# SOLAR SYSTEM EPHEMERIS NOISE

Stephen Taylor

NANOGrav PFC Senior Postdoctoral Fellow

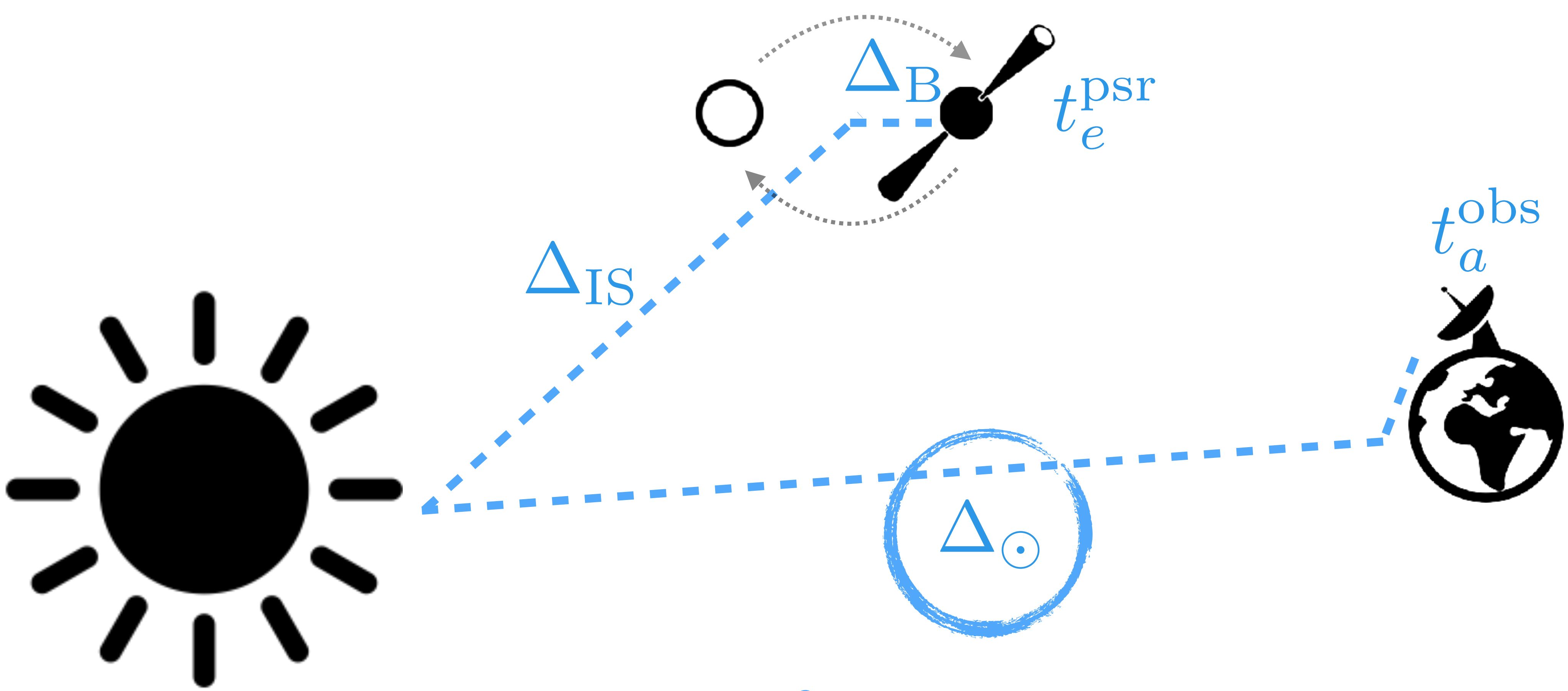
CALIFORNIA INSTITUTE OF TECHNOLOGY

Chair, NANOGrav GW Group
Co-chair, International Pulsar Timing Array GW Group

# The Solar-System Ephemeris

Tracing a **TOA** back from an **observatory to the emission time at the pulsar** involves a chain of corrections

$$t_e^{\mathrm{psr}} = t_a^{\mathrm{obs}} - \Delta_{\odot} - \Delta_{\mathrm{IS}} - \Delta_{\mathrm{B}}$$



# The Solar-System Ephemeris

1. all TOAs are referenced to the **quasi-inertial frame of the SSB** (need Roemer delay)
2. Roemer delay dependent on **masses & orbits** of all important dynamical objects
3. don’t need SSB to navigate probes to planets (accurate SSB is not a big priority)
4. the **Roemer is not fit for in Tempo2**, it is subtracted from pre-fit JPL solutions

# The Solar-System Ephemeris

Roemer delay

$$\Delta_\odot = -\frac{\vec{r}^{\rm obs} \cdot \vec{R}^{\rm BB}}{c}$$

Observatory position

$$\vec{r}^{\rm obs} = \vec{r}^{\rm SSB-EB} + \vec{r}^{\rm EB-obs}$$

Barycenter position dependent on **masses & orbits** of all important dynamical objects

