SKA PSS Folding Optimization

Document Number ..... 2018_06
Revision: ..... Revision 0.5
CIDL ID ..... REP_20
Author ..... C.Baffa, E.Giani
Date ..... 20 Dec 2018
Status ..... Draft
Classification ..... Unrestricted

Prepared By	Name	C.Baffa	Signature	
	Organisation	INAF OAA	Date	
	Name	E.Giani		
	Organisation	INAF OAA		
Reviewed By	Name			
	Organisation			
	Name			
	Organisation			
	Name			
	Organisation			

The Pulsar Search Sub-Element (PSS) of SKA Phase 1 is a critical digital-signal-processing component of the telescope that is required to search detected tied-array beams for pulsars and fast transients. This document contains an analysis of the parameter space of the folding optimization phase in the 3D mode and the test of an iterated 2D mode.

DOCUMENT HISTORY

Issue	Date	ECN	Change Description
1.0			

DOCUMENT SOFTWARE

Application		Version	Filename
Wordprocessing	MsWord		

ORGANISATION DETAILS

Designation	Name	Organisation
Primary Author and Responsible Organisation1	C.Baffa	INAF OAA
Contributors	E.Giani	INAF OAA

[1] The “Responsible Organisation” is responsible for ensuring that the IP Declaration in Appendix A is accurate and completed in accordance with the SKA IP Policy.

TABLE OF CONTENT

LIST OF TABLES

LIST OF FIGURES

LIST OF ACRONYMS AND ABBREVIATIONS

(prune this list to only what is referenced in this document, add more if necessary)

CDR	Critical Design Review
CSP	Central Signal Processor
DDD	Detailed Design Document
FLDO	FoLDing and Optimisation processing module
FPGA	Field Programmable Gate Array
GPU	General Processing Unit
INAF	National Institute for Astrophysics
I/O	Input/Output
IP	Intellectual Property
QA	Quality Assurance
QAP	Quality Assurance Plan
RFI	Radio Frequency Interference
RMP	Risk Management Plan
SKA	Square Kilometre Array
SKAO	SKA Organisation (or office)
TDTref	TDT reference folding code (by M.Keith)
TDT	Time Domain Team

1 INTRODUCTION

1.1 Purpose of Document

The purpose of this document is to analyse the parameter space of the optimization phase of the FLDO module as described in the relevant section of Detailed Design for the SKA Phase 1 CSP Pulsar search sub-element (RD1, section 4.3.1.18Scope of Document

1.2 Intended Audience

This document is expected to be used by the CSP PSS Sub-Element Consortium Engineering during the FLDO developing phase.

2 APPLICABLE AND REFERENCE DOCUMENTS

2.1 Applicable Documents

The following documents at their indicated revision form part of this document to the extent specified herein.

Table 2-1 Applicable Documents

Ref No	Document/Drawing Number	Document Title	Issue Number
AD1			
AD2			

2.2 Reference Documents

The following documents provide useful reference information associated with this document. These documents are to be used for information only. Changes to the date and/or revision number do not make this document out of date.

Table 2-2 Reference Documents

Ref No	Document/Drawing Number	Document Title	Issue Number
RD1	SKA-TEL-CSP-000008 NIP-PSS-DDD	SKA CSP Pulsar Search Sub-element Detailed Design Document (ED-4a)	
RD2			

3 FOLDING AND OPTIMIZATION

The FLDO is responsible for producing the periodicity search data products that form the output of the CSP-NIP-PSS sub-element. FLDO is a software/firmware module.

The FLDO performs the de-dispersion and binning of input data using the values specified for each candidate, and produces an optimized profile set for each one. Outputs from the FLDO are optimized parameters with the associated profiles.

The fine tuning of candidates parameters (optimization) is the last part of FLDO. This report explores the useful range of optimization in terms of scan time and candidate parameters. This optimization is performed by a simple brute force grid search, well suited for a parallel engine. The standard mode is performed on a 3D grid. The three axis of the grid are the period of the pulsar candidates (per), its dispersion measure (DM) and the derivative of the period, or acceleration (pdot).

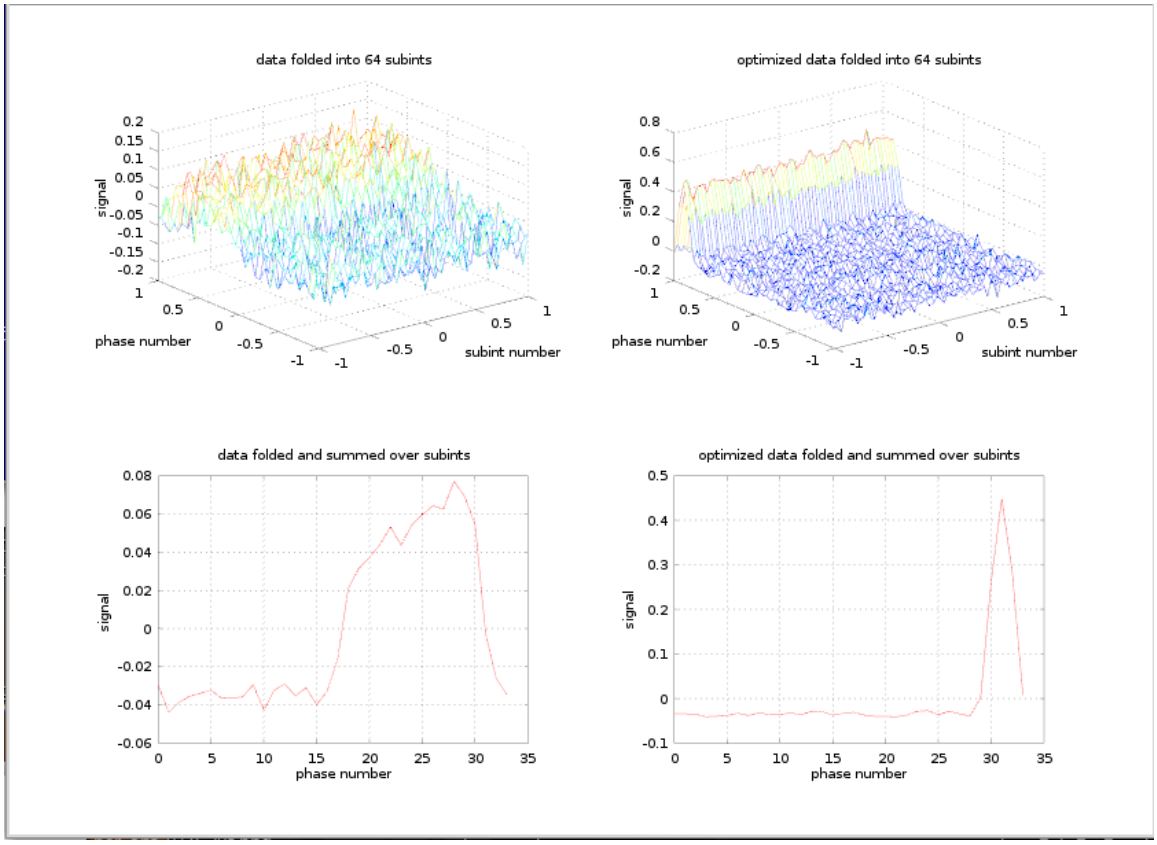
In section 5 are reported the results of an attempt to improve the optimization algorithm. Instead of a full 3D optimization search, a procedure to iteratively attempt few 2D optimizations is described. The final tests, comparing the performances of the 3D and of the 2D optimization procedure are also presented. The amount of time devoted to these test was not negligible, however a more extended test effort is needed. The planned extension will sample all parameter space available, but the most needed extension is on pdot space, now tested only on a small strip around zero.

