# Skyrme functional fitting at 13% of the computational expense

Jorge Moré, Jason Sarich, and **Stefan Wild**

Mathematics and

Computer Science Division

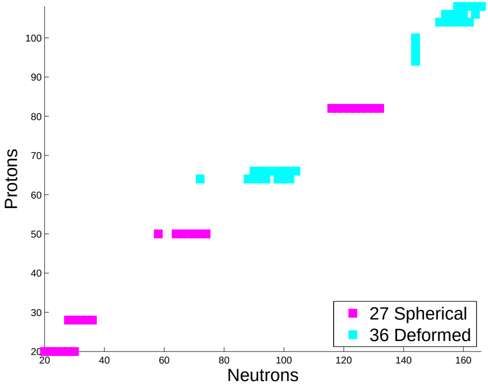
Argonne National Laboratory

Argonne NATIONAL LABORATORY

3rd LACM-EFES-JUSTIPEN Workshop
February 25, 2009

# Our Experiences with Density Functional Fitting

## Disclaimer: I am not a nuclear physicist!

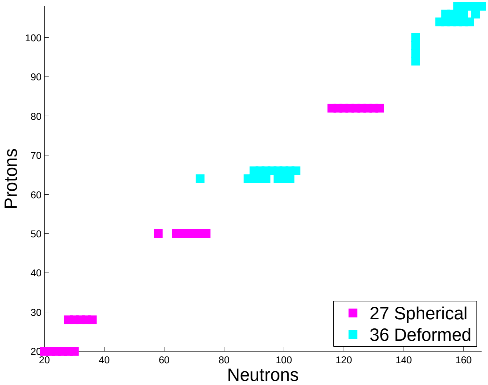


# Our Experiences with Density Functional Fitting

## Disclaimer: I am not a nuclear physicist!

- But others are (Markus Kortelainen, Thomas Lesinski, Witek Nazarewicz, Nicolas Schunck, Mario Stoitsov, . . . )

- Optimization approach is general; other functional forms possible and expected



# Our Optimization/NLS/NLR Problem

$2$ measurement types: binding energy (E), radius (R)
$\mathcal{N}_E =$ 63 nuclei (27 spherical, 36 deformed)
$\mathcal{N}_R =$ the spherical nuclei from $\mathcal{N}_E$

$d_{t,i}$ experimental data ($t \in \{E, R\}$) for $i$th nucleus

$s_{t,i}(\theta)$ HFBTHO simulation of type $t$ for $i$th nucleus

$\theta$ $9$ simulation parameters from intervals of interest

$$f(\theta) = \frac{1}{\sigma_E^2} \sum_{i=1}^{63} \left(d_{E,i} - s_{E,i}(\theta)\right)^2 + \frac{1}{\sigma_R^2} \sum_{i=1}^{27} \left(d_{R,i} - s_{R,i}(\theta)\right)^2 ,$$

$\sigma_t$ weights: $\sigma_E =$2 [MeV], $\sigma_R = .02$ [fm]
Chosen to balance goals of energy and radius fitting

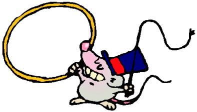
# Why is This a Challenging Problem?

$$\min_{\theta \in \Theta \subseteq \mathbb{R}^n} \left\{ f(\theta) = \sum_{t=1}^{T} \frac{1}{\sigma_t^2} \sum_{i \in \mathcal{N}_t} \left( d_{t,i} - s_{t,i}(\theta) \right)^2 \right\}$$

Small Scale: Few variables ($n = 9$) and equations ($|\mathcal{N}_E| + |\mathcal{N}_R| = 90$)

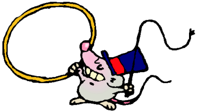
Smooth: If $s$ is smooth then $f$ is

Unconstrained: No nonlinear constraints on $\theta$



# Why is This a Challenging Problem?

$$\min_{\theta \in \Theta \subseteq \mathbb{R}^n} \left\{ f(\theta) = \sum_{t=1}^{T} \frac{1}{\sigma_t^2} \sum_{i \in \mathcal{N}_t} \left( d_{t,i} - s_{t,i}(\theta) \right)^2 \right\}$$



Small Scale: Few variables ($n = 9$) and equations ($|\mathcal{N}_E| + |\mathcal{N}_R| = 90$)

