Euclidean automata

Andras Kornai
Department of Algebra
Budapest University of Technology and Economics
andras@kornai.com

In this note we introduce Euclidean automata (EA), a simple generalization of Finite State Automata (FSA). EA operate not on symbols from a finite alphabet as usual, but rather on vectors from a parameter space P, typically $\mathbb{R}^n$. The main motivation for EA comes from classification problems involving a forced choice between a finite number (in the most important case, only two) alternatives. Since we want classifications to be stable under small perturbation of inputs, ideally the set of points in P classified to a given value should be open, yet it is evident that we cannot partition $\mathbb{R}^n$ or $\mathbb{C}^n$ into finitely many disjoint open sets. Approximate solutions thus must give up non-overlapping, e.g. by permitting probabilistic or fuzzy outcomes, or exhaustiveness, e.g. by leaving 'gray areas' near decision boundaries where the system produces no output. EA, as we shall see, sacrifice non-overlapping but maintain sharp, deterministic decision boundaries.

Using EA we offer the beginnings of an analysis of being in a conflicted state, some situation where we know that we should do A, since it is 'the right thing to do' yet we have a strong compulsion to do some B (including doing nothing) instead. To anchor the discussion we will use several specific examples, such as refraining from or taking some drug such as tobacco, alcohol, or heroin, that is generally agreed to have pleasant short-term but harmful long-term effects; slipping into some recreational activity while there is still work to do; keeping or not keeping some promise; etc.

The problem is complex, arguably it is the single most complex problem we have to face in everyday life. Therefore, some simplification will be necessary, and we will state the main problem in a way that already abstracts away from certain aspects that would take us far from our goal of analyzing internal conflict. First, we are not interested in defending the specific moral premisses used in the analysis of drug addiction, laziness, and similar examples of conflict: our focus is with the conflicted state itself, not the individual components. Second, we will pay only limited attention of the issue of how we know that A is the right thing rather than some alternative A' or even B – we are interested in the situation when we already *know* that A is right and B is wrong. Third, real life conflicts are rarely between two, laboratory pure components: often there are multiple factors, but the binary case must be addressed first. Finally, conflicts are often gradient (perhaps a small glass of wine is quite OK where a bottle would not be) but here we try to work with as simple and minimalistic a setup as we can.

The mainstream assumption, embodied in AGI architectures like OpenCog (Hart and Goertzel 2008) is that there is some *utility function* that the agent intends to maximize. If this function changes at all, it changes only adiabatically, on the order of weeks and months, while the decision to do the right thing often has to be taken on a subsecond scale. Certain issues therefore can be stated in terms of a single utility function that gets discounted on different scales. Let $u(t)$ measure the sensation of somatic well-being on a scale of -1 (suffering) to +1 (exaltation) at time t. If our interest is in maximizing $\int_0^T u(t)e^{-Ct}dt$, choosing a large C leads to behavior that focuses on the momentary exhilaration, while choosing a small C models maximizing long-term well-being. This is a nice and simple picture: if a behavioral alternative, say smoking a cigarette, has some known effect expressible as a transform A, while B has effect B, we simply compute $\int_0^T A[u(t)]e^{-Ct}dt$ and compare it to $\int_0^T B[u(t)]e^{-Ct}dt$.

Such an analysis, however, would suggest that conflict is restricted to a few marginal cases when our best estimate of A and B carry large uncertainties, and is therefore largely an epistemological issue: as soon as we have better estimates the conflict disappears. While this is a known philosophical position going back to antiquity,[1] it is not at all helpful in predicting behavior: in reality, people spend a lot more time in conflicted states than this analysis would suggest. Even more damning, it ignores the central case, when the impact of A and B are perfectly known. Rare is the addict who doesn't know she should quit, or the promise breaker who doesn't know better – the problem is not lack of knowledge, but failure to act on it.

