ChatGPT

Paraconsistency, Ambiguity, and Multi-Hypothesis Reasoning in Cognition and UI Design

Paraconsistent Logic and Contradictory Beliefs in Cognition

Human reasoning tends to avoid holding explicit contradictions at the same moment – cognitive dissonance causes stress, and people quickly try to resolve or discard one of the opposing beliefs [1] [2]. However, in building robust cognitive models (and by extension robust UIs), we can draw inspiration from **paraconsistent logic**, a family of formal logics that tolerate contradictions without "exploding" into triviality. Paraconsistent logics were developed specifically to handle contradictory information in a controlled, non-trivial way [3] [4]. Unlike classical Boolean logic (where any contradiction makes the system infer anything), paraconsistent frameworks allow a system to **accept inconsistency** and continue reasoning soundly [5] [6].

One well-known example is **Belnap's four-valued logic** (and related Paraconsistent Annotated Logics), in which a proposition can take one of multiple truth-values beyond just True or False [7]:

- **True (T)** – definitely true (and not false).
- **False (F)** – definitely false (and not true).
- **Both (T ∧ F)** – *true and false at once*, a contradictory state (dialetheia) [7].
- **Neither (⊥)** – *neither true nor false*, an indeterminate or unknown state [7].

Such multi-valued logics provide formal "operators" to handle overlap and ambiguity. They violate the law of the excluded middle and **principle of explosion** (ex contradictione) on purpose, so that a statement can be marked as both true and false without the entire system collapsing [8] [7]. In practice, this means a reasoning system (or UI state machine) could flag a condition as *"contradictory"* instead of forcing a binary choice. For instance, a UI element might be in a superposed state of **both "on" and "off"** (if it has conflicting inputs) – and the underlying logic would treat this as a legitimate third state, not an error.

Crucially, paraconsistent approaches have been explored in cognitive science and AI to model how an agent might handle contradictory evidence. In AI, researchers have implemented **paraconsistent artificial neural networks** that integrate these logics. For example, an artificial neural network based on paraconsistent logic was used to detect Alzheimer's disease from noisy, conflicting medical data [9]. Its ability to reason with contradictory inputs (without immediately rejecting them) yielded promising results [9]. Similarly, algorithms like *PAL2v* (Paraconsistent Annotated Logic with 2 values) have been used to build neural "cells" that learn by *contradiction extraction* – effectively parsing out and gradually resolving conflicts in sensor information [10] [11]. These systems start by identifying the largest inconsistency between signals and work toward a **consensus value** rather than throwing an error [11]. This is directly analogous to a UI or decision system that, when faced with conflicting telemetry or user input, can **maintain both signals in parallel and seek a resolution** instead of outright failing.

In summary, **dialetheic models** (which admit true contradictions) and paraconsistent logics give us a blueprint for representing overlapping or paradoxical states. They demonstrate that a well-designed cognitive system – or interface – can hold "p and ¬p" simultaneously in a *marked* way. For a UI, this suggests we can have a **"contradiction state"** (visually indicated, e.g. with a special icon or color) to represent overlapping/ambiguous status, preserving both pieces of information. Rather than force an immediate choice, the system can flag the conflict and continue operating with both values present. This approach is supported by formal logic and could reduce user confusion compared to silent failures or arbitrary resolution of the conflict.

Predictive Coding and Coexisting Hypotheses in Perception

The brain often entertains **multiple hypotheses** in perception and gradually reconciles them – an idea central to modern **predictive coding** (predictive processing) theories of cognition. In predictive coding frameworks, the brain builds hierarchical generative models that predict sensory inputs; perception is the result of these predictions tuned by incoming data (prediction errors) [12] [13]. Notably, when sensory information is ambiguous or noisy, the brain does not commit to a single interpretation right away – it carries *several* plausible predictions concurrently, effectively a **superposition of hypotheses**, until sufficient evidence accumulates to favor one interpretation.