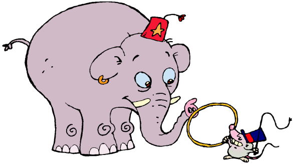
Smooth: If $s$ is smooth then $f$ is

Unconstrained: No nonlinear constraints on $\theta$

Computationally Expensive: Evaluating all $s_{t,i}(\theta)$ takes 9 CPU-hours

Derivative-Free: Don't have access to $\frac{\partial}{\partial\theta} s_{t,i}(\theta)$

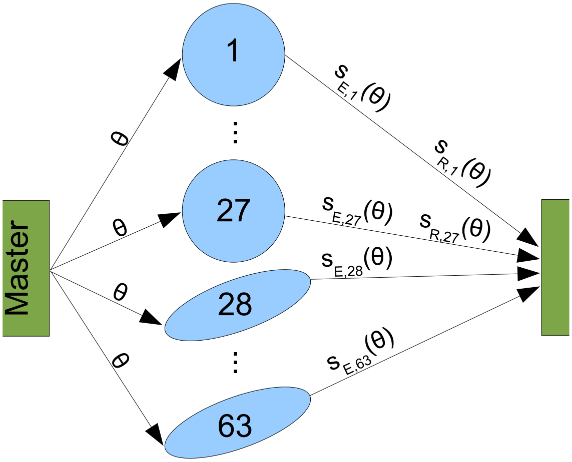
Without both CE and DF this would be "easy"



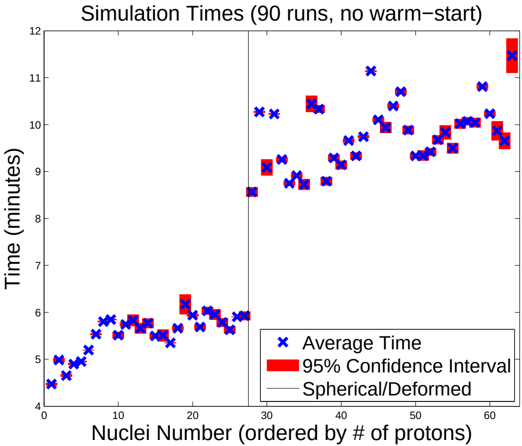
# We Cannot Afford 9 Hours per Evaluation!

## Move to a cluster

- 64 processors for 63 nuclei
- Simulate each nucleus independently
- Will be bound by the longest HFBTHO simulation time

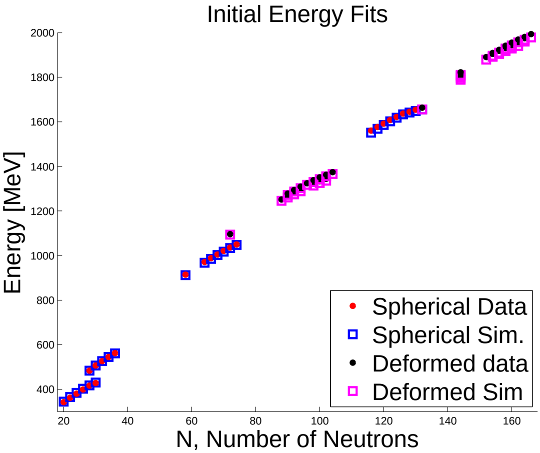


# $f(\theta)$ Now Requires 12 Minutes of Wall Clock




- 100 evaluations $\approx$ 20 hours
- Can further speed up by warm-starting fixed point iteration
- Can use fewer processors by exploiting load balancing
- More realistic/complex simulators should make use of parallelism

# Improve Fit in the Region of an Initial Point, $\theta^{(0)}$

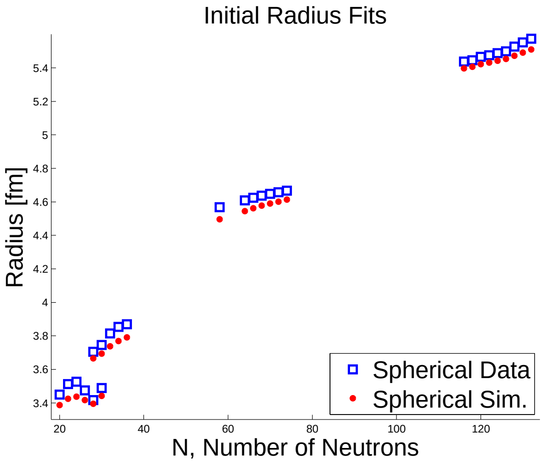



$\theta^{(0)}$ comes from experience/ fitting fewer nuclei

**By the eyeball norm:**

- Good (relative) energy fit
  Avg. error: 7.42 MeV
- Not as good radius fit
  Avg. error: .06 fm

