International Journal of Engineering, Mathematical and Physical Sciences
ISSN: 2517-9934
Vol:10, No:2, 2016

Modeling Bessel Beams and Their Discrete Superpositions from the Generalized Lorenz-Mie Theory to Calculate Optical Forces over Spherical Dielectric Particles

Leonardo A. Ambrosio, Carlos. H. Silva Santos, Ivan E. L. Rodrigues, Ayumi K. de Campos, Leandro A. Machado

***Abstract*—**In this work, we propose an algorithm developed under Python language for the modeling of ordinary scalar Bessel beams and their discrete superpositions and subsequent calculation of optical forces exerted over dielectric spherical particles. The mathematical formalism, based on the generalized Lorenz-Mie theory, is implemented in Python for its large number of free mathematical (as SciPy and NumPy), data visualization (Matplotlib and PyJamas) and multiprocessing libraries. We also propose an approach, provided by a synchronized Software as Service (SaaS) in cloud computing, to develop a user interface embedded on a mobile application, thus providing users with the necessary means to easily introduce desired unknowns and parameters and see the graphical outcomes of the simulations right at their mobile devices. Initially proposed as a free Android-based application, such an App enables data post-processing in cloud-based architectures and visualization of results, figures and numerical tables.

***Keywords*—**Bessel Beams and Frozen Waves, Generalized Lorenz-Mie Theory, Numerical Methods, Optical Forces.

I. Introduction

OPTICAL force control includes outstanding contributions in areas such as quantum communications and computing, besides being an important tool in medicine, particularly in biomedical optics, where optical bistouries and tweezers are widely employed for applications such as particle characterization, biophysics study of biological systems, measurement of stretching of DNA and so on [1], [2]. In this context, this work initially presents a brief review and delineate some aspects and properties of particular classes of non-diffracting optical wave fields [viz., Bessel beams (BBs) and their discrete equal-frequency superpositions (Frozen Waves or FW)] [3]-[6], their modeling and interactions with dielectric spherical particles, demonstrating how one can compute optical forces (or, equivalently, radiation pressure cross sections) exerted on such micro-sized scatterers starting from the so-called generalized Lorenz-Mie theory (GLMT) [7]. For arbitrary-sized particles, this modeling requires some large amount of vector calculus under the numerical complex data representations which, consequently, may imply on the adoption of high performance computational approaches. Additionally, the description of the spatial intensity profile of the incident arbitrary-shaped beam involves the determination of beam-shape coefficients (BSCs), whose numerical calculation can be quite time-consuming (especially when defined in terms of double or triple integrals over a spherical coordinate system [7]). Furthermore, there are few toolboxes made available online (most still constructed using Fortran or Pascal language) for which the user can readily perform optical force simulations even from single scalar BBs, which motivated this work to also present some subroutines developed in Python and using SciPy library to easily model those optical wave fields and their discrete equal-frequency superpositions in order to promptly calculate all the Cartesian components of the optical forces. This script language was initially adopted first because of its easily available and fast development resources, and also due to its wide number of free online available libraries and documentation. Besides, this language is hardware and software multiplatform acceptable, from mobile to server architectures, being possible to execute it in different operational systems. Because this numerical solution description and computational approach is interesting for its data visualization, multi-processing possibilities, parallel processing and interface design modeling libraries that are freely available online, we have been conceiving the development of a full user interface toolbox. Here, we discuss on our recent advancements for this free Android-based application, which enables data post-processing in cloud-based architectures and visualization of results, figures and numerical tables from mobile devices.

As a study of case, this work presents the simulation of intensity profiles (field intensity) and longitudinal optical forces for two commonly adopted FWs which can be of practical use in optical trapping or tweezers systems: a constant longitudinal intensity profile (LIP) and a growing exponential LIP, both valid only within some pre-defined

L. A. Ambrosio is with the University of São Paulo (USP), School of Electrical and Computer Engineering (EESC), Department of Electrical and Computer Engineering (SEL), São Carlos, SP 13566-590 Brazil. (phone: +55-16-988191860; e-mail: leo@sc.usp.br). Financial support: FAPESP (#2014/04867-1)

C. H. Silva-Santos and I. E. L. Rodrigues are with São Paulo Federal Institute of Education, Science and Technology (IFSP), câmpus Itapetininga, Itapetininga, SP 18202-000 Brazil (phone +55-15-3375-9944; e-mail: carlos.santos@ifsp.edu.br, ivan.elr93@gmail.com).

A. K. Campos is with São Paulo Federal Institute of Education, Science and Technology (IFSP), câmpus Itapetininga, Itapetininga, SP 18202-000 Brazil. (phone +55-15-3375-9944; ayumicampos@hotmail.com).

