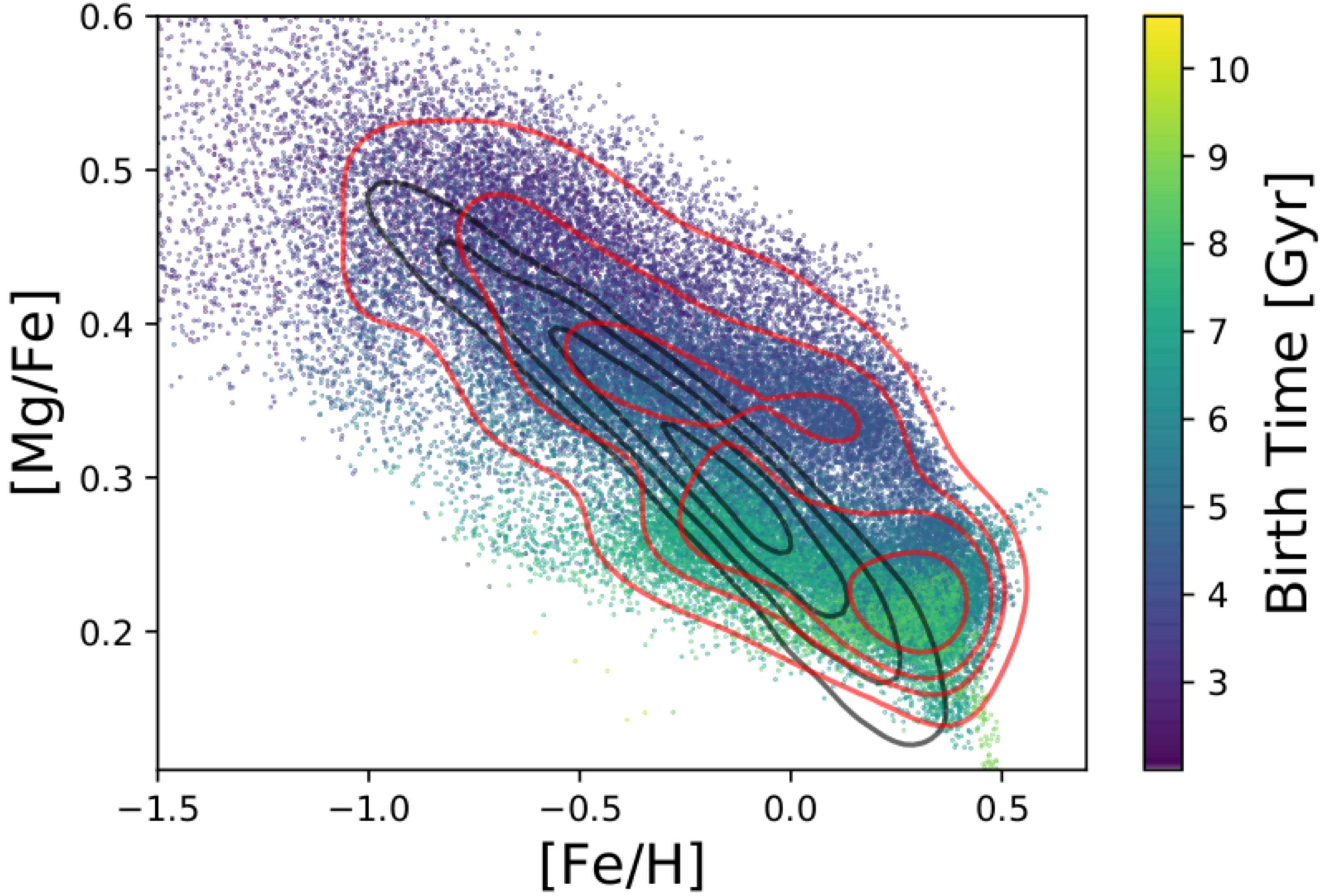


Inferring Galactic Parameters from Chemical Abundances:

A Multi-Star Approach

JAN RYBIZKI (MPIA)

OLIVER PHILCOX (PRINCETON)

JINA-CEE ONLINE SEMINAR

Nov 22, 2019

Outline

1. *Chempy*: A Fast and Flexible GCE Model

Outline

1. *Chempy*: A Fast and Flexible GCE Model

2. Choosing Yield Tables for Hydrodynamical Simulations

Outline

1. *Chempy*: A Fast and Flexible GCE Model

2. Choosing Yield Tables for Hydrodynamical Simulations

3. Scoring Nucleosynthetic Yield Tables

Outline

1. *Chempy*: A Fast and Flexible GCE Model

2. Choosing Yield Tables for Hydrodynamical Simulations

3. Scoring Nucleosynthetic Yield Tables

4. Inferring Galactic Parameters with Multi-Star Inference

Outline

1. *Chempy*: A Fast and Flexible GCE Model
2. Choosing Yield Tables for Hydrodynamical Simulations
3. Scoring Nucleosynthetic Yield Tables
4. Inferring Galactic Parameters with Multi-Star Inference
5. Further Extensions

Building a Fast and Flexible Galactic Chemical Evolution Model with *Chempy*

github.com/jan-rybizki/Chempy

Building a Fast and Flexible GCE: *Chempy*

- *Chempy* (including **chemical yields**, and **SSP** parameters):

SSP = Simple Stellar Population (A group of stars born at the same time in the same environment)

github.com/jan-rybizki/Chempy

Building a Fast and Flexible GCE: *Chempy*

- *Chempy* (including **chemical yields**, and **SSP** parameters):
  - **IMF** integrated, metallicity-dependent yield over time

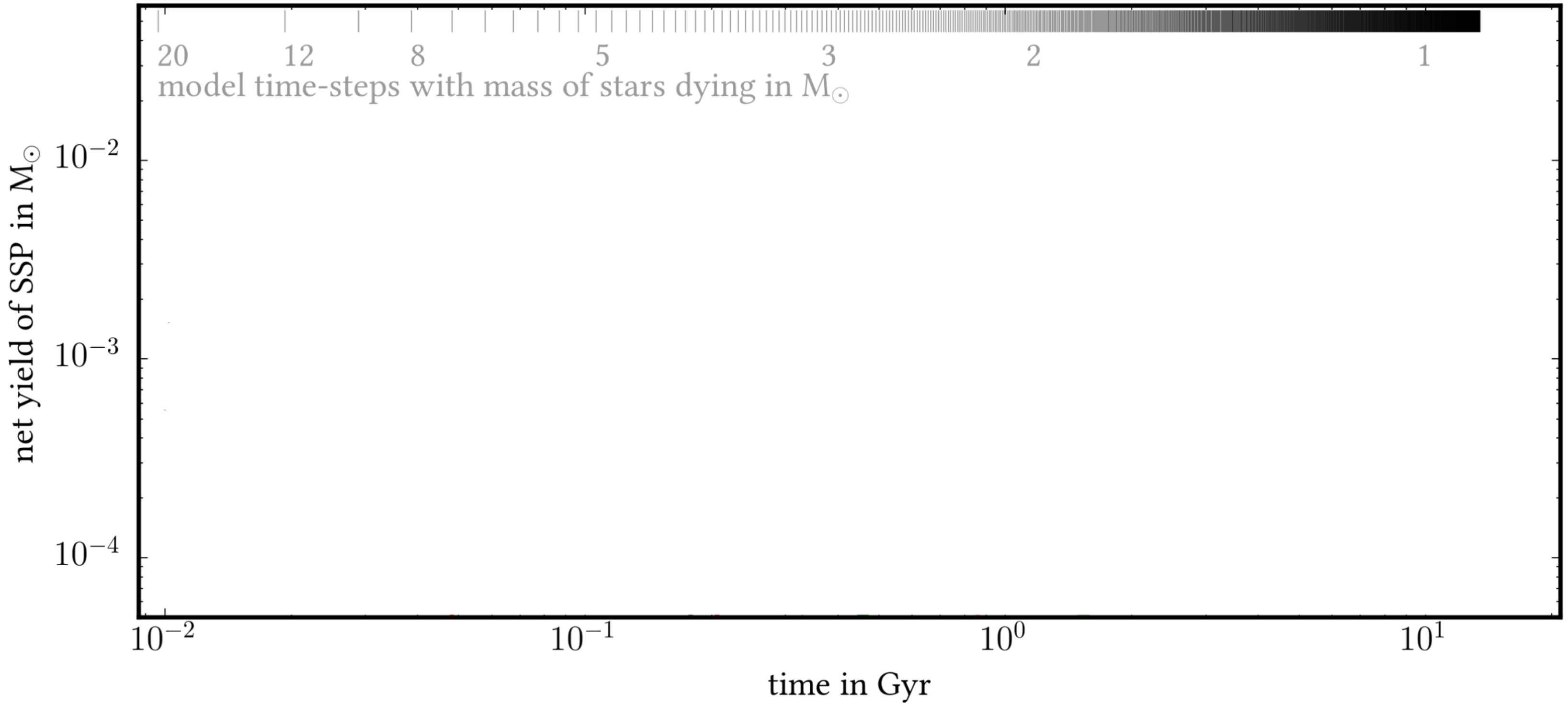
SSP = Simple Stellar Population
(A group of stars born at the same time in the same environment)

IMF = **Initial Mass Function**
(Contribution of stars born in a certain mass range to total mass)

github.com/jan-rybizki/Chempy

Building a Fast and Flexible GCE: *Chempy*

- *Chempy* (including **chemical yields**, and **SSP** parameters):
  - **IMF** integrated, metallicity-dependent yield over time



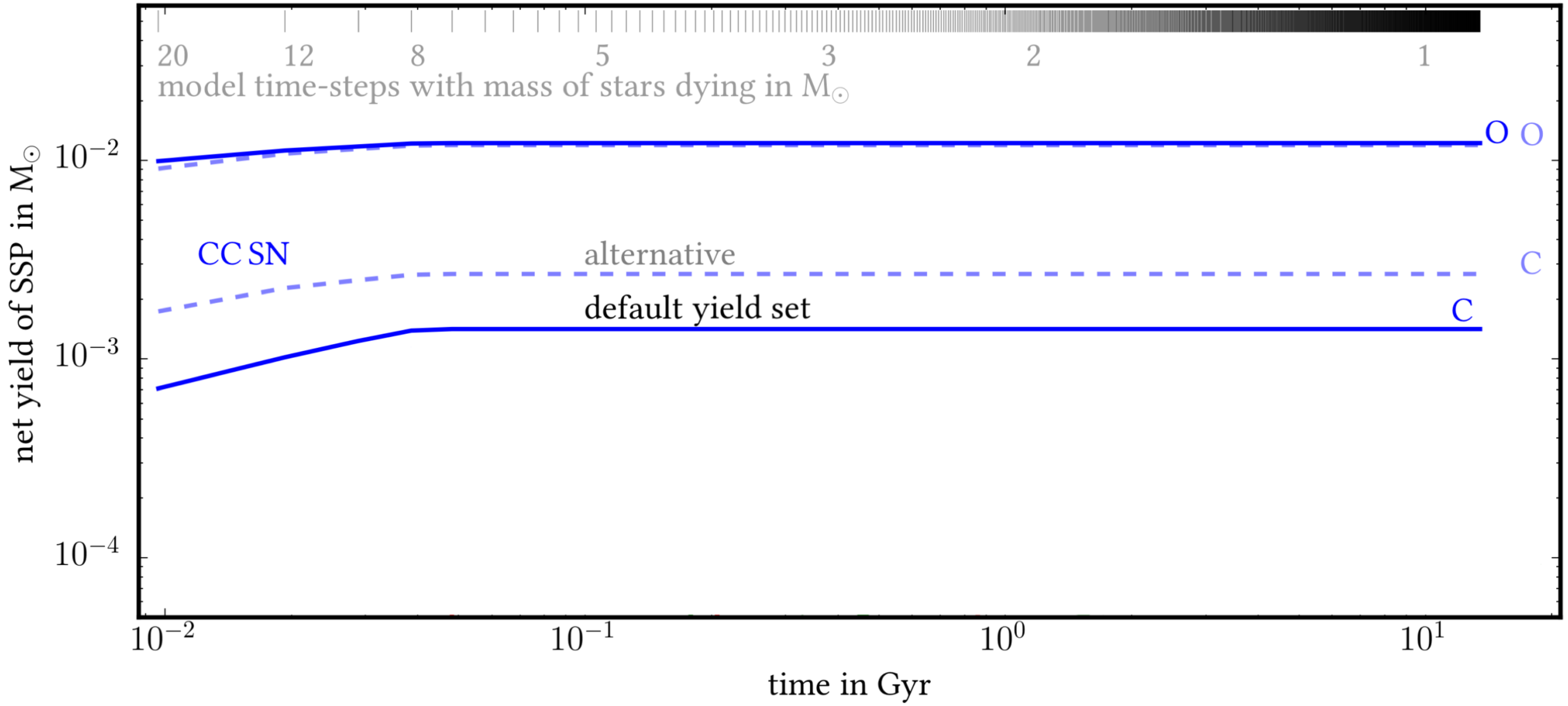
SSP = Simple Stellar Population (A group of stars born at the same time in the same environment)

IMF = **Initial Mass Function** (Contribution of stars born in a certain mass range to total mass)

github.com/jan-rybizki/Chempy

Building a Fast and Flexible GCE: *Chempy*

- ❑ *Chempy* (including **chemical yields**, and **SSP** parameters):
  - ❑ **IMF** integrated, metallicity-dependent yield over time