4 OPTIMIZATION ANALYSIS – 3D PROCEDURE

4.1 Optimization procedure – 3D procedure

In the last step of FLDO, the candidate parameters are “Optimized”. The optimization procedure computes the effects on the computed S/N when small perturbations on all three candidate parameters are applied to the candidate parameters: period, DM and pdot. The procedure selects the perturbed parameter corresponding to the set with higher S/N in the three dimensions perturbation space. This optimization is performed by a simple brute force grid search, well suited for a parallel engine. On each variable (Per, DM, pdot) a fixed number of steps (Steps) are performed (see Figure 5-1, left panel). The work of optimization can be seen in Figure 4-1 where the input data are ‘aligned’ to have a well behaved profile.

Figure 4-1 A well behaved optimization. Data present in left panel are perturbet to obtain the aligned profiles in the right panel.



Some efforts have been devoted to the possible use of an “amoeba style”[2] optimization approach. The results were not good, probably because of the 'clumpiness' of the numerical function. Further development was devoted toward the research of the efficiency of a different schema of brute force grid search approach. The idea is to perform a two dimension space period-DM optimization, followed by a simple unidimensional search along acceleration. We present some preliminary results in chapter 5.

[2] Also called Nelder-Mead or downhill simplex maximization method.

We note the high S/N obtained with the 3D approach on 'empty' test vector or in situations very far from the synthetic pulsar signal present. We routinely have 'best' S/N values around 4. This is expected, as we search for the maximum value in a 'noisy' population of 8000 (20^3) samples.

The optimization phase, initially a small portion of FLDO total time, in the last iterations has assumed a larger and larger fraction of total time, due to improved efficiency of the folding portion. Hence we researched alternative approaches.

Table 4-1 The default parameter space of TDT reference implementation,

Default candidate value			
Variable	**Default value**	**Optimization range**	
Period (per)	1.1234 ms	19 steps * 0.0001e-3 s	
Dispersion Measure (DM)	10 pc/cm^3	19 steps * 1 pc/cm^3	
Acceleration (pdot)	0 ms/s	No optimization	
Default input data parameters			
Variable	**Default value**	**Variable**	**Default value**
Duration	15s	Channels interval	-0.075Mhz
Sample interval	50µs	Bit per sample	8
Frequency channels	256	Numerical format	unsigned

4.2 Starting optimization range

To start with a working baseline, the original conditions of the Michael Keith's TDT reference folding code was assumed[3]. The default parameter space is summarized in Table 4-1. Those values refer to a very short demonstrative integration.

The initial basic values used for testing the GPU version of FLDO are shown in Table 4-2. These values are derived from the TDT reference implementation (henceforth TDTref), adapted for the longer data sample.

After few trials it appeared clearly the Table 4-2 values were not able to adapt to different situations. So we derived theoretically the variations of these limits respect to integration time, dispersion measure and integration periods. The dependences assumed are listed in Table 4-3.

[3] See https://gitlab.com/SKA-TDT/tdt-reference-implementations/tree/master/folding_optimsation

Table 4-2 The default parameter space of GPU's FLDO implementation

Default candidate value			
Variable	**Default value**	**Optimization range**	
Period (per)	1.1234 ms	19 steps * 0.00002e ms	
Dispersion Measure (DM)	10 pc/cm^3	19 steps * 0.10 pc/cm^3	
Acceleration (pdot)	0 ms/s	19 steps * 0.02e^{-7} ms/s	
Default input data parameters			
Variable	**Default value**	**Variable**	**Default value**
Duration	30-530s (up to 2^{23} samples)	Channels interval	-0.075Mhz
Sample interval	64μs	Bit per sample	8
Frequency channels	4096	Numerical format	unsigned

Table 4-3 Adopted variable optimization dependence on scan duration and candidate parameters

Variation of optimization intervals		
Variable	**Interval proportional to**	**Adopted value**
Period (per)	per/duration	2e-4 * per / duration s
Dispersion Measure (DM)	per	80 * per pc/cm^3
Acceleration (pdot)	per/duration/duration	2e-3 * per / duration / duration s
# of optimization steps (Steps)	N.A.	20

4.3 The optimization convergence interval

Many tests aimed at finding the candidate parameter intervals in which the optimization procedure converges to the original values were performed. The basic procedure was to use a standard data vector, containing noise and a simulated pulsar with parameters listed in Table 4-1. On this data vector we performed a folding using the parameter values of a set of artificial candidates, with parameters very near the exact, original values. The dimension of the three dimensional space containing the candidates which can bring to the real values during optimization is the aim of present research.

