FUSE: an open-source framework for tokamak power plant design and operations

O. Meneghini, T. Slendebroek, B. C. Lyons, T. F. Neiser, A. G. Ghiozzi, J. T. McClenaghan, N. Shi, S.Denk, M. Yoo, G. Avdeeva, G. Dose, L. Stagner, J. Harvey, J. Guterl, D. B. Weisberg, T. B. Cote, M. Clark, A. Zalzali, H. Anand, S. P. Smith, R. M. Nazikian, B. A. Grierson, J. Candy

DIII-D SET
Mar 27th 2025

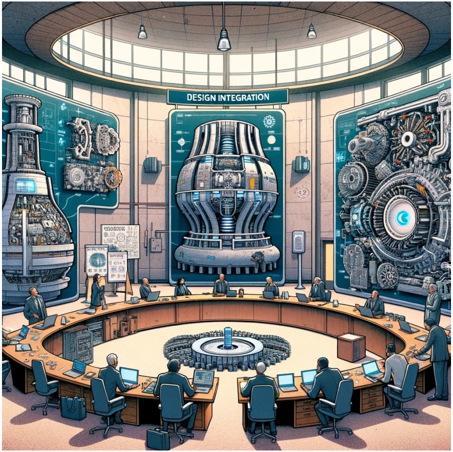
GENERAL ATOMICS

"Integration is when projects are most at risk of cost and schedule overrun" — US-GAO about NASA

Typical FPP design workflow:

1. Broad systems code analysis to identify one target design
   (Last time system optimized as integrated)
2. Experts run high-fidelity analyses on each sub-system
3. Issues!

- **Mismatched outcomes** between systems code and experts
  $\Longrightarrow$ Diverging teams
- **Premature design lock-in** limit exploration of alternative designs
  $\Longrightarrow$ Narrow scope
- **Manual integration** of disparate analyses into a cohesive design
  $\Longrightarrow$ Inefficient and costly



Why is it done this way?

What's the solution?

Solution is quite clear: Continuous integration and iterative design that grows in fidelity with the design maturity

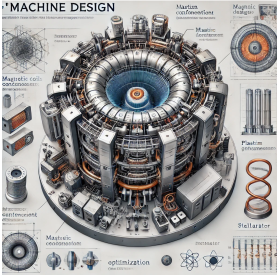
- Continuous integration implies
  - **Multi-physics** – Coupling plasma, controls, engineering, costing, risk
  - **Stationary & Dynamic** – Steady state or time evolving problems
- Iterative design implies
  - **Fast** – Rapid turnaround
  - **Scalable** – Expansive optimization and UQ studies
- Fidelity grows with the design maturity implies
  - **Fidelity hierarchy** – From quick and simple to first-principles but slow
  - **Extensible** – Evolving field, always new physics understanding

We don't do it this way because **we just don't have the right tools!**

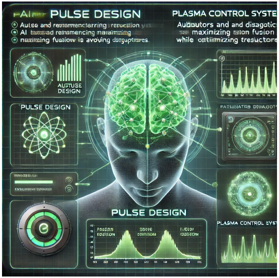
So let's build one! ... it's a theorist's dream!

With the right framework we should be able to (eventually) support all three main applications of integrated modeling

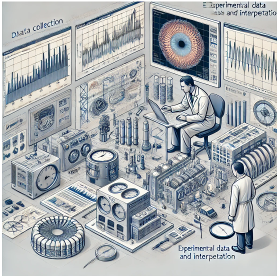
1 MACHINE DESIGN



2 PULSE DESIGN



3 DATA ANALYSIS



- Same theory-based models
- Same act./diag. models
- Same machine-agnosticity

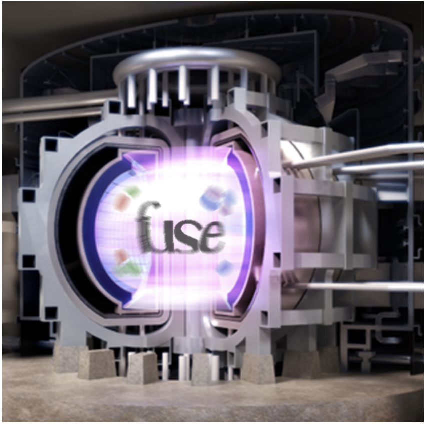
- Same integrated workflows
- Same data structures
- Same need for speed, always!

