PONTIFÍCIA UNIVERSIDADE CATÓLICA
DO RIO DE JANEIRO

Hélvio de Farias Costa Peixoto

**A Fast Multipole Method for High Order
Boundary Elements**

Tese de Doutorado

Thesis presented to the Programa de Pós–graduação em Engenharia Civil of PUC-Rio in partial fulfillment of the requirements for the degree of Doutor em Engenharia Civil.

Advisor: Prof. Ney Augusto Dumont

Rio de Janeiro
April 2018

Hélvio de Farias Costa Peixoto

A Fast Multipole Method for High Order Boundary Elements

Thesis presented to the Programa de Pós–graduação em Engenharia Civil of PUC-Rio in partial fulfillment of the requirements for the degree of Doutor em Engenharia Civil. Approved by the undersigned Examination Committee.

Prof. Ney Augusto Dumont
Advisor
Departamento de Engenharia Civil e Ambiental– PUC-Rio

Profª. Deane de Mesquita Roehl
Departamento de Engenharia Civil e Ambiental– PUC–Rio

Prof. Raul Rosas e Silva
Departamento de Engenharia Civil e Ambiental– PUC–Rio

Prof. Alvaro Luiz Gayoso de Azeredo Coutinho
Universidade Federal do Rio de Janeiro

Prof. Eduardo Toledo de Lima Júnior
Universidade Federal de Alagoas

Prof. Rodrigo Bird Burgos
Universidade do Estado do Rio de Janeiro

Prof. Márcio da Silveira Carvalho
Vice Dean of Graduate Studies
Centro Técnico Científico– PUC-Rio

Rio de Janeiro, April the 13th, 2018

Hélvio de Farias Costa Peixoto

Graduated in Civil Engineering at Universidade Federal de Alagoas (UFAL) in 2011. Holds a master's degree in Civil Enginering (structures) from the Pontifícia Universidade Católica do Rio de Janeiro (2014). Has been awarded with the state government scholarship for outstanding graduate students both in his master's and PhD's years (FAPERJ Nota 10). Research interests include, but are not limited to: the Boundary Element Method, Computational Mechanics, Fast-Multipole Method and the Discrete Element Method.

Acknowledgments

The elaboration of this thesis, throughout six year, two of which on the Master's, would not be possible without the help and fellowship of countless people which, unfortunately, I will not be able to thank nominally. Thus, I thank the ones below on behalf of the many others that by chance were not cited.

Initially, I thank Marília for all the years of support, friendship and companionship. The person to whom I trully owe the strength and encouragement to complete this thesis.

I thank Prof. Ney Dumont for the vote of confidence by accepting to advisor me during these 6 year at PUC-Rio, and also for all the knowledge which you had the generosity to share with me.

To my uncles Sara and Carlos, and all the other cariocas of the family, I thank you all immensily for the warm welcome over these years, for the love and affection, and also for the countless hours of music on the piano.

To my parents, I thank for the guidance given throughout my whole life, and for the relentless effort to provide me with the best education possible, which culminated in this work.

To all my family, Peixoto and Farias Costa, I thank for all the conversations and for being a safe harbour during my whole life.

My sincere gratitude to my friends Larissa Novelino and Graciele Tanaka, for all your help and companionship, even if from a distance.

I thank the friends Wellington Tatagiba, Rafael Abreu, Adriano Malko, Cesar Ferrante, Renan Costa, Felipe Pinheiro, Alan Borges, Daiana Eringer, Letícia Oliveira, among many others, for all the shared moments inside and outside PUC-Rio. I wish best of luck for you all in your academic career.

To Tecgraf Institute, in the person of Prof. Deane Roehl and all my coworkers, I thank for the opportunity to grow and to contribute to national science.

I also thank all PUC-Rio employees, specially secretaries Rita Leite and Fátima Dinoá, for your dedicated and thoughtfull work.

Finally, I thank PUC-Rio, FAPERJ and CNPq for the financial support.