A somewhat richer model presumes not just one utility function but several: u_1 for somatic well-being, u_2 for reproductive success, u_3 for danger avoidance, and so forth. Under such a view, conflicts between A and B are simply

[1] Unlike Mohists and Yangists seeking grounds for right choice Chuang-Tzu's ideal is to have no choice at all, because reflecting the situation with perfect clarity you can respond only in one way. (Graham 1989:190)

cases when some u_i would lead to one choice but another u_j would lead to the other. Since there can be large domains where the different us lead to different choices even if they are selected from otherwise well-behaved classes of functions (e.g. piecewise linear or low-order polynomial), this model escapes the first criticism discussed above, but not necessarily the second, a matter we will discuss shortly. Such a model fits well in multi-agent theories of the mind (Minsky 1986), by assigning each agent A_i a dedicated utility function u_i.

We will frame the problem in terms of multiple (competing) utility functions, each with its own little homunculus intent on maximizing it, but first we have to discuss two significant reduction strategies. The first one would replace the u_i with their weighted sum $\sum_i w_i u_i$ using static or very slowly changing weights. This makes a lot of sense when choices are evaluated in terms of some resource that behaves additively, such as memory or CPU expenditure, as long as there is only one of these which is truly scarce. But as soon as the system deals with several resource dimensions (e.g. CPU time, RAM, and disk space can all be limiting) we are back to the multiple optimization scenario, except it is now the resource tallies r_j that are to be minimized subject to (slowly changing) tradeoffs between them. For the problem at hand, moral correctness must be considered a separate resource on its own, since it is well understood that most problems have simple solutions as long as the moral constraints are ignored.

The second reduction strategy is based on a hard-line interpretation of a single utility, say u_1 (somatic well-being). Competing utilities, such as u_3 (danger avoidance), are considered epiphenomenal: big danger just means a high probability of complete zeroing out of u_1, and a strategy aimed at maximizing the area under the u_1 curve will result in some degree of u_3 maximization just because of this. Similarly, in a 'selfish gene' calculus, the intent is maximizing the area under the u_1 curves for all progeny, thus low reproductive success is penalized without ascribing any specific utility u_2 to high reproductive success. Note that this strategy does not guarantee a hierarchy among the u_i, because reducing u_j to u_i does not guarantee that a reduction in the other direction is infeasible. For example, taking as primary the (future-discounted) somatic well-being of progeny will make direct somatic well-being of the individual an important factor even if it receives zero direct weight in the sum, since the individual deprived of well-being is very unlikely to make the effort to reproduce successfully.

In Section 1 we introduce our principal formal tool for analyzing conflict by a series of examples and a simple definition. In Section 2 we discuss how internal conflict can be analyzed in terms of the model, and discuss an essential fact about conflicted states, that orderings are not transitive. Some conclusions are offered in Section 3.

Euclidean automata

Euclidean automata (EA) are obtained from standard finite state automata (FSA) by undoing the major abstraction concerning inputs. In FSA, inputs are simply selected from some finite alphabet Σ. In EA, *inputs* are given as parameter vectors from a parameter space P, typically $\mathbb{R}^n$, and *states* are simply subsets P_i of P indexed from a finite index set S. If $P_i \cap P_j = \emptyset$ for all $i, j \in S$ we call the EA *deterministic*, if $\bigcup_{i \in S} P_i = P$ we call it *complete*.

Experience with general systems theory shows that undoing the abstraction concerning outputs as well would lead to a theory that is too general to have any utility, and we will refrain from doing so. We will define Euclidean versions of finite state transducers (FSTs) and Eilenberg machines (XMs) that we will call Euclidean transducers (ETs) and Euclidean Eilenberg machines (EEMs), keeping the output alphabet of the transducer, and the side-effects of machines both discrete and finite. But before turning to the formal definition, let us provide some informal, easy to grasp examples both to familiarize the reader with the terminology and to compare and contrast Euclidean automata to better known models.