L. A. Machado is with School of Technology, Science and Technology (FATEC), Itapetininga, SP 18202-000 Brazil (phone +55-15-3375-9944; leandro.almachado@gmail.com

spatial longitudinal range.

II. The Generalized Lorenz-Mie Theory and Theoretical Description of Frozen Waves for Optical Force Calculations

Ideal ordinary or zero-order BBs first appeared as solutions to the scalar wave equation in the form $\psi(r,z) = J_0(k_r r)\exp(-ik_z z)$ [8], [9], ω being the operating frequency, $J_0(.)$ the zero-order Bessel function and k_r (k_z) the radial (axial) wave number for a cylindrical coordinate system (r,ϕ,z), assuming a time convention $\exp(i\omega t)$. In general, $\psi(r.z)$ is directly related to a particular transverse electric field component, thus corresponding to a linearly polarized wave field under the paraxial regime.

Despite being physically unrealizable, ideal BBs are introduced as a first approximation of truncated BBs generated from finite apertures. In biomedical optics, they are frequently incorporated as optical beams for optical trapping and manipulation of biological particles [10]-[14]. Although they possess several advantages in terms of resistance to diffraction, they suffer from a natural inability of realizing effective three-dimensional (3D) traps. Several traps along multiple planes can be easily accomplished with single BBs, but one cannot mechanically displace any trapped particle along its axis simply because there is always a natural predominance of *repulsive* (over attractive) longitudinal forces. Recently, however, one of the authors theoretically investigated the possibility of replacing, in optical trapping systems, single BBs by a new class of non-diffracting beams called FWs [15]-[17]. These waves are still a solution to the scalar wave equation, but such a solution is now achieved by means of a suitable discrete superposition of BBs, all with the same frequency and order but carrying distinct radial and transverse wave numbers (paraxiality requirements being always satisfied). The main characteristic of FWs relies on the fact that one can design any desired longitudinal intensity pattern $F(z)$ along some pre-defined distance $0 \le z \le L$ (or $-L/2 \le z \le L/2$) thus allowing for the possibility of achieving effective 3D traps *only from z-propagating BBs with the same order*. For example, suppose the above solution $\psi(r,z)$. For a z-propagating ideal FW, one has the following superposition of $2N + 1$ BBs:

$$\Psi(r,z,t) = e^{i\omega t}\sum_{q=-N}^{N} A_q J_0\left(k_{rq} r\right) e^{-ik_{zq} z}. \tag{1}$$

In (1), k_{rq} and k_{zq} represent, respectively, the radial and axial wave numbers for the q-th BB, and A_q are the complex coefficients of the expansion and intrinsically related to $F(z)$:

$$A_q = \frac{1}{L}\int_{-L/2}^{L/2} F(z) e^{i(2\pi/L)qz} dz. \tag{2}$$

Further details and particular examples of LIPs can be found elsewhere [3]-[5]. We emphasize, however, that in (1) and (2) only propagating (non-evanescent) BBs are considered.

In evaluating the optical forces exerted on spherical particles by FWs one must, as is the common practice for arbitrary-sized scatterers, first compute the so-called BSCs which describes the spatial intensity distribution of the incident beam. In the context of the GLMT, EM fields are expanded in terms of vector spherical harmonics, the complex coefficients of such an expansion being those BSCs [7]. Several techniques are available for computing them, from exact approaches involving triple or double integrations over the spherical coordinates, quadrature schemes and, finally, the well-known localized approximations [7], [18]. Here, however, we consider a more adequate procedure for deriving the electromagnetic (EM) fields of *vector* BBs starting from the vector potential **A** and imposing, for instance, Lorenz gauge condition. This procedure (which demands working in Fourier space) has been recently proposed by Moreira et al. and furnishes exact solutions to the BSCs for *vector* BBs with circular polarization without the need for any integration [19].

Extending the derivations in [19], it is possible to show that, for an ordinary (zero-order) FW with circular polarization and displaced (r_0,ϕ_0,z_0) from the origin, the BSCs for TE and TM modes read as