Abstract

Peixoto, Hélvio de Farias Costa; Dumont, Ney Augusto (Advisor). **A Fast Multipole Method for High Order Boundary Elements**. Rio de Janeiro, 2018. 64p. Tese de doutorado – Departamento de Engenharia Civil e Ambiental, Pontifícia Universidade Católica do Rio de Janeiro.

The Fast Multipole Method (FMM) has been used since the 1990s with the Boundary Elements Method (BEM) for the simulation of large-scale problems. This method relies on Taylor series expansions of the underlying fundamental solutions to cluster the nodes on the discretised boundary of a domain, aiming to reduce the computational time required to carry out the simulation. It has become an important tool for the BEMs, as they present matrices that are full and nonsymmetric, so that the improvement of storage allocation and execution time is not a simple task. The application of the FMM to the BEM ends up with a very intricate code, and usually changing from one problem's fundamental solution to another is not a simple matter. This work presents a kernel-independent formulation of the FMM, here called the General Fast Multipole Method (GFMM), which is also able to deal with high order, curved boundary elements in a straightforward manner. This is an important feature, as the fast multipole implementations reported in the literature only apply to constant elements. All necessary aspects of this method are presented, starting with the mathematical basics of both FMM and BEM, carrying out some numerical assessments, and ending up with the solution of large potential problems. The application of the GFMM to both potential and elasticity problems is discussed and validated in the context of BEM. Furthermore, the formulation of the GFMM with the Simplified Hybrid Boundary Elements Method (SHBEM) is presented. Several numerical assessments show that the GFMM is highly efficient and may be as accurate as arbitrarily required, for problems with up to many millions of degrees of freedom. The literature proposes that the FMM is capable of reducing the time complexity of the BEM algorithms from $\mathcal{O}(N^2)$ to $\mathcal{O}(N)$, where N is the number of degrees of freedom. In fact, it is shown that the GFMM is able to arrive at such time reduction without resorting to techniques of computational optimisation.

Keywords

Resumo

Peixoto, Hélvio de Farias Costa; Dumont, Ney Augusto. **Um Método Fast Multipole para Elementos de Contorno de Alta Ordem**. Rio de Janeiro, 2018. 64p. Tese de Doutorado – Departamento de Engenharia Civil e Ambiental, Pontifícia Universidade Católica do Rio de Janeiro.

Desde a década de 1990, o Método Fast Multipole (FMM) tem sido usado em conjunto com o Métodos dos Elementos de Contorno (BEM) para a simulação de problemas de grande escala. Este método utiliza expansões em série de Taylor para aglomerar pontos da discretização do contorno, de forma a reduzir o tempo computacional necessário para completar a simulação. Ele se tornou uma ferramenta bastante importante para os BEMs, pois eles apresentam matrizes cheias e assimétricas, o que impossibilita a utilização de técnicas de otimização de solução de sistemas de equação. A aplicação do FMM ao BEM é bastante complexa e requer muita manipulação matemática. Este trabalho apresenta uma formulação do FMM que é independente da solução fundamental utilizada pelo BEM, o Método Fast Multipole Generalizado (GFMM), que se aplica a elementos de contorno curvos e de qualquer ordem. Esta característica é importante, já que os desenvolvimentos de fast multipole encontrados na literatura se restringem apenas a elementos constantes. Todos os aspectos são abordados neste trabalho, partindo da sua base matemática, passando por validação numérica, até a solução de problemas de potencial com muitos milhões de graus de liberdade. A aplicação do GFMM a problemas de potencial e elasticidade é discutida e validada, assim como os desenvolvimentos necessários para a utilização do GFMM com o Método Híbrido Simplificado de Elementos de Contorno (SHBEM). Vários resultados numéricos comprovam a eficiência e precisão do método apresentado. A literatura propõe que o FMM pode reduzir o tempo de execução do algoritmo do BEM de $\mathcal{O}(N^2)$ para $\mathcal{O}(N)$, em que N é o número de graus de liberdade do problema. É demonstrado que esta redução é de fato possível no contexto do GFMM, sem a necessidade da utilização de qualquer técnica de otimização computacional.

Palavras-chave

Método fast multipole,   elementos de contorno,   métodos variacionais.

Table of contents

List of figures

Faithless is he that says farewell when the road darkens.