This analysis hit some difficulties. The first, and unavoidable, is the 'clumpiness' of the data, coming from a *perturbation* problem.

A second problem is the three dimensionality of the optimization space, which can bring to results as in Figure 4-2 of obviously difficult interpretation. A much better understanding can be obtained using a search along only one of the axes. In Figure 4-3 the effects of the periodicity of the input

data are evident. In these 1D optimization graphs on the X axis are reported the candidate input values, while on the Y axis there are the optimized (output) values.

Figure 4-2 A Three dimensional optimization attempt. It evident the complexity of the general schema.

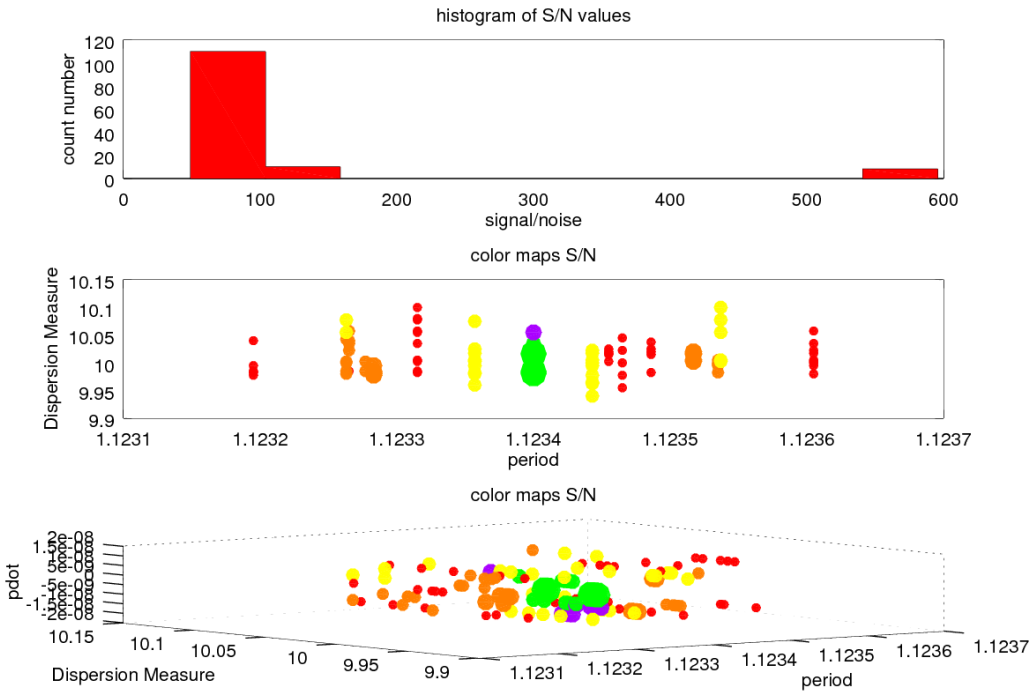
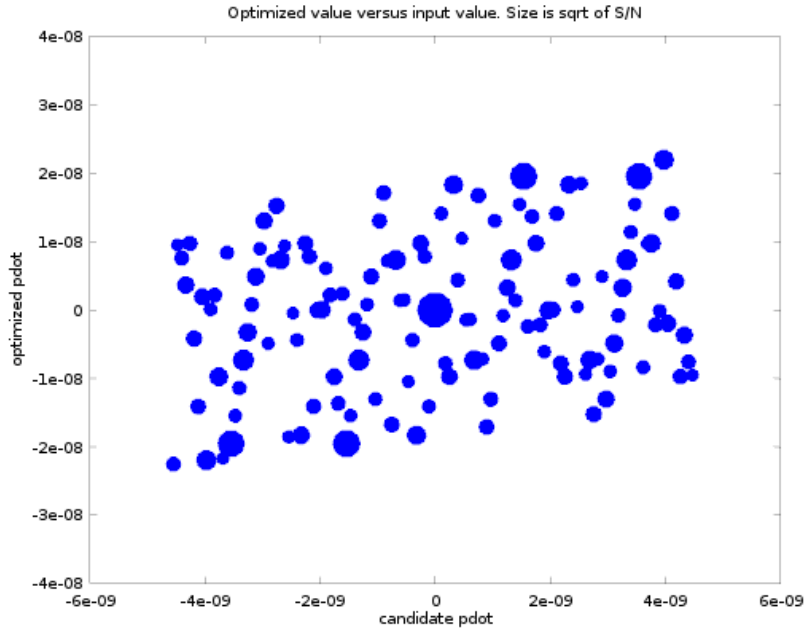
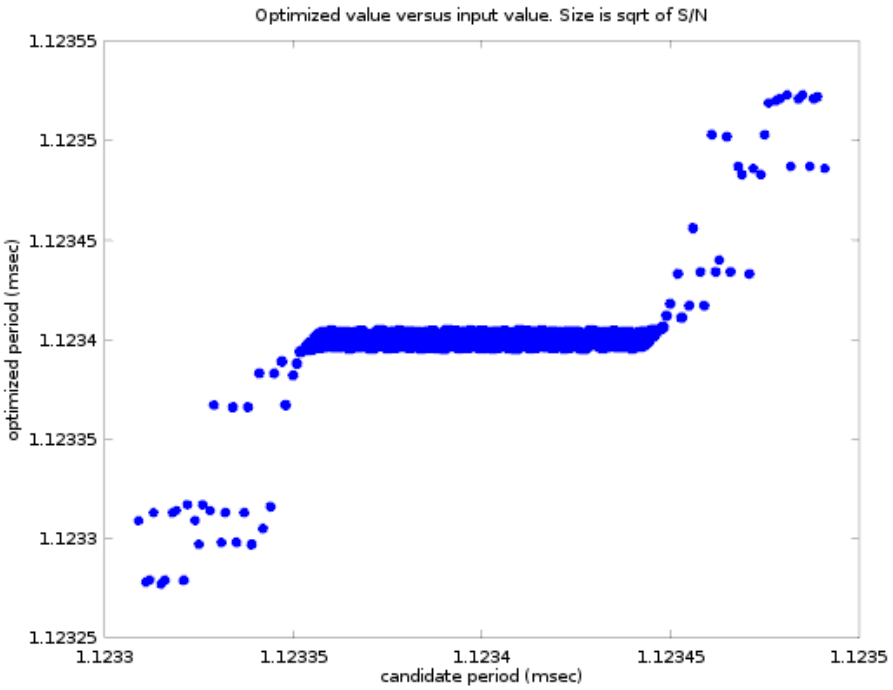


Figure 4-3 Optimization along pdot only. The effects of the periodicity of input data are evident.



