# AI and Machine Learning In Nuclear Fusion

*Andrew Magdy Kamal Nassief*
*The Lonero Foundation*
*Quantum Engineering and Distributed Computing Systems*

**Abstract:** With the emergence of regressional mathematics and algebraic topology comes advancements in the field of artificial intelligence and machine learning. Such advancements when looking into problems such as nuclear fusion and entropy, can be utilized to analyze unsolved abnormalities in the area of fusion related research. Proof theory will be utilized throughout this paper. For logical mathematical proofs: n represents an unknown number, e represents point of entropy, and m represents maximum point, f represents fusion. This paper will look into analysis of the topic of nuclear fusion and unsolved problems as hardness problems and attempt to formulate computational proofs in relation to entropy, fusion maximum, heat transfer, and entropy transfer mechanisms. This paper will not only be centered around logical proofs but also around computational mechanisms such as distributed computing and its potential role in analyzing computational hardness in relation to fusion related problems. We will summarize a proposal for experimentation utilizing further logical proof formalities and the decentralized-internet SDK for a computational pipeline in order to solve fusion related hardness problems.
**Keywords:** Artificial Intelligence, Nuclear Fusion, Distributed Computing, Decentralization, Algebraic Topology, Logic,
Proof Theory, Data Visualization, Data Mining, Parallel Processing, Data Pipelines, Entropy, Computational Analysis, Logical Proof, Logic Proofs, Machine Learning



## 1.0 Problem Statement

Nuclear Fusion in terms of a problem needing to be solved in modern day engineering and alternative energy research is one of the hardest. Given things needed to take into account such as entropy and fusion reaction[1], mathematical techniques are becoming more and more critical in relation to this area of research. Magnetic confinement and inertia are key points to consider in relation to nuclear fusion research and having stable fusion related states[2]. AI has been optimized many times in the past for fusion related research, infact while I was doing research given my interest in parallel processing algorithms, I noticed earlier variants of researchers trying to make their own algorithms. This have included an ABC inspired colony algorithm[3] for fusion related research. The fact is many people want to build optimization algorithms[4] in relation to fusion, and solving nuclear fusion is considered a hidden treasure for nuclear physicist everywhere. Infact there are a series of different optimization algorithmic approaches including the BB-BC approach[5], sensory and network mechanisms[6], as well as design automation[7]. Outside of this, large amounts of data have hence remained unexplored. Given the fact that many algorithmic approaches are in the race to find the answer, it is quite obvious we aren't fully there yet. However, perhaps new approaches that integrate different mathematical strategies or more modern forms of computing may help us.

## 1.0(a) Approach

Our current approach is a proposition for a mathematically derived technique targeting the complexity of nuclear fusion. Nuclear Fusion as a problem has key points to consider in terms of how we as scientist can utilize logical proofs in order to analyze it. This is why in the case of unknowns or uncertainties, we must be able to communicate computationally in the most efficient way as possible. This is why for things such as entropy, energy transfer, stable states, we must consider it from the point of an algorithm trying to figure out uncertainties in a theoretically done experimentation phase.

## 1.0(b) Mathematical Proofs

First thereof, let us establish the main logical proof:

$$\forall(n) :\Leftrightarrow \lim_{e \to m}(f) \tag{1}$$

For all unknowns, we want to computationally derive as defined as, the limit of fusion from the points of entropy to the "maximum point" or point maximum.

In terms of recursion in uncertainties:

**Proposition 1** *for*

$$\begin{gathered} \forall(n) \\ \textit{We start by derving unknowns in the equation as a variable i.e the "unknown \#"} \\ \forall(n) :\Leftrightarrow lim(f) \\ \textit{We also want to hence set the logical equivalence} \\ \lim_{e \to m}(f) \end{gathered} \tag{2}$$

*And we need to make sure the limit is properly defined*

**Theorem 1**

$$\perp \exists(n) \tag{3}$$

*If there doesn't exist an unknown, you then derived your main formalities*

**Theorem 2**

$$\top \exists(n) \tag{4}$$

*When an unknown is existant, you move on to the equation of proposition* 1

**Definition 1**

$$\supset \forall(n) \tag{5}$$

*If and only if, you derive the unknowns, then the computer halts so*:

