

**Application of NESOPT:  
New Surface Optimizer NESVD  
To QOS GB4**

Prashant Valanju, Lee Berry,  
Dennis Strickler, and William Miner Jr.

QOS Project Meeting Presentation

Feb. 28, 2001



## **Input [xc(nvar)] & Output [fvec(mfun)] vectors for NES\_OPT function called by LMOPT at each iteration**

- The input xc(nvar) contains cr(m,n) and cz(m,n) to be varied
- NESVD can use many cr, cz coeffs, but only a subset is varied
  - Either chose all cr, cz > min value (in meters)
  - Or set which ones you want to vary in nescon file

**The output fvec(mfun) contains nuvh Berr and nvar xcshifts**

- $Berr(nu,nv) = [B \cdot n(\text{coils}) - B \cdot n(\text{Plas}) - B \cdot n(\text{ext})] / |B_{\text{total}}|$
- $Xc\_shift(nvar) = xcwght \cdot [xc(\text{current}) - xc(\text{original})] / xcmax$
- **Optional: Limit max xc shift to xcmax by**  
 $Xc\_shift(nvar) = xcwght \cdot \text{Sinh}[\{(xc(\text{current}) - xc(\text{original}))\} / xcmax]$

# NesOpt Control File (NESCON) Contains

- A Namelist of optimizer controls followed by
- A table of r, z coeffs to vary (out of all NESVD cr, cz)

```
&NESCON
IWRT      = 1,           !Num of steps after which to write answers
TOL       = 1.0E-6,     !LM tolerance
EPSFCN    = 1.0E-4,     !LM control
NITER_MAX = 100,        !Maximum number of LM iterations
MAX_PROCESSORS = 1,     !Maximum number of processors to use
XCWGT     = 1.0,        !Relative weight for control on cr, cz variation
XCMA      = 0.01,       !Maximum change allowed in cr, cz (m)
ALM       = 0.0,        !Exponent weight m^alm for cr, cz control
ALN       = 0.0,        !Exponent weight n^aln for cr, cz control
RCMIN     = 0.01,       !Minimum size of cr to be varied (m) (set <0 to use table below)
ZCMIN     = 0.01,       !Minimum size of cz to be varied (m) (set <0 to use table below)
/
-- Follow this by a list of m, n, rmn or zmn to vary,
-- For Example: r 1 2 ini_value or z 2 3 (DO NOT REMOVE THESE 2 COMMENT LINES)
r 1 2 2.03e-1
z 2 3 3.27e-1
```

## Screen dump & Output File \*.opt Contain

- Starting nesinp file (same as nescoil input file)
- Initial nescon file (so this optimizer run can be reproduced later)
- Initial state info on cr, cz coeffs (starting surface), followed by
  - For each iwrt'th loop:
    - Current values of cr, cz, and shift from previous and from original
    - Current values of Berr, Jmax etc.
- Info at end of LM loop : Lowest RMS Berr found at iter number n
- The info at every iteration step is enough for post-processing into:
  - (Gnu)Plots of surface sections as optimizer makes them
  - NESVD run for more detailed svd control
  - Cutting discrete coils to check error evolution