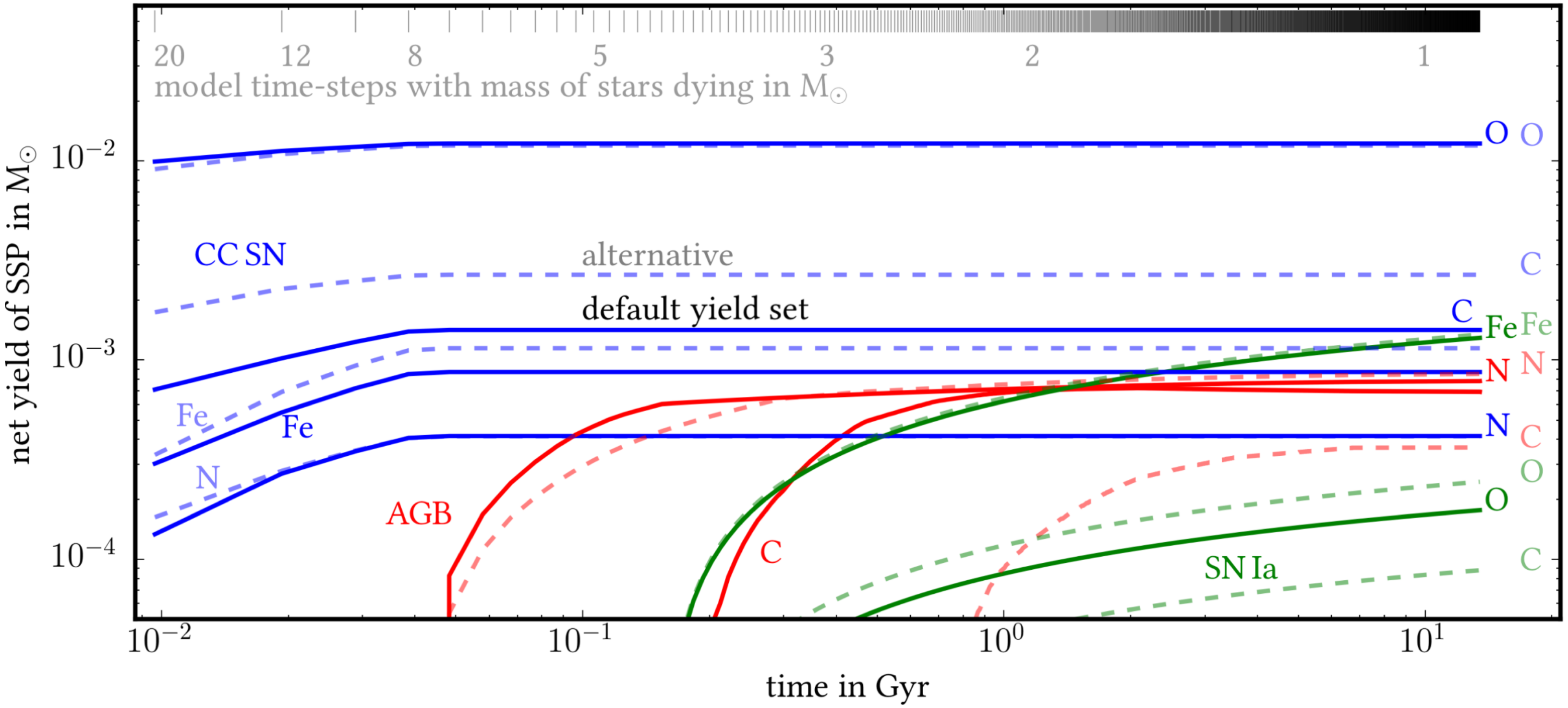
SSP = Simple Stellar Population (A group of stars born at the same time in the same environment)

IMF = Initial Mass Function (Contribution of stars born in a certain mass range to total mass)

github.com/jan-rybizki/Chempy

Building a Fast and Flexible GCE: *Chempy*

- *Chempy* (including **chemical yields**, and **SSP** parameters):
  - **IMF** integrated, metallicity-dependent yield over time



SSP = Simple Stellar Population (A group of stars born at the same time in the same environment)

IMF = Initial Mass Function (Contribution of stars born in a certain mass range to total mass)

github.com/jan-rybizki/Chempy

Building a Fast and Flexible GCE: *Chempy*

- *Chempy* (including **chemical yields**, and **SSP** parameters):
  - **IMF** integrated, metallicity-dependent yield over time

- *Chempy* (**yields**, **SSP** + Interstellar Medium **(ISM)** parameters):
  - Outputs **ISM abundances** over time

github.com/jan-rybizki/Chempy

Building a Fast and Flexible GCE: *Chempy*

- *Chempy* (including **chemical yields**, and **SSP** parameters):
  - **IMF** integrated, metallicity-dependent yield over time

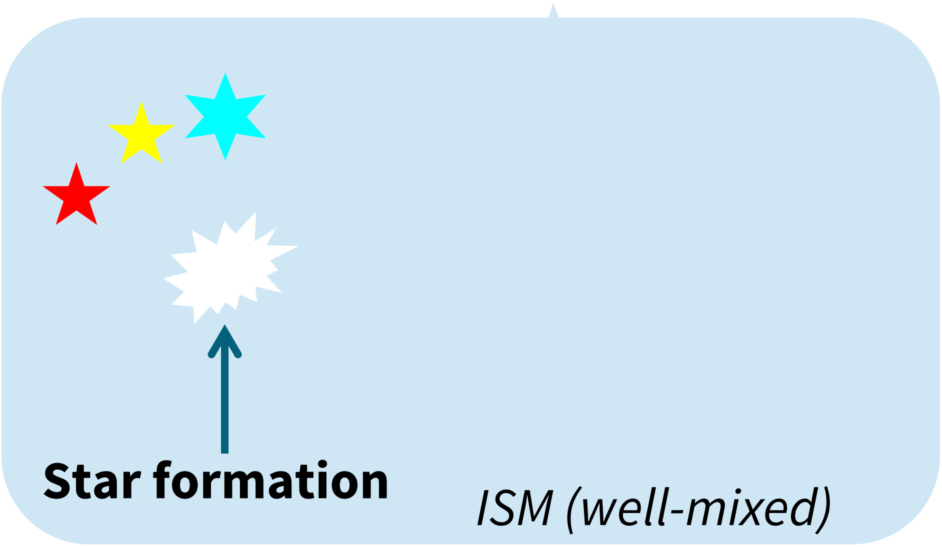
- *Chempy* (**yields**, **SSP** + Interstellar Medium **(ISM)** parameters):
  - Outputs **ISM abundances** over time

github.com/jan-rybizki/Chempy

Building a Fast and Flexible GCE: *Chempy*

- ❑ *Chempy* (including **chemical yields**, and **SSP** parameters):
  - ❑ **IMF** integrated, metallicity-dependent yield over time

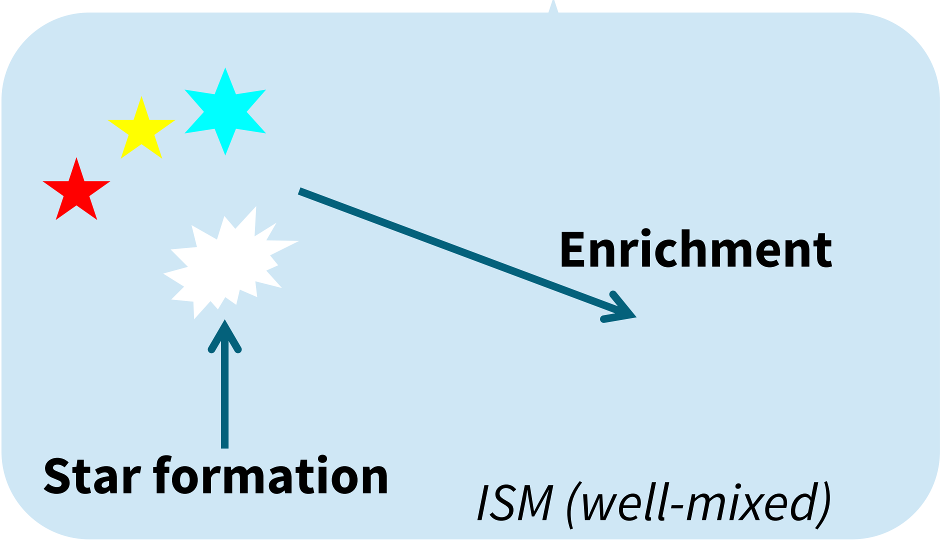
- ❑ *Chempy* (**yields**, **SSP** + Interstellar Medium **(ISM)** parameters):
  - ❑ Outputs **ISM abundances** over time



github.com/jan-rybizki/Chempy

Building a Fast and Flexible GCE: *Chempy*

- ❑ *Chempy* (including **chemical yields**, and **SSP** parameters):
  - ❑ **IMF** integrated, metallicity-dependent yield over time
- ❑ *Chempy* (**yields**, **SSP** + Interstellar Medium **(ISM)** parameters):
  - ❑ Outputs **ISM abundances** over time

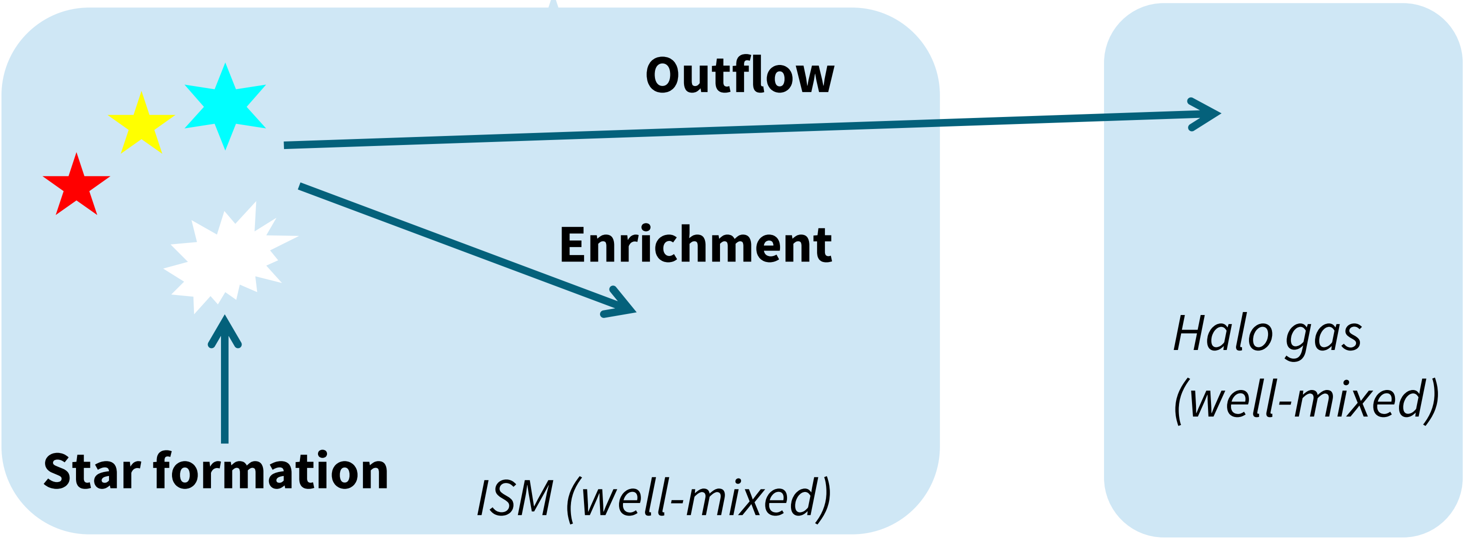


github.com/jan-rybizki/Chempy

Building a Fast and Flexible GCE: *Chempy*

- *Chempy* (including **chemical yields**, and **SSP** parameters):
  - **IMF** integrated, metallicity-dependent yield over time

- *Chempy* (**yields**, **SSP** + Interstellar Medium **(ISM)** parameters):
  - Outputs **ISM abundances** over time

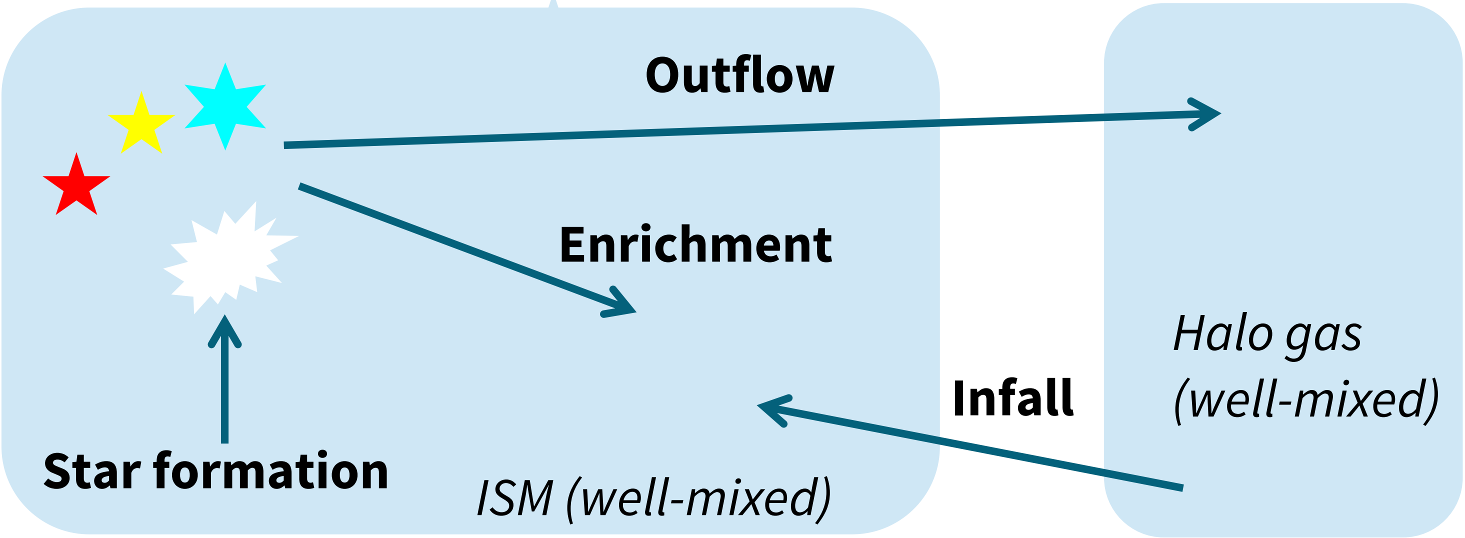


github.com/jan-rybizki/Chempy

Building a Fast and Flexible GCE: *Chempy*

- ❑ *Chempy* (including **chemical yields**, and **SSP** parameters):
  - ❑ **IMF** integrated, metallicity-dependent yield over time

- ❑ *Chempy* (**yields**, **SSP** + Interstellar Medium **(ISM)** parameters):
  - ❑ Outputs **ISM abundances** over time

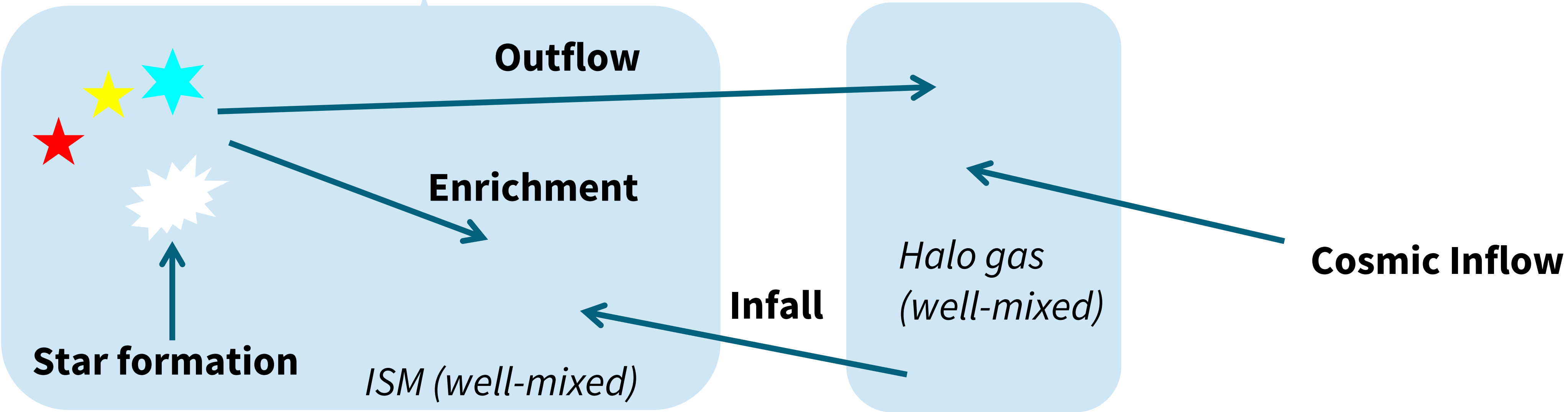


github.com/jan-rybizki/Chempy

Building a Fast and Flexible GCE: *Chempy*

- *Chempy* (including **chemical yields**, and **SSP** parameters):
  - **IMF** integrated, metallicity-dependent yield over time

