BBA Measurement, Analysis & Application Development

J. Beebe-Wang
J. Ziegler

APEX Workshop
11/20/2012

Outline

Measurements and results:

run 2010, 2011, 2012

Analysis:

BBA model sensitivity, Measurement repeatability, Accuracy limitation

Documentation:

BBA Database; BBA elog; run-time log files, paper & tech note

Application Development:

pre 2012; 2012; post 2012

APEX plan for run 2013

Why is BBA Measurement Difficult in RHIC

1. Limited dipole correctors. In each 3-corrector bump there are 6-7 quads 2 DXs, 2 D0s;
2. Can not decouple the effects of Q1, Q2, Q3 and they are equally strong;
3. Unknown magnet alignment errors;
4. Unknown beam positions and angles at the first corrector (beam angle at triplet);
5. Very large ring. When all the BPMs in all arcs are used, some BPMs contribute to noise more than to signal;
6. Signal/Noise ratio is low.

BBA Limitations on RHIC by Design

1. BBA requires a BPM-quad pair located near each other and independent control of the quads;
2. only quads Q1 - Q9 have the possibility of independent control;
3. All Q1 and Q3 quads have associated BPMs, but Q2 quads don't;
4. Q1, Q3, Q4, Q7, and Q8 quads are associated with dual-plane BPMs;
5. Q5, Q6, and Q9 have only one plane available;
6. All of Q1, Q2, and Q3 quadrupoles allow for independent power supply control;
7. None of Q4, Q5, or Q6 have independent control;
8. However, it is possible to perform BBA on the associate BPMs using trim quads TQ4, TQ5, and TQ6 which are independently controlled;
9. Independent control of Q7, Q8, and Q9 quads varies by IR and ring.

BBA-able Quadurpole-BPM Pairs (Blue)

(Only those quads with independent control and a nearby BPM plane can be used for BBA)

	IR 2		IR 4		IR 6		IR 8		IR 10		IR 12	
	Sector 1	Sector 2	Sector 3	Sector 4	Sector 5	Sector 6	Sector 7	Sector 8	Sector 9	Sector 10	Sector 11	Sector 12
Q1	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV
Q2	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC
Q3	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV
Q4	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV
TQ4	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV	IC HV
Q5	H	V	V	H	H	V	V	H	H	V	V	H
TQ5	IC H	IC V	IC V	IC H	IC H	IC V	IC V	IC H	IC H	IC V	IC V	IC H
Q6	V	H	H	V	V	H	H	V	V	H	H	V
TQ6	IC V	IC H	IC H	IC V	IC V	IC H	IC H	IC V	IC V	IC H	IC H	IC V
Q7	HV	HV	IC HV	IC HV	HV	HV	HV	HV	IC HV	IC HV	HV	HV
Q8	HV	IC HV	IC HV	HV	HV	IC HV	IC HV	HV	IC HV	IC HV	IC HV	HV
Q9	IC H	V	V	IC H	IC H	V	V	IC H	IC H	IC V	V	IC H

IC = Independent Control of power supply is available
H, V = Horizontal or Vertical BPM plane is available

	H+V
Q1	24
Q2	0
Q3	24
Q4	0
TQ4	24
Q5	0
TQ5	12
Q6	0
TQ6	12
Q7	8
Q8	14
Q9	7
Total	125

BBA-able Quadurpole-BPM Pairs (Yellow)

(Only those quads with independent control and a nearby BPM plane can be used for BBA)

	IR 2		IR 4		IR 6		IR 8		IR 10		IR 12	
	Sector 1	Sector 2	Sector 3	Sector 4	Sector 5	Sector 6	Sector 7	Sector 8	Sector 9	Sector 10	Sector 11	Sector 12
Q1	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC
	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV
Q2	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC
Q3	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC
	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV
Q4												
	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV
TQ4	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC
	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV
Q5												
	V	H	V	V	H	H	V	V	H	V	V	H
TQ5	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC
	V	H	V	V	H	H	V	V	H	V	V	H
Q6												
	H	V	H	H	V	V	H	H	V	H	H	V
TQ6	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC	IC
	H	V	H	H	V	V	H	H	V	H	H	V
Q7			IC	IC					IC	IC		
	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV
Q8	IC			IC	IC			IC	IC	IC		IC
	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV	HV
Q9		IC	IC			IC	IC		IC	IC	IC	
	V	H	V	V	H	H	V	V	H	V	V	H

	H+V
Q1	24
Q2	0
Q3	24
Q4	0
TQ4	24
Q5	0
TQ5	12
Q6	0
TQ6	12
Q7	8
Q8	14
Q9	7
Total	125

IC = Independent Control of power supply is available
H, V = Horizontal or Vertical BPM plane is available

RHIC BBA Activity 2010

RHIC 2010 Activities

Measurements:

16 in Blue; 25 in Yellow

Results:

1) Half of the results with re-measurements were inconsistent;

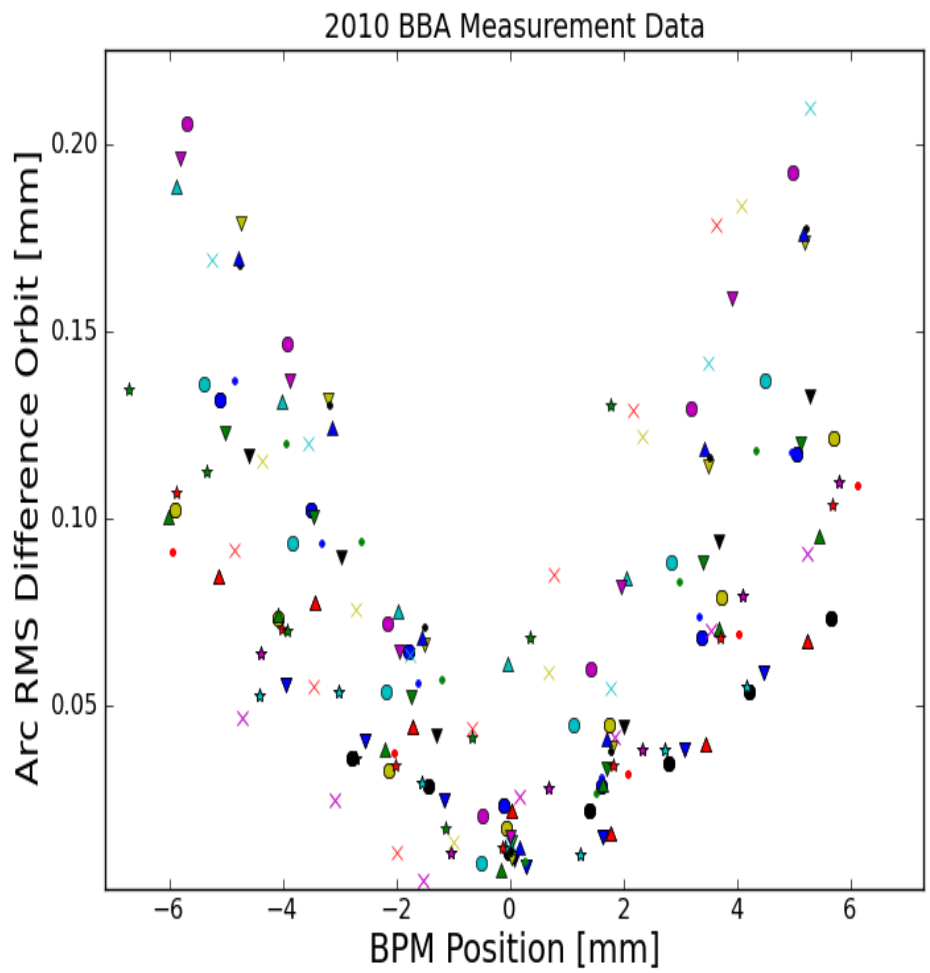
2) No error analysis;

3) Average error bar: 178μm

(reanalyzed in 2012)

Corrections:

None after offset reversal fix



RHIC BBA Activity 2011

RHIC 2011 Activities

Measurements:

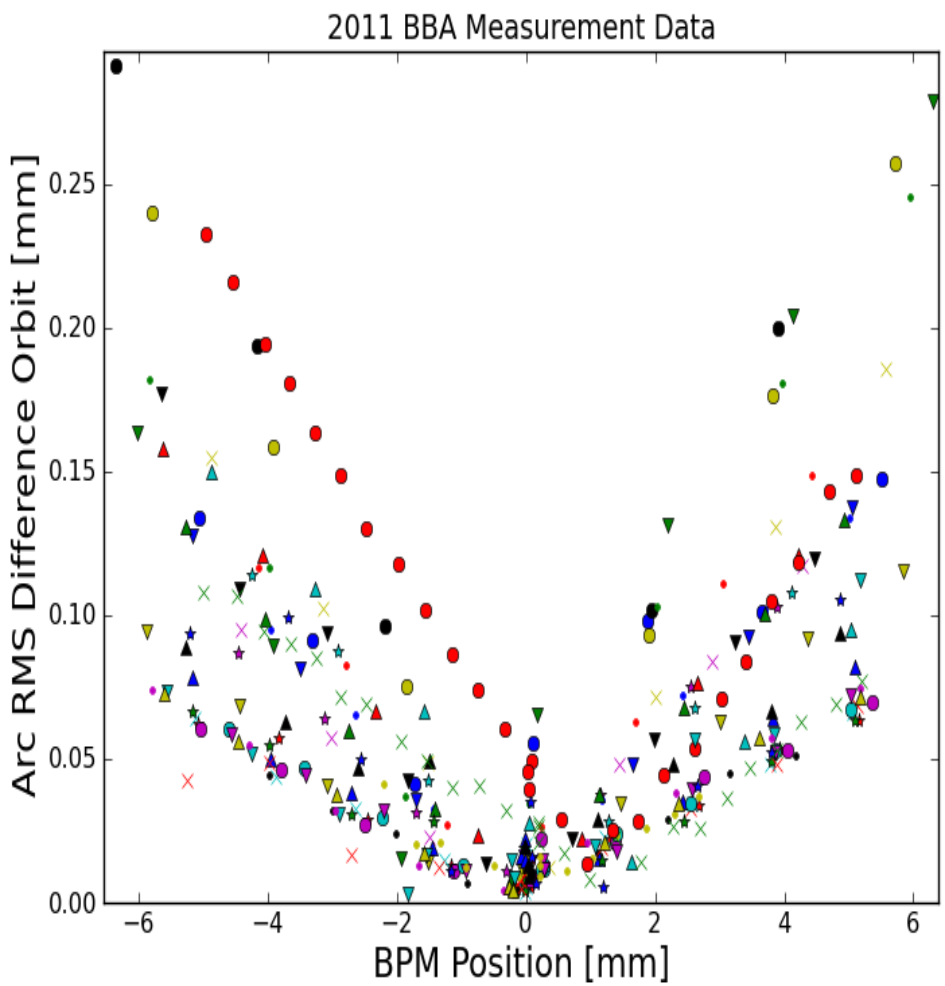
13 in Blue; 1 in Yellow

Results:

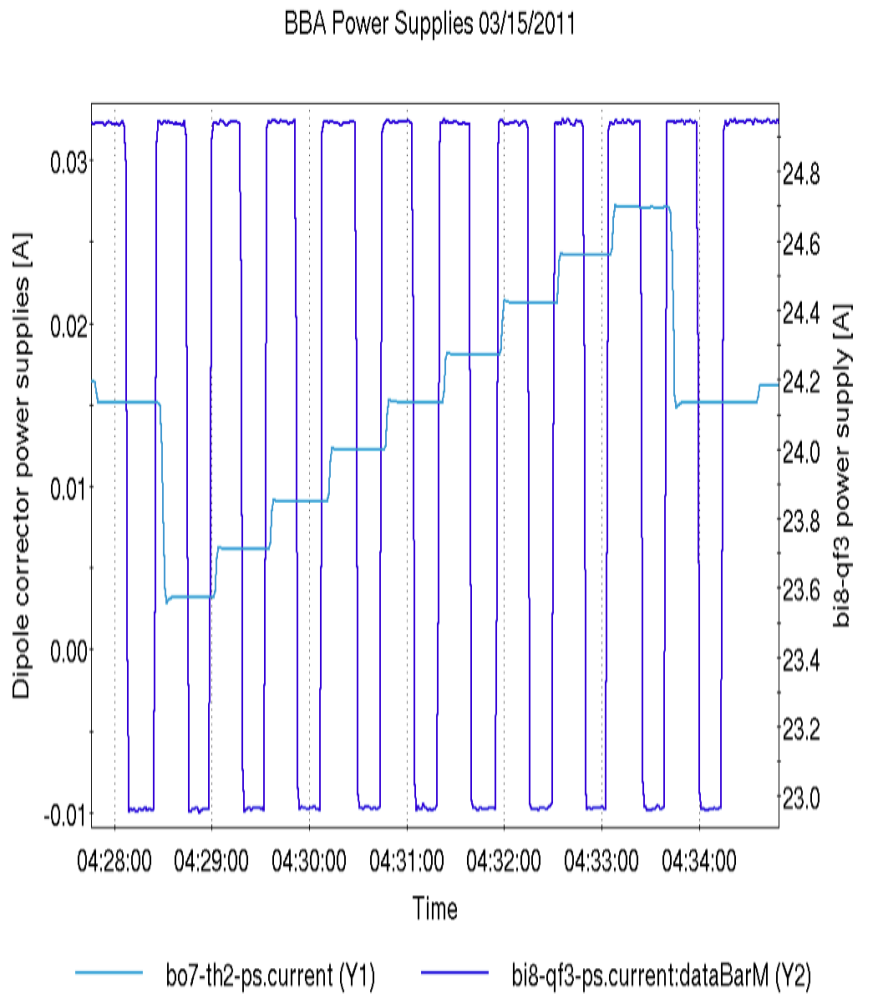
1) Almost all results with re-measurements were inconsistent;
2) No error analysis;
3) Average error bar: 314μm (reanalyzed in 2012)

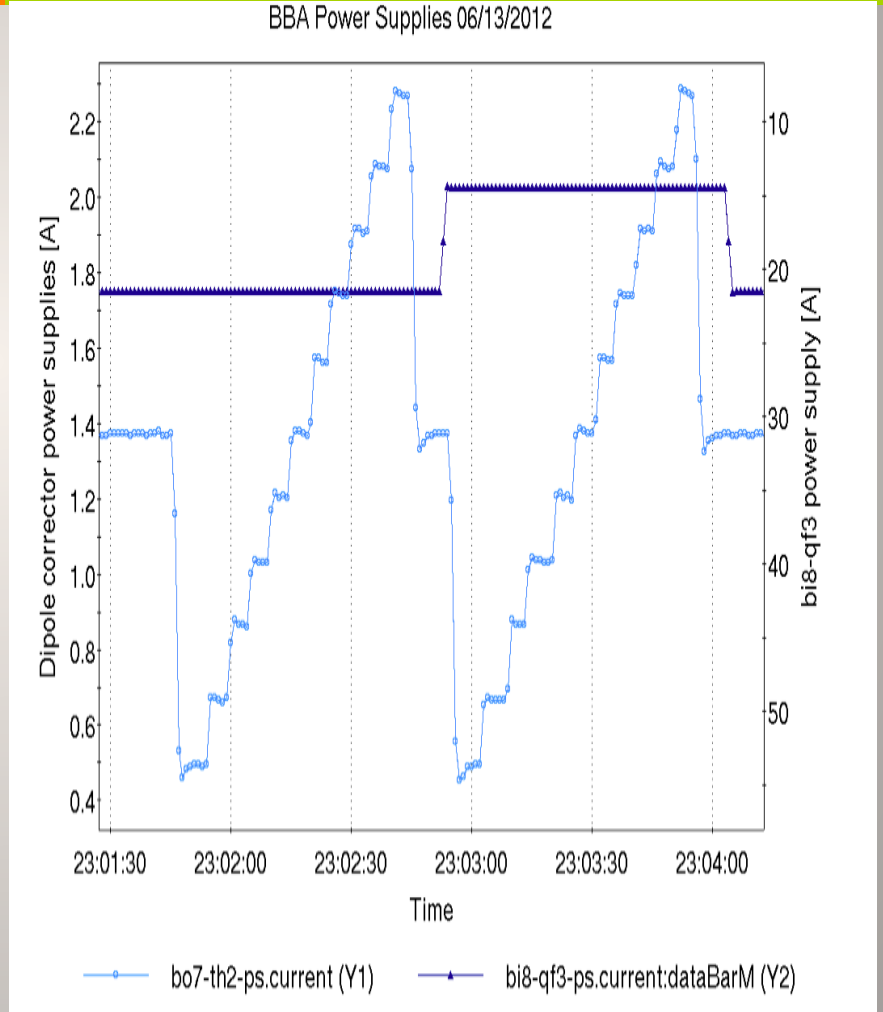
Corrections:

