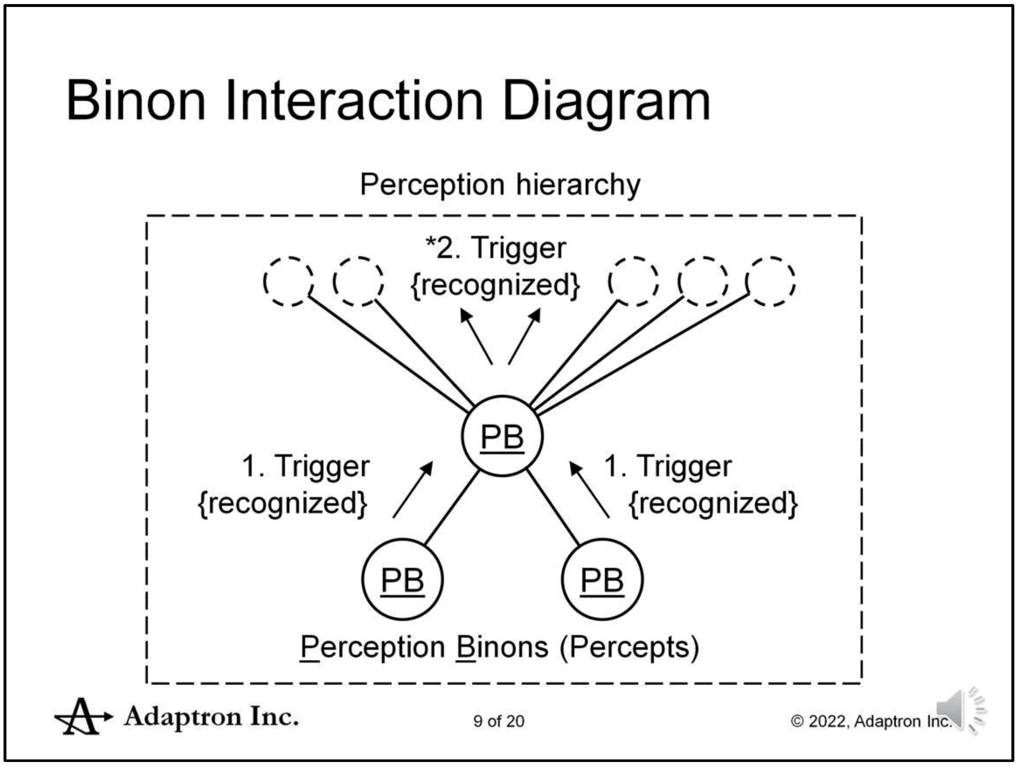
When the things that binons represent are first experienced their binons are novel. When they are experienced the second time and activated as a result they become familiar. They stay familiar upon further activation when experienced.



Binons are functional components just like objects in object oriented software. They interact with each other.

They send requests (aka. messages or commands) to other binons asking them to perform a program (aka. a method).

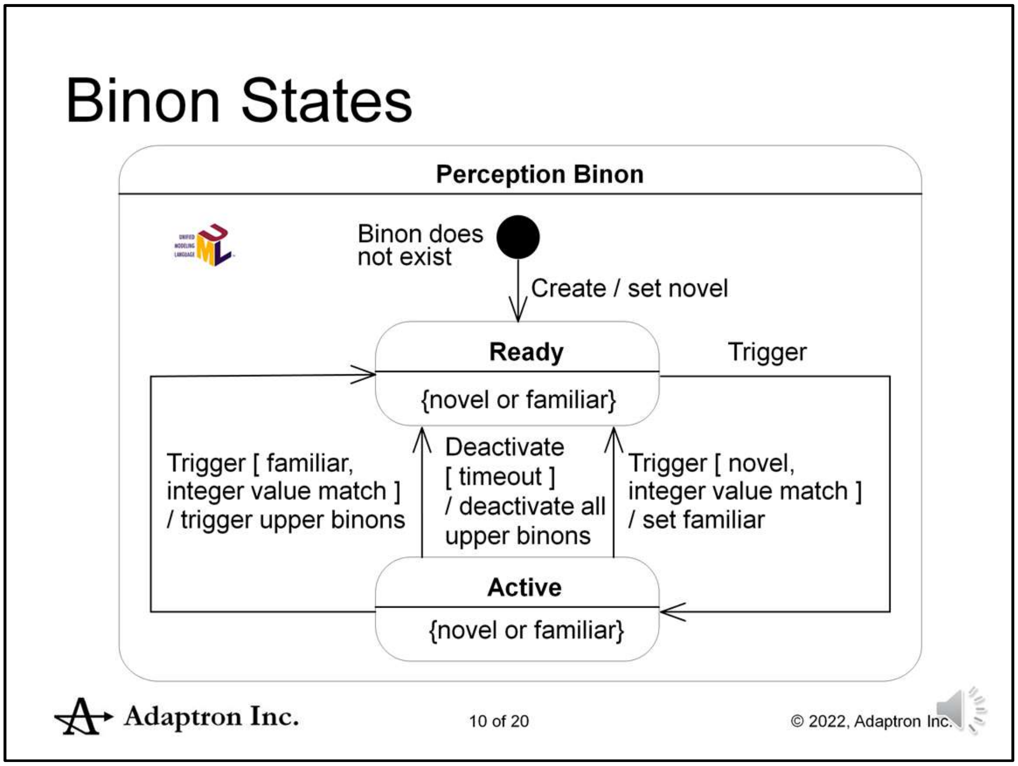
And these programs take the form of When stimulated with the appropriate request, if a condition is true then perform an operation which may include sending requests to other binons. The result is a feedforward wave of activation up a hierarchy of binons for perception as illustrated in the next slide.