- *Chempy* (**yields**, **SSP** + Interstellar Medium **(ISM)** parameters):
  - Outputs **ISM abundances** over time



github.com/jan-rybizki/Chempy

Building a Fast and Flexible GCE: *Chempy*

- *Chempy* (including **chemical yields**, and **SSP** parameters):
    - **IMF** integrated, metallicity-dependent yield over time

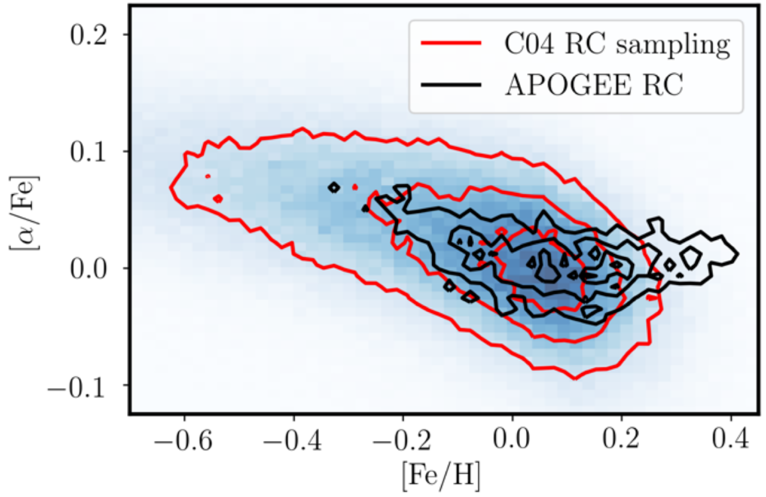
- *Chempy* (**yields**, **SSP** + Interstellar Medium **(ISM)** parameters):
    - Outputs **ISM abundances** over time – compare to **observational data**

github.com/jan-rybizki/Chempy

Building a Fast and Flexible GCE: *Chempy*

- *Chempy* (including **chemical yields**, and **SSP** parameters):
  - **IMF** integrated, metallicity-dependent yield over time

- *Chempy* (**yields**, **SSP** + Interstellar Medium **(ISM)** parameters):
  - Outputs **ISM abundances** over time – compare to **observational data**

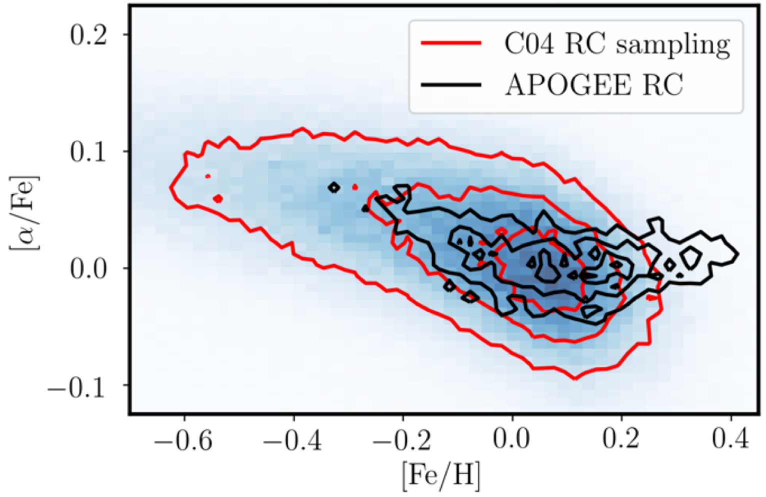


github.com/jan-rybizki/Chempy

Building a Fast and Flexible GCE: *Chempy*

- *Chempy* (including **chemical yields**, and **SSP** parameters):
  - **IMF** integrated, metallicity-dependent yield over time

- *Chempy* (**yields**, **SSP** + Interstellar Medium **(ISM)** parameters):
  - Outputs **ISM abundances** over time – compare to **observational data**



*This needs **no fiducial model**,*
*Input your data → Obtain **GCE parameters***

Rybizki, Just & Rix (2017)

github.com/jan-rybizki/Chempy

Parameter Inference with *Chempy*

- *Chempy* can be inserted in a **Bayesian** framework to **infer** Galactic parameters.

Rybizki, Just & Rix (2017)

github.com/jan-rybizki/Chempy

Parameter Inference with *Chempy*

- *Chempy* can be inserted in a **Bayesian** framework to **infer** Galactic parameters.

SSP Parameters

- Initial Mass Function Slope
- Number of SN Ia

Rybizki, Just & Rix (2017)

github.com/jan-rybizki/Chempy

Parameter Inference with *Chempy*

- ❑ *Chempy* can be inserted in a **Bayesian** framework to **infer** Galactic parameters.

SSP Parameters
- Initial Mass Function Slope
- Number of SN Ia

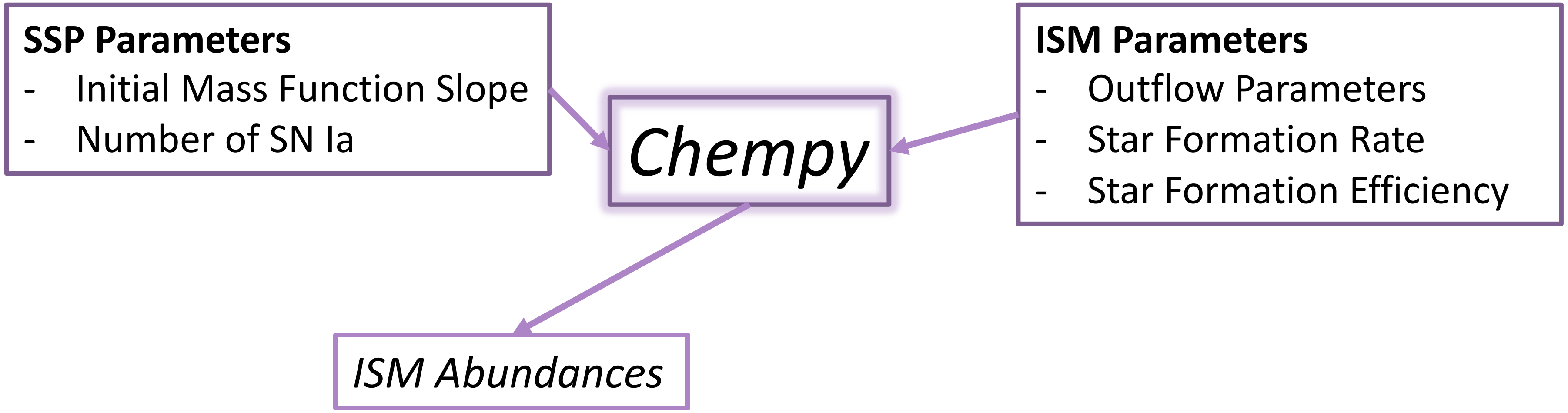
ISM Parameters
- Outflow Parameters
- Star Formation Rate
- Star Formation Efficiency

Rybizki, Just & Rix (2017)

github.com/jan-rybizki/Chempy

Parameter Inference with *Chempy*

- ❑ *Chempy* can be inserted in a **Bayesian** framework to **infer** Galactic parameters.

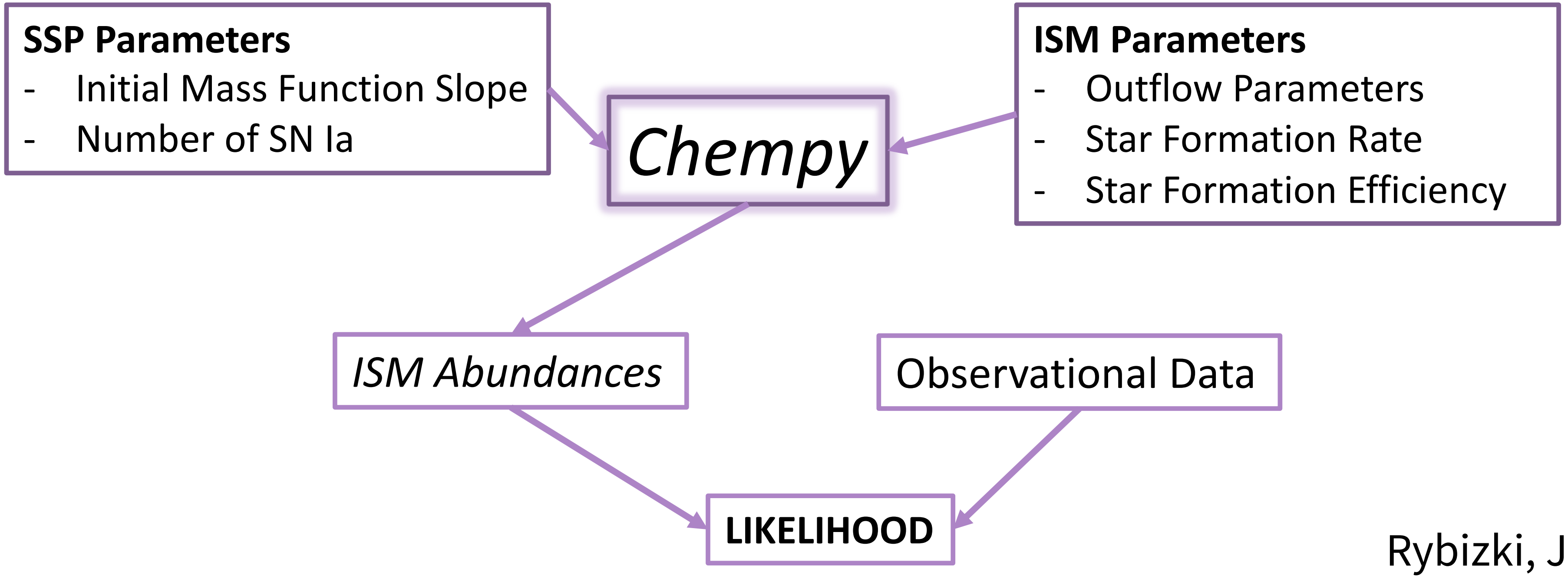


Rybizki, Just & Rix (2017)

github.com/jan-rybizki/Chempy

Parameter Inference with *Chempy*

- *Chempy* can be inserted in a **Bayesian** framework to **infer** Galactic parameters.

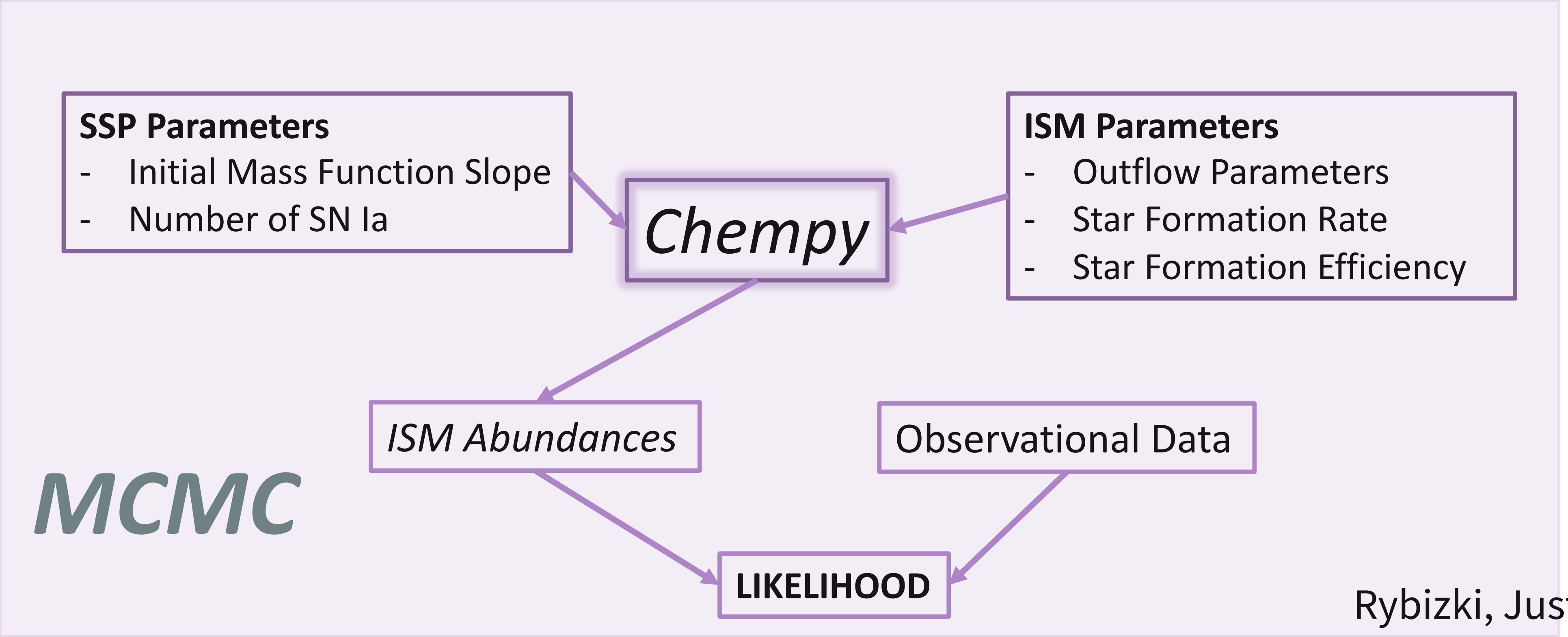


Rybizki, Just & Rix (2017)

github.com/jan-rybizki/Chempy

Parameter Inference with *Chempy*

- ❑ *Chempy* can be inserted in a **Bayesian** framework to **infer** Galactic parameters.



