Using Least Squares to Specify q in FGNIMEQ

E. C. Howell (Tech-X)

NIMROD Winter Coding Camp

January 16, 2023

Work Supported by US DOE

Outline

- **Specifying q directly in FGNIMEQ is useful**
- Using least squares to fit a q-profile
- Example cases: DIII-D IBS
- Future Work

Safety factor is fundamental quantity of interest

- The ability to set q directly is useful when setting up an equilibrium
  - Theoretical studies often start by specifying q
  - Tweak the q-profile in experimental reconstructions
- Grad-Shafranov equation describes force balance in terms of pressure $P(\psi)$ and toroidal magnetic field $F(\psi) = RB_\varphi$:

$$\Delta^* \psi = -F(\psi)F'(\psi) - \mu_0 R^2 P'\ (\psi)$$

- q depends on F and the flux surface geometry:

$$q(\psi) = \frac{F(\psi)V'(\psi)}{2\pi}\langle R^{-2}\rangle \rightarrow F(\psi) = \frac{2\pi q(\psi)}{V'(\psi)}\frac{1}{\langle R^{-2}\rangle}$$

- Specifying q transform G.S. into integro-differential equation

Picard iterations diverge due to “explicit” treatment of FSA quantities

- Naive approach
  - Start with an initial $F_0(\psi)$
    - E.g., Start from reconstruction or use cylindrical q
  - Solve G-S equation
  - Compute V_n' and $\langle R^{-2} \rangle_n$
  - Compute new $\mathrm{F}_{n+1}(\psi)$ from last V_n' and $\langle R^{-2} \rangle_n$
    - $F_{n+1}(\psi) = \frac{2\pi q}{V_n'} \frac{1}{\langle R^{-2} \rangle_n}$
  - Iterate

- Challenge: Small changes in $F(\psi)$ result in large changes in $V'(\psi)$ and $\langle R^{-2} \rangle$

Outline

- Specifying q directly in FGNIMEQ is useful
- **Using least squares to fit a q-profile**
- Example cases: DIII-D IBS
- Future Work

Approach: Treat q-solve as a reconstruction problem

- Specify desired q at a set of points:

$$q_i^o \equiv q_{desired}(\psi_i)\ \forall\, \psi_i$$

- Define a parametrization of $F(\psi) \equiv F(\psi|p_1, p_2, \dots, p_n)$
- Solve G-S eqn. for a set of parameters $\vec{p}$, and then compute the modeled q

$$q_i^M \equiv q(\psi_i|\vec{p})\ \forall(\psi_i)$$

- Find parameters that minimize the weighted square error between q_i^o and q_i^M

$$g^2 = \sum_i w_i\left(q_i^o - q_i^M\right)^2$$

- Robust algorithms exist to minimize g^2
  - Here we've adapted the V3FIT algorithm to minimize g^2[1]
- **Question:** Is sampling q enough, or do we need addition constraints?

[1] J. D. Hanson et al., Nucl. Fusion 49 (2009) 075031

A quasi-newton is used to minimize the square error [1]

- Netwon's Method:
  - Find roots where $\nabla g^2(\vec{p}) = 0$
  - For $\nabla g^2(\vec{p}) \neq 0$, linearize to find $\overrightarrow{\delta p}$ such that $\nabla g^2\left(\vec{p} + \overrightarrow{\delta p}\right) = 0$:

$$\nabla g^2\left(\vec{p} + \overrightarrow{\delta p}\right) = \nabla g^2(\vec{p}) + \nabla\nabla g^2(\vec{p}) \cdot \overrightarrow{\delta p} = 0$$

- Definitions:

i-th error: $e_i \equiv \sqrt{w_i}\left(q_i^o - q_i^M\right)$

Jacobi matrix: $J_{ij} = -\frac{\partial e_i}{\partial p_j}$

Gradient vector: $-\frac{1}{2}\nabla g^2 = J^T \cdot e$

Hessian Matrix: $H = \frac{1}{2}\nabla\nabla g^2 = J^T \cdot J$

- Quasi-Newton method: $J^T \cdot J \cdot \delta p = J^T \cdot e$

SVD quasi-inverse addresses ill-conditioned Hessian [1]