$$\begin{aligned}
g_{n,m}^{TE}\big|_{+} &= \frac{4\pi(i)^{n-m+1}}{\sqrt{2n(n+1)}}\sum_{q=-N}^{N} A_q \psi_{-m+1}\left(\vec{k}_q;\vec{r}_0\right) c_{-}^{nm} Q_n^{m-1}\left(\frac{k_{zq}}{k}\right)\\
g_{n,m}^{TE}\big|_{-} &= \frac{4\pi(i)^{n-m-1}}{\sqrt{2n(n+1)}}\sum_{q=-N}^{N} A_q \psi_{-m-1}\left(\vec{k}_q;\vec{r}_0\right) c_{+}^{nm} Q_n^{m+1}\left(\frac{k_{zq}}{k}\right)\\
g_{n,m}^{TM}\big|_{+} &= -i\frac{4\pi(i)^{n-m+1}}{\sqrt{2n(n+1)}}\sum_{q=-N}^{N} A_q \psi_{-m+1}\left(\vec{k}_q;\vec{r}_0\right)\times\\
&\quad\left\{\frac{k_{zq}}{k} c_{-}^{nm} Q_n^{m-1}\left(\frac{k_{zq}}{k}\right) - m\frac{k_{rq}}{k} Q_n^{m}\left(\frac{k_{zq}}{k}\right)\right\}\\
g_{n,m}^{TM}\big|_{-} &= i\frac{4\pi(i)^{n-m-1}}{\sqrt{2n(n+1)}}\sum_{q=-N}^{N} A_q \psi_{-m-1}\left(\vec{k}_q;\vec{r}_0\right)\times\\
&\quad\left\{\frac{k_{zq}}{k} c_{+}^{nm} Q_n^{m+1}\left(\frac{k_{zq}}{k}\right) - m\frac{k_{rq}}{k} Q_n^{m}\left(\frac{k_{zq}}{k}\right)\right\},
\end{aligned} \tag{3}$$

where

$$\begin{aligned}
c_{\pm}^{nm} &= \sqrt{n(n+1) - m(m\pm 1)}\\
\psi_v\left(\vec{k}_q;\vec{r}_0\right) &= J_v\left(k_{rq}\rho_0\right) e^{-iv s\phi_0} e^{-ik_{zq} z_0},
\end{aligned} \tag{4}$$

with n a positive integer, $-n \le m \le n$, $k = 2\pi/\lambda$ and $Q_n^m(.)$ being the associated Legendre polynomials. After the BSCs have been found from (3) and (4) and assuming that both the propagating medium is lossless, the rectangular components of the radiation pressure cross-section (or, equivalently, the optical forces) are readily found:

$$\begin{bmatrix} C_{pr,x} \\ C_{pr,y} \end{bmatrix} = \frac{1}{4k^2}\begin{bmatrix} \mathrm{Re} \\ \mathrm{Im} \end{bmatrix}\sum_{n=1}^{\infty}\frac{i}{n+1}\left\{\sqrt{\frac{n(n+2)}{(2n+3)(2n+1)}}\times\right.$$
$$\left[\sum_{m=-n}^{n}\sqrt{(n+m+2)(n+m+1)}\times\right.$$
$$\left[\left(a_{n+1}+a_n^*-2a_{n+1}a_n^*\right)g^{TM}_{(n+1),-(m+1)}g^{TM*}_{n,-m}\right.$$
$$+\left(a_n+a_{n+1}^*-2a_na_{n+1}^*\right)g^{TM*}_{(n+1),(m+1)}g^{TM}_{n,m}$$
$$+\left(b_{n+1}+b_n^*-2b_{n+1}b_n^*\right)g^{TE}_{(n+1),-(m+1)}g^{TE*}_{n,-m}$$
$$\left.+\left(b_n+b_{n+1}^*-2b_nb_{n+1}^*\right)g^{TE*}_{(n+1),(m+1)}g^{TE}_{n,m}\right]\Big]$$
$$-\frac{1}{n}\sum_{m=-n}^{n}\sqrt{(n+m+2)(n+m+1)}\times$$
$$\sqrt{(n-m)(n+m+1)}\times$$
$$\left[\left(a_n+b_n^*-2a_nb_n^*\right)g^{TM}_{n,m}g^{TE*}_{n,(m+1)}\right.$$
$$\left.\left.-\left(b_n+a_n^*-2b_na_n^*\right)g^{TE}_{n,m}g^{TM*}_{n,(m+1)}\right]\right\} \quad (5)$$