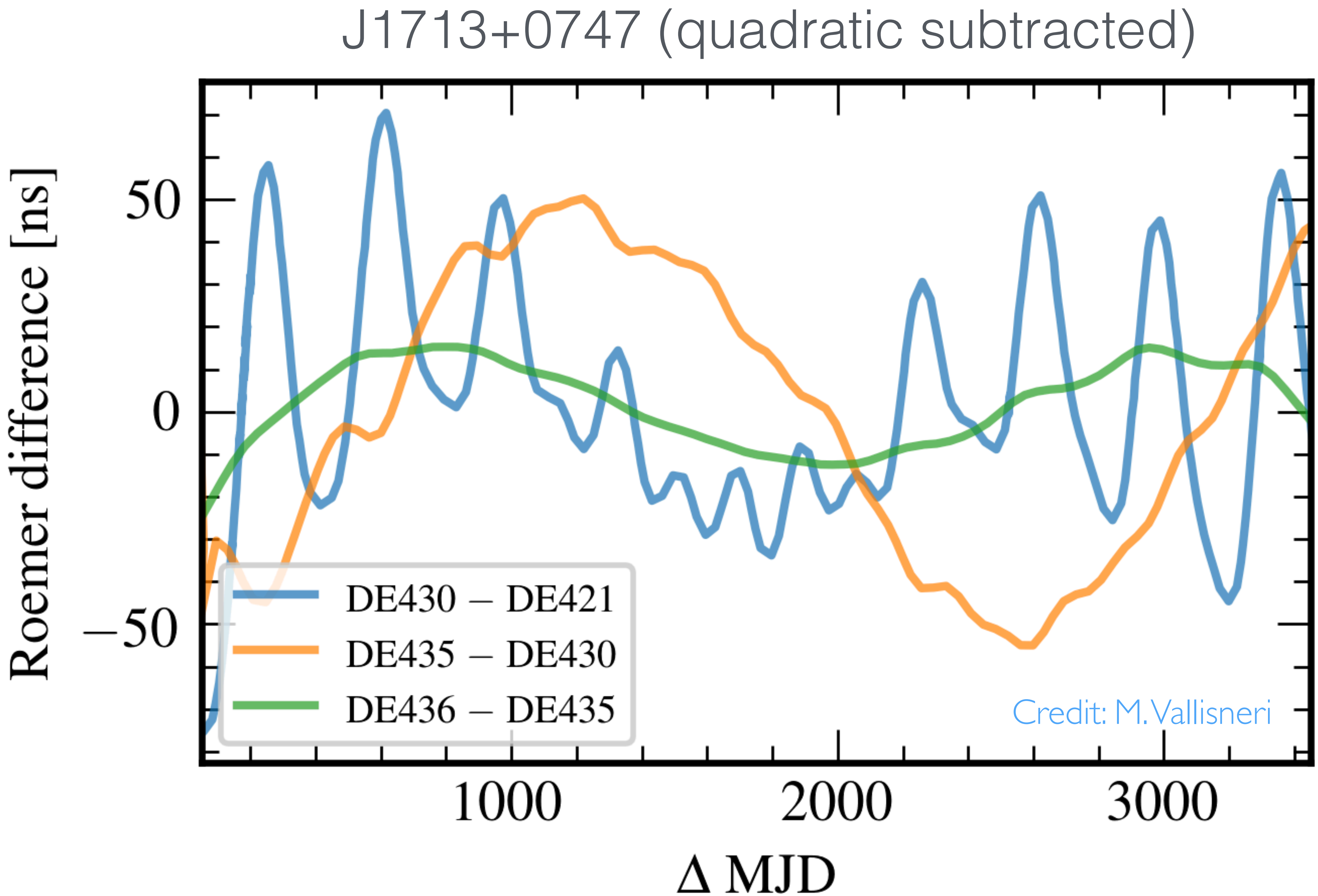
Small error in barycenter position

$$\delta\Delta_\odot = \frac{\vec{e}(t) \cdot \vec{R}^{\rm BB}}{c}$$

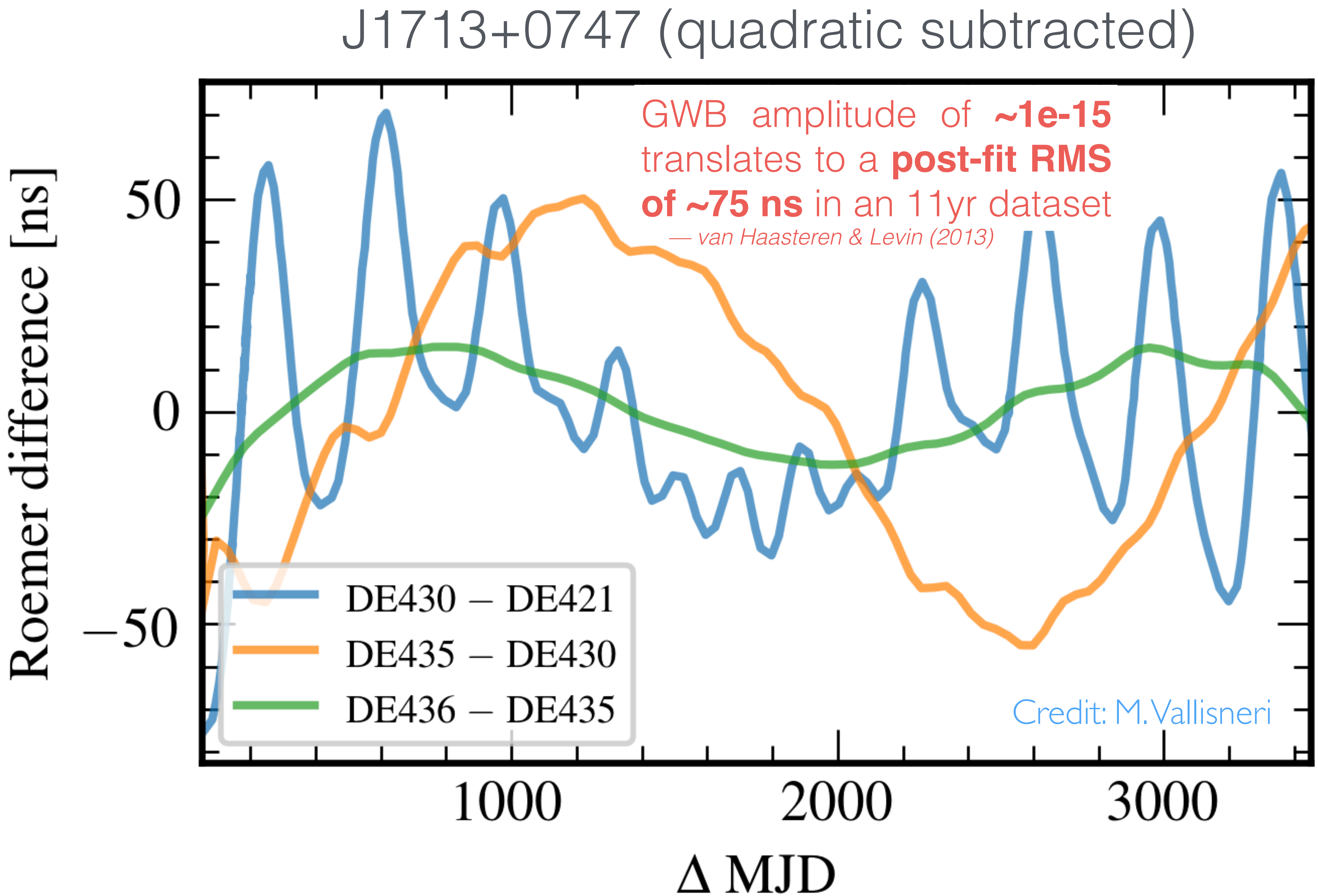
# JPL Ephemerides

| | |
|---|---|
| **DE421** | includes updates to Saturn's orbit. Dominant uncertainty likely to be Jupiter |
| **DE430** | includes updates to Mercury's orbit. Dominant uncertainty still likely to be Jupiter |
| **DE435** | created in Jan 2016 for Cassini, this is an incremental improvement to Saturn |
| **DE436** | incremental improvement to DE435 |
| **DE438** | includes Juno corrections for Jupiter |

