Tunable Kernel-Nulling for direct detection of exoplanets

1. Calibration and performance

V. Foriel[1], F. Martinache[1], D. Mary[1] and R. Laugier[2]

[1] Université Côte d'Azur, Observatoire de la Côte d'Azur Nice, CNRS, Laboratoire Lagrange, Nice, France
[2] KU Leuven university, Leuven, Belgium

ABSTRACT

Context. Lorem ipsum
Aims. Lorem ipsum
Methods. Lorem ipsum
Results. Lorem ipsum
Conclusions. Lorem ipsum

Key words. Lorem ipsum

1. Introduction

1. Nulling interferometry
2. Kernel nulling
3. Integrated optics & phase shifters

2. Materials and conditions

1. VLTI/ASGARD (/NOTT?)
2. Integrated optics & phase shifters
3. Studied architecture
4. Observation conditions (Vegga-like star, noise etc.)

3. System modelisation & calibration

1. Calibration methods (Fig 4 & 5)
    Genertic Algorithm
    Input obstruction
    Machine Leaning?

3.1. Algorithms

In order to calibrate the component, two algorithms have been developed. The first is based on the principle of a genetic algorithm, the second on the principle of input obstruction. Both algorithms have been implemented in Python and are based on the same principle: we seek to optimize the component's performance by introducing phase variations on each phase shifter. The difference between the two algorithms lies in the way these phase variations are introduced and the metrics used to evaluate the component's performances. We will present both methods and discuss their advantages and limitations.

3.1.1. Genetic

The straightforward approach is based on the principle of deterministic genetic algorithm, which smply consist in successively introducing a phase variation $\Delta\phi$ on each phase shifter φ_n and looking at how it improve or decrease the system performances according to some metrics.

This calibration process is performed using a point source that simulates a single star. The goal is to nullify its signal on the dark outputs. However, since kernels are built using the difference of two dark outputs, it is possible to obtain very good kernel-nulls without necessarily having well nullified the star's light. To address this issue, we define two different metrics, one measuring the average of kernel-null depth M_K that we want to minimize, the other measuring the intensity obtained on the bright output of the component M_B that we want to maximize (which, by construction, minimize the total dark output intensity).

Given our component's architecture, we can associate the metric M_B with each phase shifter that affects the bright output, namely $\varphi_{1\rightarrow5}$ and φ_7 that we will call φ_b. All other phase shifters are then associated with the metric M_K (cf. Algorithme 1 line 3-4) and will be called φ_k. Thus, we are searching for the optimal phase shifts $\varphi_n^* = \varphi_k^* \cup \varphi_b^*$ such as:

$$\varphi_k^* = \arg\max_{\varphi_k}\left(M_K(\varphi_k)\right) \tag{1}$$

$$\varphi_b^* = \arg\min_{\varphi_b}\left(M_B(\varphi_b)\right) \tag{2}$$

where

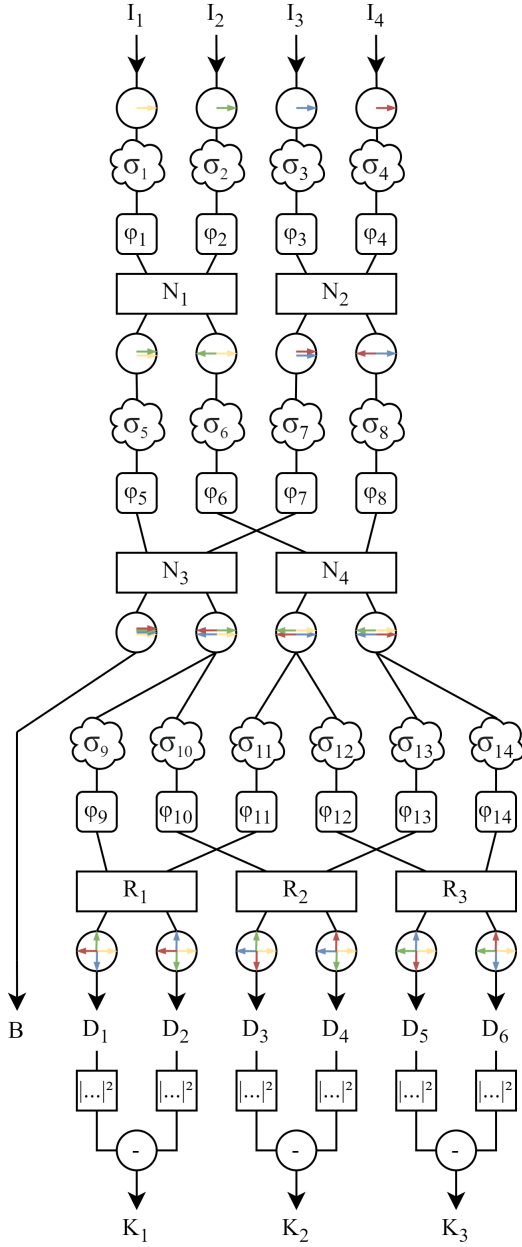$$M_K(\varphi_k) = \frac{1}{3}\sum_{n=1}^{3}|K_n(\varphi_k)| \tag{3}$$