Rybizki, Just & Rix (2017)

Can *Chempy* be used to determine GCE parameters for hydrodynamical simulations?

github.com/oliverphilcox/ChempyMulti

Predicting SSP Parameters for Hydro Sims

- **Goal**: Predict the best SSP parameters for hydrodynamical simulations to make sure they are consistent with observations

Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Predicting SSP Parameters for Hydro Sims

- **Goal**: Predict the best SSP parameters for hydrodynamical simulations to make sure they are consistent with observations

- **Method**:
  - Determine optimal SSP parameters for specific simulation using *Chempy*
    - Optimize using the **proto-Solar** abundances alone

Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Predicting SSP Parameters for Hydro Sims

- **Goal**: Predict the best SSP parameters for hydrodynamical simulations to make sure they are consistent with observations

- **Method**:
  - Determine optimal SSP parameters for specific simulation using *Chempy*
    - Optimize using the **proto-Solar** abundances alone

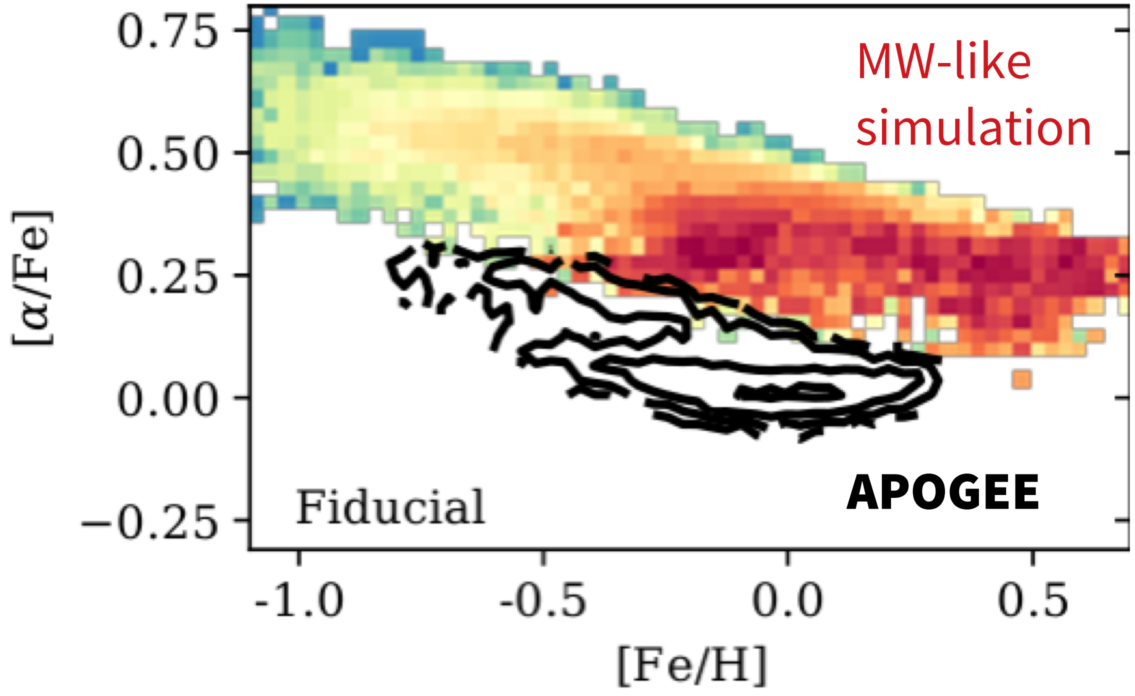
  - Run simulation with **optimized** parameters and compare results!

Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Predicting SSP Parameters for Hydro Sims

- **Goal**: Predict the best SSP parameters for hydrodynamical simulations to make sure they are consistent with observations

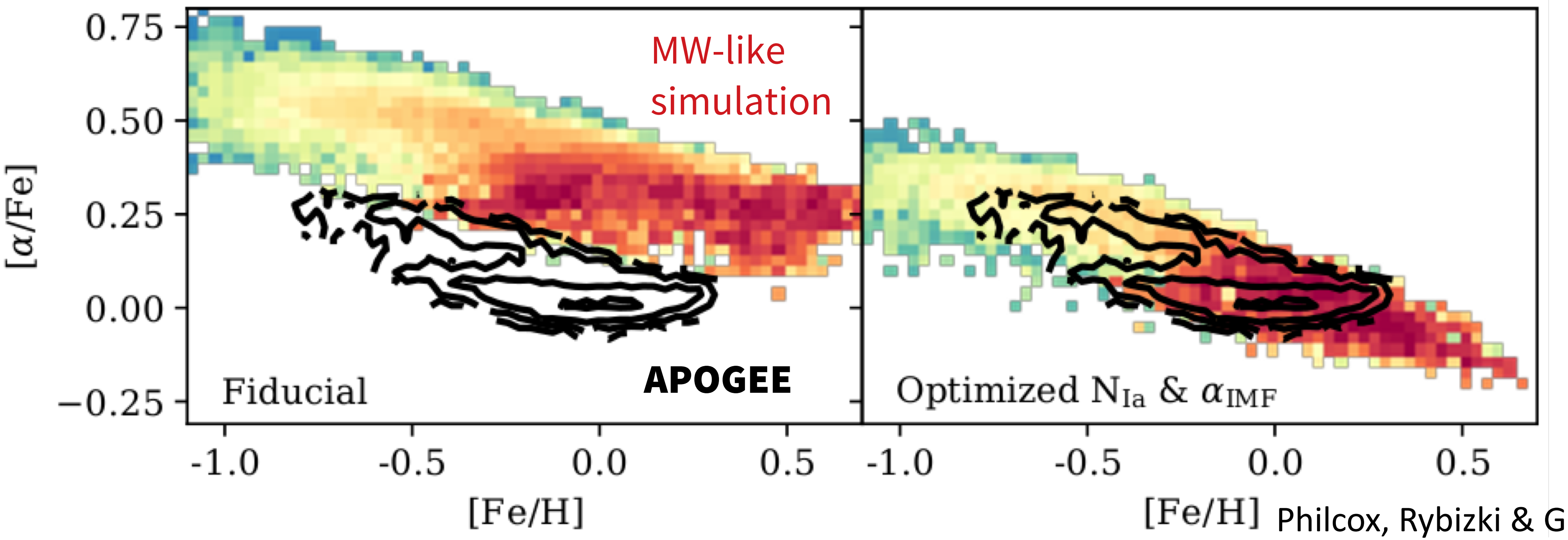


Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Predicting SSP Parameters for Hydro Sims

- ❑ **Goal**: Predict the best SSP parameters for hydrodynamical simulations to make sure they are consistent with observations

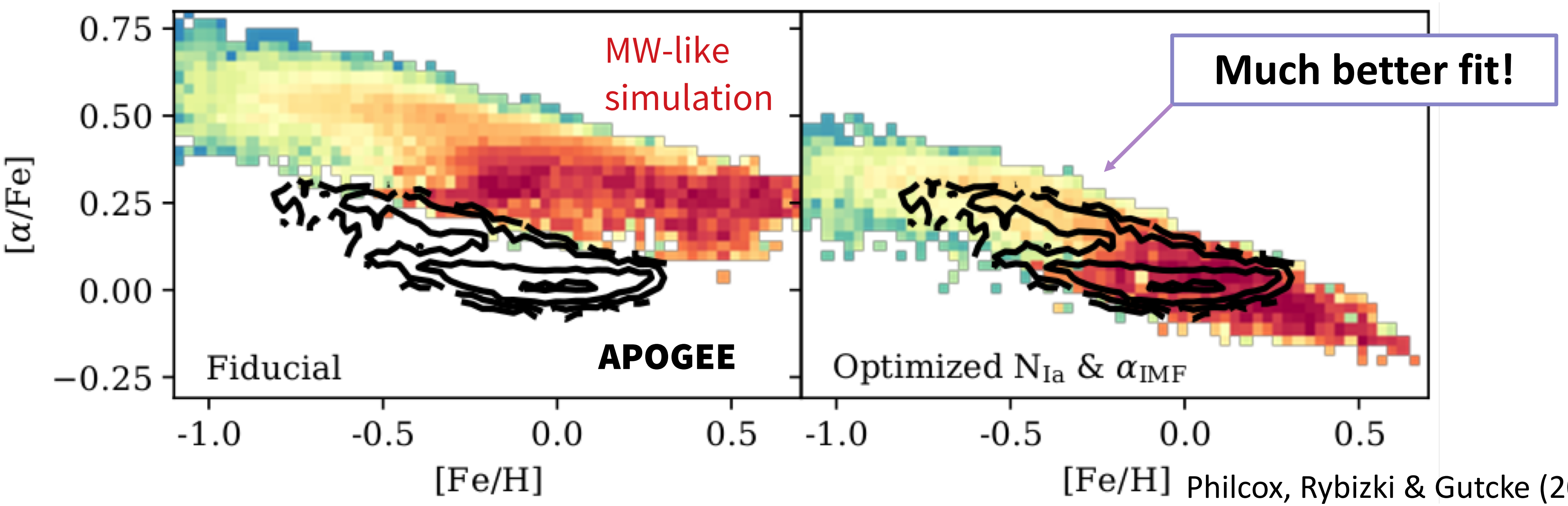


Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Predicting SSP Parameters for Hydro Sims

- **Goal**: Predict the best SSP parameters for hydrodynamical simulations to make sure they are consistent with observations

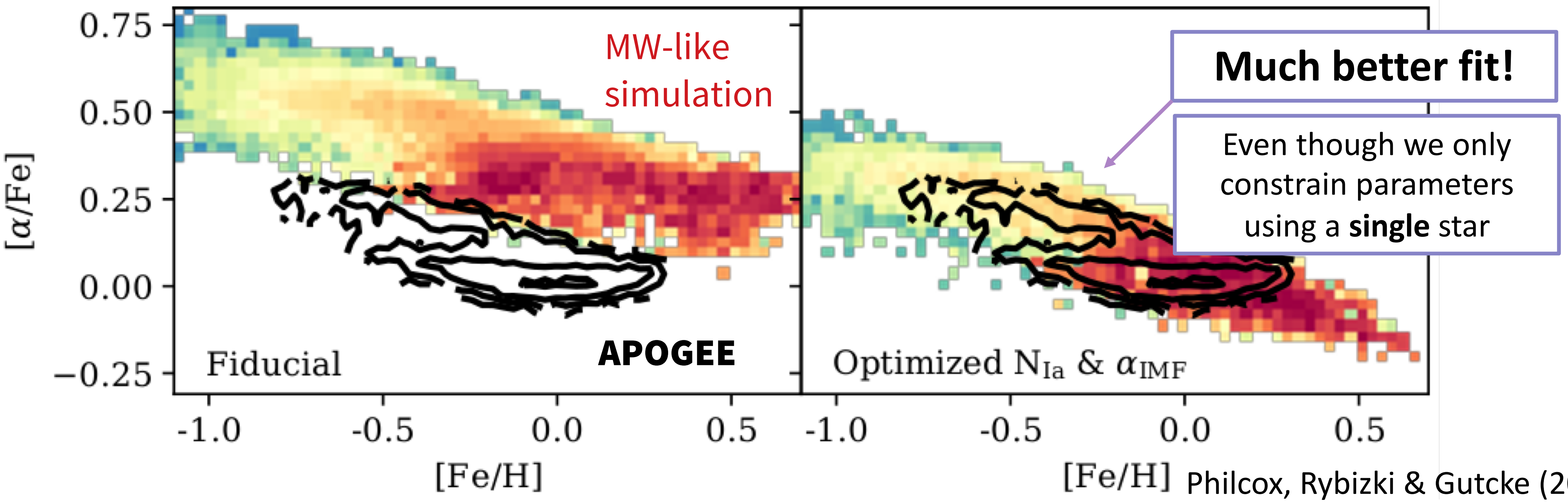


Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Predicting SSP Parameters for Hydro Sims

- ❑ **Goal**: Predict the best SSP parameters for hydrodynamical simulations to make sure they are consistent with observations



Philcox, Rybizki & Gutcke (2018)

How well do our Yield Tables match Observational Data?

github.com/oliverphilcox/ChempyMulti

Yield Table Scoring

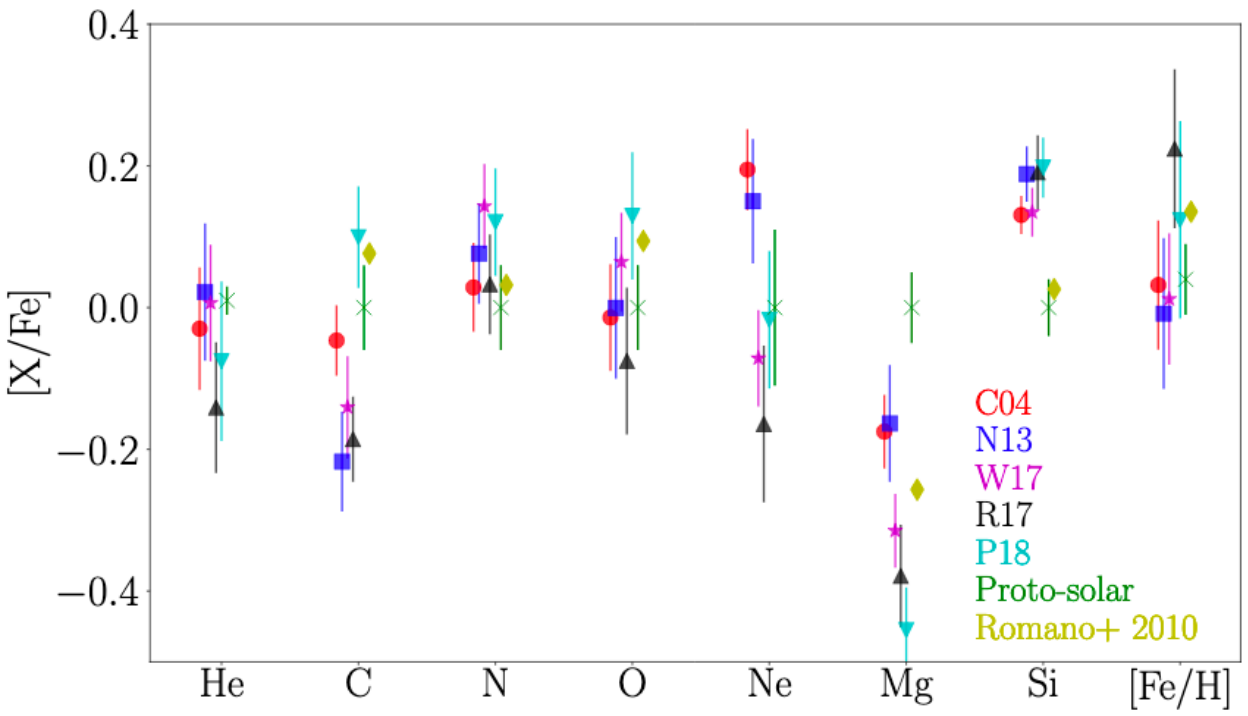
- Different Yield Tables predict **very different** proto-Solar abundances

Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Yield Table Scoring

- Different Yield Tables predict **very different** proto-Solar abundances

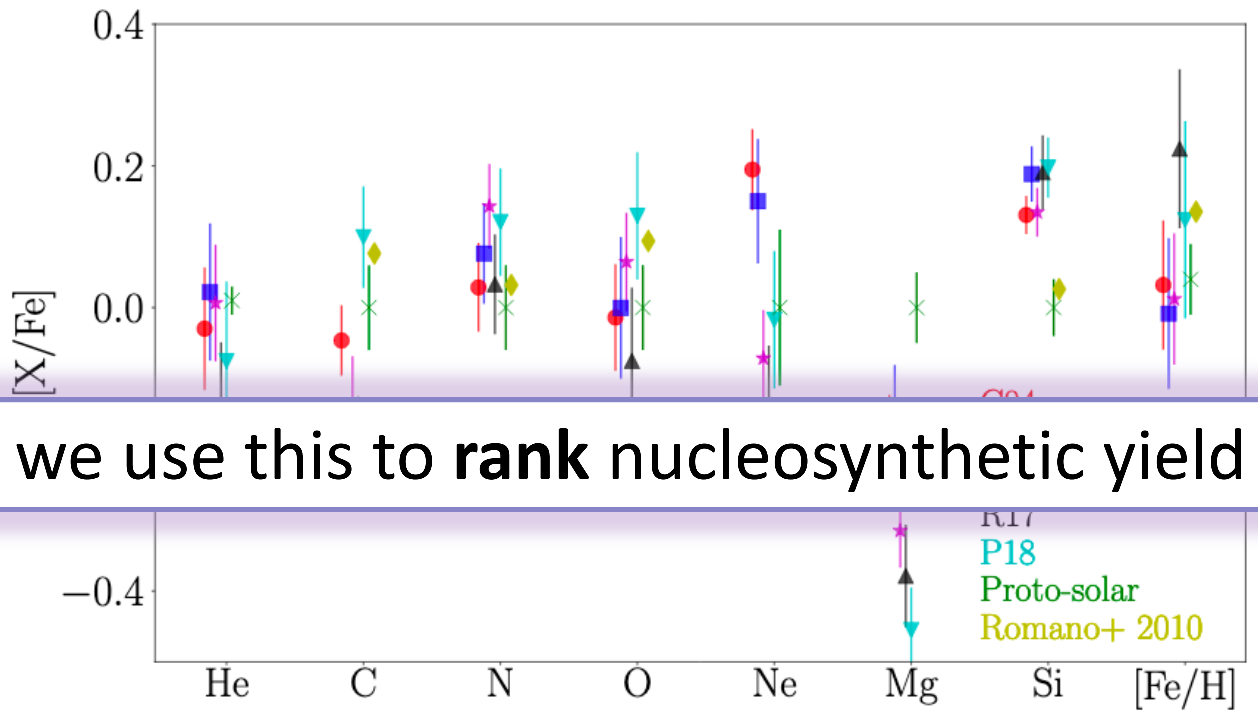


Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Yield Table Scoring

- Different Yield Tables predict **very different** proto-Solar abundances



Can we use this to **rank** nucleosynthetic yield tables?

Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Yield Table Scoring

- **Method:**
  - Compare *Chempy* observations to **proto-Solar** abundances

Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Yield Table Scoring

- **Method:**
  - Compare *Chempy* observations to **proto-Solar** abundances
  - **Marginalize** over SSP and ISM parameters

Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Yield Table Scoring

- **Method:**
  - Compare *Chempy* observations to **proto-Solar** abundances
  - **Marginalize** over SSP and ISM parameters
  - Include **model error** to account for modeling inaccuracies

Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Yield Table Scoring

- **Method:**
  - Compare *Chempy* observations to **proto-Solar** abundances
  - **Marginalize** over SSP and ISM parameters
  - Include **model error** to account for modeling inaccuracies

- **Model Comparison Statistics:**

Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Yield Table Scoring

- **Method:**
  - Compare *Chempy* observations to **proto-Solar** abundances
  - **Marginalize** over SSP and ISM parameters
  - Include **model error** to account for modeling inaccuracies

- **Model Comparison Statistics:**
  - Bayes Factor
    - $B \sim \int d\Theta \, \mathrm{Posterior}(\Theta \mid Data)$

Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Yield Table Scoring

- **Method:**
  - Compare *Chempy* observations to **proto-Solar** abundances
  - **Marginalize** over SSP and ISM parameters
  - Include **model error** to account for modeling inaccuracies

- **Model Comparison Statistics:**
  - Bayes Factor
    - $B \sim \int d\Theta \text{ Posterior}(\Theta \mid Data)$
  - Cross-Validation
    - Use $n-1$ elements to predict n-th element

Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Yield Table Scoring

Comparing Core-Collapse Supernovae Yield Tables using 28 Elements

Yield Set	Bayes Score
C04	−1.21
N13	−5.69
W17	−0.78
R17	−6.11
P18	0.86

Philcox, Rybizki & Gutcke (2018)

github.com/oliverphilcox/ChempyMulti

Yield Table Scoring

Comparing Core-Collapse Supernovae Yield Tables using 28 Elements

Yield Set	Bayes Score
C04	-1.21
N13	-5.69
W17	-0.78
R17	-6.11
P18	0.86

Nikos Prantzos' yields perform best here!

Philcox, Rybizki & Gutcke (2018)

Inferring SSP Parameters from Large Stellar Datasets with *Chempy*

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- How can we extend *Chempy* to run predict chemical abundances for **many** (~ 100) stars?

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- How can we extend *Chempy* to run predict chemical abundances for **many** (~ 100) stars?
  - Each star forms at a **different** time in a **different** ISM environment

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- How can we extend *Chempy* to run predict chemical abundances for **many** (~ 100) stars?
  - Each star forms at a **different** time in a **different** ISM environment
  - Requires free parameters **for each star** - a high-dimensional problem!

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- How can we extend *Chempy* to run predict chemical abundances for **many** (~ 100) stars?
  - Each star forms at a **different** time in a **different** ISM environment
  - Requires free parameters **for each star** - a high-dimensional problem!
- **Solution**:
  - Neural Networks for Speed (1000x speed boost)

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- How can we extend *Chempy* to run predict chemical abundances for **many** (~ 100) stars?
  - Each star forms at a **different** time in a **different** ISM environment
  - Requires free parameters **for each star** - a high-dimensional problem!
- **Solution**:
  - Neural Networks for Speed (1000x speed boost)
  - **Hamiltonian Monte Carlo** methods for sampling

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- Run 3 sets of inference:

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- Run 3 sets of inference:
    1. Mock data from *Chempy* with **correct** yield set.

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- Run 3 sets of inference:
  1. Mock data from *Chempy* with **correct** yield set.
  2. Mock data from *Chempy* with **incorrect** yield set.

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- Run 3 sets of inference:
  1. Mock data from *Chempy* with **correct** yield set.
  2. Mock data from *Chempy* with **incorrect** yield set.
  3. Mock data from **IllustrisTNG** with **correct** yield set.

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- Run 3 sets of inference:
  1. Mock data from *Chempy* with **correct** yield set.
  2. Mock data from *Chempy* with **incorrect** yield set.
  3. Mock data from **IllustrisTNG** with **correct** yield set.

- Free **SSP** parameters:

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- Run 3 sets of inference:
  1. Mock data from *Chempy* with **correct** yield set.
  2. Mock data from *Chempy* with **incorrect** yield set.
  3. Mock data from **IllustrisTNG** with **correct** yield set.

- Free **SSP** parameters:
  1. High-mass slope of the (global) IMF

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- ❑ Run 3 sets of inference:
  1. Mock data from *Chempy* with **correct** yield set.
  2. Mock data from *Chempy* with **incorrect** yield set.
  3. Mock data from **IllustrisTNG** with **correct** yield set.

- ❑ Free **SSP** parameters:
  1. High-mass slope of the (global) IMF
  2. SN1a Normalization

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- Run 3 sets of inference:
  1. Mock data from *Chempy* with **correct** yield set.
  2. Mock data from *Chempy* with **incorrect** yield set.
  3. Mock data from **IllustrisTNG** with **correct** yield set.

- Free **SSP** parameters:
  1. High-mass slope of the (global) IMF
  2. SN1a Normalization

- Other parameters:

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- Run 3 sets of inference:
  1. Mock data from *Chempy* with **correct** yield set.
  2. Mock data from *Chempy* with **incorrect** yield set.
  3. Mock data from **IllustrisTNG** with **correct** yield set.

- Free **SSP** parameters:
  1. High-mass slope of the (global) IMF
  2. SN1a Normalization

- Other parameters:
  - **Model error** parameter per element

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- ❑ Run 3 sets of inference:
  1. Mock data from *Chempy* with **correct** yield set.
  2. Mock data from *Chempy* with **incorrect** yield set.
  3. Mock data from **IllustrisTNG** with **correct** yield set.

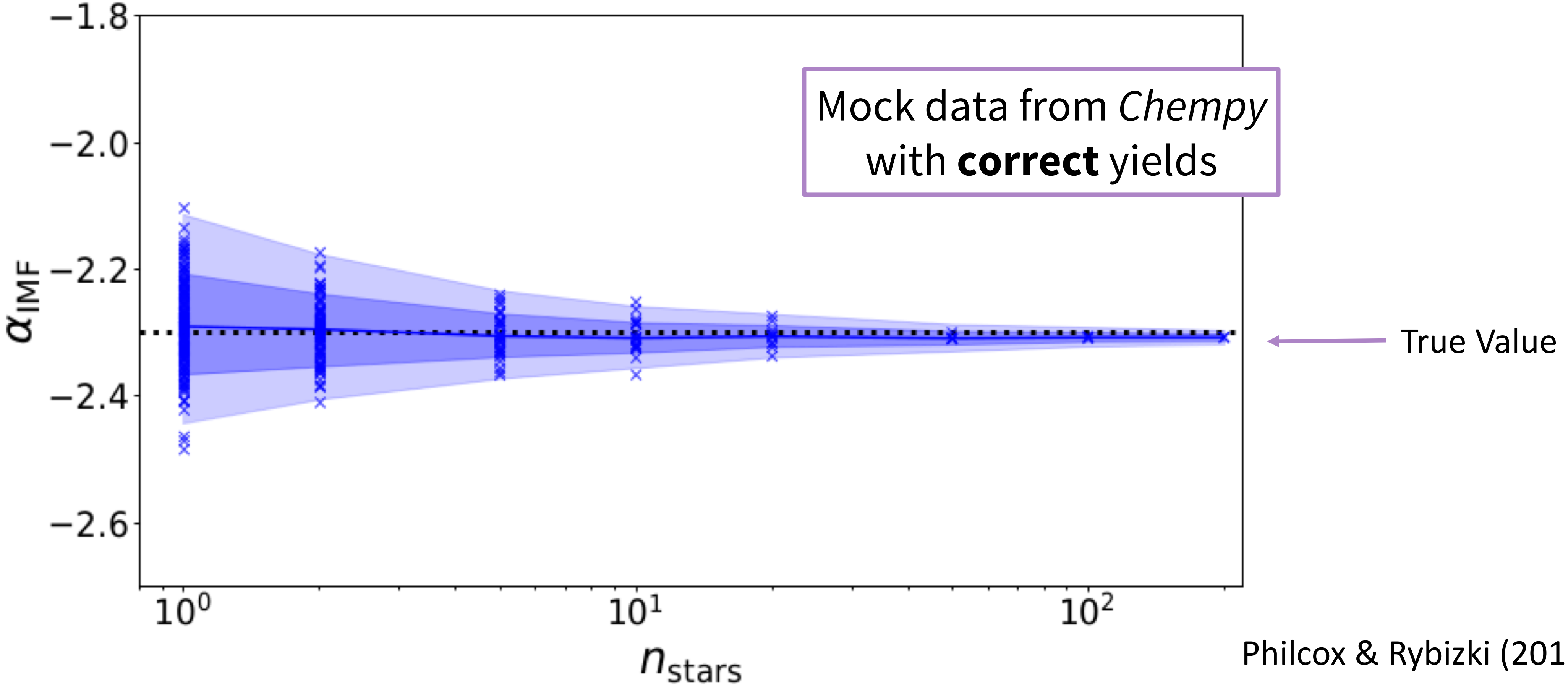
- ❑ Free **SSP** parameters:
  1. High-mass slope of the (global) IMF
  2. SN1a Normalization

- ❑ Other parameters:
  - ❑ **Model error** parameter per element
  - ❑ **ISM** parameters and **age** per star