Example 1. Elevator A three-stop elevator running from the basement to the top (first) floor will have three main input parameters, the reading from the current position sensor, a real number between -1 and $+1$; the reading from the engine sensor, with possible values going up, stopped, going down; and the reading from the weight sensor, with any possible nonnegative reading, but in effect quantized to two discrete values, "above safety limit" or "below safety limit". By having a finite state space, even the continuous parameters such as height above ground are effectively quantized: whether the value is 0.3 or 0.9 makes no difference no matter how the other parameters are set: we see the same transition function at both. We will call two parameter vectors *indistinguishable* as long as this is true in regards to transitions for EA, both transitions and outputs for ETs, and both transitions and effects for EEMs. By relying on representatives from indistinguishable classes of parameter settings we can *skeletonize* Euclidean automata and obtain classical FSA, but as we shall see, key aspects of EA behavior go beyond what the skeleta can do.

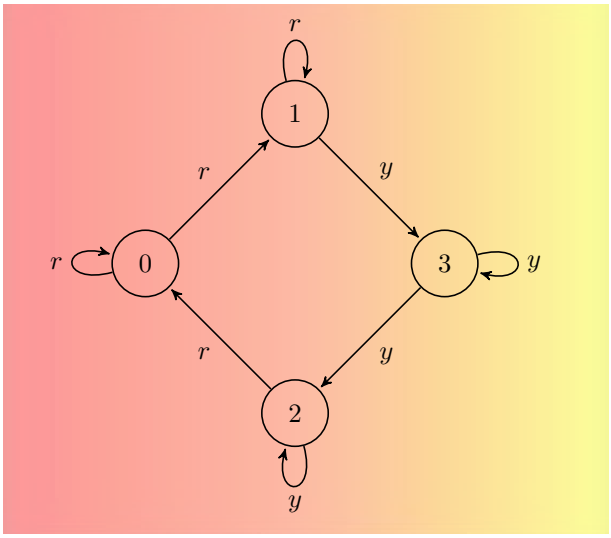
Example 2. GSM phone Near national borders, GSM handsets behave like EA: depending on which country the phone is at, it will send the user welcome messages describing the price of a call, etc. We can think of P as being composed of two parameters, longitude and latitude, or as being composed of several parameters representing the signal strengths from various cell towers. Either way, it is the values of these continuous parameters that determine (in addition to keyboard input) the behavior of the EA. Two aspects of this example are worth emphasizing: first, that the immediate behavior of the EA is determined both by the input and its previous state (so the natural formulation will resemble Mealy, rather than Moore, automata) and second, that the output of one EA can impact the input of other EA, for we may very well conceive of cell towers themselves as Euclidean automata (though the changes in their inputs are effected by changes in electricity, call load, etc. rather than by changes in their physical location).

Example 3. The heap The heap or *sorites* paradox, known since antiquity, probes the vagueness of concepts like 'heap' – clearly one grain is not a heap, and if k grains are not a heap $k + 1$ grains will also not be, so the conclusion that 10,000 grains are not a heap seems inevitable. Here we

will take the following form of the paradox (Sainsbury and Williamson 1995):

> Imagine a painted wall hundreds of yards or hundreds of miles long. The left-hand region is clearly painted red, but there is a subtle gradation of shades, and the right-hand region is clearly yellow. The strip is covered by a small double window which exposes only a small section of the wall at any time. It is moved progressively rightwards, in such a way that at each move after the initial position the left-hand segment of the window exposes just the area that was in the previous position exposed by the right-hand segment. The window is so small relative to the strip that in no position can you tell the difference in colour between what the two segments expose. After each move, you are asked to say whether what you see in the right-hand segment of the window is red. You must certainly answer "Yes" at first. At each subsequent move you can tell no difference between a region you have already called red and the one for which the new question arises. It seems that you must after every move call the new region red, and thus, absurdly, find yourself calling a clearly yellow region red.