- SVD of J: $J = U \cdot W \cdot V^T$
  - U and V are unitary
  - W is a NxM whose diagonal are ordered singular values $\mathrm{W_{ii}} = w_i > w_{i+1}$
- k-th quasi-inverse of W:

$$\left(W^{k\sim1}\right)_{ij} \equiv \begin{cases} \delta_{ij} w_i^{-1} \;\; for \quad i \leq k \\ \quad 0 \; otherwise \end{cases}$$

- Truncating W removes small singular values
- SVD inverse of J : $J^{k\sim1} = V \cdot W^{k\sim1} \cdot U^T$
- Solution to Quasi-Newton method: $J^T \cdot J \cdot \delta p = J^T \cdot e$

$$\mathrm{\delta p \approx V \cdot W^{k\sim1} \cdot U^T \cdot e}$$

Basic algorithm

- Python script qsolver_runner.py performs optimization
  - Start with an initial guess for $\vec{p}$
  - Call fgnimeq to solve G-S equation
  - Compute q_i^M and e_i
  - Compute Jacobi matrix (J) using numerical differentiation
    - Independently vary each parameter and compute $-\frac{\partial e_i}{\partial p_j}$ using central difference
  - Compute δp using SVD inverse
  - Search along $\vec{p} + \theta\delta p$ with $\theta \in (0,1]$ for a better fit
  - Iterate until convergence

Practical consideration

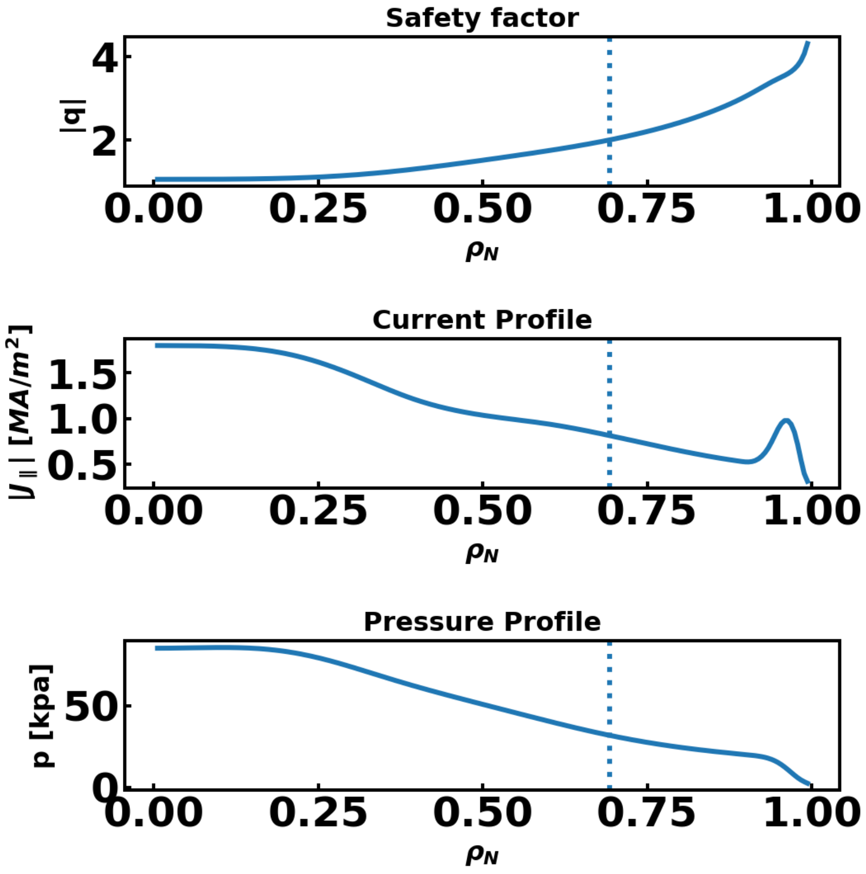
- Use limited equilibria to fit q-profile
  - Negates need for FSA to find seperatrix, and reduces mesh requirements
- Question: Can we then use to adjust the core q-profile in diverted eq?
  - q_{95} depends on the flux surface shape and the enclosed current
  - Thus if we match q for $\psi > \psi_{cutoff}$, then it may work
- Due to large number of fgnimeq calls it helps to reduce gsh_tol
  - But majority of time spent factoring matrix
  - When porting to abstract infrastructure, may be useful to thing treat as a library
- I parametrize $F(\psi)$ using cubic splines
  - Parameters are knot amplitudes
  - Experimentation suggest 20-30 knots, which are constrained by q at 150 ψ_i