$$C_{pr,z} = \frac{1}{4k^2}\sum_{n=1}^{\infty}\frac{1}{n+1}\left\{\sqrt{\frac{n(n+2)}{(2n+3)(2n+1)}}\times\right.$$
$$\left[\sum_{m=-n}^{n}\sqrt{(n+m+1)(n-m+1)}\times\right.$$
$$\left[\left(a_{n+1}+a_n^*-2a_{n+1}a_n^*\right)g^{TM}_{(n+1),-(m+1)}g^{TM*}_{n,-m}\right.$$
$$+\left(a_n+a_{n+1}^*-2a_na_{n+1}^*\right)g^{TM*}_{(n+1),(m+1)}g^{TM}_{n,m}$$
$$+\left(b_{n+1}+b_n^*-2b_{n+1}b_n^*\right)g^{TE}_{(n+1),-(m+1)}g^{TE*}_{n,-m}$$
$$\left.+\left(b_n+b_{n+1}^*-2b_nb_{n+1}^*\right)g^{TE*}_{(n+1),(m+1)}g^{TE}_{n,m}\right]\Big]$$
$$-\frac{1}{n}\sum_{m=-n}^{n}\sqrt{(n+m+2)(n+m+1)}\times$$
$$\sqrt{(n-m)(n+m+1)}\times$$
$$\left[\left(a_n+b_n^*-2a_nb_n^*\right)g^{TM}_{n,m}g^{TE*}_{n,(m+1)}\right.$$
$$\left.\left.-\left(b_n+a_n^*-2b_na_n^*\right)g^{TE}_{n,m}g^{TM*}_{n,(m+1)}\right]\right\} \quad (6)$$

In (5) and (6), a_n and b_n are the Mie coefficients dependent upon the geometric and EM properties of the scatterer and involving spherical Bessel (or Ricatti-Bessel) functions [7]. Significant computational effort may be required for large particles, since the number of BSCs and Mie coefficients required to correctly predict the optical force profiles increases with their radii. For a spherical particle with radius a, a simplistic estimative of n can be given by the nearest integer to the product $ka = 2\pi(a/\lambda)$ [7]. As an example, for dipole particles (far from resonances), $n = 1$. For $a = 10\lambda$, $n = 63$. Therefore, once a given LIP [$F(z)$] has been chosen, A_q can be found from (2), which now leads to the determination of all the BSCs in (3) needed to calculate (5) and (6) once the geometric and EM (permittivity and permeability) properties of the particle are known. Numerical calculations now come into play. Notice, finally, that the procedure just outlined can be used as a guideline for optical force calculations from arbitrary-shaped beams, even though, depending on the beam, one may not avoid going into triple, double or single integrals in order to find the corresponding BSCs (and, consequently, also increasing computational effort).

III. Computational Approaches and Some Numerical Results

These numerical modeling and applications are, in most of the cases, adopted by non-computer scientists or programmer specialists. Therefore, the user or developer training process might spend a significant amount of time, depending on the chosen technological to perform it, trying to get familiar with them [20]. Furthermore, because of the numerical complexity level exposed here, per passing by the data visualizations and, consequently, the computational powerful requirements, the programming language adopted must include numerical libraries and computational multiplatform acceptance to attempt this data processing variety. In view of that, we have introduced the Python script language as an interesting alternative for the already developed Fortran source codes for these Bessel Beam calculations under GLMT equations to calculate optical forces over scatterers with spherical symmetry/geometry [15]-[18]. This language presents sophisticated and wide range of numerical approaches with efficient data processing [21], [22]. Furthermore, it is easier to be used in these applications [23] and possesses interesting data visualization approaches and multiplatform acceptance [24], [25]. The latter includes the mobile device acceptance, such as here presented: an initial mobile device interface that sends data for cloud computing (where the simulations occurs), from which some numerical data are streamed back in order to generate graphical information right at the display of the mobile device. Therefore, this section is organized as follows: A first subsection briefly comments on the Python routines necessary to implement the previous Bessel Equations. Then, a second subsection gives some insights into the mobile interface under-construction in a Cordova project developed under the Ionic framework.

A. Python Source Code

The development process was preceded by numerical studies, where initially the C++, Java and Python languages were considered in order to successfully develop our numerical approach.

As already mentioned, Python was chosen mainly because of its easy of understanding, processing performance and multiplatform (hardware and software) acceptance. But, apart from that, an another important and specific detail contributes to its adoption, viz., its open source license maintained by the Python Software Foundation (PSF) [26]. In addition, it also has interesting numerical libraries such as NumPy and SciPy, and output data graphical visualization with MatPlotLib, for which details and examples of codes can be freely accessed online [27]. The source code developed here is also supported by the open source cross-platform Spyder IDE.

All main vector equations were developed under reuse, being specified in Python function, which is easily designed using the command word *spec*, as presented in Fig. 1, where some examples of numerical functions are also presented. For instance, at line #93 (#108) the real part of the longitudinal (radial) wave vector is defined, according to the time-harmonic convention $\exp(i\omega t)$.

```
93  def betarm(m):
94      #parte real da componente longitudinal do
95      #vetor de onda, k_{zq} com q = m
96      return Qbeta+(2*math.pi*m)/L
97
98  def betam(m):
99      #componente longitudinal do vetor
100     # de onda, k_{zq} com q = m
101     return betarm(m)-1j*betaim(m)
102
103 def kphom(m):
104     #componente transversal do vetor de onda,
105     # k_{phoq} com q = m
106     return cmath.sqrt((k(indref,w))**2 + (betam(m))**2)
107
108 def kphomr(m):
109     #parte real da componente transversal do vetor da onda,
110     #k_{phoq} com q = m
111     return numpy.real(kphom(m))
112
113 def kphomi(m):
114     #parte imaginária da componente transversal
115     #do vetor da onda, k_{phoq} com q = m
116     return numpy.imag(kphom(m))
```

Fig. 1 Python Source Code Example

In order to allow easier and motivated interfaces, this work team group is developing a mobile device interface whose details are given in the next subsection.