We will model this situation by a EA with four states numbered 0 to 3, and a single numerical parameter corresponding to the wavelength at the spectral peak and running from 720 (red, left end of wall) to 570 (yellow, right end of wall). The arcs are 01, 13, 32, 20 and the self-loops 00, 11, 22, 33. Outputs chosen from a two-letter alphabet $\{r, y\}$ are emitted on arcs (Mealy machine) rather than at states (Moore machine) according to the following rule: the 00, 01, 20, and 11 arcs emit r, the 33, 32, 13, and 22 arcs emit y. Euclideanity is expressed by dividing the input range in three non-overlapping intervals: the machine receives input in the [720-620] range it settles in state 0, if the input is in the [570–590] range it goes to state 3, and in the 'orange' range (590-620) it will stay in state 1 if previously it was in state 1, and in state 2 if previously it was in state 2.



If we provide input to this EA with slowly decreasing wavelengths λ running from 720 to 570 nanometers, it will move from state 0 to state 1 at $\lambda = 620$, and from there to state 3 at $\lambda = 590$. The output switches from r to y when the 13 arc is first used, at $\lambda = 590$. When we perform the opposite experiment, increasing wavelengths from 570 to 720 in small increments, the EA will switch from y to r as it passes from 2 to 0 at $\lambda = 620$. In the entire orange region, the model shows hysteresis: if it came from the red side it will output red, if it came from the yellow side it will output yellow.

The heap is an important philosophical issue on its own right, but we must leave its full discussion to another occasion, confining ourselves to a couple of remarks. First that on the EA account the sorites paradox is not an edge phenomenon, restricted to some critical point when the non-heap becomes a heap and red becomes yellow (Sainsbury 1992), but something that characterizes a substantive range of parameters with non-zero measure. Second, that the hysteresis seen in the example EA is consistent with perception studies on single parameter spaces (Schöne and Lechner-Steinleitner 1978, Poltoratski and Tong 2005, Hock et al. 2005).

The real take-home lesson from the heap, as far as our current task of accounting for internal conflict is concerned, is that such conflicts can be modeled as nondeterministic states sharing the same range of input parameters. If we recast the paradox of the painted wall in terms of moral precepts, we see the conflict emerging between two, in themselves very reasonable maxims:

Factuality I ought to report things as I see them

Consistency I ought not report differences where I don't see any

Importantly, the conflict arises even though we see the first precept as superior to the second one. Consistency is at best a refinement of Factuality, and we have a large number of warnings attached to it, from *Si duo faciunt idem, non est idem* to Emerson's famous quip 'A foolish consistency is the hobgoblin of little minds'. Eventually, if λ is made small enough, we sacrifice Consistency and say "No" because we can't live with a strong violation of Factuality. Before turning to a more detailed analysis in the next Section, let us first define EA, ETs, and EEMs.

Definition 1.1 A *Euclidean automaton* (EA) over a parameter space P is defined as a 4-tuple $(\mathcal{P}, I, F, T)$ where $\mathcal{P} \subset 2^P$ is a finite set of states given as subsets of P; $I \subset \mathcal{P}$ is the set of initial states; $F \subset \mathcal{P}$ is the set of accepting states; and $T : P \times \mathcal{P} \to \mathcal{P}$ is the transition function that assigns for each parameter setting $\vec{v} \in P$ and each state $s \in \mathcal{P}$ a next state $t = T(\vec{v}, s)$ that satisfies $\vec{v} \in t$.

Definition 1.2 A *Euclidean transducer* (ET) over a parameter space P is defined as a 5-tuple $(\mathcal{P}, I, F, T, E)$ where $\mathcal{P}, I, F$, and T are as in Def. 1.1 and E is an emission function that assigns a string (possibly empty) over a finite alphabet Σ to each transition defined by T.

Definition 1.3 A *Euclidean Eilenberg Machine* (EEM) over a parameter space P is defined as a 5-tuple $(\mathcal{P}, I, F, T, R)$ where $\mathcal{P}, I, F$, and T are as in Def. 1.1 and R is a mapping

$P \times \mathcal{P} \to P$ which assigns to each transition a (not necessarily linear, or even deterministic) transformation of the parameter space.