None



BBA Procedure Before and After Run 2012





BBA Measurement Status (at Q1, Q3 & tQ4)

During RHIC run12

	bh1	bv1	bh3	bv3	bh4	bv4		bh1	bv1	bh3	bv3	bh4	bv4
bi1-	6/13/12	6/13/12	6/13/12	6/13/12			yo1-	5/30/12	5/30/12	6/13/12	5/30/12	6/13/12	6/13/12
bo2-	6/13/12	6/13/12	6/13/12	6/13/12			yi2-	5/30/12	5/30/12	5/30/12	6/13/12	6/13/12	6/13/12
bo3-	6/13/12	6/13/12	6/13/12	6/13/12			yi3-	6/13/12	6/13/12	5/30/12	6/13/12		
bi4-	6/13/12	6/13/12	6/13/12	6/13/12			yo4-	6/13/12	6/13/12	6/13/12	6/13/12		
bi5-	5/2/12	5/2/12	5/30/12	5/30/12	5/2/12	6/13/12	yo5-	5/16/12	5/16/12	5/16/12	5/16/12	5/16/12	5/16/12
bo6-	5/2/12	5/2/12	5/2/12	5/2/12	5/2/12	5/2/12	yi6-	5/16/12	5/16/12	5/16/12	5/16/12	5/16/12	5/16/12
bo7-	5/30/12	5/30/12	5/30/12	6/13/12	6/13/12	6/13/12	yi7-	5/16/12	5/16/12	5/16/12	5/16/12	6/13/12	5/17/12
bi8-	5/30/12	5/30/12	6/13/12	5/30/12	6/13/12	6/13/12	yo8-	5/30/12	5/30/12	6/13/12	5/17/12	5/17/12	6/13/12
bi9-	5/30/12	5/30/12	6/13/12	6/13/12			yo9-	5/17/12	6/13/12	6/13/12	5/30/12	6/13/12	6/13/12
bo10-	6/13/12	6/13/12	6/13/12	5/30/12			yi10-	5/17/12	5/30/12	6/13/12	6/13/12	6/13/12	6/13/12
bo11-	5/30/12	5/30/12	5/30/12	6/13/12	6/13/12	6/13/12	yi11-	6/13/12	6/13/12	5/30/12	5/30/12	6/13/12	6/13/12
bi12-	5/30/12	5/30/12	5/30/12	6/13/12	6/13/12	6/13/12	yo12-	5/30/12	5/30/12	6/13/12	5/30/12	6/13/12	6/13/12

RHIC BBA Activity 2012

RHIC 2012 Activities

Measurements:

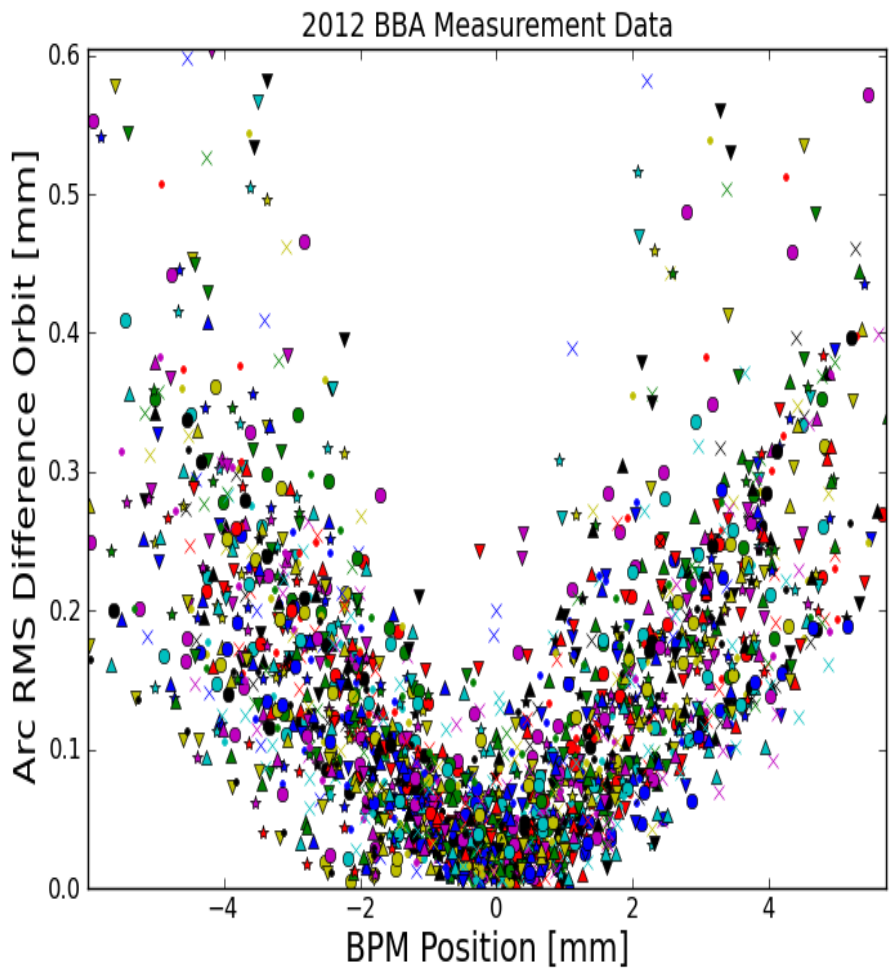
60 in Blue; 68 in Yellow
(Finished all Q1s & Q3s)

Results:

1) Most of re-measurements were consistent;
2) With error analysis;
3) Average error bar: 138μm

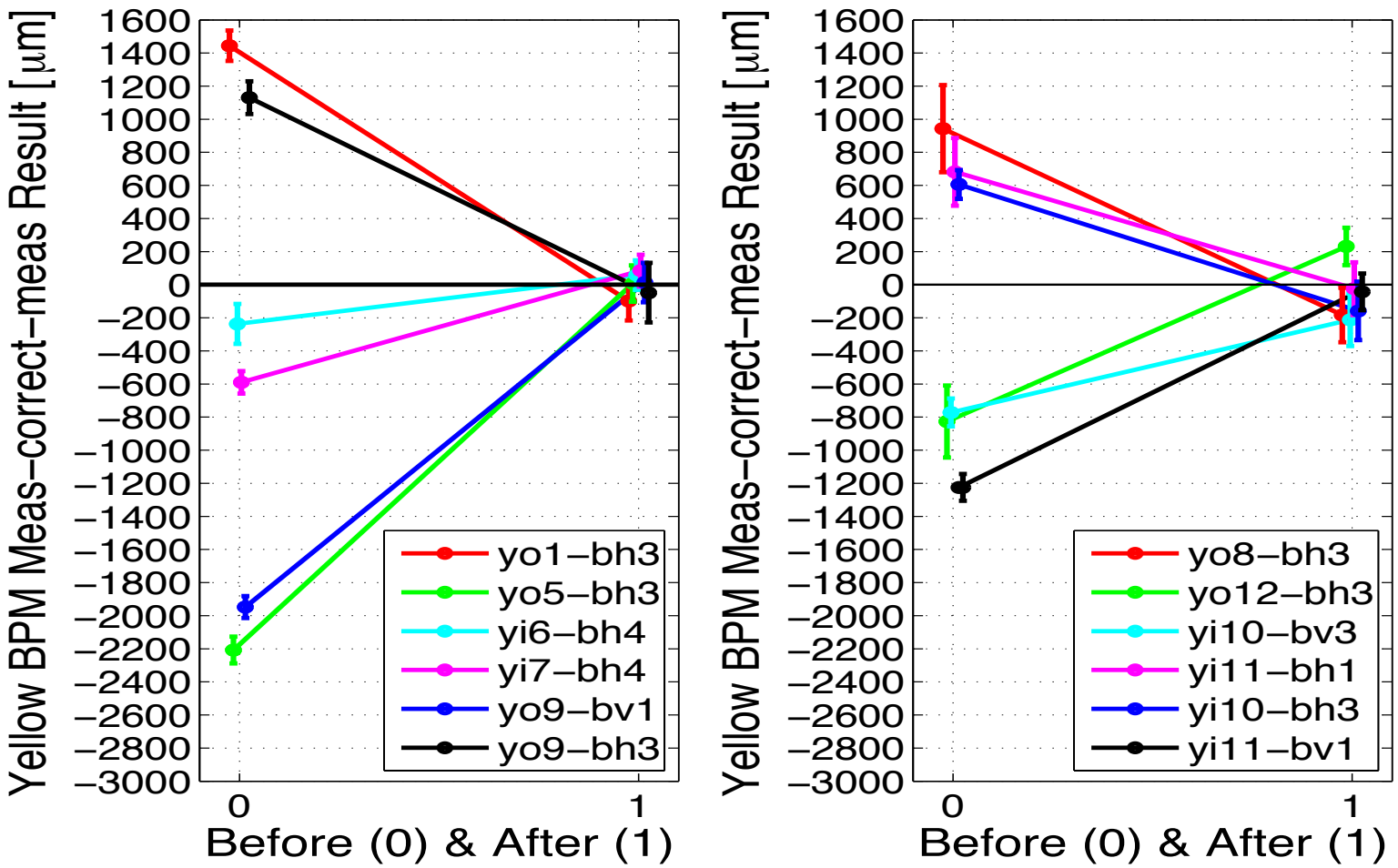
Corrections:

All BPMs that are measured (60 in Blue; 68 in Yellow)