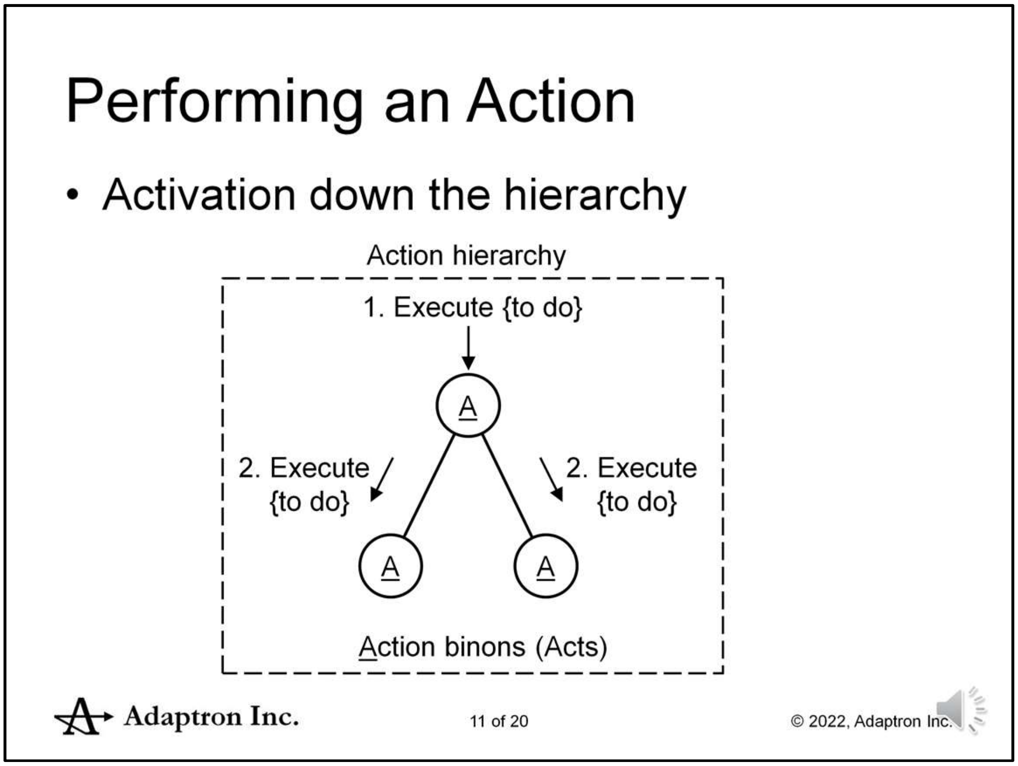
This is a binon interaction diagram. It captures the order in which the perception process occurs.

A perception binon is always waiting for its lower parts to be recognized. When both lower binons are recognized they trigger the upper binon and it then notifies all of its upper binons, notifying them that is has been recognized.