Empirical phenomena like **multistable perception** (e.g. the Necker cube or Rubin's vase illusion) showcase this principle. In binocular rivalry or other ambiguous stimuli, observers alternately perceive one interpretation and then another, with occasional moments of uncertainty or fusion. A predictive coding account explains this as follows: at any given time one hypothesis is dominant (one interpretation is consciously perceived), but the suppressed alternative is still represented in the system as a latent prediction that hasn't been fully eliminated [13]. Small residual prediction errors from the "losing" interpretation build up over time until they trigger a switch in perception [13]. In other words, **the brain keeps competing explanations active in parallel** – a dynamic competition rather than a single locked state.

Neuroscientific models support this view. For example, Hohwy et al. (2008) and others argue that during binocular rivalry, the brain's higher-level visual areas encode both possible percepts; the currently inactive one still generates *prediction error signals* because the sensory input partially "fits" an alternate explanation [13]. These error signals gradually weaken the confidence (precision) of the current hypothesis, eventually causing a flip to the alternative percept [14] [15]. At the point of flipping, the brain's internal belief can be said to *relax into a mix* – effectively **equally weighting both interpretations** just before one wins out [14] [15]. Indeed, experimental participants sometimes report a transient "mixed" percept or uncertainty during such tasks, rather than a crisp either/or perception [16]. This **coexistence interval** is precisely what a UI could represent as an ambiguous state (e.g. showing overlapping visuals or a "schrödinger" icon) before clarity is reached.

Furthermore, advanced predictive-processing models suggest the brain may use a **mixture-of-experts** strategy: multiple concurrent generative models that each propose a hypothesis about the causes of sensory input. Instead of a single Gaussian belief distribution, the brain might maintain several discrete "peaks" – distinct hypotheses each with its own confidence level [17]. For instance, upon hearing a knock at the door, your mind might strongly consider "it's Alice" and separately "it's Bob" as two parallel predictions, rather than averaging them into a vague composite [17]. Each hypothesis is like an expert model with a binary prediction ("Alice is at the door" vs "Bob is at the door") accompanied by a confidence estimate [18]

[17]. This perspective aligns with Marvin Minsky's *Society of Mind* and multiple-model approaches in AI, where the brain is seen as **multiple agents or modules** negotiating a result. In predictive coding terms, it means top-down signals might carry several distinct *predictions* at once (each tagged with precision/confidence), and bottom-up signals report back errors for each [18] [19].

Key takeaways for design: The mind's ability to entertain coexisting hypotheses suggests UIs can do the same. Instead of immediately forcing one interpretation of uncertain data, the interface could present **multiple possibilities concurrently**. For example, a AR/MR (augmented reality) interface might display two faint overlapping outlines for a detected object when the system is unsure which it is – reflecting two active predictive models. As more data comes in, one outline could solidify while the other fades. Importantly, this wouldn't be seen as a mistake, but as a *normal state of inference* (much like the brain's perceptual ambiguity state). Users could even be given subtle controls to "nudge" the hypothesis they think is correct – akin to adjusting the brain's prior – for instance, selecting which overlay seems more plausible to them, thereby training the system's prediction. By mirroring the brain's **graceful handling of ambiguity**, the UI becomes more transparent about uncertainty and avoids sudden jumps or oscillations that might startle or confuse the user.

Non-Classical Logic in Neural Networks and AI Systems

Classic two-valued logic often forces a strict true/false resolution that doesn't reflect the gradations or conflicts in real-world data. **Non-classical logics** – including paraconsistent logic, fuzzy logic, and other multi-valued systems – have therefore been embraced in AI to make algorithms more tolerant of ambiguity and inconsistency. In neural networks, this can mean either integrating logic at a structural level or using logic-inspired **truth values** in the processing of signals.