B. Mobile Device Interface

Considering the future possibility of freely distributing this package for interested users, a usable interface is perhaps mandatory in order to motivate its adoption. Furthermore, the current software usability techniques are, generally, focused on the multi-platform acceptance, twofold desktop and mobile ones [28].

Considering our numerical large scale and complex application, the current mobile devices are not capable of fully dealing with it. Therefore, some alternative technological approaches are necessary to support these back-end computational demands, the cloud computing being an interesting model to suppress this large scale computational requirements, which incurs in some case for high performance computing solutions that are suitable for the cloud computing solutions [29].

On the other hand, in the software front-end one can say that mobile devices are interesting resources to improve the user interface, with some interactive resources including touch-screen that can maximize the user interface experience [30]. In addition, its mobility also improves the mobile and geographical usage. In this sense, Fig. 2 presents a schematic view of this numerical user interface proposal where the mobile device communicates with a cloud computing within which data are processed in the Pyhton source codes and from which streamed text data is returned/sent to the mobile device to generate the graphical visual results.

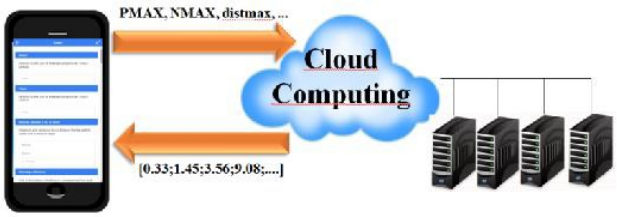

Fig. 2 Cloud and mobile connection model

Fig. 3 presents a simple interface under development by this work team group to attempt this mobility demand. It has been performed under the Apache Cordova to support Web 2.0 applications (HTML5, CSS and JS). The development itself has been made under the Ionic Framework, allowing the application to run in Android and iOS platforms. We intend to comment on that during the conference.

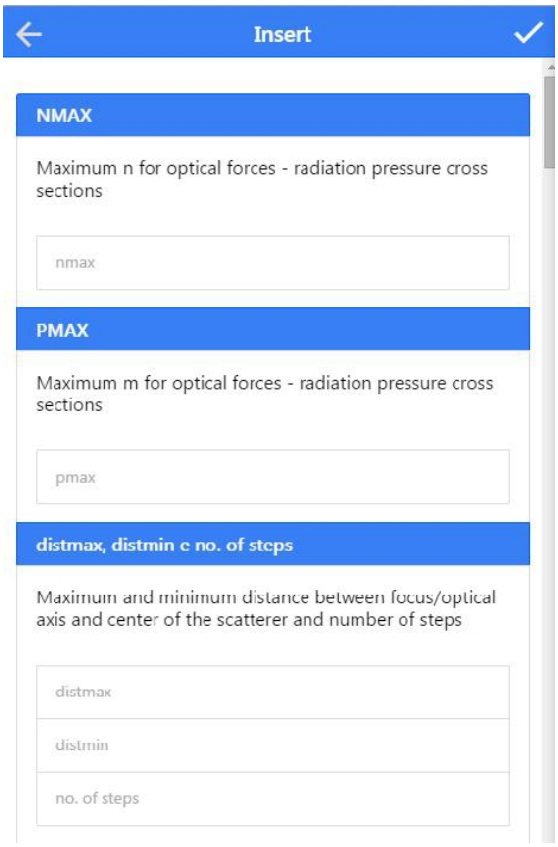

Fig. 3 Input interface parameters

C. Numerical Results

In subsection A (Section III) it was considered the development of a numerical algorithm devoted to the task of evaluating the optical forces exerted exclusively over spherical particles. Python language was adopted and the equations of Section I were then implemented. This means that, at first, only those forces arising from the interaction between

spherical scatterers and zero-order scalar BBs and their discrete superpositions can be analyzed. Obviously, higher order beams, in addition to other laser beams (e.g., Gaussian beams), shall be included in due time.

