Reflector Antenna Analysis using Physical Optics on Graphics Processing Units

Oscar Borries[1,2], Hans Henrik Brandenborg Sørensen[1], Bernd Dammann[1],
Erik Jørgensen[2], Peter Meincke[2], Stig Busk Sørensen[2], Per Christian Hansen[1]
[1]Technical University of Denmark, DTU Compute, Kgs. Lyngby, Denmark, {opbo,hhbs,beda,pcha}@dtu.dk
[2]TICRA, Læderstræde 34, DK-1201 Copenhagen, Denmark, {ob,ej,pme,sbs}@ticra.com

***Abstract*—The Physical Optics approximation is a widely used asymptotic method for calculating the scattering from electrically large bodies. It requires significant computational work and little memory, and is thus well suited for application on a Graphics Processing Unit. Here, we investigate the performance of an implementation and demonstrate that while there are some implementational pitfalls, a careful implementation can result in impressive improvements.**

***Index Terms*—Physical Optics, Computational Electromagnetics, Graphical Processing Units**

I. Introduction

The Physical Optics (PO) approximation constitutes a high-frequency approximation to the induced surface current density $\boldsymbol{J}_{\mathcal{S}}$ on a perfectly electrically conducting scatterer $\mathcal{S}$ given by

$$\boldsymbol{J}_{\mathcal{S}} = \begin{cases} 2\hat{\boldsymbol{n}} \times \boldsymbol{H}_i & \text{if illuminated} \\ 0 & \text{otherwise} \end{cases} \tag{1}$$

where $\boldsymbol{H}_i$ is the incident magnetic field and $\hat{\boldsymbol{n}}$ is the outward unit normal vector of $\mathcal{S}$. PO is particularly applicable to reflector antenna systems since these are always electrically large. Finding $\boldsymbol{J}_{\mathcal{S}}$ from (1) requires very little work itself, but computing the far field from $\boldsymbol{J}_{\mathcal{S}}$ requires evaluation of a surface integral of the form

$$\begin{aligned}\boldsymbol{E}_{\text{far}}(x,y,z) =& \frac{e^{-jk|\boldsymbol{r}|}}{|\boldsymbol{r}|}\frac{jk\eta_0}{4\pi}\hat{\boldsymbol{r}}\times \\ & \left[\hat{\boldsymbol{r}} \times \iint \boldsymbol{J}_{\mathcal{S}}(\boldsymbol{r}')e^{jk\hat{\boldsymbol{r}}\cdot\boldsymbol{r}'}dS'\right],\end{aligned} \tag{2}$$

where $k = 2\pi/\lambda$, λ is the wavelength and j is the imaginary unit, while $\boldsymbol{r} = x\hat{\boldsymbol{x}} + y\hat{\boldsymbol{y}} + z\hat{\boldsymbol{z}}, \boldsymbol{r}' = x'\hat{\boldsymbol{x}} + y'\hat{\boldsymbol{y}} + z'\hat{\boldsymbol{z}}$ denote observation and integration points, respectively. This integral takes up considerable computational resources for large reflectors.

In particular, for PO applied to a dual reflector setup, calculating the incident magnetic field on the main reflector due to the surface current distribution on the sub-reflector typically constitutes the vast majority of the computational load. The surface integral to be evaluated is of the form

$$\begin{aligned}\boldsymbol{H}_i(x,y,z) =& -\frac{1}{4\pi}\iint \left(\hat{\boldsymbol{R}} \times \boldsymbol{J}_{\mathcal{S}}(x',y',z')\right) \\ & \frac{1+jk|\boldsymbol{R}|}{|\boldsymbol{R}|^2}e^{-jk|\boldsymbol{R}|}dS',\end{aligned} \tag{3}$$

where $\boldsymbol{R} = \boldsymbol{r} - \boldsymbol{r}' = (x-x')\hat{\boldsymbol{x}} + (y-y')\hat{\boldsymbol{y}} + (z-z')\hat{\boldsymbol{z}}$ and $\boldsymbol{J}_{\mathcal{S}}$ is the surface current density on the subreflector.

II. Algorithm

Defining a set of field points $\mathcal{F}$ containing N points and a current distribution $\mathcal{J}$ discretized in M points, the rough outline of the code is given below. The operator $\mathcal{V}(J)$ refers to the integrand in (3), and w_j are the integration weights.