From the previous example it can be concluded the convergence space can be analysed a dimension a time, to carefully determine the useful interval and the spacing of the search grid. In this situation, a 1D optimization graph (such as the Figure 4-3) should appear as a broad horizontal line (the convergence interval) ad some scattered points outside the convergence range. An example is presented in Figure 4-4.

Figure 4-4. 1D - Optimization graph. It is evident the convergence interval.



4.4 The convergence tests – default parameters

Upon identified the convergence interval, a verification of the formulae in Table 4-3 has been performed. Most of the verification work has been done on input data with a synthetic pulsar with the default parameters cited above. More work is needed with synthetic pulsar with different characteristics. This will be the object of successive tests (see 4.5).

The results are reported in the different panels of few figures. In particular:

1. Figure 4-5 shows the different 1D plot of the effect of scan duration on period optimization. From upper left, going clockwise, we have 30s, 60s, 530s and 120s scan time. It is evident the convergence interval becomes narrower as scan time increase. It is also visible, mainly on longer scans, the effect of search granularity. This last effect is expected to be eased with the future recursive grid optimization.
2. Figure 4-6 shows that scan duration has no effect on dispersion measure optimization. Upper panels have 30s scan time, the right one is an enlargement of left one to show the granularity of results. Lower panel have 530s scan time. The upper right panel is an enlargement of left one to show the granularity of results. The lower right panel show the effects of a grid search coarser than that used on left one. It also shows the larger convergence interval.
3. Figure 4-7 shows the different 1D plot of the effect of scan duration on period derivative optimization. From upper left, going clockwise, we have 30s, 60s, 530s and 120s scan time. The convergence interval becomes narrower much faster than in the period case.

Figure 4-5 Effects of scan duration on period optimization. Upper left panel 30s scan, upper right 60s scan. Lower left 120s scan lower right 530s scan.

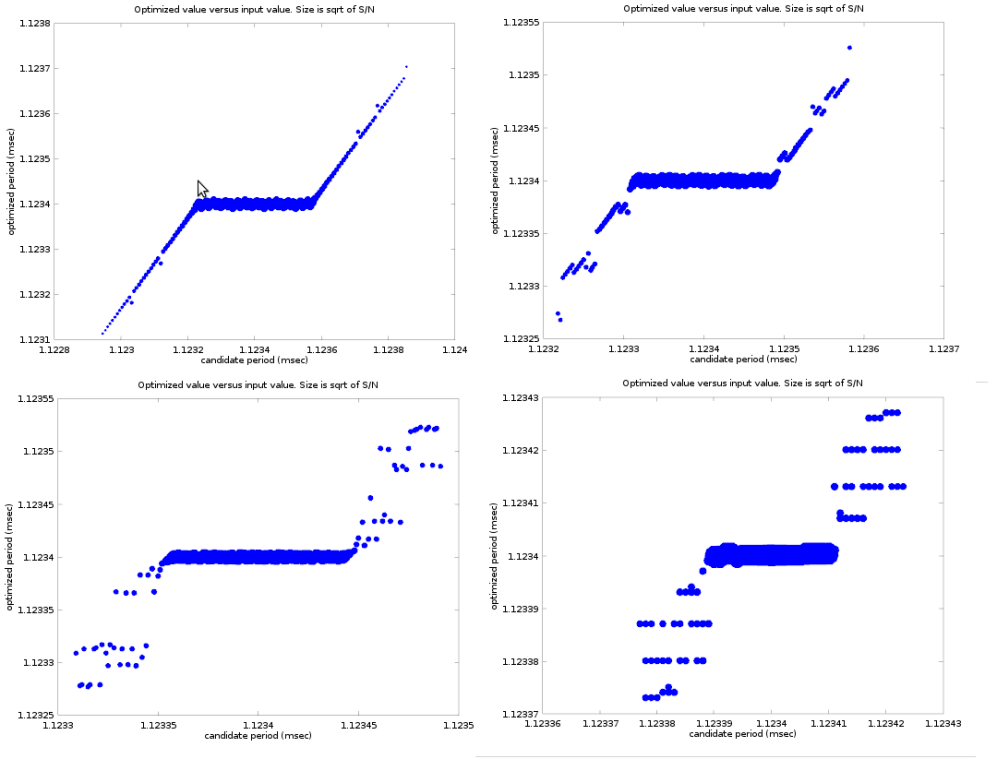


Figure 4-6 No effects of scan duration on DM optimization. Upper panels have 30s scan. Lower panels have 530s scan. The upper right panel is an enlargement of left one to show the granularity of results. The lower right panel show the effects of a grid search coarser than that used on left one.