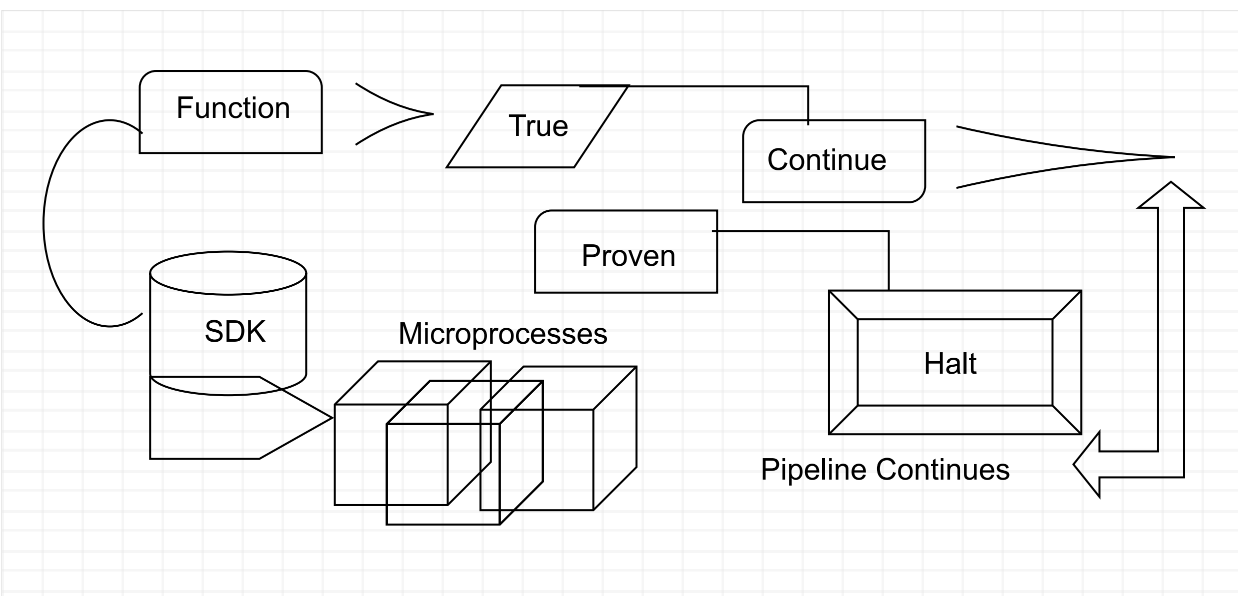
$$\supset \forall(n) :\Leftrightarrow lim(f) \ \vdash \ \rightarrow \ end$$

$$e \rightarrow m$$

*If and only if you when you have thorough unknowns as proven, then this halts*

## 2.0 Experimentation

One should have somewhat of a proposed experimentation approach. Given the extensiveness of the data needing to be observed in a hypothetical algorithm, this may be quite processing heavy. The reliance than on parallel processing and shared computing can likely be fulfilled by building an algorithmic pipeline through the decentralized-internet SDK.




Above shows an UML chart of the algorithmic pipeline design proposal centered around logical proof functionality.

### 2.0(a) Data Findings

For proposed data findings, data is derived and stored in relation to the problem at hand. That problem is in relation to nuclear fusion. We then have the following proposition in place post halt:

$$\models \forall(n) \equiv \exists \, end$$

$$Model \ all \ existent \ unknowns \ that \ are \ now \ knowns \ as \ defined \ at \ the \ end$$

At a computationally derived halting process, data can either be categorized into a database or formulate an output based off preferences coded in the hypothetical algorithmic pipeline. This allows you to quickly garnish the formulas needed in relation to said research.

## 3.0 Discussion

Viewing the problem of figuring out Nuclear Fusion through AI or by algorithmic means, as computational hardness, is unconventional. However, in order to conventionally tackle an unconventional problem, one likely needs to utilize lesser tested techniques. The algorithmic capabilities of this is built upon the idea of computing pipelines configured to solve this problem. Recursion and mathematical logic built likely through some regressional AI is one step, and the other is offloading the computational processing. When the two are added together, one is working on maximizing the efficiency of such research capabilities.