# Improve Fit in the Region of an Initial Point, $\theta^{(0)}$




$\theta^{(0)}$ comes from experience/ fitting fewer nuclei

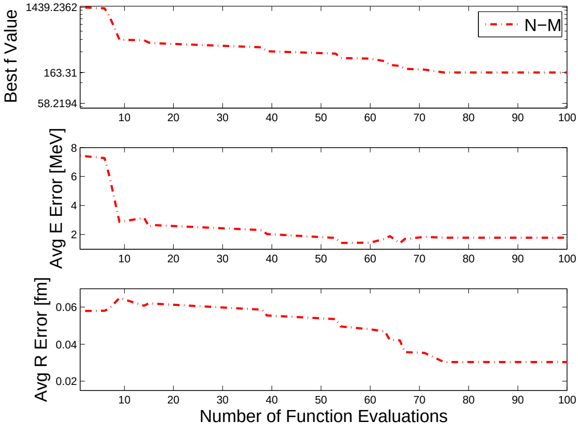
**By the eyeball norm:**

- Good (relative) energy fit
  Avg. error: 7.42 MeV
- Not as good radius fit
  Avg. error: .06 fm

# Optimization by the Nelder-Mead Method [1965]

## In 100 evaluations:

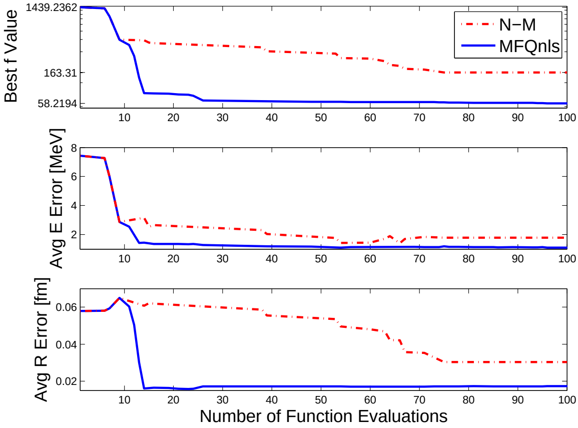
- Reduce objective from $f = 1439$ to $f = 163$
- Reduce (avg) energy error from 7.42 MeV to 1.78 MeV
- Reduce (avg) radius error from .06 fm to .03 fm



# Optimization by MFQnls [Moré & Wild 2009]

## In 100 evaluations:

- Reduce objective from $f = 1439$ to $f = 58$
- Reduce (avg) energy error from 7.42 MeV to 1.10 MeV
- Reduce (avg) radius error from .06 fm to .017 fm

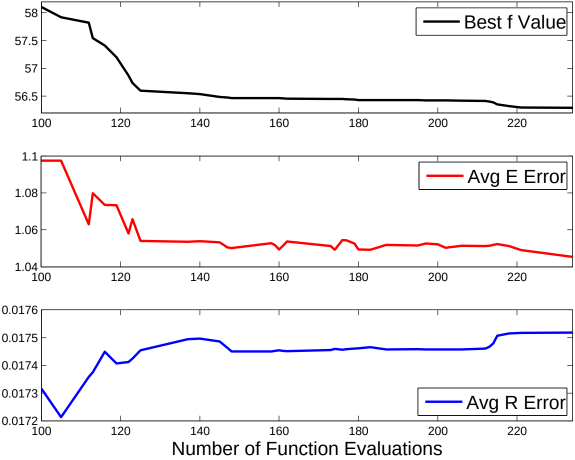


# More Optimization by MFQnls

## After 100 evaluations:

- Reduce objective from $f = 1439$ to $f = 56.3$
- Reduce (avg) energy error from 7.42 MeV to 1.045 MeV
- Reduce (avg) radius error from .06 fm to .0175 fm

Stopped because of computational budget



# N-M vs. MFQnls Summary

| Solver | Evals | $f$ | E Error [MeV] | R Error [fm] |
|---|---|---|---|---|
| $\theta^{(0)}$ | 1 | 1439 | 7.426 | .0579 |
| N-M | 100 | 163.3 | 1.779 | .0303 |
| MFQnls | 13 | 137.5 | 1.425 | .0301 |
| | 100 | 58.22 | 1.097 | .0173 |
| | 234 | 56.29 | 1.045 | .0175 |

## Conclusion:

In less than 1 CPU-week (2.6 hours wall clock), MFQnls is able to get a better solution than the Nelder-Mead solution after 7.6 CPU-weeks (20 hours wall clock).