We have already seen examples of EA. A particularly relevant ET example is a vector quantizer (Gersho and Gray 1992), and if $P = \mathbb{R}$, an AD converter. Since Eilenberg machines (Eilenberg 1974) are less well known, we discuss the simplest cases individually. For $|\mathcal{P}| = 1$ we have a single mapping $P \to P$, and for $|\mathcal{P}| = k$ we have a finite family of $P \to P$ mappings. As the sets P_i collected in $\mathcal{P}$ may be overlapping, there is no guarantee that the mappings together describe a function (as opposed to a relation) over P, and even in the locally deterministic case EEMs are capable of realizing multivalued functions. Another example is

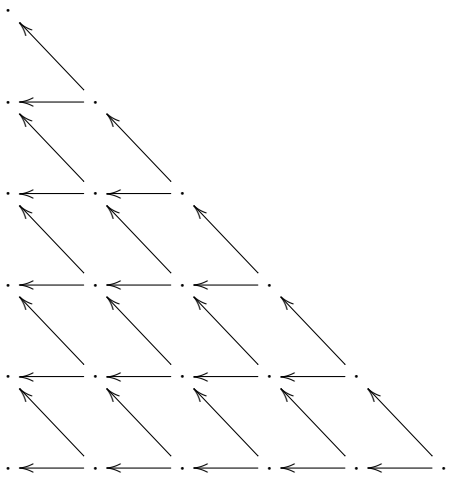
Example 4. The Artificial Neuron The elementary building blocks of Artificial Neural Networks (ANNs), both with sigmoid squishing and without, can be conceived of as two-state EEMs. The parameter space has d dimensions where d counts the number of inputs (dendrites), and the operation of the EEM is deterministic: if the sum of the inputs is smaller than the threshold (after squishing in a sigmoid AN, or without squishing in a linear AN) the unit gets in state 0, otherwise it gets in state 1. The output function is constant 0 in state 0, 1 in state 1.

Notice that ANs can also be conceptualized as ETs, with output alphabet $\Sigma = \{0, 1\}$, and inputs taken from Σ^d – this is because in standard ANs the outputs do not depend on the details of the input vector, just on the state it transitions to. In general, where there is no need to distinguish the subtypes, or the subtype is evident from context, we will speak of Euclidean Machines (EMs) as a cover term for EA, ETs, and EEMs.

The conflicted state

EA are rather limited computational devices, yet they have enough power to serve as homunculi in our model of decision-making. In what follows, we think of EA as receiving input in discrete time, but this is not essential for reaching, and maintaining, conflicted states. We will study asynchronous networks composed of EA, with particular attention to serially connected EA $A_1, A_2, \ldots, A_k$ where each A_{i+1} receives as its input the output of A_i, possibly cyclically. As our first case, we look at the case $k = 2$, the kind of direct conflict familiar from 'Never Give Up' cartoons:

We need two EA to model the situation, Frog and Stork, which we can assume to be isomorphic. At time t, each can be represented by two parameters, $p(t)$ corresponding to the power reserve it has, and $q(t)$ corresponding to the pressure it exerts on the other. We are less interested in the death spiral F and S can found themselves than in paths to disengagement, if there are any. We assume that for each party its $p(t+1)$ depends on the other's $q(t)$ deterministically, and that each party can set its $q(t+1)$ nondeterministically between 0 (standing down) and its own $p(t)$ (maximum effort to kill the other). If we take the abscissa p and the ordinate q_f, the skeleton can be depicted as follows:



The Stork in (p, q_f) space

At every point, the Stork has an option of applying as little force as it wishes, but no more than its power reserve. This choice is free in the sense of moral philosophy, it is only at the edges of the diagram that we see compulsion (deterministic behavior). The nondeterministic choices provide room for a broad variety of strategies ranging from escalation through tit for tat to turning the other cheek. If we couple an escalating Frog to an escalating Stork we obtain the death spiral discussed above, and importantly, a tit for tat player will also die as long as the other party relentlessly ratchets up the pressure – the only recourse of the non-aggressor is to take the aggressor with them to the grave.