One line of research, especially by scholars in Brazil and elsewhere, created **Paraconsistent Artificial Neural Networks (PANN)** that incorporate paraconsistent reasoning at their core. These networks use neurons or units that output not just a probability between 0 and 1, but an **annotated pair of values** representing evidence *for* and *against* a proposition. In effect, a single neuron can signal "this is somewhat true and somewhat false at the same time." By doing so, the network can hold contradictory indications without nullifying them. For example, Lopes et al. (2011) implemented a neural network with a paraconsistent logic layer to diagnose Alzheimer's; it was able to handle conflicting medical indicators (some tests suggesting disease, others not) and still reach a decision without the inconsistency causing a system crash [9]. The paraconsistent layer would mark certain inputs as conflicting and guide the network to treat them with caution rather than simply averaging them out or ignoring one. This demonstrates the **practical benefit**: the AI didn't have to throw away data or blindly choose one source over another – it could absorb *both* sides of contradictory evidence and analyze patterns in that state [9].

Similarly, frameworks like **Paraconsistent Annotated Logic (PAL)** define a lattice of truth values that neural systems can use when aggregating signals. In a PAL-based system, each piece of data might come with a degree of truth μ and a degree of falsity λ. A fully consistent piece of evidence might be $\mu = 1, \lambda = 0$ (totally true) or $\mu = 0, \lambda = 1$ (totally false). But a contradictory signal could be $\mu = 0.7, \lambda = 0.8$, meaning there is strong evidence both for and against – a conflict the system will register rather than discard. Researchers have shown that by applying algorithms to **extract the "effect" of contradiction** gradually, such systems can still converge on useful outputs [20] [21]. For instance, a paraconsistent algorithm may start by identifying which two inputs have the highest degree of mutual contradiction and then reducing that contradiction step by step (e.g. by discounting one source slightly or finding a compromise value) [11].

In the end, the system arrives at a *consensus value* that best reconciles the inputs [11]. The key is that during the process **the contradictory inputs are both retained** – they are marked and manipulated but not thrown out. This approach is ideal for domains like sensor fusion (e.g., combining data from EEG, heart rate, and user reports in a wellness app) where signals might not always agree. Instead of programming "if readings disagree, raise error," a paraconsistent network-based system would continue operating, showing perhaps an intermediate reading or a range, and internally working to figure out the best estimate.

Beyond paraconsistency, **fuzzy logic** is another non-classical paradigm influential in cognitive-inspired design. Fuzzy logic allows truth values anywhere in [0,1], effectively representing degrees of truth. While not aimed at contradictions per se, fuzzy systems handle *vagueness* and partial truth well – for example, a rule like "overlay is **kind of** visible" can be represented. In UI terms, fuzzy logic could let an element be in a graduated state (0.5 visible) rather than a hard show/hide, reflecting uncertainty or partial activation.

The big picture is that **connectionist models don't have to be strictly binary in their representations of state**. By using multi-valued or non-monotonic reasoning rules, neural networks can emulate a form of **common-sense tolerance**: they can hold multiple pieces of information that *logically conflict* and still function. This is analogous to how humans might compartmentalize conflicting beliefs or how societies handle conflicting information sources without "exploding" into incoherence. For UI/UX design, this suggests technical implementations like a **tri-state flag** for UI elements (true/false/"both") or an internal representation of UI state as probabilities/confidences rather than booleans. The UI could then render elements in proportion to their truth value (e.g., a button might appear with 50% opacity if it's in an "uncertain" activation state). Internally, adopting a paraconsistent approach means the system's state machine would not assert inconsistent states are impossible – instead, it explicitly checks for them and routes to a *distinct state handler* (one that might log the conflict and await further input). This leads to more resilient UIs that mirror the brain's and AI's ability to handle real-world messiness.

Incongruity, Paradox, and Dual Frames in Creative Cognition

Creative cognition – exemplified by humor, metaphor, and imaginative play – often **thrives on paradox and the coexistence of incompatible frames of reference**. A joke, for instance, sets up one interpretation (frame) and then suddenly reveals a contradictory interpretation, and the *simultaneous presence* of those two meanings is what makes it funny [22]. Cognitive scientists describe humor processing in terms of **incongruity detection and resolution**. In Suls' two-stage model and Attardo & Raskin's theories, the mind initially activates a normal scenario or schema, then encounters a cue (punchline) that *doesn't fit*, creating an incongruity [23]. Crucially, at that moment of recognizing the conflict, both the original interpretation and a nascent alternative are present in mind. The resolution stage is when the brain finds a new frame that makes the cue make sense, often by flipping a key assumption [23] [22]. The **spark of humor** comes right at the junction where the two frames overlap – you momentarily hold the "straight" interpretation and the absurd or hidden interpretation together, appreciating their clash.