There's a great potential to exploit these synergies!

It is with this vision that FUsion Synthesis Engine framework was developed from the ground-up

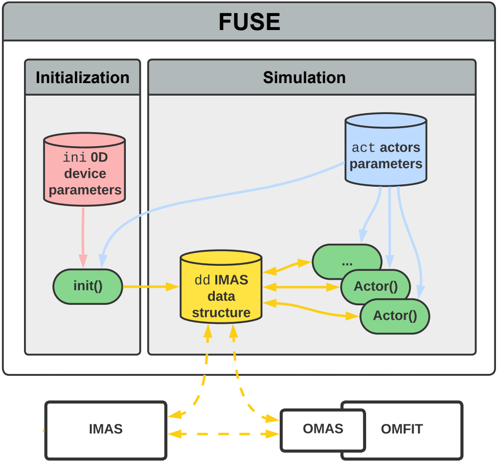
1. Born to support nascent **FPP design** industry
2. Now focusing on **time dependence and pulse design**
3. With future goal of enabling **integrated data analysis**

- Applying **lessons learned** from GA modeling expertise
  OMFIT, OMAS, STEP, TGYRO, TGLF-NN, EPED-NN, EFIT-AI, TokSys, GASC, ...
- All in one language: **Julia**
  - High-level like Python
  - As fast as C
  - Auto-differentiable
- Built around the **ITER IMAS** ontology



Few (strict) design rules enable high degree of modularity

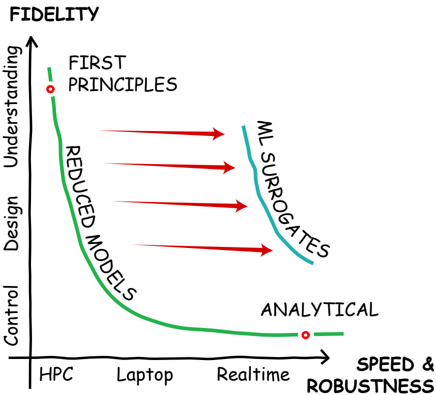
1. All data is stored in a centralized `dd` data structure (IMAS based)
2. Actors only talk via `dd`
3. Actor functionality set by `act` parameters
4. `dd` can be initialized from 0D `ini` parameters
5. FUSE interfaces to outside world only via `dd`



GENERAL ATOMICS

Whole fidelity spectrum is supported, but generally try to balance fidelity with speed and use ML when advantageous

- Want to **capture realistic system dynamics**
  1. Whenever possible, use of **physics-based** (reduced) models
  2. **Sufficient fidelity** to get interfaces between subsystems about right, so higher-fidelity simulations do not upend couplings
- While enabling **rapid design iterations**
  1. **Julia** for high performance
  2. **Tightly coupling** of models
  3. Break efficiency-fidelity tradeoff with **ML surrogates**



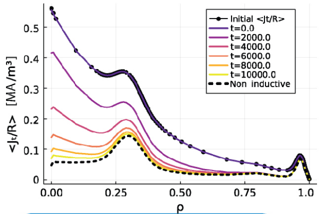
How to integrate models:

- (re)-write in Julia (preferred)
- In memory coupling
- File-based (last resort)

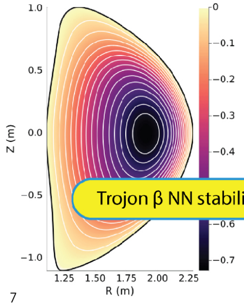
FUSE models span from the plasma core to the site boundary