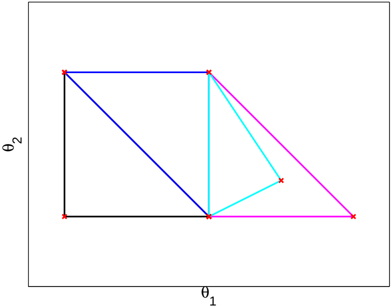
# Why is MFQnls Doing So Much Better Here?

Both methods will (eventually) find a *local* solution

**Each evaluation of $f$ is expensive (*valuable*)**

N-M:

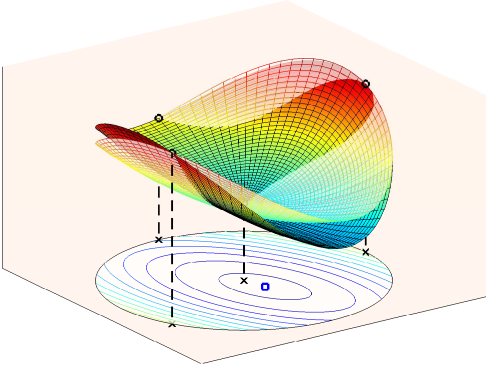
1. Only remembers the last $n+1$ evaluations
2. Neglects the *magnitudes* of the function values (order only)
3. Doesn't take into account the special (LS) problem structure




Remember: N-M is still a powerful method in other settings

# A Sketch of MFQnls

Idea: Build a quadratic model of the objective $f$ in a neighborhood of the current $\theta = \theta^{(k)}$



$$q(\theta^{(k)} + s) = c + g^T s + \frac{1}{2} s^T H s:$$

Find the $\frac{(n+1)(n+2)}{2}$ model coefficients $c, g, H = H^T$ so:

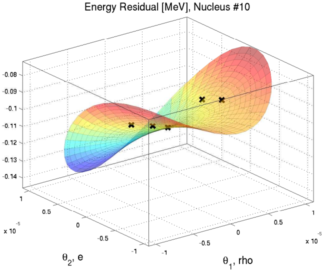
$$q(\hat{\theta}^{(i)}) = f(\hat{\theta}^{(i)}) \quad \forall \hat{\theta}^{(i)} \in \mathcal{Y}$$

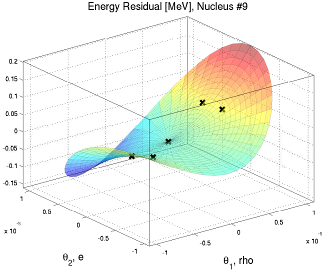
$\mathcal{Y}$ =interpolation set based on the $\theta$ at which we've previously evaluated $f$

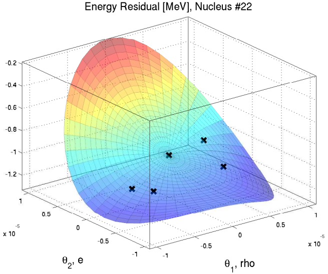
# Taking Advantage of the Structure of $f$

Build an interpolation model for each residual $\frac{d_i - s_{t,i}(\theta)}{\sigma_i}$.

Allows us to obtain an approximate Jacobian, $J \approx \begin{bmatrix} g_1^T \\ \vdots \\ g_m^T \end{bmatrix}$



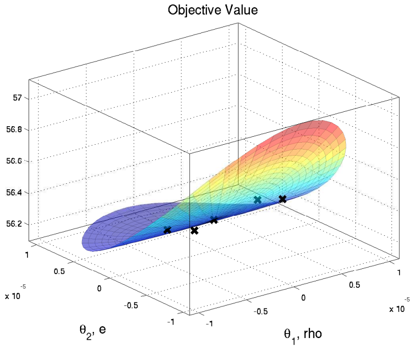



# Taking Advantage of the Structure of $f$

Build an interpolation model for each residual $\frac{d_i - s_{t,i}(\theta)}{\sigma_i}$.
Allows us to obtain an approximate

$$\text{Jacobian, } J \approx \begin{bmatrix} g_1^T \\ \vdots \\ g_m^T \end{bmatrix}$$



# Sensitivity

Assuming the errors $\frac{d_{t,i}-s_{t,i}(\theta)}{\sigma_t}$ are

- iid Normal with mean zero (← big assumption?)