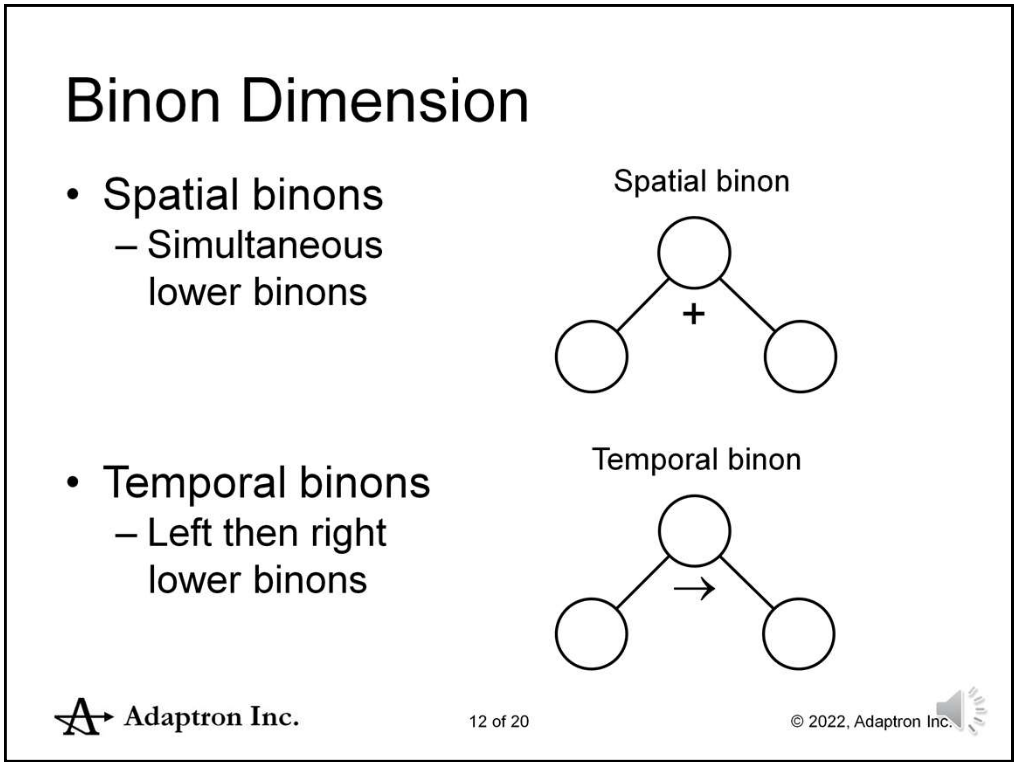
In this way Adaptron identifies the most complex things in the world that it knows of.



Perception binons transition between two states as they process their requests. Upon creation they are novel and ready to handle stimulating commands. When they are triggered by the first lower binon they become active, waiting for the stimulation of the second lower binon. When stimulated by the second trigger command if it is active and novel and its integer value is a match then the binon is set to familiar and returns to the ready state. But when it is stimulated by the second trigger command if it is familiar and its integer value is a match then it triggers all its upper binons and returns to the ready state. When the lowest level binons time out because they do not receive their second request in time they deactivate all their upper binons which return to the ready state.



For performing actions the wave of activation flows down the action hierarchy to the lowest action binons which then activate the actuators. There is more detail about this in the presentation on action control.