FUSE models span from the plasma core to the site boundary

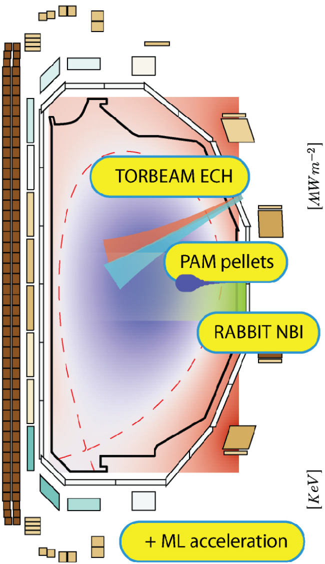
Current evolution w/ sawteeth



Fixed boundary equilibrium



Trojon β NN stability



TORBEAM ECH

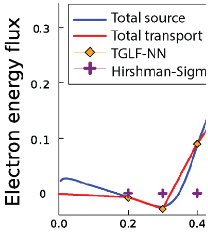
PAM pellets

RABBIT NBI

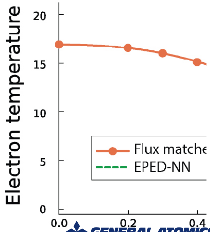
+ ML acceleration

Free boundary equilibrium w/ coil + passive couplings

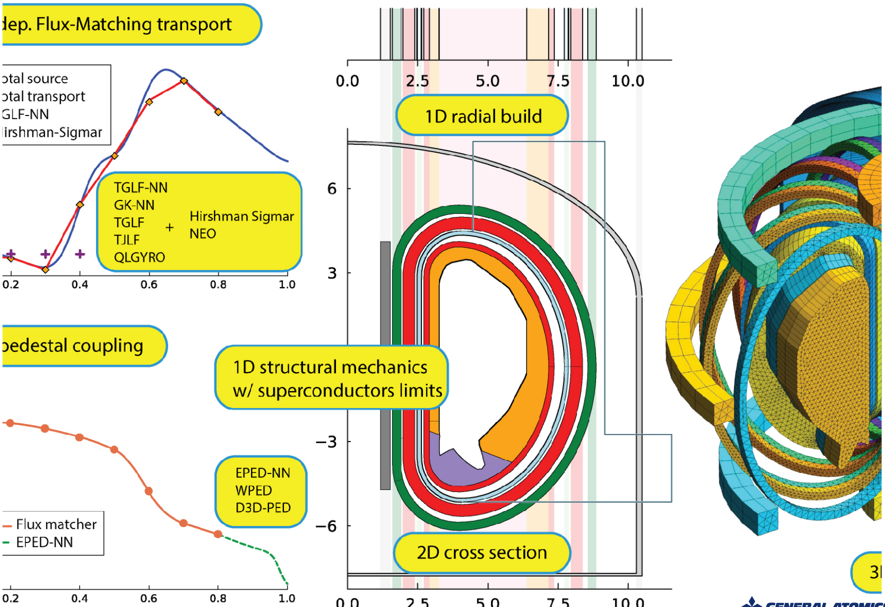
Time dep. Flux-M



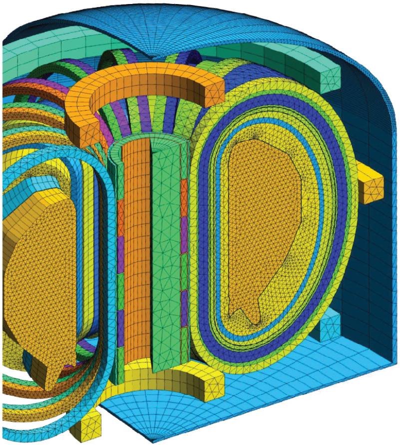
Core-pedestal co



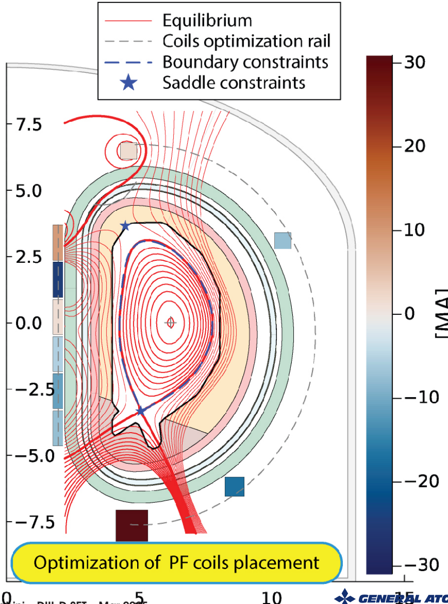
FUSE models span from the plasma core to the site boundary