### 3.0(a) Technological Implication

The technological implication of utilizing a logical proof and algorithmic grid computing pipeline would likely allow the problem to be solved much quicker. There is even parallel processing mechanisms that go beyond the realm of classical computing. In such capabilities of creating said theoretical algorithmic pipeline, you can likely offload the processes much faster of recursively figuring out solutions to derived unknowns in the area of Nuclear Fusion.

### 3.0(b) Case Studies Discussions

Outside of the fact that many pipelines and programs have been built untop the decentralized-internet SDK, there are other grid computing networks such as BOINC[8] that harness the capabilities of parallel processing or shared computational processing in order to solve problems quicker. Harnessing the power of a grid computing pipeline with other data oriented structuring algorithms may even likely add to the efficiency. Outside of this, in the past people have tried parallel processing clustering algorithms for Nuclear Fusion, amongst other things. This is why likely many applicable case studies have existed. However, none utilized a recursive-proof and first order logic oriented regressional approach.

### 3.0(c) Further Derived Experimentation

I think the next phase is to build out a pipeline and garnish/harvest fast amounts of data in regards to the unknowns and "unknown proofs". Having the core logic, one then needs to approach the targeting of such data and perhaps the database structure or parsing method. This will hence allow someone to formalize solving our unknowns utilizing such computationally derived mathematical techniques.

## 4.0 Conclusion

Nuclear Fusion as a vastly unsolved problem may likely be solved utilizing unconventional techniques derived around logical proofs, targeting, and algorithmic/data collection pipelines. The proposal for a recursive mathematically derived pipeline that offsets heavy processing in relation to recursive logical proofs, can utilize parallel processing or grid computing networks. Such networks is where the decentralized-internet SDK comes in. Thinking of Nuclear Fusion in terms of a hardness problem with uncertainties may be the best way to actually target such problems in relation to entropy and fusion stability.

## 5.0 References


**[1]** How does the nuclear fusion of the sun constitute an increase in the entropy of its system? . Quara (2018). Available at: https://www.quora.com/How-does-the-nuclear-fusion-of-the-sun-constitute-an-increase-in-the-entropy-of-its-system. (Accessed: 29th September 2020)

**[2]** Nuclear Fusion Power. Nuclear Fusion : WNA - World Nuclear Association (2020). Available at: https://www.world-nuclear.org/information-library/current-and-future-generation/nuclear-fusion-power.aspx. (Accessed: 29th September 2020)

[**3**] A. Gómez-Iglesias, M. A. Vega-Rodríguez, F. Castejón, M. Cárdenas-Montes and E. Morales-Ramos, "Artificial Bee Colony Inspired Algorithm Applied to Fusion Research in a Grid Computing Environment," 2010 18th Euromicro Conference on Parallel, Distributed and Network-based Processing, Pisa, 2010, pp. 508-512, doi: 10.1109/PDP.2010.50.

[**4**] Greene, T. Scientists think we'll finally solve nuclear fusion thanks to cutting-edge AI. Neural | The Next Web (2020). Available at: https://thenextweb.com/neural/2020/04/27/scientists-think-well-finally-solve-nuclear-fusion-thanks-to-cutting-edge-ai/. (Accessed: 29th September 2020)

[**5**] Yalcin, Y., Pekcan, O. Nuclear Fission–Nuclear Fusion algorithm for global optimization: a modified Big Bang–Big Crunch algorithm. Neural Comput & Applic 32, 2751–2783 (2020). https://doi.org/10.1007/s00521-018-3907-1

**[6]** World Scientific Available at: https://www.worldscientific.com/doi/abs/10.1142/9789814313155/_0003. (Accessed: 29th September 2020)

**[7]** International Nuclear Information System Available at: https://inis.iaea.org/search/search.aspx?orig_q=RN:22027182. (Accessed: 29th September 2020)

**[8]** High-Throughput Computing with BOINC. BOINC Available at: https://boinc.berkeley.edu/. (Accessed: 29th September 2020)