# JPL Ephemerides

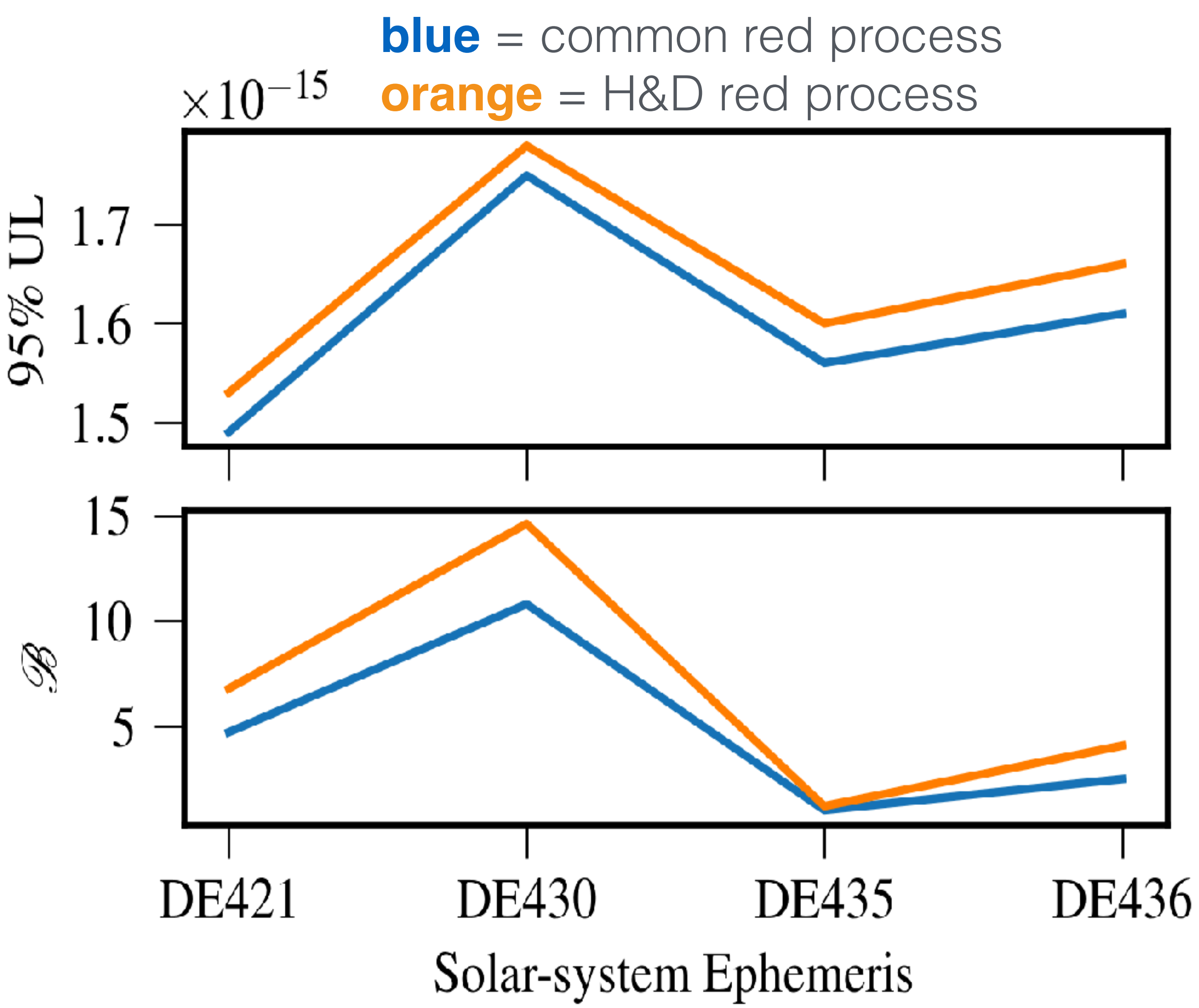


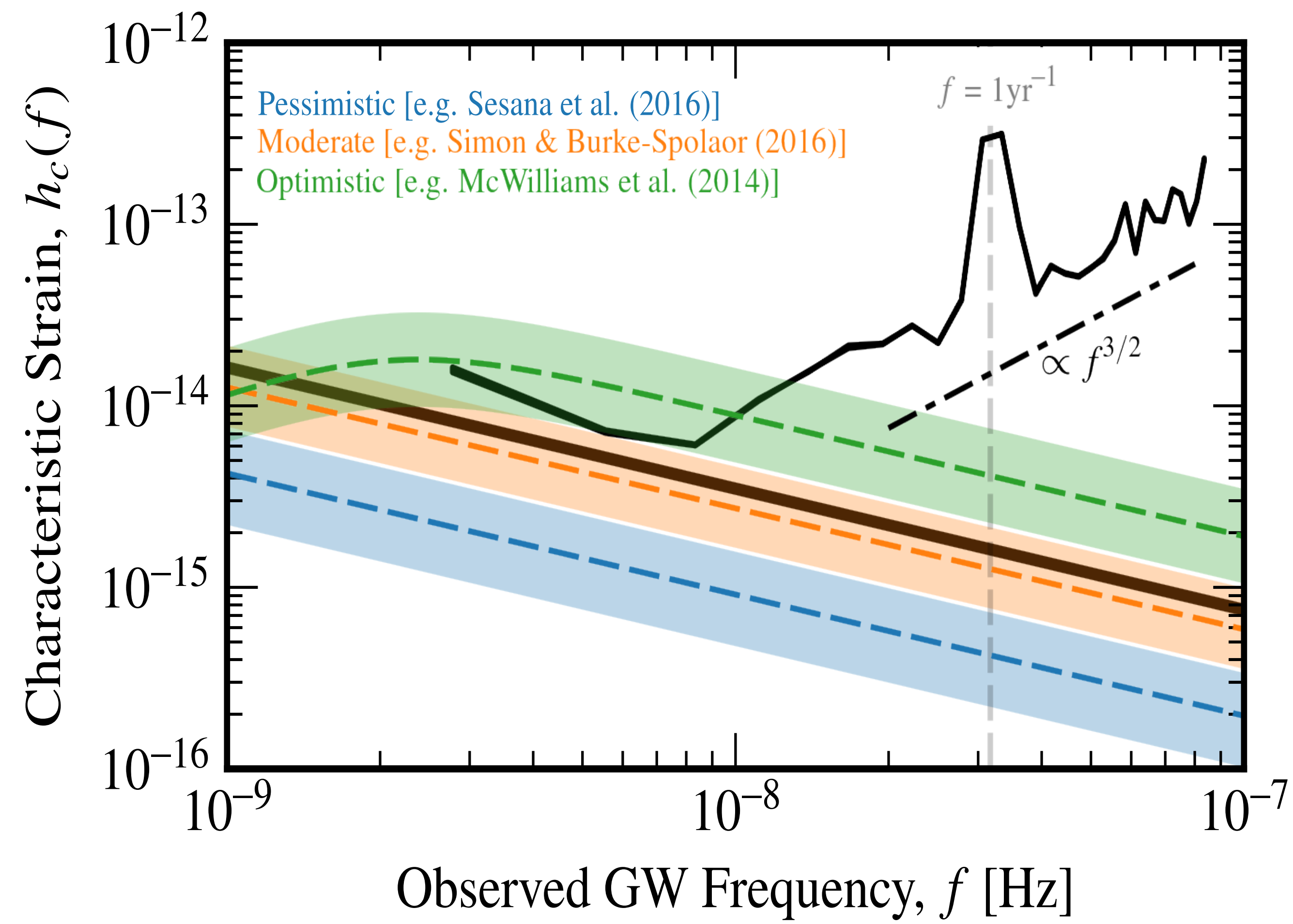
# JPL Ephemerides



# NANOGrav 11-year Results

**upper limits** and **detection statistics** are sensitive to our choice of ephemeris model




…mitigated by our **new Bayesian ephemeris uncertainty model**

# Modeling Ephemeris Uncertainties

**Current Bayesian Model**

- timing model
- white noise
- intrinsic red noise
- common red noise (or GWB)