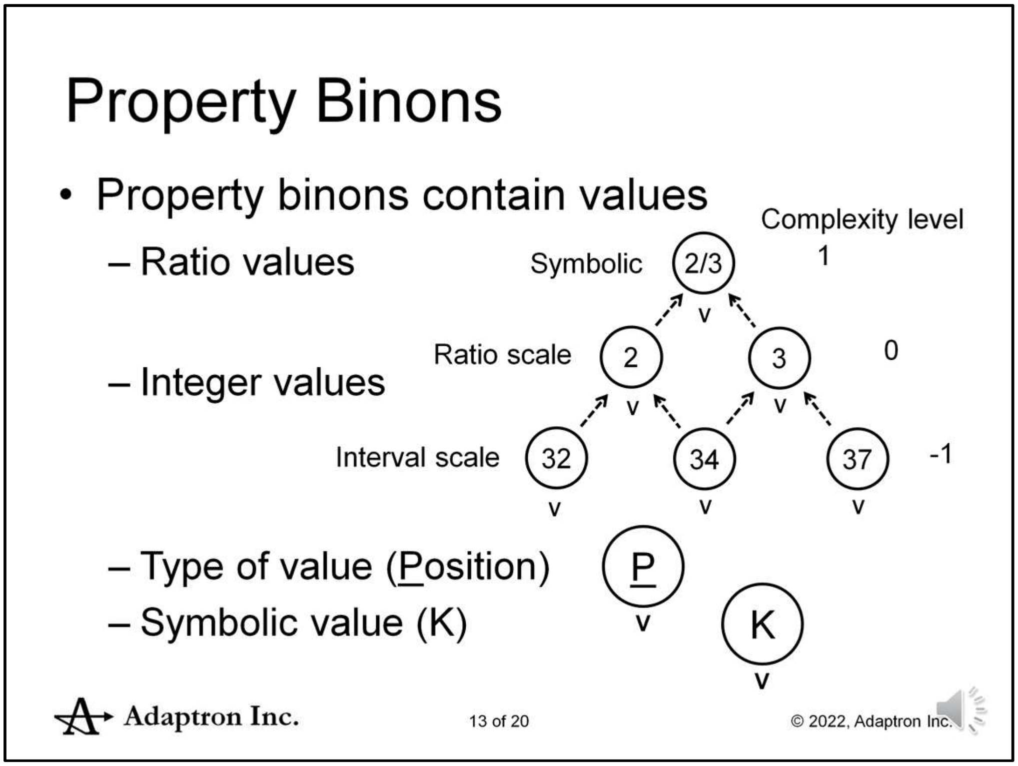


Binons can be divided into spatial and temporal ones.

The plus sign indicates that both lower binons must occur simultaneous in the same time interval.

And the arrow indicates that lower binons occur in sequence.

If there is no symbol then the diagram applies to both spatial and temporal recognition.



The three types of binons are property binons, entity binons and control binons.

Property binons are used to represent values. They can be used as the lowest binons in a network.

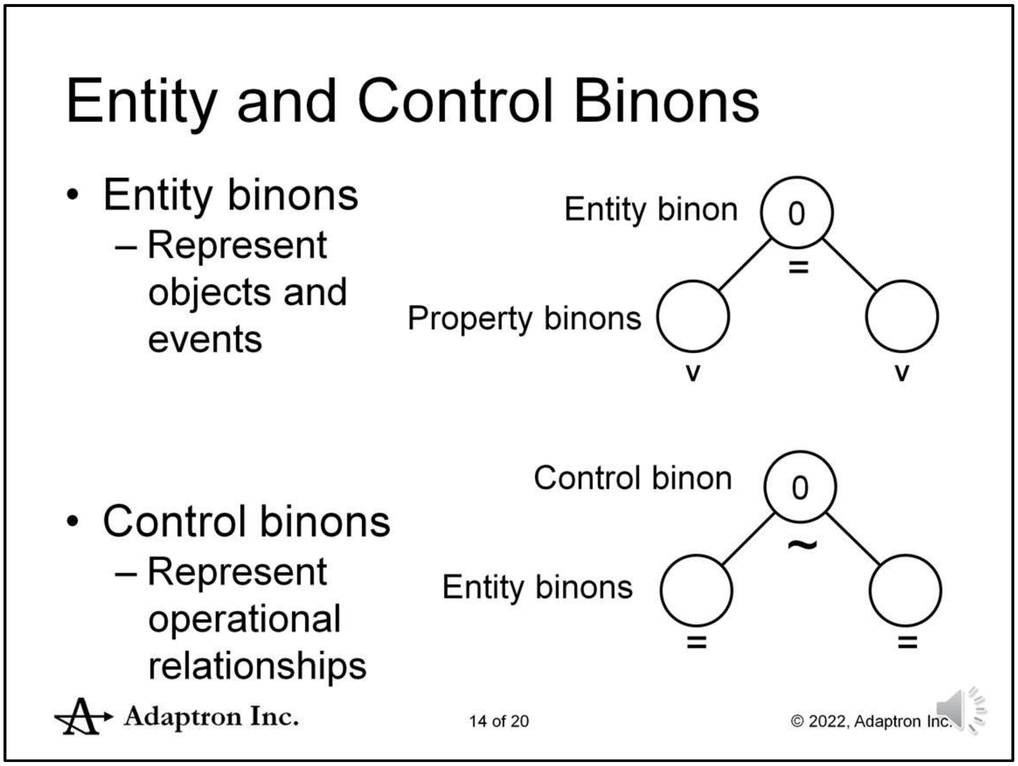
But when one value is derived from another they are not linked together. Instead the dashed arrows shows how their values are derived. The integer values for interval and ratio scale values are obtained from sensors.