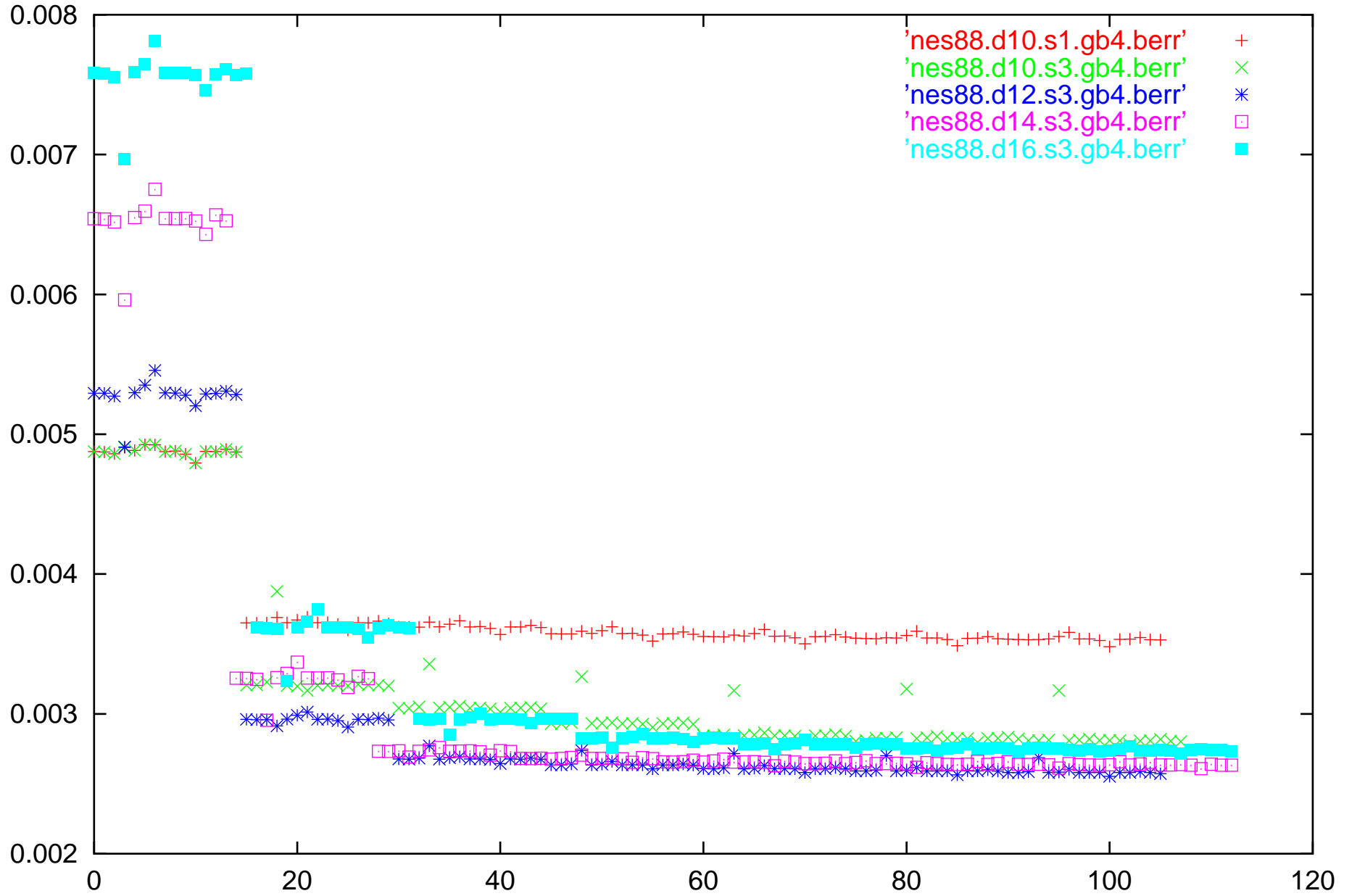
## Summary of First NESOPT Runs (GB4)

- NESOPT is now running on k.nersc, linux, and mcurie
- Currently using single processor
- Working on multiprocessor run