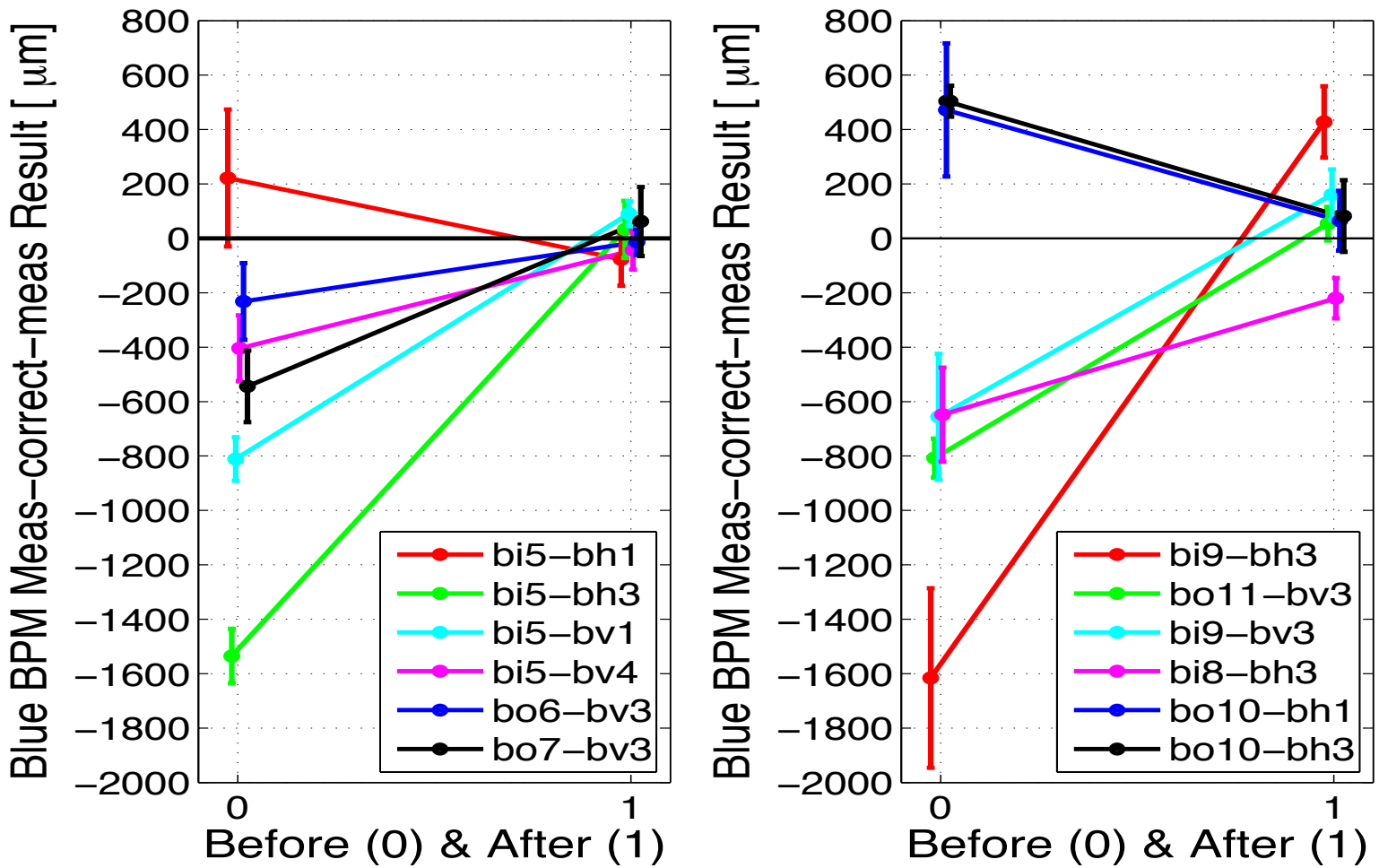
Results from the BBA Analysis

All BPMs in Yellow measured-corrected-remeasured in run 12



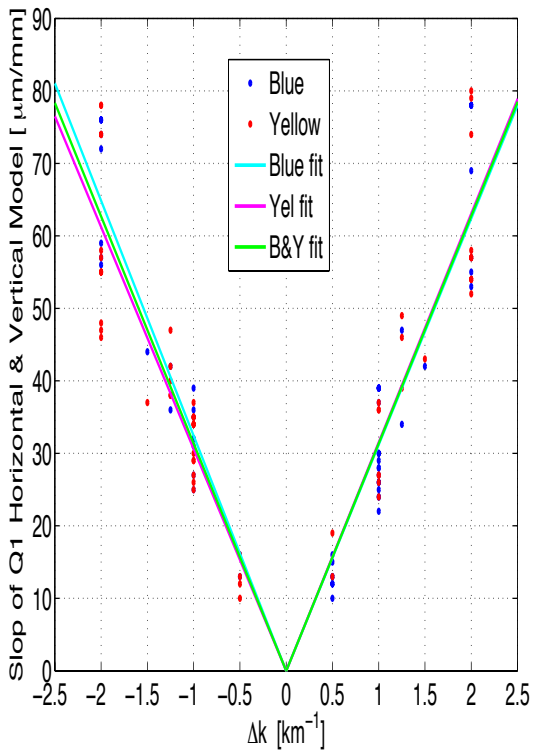
Results from the BBA Analysis

All BPMs in Blue measured-corrected-remeasured in run 12



BBA Model Sensitivity Analysis

Include all Q1, Q3 and tQ4 Measurement data obtained during run 2012

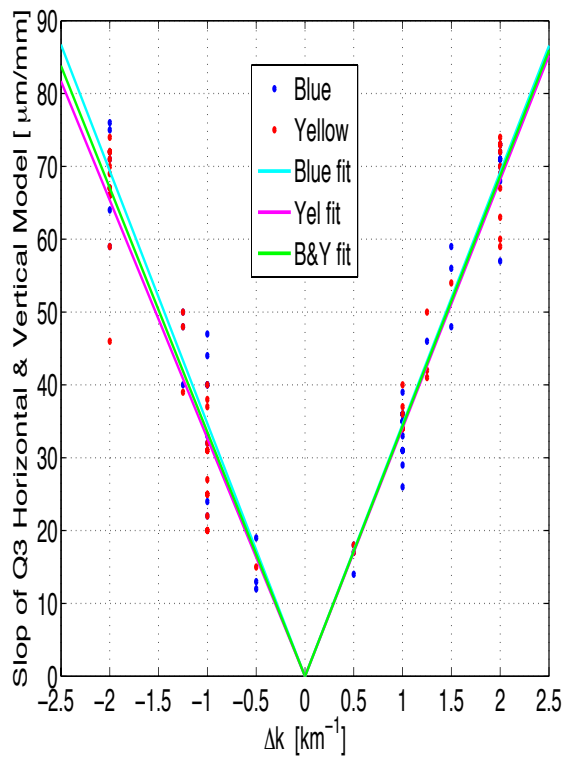


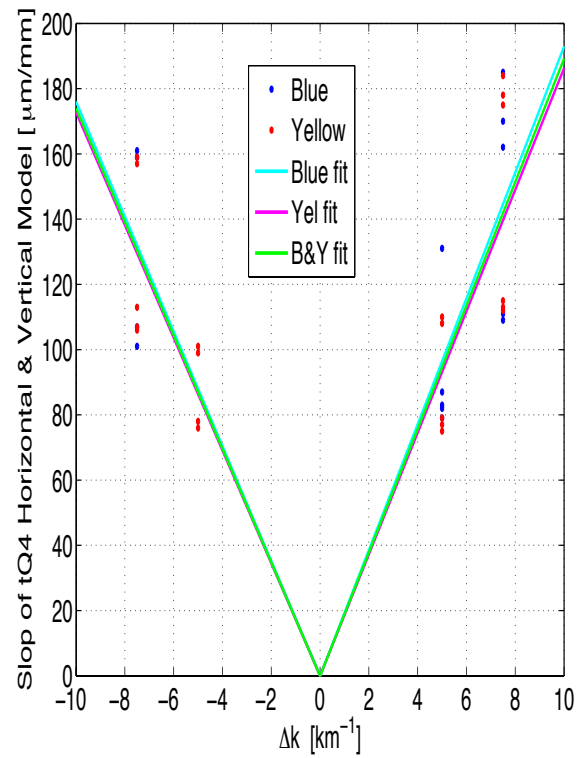
Q1

<d(dA/dx)/dk> = 31.4

Q3

<d(dA/dx)/dk> = 33.9



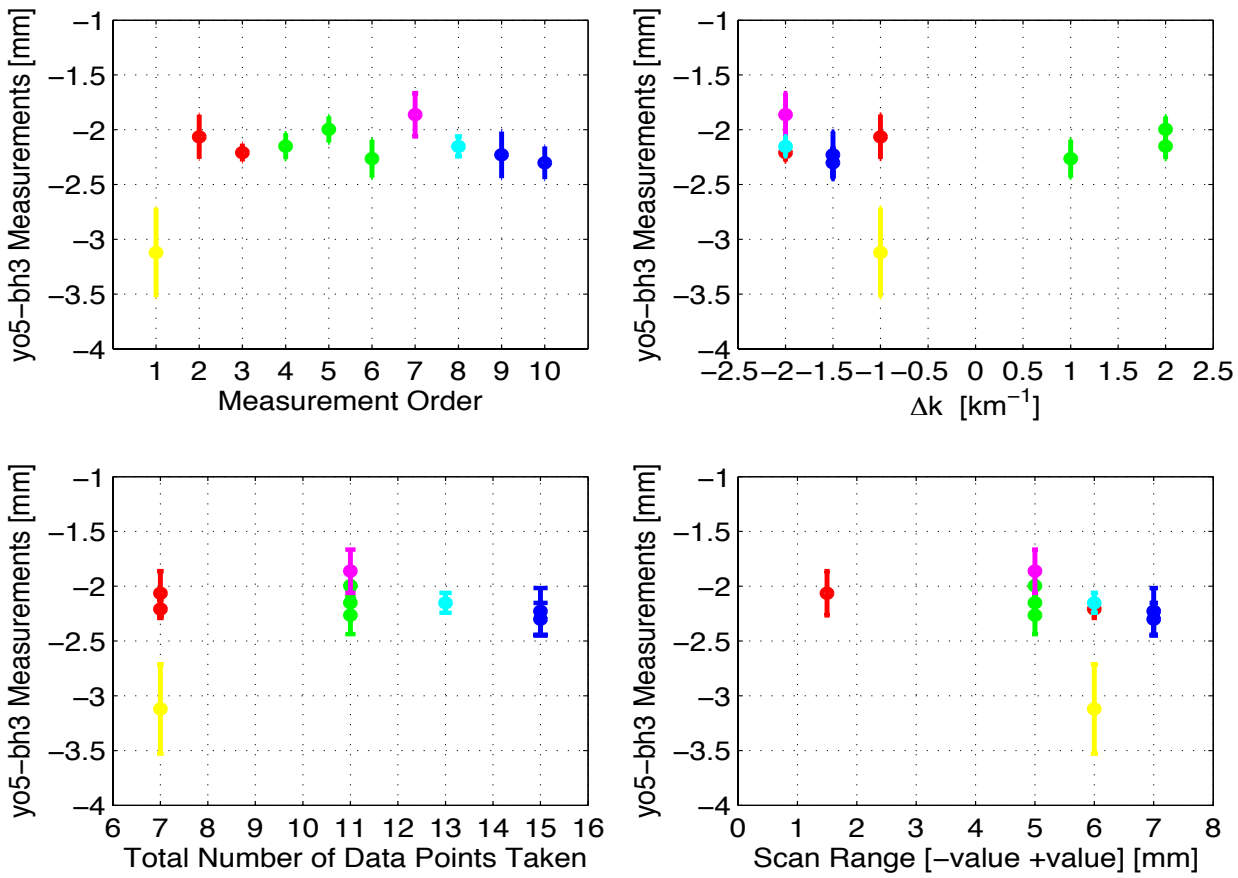


tQ4

<d(dA/dx)/dk> = 18.2

BBA Measurement Repeatability

yo5-bh3 Measurements during run 12



Yellow: Au104 Phys
Fill 11882, Δk=-1e-3
3/9/2010, 23:21

Red: pp12b-v2 Phys
Fill 16600 Δk=-(1,2)e-3
3/21/2012, 18:10,19:03

Green: U12-v2 Set up
Fill 16756 Δk=(1,2)e-3
4/22/2012, 9:50-10:00

Purple: U12-v2 Set up
Fill 16756 Δk=-2e-3
4/22/2012, 10:04

Cyan: U12-v2 Set up
Fill 16758 Δk=-2e-3
4/22/2012, 11:56

Cyan: U12-v2 Set up
Fill 16758 Δk=1.5e-3
4/22/2012, 12:01, 12:06

2012 measurements: Center_ave = -2137μm, Center_RMS = 140μm, Error_ave = 149μm, Error_RMS = 50μm

RHIC BBA Activity Documentations

1. BBA Database tables:
   a) "BBAelem": Complete elements (384 total) with 15 searchable attributes;
   b) "BBAMeas": All measurements (257 sets) with 37 searchable attributes;
   c) Accessible online: http://www.cadops.bnl.gov/Controls/apps/dbbrowser/

2. BBA elog was established at the beginning of run 2012. Records:
   a) pre APEX planning, notes during APEX sessions, post APEX summaries;
   b) Data process preliminary and final graphs and results;
   c) BBA discussions and communications.
   Accessible online: http://www.cadops.bnl.gov/cgi-bin/elog/

3. Run-time log-files are saved automatically by the measurement script.

4. PAC 2011 paper reported 2010 and 2011 activities and major results.

5. Tech Note reports 2012 activities and major results.

Summary of BBA Measurement and Correction

1. Now RHIC BBA generally produces reliable results.

2. It takes 3-4 min for each set of measurement with good accuracy/resolution.

1. Measurements at Q1 & Q3 completed in both rings (24H, 24V each ring).

2. BPM at tQ4 completed in Blue IR 6, 8, 12; in Yellow IR 2, 6, 8, 10, 12.
   (12 of 24 in Blue, 20 of 24 in Yellow).

4. Confirmed some of large offsets. Results are generally good.

5. Offset corrections are installed on all BPMs measured.

6. There is some room for measurement improvement
   a) Systematic measurement (initial investigation in IR 6 during APEX)
   b) Iteration through measure-align-measure-align

Summary of BBA Analysis

1. All the existing measurement data in the RHIC BBA history (after BPM offset fix in 2010) are revisited.
2. All the useful data are identified and and reanalyzed with the 2012 analysis program.
3. All the useful results were used during run 2012
4. Accuracy limitations of BBA on RHIC: ~0.1 mm
   a) BPM accuracy is temperature dependent: $\Delta 10^{\circ}$F ~ Δ1ns trigger time ~ +/-0.1mm up to 1mm BPM reading. BPM resolution is about 10μm for a constant temperature;
   b) Quads physical misalignment relative to each other: ~ +/-0.06mm;
   c) BPM position with respect to the outside fiducials: ~ +/-0.13mm;
   d) Unknown beam positions and angles at the first corrector (beam angle at triplet). It was estimated to have the effect ~ +/-0.1mm
   e) Repeatability under same machine settings: ~ +/-0.1mm

APEX Plan for Run 2013

1. Obtain data on all measurable Quad-BPM planes.

   Not_yet_measureed / total_measurable: Blue 65/125, Yellow 57/125

2. Further explore and improve BBA measurement.

   For example: Iteration through measure-align-measure-align

3. Further explore and improve BBA analysis.

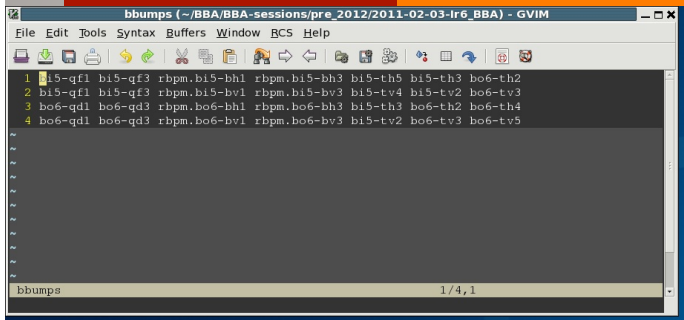
   For example: Using signals from a subset of BPMs which have above average signal/noise ratio

4. Further explore and improve BBA offset corrections.

   For example: Work toward the correction model by minimizing the beam deflection in quads including the magnet survey measurement

5. Testing, improving, and fine tuning application code.

BBA Code: Past



BBA Code: Past

bbumps (~/BBA/BBA-sessions/pre_2012/2011-02-03-Ir6_BBA) - GVIM

File Edit Tools Syntax

```
bi5-qf1 bi5-qf3 r
bi5-qf1 bi5-qf3 r
bo6-qd1 bo6-qd3 r
bo6-qd1 bo6-qd3 r
```

bbumps

getBumpOptics.pl (/cfs/z/owl/satogata/BBA/bin) - GVIM3

File Edit Tools Syntax Buffers Window RCS Help

```perl
#!/usr/bin/perl
#####################################################################
# Reads three-bump swns from file, gets twiss parameters from model and prints
# these all out in a convenient format:
#   swn scoord beta alpha
#####################################################################
sub usage() {
    print "Usage: getBumpOptics.pl ramp stone bumpFile\n";
    print "  Reads corrector SWNs from a bump file, fetches optics for each of\n";
    print "  them, and prints output SWNs, optics to the file bumpFile.optics\n";
    print "Output format: corrSwn scoord beta alpha phase\n";
    exit(0);
}

usage() if ($#ARGV!=2);

my $ramp=$ARGV[0];
my $stone=$ARGV[1];
my $infile=$ARGV[2];
my $outfile=$ARGV[2].".optics";

open IN, "<$infile" or "Error: Couldn't open $infile for reading: $!\n";
open OUT, ">$outfile" or "Error: Couldn't open $outfile for writing: $!\n";
%doneList={};
while (<IN>) {
    chomp;
    @names=split /\s+/; # quad1 quad3 bpm1 bpm3 corr1 corr2 corr3
    for (my $n=4;$n<=6;++$n) {
    $swn=$names[$n];
    next if (exists $doneList{$swn});
    print OUT "$swn ";
    my $plane=($swn=~/th/ ? 0 : 1);
    open TWISS,"/ride/release/X86/scripts/getTwissParameters $ramp $stone $swn|";
    my $gotLattParams=0;
    while (<TWISS>) {
        chomp; s/^\s+//g; s/\s+/ /g;
        $gotLattParams=1 if (/The lattice parameters are/);
        next if (!$gotLattParams);
        my ($param,$val)=split;
        if ($plane==0 && $param eq "betax") { $beta=$val; }
        if ($plane==0 && $param eq "alphax")   { $alpha=$val; }
        if ($plane==0 && $param eq "mux")   { $mu=$val; }
        if ($plane==1 && $param eq "betay") { $beta=$val; }
        if ($plane==1 && $param eq "alphay") { $alpha=$val; }
        if ($plane==1 && $param eq "muy")   { $mu=$val; }
        if ($param eq "scoord") { print OUT "$val $beta $alpha $mu "; last; }
    }
    print OUT "\n";
    $doneList{$swn}=1;
    print "did $swn\n";
    }
    close TWISS;
}
close IN;
close OUT;

#####################################################################
# Matrices are ( C1 S1 ; C1' S1' )
#
# $dfi=($phase1-$phase0)*2.0*3.1415926535;
# $sdfi=sin($dfi); $cdfi=cos($dfi);
# $C=sqrt($beta1/$beta0)*($cdfi+$alpha0*$sdfi);
# $S=sqrt($beta0*$beta1)*$sdfi;
# $Cp=-1*(($alpha1-$alpha0)*$cdfi+(1+$alpha0*$alpha1)*$sdfi)/sqrt($beta0*$beta1);
# $Sp=sqrt($beta0/$beta1)*($cdfi-$alpha1*$sdfi);
#
# dx1' = 1/S1 ; dx2' = -(C2 S1 + S2 S1p)/(S1*S2) ; dx3' = S2/S1^2
```

/cfs/z/owl/satogata/BBA/bin/getBumpOptics.pl [perl] 31/69,1

BBA Code: Past

bbumps (~/BBA/BBA-sessions/pre_2012/2011-02-03-Ir6_BBA) - GVIM

```
bi5-qf1 bi5-qf3 r
bi5-qf1 bi5-qf3 r
bo6-qd1 bo6-qd3 r
bo6-qd1 bo6-qd3 r
```