Here, we present two simple examples of FWs, the first being given by an ideal constant intensity profile, the second by an also ideal growing exponential, both limited to the range $-0.3L \le z \le 0.3L$ with $L = 1$ mm. The expected longitudinal intensity patterns are seen in Figs. 4 and 5, respectively. They were obtained by superposing $2N + 1 = 31$ BBs, all with the same wavelength (in vacuum) of $\lambda = 1064$ nm. In contrast with the ideal constant and exponential slopes, here we some unavoidable ripples due mainly to the fact that, from (2), a finite sum (31 terms) must be imposed in deriving $F(z)$ [3-5]. These are examples of graphical outputs at the display of the mobile device.

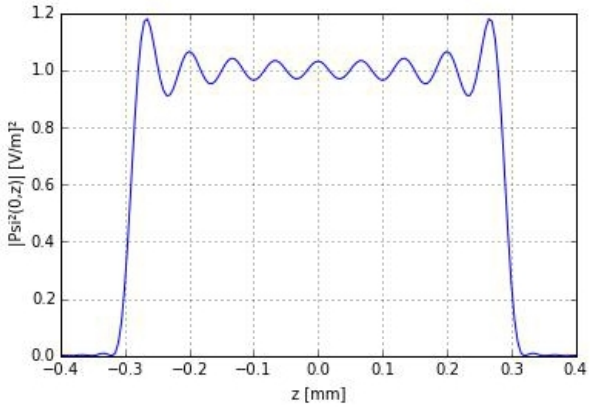

Fig. 4 $|\Psi(0,z)|^2$ for an ordinary FW with a constant longitudinal intensity profile. $L = 1$ mm

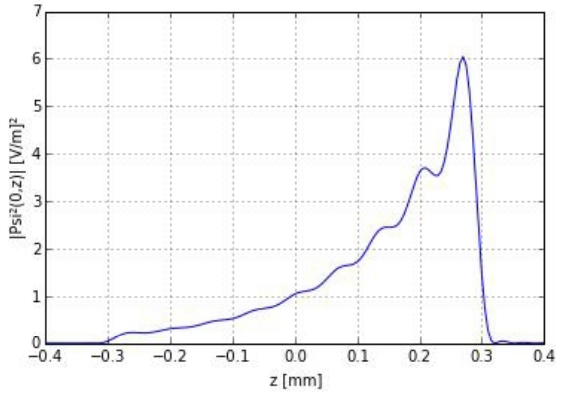

Fig. 5 $|\Psi(0,z)|^2$ for an ordinary FW with a growing exponential profile. $L = 1$ mm

Consider now a dielectric sphere with radius $a = \lambda/50$, i.e., a nano-scatterer of radius ≈ 21.3 nm, immersed in water. The relative permittivity (to water) is taken as 1.1. When incident upon by the FWs shown in Figs. 4 and 5 (and disregarding any Brownian motion effects), the expected longitudinal force component [see (6)] is as shown in Figs. 6 and 7. All special functions were implemented in Python using open-source subroutines.

It is clearly seen in Figs. 6 and 7 that some of the points associated with zero longitudinal force (marked as "A", "B" and so on) indeed correspond to positions of stable longitudinal equilibrium. It can be shown, additionally, that such points do correspond to effective three-dimensional traps because of the presence of radial forces directing the particle towards the optical axis ($r = 0$) of the beam. We intend to comment more on the physical interpretation of those slopes, on the mathematical derivation of FWs and on the numerical code developed in Python and under adaptation for easy-of-use from mobile devices during the conference.

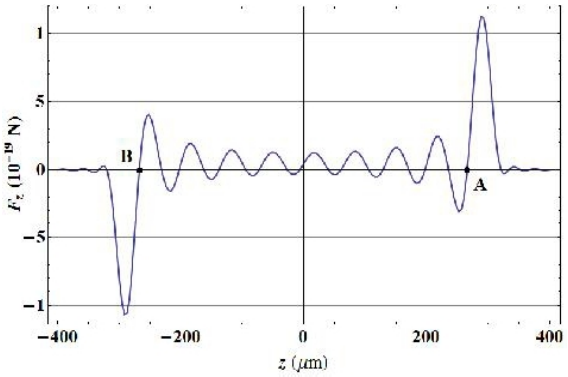

Fig. 6 Longitudinal force ($r = 0$), calculated from $C_{pr,z}$ as given by (6), over a dielectric sphere of radius $a = \lambda/50$ and relative permittivity (to water) of 1.1, assuming the constant pattern FW (Fig. 4). Examples of points of stable equilibrium are marked as "**A**" and "**B**"