Ratio scale values are obtained by subtracting one interval scale value from another. If 32 and 34 were positions the distance between them would be the ratio scale value of 2 for distance.

Ratios are obtained by dividing one ratio scale value by another as in the derivation of 2/3. Later in the presentation

I describe how a ratio can be represented as an integer for symbolic purposes at complexity level 1.

If it is desirable to just illustrate what type of value the property binon contains then the first letter of the property is used as the capital P from position, T for time, I for Intensity etc. If the property value is a symbolic character it can be placed in the binon instead.

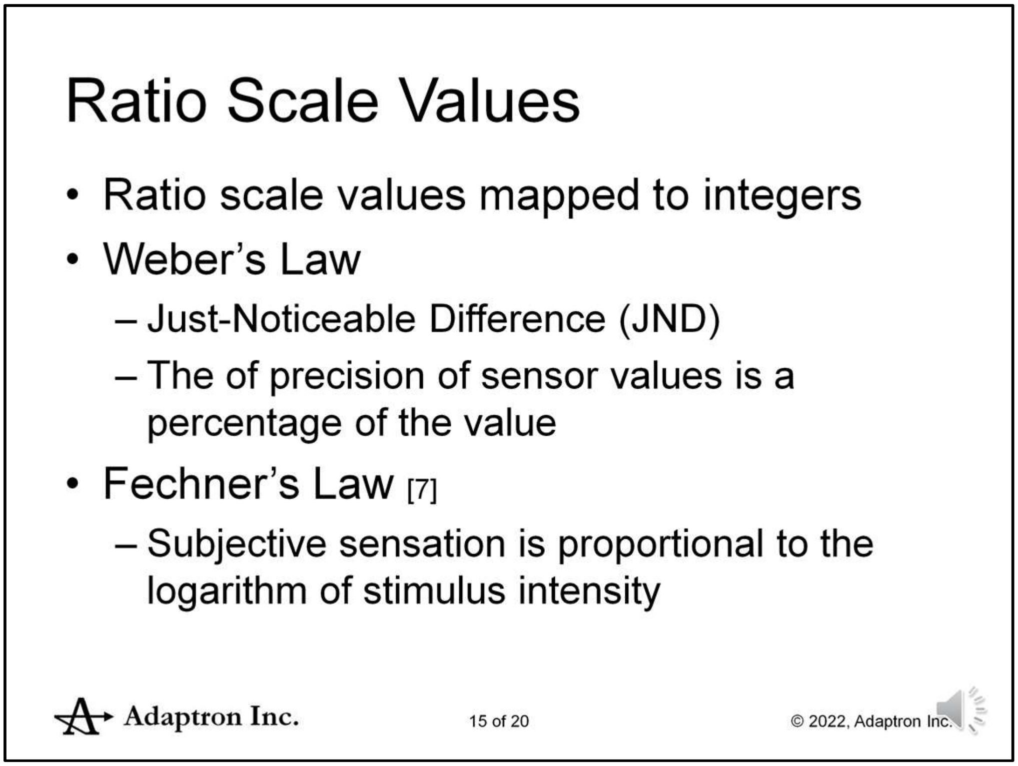


Entity binons are used to represent objects and events. They are formed into the deep overlapping compositional hierarchies.

At their lowest level they are composed of property binons.

Control binons associate entity binons and represent operational relationships between them for controlling their execution.

Entity binons are drawn with an equal sign under them and control binons have a tilde under them.



How are ratio scale values represented as integers in property binons? They are real numbers and can span many orders of magnitude.