+

**Expanded Model**

- ephemeris uncertainty term

**GOAL** *marginalize over ephemeris differences to isolate GW signal from choice of DE—*

# Modeling Ephemeris Uncertainties

ephemeris uncertainty term

## physically motivated

- *Fourier expansion of barycenter error vector [Lentati, Taylor, Mingarelli et al. (2015)]*
- *planet mass perturbation [Champion et al. (2010)]*
- *dipolar spatially-correlated red process [Tiburzi et al. (2016)])*

## phenomenological

- *Roemer mixture model*
- *PCA of Roemer delays from DE421, DE430, etc. to construct empirical basis*
- *[maybe] PCA of Roemer delays from many, many perturbed ephemerides*

# Modeling Ephemeris Uncertainties

ephemeris uncertainty term

**physically motivated**

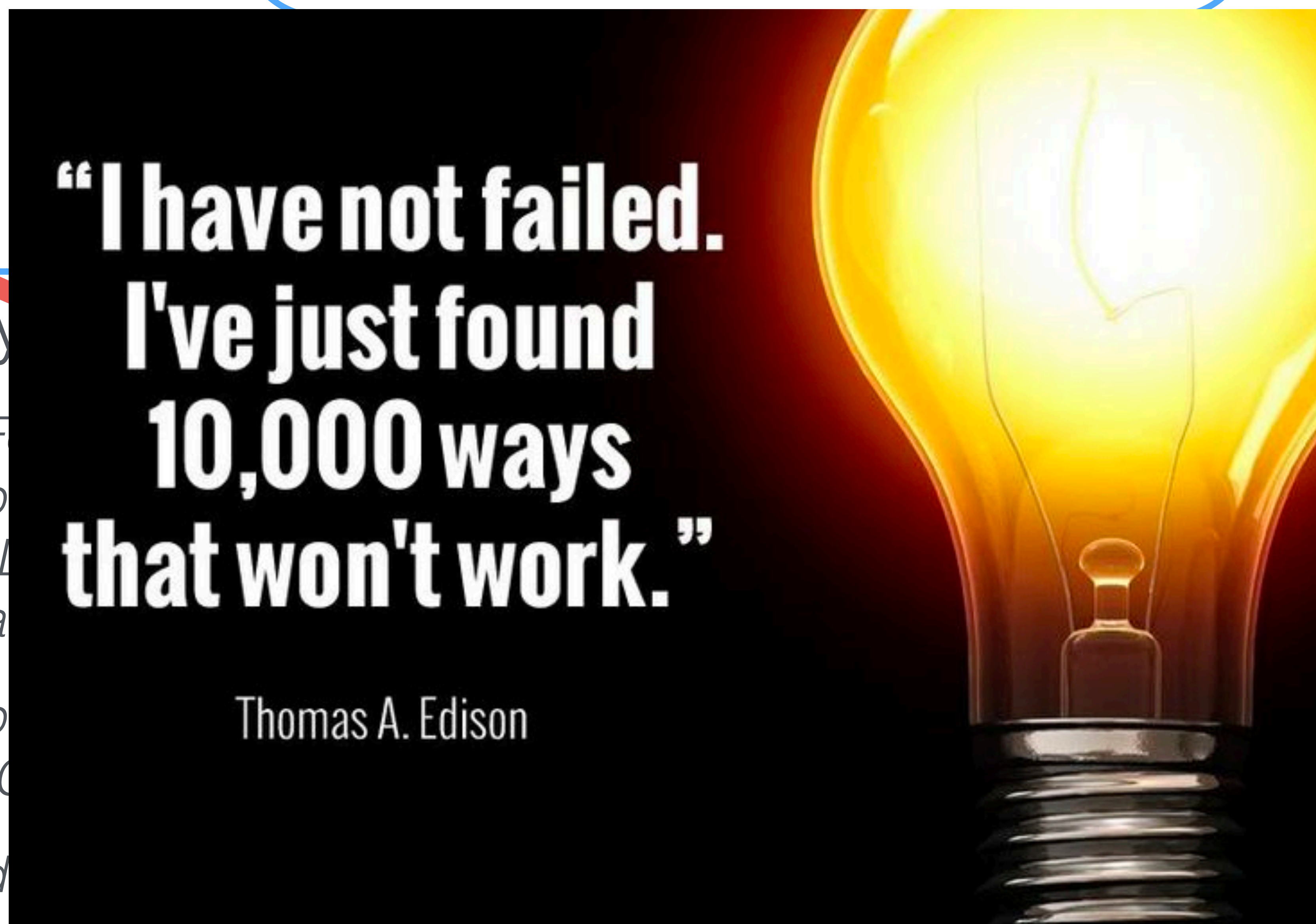
- *Fourier expansion of barycenter error vector [Lentati, Taylor, Mingarelli et al. (2015)]*
- *planet mass perturbation [Champion et al. (2010)]*
- *dipolar spatially-correlated red process [Tiburzi et al. (2016)])*

**phenomenological**

- *Roemer mixture model*
- *PCA of Roemer delays from DE421, DE430, etc. to construct empirical basis*
- *[maybe] PCA of Roemer delays from many, many perturbed ephemerides*

# Modeling Ephemeris Uncertainties

ephemeris uncertainty term




phy

- F
  b
  [D
  a
- p
  [C
- d
  process [Tiburzi et al. (2016)]