Fig. 1. A model of our Kernel-Nuller architecture. It takes 4 input electric fields I_n that are supposed to be co-phased and have the same amplitude. It outputs one "bright" electric field B that should contain all the on-axis light and 6 "dark" electric fields D_n where the on-axis light is nulled. Additionally, we introduce 3 "kernel" outputs resulting from the difference of two dark output intensities. The interferometric operations (nulling N_n and cross-recombination R_n) are performed by MMI presented in section 2.2. In this model, the phase perturbations σ_n induced by manufacturing errors are associated with a phase offset on each phase shifter φ_n such that $\Phi_n = \varphi_n + \sigma_n$. The circles with colored arrows show, at different steps, the phase of each input signal we should have under ideal conditions. Thus, the goal of the calibration is to find the correct phase φ_n to inject on each phase shifter to compensate for these perturbations.

$$M_B(\varphi_b) = |B(\varphi_b)|^2 \tag{4}$$

$$\varphi_k = \varphi_{\{6,8,9,10,11,12\}} \tag{5}$$

$$\varphi_b = \varphi_{\{1,2,3,4,5,7\}} \tag{6}$$

The genetic algorithm is then structured as follows:

1. Initialization: we initialize the phases of each phase shifter φ_n to 0. (cf. Algorithme 1 line 1)

2. Mutation: we consider two phase variations of $+\Delta\phi$ and $-\Delta\phi$, successively on each phase shifter φ_n, always in the same order (cf. Algorithme 1 line 6-10).

3. Selection: we retain the configuration that most improves the metric associated with that phase shifter (cf. Algorithme 1 line 11-25).

   If $n \in [1,5] \cup 7$, we keep the variation that most increased metric M_B so we redirect the entire star flux to the bright output (cf. Algorithme 1 line 11-17).

   If $n \in 6 \cup [8,14]$, we keep the variation that most reduced metric M_K so we symmetrize dark outputs pairs to enhance kernel-nulls depth (cf. Algorithme 1 line 18-24).

4. Convergence: we repeat steps 2 and 3, reducing the value of $\Delta\phi$ by a factor of $\beta \in [0.5, 1[$ each time. A β factor close to 1 allows for slow but more precise convergence, whereas a β factor of 0.5 will yield the fastest convergence but may lead to an under-optimal solution (cf. Algorithme 1 line 5, 27).

5. Stopping point: we stop when the phase variation $\Delta\phi$ introduced is smaller than the uncertainty in the injected phase, which must be determined through prior characterization of the phase shifters (cf. Algorithme 1 line 5).

This algorithm has the advantages of being very easy to adapt to other Kernel-Nuller architectures and does not require any moving parts during the calibration process, allowing for fast and automated calibration. However, it suffers from a limitation related to photon noise. Specifically, the phase to inject using the shifters near the outputs depend on the phase injected in the previous layers. Thus, the calibration of the final shifters can only be finely tuned once the initial shifters have been calibrated, which implies very little flux on the dark outputs during calibration and therefore a greater sensitivity to photon noise. Numerically, even accounting for photon noise, it is possible to achieve kernel-null depths on the order of 10^{-8}. [ADD LABORATORY RESULTS]. Another disadvantage of this algorithm is that, for some given initial conditions, the convergence can be suboptimal and there is no way to know when this suboptimal solution happen. The frequency of these suboptimal cases decreases when β is close to 1, which implies a slower convergence.

Algorithme 1 : Straightforward approach (genetic)

Inputs : ε: minimal phase increment
β: phase variation reduction factor
($\beta \in [0.5, 1[$)
Output : φ: optimized phase shifts vector