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

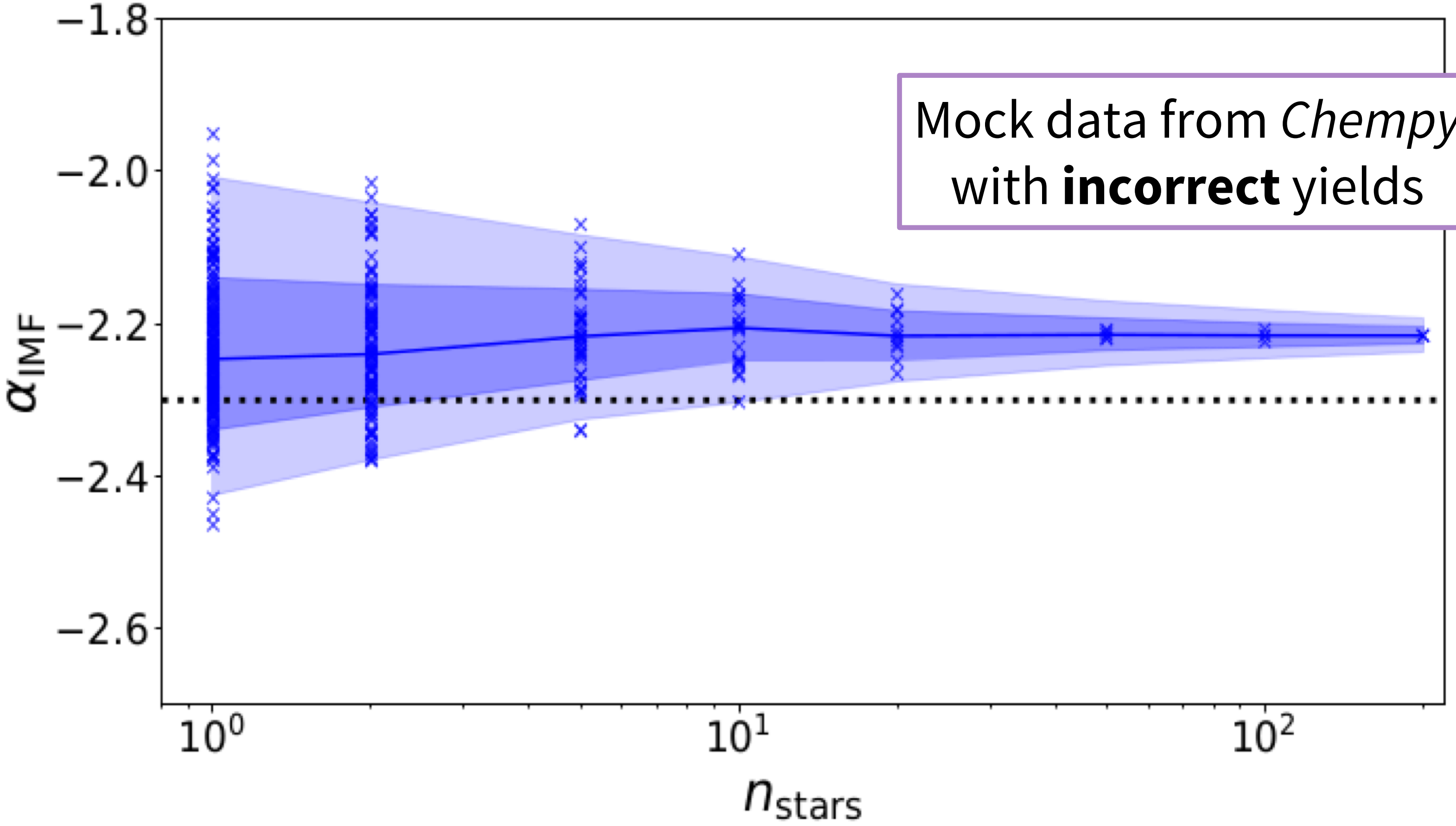
Multi-Star Inference with *Chempy*



Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*



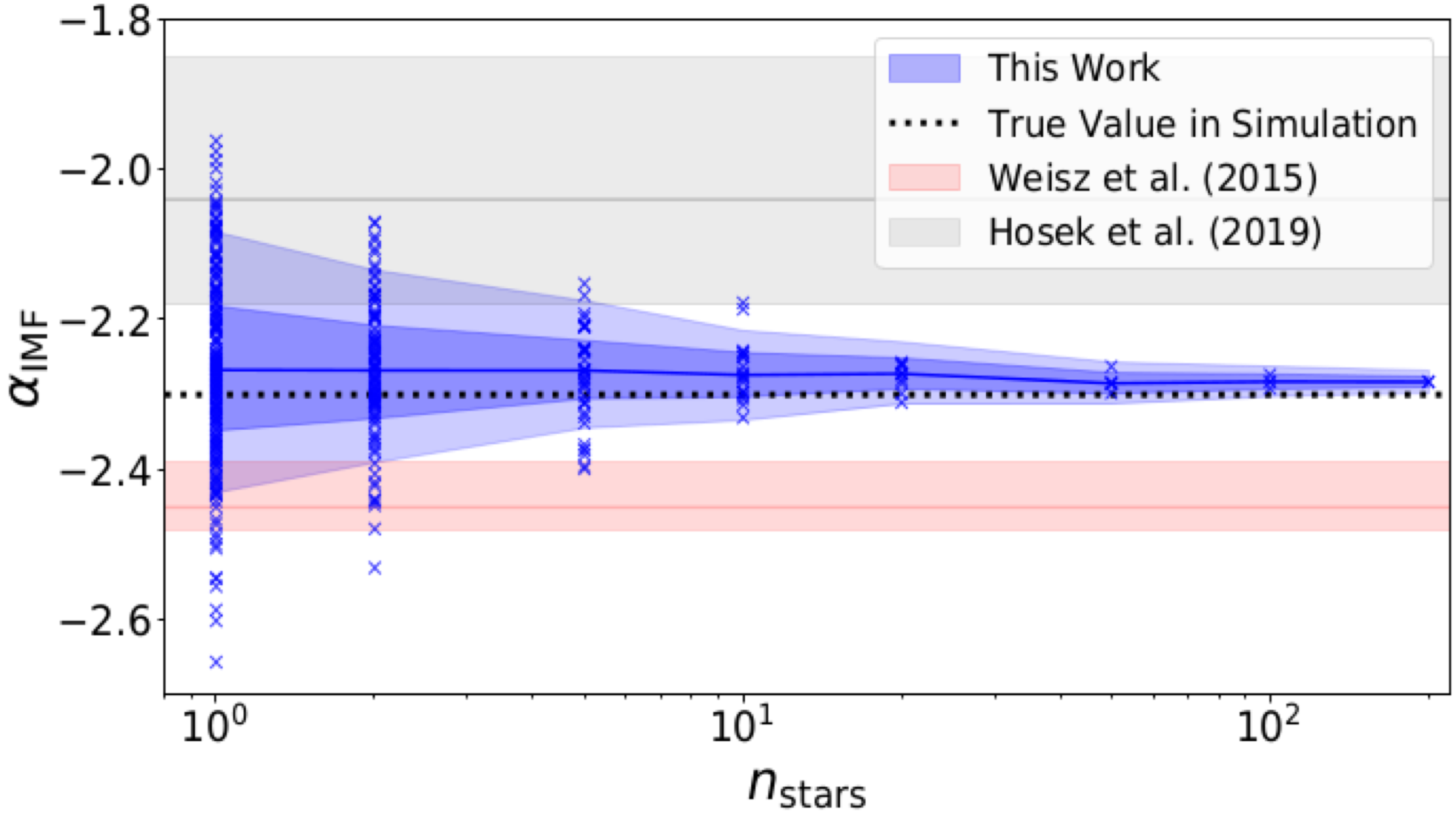
Mock data from *Chempy* with **incorrect** yields

True Value

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*



Mock data from **IllustrisTNG** with **correct** yields

Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- **Model errors** indicate errors in our yield tables.

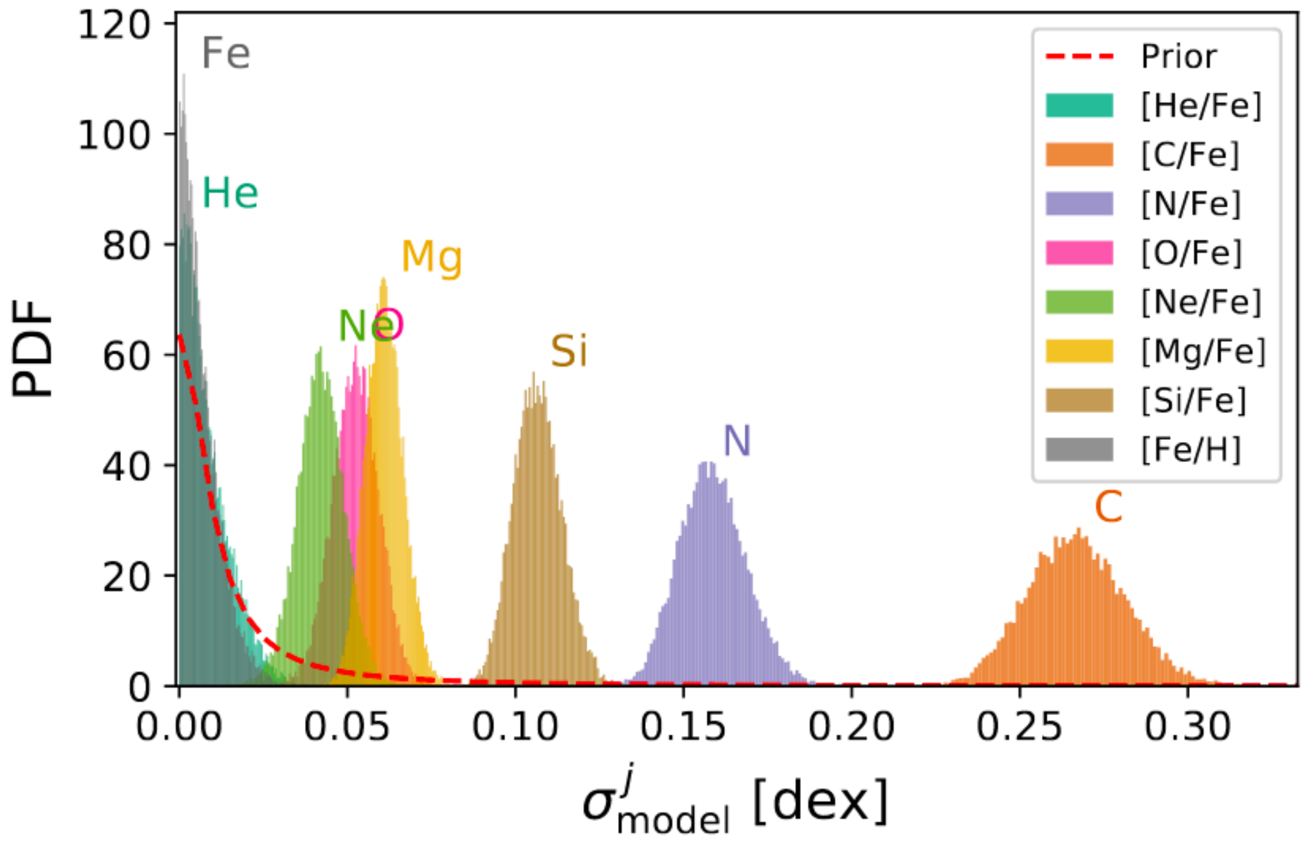
Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- **Model errors** indicate errors in our yield tables.

Model Error Distributions for analysis with **incorrect** yield set



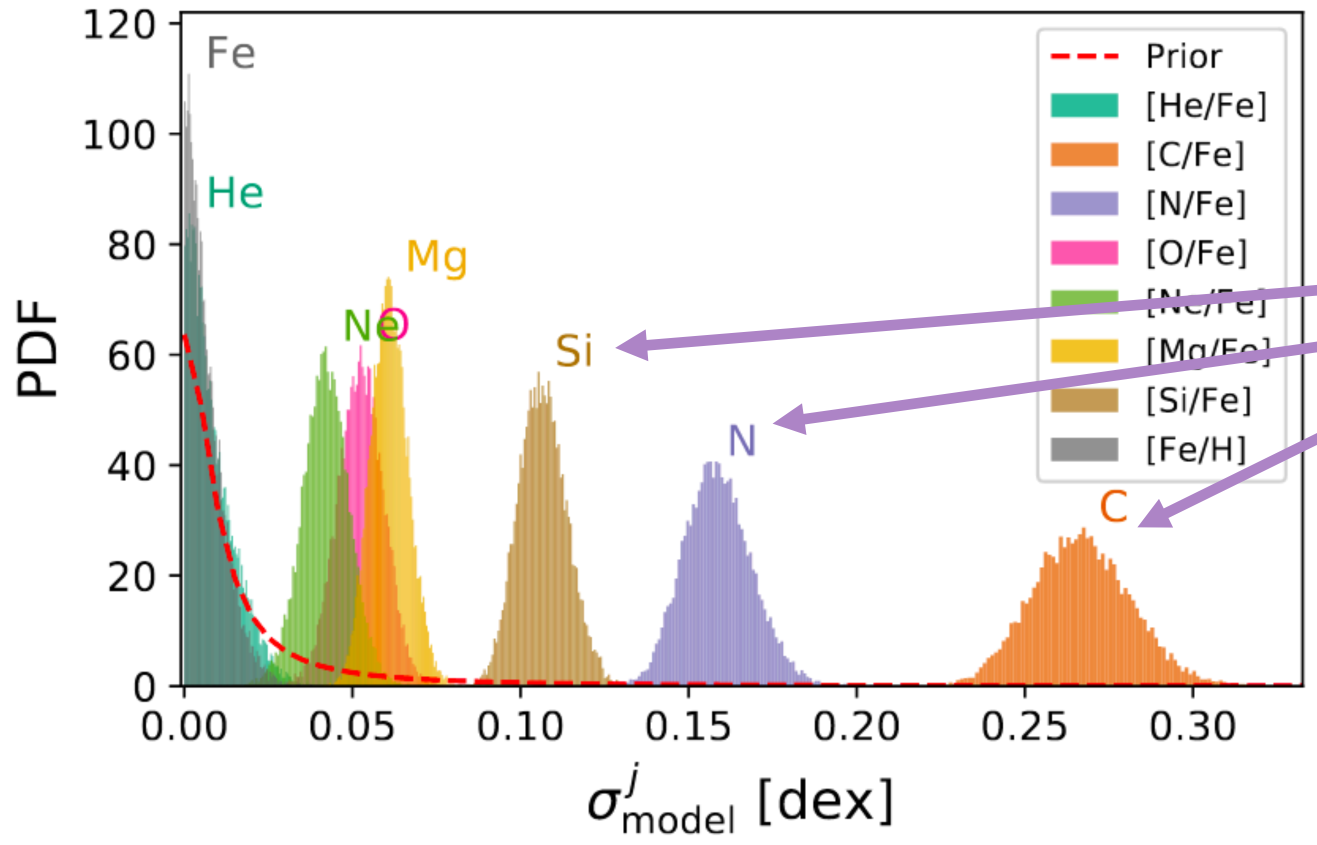
Philcox & Rybizki (2019)

github.com/oliverphilcox/ChempyMulti

Multi-Star Inference with *Chempy*

- **Model errors** indicate errors in our yield tables.

Model Error Distributions for analysis with **incorrect** yield set



Si, N and C ***are*** *the most discrepant elements between our yield sets!*

Philcox & Rybizki (2019)

What’s Next For *Chempy*?

Extensions to the *Chempy* model

- *Chempy* can be **easily** extended to incorporate more complex GCE physics.

Extensions to the *Chempy* model

- *Chempy* can be **easily** extended to incorporate more complex GCE physics.
- For example:
    - A second gas phase (e.g. *hot-cold* gas)

Extensions to the *Chempy* model

- *Chempy* can be **easily** extended to incorporate more complex GCE physics.
- For example:
  - A second gas phase (e.g. *hot-cold* gas)
  - Additional nucleosynthetic processes (e.g. Neutron star mergers, Hypernovae)

Extensions to the *Chempy* model

- *Chempy* can be **easily** extended to incorporate more complex GCE physics.
- For example:
  - A second gas phase (e.g. *hot-cold* gas)
  - Additional nucleosynthetic processes (e.g. Neutron star mergers, Hypernovae)
- Addition of **yield table uncertainties** into the GCE model

Extensions to the *Chempy* model

- *Chempy* can be **easily** extended to incorporate more complex GCE physics.
- For example:
    - A second gas phase (e.g. *hot-cold* gas)
    - Additional nucleosynthetic processes (e.g. Neutron star mergers, Hypernovae)
- Addition of **yield table uncertainties** into the GCE model
- What do <u>you</u> want to see GCE models incorporating?

Future Projects

- ❑ Run the multi-star inference on **real data** to compute Milky Way SSP parameters

Future Projects

- ❑ Run the multi-star inference on **real data** to compute Milky Way SSP parameters

- ❑ Parameter optimization of chemical evolution for hydrodynamical simulations including **SSP** parameters and **yield tables**.

Future Projects

- Run the multi-star inference on **real data** to compute Milky Way SSP parameters

- Parameter optimization of chemical evolution for hydrodynamical simulations including **SSP** parameters and **yield tables**.

- Infer the **yield tables** empirically from observational data.

Summary

- *Chempy* is a **fast** and **flexible** GCE model, allowing for Bayesian exploration of parameter space.
- Using *Chempy* we can optimize SSP parameters for **hydrodynamical simulations** to ensure they match observational data.
- By extending to **multiple stars** we can precisely constrain global Galactic parameters
- And maybe even yield tables…

Thank you for your attention!

Additional Slides

Neural Networks

- Evaluation of the *Chempy* function is slow ($\sim$ 1s)

Neural Networks

- Evaluation of the *Chempy* function is slow ($\sim$ 1s)
- Use neural networks to **predict** output chemical elements from input parameters.

Neural Networks

- Evaluation of the *Chempy* function is slow ($\sim$ 1s)
- Use neural networks to **predict** output chemical elements from input parameters.
- This acts as a **fast**, **non-linear** interpolator, which is **differentiable**.

Neural Networks

- Evaluation of the *Chempy* function is slow (~ 1s)
- Use neural networks to **predict** output chemical elements from input parameters.
- This acts as a **fast**, **non-linear** interpolator, which is **differentiable**.