Outline

- Specifying q directly in FGNIMEQ is useful
- Using least squares to fit a q-profile
- **Example cases: DIII-D IBS**
- Future Work

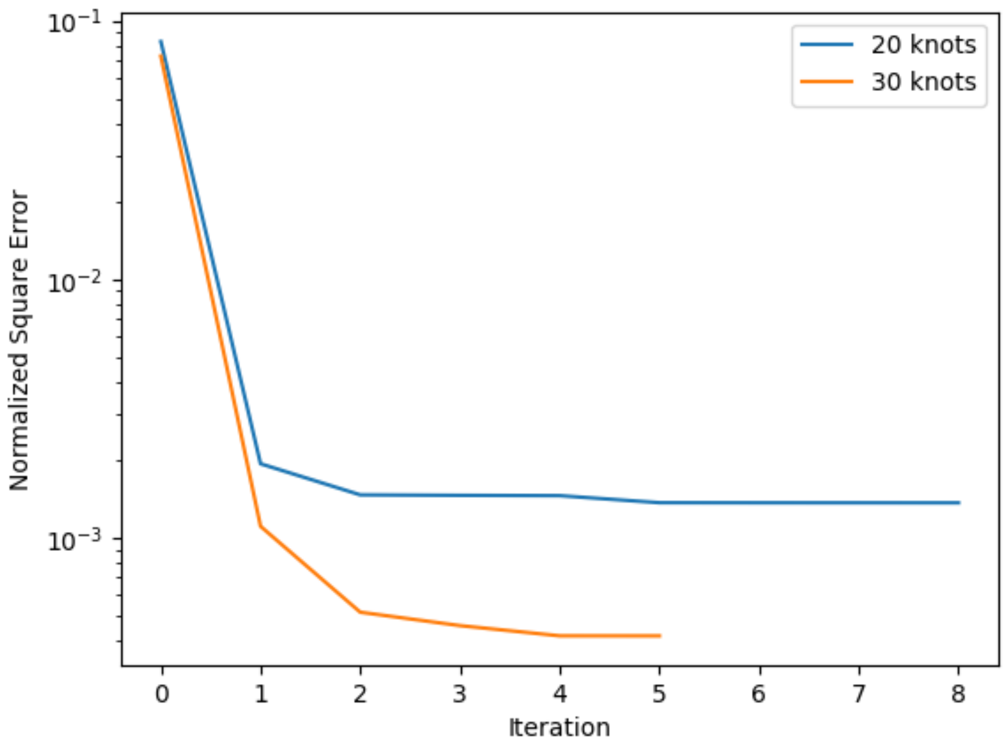
Test case, try to decrease q_0 in reconstructions DIII-D IBS discharge

- Original reconstructions fixed $q_0 = 1.04$
- Large region of low shear magnetic shear
- Simulations show core mode in low shear region
- Will reducing q_0 stabilize core modes
- Want to change q_0 with minimal changes elsewhere
- Test case: decrease $q_0 = 0.95$ while keeping q fixed for $\psi > 0.5$
- Despite small change in q_0 profile, naive approaches to tweak q fail.