Algorithm 1 Pseudocode for implementing (3).

```
Zero $\mathcal{F}$
**for** $i = 1, N$ **do**
    **for** $j = 1, M$ **do**
        $\mathcal{F}_i = \mathcal{F}_i + \mathcal{V}(J_j)w_j$
    **end for**
**end for**
```

The inner part of this algorithm is evaluated MN times and requires an array of size $M + N$. Further, since the operator $\mathcal{V}(J)$ requires $\mathcal{O}(MN)$ operations and M and N scale as $\mathcal{O}(\lambda^{-2})$, the calculation of $\mathcal{F}$ requires $\mathcal{O}(\lambda^{-4})$ operations. This suggests why the calculation of (3) is a significant computational load for electrically large $\mathcal{S}$.

III. Performance on a GPU

To implement the code on a GPU (Graphics Processing Unit), we use the CUDA framework [1] developed by Nvidia. A first implementation is made by porting the existing code from the GRASP software package [2]. As a testcase the scattered field from a 20 m square plate, excited by a plane wave at 3 GHz, at normal incidence and polarized parallel to two of the sides, is computed.

The performance, relative to the highly optimized GRASP code, is given in columns 2 and 3 in Table I. The timings are done on a quad-core 2.9 GHz Intel i5, and an Nvidia GTX 670 Mini-ITX. The retail prices of just the processor and the graphics card are roughly equivalent. The performance in double precision is somewhat disappointing, only a factor of 3. Although the GPU is faster than the reference implementation, it might not seem enough to warrant the additional coding needed.

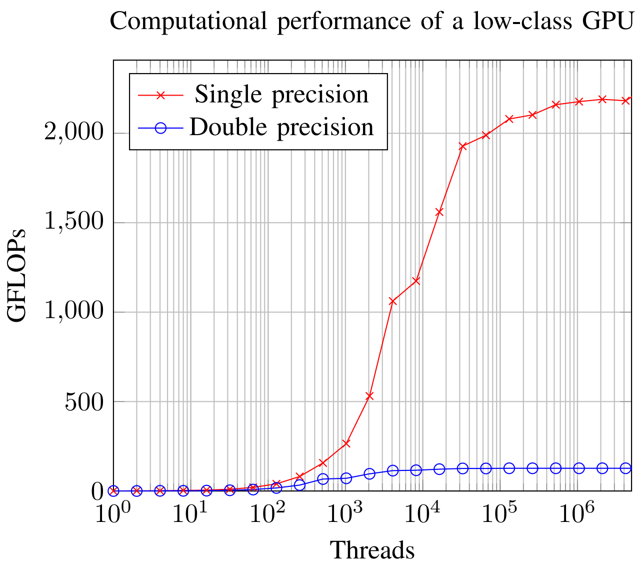

Fig. 1. Computational performance of the Nvidia GTX 670, typical of low-end GPUs. Along the x-axis is the number of threads launched, proportional to the amount of work done on the GPU. Along the y-axis is the number of floating point operations per second.

$M = N =$	GRASP Time [s]	GPU double Time [s]	GPU single Time [s]
200^2	8.7	3.2	0.8
300^2	44.4	14.0	2.4
400^2	142	42.6	7.1
500^2	353	101	16.4
600^2	707	210	33.8

TABLE I
THE PERFORMANCE OF THE IMPLEMENTATION, BOTH IN GRASP AND ON THE LOW-END GPU, BOTH IN DOUBLE AND SINGLE PRECISION.

To discover why, we apply a High-Performance Computing benchmarking suite currently under developement at the Technical University of Denmark [3] to investigate the behaviour of the GPU in detail. The key issue here, the number of floating point operations per second (flops), is shown in Figure 1. From this figure, we see an extreme discrepancy between the performance in single and double precision. Provided that there is sufficient work for the entire card, the single precision performance is roughly 2.2 teraflops while the double precision performance is only around 150 gigaflops.

Since the real performance advantage of low-end GPUs thus is in the single precision domain, we implement the code in single precision in CUDA to see the difference in performance. The fourth column in Table I demonstrates the results.

IV. Single Precision

Considering whether single precision is sufficiently accurate for an application requires analysing the behaviour of the complex exponential appearing in (3). Since all other operations are accurate to machine precision, the dominant error term is the term

$$e^{-jk|\boldsymbol{R}|} = \cos(k|\boldsymbol{R}|) - j\sin(k|\boldsymbol{R}|) \quad (4)$$

and thus the accuracy of $\cos$ and $\sin$ for a given architecture. For Nvidia CUDA, the accuracy of trigonometric single precision special functions is 2 ULP (Units in Last Place) throughout the range $[-2\pi; 2\pi]$, meaning that the number of contaminated digits is roughly $\log_2(2) = 1$ in IEEE single precision [4]. This means that the relative error is approximately 10^{-6}.