For example, consider the pun or classic joke given in one study [24] [25]: A father tells his son to look at four trees, saying an alcoholic would see eight. The son replies there are only **two** trees – revealing the father actually saw double. The humor arises when the reader/listener realizes the father is drunk. Initially, one parses the dialogue as a sincere lesson (frame 1: sober father). The son's reply introduces a contradiction (only two trees, not four; frame 1 cannot explain this). The brain then retrieves frame 2: the father is impaired, causing him to *believe* four trees are present [22]. At the punchline, the two frames ("sober teaching" vs "drunken hallucination") are entertained at once – we mentally compare them, see how the situation flips, and laughter ensues as the incongruity is conceptually resolved by adopting the new frame. Cognitive linguistic theories like **Frame-shifting** and **Conceptual Blending** formalize this: we often blend two disparate frames to create novel meaning, and humor is a case where the blend initially seems impossible (a contradiction) until a clever twist binds them [26].

Beyond jokes, many creative insights and metaphors are born from holding conflicting ideas together. Arthur Koestler, in *The Act of Creation*, called this *bisociation*: the intersection of two self-consistent but normally incompatible frames of thought. For instance, a designer might consider a UI element that is both **a button and a container** at the same time – a paradox in standard UI logic. By keeping that dual notion alive, one might invent a new UI component (e.g., a panel that can be clicked like a button for quick action, yet also expands to hold content). In the arts, visual illusions or surreal paintings deliberately force the viewer to see two realities at once (e.g., the famous young lady/old hag ambiguous figure, or Escher's impossible objects). The *experience* of these works comes from the mind's attempt to reconcile the irreconcilable – effectively enjoying a controlled cognitive paradox.

Implications for design modes: We can incorporate this principle by offering a **"creative" or "paradox" mode** in our systems. In such a mode, **dual states are not only allowed but emphasized**. For example, the UI might deliberately overlay two theme presets or two visualization modes on top of each other (perhaps with a shimmering or oscillating effect) to stimulate creative interpretation. This would mirror how a joke overlays meanings. In a design tool, one could enable a "blend frames" feature that shows two design alternatives superposed, encouraging the user to find a novel compromise – analogous to blending two mental spaces. In an educational app, instead of immediately correcting a child's wrong answer, the UI could momentarily display the child's answer and the correct answer together, prompting the child to compare and resolve the discrepancy (a form of guided incongruity resolution). By treating paradox as a *first-class citizen* in the interface, we acknowledge that sometimes the path to insight is through **sustained ambiguity** and toggling perspectives, rather than through immediate clarification.

Design Implications for the "Donut of Awareness" UX/Tech

Bringing these insights together, we can outline concrete ways to design the *Donut of Awareness (DoA)* interface and system to gracefully handle contradiction, ambiguity, and multiple simultaneous states. The goal is to make the UI cognitively natural (brain-inspired) and resilient, rather than forcing binary choices or error states prematurely. Here are key applications:

- **UI State Design – Embracing Ambiguous Overlays:** Allow interface elements to display **contradictory statuses simultaneously** when needed. For example, an overlay panel might be tagged as both "locked **and** previewing" if there's conflicting input about its state. Instead of this being an error, the UI would render a special *"superposed" visual state* – perhaps a combined icon or a subtle **shimmer effect** indicating duality. This corresponds to adding a third UI state beyond the usual binary. The visual language should make it clear that **both conditions apply at once** (just as paraconsistent logic holds $p \wedge \neg p$). Users thus see that the system is in a temporarily unresolved condition, not a broken one.