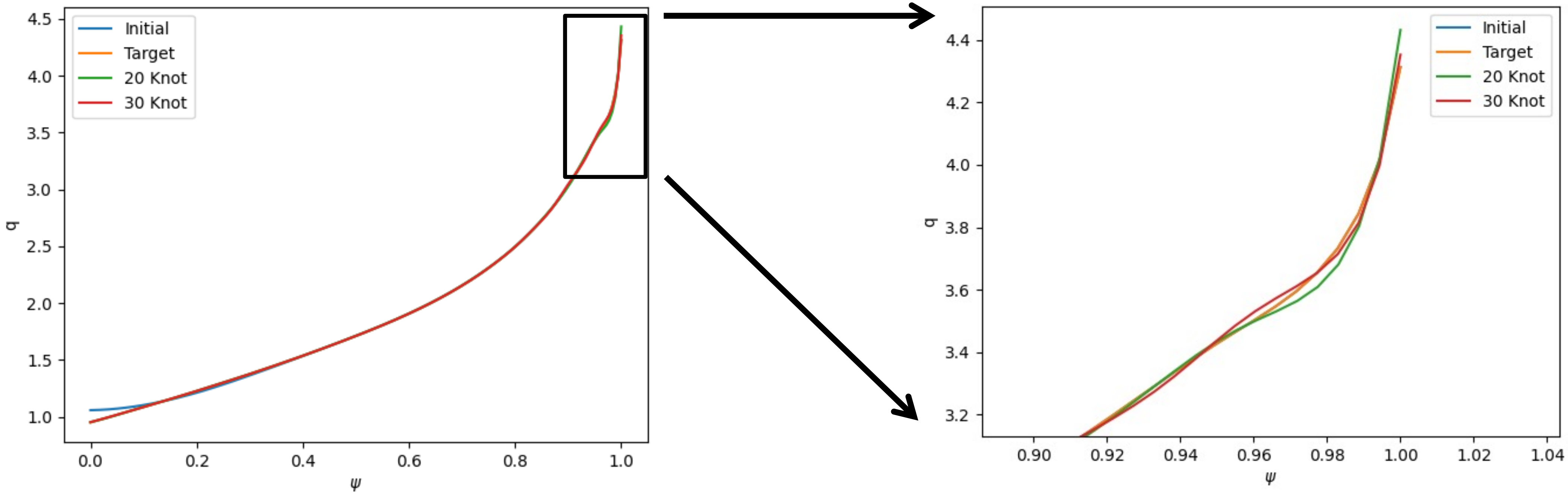


Algorithm rapidly converges when a sufficient number of parameters are used

- Error of 0.1-0.2% achieved after first step
- 20knots achieves error of 0.136%
- 30knots 0.05% error after 2 steps
  - Final error of 0.04%
- Using equally spaced knots in psi
  - Large number of knots needed to resolve pedestal (next slide)