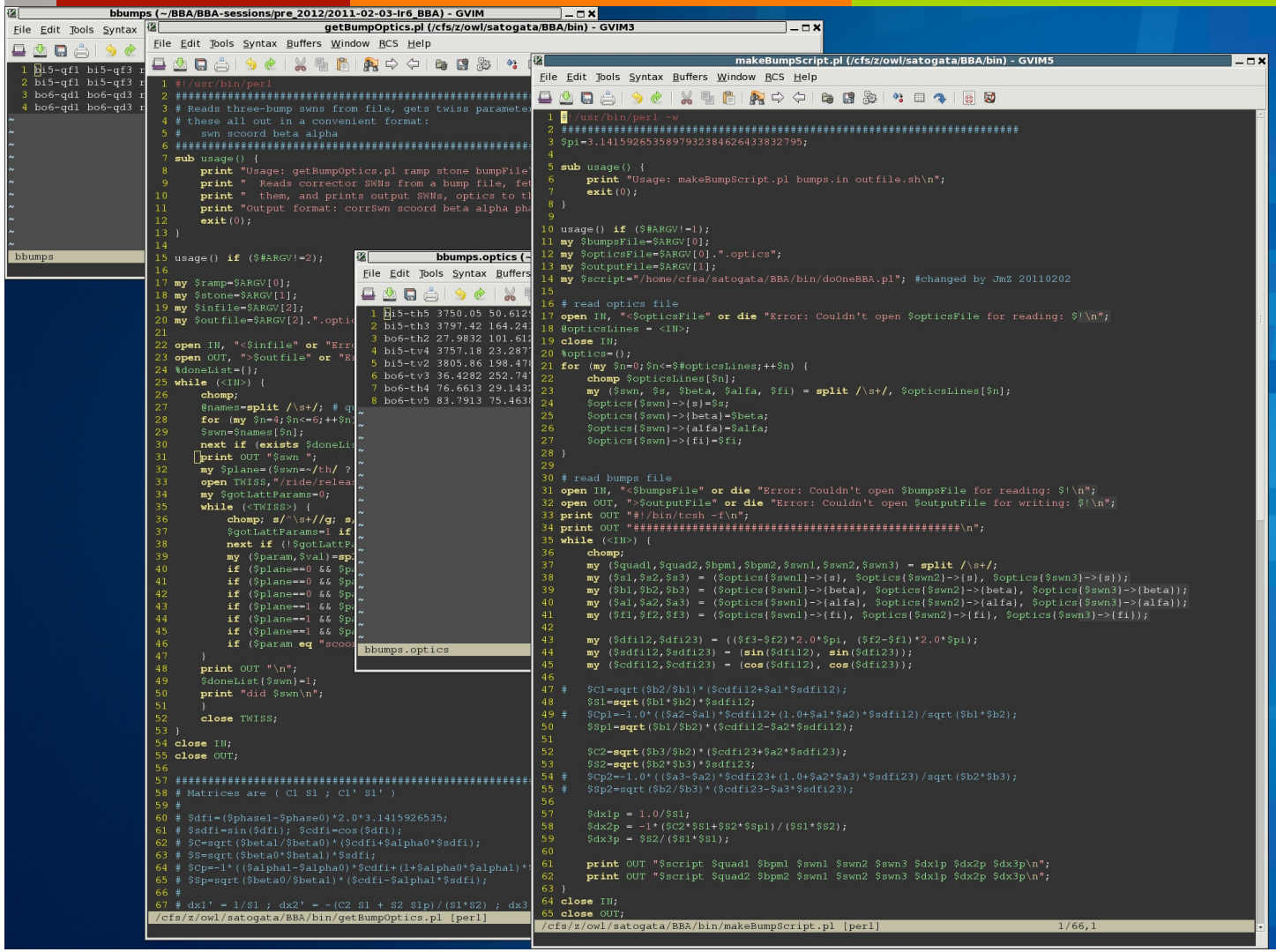
getBumpOptics.pl (/cfs/z/owl/satogata/BBA/bin) - GVIM3

```
#!/usr/bin/perl
#####################################################################
# Reads three-bump swns from file, gets twiss parameters from model and prints
# these all out in a convenient format:
#    swn scoord beta alpha
#####################################################################
sub usage() {
    print "Usage: getBumpOptics.pl ramp stone bumpFile\n";
    print "  Reads corrector SWNs from a bump file, fetches optics for each of\n";
    print "  them, and prints output SWNs, optics to the file bumpFile.optics\n";
    print "Output format: corrSwn scoord beta alpha phase\n";
    exit(0);
}

usage() if ($#ARGV!=2);

my $ramp=$ARGV[0];
my $stone=$ARGV[1];
my $infile=$ARGV[2];
my $outfile=$ARGV[2].".optic

open IN, "<$infile" or "Erro
open OUT, ">$outfile" or "Er
%doneList={};
while (<IN>) {
    chomp;
    @names=split /\s+/; # q
    for (my $n=4;$n<=6;++$n
    $swn=$names[$n];
    next if (exists $doneLis
    print OUT "$swn ";
    my $plane=($swn=~/th/ ?
    open TWISS,"/ride/releas
    my $gotLattParams=0;
    while (<TWISS>) {
        chomp; s/^\s+//g; s
        $gotLattParams=1 if
        next if (!$gotLattP
        my ($param,$val)=sp
        if ($plane==0 && $p
        if ($plane==0 && $p
        if ($plane==0 && $p
        if ($plane==1 && $p
        if ($plane==1 && $p
        if ($plane==1 && $p
        if ($param eq "scoo
    }
    print OUT "\n";
    $doneList{$swn}=1;
    print "did $swn\n";
    }
    close TWISS;
}
close IN;
close OUT;

#####################################################################
# Matrices are ( C1 S1 ; C1' S1' )
#
# $dfi=($phase1-$phase0)*2.0*3.1415926535;
# $sdfi=sin($dfi); $cdfi=cos($dfi);
# $C=sqrt($beta1/$beta0)*($cdfi+$alpha0*$sdfi);
# $S=sqrt($beta0*$beta1)*$sdfi;
# $Cp=-1*(($alpha1-$alpha0)*$cdfi+(1+$alpha0*$alpha1)*$sdfi)/sqrt($beta0*$beta1);
# $Sp=sqrt($beta0/$beta1)*($cdfi-$alpha1*$sdfi);
#
# dx1' = 1/S1 ; dx2' = -(C2 S1 + S2 S1p)/(S1*S2) ; dx3' = S2/S1^2
/cfs/z/owl/satogata/BBA/bin/getBumpOptics.pl [perl]          31/69,1
```

bbumps.optics (~/BBA/BBA-sessions/pre_2012/2011-02-03-Ir6_BBA) - GVIM4

```
bi5-th5 3750.05 50.6129 -2.6672 28.324
bi5-th3 3797.42 164.243 -3.51006 28.5212
bo6-th2 27.9832 101.612 -9.95141 0.234148
bi5-tv4 3757.18 23.2877 -2.32969 29.0813
bi5-tv2 3805.86 198.478 19.4051 29.4632
bo6-tv3 36.4282 252.747 4.69925 0.227262
bo6-th4 76.6613 29.1432 3.69542 0.575839
bo6-tv5 83.7913 75.4638 3.74799 0.335363
bbumps.optics                                  1/8,1
```

BBA Code: Past



BBA Code: Past

bbumps (~/BBA/BBA-sessions/pre_2012/2011-02-03-lr6_BBA) - GVIM

```
bi5-qf1 bi5-qf3 r
bi5-qf1 bi5-qf3 r
bo6-qd1 bo6-qd3 r
bo6-qd1 bo6-qd3 r
```

getBumpOptics.pl (/cfs/z/owl/satogata/BBA/bin) - GVIM3

```
#!/usr/bin/perl
##########################################################
# Reads three-bump swns from file, gets twiss paramete
# these all out in a convenient format:
#    swn scoord beta alpha
##########################################################
sub usage() {
    print "Usage: getBumpOptics.pl ramp stone bumpFile
    print "  Reads corrector SWNs from a bump file, fe
    print "  them, and prints output SWNs, optics to t
    print "Output format: corrSwn scoord beta alpha ph
    exit(0);
}

usage() if ($#ARGV!=2);

my $ramp=$ARGV[0];
my $stone=$ARGV[1];
my $infile=$ARGV[2];
my $outfile=$ARGV[2].".optic

open IN, "<$infile" or "Err
open OUT, ">$outfile" or "E
%doneList={};
while (<IN>) {
    chomp;
    @names=split /\s+/; # q
    for (my $n=4;$n<=6;++$n
    $swn=$names[$n];
    next if (exists $doneLi
    print OUT "$swn ";
    my $plane=($swn=~/th/ ?
    open TWISS,"/ride/relea
    my $gotLattParams=0;
    while (<TWISS>) {
        chomp; s/^\s+//g; s
        $gotLattParams=1 if
        next if (!$gotLattP
        my ($param,$val)=sp
        if ($plane==0 && $p
        if ($plane==0 && $p
        if ($plane==0 && $p
        if ($plane==1 && $p
        if ($plane==1 && $p
        if ($plane==1 && $p
        if ($param eq "scoo
    }
    print OUT "\n";
    $doneList{$swn}=1;
    print "did $swn\n";
    }
    close TWISS;
}
close IN;
close OUT;

##########################################################
# Matrices are ( C1 S1 ; C1' S1' )
#
# $dfi=($phase1-$phase0)*2.0*3.1415926535;
# $sdfi=sin($dfi); $cdfi=cos($dfi);
# $C=sqrt($beta1/$beta0)*($cdfi+$alpha0*$sdfi);
# $S=sqrt($beta0*$beta1)*$sdfi;
# $Cp=-1*(($alpha1-$alpha0)*$cdfi+(1+$alpha0*$alpha1)*
# $Sp=sqrt($beta0/$beta1)*($cdfi-$alpha1*$sdfi);
#
# dx1' = 1/S1 ; dx2' = -(C2 S1 + S2 S1p)/(S1*S2) ; dx3
```

bbumps.optics

```
bi5-th5 3750.05 50.612
bi5-th3 3797.42 164.24
bo6-th2 27.9832 101.61
bi5-tv4 3757.18 23.287
bi5-tv2 3805.86 198.47
bo6-tv3 36.4282 252.74
bo6-th4 76.6613 29.143
bo6-tv5 83.7913 75.463
```

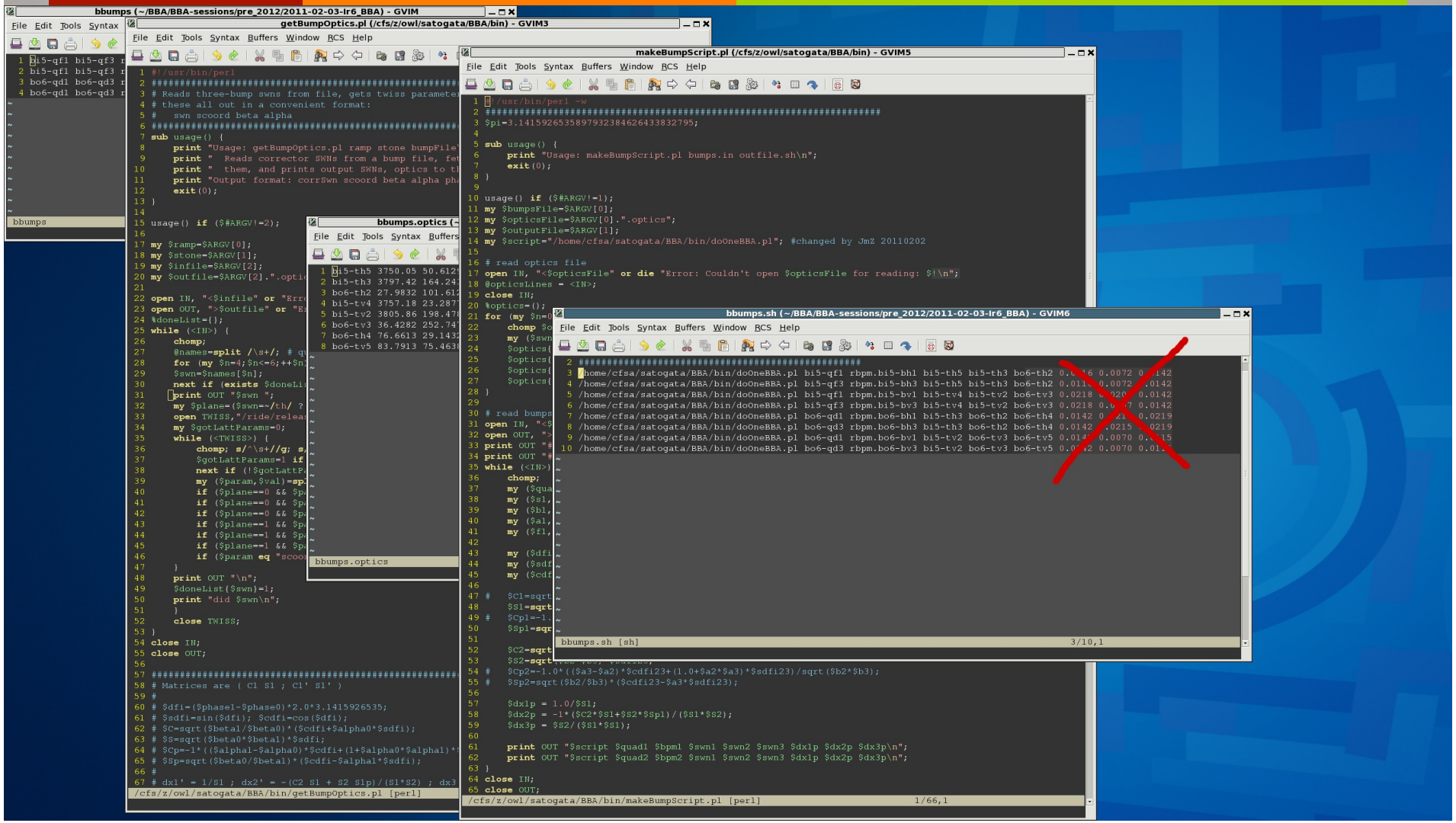
makeBumpScript.pl (/cfs/z/owl/satogata/BBA/bin) - GVIM5

```
#!/usr/bin/perl -w
##########################################################################
$pi=3.14159265358979323846264338327 95;

sub usage() {
    print "Usage: makeBumpScript.pl bumps.in outfile.sh\n";
    exit(0);
}

usage() if ($#ARGV!=1);
my $bumpsFile=$ARGV[0];
my $opticsFile=$ARGV[0].".optics";
my $outputFile=$ARGV[1];
my $script="/home/cfsa/satogata/BBA/bin/doOneBBA.pl"; #changed by JmZ 20110202

# read optics file
open IN, "<$opticsFile" or die "Error: Couldn't open $opticsFile for reading: $!\n";
@opticsLines = <IN>;
close IN;
%optics=();
for (my $n=0
    chomp $o
    my ($swn
    $optics(
    $optics(
    $optics(
    $optics(
}

# read bumps
open IN, "<$
open OUT, ">
print OUT "#
print OUT "#
while (<IN>)
    chomp;
    my ($qua
    my ($s1,
    my ($b1,
    my ($a1,
    my ($f1,

    my ($dfi
    my ($sdf
    my ($cdf

#   $C1=sqrt
    $S1=sqrt
#   $Cp1=-1.
    $Sp1=sqr

    $C2=sqrt
    $S2=sqrt
#   $Cp2=-1.0*(($a3-$a2)*$cdfi23+(1.0+$a2*$a3)*$sdfi23)/sqrt($b2*$b3);
#   $Sp2=sqrt($b2/$b3)*($cdfi23-$a3*$sdfi23);

    $dx1p = 1.0/$S1;
    $dx2p = -1*($C2*$S1+$S2*$Sp1)/($S1*$S2);
    $dx3p = $S2/($S1*$S1);

    print OUT "$script $quad1 $bpm1 $swn1 $swn2 $swn3 $dx1p $dx2p $dx3p\n";
    print OUT "$script $quad2 $bpm2 $swn1 $swn2 $swn3 $dx1p $dx2p $dx3p\n";
}
close IN;
close OUT;
/cfs/z/owl/satogata/BBA/bin/makeBumpScript.pl [perl]                    1/66,1
```