FUSE models span from the plasma core to the site boundary

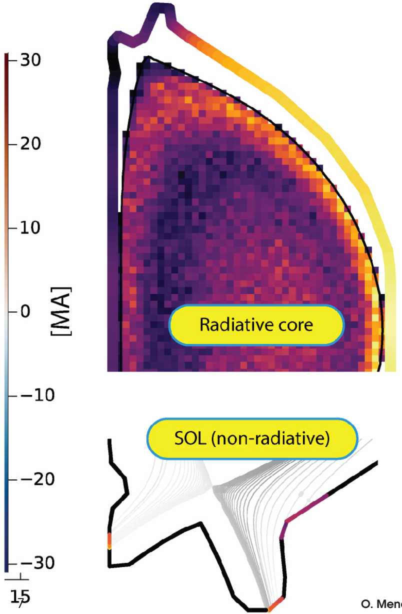


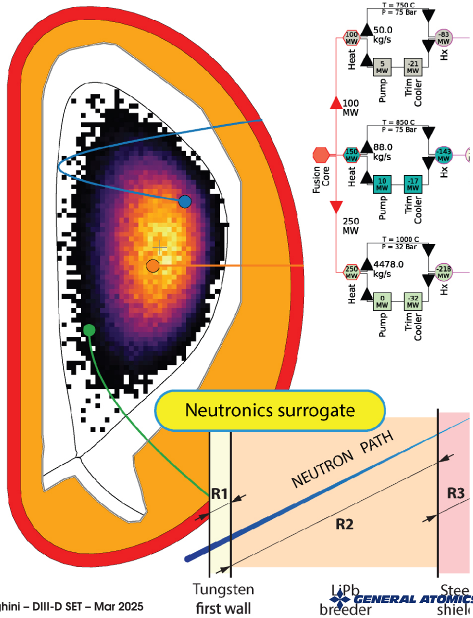
3D CAD and mesh



Optimization of PF coils placement

FUSE models span from the plasma core to the site boundary

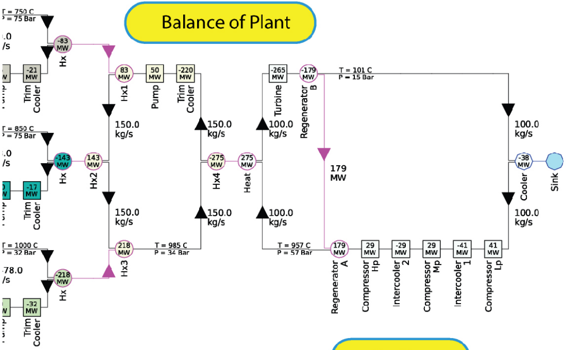


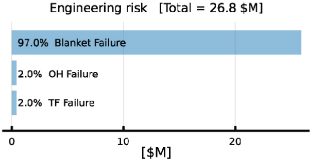


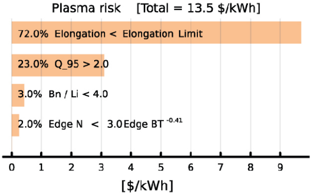
GENERAL ATOMICS

FUSE models span from the plasma core to the site boundary

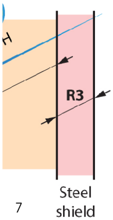
Balance of Plant

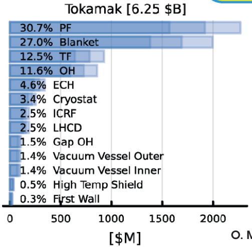


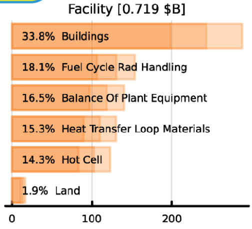




Costing





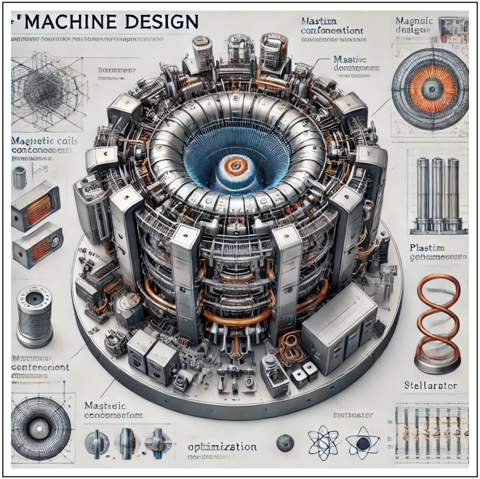


Risk

From idea to pre-conceptual designs in ~~months~~ minutes and evaluate wildly different concepts on same footing

1) MACHINE DESIGN

Well developed



FUSE uses a multi-objective constrained optimization workflow to enable design explorations and trade studies