Input

Hidden Layer → $$\mathbf{h} = \mathbf{W}_0 \cdot \mathbf{x} + \mathbf{b}_0$$

Output → $$\mathbf{y} = \mathbf{W}_1 \cdot f(\mathbf{h}) + \mathbf{b}_1$$

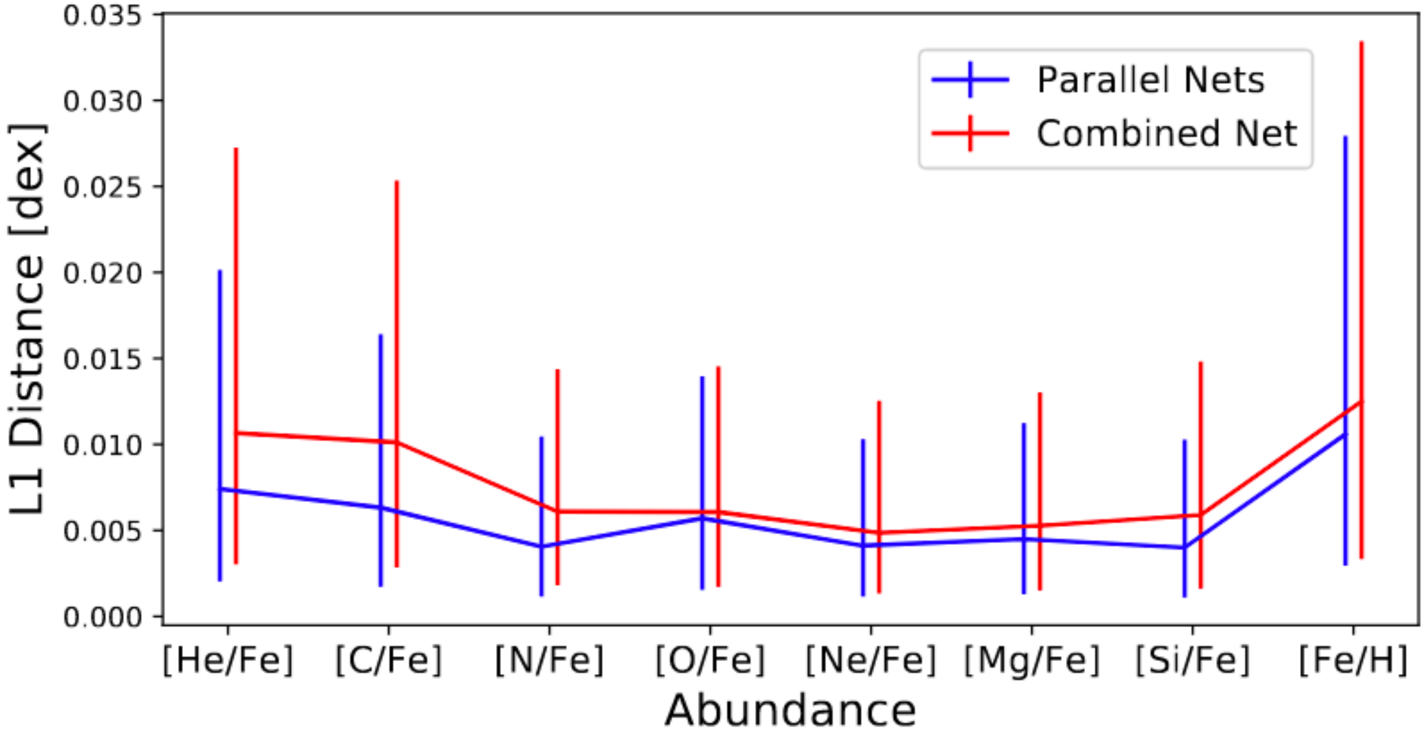
Non-Linearity

Neural Networks

- This is trained by running *Chempy* on $\sim 10^4$ points in parameter space.

Neural Networks

- This is trained by running *Chempy* on ~10^4 points in parameter space.



Errors are much below observational error

Philcox & Rybizki (2019)

Hamiltonian Monte Carlo (HMC)

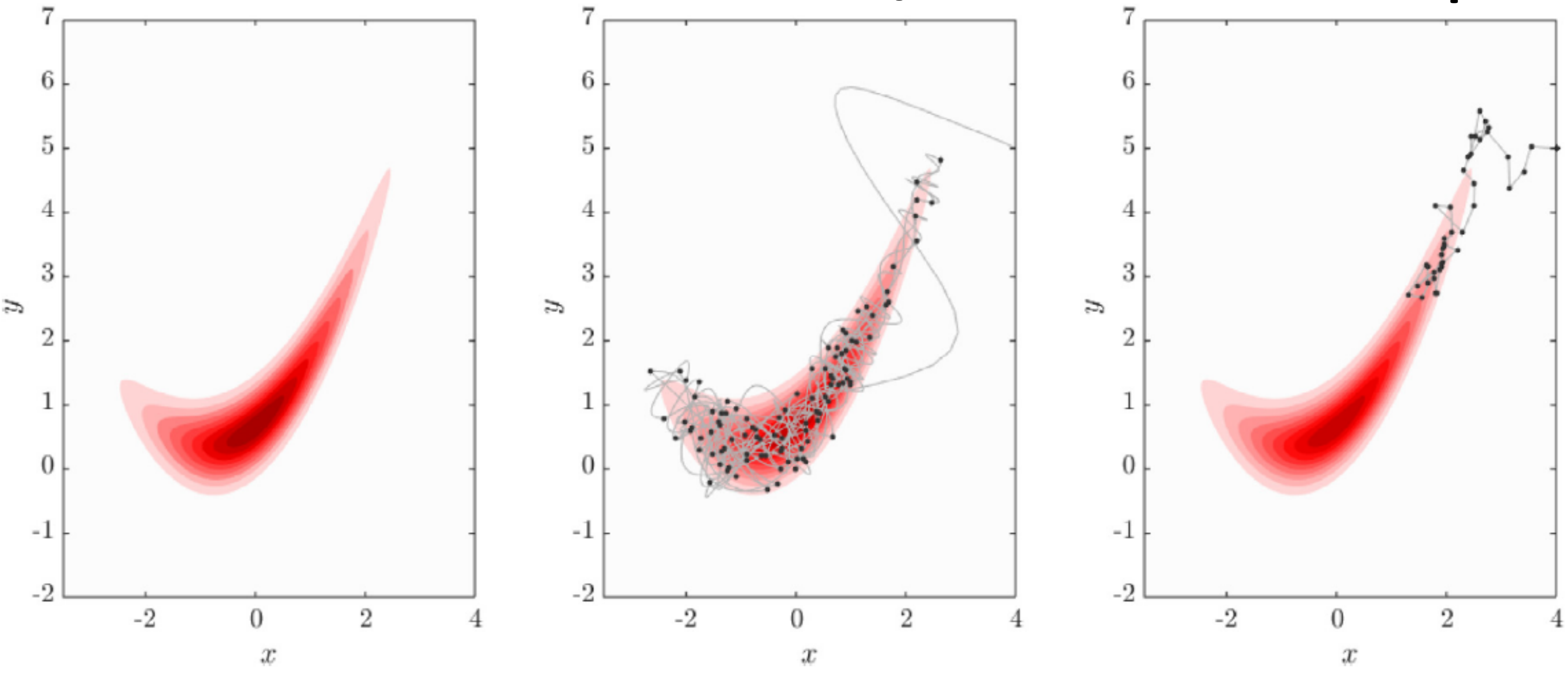
- Markov Chain Monte Carlo (MCMC) is **slow** and **unsuitable** for high-dimensional problems.
- MCMC works by jumping between points in parameter space at random.
- HMC preferentially samples where the posterior is **large.**
- It's **much more efficient** but requires a **differentiable** model.

Betancourt (2017)

Hamiltonian Monte Carlo (HMC)

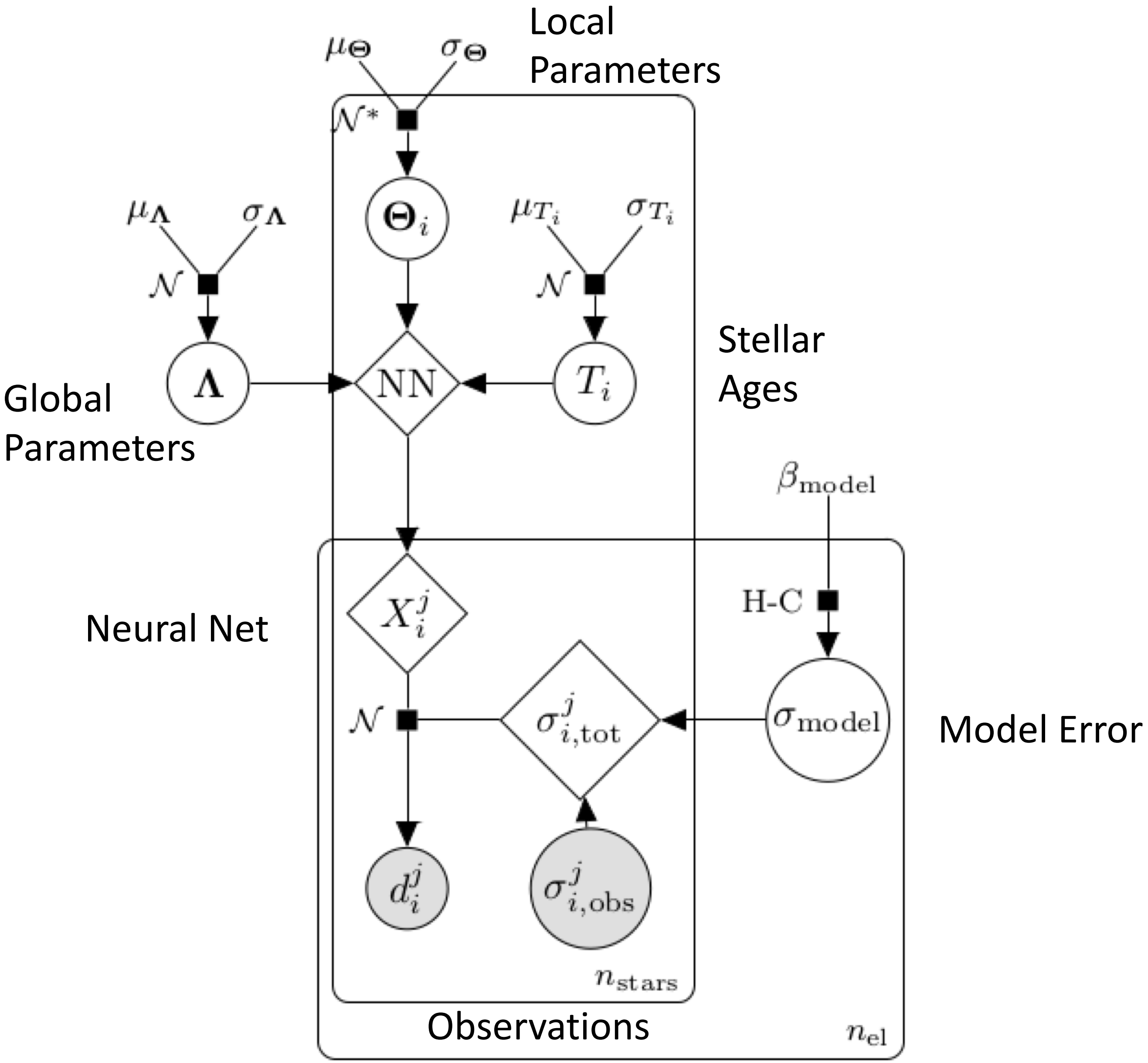
- ❑ HMC preferentially samples where the posterior is **large.**

Posterior **HMC Samples** **MCMC Samples**

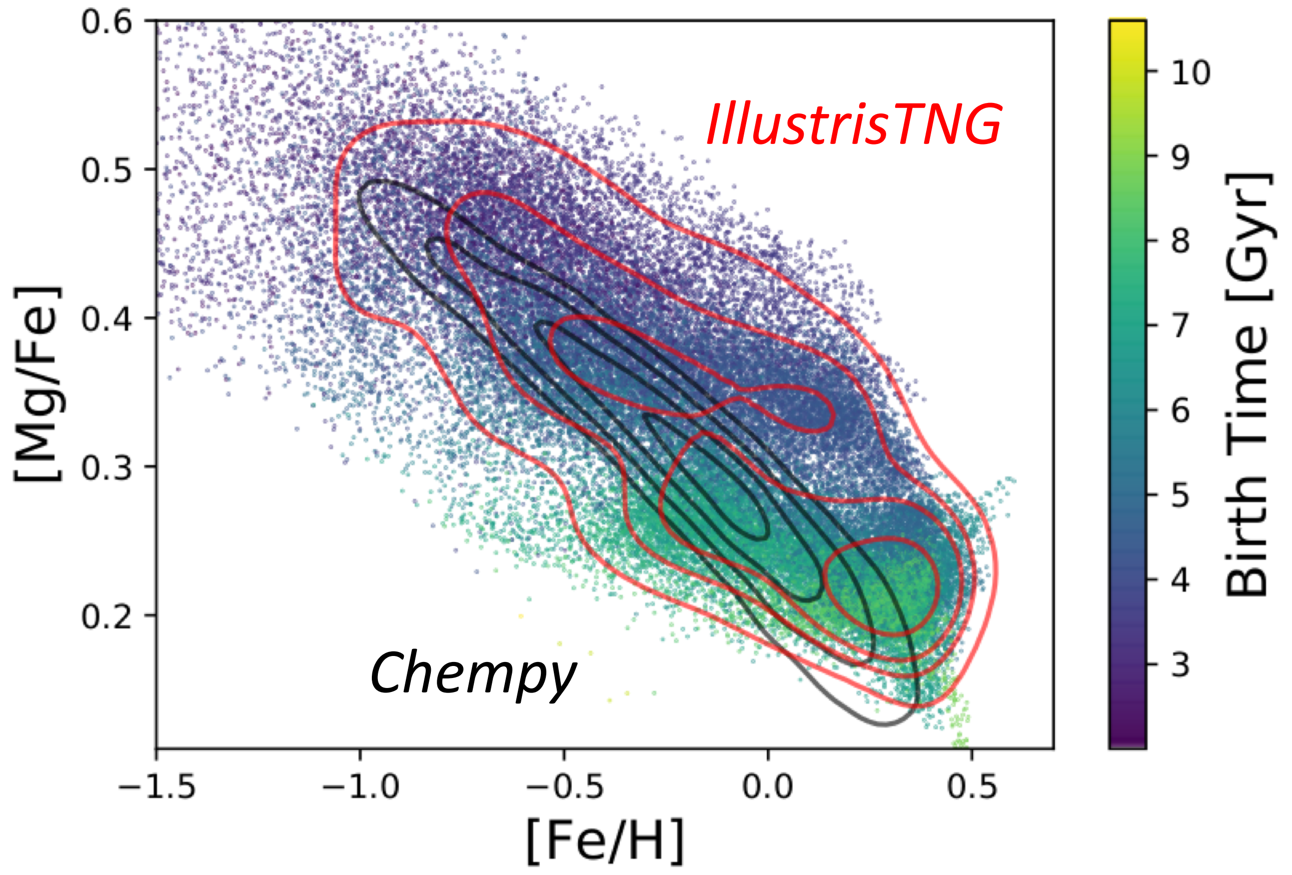


Wang+ (2019)