ical

- odel
- lays from
  c. to
  l basis
- pemer
  many
  ides

# Physical Ephemeris Uncertainty Model

Model is **11-D**

**1** frame drift-rate about ecliptic "z"

**1** Jupiter mass perturbation *(constrained by IAU prior)*

**1** Saturn mass perturbation *(constrained by IAU prior)*

**1** Uranus mass perturbation *(constrained by IAU prior)*

**1** Neptune mass perturbation *(constrained by IAU prior)*

**6** Jupiter orbital element perturbations

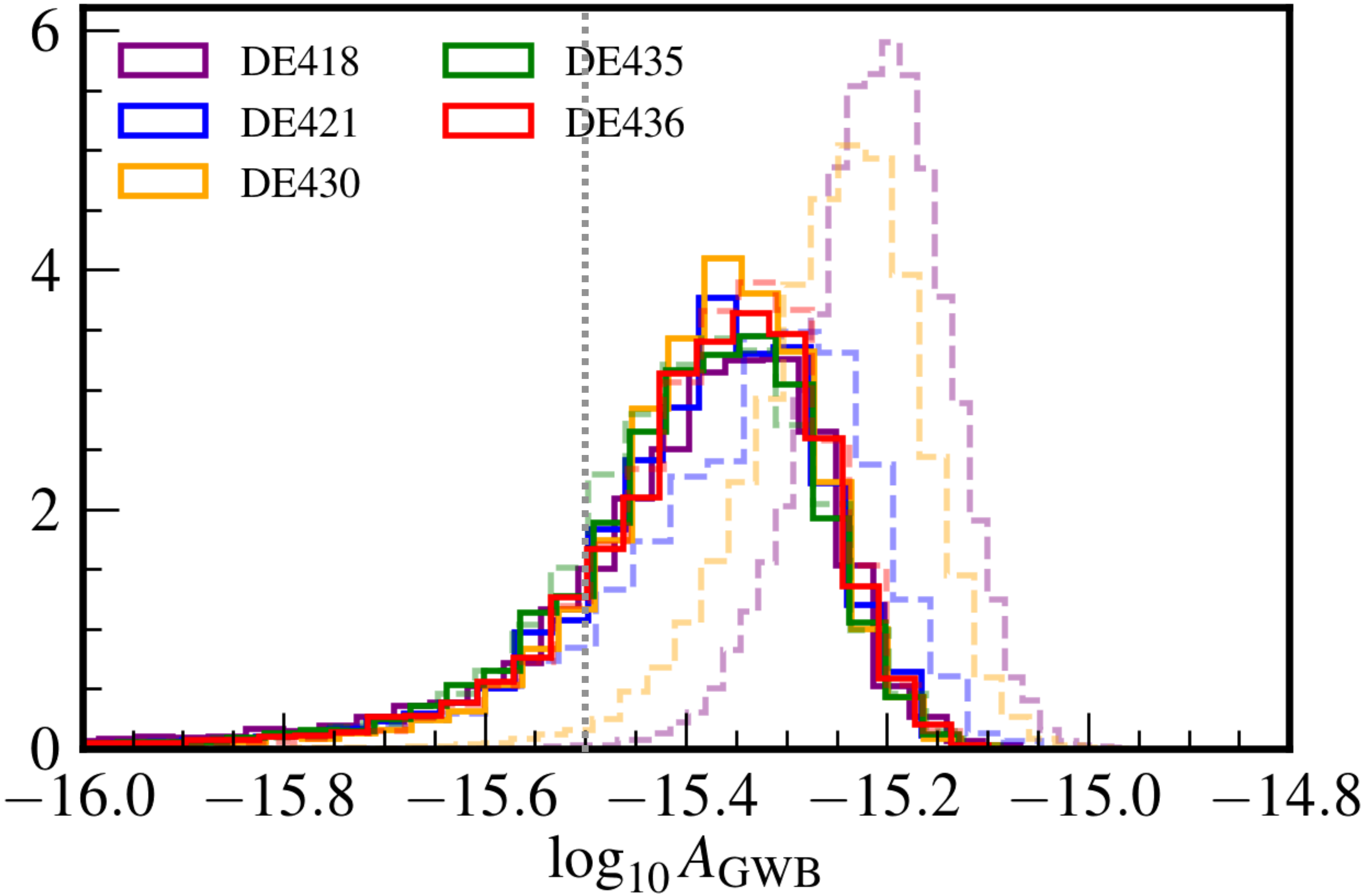
*(1) semi-major axis*
*(2) eccentricity*
*(3) inclination*
*(4) longitude of the ascending node*
*(5) longitude of perihelion*
*(6) mean longitude*

# Physical Ephemeris Uncertainty Model

— 36 pulsars
— 11 years
— equally sampled w/ 500 ns precision
— dataset created under DE436

— dashed = no ephemeris uncertainty modeling
— solid = physical ephemeris uncertainty model

**moderate GWB injection**

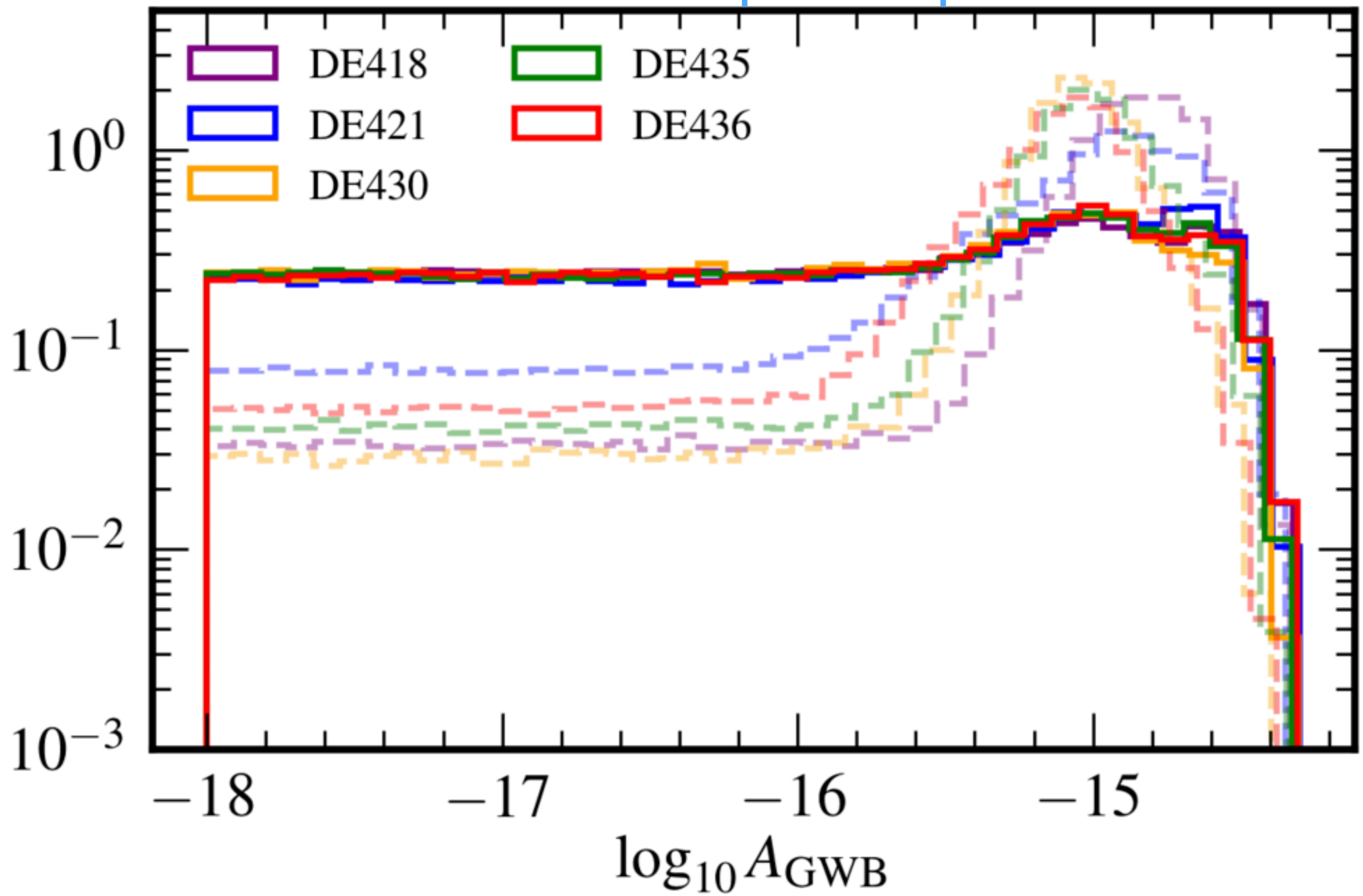


# Physical Ephemeris Uncertainty Model

## 11-year dataset simulations

*(created with exactly the same pulsars, noise properties, and sensitivity as the real dataset)*