OBJECTIVES

1. min capital cost
2. max q_{95}

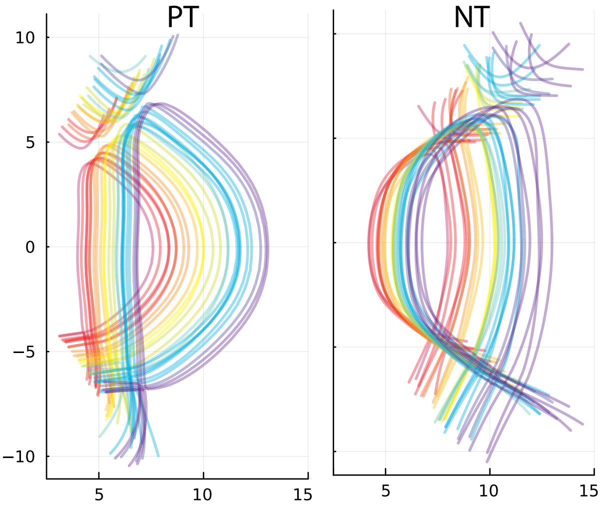
CONSTRAINTS

- $P_{\text{electric}} = 250 \pm 50$ MW
- flattop $= 1.0 \pm 0.1$ (h)
- TBR $= 1.1 \pm 0.1$
- $P_{\text{sol}}/P_{\text{LH}} > 1.1$ (for $+\delta$)
- $P_{\text{sol}}/R < 15$ (MW/m)

ACTUATORS

- $5.0 < R_0 < 10.0$ (m)
- $3.0 < B_0 < 15.0$ (T)
- $4.0 < I_p < 22$ (MA)
- $1.5 < \kappa < 2.2$
- $|\delta| < 0.7$
- $1.1 < z_{\text{eff,ped}} < 3.5$
- $0.4 < f_{\text{GW,ped}} < 0.85$
- Impurity: Ne, Ar, Kr
- $0 < P_{\text{EC}} < 100$ (MW)
- $0 < \rho_{\text{EC}} < 0.9$
- $0 < P_{\text{NB}} < 50$ (MW)

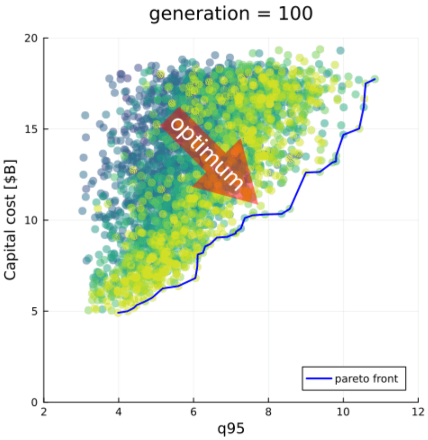
eg. Trade study for positive-δ VS negative-δ FPP



FUSE multi-objective constrained optimization workflow enables designs exploration and trade studies

A **Genetic algorithm** steers solution towards the **Pareto front**

- **Each point is a full machine design** that takes ~ 1 min to run
- Highlights **complex system dynamics** and exposes **objectives trade-offs**
- Helps different stakeholders **identify a target design**
  (scientists, investors, policymakers,...)
- **Scalable parallel execution** runs 10k+ cases in few hours on small cluster

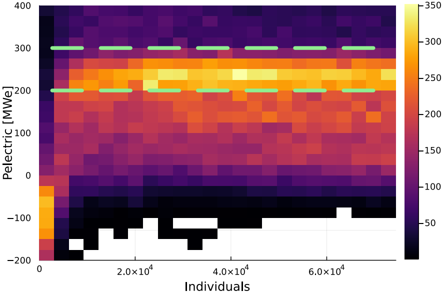


FUSE multi-objective constrained optimization workflow enables designs exploration and trade studies

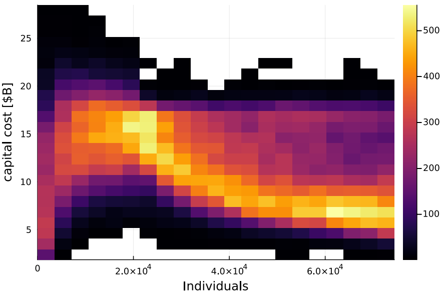
A **Genetic algorithm** steers solution towards the **Pareto front**

It takes 10's of thousands of full designs to find optimal solutions that satisfies the constraints. Eg:

Power generation constraint



Minimum cost objective

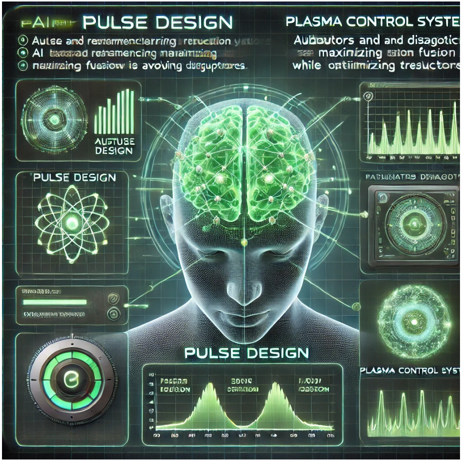


Accuracy, speed, scalability, and robustness are all key

Time-dependent capabilities that enable fast, high-fidelity, and machine-agnostic pulse design with PCS integration

2) PULSE DESIGN

Under development



<u>STAGE 2A</u>

Feed-forward simulation

For scientists

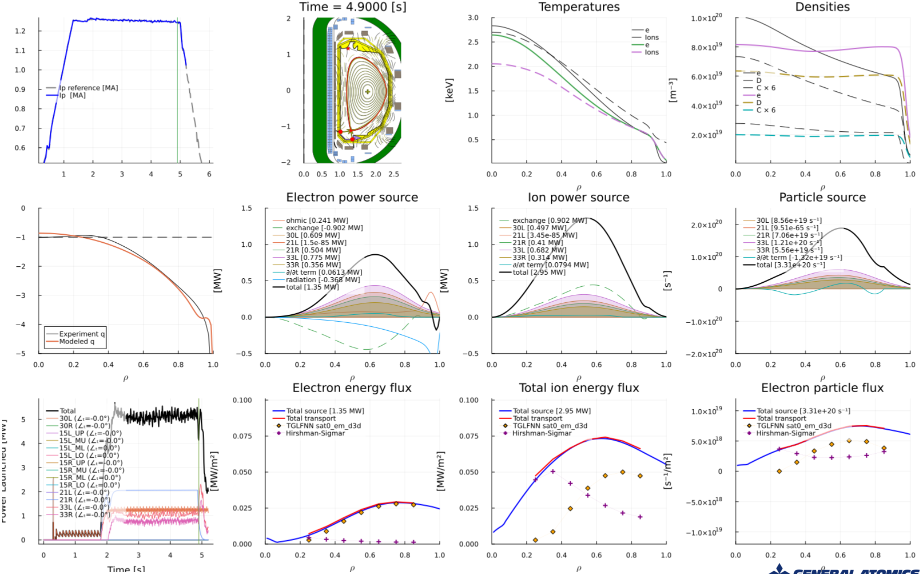
<u>STAGE 2B</u>

Feed-back simulation

For physics-operators

STAGE 2A: Feed-forward modeling of DIII-D pulses (Nearly ready for public use)

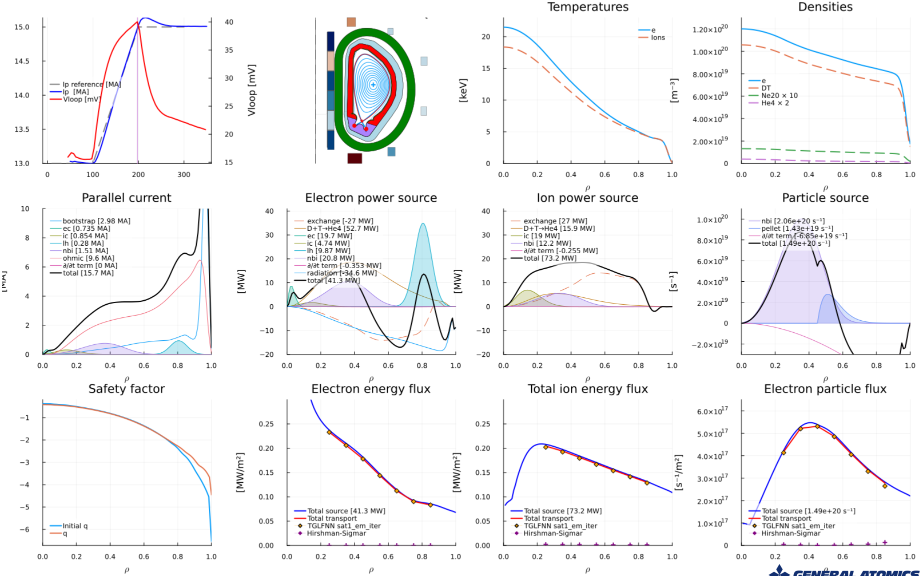
5 sec of DIII-D modeled from first principles in ~ 50 sec on a laptop