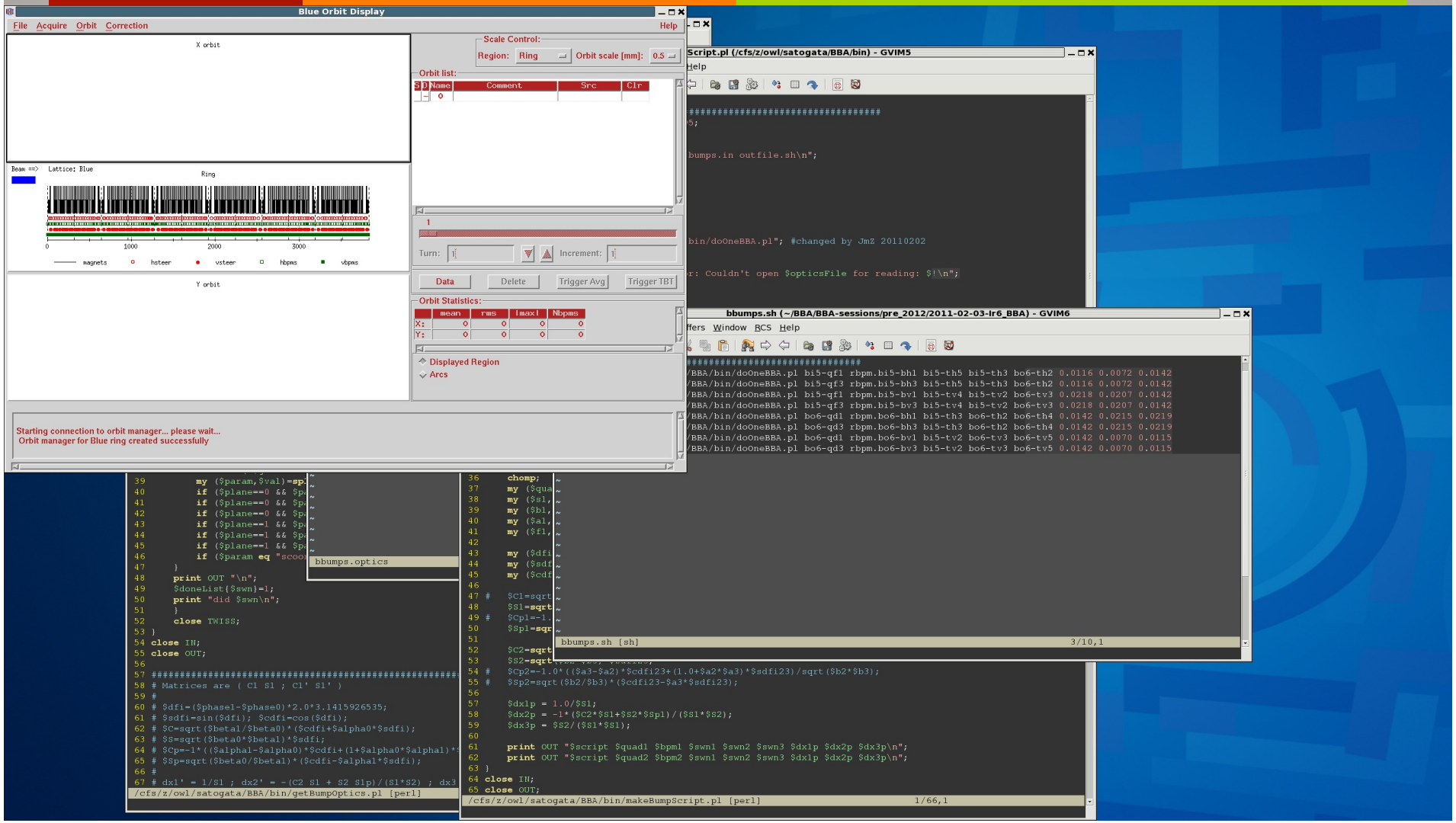
bbumps.sh (~/BBA/BBA-sessions/pre_2012/2011-02-03-lr6_BBA) - GVIM6

```
##############################################
/home/cfsa/satogata/BBA/bin/doOneBBA.pl bi5-qf1 rbpm.bi5-bh1 bi5-th5 bi5-th3 bo6-th2 0.0116 0.0072 0.0142
/home/cfsa/satogata/BBA/bin/doOneBBA.pl bi5-qf3 rbpm.bi5-bh3 bi5-th5 bi5-th3 bo6-th2 0.0116 0.0072 0.0142
/home/cfsa/satogata/BBA/bin/doOneBBA.pl bi5-qf1 rbpm.bi5-bv1 bi5-tv4 bi5-tv2 bo6-tv3 0.0218 0.0207 0.0142
/home/cfsa/satogata/BBA/bin/doOneBBA.pl bi5-qf3 rbpm.bi5-bv3 bi5-tv4 bi5-tv2 bo6-tv3 0.0218 0.0207 0.0142
/home/cfsa/satogata/BBA/bin/doOneBBA.pl bo6-qd1 rbpm.bo6-bh1 bi5-th3 bo6-th2 bo6-th4 0.0142 0.0215 0.0219
/home/cfsa/satogata/BBA/bin/doOneBBA.pl bo6-qd3 rbpm.bo6-bh3 bi5-th3 bo6-th2 bo6-th4 0.0142 0.0215 0.0219
/home/cfsa/satogata/BBA/bin/doOneBBA.pl bo6-qd1 rbpm.bo6-bv1 bi5-tv2 bo6-tv3 bo6-tv5 0.0142 0.0070 0.0115
/home/cfsa/satogata/BBA/bin/doOneBBA.pl bo6-qd3 rbpm.bo6-bv3 bi5-tv2 bo6-tv3 bo6-tv5 0.0142 0.0070 0.0115
bbumps.sh [sh]                                                          3/10,1
```