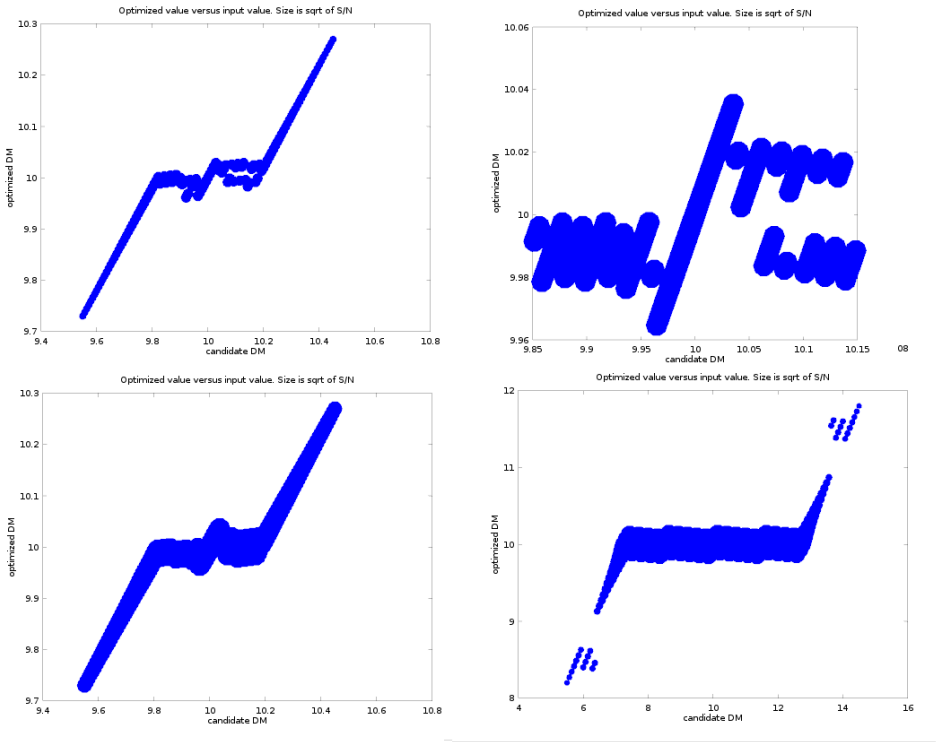
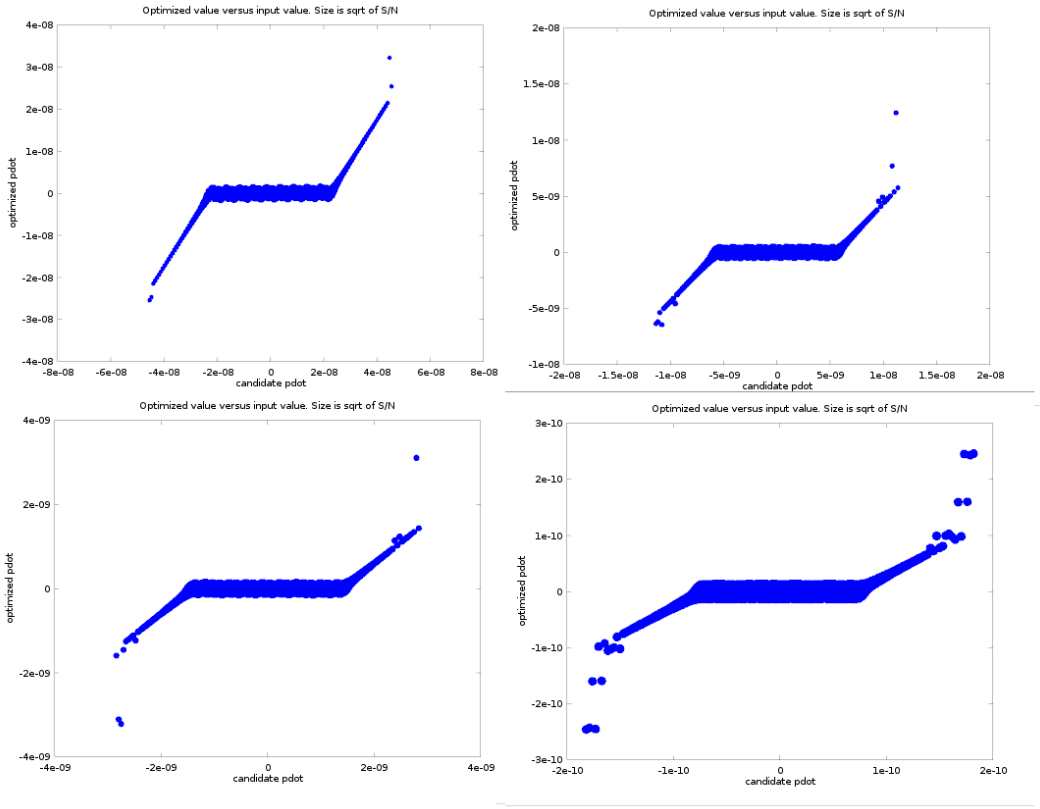


Figure 4-7 Effects of scan duration on period derivative (pdot) optimization. Upper left panel 30s scan, upper right 60s scan. Lower left 120s scan, lower right 530s scan.



4.5 The convergence tests – different parameters

The convergence interval needs to be verified also for candidates with parameters different from the standard one defined in Table 4-2. These tests, exploring the 3-dimestion candidates' parameter space, will give more confidence on the formulae in Table 4-3.

Some of the test results are documented in the next figures. In particular:

1. Figure 4-8 shows, by means of the different 1D plot of DM optimization, the effect of different DM and scan duration on period optimization. Left panel shows the results for a scan of 60s and a DM of 100, while the right panel shows the results for a scan of 530s and a DM of 100. It is visible the effect of increased S/N, due to longer scan time, while other proprieties appears to be unchanged.

2. Figure 4-9 shows the effect of a larger DM value and some details on S/N results. Left panel shows the 1D plot result for period optimization, with the expected convergence interval and the usual S/N trend. The right panel shows the histogram of the S/N of the output candidates. The shape of the histogram shows the expected distribution for the two different candidate populations: the candidates in inner space converging to the correct results and accounted by the right peak, and the broader distribution of farther out candidates, outside converging space, distributed on the left bins.

Figure 4-8 Effects of different DM on optimization. Left panel 60s scan and DM of 100, right panel 530s scan and DM of 100.