'Never Give Up' is closely related to, but not identical with, the better known 'War of Attrition' game introduced by J. Maynard Smith (1974), the most salient difference being that in wars of attrition the resource (wait time) of the players is infinite while here the resource (power reserve) that the players start out with is finite, and may even be known in advance. We will not pursue a full game-theoretic analysis here, as our chief concern is not with the possible outcomes at the individual or population level, but rather with formalizing the moral calculus that can operate within the domain of free will. For this the simple Stork model is insufficient, as it lacks the critical variables corresponding to the hopes and fears of the player. A central idea of the paper is that such hopes and fears are simply internal models of the diagram edges, but before we turn to this in the concluding

section, let us take a closer look at the next simplest case, $k = 3$, known in popular culture as a Mexican standoff.

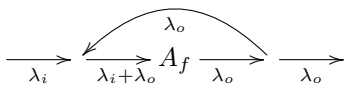
The pioneers of cybernetics were already aware of the *circularity of value* anomaly, exhibited e.g. by rats starved both for sex and for food: they prefer sex to exploration, exploration to food, and food to sex. If we model different drives by different agents, circularity of value anomalies boil down to Condorcet's paradox, but one does not need to subscribe to a society of mind assumption to see McCulloch's (1945) point that such circular preferences are "sufficient basis for categorical denial of the subsumption that values were magnitudes of any kind". (For the converse, that transitivity plus some additional assumptions are sufficient for expressing preferences in terms of utility functions see von Neumann and Morgenstern 1947.) Circularity of value is seen in many settings besides economics (McCulloch mentions neurophysics 'conditioned reflexology' and experimental aesthetics) and they demonstrate rather clearly that utility-based models are too simplistic for describing the behavior of rats, let alone those of humans or AGIs.

McCulloch's original model of the phenomenon does not lend itself to easy reproduction in terms of our contemporary understanding of networks, which no longer conceptualizes behavior in terms of reflex loops. The Euclidean Machines advanced here have the advantage that their main features can be analyzed without reference to recurrent behavior or nuances of timing. For $k = 2$ and 3 only cyclic conflict models are available, but for $k \geq 4$ we obtain a broader variety by optionally adding chords to the main cycle. Taking into account which parameters in the input of A_i are output by A_j we obtain a rich typology of conflict. We begin with the simplest case, the 4-state machine of our Example 3, which represents conflicted behavior in a forced binary choice.

To see how this conflict is created, consider two homunculi, A_f in charge of factuality, and A_c in charge of consistency, with A_c the weaker of the two, so that in a game of Never Give Up A_f will eventually win. Without consistency, A_f by itself is not particularly conflicted: it will opt for red when the input wavelength is sufficiently large, say at $\lambda > 620$, and for yellow when $\lambda < 590$. The simplest approach is to represent this by a linear function $y = (\lambda - 605)/15$, which is -1 or less at the unambiguously yellow range, $+1$ or more in the unambiguously red range. Many alternate functions could be considered (radial basis neural nets are a very attractive possibility) but we would like to see the qualitative emergence of conflicted states without fine-tuning the network response. Skeletonizing A_f leads to a simple two-state automaton, outputting r in the 605 to 720 range; y in the 570 to 605 range. The behavior at the boundary of the attractor basins (which we take to be 605) is irrelevant not just because this is a zero measure set, but because this behavior is completely shadowed by hysteresis.