Use Jacobian approximation at $\hat{\theta}$ to estimate the covariance matrix

$$Cov(\hat{\theta}) \approx 2\left(\nabla^2 f(\hat{\theta})\right)^{-1} \approx \left(\hat{J}^T \hat{J}\right)^{-1}$$

# Sensitivity

Assuming the errors $\frac{d_{t,i} - s_{t,i}(\theta)}{\sigma_t}$ are

- iid Normal with mean zero (← big assumption?)

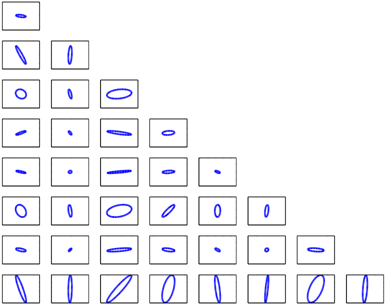
Use Jacobian approximation at $\hat{\theta}$ to estimate the covariance matrix

$$Cov(\hat{\theta}) \approx 2\left(\nabla^2 f(\hat{\theta})\right)^{-1} \approx \left(\hat{J}^T \hat{J}\right)^{-1}$$

## Correlations

$$Cor(\theta_i, \theta_j) = \frac{Cov(\theta)_{i,j}}{\sqrt{Cov(\theta)_{i,i} Cov(\theta)_{j,j}}}$$

Reduce the problem by accounting for strong correlations.



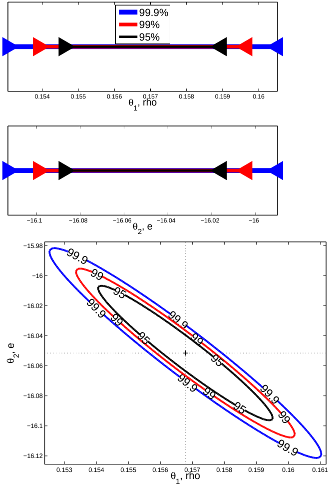
# Sensitivity of the Parameters

$100(1-\alpha)\%$ confidence interval for a single parameter:

$$\Pr\Big(\theta_j \in [\hat{\theta}_j - \Delta, \hat{\theta}_j + \Delta]\Big) \approx 1-\alpha$$

$100(1-\alpha)\%$ confidence regions for subsets of parameters:

$$\Pr\Big((\theta_1, \theta_2) \in \mathcal{E}(\hat{\theta})\Big) \approx 1-\alpha$$




Caution: These are *approximations* in the weakest sense!

# Some Concluding Remarks

**Good derivative-free optimization technology exists! But:**

- Need big machines and a "distributed mindset" to make computation feasible for realistic functionals
- Scaling and problem formulation are critical
- Exploit structure to the fullest extent possible

# Some Concluding Remarks

## Good derivative-free optimization technology exists! But:

- Need big machines and a "distributed mindset" to make computation feasible for realistic functionals
- Scaling and problem formulation are critical
- Exploit structure to the fullest extent possible

## Making the most of your computational budget with MFQnls

- By exploiting structure can get a good solution at 13% of the computational cost
- Can get *approximate* sensitivity at little/ no additional cost
- $f$ appears to be quite smooth, global analysis in progress

# Collaborators and References

Argonne National Laboratory: Jorge Moré, Jason Sarich, Stefan Wild

Oak Ridge National Laboratory: Markus Kortelainen, Thomas Lesinski, Witek Nazarewicz, Nicolas Schunck, Mario Stoitsov, and others

Office of Science U.S. Department of Energy

UNEDF

SciDAC Scientific Discovery through Advanced Computing

- Some papers and links available at `http://mcs.anl.gov/~wild`

DFO Conn, Scheinberg, and Vicente. Introduction to Derivative-Free Optimization. SIAM, 2008.

Stats Seber and Wild (no relation!). Nonlinear Regression. Wiley, 1989.

Thank You!