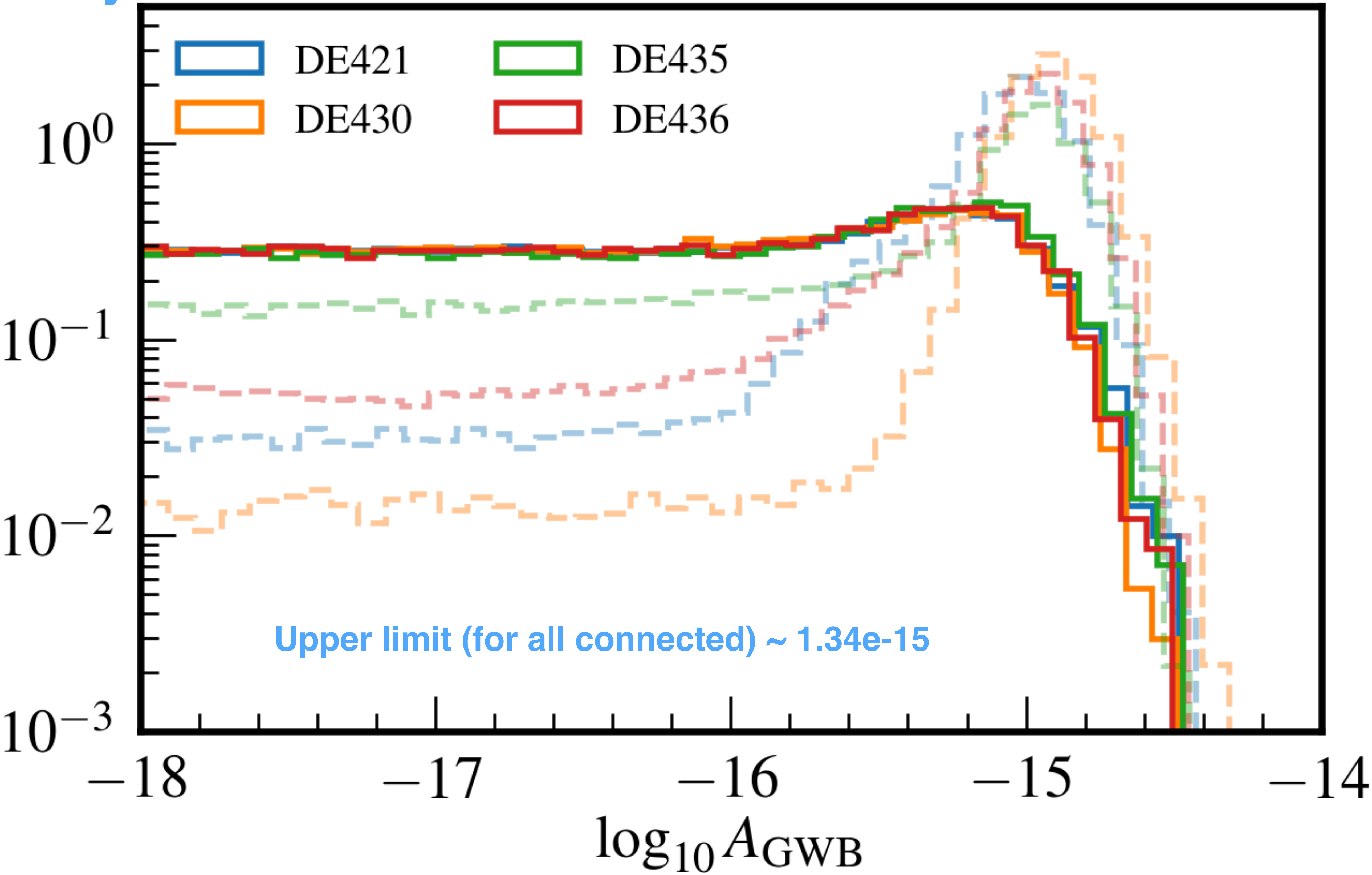
**Uninformative Jupiter orbit priors**



# Physical Ephemeris Uncertainty Model

## 11-year

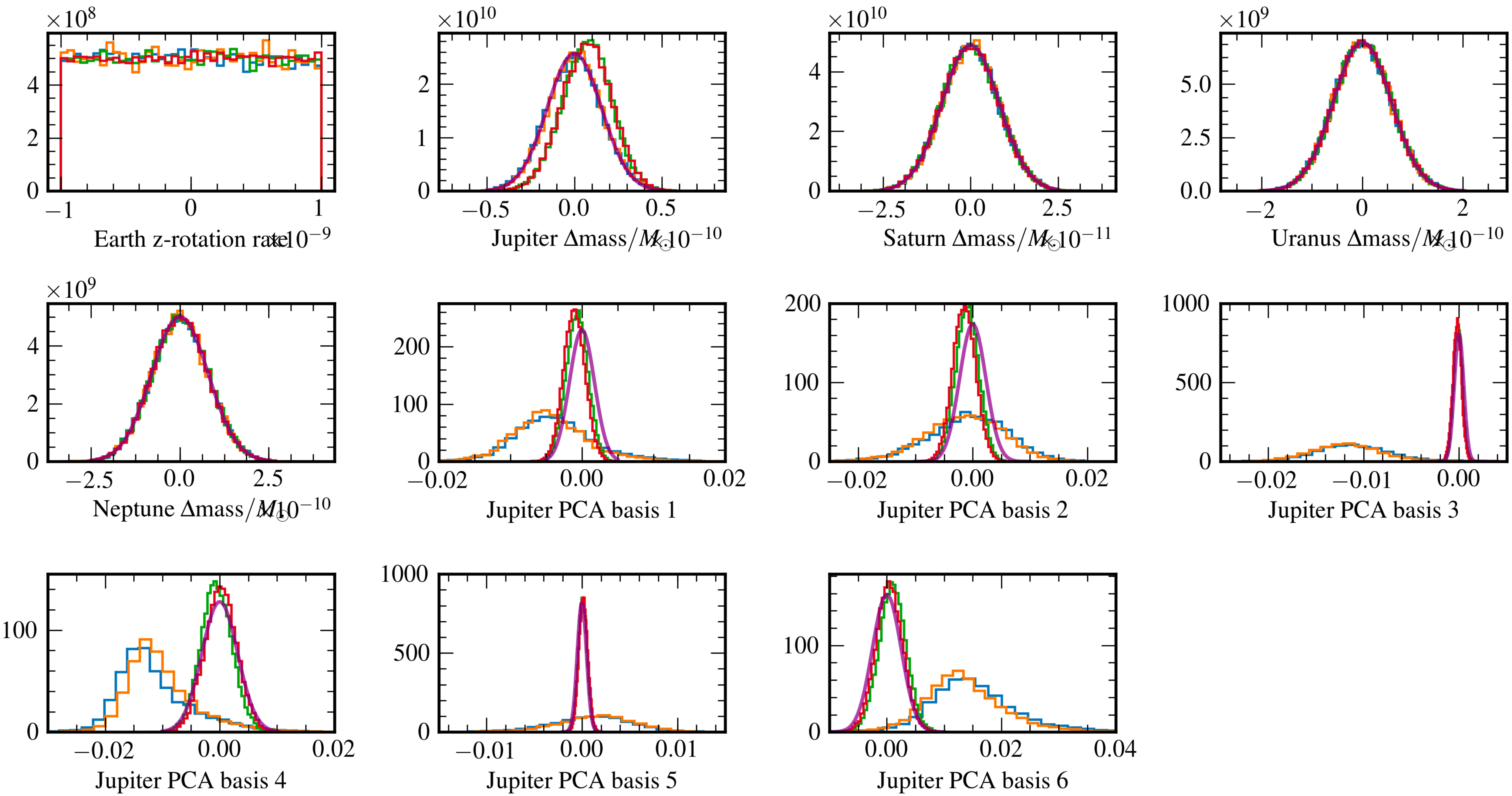
Uninformative Jupiter orbit priors



# Physical Ephemeris Uncertainty Model

## 11-year

**purple = prior distribution**
**blue = DE435 (uninformative prior)**