J. R. R. Tolkien, *The Lord of The Rings.*

1
Introduction

The Fast Multipole Method (FMM), initially conceived by Greengard and Rokhlin [1], has been acknowledged as one of the most important algorithms of the last century [2]. Although developed for the simulation of Coulombic and gravitational fields, it has proven to be a powerful tool to be used with the Boundary Element Method (BEM) [3, 4].

The FMM relies on a series expansion of the problem's underlying fundamental solution. It aims to speed up the matrix-vector products that are part of the iterative solution of a linear equation system by reducing the number of product operations from, typically, $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$, or even $\mathcal{O}(N)$[3, 5]. Such order $\mathcal{O}(N)$ has been in fact achieved in this thesis, as presented in Section 7.

The evaluation and manipulation of matrices in the BEM, as classicaly done in the literature [6], may become computationally expensive. These matrices are dense, which prevent the possibility of using algorithms to optimise storage allocation. The classical simulation of large problems, with hundreds of thousands – or even millions – of degrees of freedom, is nearly impossible to carry out in an usual desktop [4].

Several methods have been proposed as an attempt to overcome these issues [7]. The pFFT [8] uses a Fast Fourier Transform to perform the potential computations, while the Adaptive Cross Aproximation (ACA) [9, 10] tries to split the problem's matrices into blocks that may be approximated by low rank submatrices. The FMM takes into account the distance between nodes (poles) in order to cluster them and circumvent the need to directly correlate each field node to each source of the numerically discretised problem [4, 3]. Gholami *et al.* [11] present a thorough comparative study between some of theses methods with up to impressive 600 billions degrees of freedom on a parallel setup.

As reported in the literature [3], one major drawback of the FMM (as compared to the ACA, for instance) is that it requires different expansion patterns for different fundamental solutions, which prevents a code to be generally applicable and may even require modifications of the most basic algorithm of an already existing code. Liu [3] presents several of these developments for 2- and 3D problems of both potential and elasticity fields.

In this thesis, the proposed FMM scheme is independent of the problem's fundamental solution and relies on geometric and topological features that are not seen in the technical literature. The kernel-independent formulation presented herein is actually more simple than the ones already shown in the literature [12, 13] which rely on equivalent density representations of the potentials. It is also different from the Generalized FMM presented by Létourneau *et al.* [14], and the Black-box FMM proposed by Fong & Darve [15], which actually deal with kernels defined numerically.

The Taylor series expansion naturally leads to approximation errors, which are related to the distance between source and field poles about which expansions are carried out. Thus, distance assessment between poles is a key part of the method. Numerous strategies have been used in order to map adjacency in a FMM context [16, 3]. These are appropriate for mapping constant elements (visualised as straight segments, but handled as single points), as they are easily represented by their midpoint. Moreover, dealing with curved, high order elements seems to have been an impossibility in the classical literature on the FMM. This is no longer the case, as the present developments are based on a conceptual improvement of the BEM proposed by Dumont [17] regarding the interpolation normal derivatives (normal fluxes for potentials, and tractions for elasticity). In order to adequately map adjacencies for high order elements (e.g. linear and quadratic), this thesis extends a hierarchical refinement strategy developed by Novelino [18], which works fine for convex domains. The handling of multiply-connected domains is a contribution of this work [19].

The General Fast Multipole Method (GFMM), as presented by Peixoto [20] and initially applied to potential problems by Novelino [18], is further discussed and extended to elasticity in this thesis. The evaluation of results at internal points by Somigliana's identity [21] in the GFMM is also formulated. Finally the application to the Simplified Hybrid Boundary Element Method [22] for both potential and elasticity problems is developed. Several numerical assessments are shown and thoroughly examined.

In Chapter 2 it is presented the formulation of the Consistent Boundary Element Method, its important improvement on the interpolation of tractions, which allows the easy handling of high order elements. Moreover, it is shown the mathematical basis of both the Hybrid Boundary Element Method, and the Simplified Hybrid Element Method.

Chapter 3 discusses the usual application of the Fast Multipole Method to the Boundary Element Method, as proposed by Liu [3].