Therefore, to find the precision for large arguments, we consider

$$\log_{10}\left(\frac{k|\boldsymbol{R}|}{2\pi}\right) = \zeta \quad (5)$$

Thus, the relative error will be $10^{-(6-\zeta)}$.

In GRASP, the natural error criterion is a specification of the error level, i.e. an 80 dB criterion suggests that the power $|\boldsymbol{E}|^2$ is accurate to 80 dB below peak. This criterion corresponds to $\log_{10}\left(\frac{80}{2}\right) = 4$ digits of accuracy, suggesting that ζ can be no more than 2. This means that we require

$$|\boldsymbol{R}| < \frac{2\pi 10^{\zeta}}{k} = 10^{\zeta}\lambda \quad (6)$$

The applicability of single precision in Physical Optics thus clearly depends on both the required accuracy and the case at hand. For 80 dB accuracy in a dual reflector setup, the distance between the sub- and main reflector thus cannot be more than 100 λ, a fairly low number in many cases. However, if 60 dB accuracy can be accepted, the distance cannot be more than 1000λ, a much less restrictive number.

Finally, we stress that these considerations are only applicable for the near-field scenario in (3). For the far-field computations (2), the exponential function has much smaller arguments, so there is no loss of accuracy involved in using single precision here.

V. Realistic Configurations

To demonstrate the performance on realistic scatterers, we consider here two setups. First, a Cassegrain dual reflector antenna, a typical benchmark for a Physical Optics code, designed from textbook formulas [2]. Second, we move to an actual application of a high-gain reflector antenna mounted on a satellite in low orbit, taken from an ongoing ESA project [5].

A. *Cassegrain dual reflector*

The setup is illustrated in Figure 2, with the relevant data shown in Table II.

The accuracy criterion is set to 60 dB, yielding a maximum distance of approximately 1000λ, sufficient for the present case to be analyzed in single precision. The electric far-field from the system is sampled on a $\theta\phi$-grid at a sample spacing of $\frac{1}{8}\frac{\lambda}{D}$ in θ, where D is the aperture diameter of the antenna, between $-10°$ and $10°$ in θ. Further, the sampling in ϕ is done at $5°$ increments between 0 and $90°$. This yields a total of 3350×19 points.

The time spent in each task is shown in Table III, where we see that the GPU code is more than 7 times faster than the reference GRASP implementation. From the table, we

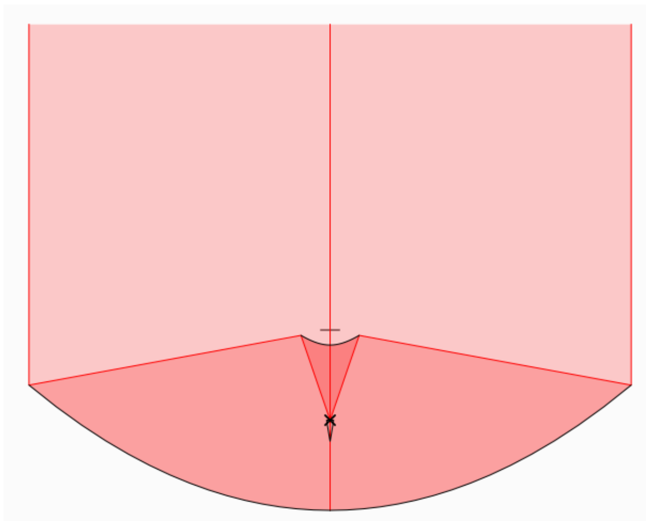

Fig. 2. Illustration of the Cassegrain dual reflector setup.

Wavelength	1 cm
Main Reflector Aperture	12 m
Main Reflector f/D	0.4
Subreflector eccentricity	1.5
Subreflector diameter	104 cm

TABLE II
SYSTEM DATA FOR THE CASSEGRAIN DUAL REFLECTOR SETUP.

Task	GRASP [s]	GPU single [s]	Speed-up
PO on sub from feed	0.9	2.8	0.3
PO on main from sub	293	47	6.2
Far-field from system	88	3.3	26.6
Total	382	53	7.2

TABLE III
TIMINGS FROM A COMPLETE PHYSICAL OPTICS ANALYSIS OF THE CASSEGRAIN DUAL REFLECTOR SETUP.