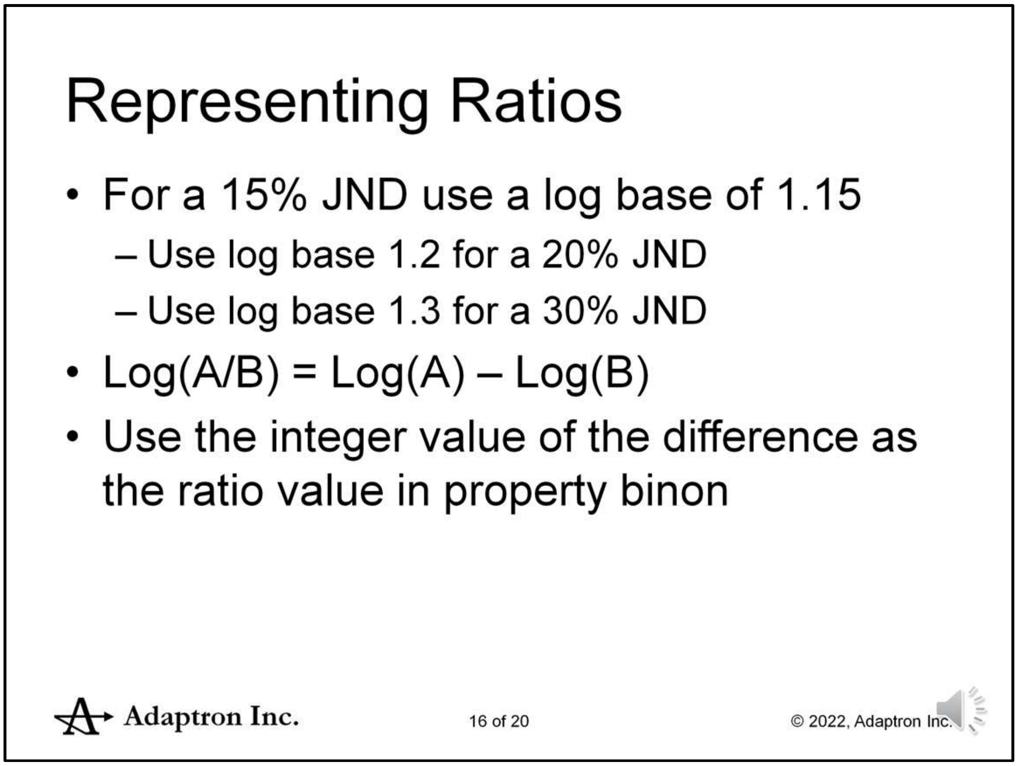
Adaptron uses two psychophysical principles from Weber and Fechner in their representation.

Weber's law says the JND is the resolution that sensors can detect and it is always a percentage of the reading value.

Fechner's law says that subjective sensation is proportional to the logarithm of stimulus intensity.

So Adaptron uses the integer part of the logarithm of a ratio scale value. The log base is dependent on the percent JND.

Be careful about what units are being used to measure a ratio scale value. For example, sound levels measured in decibels are already a logarithmic scale. But the unit of sound pressure is the Pascal and Candela are the units of luminous intensity and are not logarithmic.