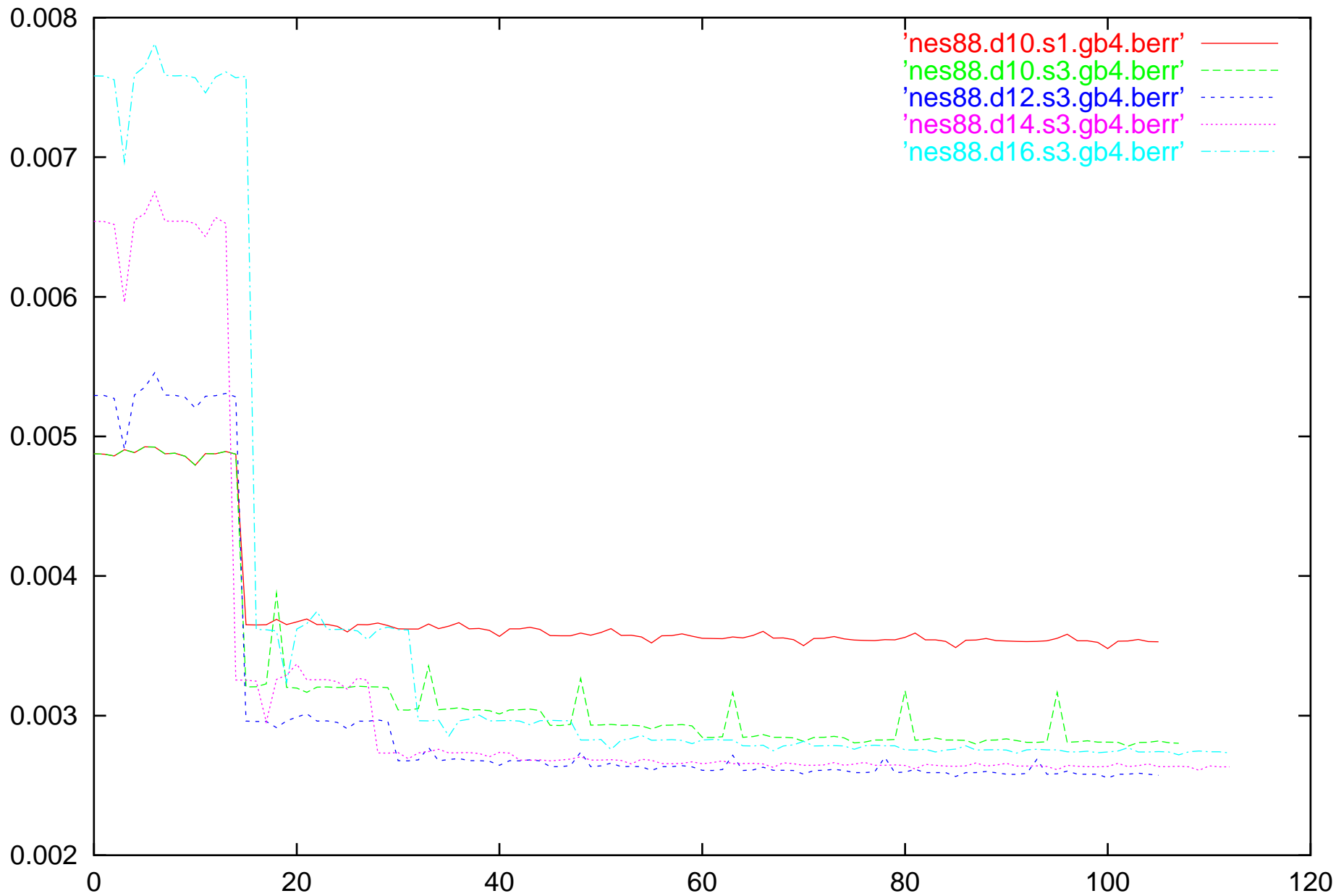
Using  $mf,nf=(5,5)$  and  $(8,8)$ ,  $nu,nv=(64,64)$ ,  $d=10,12,14,16$  cm,  $s=1$  and  $3$  cm

| Case                      | Ini Berr | Min Berr | @   | Ini Jmax | Jmax |
|---------------------------|----------|----------|-----|----------|------|
| nes55.d10.gb4.opt (no s): | 9.98E-2  | 4.50E-2  | 100 | 1.02     | 1.31 |
| nes88.d10.s1.gb4.opt:     | 4.88E-3  | 3.48E-3  | 100 | 1.54     | 1.37 |
| nes88.d10.s3.gb4.opt:     | 4.88E-3  | 2.78E-3  | 102 | 1.54     | 1.29 |
| nes88.d12.s1.gb4.opt:     | 5.29E-3  | 2.55E-3  | 100 | 2.15     | 1.71 |
| nes88.d14.s3.gb4.opt:     | 6.54E-3  | 2.60E-3  | 109 | 4.33     | 2.30 |
| nes88.d16.s3.gb4.opt:     | 7.58E-3  | 2.71E-3  | 107 | 9.19     | 3.63 |