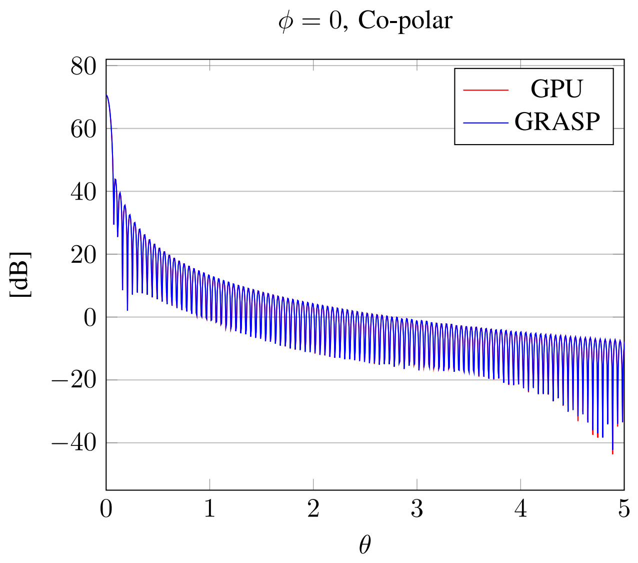

Fig. 3. The total electric field from the Cassegrain dual reflector, calculated with both the reference GRASP implementation and the GPU single precision implementation discussed here. Co-polar component at $\phi = 0$. For clarity, only the interval 0–5° is shown.

also see that particularly the far-field computation is much faster on the GPU compared to the near-field calculations. The relatively slow computation of the near-field is primarily due to the use of auto-convergence, which uses repeated runs with a small number of field points to determine the number of current points on the reflectors. This in turn causes sub-optimal occupancy of the GPU, resulting in a smaller speed-up than otherwise achievable. We stress that for repeated runs, the auto-convergence is only done once, and thus the relative gain will be much larger. This also explains why the speed-up is poorer than expected from Table I, since in the latter case, the number of field points is large enough to ensure sufficient occupancy.

Another potential reason for the near-field calculations to yield a poorer speed-up is due to the implementation of trigonometric functions on the GPU. According to the documentation, costly argument reduction techniques are applied if the argument is larger than 48039. In this case, another code path is used, requiring significantly higher register use, which in turn causes poorer performance - the CPU code does not have such issues. While the argument in the present case is lower than 48039, it is still worth noting.

In conclusion, the speed-up is a respectable 6.2 for the large PO run, while the small task of performing PO on the subreflector from the feed is so fast that the overhead from transferring memory to and from the GPU results in a slowdown, resulting in a speed-up of 0.3.

The scattered field is shown in Figures 3–4 for the co- and cross-polar components, respectively. Although hard to note from the fast oscillations, the red and blue lines are essentially identical in the co-polar component. For the much lower cross-polar component, which is nearly 100 dB below peak, the effects of single precision results in a relative difference of 3%, acceptable for the low levels. Interestingly, while this cross-polar component should be numerically zero, due to a combination of integration error and numerical noise, a distinct pattern is computed. It is important to note that both the GPU and GRASP reference solution, though yielding slightly different values, still results in roughly the same pattern, including the artificial asymmetry at $\theta \approx 3°$. This further confirms the applicability of the GPU solution, since even far below the requested accuracy, it still provides the same overall pattern as the reference solution.

B. *Low-Orbit Reflector*

To look into another case that is particularly suitable for acceleration from a GPU based implementation, we consider a high-gain antenna with a torus reflector mounted on a low-orbit satellite, designed for ocean surveillance [5]. The information for the system is described in Table IV and a figure illustrating its operating conditions and geometry is shown in Figures 5–6.

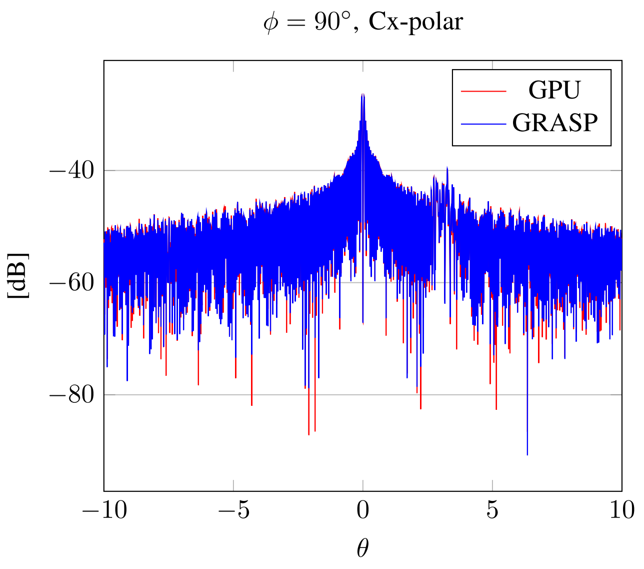

Fig. 4. The total electric field from the Cassegrain dual reflector, calculated with both the reference GRASP implementation and the GPU single precision implementation discussed here. Cross-polar component at $\phi = 90°$. Note the lack of symmetry around $\theta = 0$, indicating that both the GRASP and GPU implementations are affected by inaccuracies.