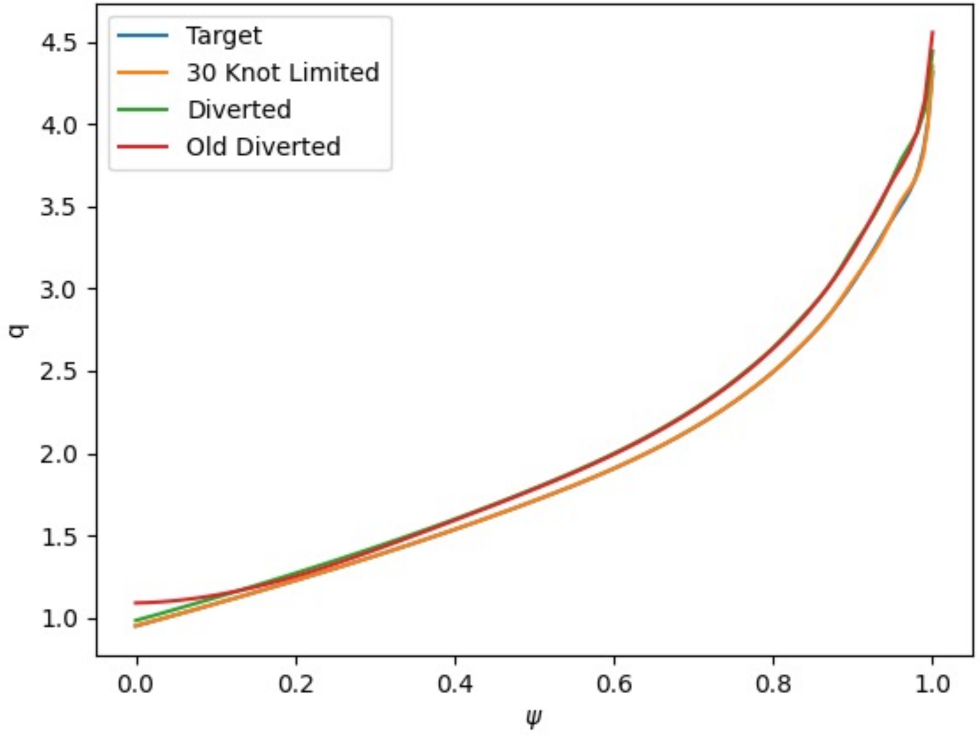


Largest differences errors q are in the pedestal

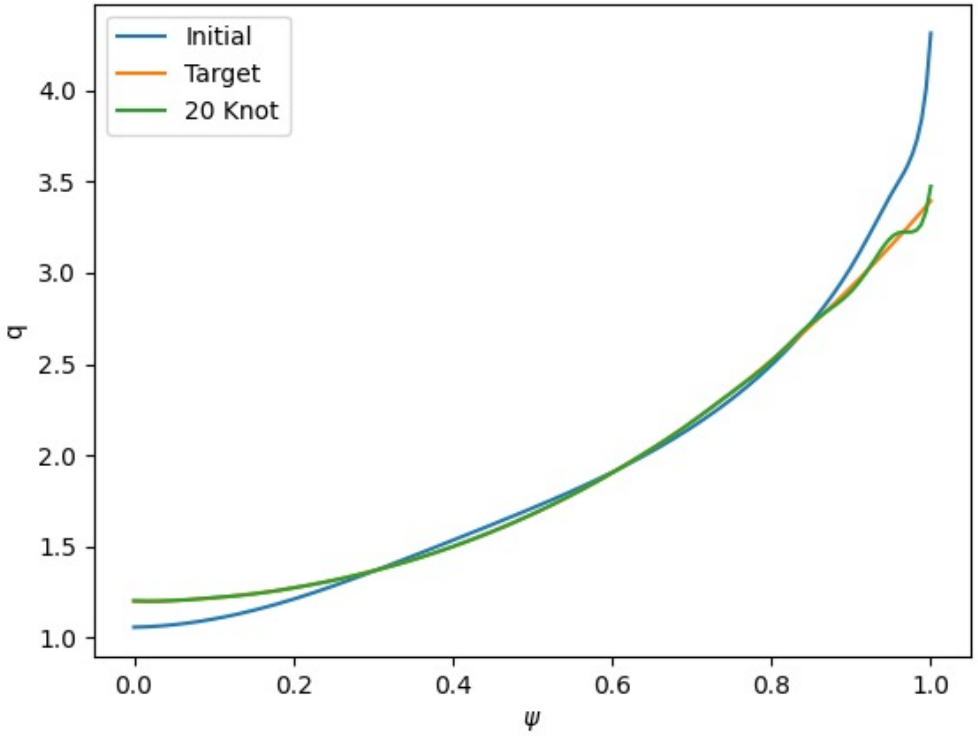


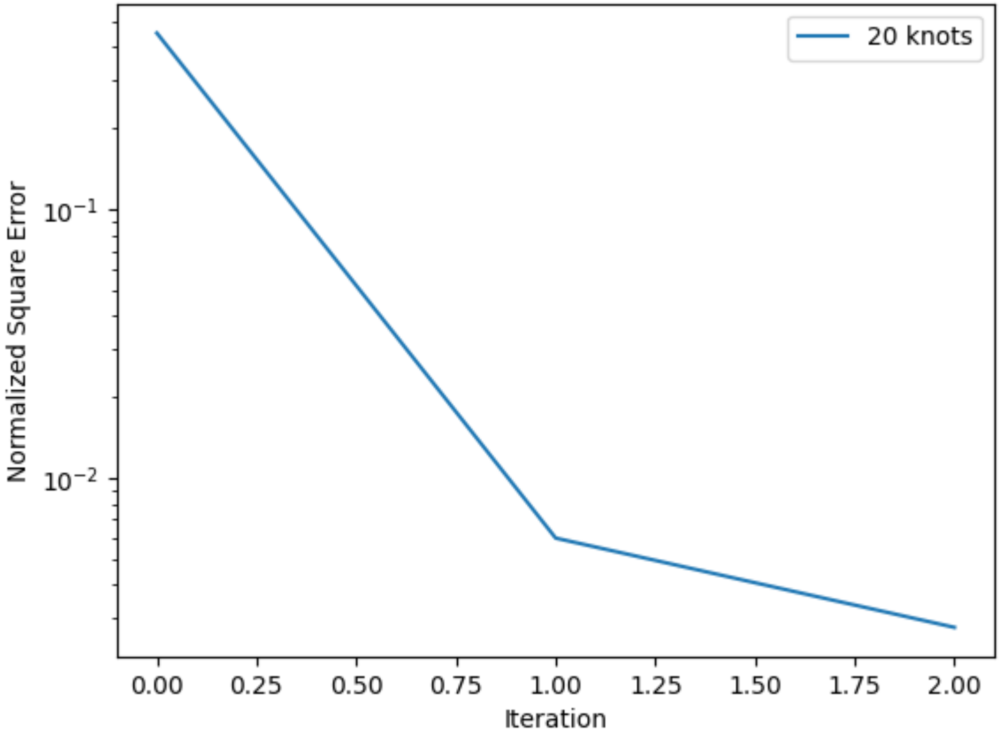
Using fitted F(psi) with diverted case results in a shift in q

- Shift due to small scrape-off layer currents in diverted cases
- O-point location
  - Old Diverted: (1.7685m, -0.0075m)
  - New: (1.7685m, -0.0058m)
- X-point location
  - Old Diverted : (1.281m, -1.165m)
  - New: (1.282m, -1.164m)
- Plasma current
  - Old Diverted: 1.50884 MA
  - New: 1.50775 MA
- Enclosed poloidal flux
  - Old Diverted : 0.30036
  - New: 0.30053



A more challenging limited case





- Specifiy q using a two-power profile
  - $q = q_0(1 + \psi^{\alpha})^{\beta}$ with $q_0 = 1.2,\ \alpha = 2,\ \beta = 1.5$
- Test uses 20 knots, error around psi=0.95 is due to pedestal pressure gradients

Script to compute fit F for q-profile for FQNIMEQ has been developed

- Script calls FGNIMEQ to solve G-S equation
- Python math libraries enable rapid prototyping and development
  - Multiple optimization, spline, and linear algebra routines
  - Poloidal FSA integration of q is more robust in python
- Would benefit from converting FGNIMEQ functionality into callable library
    - Saving mesh initialization, matrix factors, and last solution would accelerate computation
- Method works well for limited cases
  - Current choice of uniformly spaced knots is suboptimal, easy fix
- Using limited cases to fit F for diverted EQ shows promise
  - Can we use algorithm to automatically tune scrape off layer profiles to minimize currents?

- Scripts available on NIMPY repo