---

1 $\varphi \leftarrow \mathbf{0}$;   // Initialize phase shifts to 0
2 $\Delta\varphi \leftarrow \lambda/4$;   // Initial phase variation
3 **while** $\Delta\phi > \varepsilon$ **do**
4   **for** $n \leftarrow 1$ **to** 14 **do**
5         $\varphi^+ \leftarrow \varphi$;
6         $\varphi_n^+ \leftarrow \varphi_n + \Delta\varphi$;
7         $\varphi^- \leftarrow \varphi$;
8         $\varphi_n^- \leftarrow \varphi_n - \Delta\varphi$;
9   **if** $n \in \varphi_b$ **then**
10   **if** $M_B(\varphi^+) > M_B(\varphi)$ **then**
11                $\varphi \leftarrow \varphi^+$;
12   **end**
13   **if** $M_B(\varphi^-) > M_B(\varphi)$ **then**
14                $\varphi \leftarrow \varphi^-$;
15   **end**
16   **else if** $n \in \varphi_k$ **then**
17   **if** $M_K(\varphi^+) < M_K(\varphi)$ **then**
18                $\varphi \leftarrow \varphi^+$;
19   **end**
20   **if** $M_K(\varphi^-) < M_K(\varphi)$ **then**
21                $\varphi \leftarrow \varphi^-$;
22   **end**
23   **end**
24   **end**
25     $\Delta\varphi \leftarrow \beta\Delta\varphi$;   // Reduce phase variation
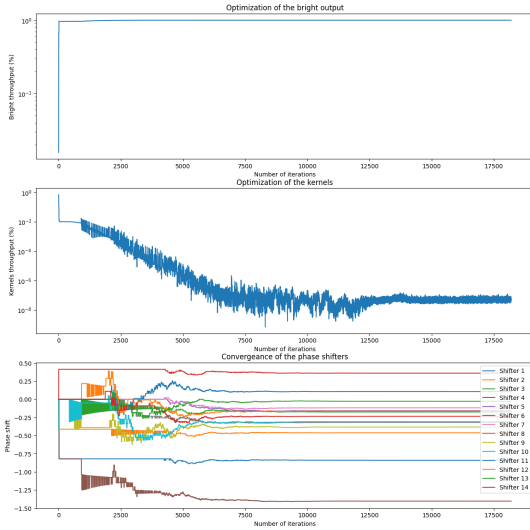26 **end**
27 **return** φ

Fig. 2. Calibration using genetic algorithm [AJOUTER DESCRIPTION]

3.1.2. Input obstruction

To overcome the limitations of the genetic algorithm, another calibration method has been implemented. This method is based on the principle of obstructing the component's inputs. We still inject a point source into the component but this time we obstruct 2 of its inputs. We then focus on one of its outputs whose transfer function is greatly simplified, to the point of having only one parameter (phase shifter) influencing this output.

The algorithm is constructed as follows:

1. We obstruct inputs 3 and 4 then seek to maximize the bright output. Given that we are insensitive to the global phase, we can use the electric field phase of the first input as a reference phase. Thus, the bright output can be maximized by only adjusting phase shifter φ_2.
2. We obstruct inputs 1 and 2 then seek to maximize the bright output, this time by adjusting phase shifter φ_4. We temporarily use the electric field phase of the third input as a reference phase.
3. We obstruct inputs 2 and 4 then again seek to maximize the bright output. This time, we adjust phase shifter φ_3, which will correspond to the phase difference at the output of the first layer of Nullers (Figure 1). We then add the phase introduced on φ_3 to that introduced on φ_4 to change the common phase of I_3 and I_4.
4. Still obstructing I_2 and I_4, we seek to minimize dark outputs 3 and 4 which are supposed to have inputs I_1 and I_3 in perfect phase opposition. For this, we adjust phase shifter φ_8.
5. We obstruct I_3 and I_4 and seek to minimize kernel 3. Given that the delays injected so far redirect the flux towards the bright output, we introduce a phase $+\frac{\pi}{2}$ on φ_2, thus transforming the bright output into a dark output and the dark outputs associated with kernel 3 into bright outputs. We then adjust phase shifter $\varphi_1 0$ to symmetrize the dark outputs (which we observe through the nulling of the kernel despite the presence of flux).
6. Similarly, we obstruct I_2 and I_4 and seek to minimize kernel 2 by introducing a phase $+\frac{\pi}{2}$ on φ_3. We then adjust phase shifter $\varphi_1 0$.
7. Finally, we obstruct I_2 and I_3 and seek to minimize kernel 1 by introducing a phase $+\frac{\pi}{2}$ on φ_4. We then adjust phase shifter φ_9.

For each step, the correct phase to inject on the considered phase shifter can be obtained either dichotomously or formally by performing a series of measurements. Indeed, for each step, the problem simplifies sufficiently to be solved analytically. For example, for the first step, we seek to maximize the bright output B which can then be written as:

$$B = \left| \left(a_1 e^{i(\theta_1 + \sigma_1 + \phi_1)} + a_2 e^{i(\theta_2 + \sigma_2 + \phi_2)} \right) e^{i(\sigma_5 + \phi_5)} \right|^2 \tag{7}$$

Where a_n and θ_n represent the amplitude and phase of the input signals respectively. σ_n corresponds to the (unknown) phase perturbation associated with shifter n and ϕ_n is the phase that we voluntarily inject via the shifter to try to compensate for this perturbation.