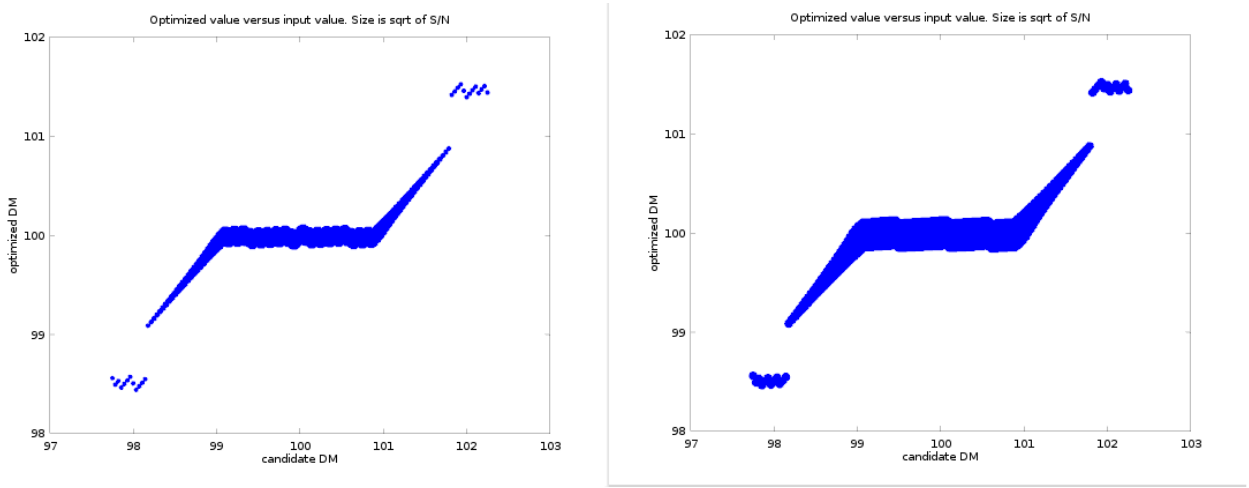
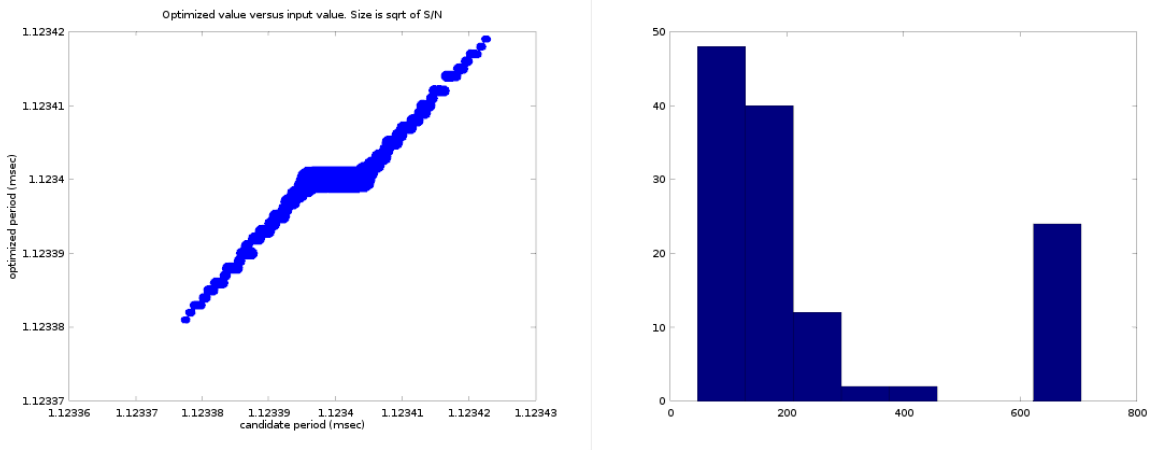


Figure 4-9 Left panel shows the results for a simulated pulsar with 1.1234ms period, 500 DM value and 530 s scan time. Right panel shows the histogram of resulting S/N.



5 OPTIMIZATION TESTS - 2D VERSION

As stated in previous section, the standard optimization procedure computes the effects on the final S/N when small perturbations are applied on all three candidate parameters. The usual approach explores a three dimensions perturbation space and selects the perturbed parameter set corresponding to the higher S/N. This brute force grid search is performed in a single step, and its computational complexity is proportional to the cube of optimization interval number ($Steps^3$).

We explored the possibility to reduce this computation time by optimizing two variables a time. The resulting computation complexity is proportional to the square of interval number ($3xSteps^2$). The resulting time reduction is thus somehow less than a factor of 7.

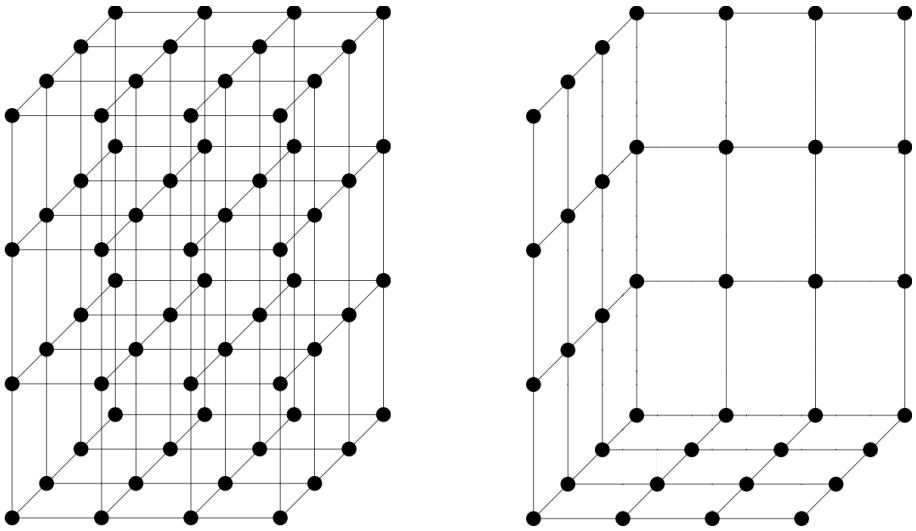


Figure 5-1 Left panel, three dimensions single pass optimization space. Right panel three passes two dimensions optimization spaces.