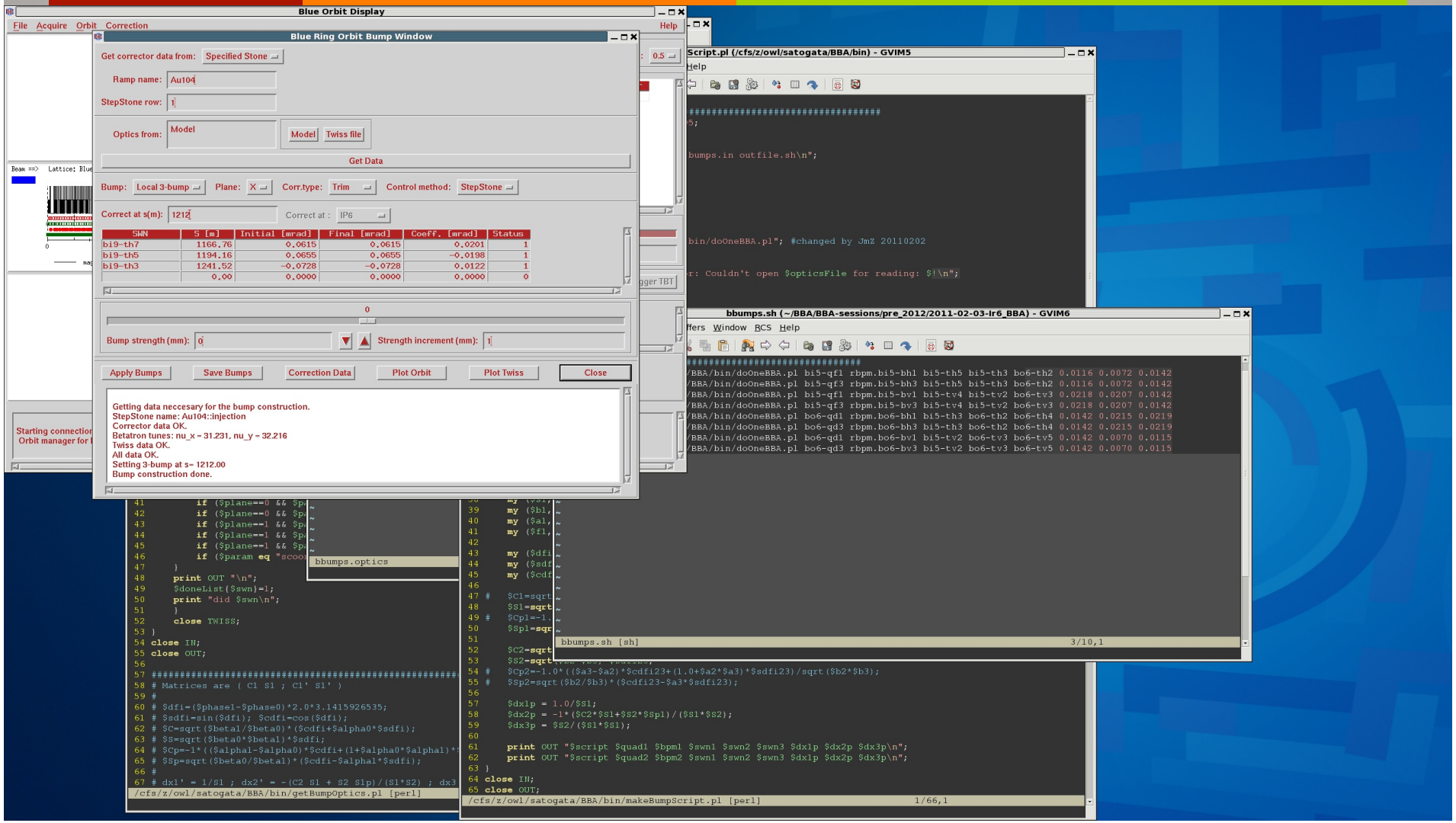
BBA Code: Past



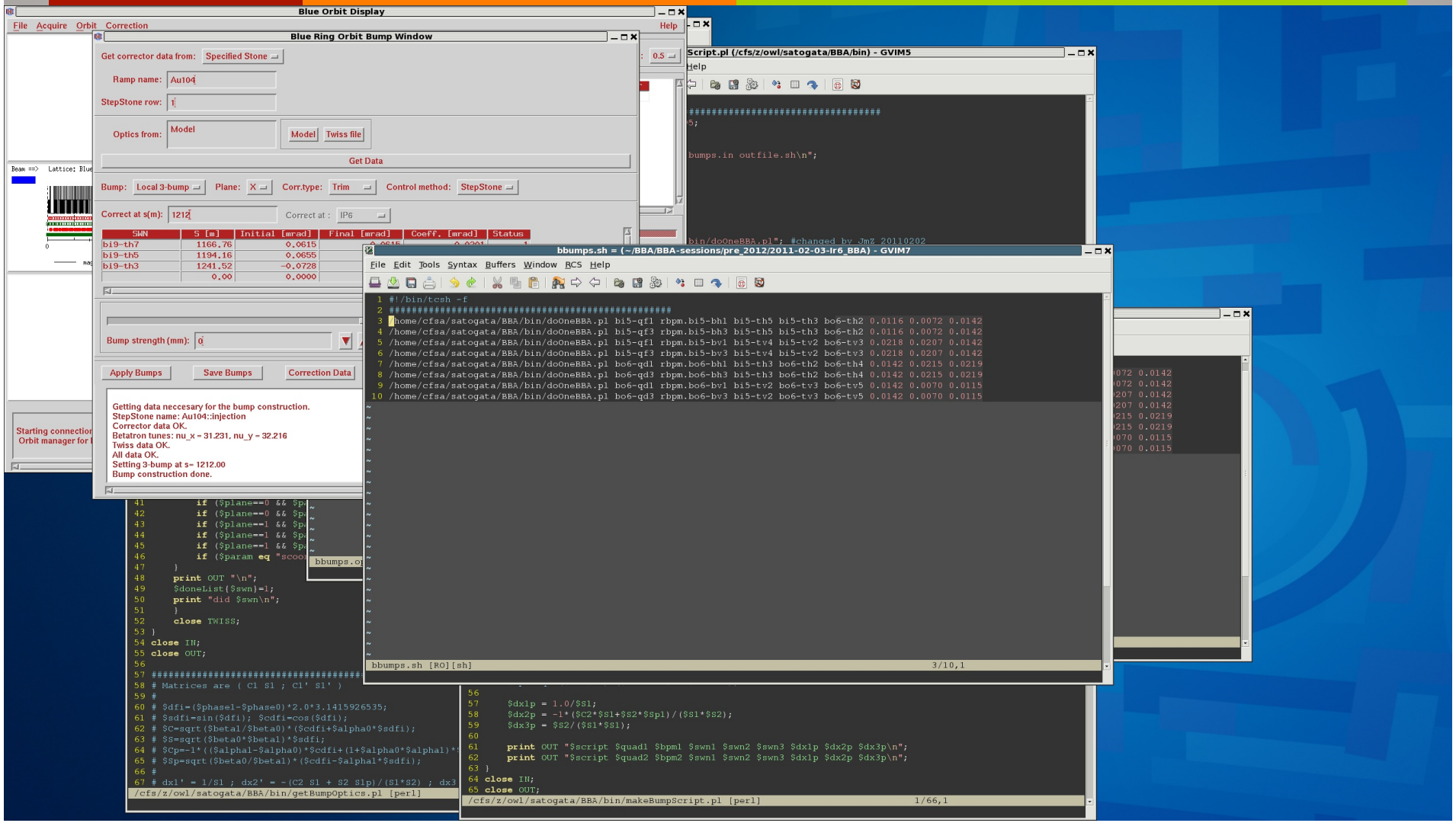
BBA Code: Past



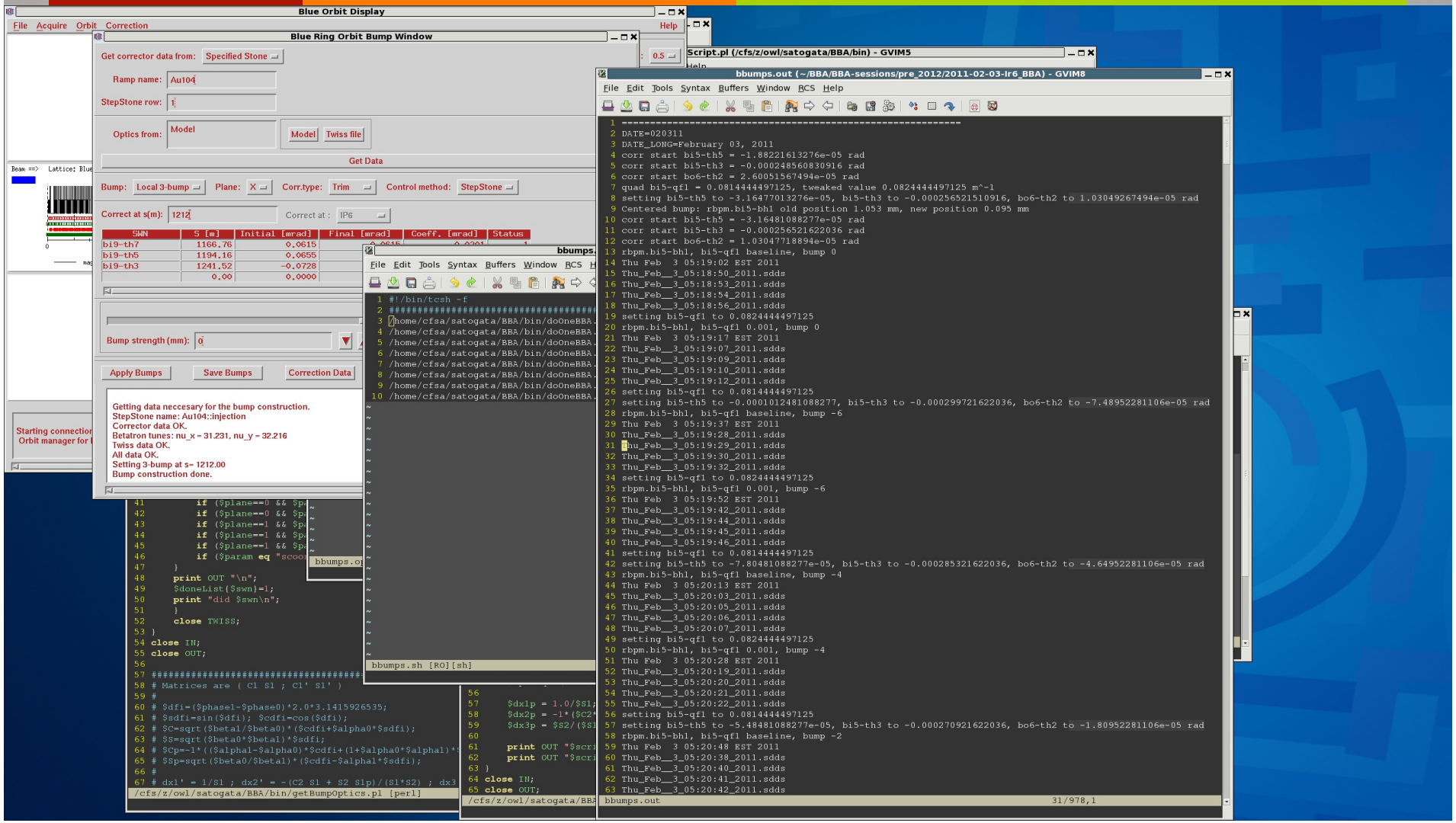
BBA Code: Past



BBA Code: Past



BBA Code: Past



BBA Code: Past

make_BBA_plots.pl (~/BBA/bin/pre_2012) - GVIM9

```
31    $max;
32    )
33
34 ##################################################################
35 #This script takes 4 arguments.  Three here & 4th is bumps.out
36 #If not given 4 arguments, call usage().
37 ##################################################################
38 $runNum=$ARGV[0]; #Run number/name e.g. "run_fy10" or "run_fy11"
39 $fillNum=$ARGV[1]; #Fill number e.g. 11836
40 $ring=$ARGV[2]; #Yellow or Blue
41
42 ##################################################################
43 #Open bumps.out and initialize various counters used for analysis.
44 ##################################################################
45 open(my $input, "<$ARGV[3]") or die "Could not open file '$ARGV[3]'\n";
46 $orbType='2'; #Initialize orbType to a value not 0 or 1
47 $numBaseOrbs=0; #Counter for the number of baseline orbits read
48 $numOffOrbs=0; #Counter for the number of offset orbits read
49 $bumpsRow=0; #Counts which row of bumps.out your on.
50 $delimCounter=0; #Counter for ==== deliminator
51
52 ##################################################################
53 #The main part of this program is opening bumps.out and
54 #looping through each line stopping after each set (for a single bpm)
55 #to do calculations and make plots.
56 ##################################################################
57 while(<$input>)
58   {##################################################################
59    #Analyze each line of bumps.out and set these variables appropriately.
60    ##################################################################
61    next if ($_=~/^#####/); #I comment out bad/useless data with 5 #'s
62    $orbType=0 if ($_=~/baseline/ && $_=~/^rbpm./); #0=baseline
63    $orbType=1 if ($_!~/baseline/ && $_=~/^rbpm./); #1=not baseline
64    $bpmPlane=0 if ($_=~/^rbpm\.[a-z0-9]{3,4}-(b|y)h[0-9]/); #0=horizontal
65    $bpmPlane=1 if ($_=~/^rbpm\.[a-z0-9]{3,4}-(b|y)v[0-9]/); #1=vertical
66    $bpmName=$1 if ($_=~/(^rbpm\.[a-z0-9]{3,4}-(b|y)(h|v)[0-9])/); #Grab bpm name
67    $date=$1 if ($_=~/^DATE=(.*)/); #Grab shorter date for filenames
68    $dateLong=$1 if ($_=~/^DATE_LONG=(.*)/); #Grab longer date
69    $stepSize=$1 if ($_ =~/bumpStep=(-?[0-9]*\.?[0-9]*)/);
70    $maxBump=$1 if ($_ =~/maxBump=(-?[0-9]*\.?[0-9]*)/);
71    $minBump=$1 if ($_ =~/minBump=(-?[0-9]*\.?[0-9]*)/);
72    if ($_ =~/BBA-Type: Repeat Bumps/) {$repeatBumps=1;}
73    if ($_ =~/BBA-Type: Regular V-Plot/) {$repeatBumps=0;}
74
75    ##################################################################
76    #If a line of bumps.out contains an orbit, put it in an array.
77    ##################################################################
78    if ($orbType==0 && $_=~/.sdds$/) {chomp $_; push(@baselines,$_); $numBaseOrbs++;}
79    if ($orbType==1 && $_=~/.sdds$/) {chomp $_; push(@offsets,$_); $numOffOrbs++;}
80
81    ##################################################################
82    #When the number of baseline and offset orbits are eq (except 0=0)
83    #format arguments and send them to cAR.pl. Output > @data2Plot.
84    #IMPORTANT!!!! Had to modify calculateArcRms.pl to hand new (run_fy11)
85    #orbit files that now use a slightly different naming convention. TODO
86    ##################################################################
87    if ($numBaseOrbs==$numOffOrbs && $numBaseOrbs!=0)
88      {$arguments="$runNum "."$fillNum "."$ring "."$bpmName ";
89       foreach $orbit1 (@baselines) {$arguments=$arguments."$orbit1 ";}
90       $arguments=$arguments."-o "; #Seperator between baselines and offsets
91       foreach $orbit2 (@offsets) {$arguments=$arguments."$orbit2 ";}
92       print "$bpmName: Using calculateArcRms.pl on orbits.\n";
93       push (@data2Plot,`/home/cfsa/satogata/BBA/bin/calculateArcRms.pl $arguments`);
```

make_BBA_plots.pl [perl] 68/317,1

Script.pl (/cfs/z/owl/satogata/BBA/bin) - GVIM5

```
63 }
64 close IN;
65 close OUT;
```

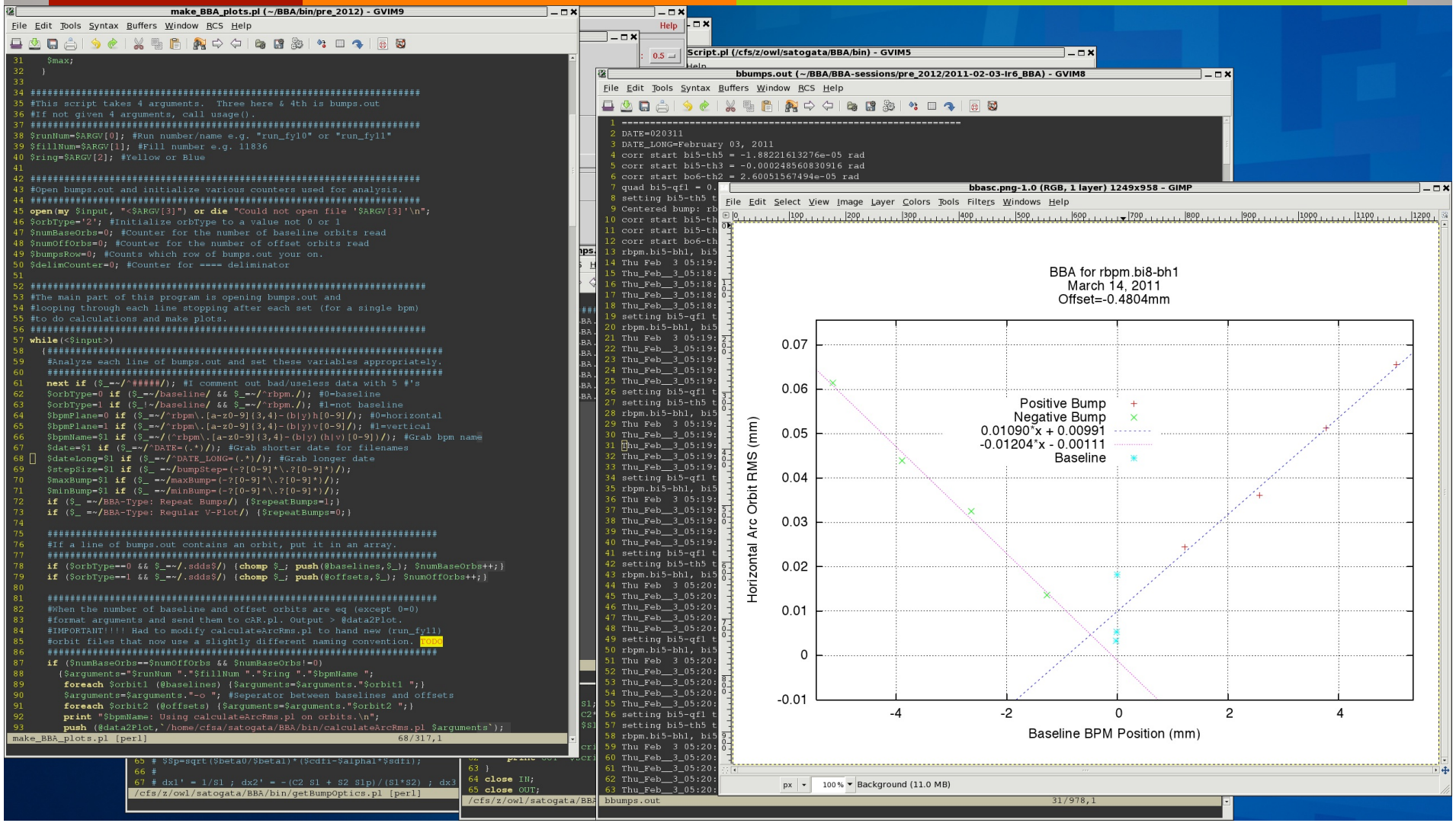
/cfs/z/owl/satogata/BBA

bbumps.out (~/BBA/BBA-sessions/pre_2012/2011-02-03-Ir6_BBA) - GVIM8

```
1 ==========================================================
2 DATE=020311
3 DATE_LONG=February 03, 2011
4 corr start bi5-th5 = -1.88221613276e-05 rad
5 corr start bi5-th3 = -0.000248560830916 rad
6 corr start bo6-th2 = 2.60051567494e-05 rad
7 quad bi5-qf1 = 0.0814444497125, tweaked value 0.0824444497125 m^-1
8 setting bi5-th5 to -3.16477013276e-05, bi5-th3 to -0.000256521510916, bo6-th2 to 1.03049267494e-05 rad
9 Centered bump: rbpm.bi5-bh1 old position 1.053 mm, new position 0.095 mm
10 corr start bi5-th5 = -3.16481088277e-05 rad
11 corr start bi5-th3 = -0.000256521622036 rad
12 corr start bo6-th2 = 1.03047718894e-05 rad
13 rbpm.bi5-bh1, bi5-qf1 baseline, bump 0
14 Thu Feb  3 05:19:02 EST 2011
15 Thu_Feb__3_05:18:50_2011.sdds
16 Thu_Feb__3_05:18:53_2011.sdds
17 Thu_Feb__3_05:18:54_2011.sdds
18 Thu_Feb__3_05:18:56_2011.sdds
19 setting bi5-qf1 to 0.0824444497125
20 rbpm.bi5-bh1, bi5-qf1 0.001, bump 0
21 Thu Feb  3 05:19:17 EST 2011
22 Thu_Feb__3_05:19:07_2011.sdds
23 Thu_Feb__3_05:19:09_2011.sdds
24 Thu_Feb__3_05:19:10_2011.sdds
25 Thu_Feb__3_05:19:12_2011.sdds
26 setting bi5-qf1 to 0.0814444497125
27 setting bi5-th5 to -0.0001012481088277, bi5-th3 to -0.000299721622036, bo6-th2 to -7.48952281106e-05 rad
28 rbpm.bi5-bh1, bi5-qf1 baseline, bump -6
29 Thu Feb  3 05:19:37 EST 2011
30 Thu_Feb__3_05:19:28_2011.sdds
31 Thu_Feb__3_05:19:29_2011.sdds
32 Thu_Feb__3_05:19:30_2011.sdds
33 Thu_Feb__3_05:19:32_2011.sdds
34 setting bi5-qf1 to 0.0824444497125
35 rbpm.bi5-bh1, bi5-qf1 0.001, bump -6
36 Thu Feb  3 05:19:52 EST 2011
37 Thu_Feb__3_05:19:42_2011.sdds
38 Thu_Feb__3_05:19:44_2011.sdds
39 Thu_Feb__3_05:19:45_2011.sdds
40 Thu_Feb__3_05:19:46_2011.sdds
41 setting bi5-qf1 to 0.0814444497125
42 setting bi5-th5 to -7.80481088277e-05, bi5-th3 to -0.000285321622036, bo6-th2 to -4.64952281106e-05 rad
43 rbpm.bi5-bh1, bi5-qf1 baseline, bump -4
44 Thu Feb  3 05:20:13 EST 2011
45 Thu_Feb__3_05:20:03_2011.sdds
46 Thu_Feb__3_05:20:05_2011.sdds
47 Thu_Feb__3_05:20:06_2011.sdds
48 Thu_Feb__3_05:20:07_2011.sdds
49 setting bi5-qf1 to 0.0824444497125
50 rbpm.bi5-bh1, bi5-qf1 0.001, bump -4
51 Thu Feb  3 05:20:28 EST 2011
52 Thu_Feb__3_05:20:19_2011.sdds
53 Thu_Feb__3_05:20:20_2011.sdds
54 Thu_Feb__3_05:20:21_2011.sdds
55 Thu_Feb__3_05:20:22_2011.sdds
56 setting bi5-qf1 to 0.0814444497125
57 setting bi5-th5 to -5.48481088277e-05, bi5-th3 to -0.000270921622036, bo6-th2 to -1.80952281106e-05 rad
58 rbpm.bi5-bh1, bi5-qf1 baseline, bump -2
59 Thu Feb  3 05:20:48 EST 2011
60 Thu_Feb__3_05:20:38_2011.sdds
61 Thu_Feb__3_05:20:40_2011.sdds
62 Thu_Feb__3_05:20:41_2011.sdds
63 Thu_Feb__3_05:20:42_2011.sdds
```

bbumps.out 31/978,1

BBA Code: Past



BBA Code: Present

BBAmeasure.py = (~/BBA/bin) - GVIM

File Edit Tools Syntax Buffers Window RCS Help

```
        raise StopIteration

##############################################################################
#Change name to something else like 'quadMove' maybe.  This is used for quad
#tweaks and is essentially the same as oneMove() above.
##############################################################################
def oneMove2( wfgado, ps, amp ):
    request = [ ( wfgado, 'liveNamesS',  'value', ','.join(ps) ),
                ( wfgado, 'liveStrengthsS', 'value', [amp] ),
                ( wfgado, 'doLiveStrengths', 'value', 'DoDeltaStrengths' ) ]
    if debug:
        for ado, parameter, property, value in request:
            print "setting", ado.systemName, parameter + ':' + property, repr(value)
    setstart = time.time()
    status = cns.adoSet( list = request )
    if status == 0:
        print "command duration =", ("%.3f") % (time.time() - setstart), "seconds"
    else:
        print 'error =', status, "command duration =", ("%.3f") % (time.time() - setstart), "seconds"
        raise StopIteration

#def MakeBump( bumpado, wfgado, requests ):
def MakeBump( bumpado, wfgado, bumpdescription ):

    global logfile
    quad, bpm, dStrength, correctorList, bumpAmpList = bumpdescription
    count = 0
    print >>logfile, 'run:' + run_name, "stone:" + stone_name, "fill:" +\
        fill_name, "ramp:" + ramp_name, "quad:" + quad, "bpm:" + bpm,\
        "dStrength:" + str(dStrength), "K:" + str(getStrength(quad)),\
        "time:" + time.strftime(time_formatf, time.localtime()), \
        "type:" + main.measType, "wait:" + str(rampstart.waittime),\
        "oPOS:" + str(orbPosS) + ' ' #Leave whitespace for regex

    if Qn:
        for bumpampl in bumpAmpList:
            if abs(bumpampl[1]) > 2.0: rampstart.waittime = 5.0
            else: rampstart.waittime = 3.0

            print "Sending ", bumpampl[1], correctorList, " to onemove()\n"
            #Make bump of size bumpampl.  correctorList here is a list of 3 corrs
            oneMove(bumpado, correctorList, bumpampl[1])
            yield (count, correctorList, bumpampl[1], True)

            #Move quad by amount dStrength
            count += 1
            oneMove2( wfgado, [quad], dStrength )
            yield (count, correctorList, bumpampl[1], True)

            #??????
            rampstart_time = rampstart.rampstart_time
            rampend_time = rampstart.rampend_time

            #Put quad back to original strength
            count += 1
            oneMove2( wfgado, [quad], -dStrength )
            yield (count, correctorList, bumpampl[1], False)

            #Print some shit to the log file
            print >>logfile, \
                  time.strftime(time_format, time.localtime(rampstart_time)).replace(' ', '_'), \
                  time.strftime(time_format, time.localtime(rampend_time)).replace(' ', '_'),\
                  bumpampl[0], getBeamDecay(ring)
            logfile.flush()
    if Ql:
        for Strength in (dStrength, -dStrength):
            for bumpampl in bumpAmpList:
                if abs(bumpampl[1]) > 2.0: rampstart.waittime = 5.0
```