GENERAL ATOMICS

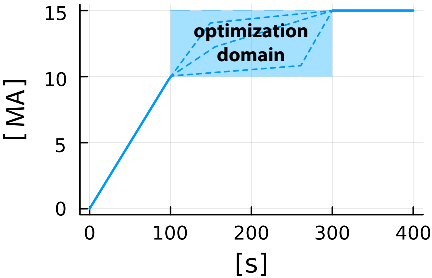
Like everything else in FUSE, this is also machine agnostic

300 sec of ITER modeled from first principles in ~ 180 sec on a laptop (faster than realtime!)

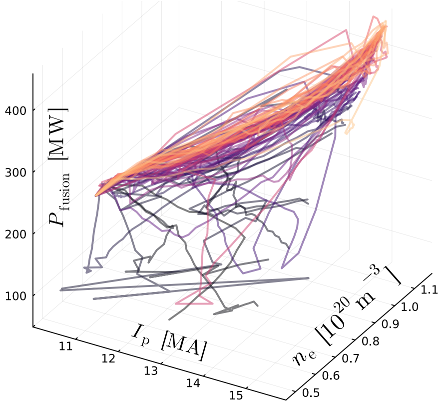


Time dependence + Optimization = Trajectory optimization

Leverage **same optimization infrastructure used for machine design**



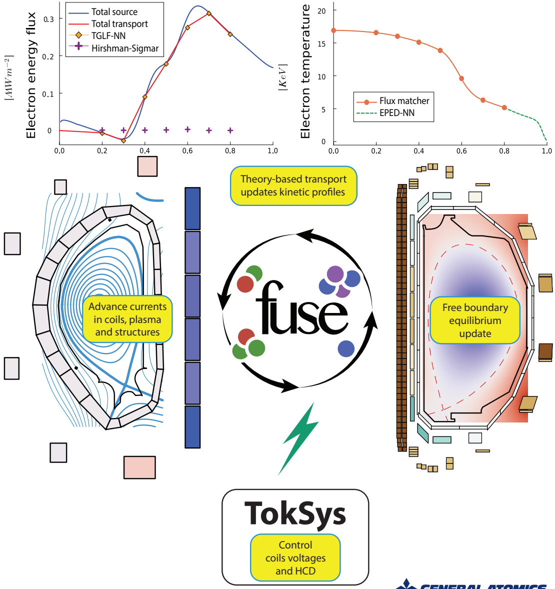
eg. Find optimal Ip and n_e ramp rates to max ITER fusion energy



- Define optimization domains for actuators time traces
- Define time-depedent objectives and constraints
- Take full advantage of HPC

STAGE 2B: Grad Hogan solver under development to model plasma dynamics in combination with control system

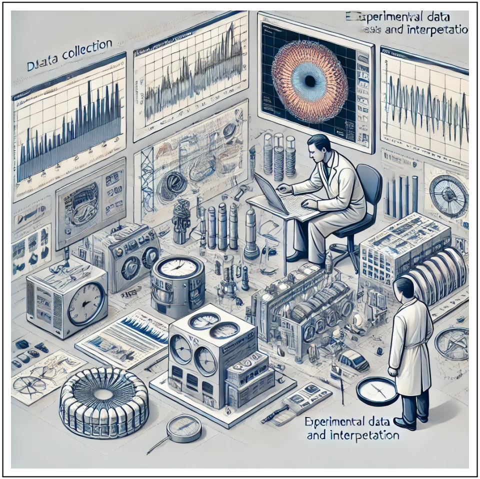
- Integration of:
  1. Free-boundary solver
  2. Theory-based transport
  3. Inductive coupling of plasma, PF coils, and conducting structures
- Co-simulation with control system
  - Demonstrated coupling with **TokSys**
  - Investigating coupling with **DIII-D PCS**



Analyses fast enough to inform operators between-shots and enable Bayesian integrated data analysis