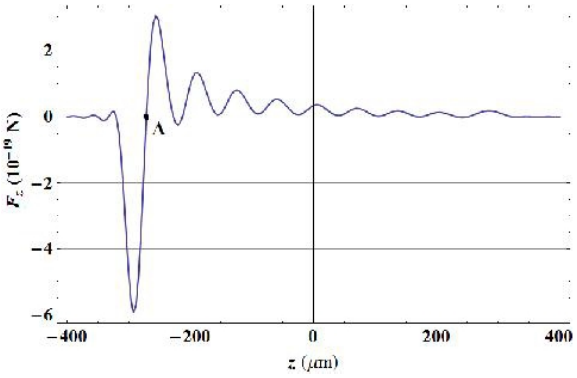

Fig. 7 Same as Fig. 6, but now for the growing exponential FW of Fig. 5

IV. Conclusions

This work is intended to update the usual Fortran numerical modeling in a Python script source code for the evaluation of optical forces exerted over spherical particles from ordinary scalar Bessel beams and their discrete superpositions, streaming data (unknowns, input parameters and so on) from and to mobile devices.

Indeed, the development of such a code is challenging, first because of all the necessary mathematical and physical background encompassing vector calculus, light scattering and

laser beam description by means of the generalized Lorenz-Mie theory, and second because it demands the inclusion of already developed functions (freely available online) in different libraries, such as associated Legendre Polynomial, spherical Bessel functions or Ricatti-Bessel functions and also derivatives with respect to their arguments [31]. Furthermore, some of these already developed functions are not completely suitable for this application, which led us to implement updates and modifications on some of their internal source codes. On the other hand, the adoption of this script language simplifies and automates the graphical results presentation, because it is now in the source code available by the Python Matplotlib library [32].

This back-end data processing solution allowed us to also present our efforts towards a mobile application interface in which the users are able to specify predefined input values and parameters and send them to cloud computing to be processed by this Python solution. Afterwards, these graphical and numerical results are streamed back to the user (mobile device) for visualization purposes.

Although initially intended to provide reliable outputs (optical forces) only for a restrict class of nondiffracting laser beams, our code can be easily modified to account for other wave fields traditionally adopted in optical trapping and micromanipulation, viz., Gaussian beams, top-hat beams, Laguerre-Gaussian beams and so on. Our initial tests have been quite promising, but the entire solution involving Bessel beams and their discrete superpositions is still under development and shall become freely available in the near future.

Acknowledgment

L. A. Ambrosio wishes to thank FAPESP (under process 2014/04867-1) for supporting this work. C. H. Silva-Santos and I. E. L. Rodrigues thank CNPq for the financial studentship. The authors also thank IFSP, campus Itapetininga, and USP by the technical support.

References

[1] H. L. Guo and Z. Y. Li, "Optical tweezers technique and its applications," *Science China Phys. Mech. Astron.*, vol. 56, pp. 2351–2360, December 1993.

[2] P. H. Jones, O. M. Maragò, and G. Volpe, *Optical Tweezers: Principles and Applications*. Cambridge: Cambridge University Press, 2015, Part III.

[3] M. Zamboni-Rached, "Stationary optical wave fields with arbitrary longitudinal shape, by superposing equal frequency Bessel beams: frozen waves," *Opt. Express*, vol. 12, pp. 4001–4006, August 2004.

[4] M. Zamboni-Rached, E. Recami, and H. E. Hernández-Figueroa, "Theory of 'frozen waves': modeling the shape of stationary wave fields," *J. Opt. Soc. Am. A*, vol. 22, pp. 2465–2475, November 2005.

[5] M. Zamboni-Rached, L. A. Ambrosio, and H. E. Hernández-Figueroa, "Diffraction-attenuation resistant beams: their higher-order versions and finite-aperture generations," *Appl. Opt*., vol. 49, pp. 5861–5869, October2010.

[6] T. A. Vieira, M. R. R. Gesualdi, and M. Zamboni-Rached, "Frozen waves: experimental generation," *Opt. Lett.*, vol. 37, pp. 2034–2036, June 2012.

[7] G. Gouesbet, and G. Gréhan, *Generalized Lorenz-Mie Theories*. Berlin: Springer-Verlag, 2011.

[8] J. Durnin, J. J. Miceli, and J. H. Eberli, "Diffraction-free beams," *Phys. Rev. Lett.*, vol. 58, pp. 1499−1501, April 1987.

[9] J. Durnin, "Exact solutions for nondiffracting beams. I. The scalar theory," *J. Opt. Soc. Am. A*, vol. 4, pp. 651−654, April 1987.

[10] V. Garces-Chavez, D. McGloin, H. Melville, W. Sibbett, and K. Dholakia, "Simultaneous micromanipulation in multiple planes using a self-reconstructing light beam", *Nature*, vol. 419, pp. 145−147, September 2002.