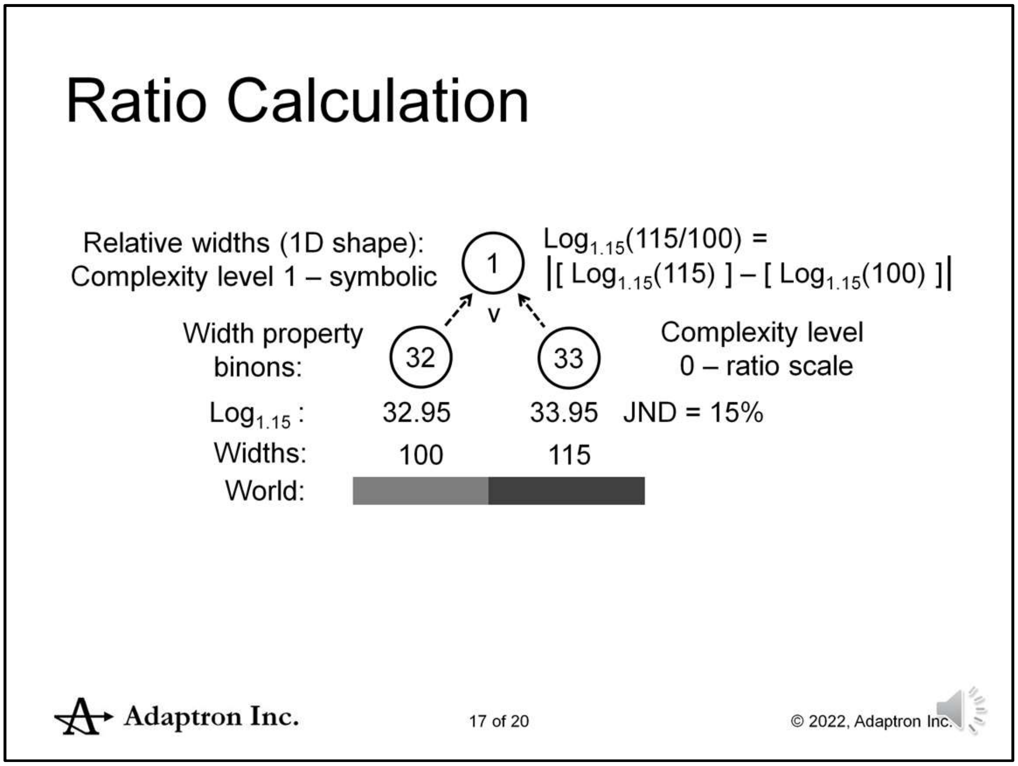


The log base to use is one plus the percent JND. For a 15% JND that means use a log base of 1.15. For a 20% JND use a log base of 1.20 etc. When you do this, the integer part of the logarithm of the ratio scale values that have no noticeable difference will all be the same.

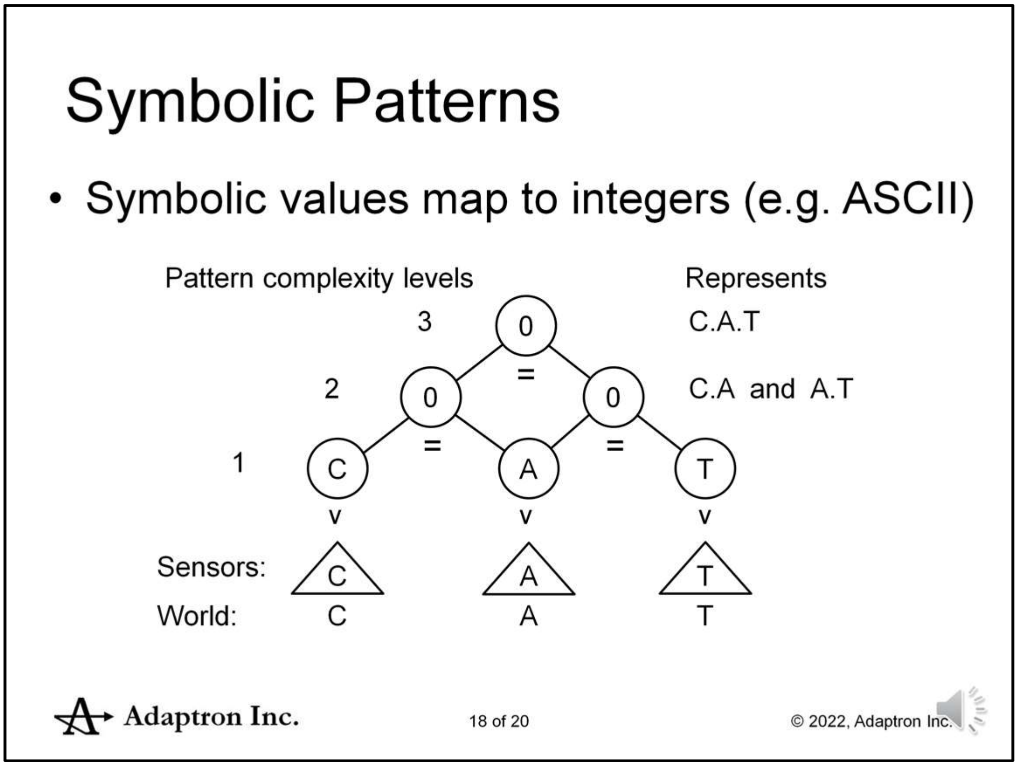
A nice feature about logarithms of ratios is that the calculation is replaced by subtraction.

Therefore for representing a ratio the difference between the two logarithms of ratio scale values can be used.

The result is an integer and is symbolic and used as the value for level 1 property binons.