3) DATA ANALYSIS

To be developed



FUSE has the potential to address key bottlenecks and limitations of today's data analyses techniques

Long experience automating data analyses on DIII-D and other tokamaks

eg. Kinetic equilibrium reconstructions on DIII-D:

	2013	2017	2019	2023
Manual	**KineticEFIT**	**KineticEFITtime**	**CAKE**	**CAKE @ NERSC**
Days	Hour	Hours	Hour	20 minutes
1 timeslice	1 timeslice	10 timeslices	50 timeslices	50 timeslices

Three main limitations:

1. Reaching speed limit with legacy serial codes and loose couplings
2. Machine specific workflows
3. Mostly classical data analysis

FUSE can address all of these:

1. Faster, with tight coupling and higher degree of parallelism
   For context, FUSE predictive model of 200 DIII-D time-slices in 50 sec on a laptop
2. IMAS-based and machine-agnostic
3. Constrain data with models (enable Bayesian data analysis)

Wrap-up

FUSE is making steady progress towards supporting all three major integrated modeling applications

1. **MACHINE DESIGN** - well developed
   - Rapidly go from FPP concept to pre-conceptual design
   - Trade studies with multi-objective constrained optimization

2. **PULSE DESIGN** - in development
   - Feed-forward sims. for trajectory opt. and FPP scenario access
   - Feed-back sims. for ops. and develop controls in data-poor FPPs

3. **DATA ANALYSIS** - to be developed
   - Faster data analyses for more/richer between-shot feedback
   - Bayesian data analyses for higher quality inference with UQ

Thank you to all the contributors!

O. Meneghini – *Project lead*
T. Slendebroek – *Framework, integration, MOOPT*
B. C. Lyons – *Everything MHD, performance*
T. F. Neiser – *TGLF-NN, GK-NN, TJLF*
J. T. McClenaghan – *EGGO, NBI, QLGYRO*
A. Ghiozzi – *NEO, costing, risk, materials*
N. Shi – *Impurity transport, self-consistent plasma*
S. Denk – *TORBEAM, DIII-D data*
M. Yoo – *MHD stability, performance*
G. Avdeeva – *PAM, TJLF*
G. Dose – *SOL, divertors, wall*
L. Stagner – *Control, MDS+*
J. Harvey – *Balance of plant*
J. Guterl – *SOL model, ADAS*
D. B. Weisberg – *Engineering, GASC port*
T. B. Cote – *Operational limits*
M. Clark – *PCS interface, OMEGA install*
A. Zalzali – *GATM interface*
D. Eldon – *SOLPS interface*
A. Gupta – *GGD, synthetic diagnostics*
K. Mclaughlink – *Neutronics, Blanket*

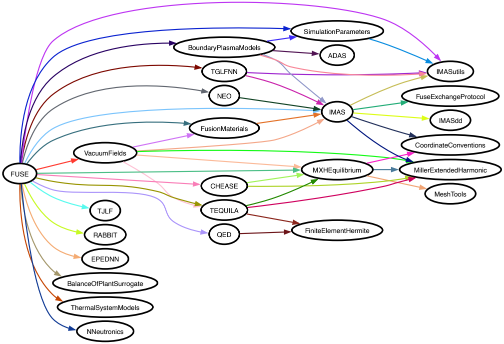
We are a growing community

- GA, UKAEA, TE, KFE, KF, ...
- Weekly hackathons at GA
- Online, on Discord
- Code-camp Dec. 2024

FUSE is completely open-source with Apache 2.0 license free for anyone to use including commercial applications

We have released the entire FUSE ecosystem on GitHub

- 25+ packages
- 200K+ lines of Julia
- Documented
- Regression tested
- Preprint on Arxiv
- `http://fuse.help`



Consider Julia for your next software project!

- High-level, fast, auto-diff
- Enthusiastic GA community
- Most Julia devs were former Python devs

Get involved with FUSE!

- **As user:**
  - Online tutorials
  - Beta testing new features
  - Tutorial days (coming soon)

- **As developer:**
  - Develop and integrate **new physics models** and **synthetic diagnostics**
  - Add **DIII-D mappings to IMAS**
  - Join code-camps on Wednesdays 13.301

- **Let's collaborate** on upcoming funding opportunities!

meneghini@fusion.gat.com

Join us to accelerate the future of fusion!

http://fuse.help