**orange = DE436 (uninformative prior)**
**green = DE435 (JPL prior)**
**red = DE436 (JPL prior)**

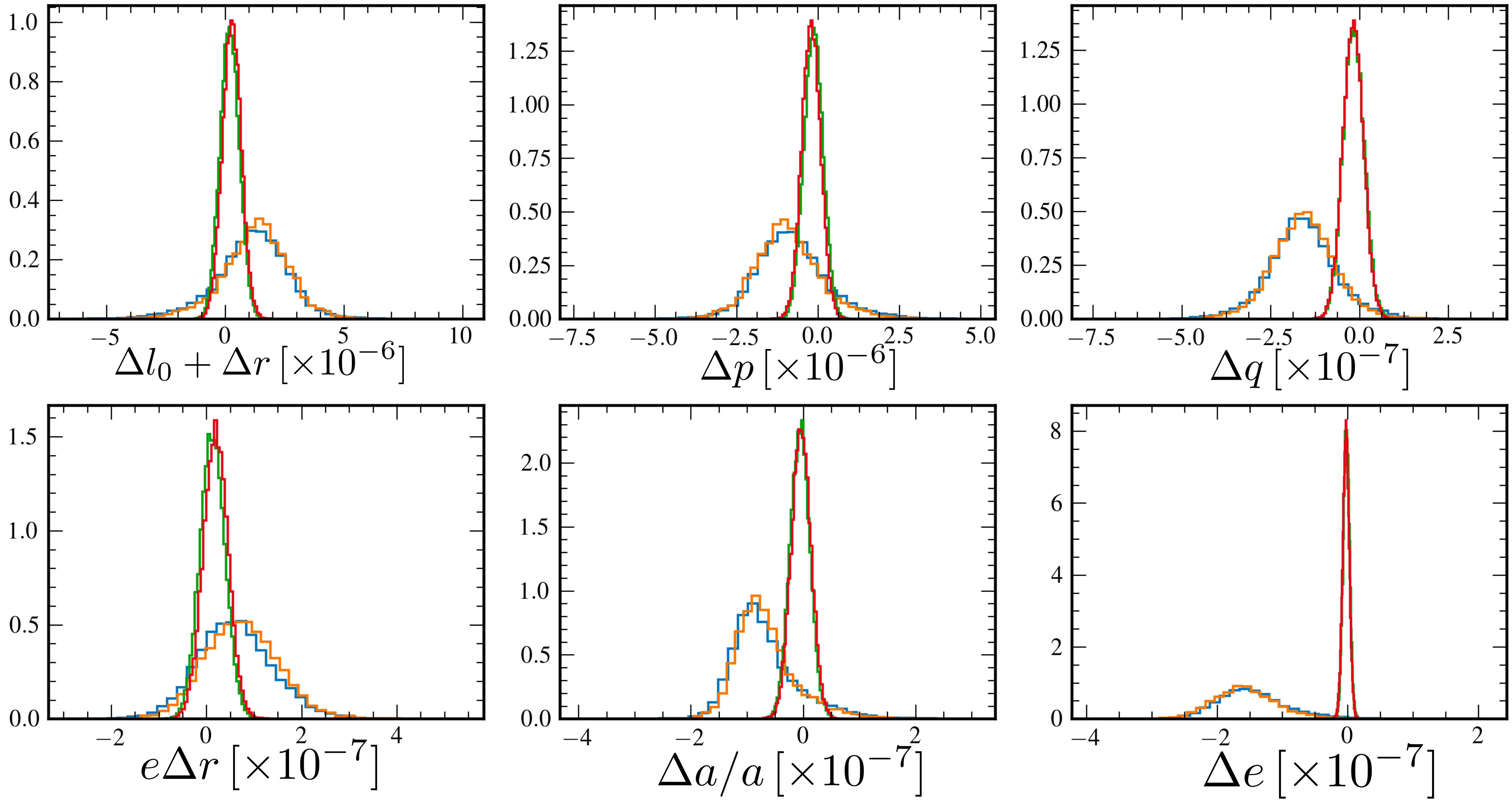


# Physical Ephemeris Uncertainty Model

## 11-year

purple = prior distribution
blue = DE435 (uninformative prior)
orange = DE436 (uninformative prior)
green = DE435 (JPL prior)
red = DE436 (JPL prior)