Sainsbury and Williamson set up the protocol taking particular care that A_c, the guardian of consistency, is always aroused: "the window is so small relative to the strip that in no position can you tell the difference in colour between what the two segments expose". At the beginning (left side, $\lambda = 720$), A_c is inactive, and A_f simply outputs r. As λ is decreased, say in 1 nanometer decrements, though the exact number is irrelevant, A_c will be active, and will always pull the decision toward the last decision, whatever it was, with force $c < f$. Skeletonizing A_c is a much more interesting issue, since in general we would need to endow this automaton with two memory registers, one to store the last output whose consistency is to be maintained, and one to store the last input to see if we are close enough that consistency is required to begin with. For an increment of 1 nm and two outputs, this would require $2 \cdot 151$ states, which is unattractive both because this number is too large, and because it is inversely proportional to the step-size, a small but arbitrary parameter unlikely to be critical for our understanding of the problem. A more attractive solution is to conceptualize A_c as an EEM, with only three states, 'neutral', 'sticking to red', and 'sticking to yellow' with three transformations of the inputs. The identity function is attached to the neutral state where two subsequent inputs are too far apart for consistency to make sense, a 'red boost' function of $+15$ is attached to the 'sticking to red state, and a 'yellow boost' function of -15 is attached to the 'sticking to yellow' state.

A subtle but important distinction from simpler additive models is that A_c is seen as manipulating the *input* of the main binary classifier A_f, rather than contributing to, or even reversing, its output. Once this is understood, we can further simplify A_c by removing its memory (third state) and assuming that it just adds back the output of A_f to its input when the unbiased input is seen as close to what it was before. For doing that, we need to address another property that the standard treatment of networks generally abstracts away from, seminumericity. A_f, as we defined it so far, takes numeric input (wavelengths measured in nanometers) from 570 to 720, and produces symbolic outputs r and y. One approach would be to freely rescale numerical values between 0 and 1 (activation level), or between -1 and $+1$ (including inhibitory effects). Textbook treatments on neural networks generally opt for this solution, without much discussion of the costs attendant to rescaling, and simply pave over the difficulties of replacing categorical variables like *red/yellow* by pseudo-numerical values such as the ± 1 we used above. Historically, the subtle interplay between the deductive and the numerical approach is well understood from the numerical side (the entire second volume of Knuth 1969 is devoted to this issue), what is called for now is a better understanding of the semi-symbolic nature of biological computation.



A_c as a feedback loop modifying A_f

The approach proposed here, inspired by semantic ideas from cognitive science (Rosch 1975), is to recast the symbolic output of Euclidean automata and transducers to numeric, e.g. to assume that the output of the classifier will be the prototypical red, say 630, or the prototypical yellow, say 580. In the hysteresis case, as we decrease the input wavelength from 720 to below the boundary point at 605, say to 600, A_f would now report yellow, but because the previous

reports were all red, the input it sees is not 600 but 630 since the previous output was mixed in by A_c. In fact, raw input has to go below 580 for the mixture to get below 605, and in the intermediate range we observe hysteresis. Conversely, if we start from low wavelengths, we need to get above 630 to get away from the yellow and have the system switch to red. By adjusting the mixture weights it would be possible to increase or decrease the range of hysteresis, in the extreme case to a point that a machine once committed to an answer will never depart from it. This is obviously maladaptive in a system of perception, but would make perfect adaptive sense in a unit dedicated to memory.

Conclusions

We have introduced Euclidean Machines, a slight generalization of the classical finite automata, transducers, and machines, and sketched how simple, but typical conflict cases can be described in terms of these. There are many other potential applications, such as modeling reasoning with EA as partially hidden information gets uncovered, but these would stretch the bounds of this paper.

Perhaps the greatest value of EA lies in the fact that they enable robust anthropocentric use of moral vocabulary. We hold, with McCarthy (1979) that

> to ascribe certain *beliefs, knowledge, free will, intentions, consciousness, abilities* or *wants* to a machine or computer program is legitimate when such an ascription expresses the same information about the machine that it expresses about a person. It is useful when the ascription helps us understand the structure of the machine, its past or future behavior, or how to repair or improve it

and indeed see our enterprise as "to do what Ryle (1949) says can't be done and shouldn't be attempted – namely, to define mental qualities in terms of states of a machine". Being in a conflicted state comes out, unsurprisingly, not as a single state of the EA, but rather as a set of nondeterministic states *tied together by their shared territory of input parameters*. The EA framework enables making commonsensical moral judgments about actions (state transitions) such that Frog can *relentlessly* ratchet up pressure on Stork, or that reporting 'red' or 'yellow' is indeed a case of conflict between two virtues, being factual and being consistent, and so forth.