Since calibration is done in the laboratory, we can assume a total intensity fixed at 1 (arbitrary unit) and that each input receives the same flux, i.e., $a_1 = a_2 = 1/\sqrt{2}$, and perfectly in phase, i.e., $\theta_1 = \theta_2 = \theta$. Since we only have access to signal intensity, we are insensitive to the global phase, which allows us to simplify the previous equation:

$$B = \frac{1}{2} \left| e^{i(\sigma_1 + \phi_1)} + e^{i(\sigma_2 + \phi_2)} \right|^2 \tag{8}$$

By maximizing B, we should then find 1 which implies that

$$\sigma_1 + \phi_1 = \sigma_2 + \phi_2 \tag{9}$$

We can use ϕ_1 as a reference (global phase) and thus fix it to 0, which then gives

$$\phi_2 = \sigma_1 - \sigma_2 \tag{10}$$

We can take different measurements of B at fixed ϕ_2 and deduce σ_1 and σ_2 by solving a system of equations.

This second calibration method has the advantage of not suffering from photon noise sensitivity since we ensure at each step that we have flux on the outputs of interest. However, it is adapted to our architecture and must be completely rethought to be adapted to another Kernel-Nuller architecture. Moreover, it requires the action of moving parts to obstruct the component's inputs, which can be a disadvantage for automated calibration.

Using this method, we numerically obtain a null depth also on the order of 10^{-8}. Thanks to the successive simplification of the problem, we find ourselves in a case where convergence to an optimal solution is assured. [ADD LABORATORY RESULTS]
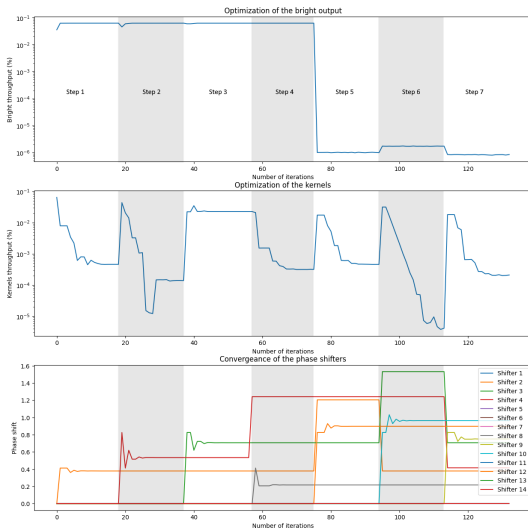

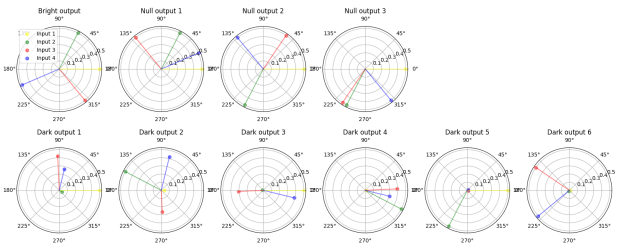**Fig. 3.** Calibration using input obstruction [AJOUTER DESCRIPTION]
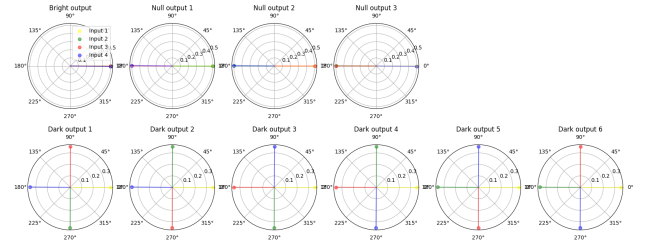

Fig. 4. Perturbed phases

Fig. 5. Calibrated phases

4. Results and limitations

1. Numerical results
   Kernel-Null depth (Fig 2 & 3)
   Kernel inversion and swapping
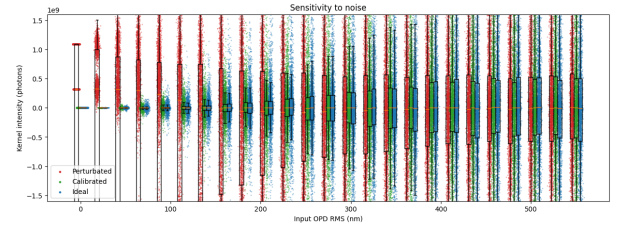2. Laboratory results
3. Laboratory limitations (ex. crosstalk)

Fig. 6. Sensitivity to input noise

5. Conclusions and prospects

1. Conditions for noticing a performance gain
2. Need of a post calibration caracterization process to identify the outputs
3. Deeper statistical analysis is required to truely caracterize performance gain (the null depth is not the only relevant parameter)
4. Architecture limitations (ex. no amplitude modulation, no photometric outputs)

References

Lorem ipsum