BBAmeasure.py [RO][python] 306/630,4

BBA Code: Present

BBAmeasure.py = (~/BBA/bin) - GVIM

File Edit Tools Syntax Buffers Window RCS Help

```
283         raise StopIteration
284
285 ##############################################################################
286 #Change name to something else like 'quadMove' maybe.  This is used for quad
287 #tweaks and is essentially the same as oneMove() above.
288 ##############################################################################
289 def oneMove2( wfgado, ps, amp ):
290     request = [ ( wfgado, 'liveNamesS',  'value', ','.join(ps) ),
291                 ( wfgado, 'liveStrengthsS', 'value', [amp] ),
292                 ( wfgado, 'doLiveStrengths', 'value', 'DoDeltaStrengths' ) ]
293     if debug:
294         for ado, parameter, property, value in request:
295             print "setting", ado.systemName, parameter + ':' + property, repr(value)
296     setstart = time.time()
297     status = cns.adoSet( list = request )
298     if status == 0:
299         print "command duration =", ("%.3f") % (time.time() - set
300     else:
301         print 'error =', status, "command duration =", ("%.3f") %
302         raise StopIteration
303
304 #def MakeBump( bumpado, wfgado, requests ):
305 def MakeBump( bumpado, wfgado, bumpdescription ):
306
307     global logfile
308     quad, bpm, dStrength, correctorList, bumpAmpList = bumpdescri
309     count = 0
310     print >>logfile, 'run:' + run_name, "stone:" + stone_name, "f
311         fill_name, "ramp:" + ramp_name, "quad:" + quad, "bpm:" + 
312         "dStrength:" + str(dStrength), "K:" + str(getStrength(qua
313         "time:" + time.strftime(time_formatf, time.localtime()),
314         "type:" + main.measType, "wait:" + str(rampstart.waittime
315         "oPOS:" + str(orbPosS) + ' '  #Leave whitespace for regex
316
317     if Qn:
318         for bumpampl in bumpAmpList:
319             if abs(bumpampl[1]) > 2.0: rampstart.waittime = 5.0
320             else: rampstart.waittime = 3.0
321
322             print "Sending ", bumpampl[1], correctorList, " to on
323             #Make bump of size bumpampl.  correctorList here is a
324             oneMove(bumpado, correctorList, bumpampl[1])
325             yield (count, correctorList, bumpampl[1], True)
326
327             #Move quad by amount dStrength
328             count += 1
329             oneMove2( wfgado, [quad], dStrength )
330             yield (count, correctorList, bumpampl[1], True)
331
332             #??????
333             rampstart_time = rampstart.rampstart_time
334             rampend_time = rampstart.rampend_time
335
336             #Put quad back to original strength
337             count += 1
338             oneMove2( wfgado, [quad], -dStrength )
339             yield (count, correctorList, bumpampl[1], False)
340
341             #Print some shit to the log file
342             print >>logfile, \
343                 time.strftime(time_format, time.localtime(rampstart_time)).replace(' ', '_'), \
344                 time.strftime(time_format, time.localtime(rampend_time)).replace(' ', '_'),\
345                 bumpampl[0], getBeamDecay(ring)
346             logfile.flush()
347     if Ql:
348         for Strength in (dStrength, -dStrength):
349             for bumpampl in bumpAmpList:
350                 if abs(bumpampl[1]) > 2.0: rampstart.waittime = 5.0
```

BBAmeasure.py [RO][python] 306/630,4

yo8-bh3_log1Q_Wed_Mar_21_18:30:04_...A/BBA-sessions/2012-03-21_APEX) - VIM

mcr@acnmcrbc6: /home/cfsb/mcr/ziegler/BBA-ScreenShot... | yo8-bh3_log1Q_Wed_Mar_21_18:30:04_...A/BBA-sessions/2...

```
1 run:run_fy12 stone:cdevCommand: fill:16600 ramp:pp12b-v2 quad:yo8-qd3 bpm:yo8-bh3 dS
  trength:-0.001 wait:3.0
2 Wed_Mar_21_18:30:07_2012 Wed_Mar_21_18:30:14_2012 -3.0
3 Wed_Mar_21_18:30:14_2012 Wed_Mar_21_18:30:20_2012 -2.0
4 Wed_Mar_21_18:30:20_2012 Wed_Mar_21_18:30:25_2012 -1.0
5 Wed_Mar_21_18:30:25_2012 Wed_Mar_21_18:30:30_2012 0
6 Wed_Mar_21_18:30:30_2012 Wed_Mar_21_18:30:35_2012 1.0
7 Wed_Mar_21_18:30:35_2012 Wed_Mar_21_18:30:40_2012 2.0
8 Wed_Mar_21_18:30:40_2012 Wed_Mar_21_18:30:45_2012 3.0
9 Wed_Mar_21_18:30:45_2012 Wed_Mar_21_18:30:51_2012 0
10 Wed_Mar_21_18:30:53_2012 Wed_Mar_21_18:31:00_2012 -3.0
11 Wed_Mar_21_18:31:00_2012 Wed_Mar_21_18:31:05_2012 -2.0
12 Wed_Mar_21_18:31:05_2012 Wed_Mar_21_18:31:10_2012 -1.0
13 Wed_Mar_21_18:31:10_2012 Wed_Mar_21_18:31:15_2012 0
14 Wed_Mar_21_18:31:15_2012 Wed_Mar_21_18:31:20_2012 1.0
15 Wed_Mar_21_18:31:20_2012 Wed_Mar_21_18:31:25_2012 2.0
16 Wed_Mar_21_18:31:25_2012 Wed_Mar_21_18:31:30_2012 3.0
17 Wed_Mar_21_18:31:30_2012 Wed_Mar_21_18:31:36_2012 0
~
~
~
~
~
~
~
~
~
~
```

yo8-bh3_log1Q_Wed_Mar_21_18:30:04_2012 1/17,1

BBA Code: Present

BBAmeasure.py = (~/BBA/bin) - GVIM

```
283         raise StopIteration
284
285 ##########################################################################
286 #Change name to something else like 'quadMove' maybe.  This is used for quad
287 #tweaks and is essentially the same as oneMove() above.
288 ##########################################################################
289 def oneMove2( wfgado, ps, amp ):
290     request = [ ( wfgado, 'liveNamesS',  'value', ','.join(ps) ),
291                 ( wfgado, 'liveStrengthsS', 'value', [amp] ),
292                 ( wfgado, 'doLiveStrengths', 'value', 'DoDeltaStrengths' ) ]
293     if debug:
294         for ado, parameter, property, value in request:
295             print "setting", ado.systemName, parameter + ':' + property, repr(value)
296     setstart = time.time()
297     status = cns.adoSet( list = request )
298     if status == 0:
299         print "command duration =", ("%.3f") % (time.time() - set
300     else:
301         print 'error =', status, "command duration =", ("%.3f") %
302         raise StopIteration
303
304 #def MakeBump( bumpado, wfgado, requests ):
305 def MakeBump( bumpado, wfgado, bumpdescription ):
306
307     global logfile
308     quad, bpm, dStrength, correctorList, bumpAmpList = bumpdescri
309     count = 0
310     print >>logfile, 'run:' + run_name, "stone:" + stone_name, "f
311         fill_name, "ramp:" + ramp_name, "quad:" + quad, "bpm:" + 
312         "dStrength:" + str(dStrength), "K:" + str(getStrength(qua
313         "time:" + time.strftime(time_formatf, time.localtime()),
314         "type:" + main.measType, "wait:" + str(rampstart.waittime
315         "oPOS:" + str(orbPosS) + ' ' #Leave whitespace for regex
316
317     if Qn:
318         for bumpampl in bumpAmpList:
319             if abs(bumpampl[1]) > 2.0: rampstart.waittime = 5.0
320             else: rampstart.waittime = 3.0
321
322             print "Sending ", bumpampl[1], correctorList, " to on
323             #Make bump of size bumpampl.  correctorList here is a
324             oneMove(bumpado, correctorList, bumpampl[1])
325             yield (count, correctorList, bumpampl[1], True)
326
327             #Move quad by amount dStrength
328             count += 1
329             oneMove2( wfgado, [quad], dStrength )
330             yield (count, correctorList, bumpampl[1], True)
331
332             #??????
333             rampstart_time = rampstart.rampstart_time
334             rampend_time = rampstart.rampend_time
335
336             #Put quad back to original strength
337             count += 1
338             oneMove2( wfgado, [quad], -dStrength )
339             yield (count, correctorList, bumpampl[1], False)
340
341             #Print some shit to the log file
342             print >>logfile, \
343                 time.strftime(time_format, time.localtime(rampstart_time)).replace(' ', '_'), \
344                 time.strftime(time_format, time.localtime(rampend_time)).replace(' ', '_'),\
345                 bumpampl[0], getBeamDecay(ring)
346             logfile.flush()
347     if Q1:
348         for Strength in (dStrength, -dStrength):
349             for bumpampl in bumpAmpList:
350                 if abs(bumpampl[1]) > 2.0: rampstart.waittime = 5.0
BBAmeasure.py [RO][python]                                  306/630,4
```

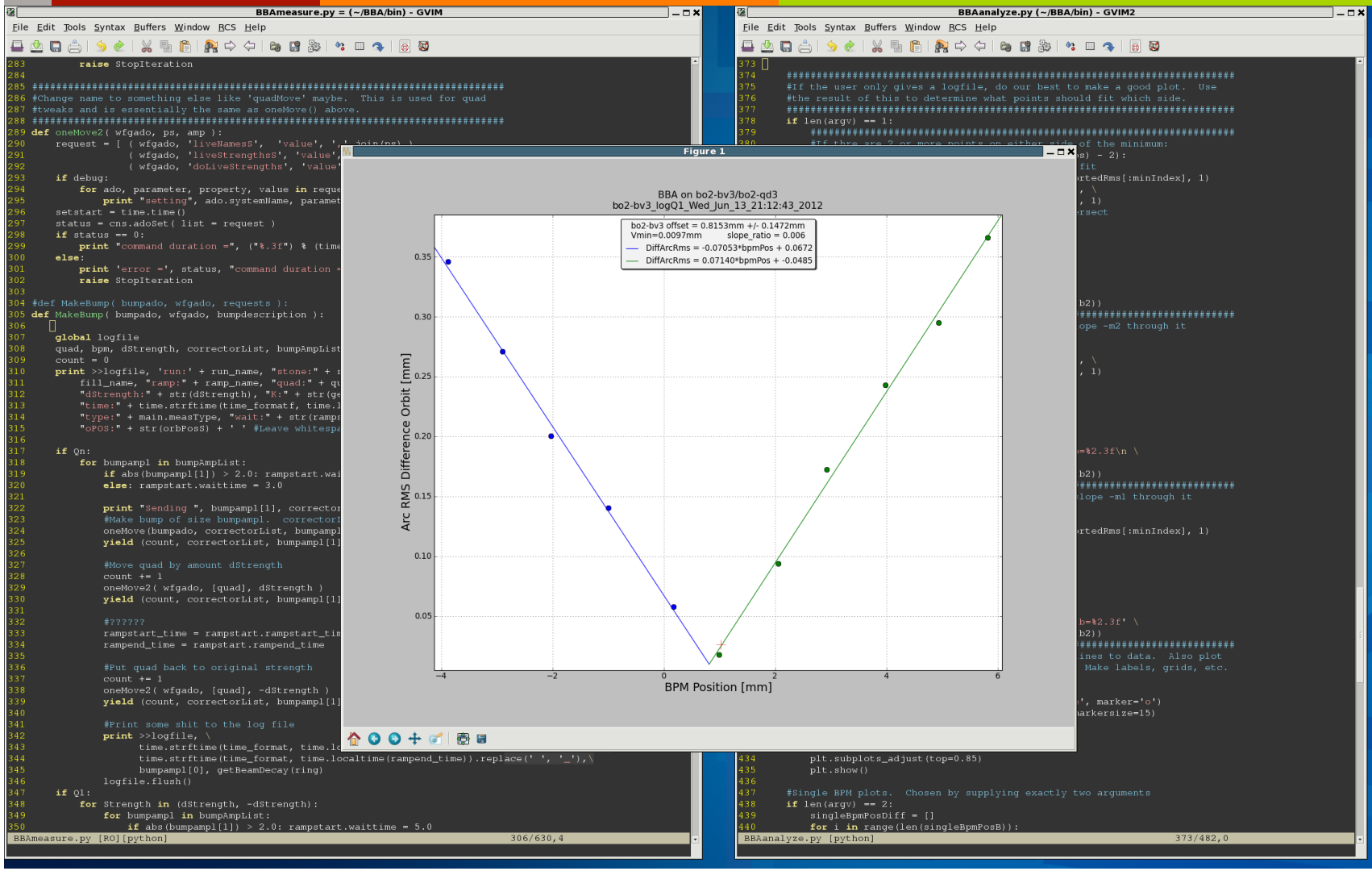
yo8-bh3_log1Q_Wed_Mar_21_18:30:04_...A/BBA-sessions/2012-03-21_APEX) - VIM

```
mcr@acnmcrbc6: /home/cfsb/mcr/ziegler/BBA-ScreenShot...   yo8-bh3_log1Q_Wed_Mar_21_18:30:04_...A/BBA-sessions/2...
 1 run:run_fy12 stone:cdevCommand: fill:16600 ramp:pp12b-v2 quad:yo8-qd3 bpm:yo8-bh3 dS
   trength:-0.001 wait:3.0
 2 Wed_Mar_21_18:30:07_2012 Wed_Mar_21_18:30:14_2012 -3.0
 3 Wed_Mar_21_18:30:14_2012 Wed_Mar_21_18:30:20_2012 -2.0
 4 Wed_Mar_21_18:30:20_2012 Wed_Mar_21_18:30:25_2012 -1.0
 5 Wed_Mar_21_18:30:25_2012 Wed_Mar_21_18:30:30_2012 0
 6 Wed_Mar_21_18:30:30_2012 Wed_Mar_21_18:30:35_2012 1.0
 7 Wed_Mar_21_18:30:35_2012 Wed_Mar_21_18:30:40_2012 2.0
 8 Wed_Mar_21_18:30:40_2012 Wed_Mar_21_18:30:45_2012 3.0
 9 Wed_Mar_21_18:30:45_2012 Wed_Mar_21_18:30:51_2012 0
10 Wed_Mar_21_18:30:53_2012 Wed_Mar_21_18:31:00_2012 -3.0
11 Wed_Mar_21_18:31:00_2012 Wed_Mar_21_18:31:05_2012 -2.0
12 Wed_Mar_21_18:31:05_2012 Wed_Mar_21_18:31:10_2012 -1.0
13 Wed_Mar_21_18:31:10_2012 Wed_Mar_21_18:31:15_2012 0
14 Wed_Mar_21_18:31:15_2012 Wed_Mar_21_18:31:20_2012 1.0
15 Wed_Mar_21_18:31:20_2012 Wed_Mar_21_18:31:25_2012 2.0
16 Wed_Mar_21_18:31:25_2012 Wed_Mar_21_18:31:30_2012 3.0
17 Wed_Mar_21_18:31:30_2012 Wed_Mar_21_18:31:36_2012 0
~
~
~
~
~
~
~
~
~
~
yo8-bh3_log1Q_Wed_Mar_21_18:30:04_2012                    1/17,1
```