Frequency	10 GHz
Projected reflector aperture	5 m
Reflector f/D	1
Clearance	1 m
Operating altitude	817 km

TABLE IV
SYSTEM DATA FOR THE LOW-ORBIT REFLECTOR.

Due to the low orbit, the far-field pattern needs to be tabulated for the complete angular region from 0 to 60° relative to boresight, and the high-gain, large aperture of the antenna requires a closely spaced sampling. This results in a far-field grid of 1130913 field points when converted to a circular region in a uv-grid. The result is shown in Figure 7.

In this case, several runs have been made, so the number of current points on the reflector has been determined in advance. Thus, the only relevant task is the determination of the large and closely sampled far-fields, which are further complicated by the somewhat complicated geometry of the reflector, requiring a densely sampled PO grid, yielding 1122328 integration points. The closely sampled far-field required nearly 4 hours to evaluate in GRASP, while it took a little under 10 minutes on the GPU. The total speedup is thus roughly a factor of 24.

VI. Conclusion

The demanding task of computing the radiated fields from an electrically large current distribution has been converted to a low-end GPU and key performance issues have been identified. Taking into account those issues, several testcases have demonstrated very significant performance gains, even when considered relative to industry-standard code, for comparable accuracies.

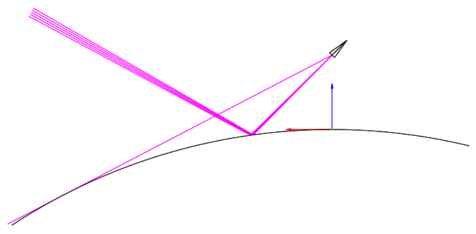

Fig. 5. Geometry for an antenna in low orbit. The reflected rays determine the boresight, while the thin line illustrates the ray at $\theta \approx 60°$ relative to boresight.

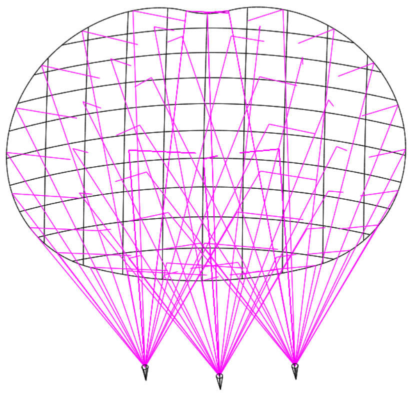

Fig. 6. Illustration of a torus reflector illuminated by three seperate feeds, used as a testcase for the GPU implementation.

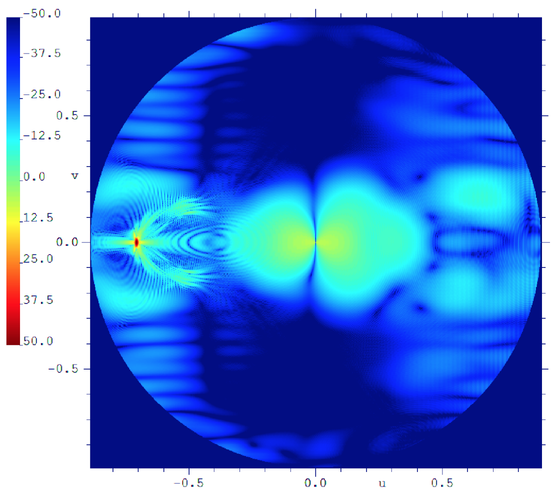

Fig. 7. An example of the result achieved for the low-orbit antenna, co-polar component. We note the small high-gain region at $u \approx -0.8$, $v = 0$. The limits for the plot are approximately ± 0.89 in both u and v.

References

[1] NVIDIA Corp. CUDA 5.5.

[2] K. Pontoppidan, *GRASP Technical Description*. TICRA, Mar. 2008.

[3] Technical University of Denmark, "GPULAB." [Online]. Available: http://gpulab.imm.dtu.dk

[4] D. Goldberg, "What every computer scientist should know about floating-point arithmetic," *ACM Computing Surveys*, vol. 23, no. 1, pp. 5–48, Mar. 1991.

[5] C. Cappellin, K. Pontoppidan, P. H. Nielsen, N. Skou, S. Søbjærg, A. Ihle, D. Hartmann, M. Ivashina, O. Iupikov, and K. v. t. Klooster, "Novel Multi-Beam Radiometers for Accurate Ocean Surveillance," in *EUCAP*, 2014.