[11] D. McGloin, V. Garcés-Chávez, and K. Dholakia, "Interfering Bessel beams for optical micromanipulation," *Opt. Lett.*, vol. 28, pp. 657−659, April 2003.

[12] A. N. Rubinov, A. A. Afanas'ev, I. E. Ermolaev, Y. A. Kurochkin, and S. Y. Mikhnevich, "Localization of spherical particles under the action of gradient forces in the field of a zero-order Bessel beam. Rayleigh-Gans approximation", *J. Appl. Spectr*., vol. 70, pp. 565−572, July 2003.

[13] V. Garces-Chavez, D. Roskey, M. D. Summers, H. Melville, D. McGloin, E. M. Wright, and K. Dholakia, "Optical levitation in a Bessel light beam", *Appl. Phys. Lett*., vol. 85, pp. 4001−4003, November 2004.

[14] G. Milne, K. Dholakia, D. McGloin, K. Volke-Sepulveda, and P. Zemánek, "Transverse particle dynamics in a Bessel beam", *Opt. Express*, vol. 15, pp. 13972−13987, October 2007.

[15] L. A. Ambrosio, and M. Zamboni-Rached, "Analytical approach of ordinary frozen waves for optical trapping and micromanipulation," *Appl. Opt*., vol. 54, pp. 2584−2593, April 2015.

[16] L. A. Ambrosio, and M. de M. Ferreira, "Time-average forces over Rayleigh particles by superposition of equal-frequency arbitrary-order Bessel beams," *J. Opt. Soc. Am. B*, vol. 32, pp. 67−74, May 2015.

[17] L. A. Ambrosio, and M. Zamboni-Rached, "Optical forces experienced by arbitrary-sized spherical scatterers from superpositions of equal-frequency Bessel beams," *J. Opt. Soc. Am. B*, vol. 32, pp. 37−46, May 2015.

[18] L. A. Ambrosio, and H. E. Hernández-Figueroa, "Integral localized approximation description of ordinary Bessel beams and application to optical trapping forces," *Biomed. Opt. Express*, vol. 2, pp. 1893−1906, July 2011.

[19] W. L. Moreira, A. A. R. Neves, M. K. Garbos, T. G. Euser, P. St. J. Russell, and C. L. Cesar, "Expansion of arbitrary electromagnetic fields in terms of vector spherical wave functions," http://arxiv.org/abs/1003.2392v3, May 2012.

[20] P. T. Hacken, "Computer-assisted language learning and the revolution in computational linguistics," *Linguistik online*, vol. 17, pp. 23–39, 2003.

[21] S. van der Walt, S. C. Colbert, and G. Varoquaux. "The NumPy array: a structure for efficient numerical computation," *Comp. Sci. & Eng*., vol. 13, pp. 22–30, March-April 2011.

[22] A. Bultheel, "Book Review: 'Numerical Methods in Engineering with Python 3 (J. Kiusalaas)'," *Euro. Math. Soc*., October 2013.

[23] E. Ayars, "Finally, a Python-Based Computational Physics Text," *Comp. Sci. & Eng*., vol. 16, pp. 6–7, January-February 2014.

[24] F. Perez, B. E. Granger, and J. D. Hunter. "Python: an ecosystem for scientific computing," *Comp. Sci. & Eng.*, vol. 13, pp. 13–21, March-April 2010.

[25] G. Rashed, and R. Ahsan, "Python in computational science: applications and possibilities," *Intl. J. Comp. Appl*., vol. 46, pp. 26–30, May 2012.

[26] Python Software Foundation, available on: https://www.python.org/psf/, accessed on: 12/15/2015.

[27] Python Source Page, available on: https://www.python.org/, accessed on 12/15/2015.

[28] K. Han, P. C. Shih, V. Bellotti, and J. M. Carroll, "It's Time There Was an App for That Too: A Usability Study of Mobile Timebanking," *Intl. J. Mob. Human Computer Interactions*, vol. 7, pp. 1–22, April-June 2015.

[29] J. T. Pintas, D. de Oliveira, K. A. C. S. Ocaña, E. Ogasawara, and M. Mattoso, "SciLightning: A Cloud Provenance-Based Event Notification for Parallel Workflows," *Service-Oriented Computing–ICSOC 2013 Workshops, Lecture Notes in Computer Science*, vol. 8377, pp. 352-365, 2014.

[30] T. Page, "Touchscreen mobile devices and older adults: a usability study," *Intl. J. Human Factors and Ergonomics*, vol. 3, pp. 65–85.

[31] Available on http://docs.scipy.org/doc/scipy/reference/special.html, accessed on 12/29/2015.

[32] J. D. Hunter, "Matplotlib: A 2D graphics environment," *Comp. Sci. & Eng*., vol. 9, 90–95, May-June 2007.