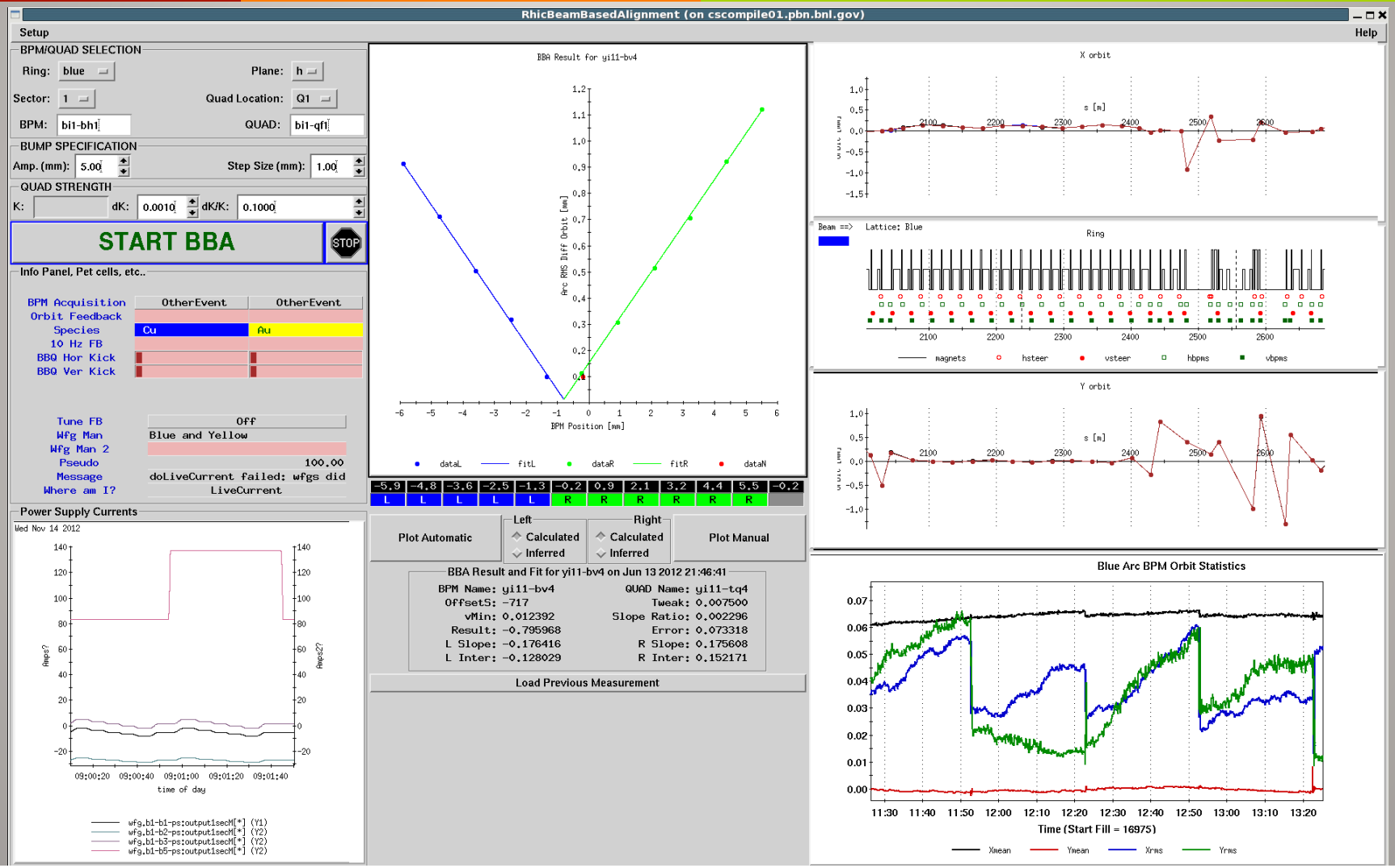
BBAanalyze.py (~/BBA/bin) - GVIM2

```
373
374     ##########################################################################
375     #If the user only gives a logfile, do our best to make a good plot.  Use
376     #the result of this to determine what points should fit which side.
377     ##########################################################################
378     if len(argv) == 1:
379         ##########################################################################
380         #If thre are 2 or more points on either side of the minimum:
381         if (minIndex >1) and minIndex < (len(sortedPos) - 2):
382             #Find y intercept and slope for plotting fit
383             m1, b1 = polyfit(sortedPos[:minIndex], sortedRms[:minIndex], 1)
384             m2, b2 = polyfit(sortedPos[minIndex + 1:], \
385                              sortedRms[minIndex + 1:], 1)
386             #find the x-value at which the lines intersect
387             intersect = (b2 - b1)/(m1 - m2)
388             #Plot the fits
427         print sortedPos
428         print sortedRms
429         plt.plot(sortedPos, sortedRms,linestyle='None', marker='o')
430         plt.plot(lastPointsPos, lastPointsRms, 'r+',markersize=15)
431         ax= plt.gca()
432         ax.xaxis.grid(True)
433         ax.yaxis.grid(True)
434         plt.subplots_adjust(top=0.85)
435         plt.show()
436
437     #Single BPM plots.  Chosen by supplying exactly two arguments
438     if len(argv) == 2:
439         singleBpmPosDiff = []
440         for i in range(len(singleBpmPosB)):
BBAanalyze.py [python]                                      373/482,0
```

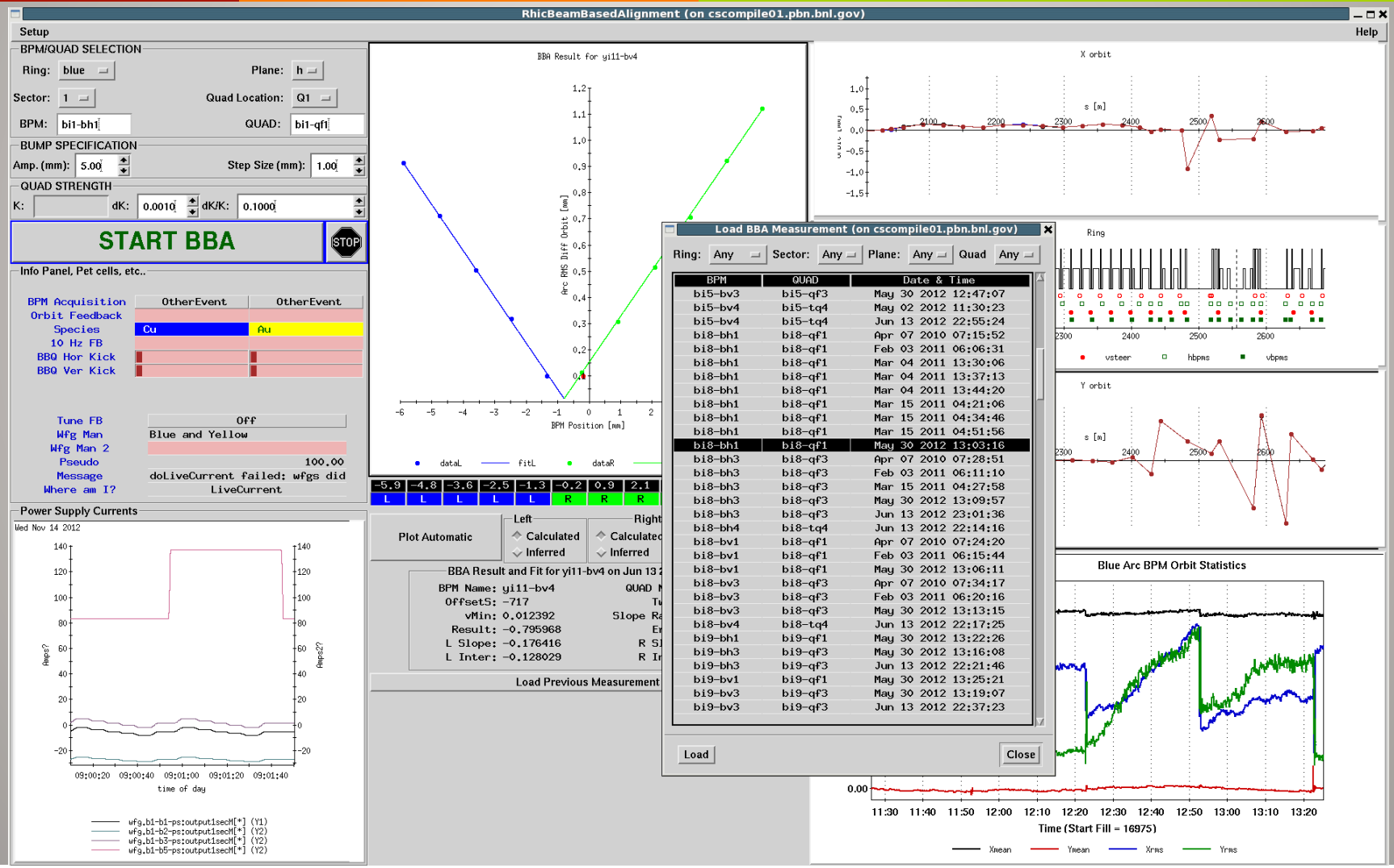
BBA Code: Present



BBA Code: Future



BBA Code: Future

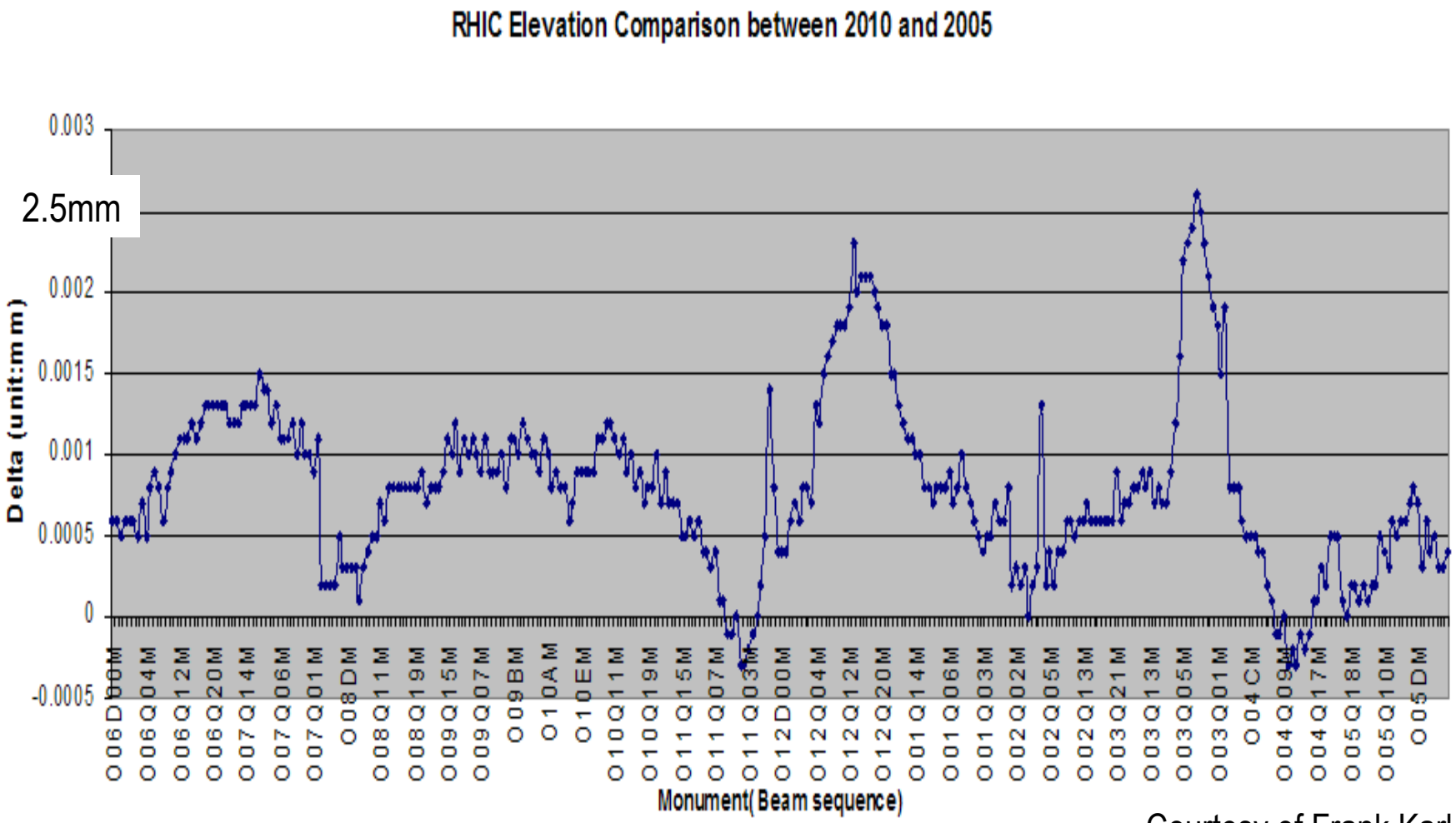


Summary of BBA Code

- Significant improvements in usability and reliability as a result of
  - Incorporation of bump manager
  - Rewriting of initial code with explicit emphasis on error handling, modularity, and usability
  - Designing measurement, analysis, and offset correction scripts in parallel
  - Adapting measurement techniques based on knowledge from previous measurements
- Stand-alone GUI application nearly ready for use
  - Measurement, Monitoring, & Analysis handled in one application
  - Knowledge from measurement database "BBA_meas" provides parameters for faster, safer, more accurate measurements
  - Element database "BBA_elem" allows BBA on the fly with no setup
  - Easily view/analyze previous measurements

RHIC Vertical Survey Results

Realignment was done except 6-7 o'clock region



Courtesy of Frank Karl