- **Multi-Valued "Membrane" Logic in Code:** Under the hood, implement tri-valued or multi-valued flags for key status variables (rather than simple booleans). For instance, a `predictionStatus` variable might take values `{ true, false, both, unknown }`. The state management (state.ui) can interpret “both” as a legitimate state and apply a distinct CSS class like `.state-contradictory` to style the element appropriately. This is analogous to having a **paraconsistent membrane** in the code that can hold overlapping predictions. The “membrane” (perhaps referring to the donut’s membrane metaphor) could use a small truth-table to decide outcomes: e.g., true + false = both (instead of error). Adopting this logic formally ensures no component crashes just because it received conflicting signals – it will propagate a **tagged contradictory state** that other components can handle. (Such an approach is inspired by multi-valued logic truth tables [7], ensuring every combination of inputs maps to a defined output state.)

- **Predictive Coding Visualization – Layered Hypotheses:** Represent the system’s multiple hypotheses about user intent or sensor input through the UI in a *transparent, visually intuitive way*. For example, if the DoA system is unsure whether the user’s “donut” (awareness focus) is in mode A or B, the interface can **show both possibilities lightly**. Perhaps two ghosted donut rings are drawn on the display, each corresponding to a candidate mode, with one slightly more pronounced if it’s currently deemed more probable. As more data arrives (or as time passes without certain confirmation), the UI can animate one ring strengthening and the other fading, reflecting confidence changes over time. This design echoes how the brain’s perception might gradually “collapse” onto one interpretation [14] [13]. It also keeps the user in the loop about uncertainty: rather than abruptly switching from A to B, the user sees a gentle transition. We can even let the user interact – e.g., clicking on a ghosted ring to tell the system “focus on this hypothesis,” akin to a **“bias slider”** that weights one predictive model over another. By visualizing competing predictions side by side, the UI treats ambiguity as informative, not as something to hide.

- **“Creative Paradox” Mode – Intentional Frame Blending:** We can introduce a mode or feature set in the DoA UX that deliberately leverages conflicting overlays for creative outcomes. For instance, a **dual overlay display** where two scene presets (say, a wireframe view and a full-color view) are shown superimposed. This could be presented as a special *paradox lens* or *humor mode* in the interface, meant for brainstorming or entertainment. The UI might, for a short period, allow normally exclusive modes to coexist (perhaps cycling or jittering between them) to spark the user’s imagination. This design is inspired by the incongruity-resolution mechanism: the user experiences a moment of “this doesn’t normally go together” and is invited to resolve or make sense of it. Such a mode could be useful in creative software where exploring an out-of-the-box combination leads to new ideas. Importantly, the transition out of the paradox can be timed or user-controlled – e.g., after a few seconds of overlap, the UI asks the user which layer they prefer, thus *resolving the incongruity* by user choice. This mimics how humor or a puzzle has a payoff after the tension of ambiguity [22].

- **Logging and Analytics – Preserve Conflicting Data:** From a technical perspective, all telemetry and analytics should be designed to **store conflicting metrics side by side** rather than choosing one and discarding the other. If the DoA system’s sensors or subsystems report two different values for what should be the same metric (for example, an EEG attention level vs. a behavioral metric of engagement), the data store would keep *both readings* linked, along with a measure of inconsistency. Analysis tools can then use paraconsistent algorithms to interpret these – for instance, computing an “inconsistency degree” or highlighting the timeframe where divergence occurred [5] [3]. The UI facing the developer or power-user could have a **“tolerance slider”** that sets how much contradiction is allowed before raising an alert. At a low tolerance, the moment two metrics diverge beyond a threshold, the system flags it (but still doesn’t crash – it just notes “contradiction detected”). At a higher tolerance, the system might ignore small contradictions and treat them as noise. This flexibility acknowledges that a bit of contradiction is normal in complex systems – just as humans can tolerate minor cognitive inconsistencies – and only when it exceeds a certain level do we need intervention. By logging everything, we avoid losing potentially valuable information that could explain system behavior. For example, if a user's self-report says they are calm but their physiological data says they are stressed, keeping both allows deeper analysis (maybe the user is unaware of their stress). A classical system might have thrown out the "outlier" – a paraconsistent-informed system will **keep the anomaly in play** as a signal to investigate.