Another key linguistic area opened up by the EA framework is the study of *hopes and fears*. Since much, perhaps too much, of our decision process is driven by our hopes and fears, some formal mechanism to deal with these is necessary for any attempt at AGI design. Putting oneself in the place of Stork and Frog, it is evident that (within the confines of this conflict) their fears are concentrated at the edge of the parameter space where $p(t) = 0$.

The classical finite state machinery (McCulloch and Pitts 1943) does not fully capture McCulloch's own ideas about neural nets. In particular, the inhibitory and excitatory mechanism are hard to capture without paying more attention to the largely neglected but conceptually nontrivial issues of scaling and thresholding. Since EA can model standard (sigmoid) NNs, a simple first step in generalizing the modern theory of ANNs to EA could be the transfer of standard training algorithms such as backprop to this domain. Housebreaking the cybernetic turtle of Grey Walter is now within reach.

Acknowledgments

We are grateful to Tibor Beke (UMass) and Peter Vida (Mannheim) for trenchant remarks on an earlier version. Work partially supported by OTKA grant #77476.

References

Eilenberg, S. 1974. *Automata, Languages, and Machines*, volume A. Academic Press.

Gersho, A., and Gray, R. M. 1992. *Vector Quantization and Signal Compression*. Springer.

Graham, A. 1989. *Disputers of the Tao*. Open Court.

Hart, D., and Goertzel, B. 2008. OpenCog: A software framework for integrative artificial general intelligence. In *Proceedings of the First AGI Conference*.

Hock, H.; Bukowski, L.; Nichols, D.; Huisman, A.; and Rivera, M. 2005. Dynamical vs. judgmental comparison: hysteresis effects in motion perception. *Spatial Vision* 18(3):317–335.

Knuth, D. E. 1969. *The Art of Computer Programming. Vol. II: Seminumerical Algorithms*. Addison-Wesley.

McCarthy, J. 1979 (1990). Ascribing mental qualities to machines. In Lifschitz, V., ed., *Formalizing common sense*. Ablex. 93–118.

McCulloch, W., and Pitts, W. 1943. A logical calculus of the ideas immanent in nervous activity. *Bulletin of mathematical biophysics* 5:115–133.

McCulloch, W. 1945. A heterarchy of values determined by the topology of nervous nets. *Bulletin of mathematical biophysics* 7:89–93.

Minsky, M. 1986. *The Society of Mind*. Simon and Schuster.

Poltoratski, S., and Tong, F. 2013. Hysteresis in the perception of objects and scenes. *Journal of Vision* 13(9):672.

Rosch, E. 1975. Cognitive representations of semantic categories. *Journal of Experimental Psychology* 104(3):192–233.

Ryle, G. 1949. *The concept of mind*. University of Chicago Press.

Sainsbury, M., and Williamson, T. 1995. Sorites. In Hale, B., and Wright, C., eds., *Blackwell Companion to the Philosophy of Language*. Blackwell.

Sainsbury, M. 1992. Sorites paradoxes and the transition question. *Philosophical Papers* 21(3):77–190.

Schöne, H., and Lechner-Steinleitner, S. 1978. The effect of preceding tilt on the perceived vertical. Hysteresis in perception of the vertical. *Acta Otolaryngol.* 85(1-2):68–73.

Smith, J. M. 1974. Theory of games and the evolution of animal conflicts. *J. Theoretical Biology* 47:209–221.

von Neumann, J., and Morgenstern, O. 1947. *Theory of games and economic behavior*. Princeton University Press.