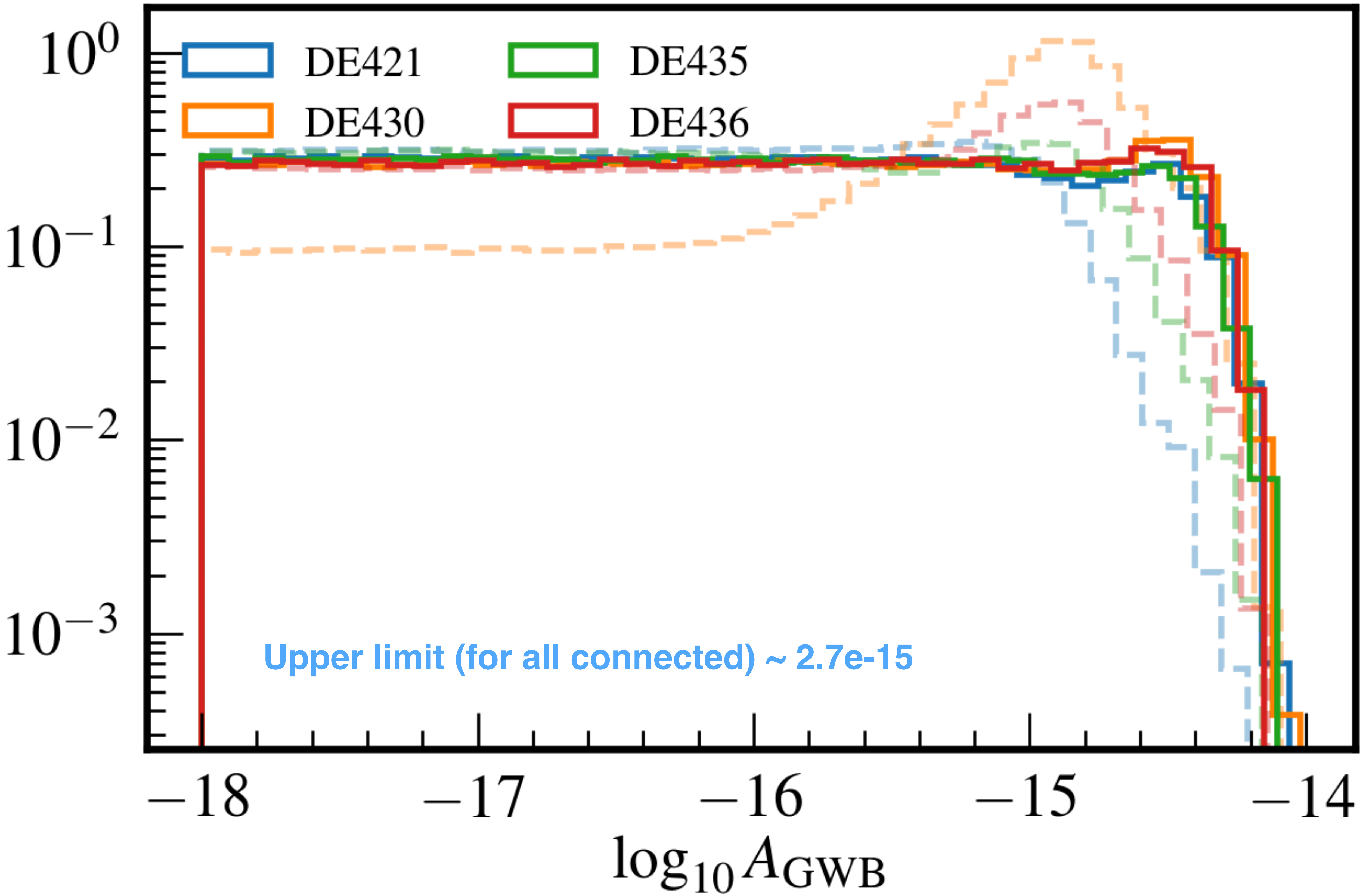
**Set III celestial mechanics coordinates**



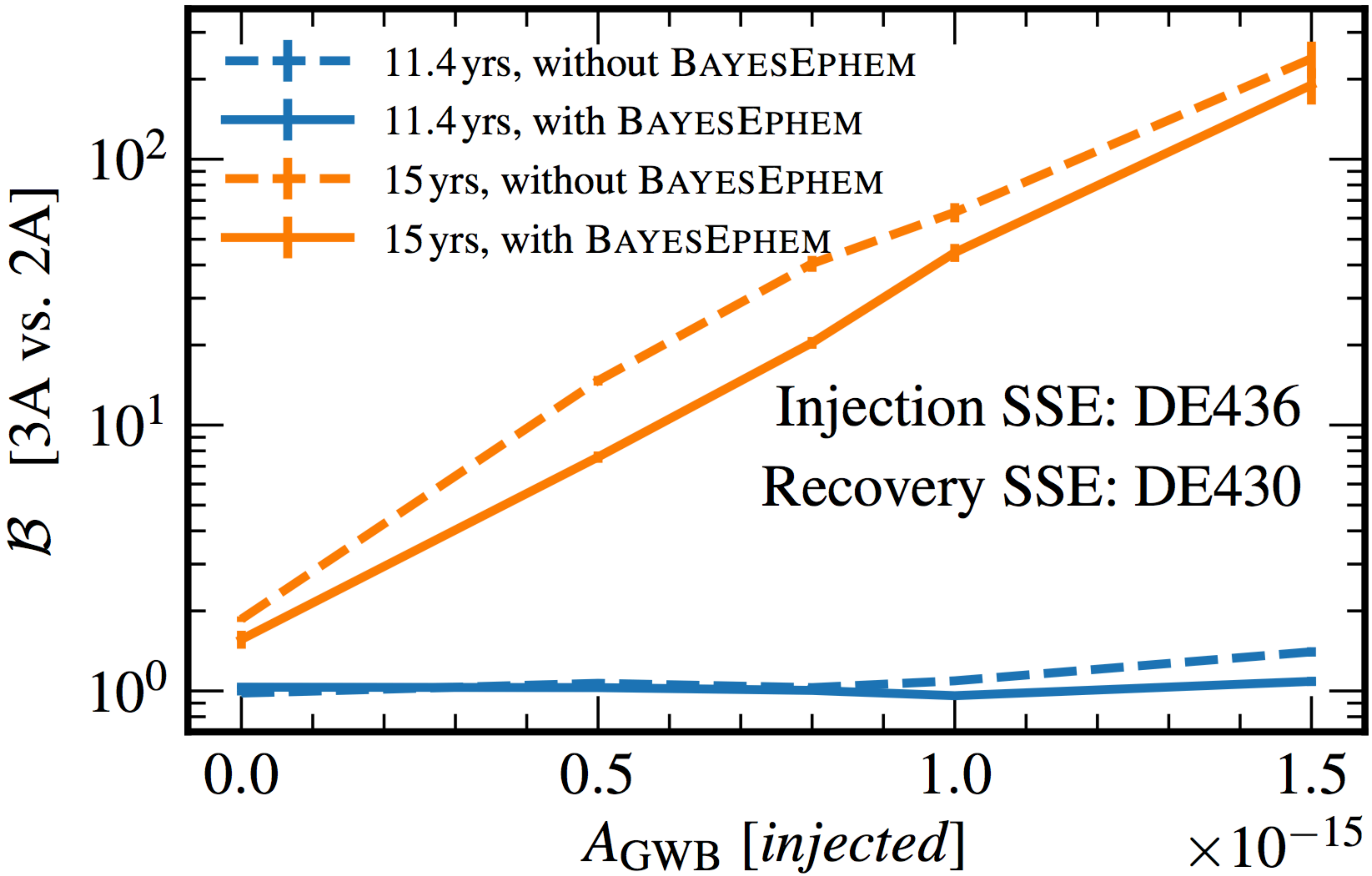
# Physical Ephemeris Uncertainty Model

## 9-year

Uninformative Jupiter orbit priors



# BayesEphem does not hinder detection prospects



# BayesEphem is easy to use

## accessible through "enterprise"
## github.com/nanograv/enterprise

```
# red noise
s = red_noise_block(prior=amp_prior, Tspan=Tspan, components=components)

# common red noise block
s += common_red_noise_block(psd=psd, prior=amp_prior, Tspan=Tspan,
                            components=components, gamma_val=gamma_common,
                            name='gw')

# ephemeris model
if bayesephem:
    s += deterministic_signals.PhysicalEphemerisSignal(use_epoch_toas=True)

# timing model
s += gp_signals.TimingModel()

# adding white-noise, and acting on psr objects
models = []
for p in psrs:
    if 'NANOGrav' in p.flags['pta'] and not wideband:
        s2 = s + white_noise_block(vary=False, inc_ecorr=True, select=select)
        models.append(s2(p))
    else:
        s3 = s + white_noise_block(vary=False, inc_ecorr=False, select=select)
        models.append(s3(p))

# set up PTA
pta = signal_base.PTA(models)
```