The procedure we have devised is composed by three phases:

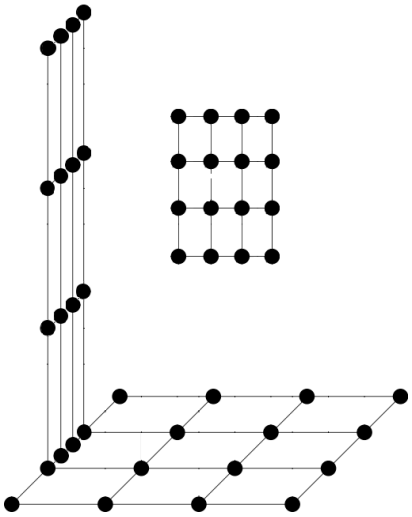
1. Optimization on the period-DM plane
2. Optimization on the DM-pdot plane
3. Optimization on the period-pdot plane

Each step uses, as starting point, the parameters values obtained from the previous optimization phase. To further exploit the advantage of having two successive optimizations along each axis, we reduced the optimization span on the second iteration. We have the following schema (see Figure 5-2):

1. Optimization on the [full period]-[full DM] plane
2. Optimization on the [reduced DM]-[full pdot] plane
3. Optimization on the [reduced period]-[reduced pdot] plane

This procedure while still requiring only 1/7 of the 3D computation time gives similar signal to noise and a slight larger convergence space. We have performed a number of tests comparing the 3D and the 2D procedures results. The amount of time devoted to these test was not negligible, however a more extended test effort is needed. The planned extension will sample all parameter space available, but the most needed extension is on pdot space, now tested only on a small strip around zero.

Figure 5-2 Progressive shrink of explored spaced during multiple 2D optimization. The position of each optimization plane is drawn in an arbitrary position, as each step starts from the results of the previous one.



5.1 2D and 3D approach comparison

We performed a relatively small number of tests to compare the 2D and 3D optimization algorithms. We executed a side to side comparison. The results are comparable. The optimized values coming from the 2D approach are as good as the 3D ones. The resulting S/N is normally slightly minor than the 3D case, about few percent, but there are cases where is similar or slightly superior.

Figure 5-3 Left panel show the results of a 3D optimization procedure over a 530s scan with candidates distributed along period axis. Right panel shows the 2D optimization results of the same scan and candidate set.

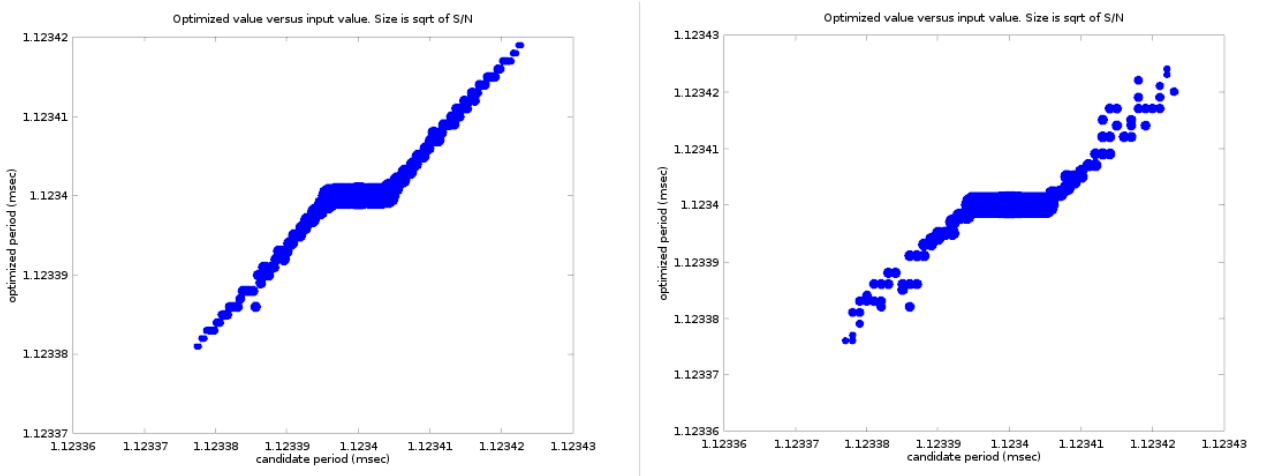


Figure 5-4 Left panel show the results of a 3D optimization procedure over a 530s scan with candidates distributed along DM axis. Right panel shows the 2D optimization results of the same scan and candidate set.

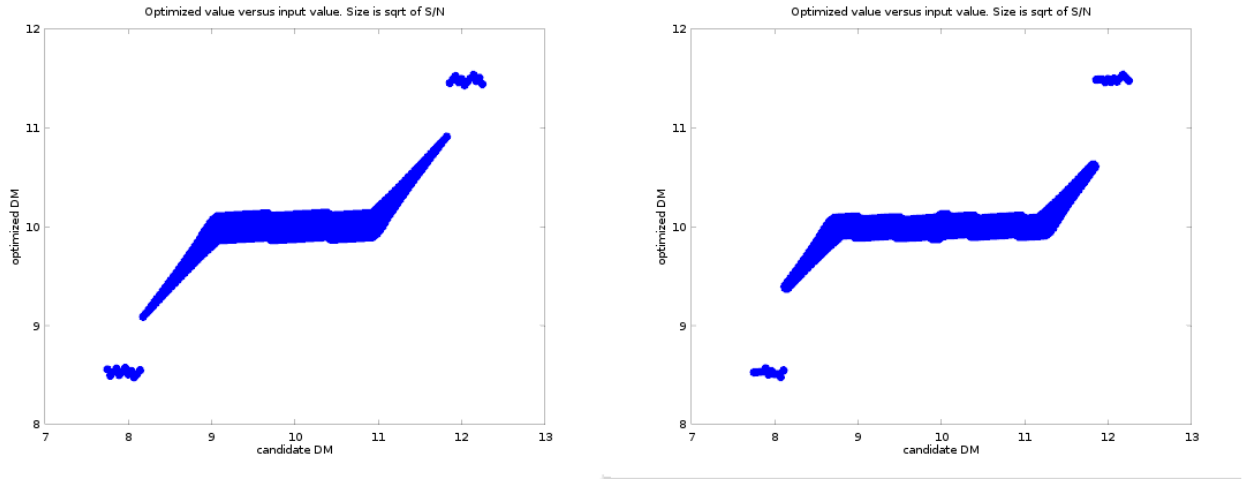
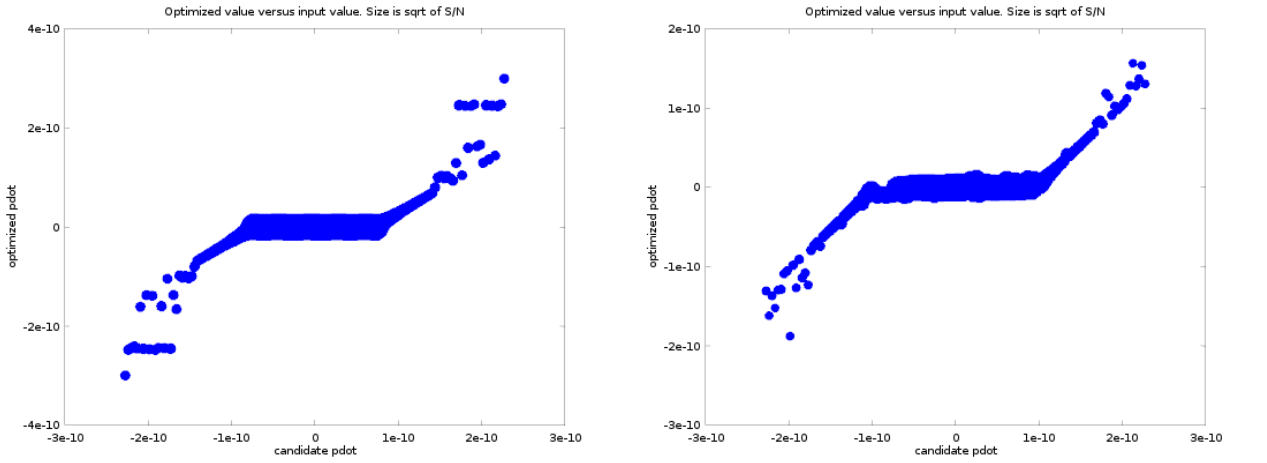