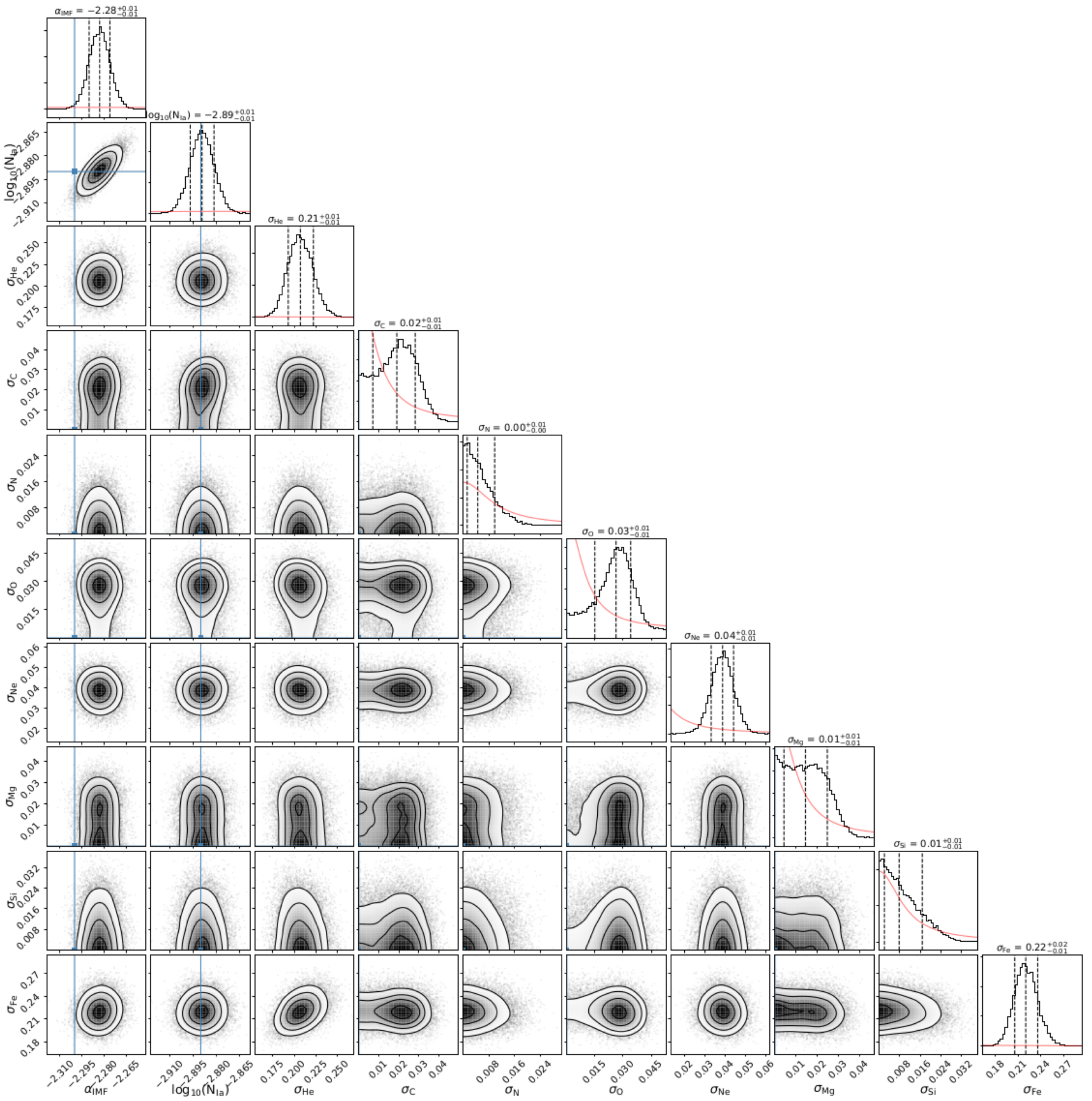
ChempyMulti Architecture



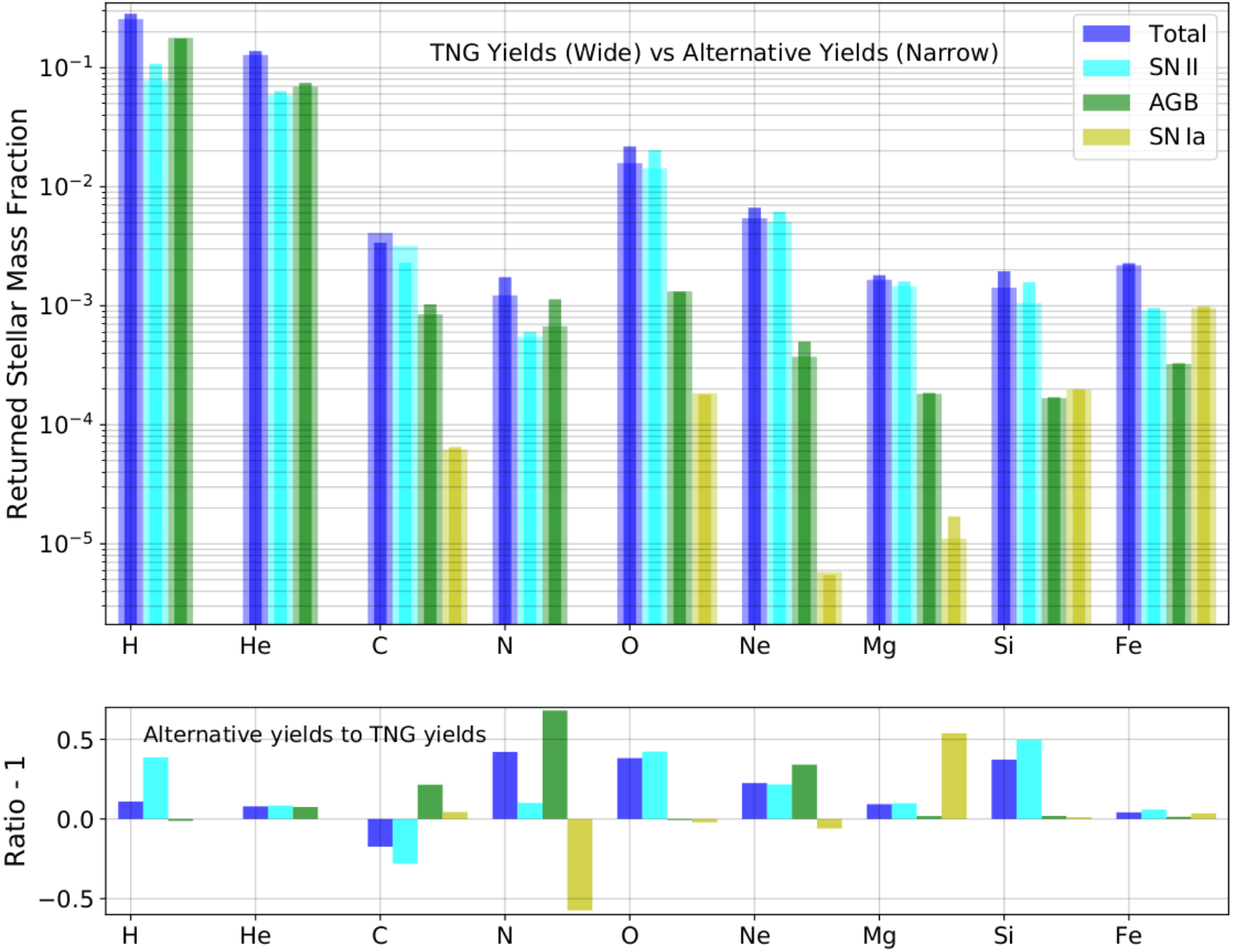
Abundance Diagrams



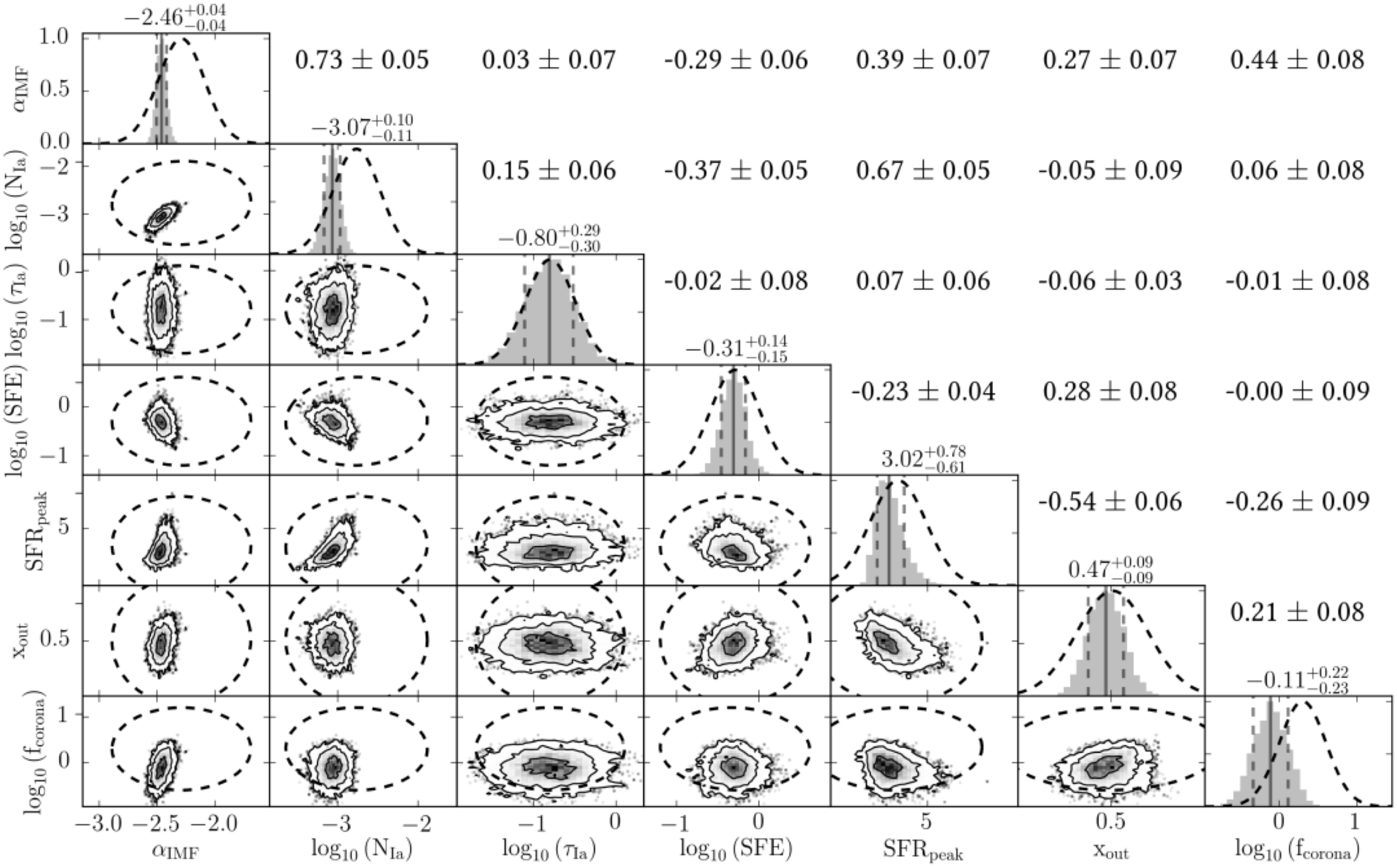
Full Corner Plot



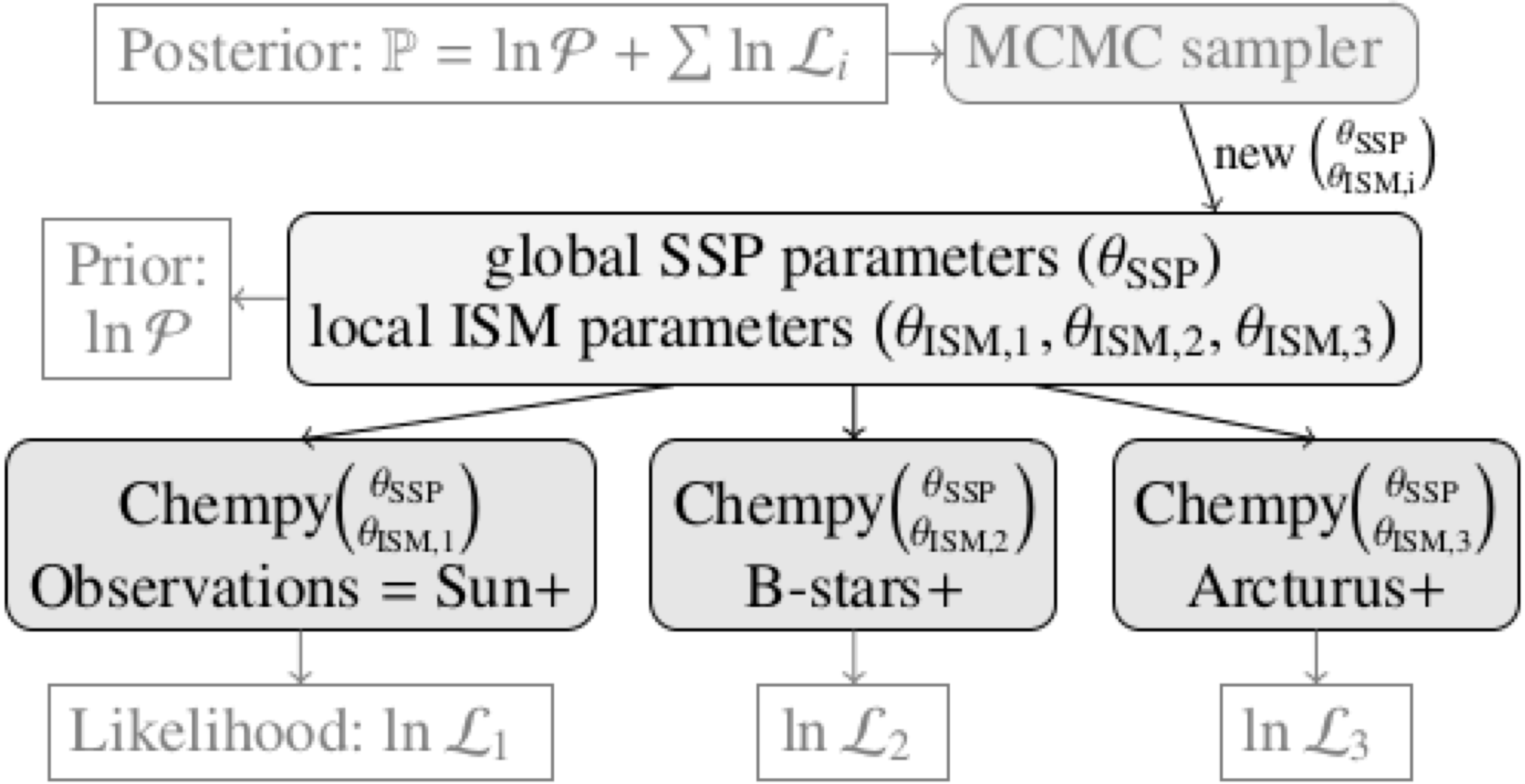
Default and Alternative Yields



Single Star Corner Plots



The *Chempy* Model



Full Hydrodynamical Simulation Optimization