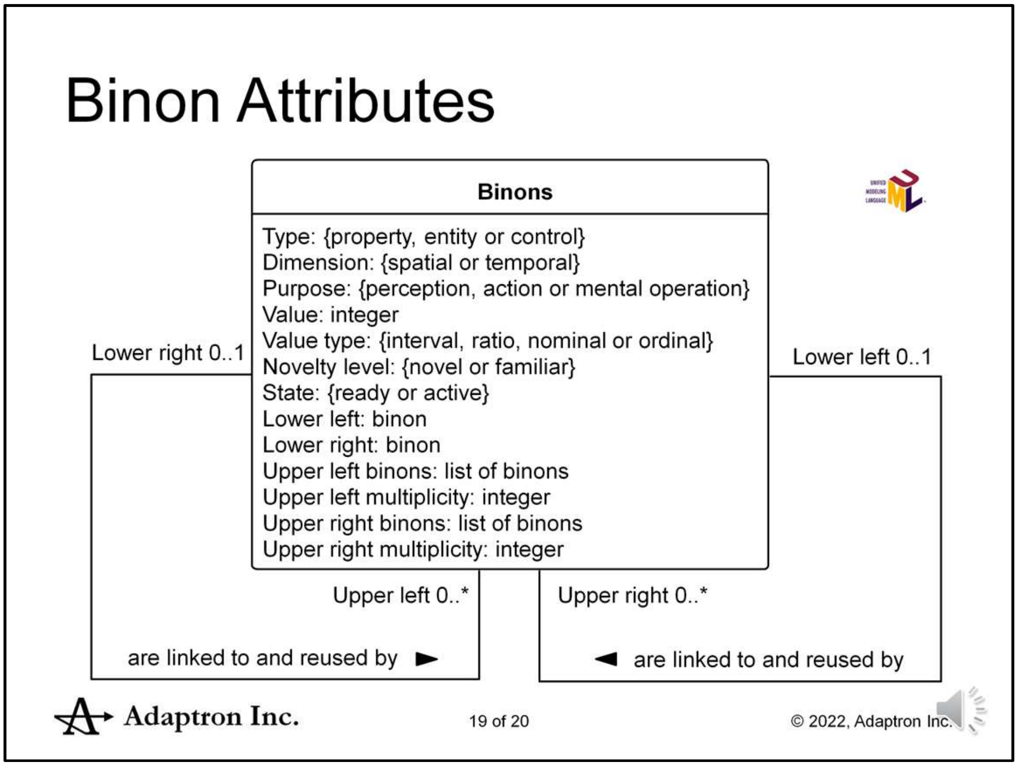


As an example, by using a 15% JND and a log base of 1.15 these two widths of 100 and 115 can be kept as the integer values of 32 and 33 in level 0 ratio scale property binons. Then the ratio of 115/100 which is just noticeable is represented by the value of one in the level 1 property binon. A ratio of anything less such as 114/100 would yield a value of 0 because it would not be noticeable using a 15% JND.



Here is an example of how text is represented with binons. Characters are already symbolic values and therefore are represented at level 1. They can be mapped to integer values as encoded in ASCII or EBCDIC.

Combinations at higher levels then represent text strings. This symbolic pattern could be recognized spatially or temporally depending on the sensors.



This UML class diagram illustrates all the attributes of a binon.

References

[1] Martensen, B. N. (2013). Perceptra: A new approach to pattern classification using a growing network of binary neurons (binons). In R. West & T. Stewart (Eds.), *Proceedings of the 12th International Conference on Cognitive Modeling (ICCM 2013)*.

[2] Fidler, S., & Leonardis, A. (2007). Towards scalable representations of object categories: Learning a hierarchy of parts. *IEEE Computer Society Conference on Computer Vision and Pattern Recognition (CVPR 2007)*.

[3] Hammer, B. (2003). Compositionality in neural systems. In M. A. Arbib (Ed.), *The Handbook of Brain Theory and Neural Networks – 2nd ed*. Cambridge, Massachusetts: MIT Press.

[4] Marcus, G. (2018). Deep learning: A critical appraisal. *Computing Research Repository (CoRR), arXiv:1801.00631v1 [cs.AI]*.

[5] Leslie, A. M., Gallistel, C. R., & Gelman, R. (2019). Where integers come from. In P. Carruthers (Ed.), *The Innate Mind: Foundations and Future*. Oxford: Oxford University Press.

[6] McCulloch, W. S. & Pitts, W. (1990). A logical calculus of the ideas immanent in nervous activity, *Bulletin of Mathematical Biology* 52(l/2), 99-115.

[7] Portugal, R. D., & Svaiter, B. F. (2011). Weber-Fechner law and the optimality of the logarithmic scale. *Minds and Machines*, 12(1), 73-81.