Figure 5-5 Left panel show the results of a 3D optimization procedure over a 530s scan with candidates distributed along pdot axis. Right panel shows the 2D optimization results of the same scan and candidate set.



We present here few 2D-3D optimization comparisons.

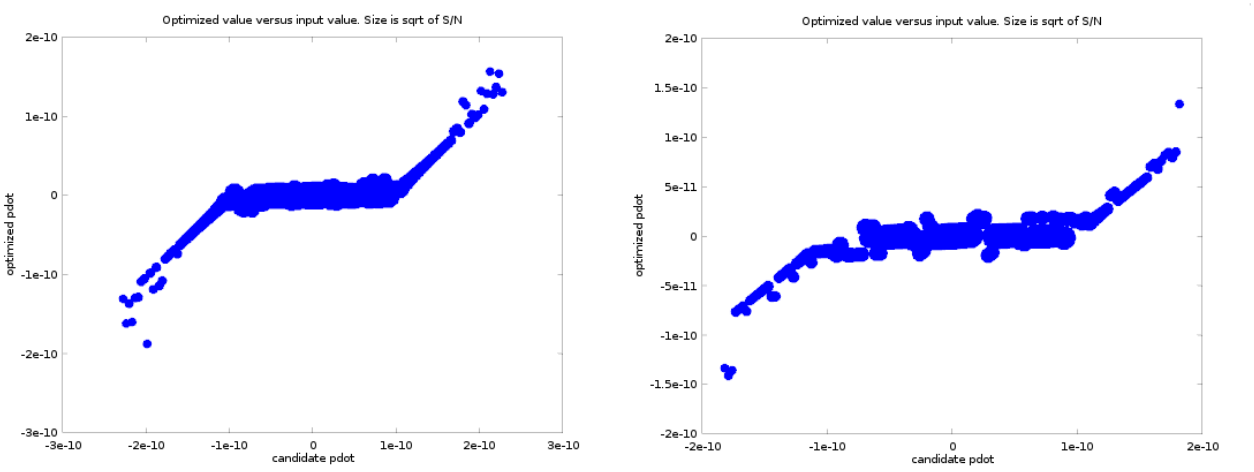
1. Figure 5-3 shows the results of optimization of a set of 128 candidates uniformly distributed along the period axis, applied on a test vector containing our standard pulsar values (see Table 4-2). The convergence space is the same, and the value obtained are of the same quality (<per>=1.1234ms, $\sigma(per)=10^{-5}$).
2. Figure 5-4 shows the results of optimization of a set of 128 candidates uniformly distributed along the DM axis, applied on a test vector containing our standard pulsar values (see Table 4-2). The convergence space is slight better in the 2D case, and the value obtained are of the similar quality (<DM>=10.00 pc/cm^3, σ(per-3D)=0.68, σ(per-2D)=0.72).
3. Figure 5-4 shows the results of optimization of a set of 128 candidates uniformly distributed along the pdot axis, applied on a test vector containing our standard pulsar values (see Table 4-2). The convergence space is similar, and the value obtained are of the similar quality (<pdot>≈-10^{-12}m/sec^2, $\sigma(pdot\text{-}3D)=1.2\times10^{-10}$, $\sigma(pdot\text{-}2D)=6\times10^{-11}$).

5.2 Failure of a mixed 2D-1D approach

To test the possibility to obtain a greater performance gain, we tested also the possibility to have a mixed 2D-1D optimization. The idea was to start with a 2D optimization along the period-DM plane followed by an optimization along the pdot axis. The results were not at par with a full 3D optimization mainly in terms of S/N and convergence. There are many candidates really near the 'correct' value which cannot get a suitable S/N value. Also using an iterative approach to use two successive applications of the 2D-1D optimization (as documented in Figure 5-6) was not satisfactory.

The speed gain of the 2D-1D over the 2D approach is marginal (a speedup of a factor of 10 against a factor of 7). So it was decided to drop the work toward this poor promising approach and to concentrate in the development of the 2D approach.

Figure 5-6 Left panel shows a successful 2D optimization. Right panel shows the use of two successive applications of the 2D-1D optimization. There are many candidates really near the 'correct' value which cannot get a suitable S/N value.



6 CONCLUSIONS

We have presented a slightly improved FLDO optimization algorithm. We got a seven fold computing time improvement. However the optimization phase is only a portion of the total FLDO processing. The fraction of time spent in optimization (for a 530s observation) can range from 4-5% (on a GTX 1080) to more than 10% (on a P100).