In conclusion, designing the DoA UX with these principles means **treating ambiguity and contradiction as natural states** rather than bugs. This approach is backed by cognitive science findings (the brain's multi-hypothesis handling, the role of incongruity in thought) and by formal logic frameworks that give us the tools to implement it. The result should be a user experience that feels more **flexible, intelligent, and trust-worthy** – it doesn't hide uncertainty or force decisions too soon. Users can see multiple facets of a situation when needed, much like we often do in real life when juggling possibilities. Embracing a bit of paradox in the interface might just lead to a deeper sense of engagement and understanding, turning what used to be confusing "edge cases" into opportunities for insight and creativity.

Sources:

- Rudnicki, K. (2020). *Humans do not reason from contradictory premises: The psychological aspects of paraconsistency.* (Evidence that humans usually resolve contradictions quickly) [1] [2] .
- Marcondes, F. et al. (2021). *Neural Network XAI Based on Paraconsistent Analysis.* Electronics, 10(21). (Explains paraconsistent logic handling contradiction without explosion) [8] [7] .
- Da Silva Filho, J.I. et al. (2011). *Paraconsistent Algorithm Extractor of Contradiction Effects – ParaExtrctr.* J. of Software Eng. & Apps, 4(10). (Using Paraconsistent Annotated Logic to filter contradictory signals and reach consensus) [5] [3] .
- SEP – *Paraconsistent Logic* (2015). (Mentions ANN based on paraconsistent logic for Alzheimer's detection) [9] .
- Weilnhammer, V. et al. (2017). *A predictive coding account of bistable perception.* PLOS Comp. Bio, 13(5):e1005536. (Predictive coding model where suppressed alternative interpretations cause switching in perception) [13] .
- Alignment Forum (2020). *Integrating Three Models of Human Cognition.* (Notes that subconscious processes weigh multiple hypotheses for ambiguous stimuli, choosing one for consciousness) [27] [28] .
- Alignment Forum (2020). (Also discusses multiple simultaneous generative models vs. a single prior – a "society of mind" predictive processing) [18] [17] .
- Zheng, W. & Wang, X. (2023). *Humor experience facilitates ongoing cognitive tasks: Evidence from pun comprehension.* Frontiers in Psychology, 14. (Describes the incongruity-resolution model of humor with frame shifting) [23] [22] .
- Coulson, S. (2006/2015). *Space Structuring Model of Humor.* (via Zheng & Wang 2023) – Frame-shifting when new info conflicts with the active frame [26] .
- (Additional cognitive and design insights derived from the above and related theoretical frameworks in cognitive science and HCI.)

[1] [2] philsci-archive.pitt.edu
https://philsci-archive.pitt.edu/17154/1/Article.pdf

[3] [4] [5] [6] [10] [11] [20] [21] Paraconsistent Algorithm Extractor of Contradiction Effects - $Paraextrctr_{ctr}$
https://www.scirp.org/html/7763.html

[7] [8] Neural Network Explainable AI Based on Paraconsistent Analysis: An Extension
https://www.mdpi.com/2079-9292/10/21/2660

[9] Paraconsistent Logic (Stanford Encyclopedia of Philosophy/Winter 2015 Edition)
https://plato.stanford.edu/archives/win2015/entries/logic-paraconsistent/index.html

[12] [13] [14] [15] [16] A predictive coding account of bistable perception - a model-based fMRI study | PLOS Computational Biology
https://journals.plos.org/ploscompbiol/article?id=10.1371/journal.pcbi.1005536

[17] [18] [19] [27] [28] Integrating Three Models of (Human) Cognition — AI Alignment Forum
https://www.alignmentforum.org/posts/6chtMKXpLcJ26t7n5/integrating-three-models-of-human-cognition

[22] [23] [24] [25] [26] Frontiers | Humor experience facilitates ongoing cognitive tasks: Evidence from pun comprehension
https://www.frontiersin.org/journals/psychology/articles/10.3389/fpsyg.2023.1127275/full