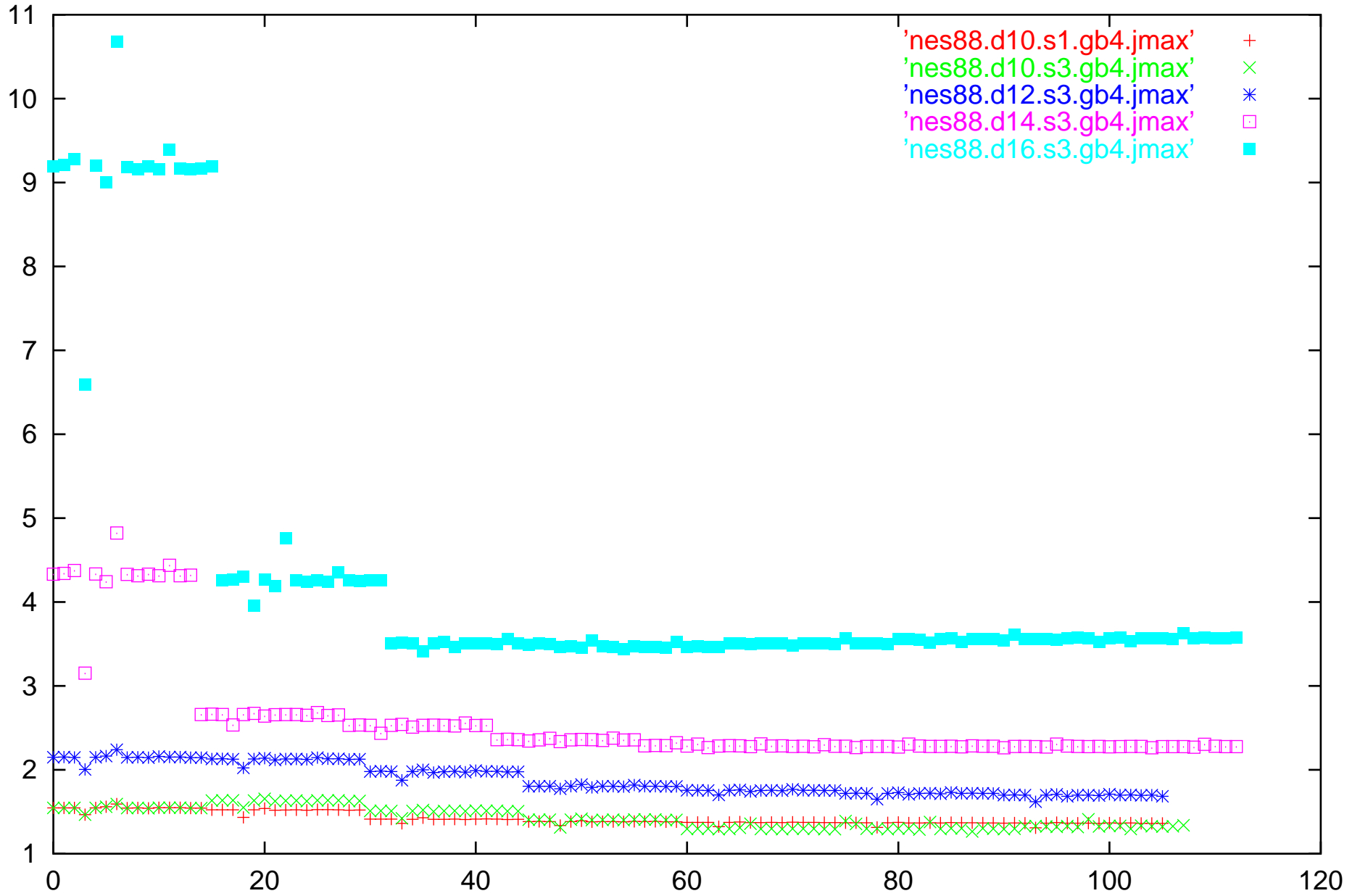
BnormErr / |Btotal| vs Nesopt Iter Number for GB4'



BnormErr/Btotal vs Nesopt Iter Num for GB4



Jamx vs Nesopt Iter num for GB4



Jmax vs Nesopt Iter Num for GB4