The General Fast Multipole Method is comprehensively discussed in

Chapter 4, as well as the strategy of hierarchical mesh refinement which is able to determine adjacency of high order elements.

A thorough mathematical application of the General Fast Multipole Method to potential and elasticity problems is presented in Chapter 5 for both the Consistent Boundary Element Method, and the Simplified Hybrid Element Method. Evaluation of internal results through Somigliana's identity is also shown.

Chapter 6 examines the overall computational aspects of the proposed General Fast Multipole Method framework for high order elements through flow charts of pseudo-algorithms.

The numerical assessments in Chapter 7 are intended to validate the formulations presented in Chapter 5.

Finally, Chapter 8 sum up the developments presented in this thesis in order to point out the relevant results achieved, and the contributions given to the literature of both the Boundary Element Method and the Fast Multipole Method.

2
The Boundary Element Method

The Boundary Element Method (BEM) [23] is a powerful tool to solve computational mechanics problems. Since it in principle requires discretisation only along the boundary, it is well suited to represent complex geometries and semi-infinite domains [6].

This chapter presents very briefly the formulation of three different boundary element approaches: the Consistent BEM (CBEM) [17], the Hybrid BEM (HBEM) [24], and the Simplified Hybrid BEM (SHBEM) [22].

The first formulation is basically the same conventional, collocation BEM given in the classical literature, but introducing a previously neglected term of rigid-body displacements [17, 25]. The HBEM is a variational formulation based on the Hellinger-Reissner potential [24, 26, 27]. The SHBEM gives up the full variational consistency of the HBEM in favor of a simplification that may turn out advantageous in terms of computational implementation [22, 25, 27].

The following developments are not intended to be comprehensive, and aim only to lay down the basics of each method and their main features for the sake of application of the GFMM in Chapter 4.

2.1
Consistent formulation of the BEM

The CBEM will be proposed for an elasticity problem, and the relevant aspects for this work will be pointed out when necessary. The development for a potential field problem follows the same procedure as just a particular application.

Let a linear elastic body, with a domain Ω and a boundary Γ, be subjected to body forces b_i in Ω, traction forces t_i on Γ_σ, and displacements u_i on Γ_u ($i = 1, 2$ for 2D problems), with the constitutive relation

$$\sigma_{ij} = C_{ijkl}\epsilon_{ij} \text{ in } \Omega \tag{2-1}$$

where σ_{ij} is the stress tensor, ϵ_{ij} the strain tensor and C_{ijkl} the symmetric elastic tensor.

Assuming small displacements, the equilibrium is given by

$$\sigma_{ij,j} + b_i = 0 \text{ in } \Omega \tag{2-2}$$

with boundary conditions given by

$$\sigma_{ij}\eta_i = \bar{t}_j \text{ in } \Gamma_\sigma \tag{2-3}$$

where Γ_σ is the part of the body's boundary that has prescribed forces.

The weighted residual statement for eqs. (2-2) and (2-3) may be written as

$$-\int_\Omega (\sigma_{ij,j} + b_i)\delta u_i^* \mathrm{d}\Omega + \int_\Gamma (\sigma_{ji}\eta_j - t_i)\delta u_i^* \mathrm{d}\Gamma = 0, \tag{2-4}$$

with the superscript * meaning that the weighting function is a fundamental solution of the problem, that is, it satisfies eq. (2-2) in Ω. By applying integration by parts twice and taking into account Green's theorem [23], one arrives at the characteristic equation of the BEM

$$\int_\Gamma \delta\sigma_{ji}^* \eta_j u_i \mathrm{d}\Gamma - \int_\Omega \delta\sigma_{ij,j}^* u_i \mathrm{d}\Omega = \int_\Gamma t_i \delta u_i^* \mathrm{d}\Gamma + \int_\Omega b_i \delta u_i^* \mathrm{d}\Omega. \tag{2-5}$$

Dumont [17] expresses the variation of the fundamental solutions for displacements δu^* and stresses $\delta\sigma^*$ in terms of arbitrary virtual force parameters δp_m^* as

$$\delta\sigma_{ij}^* = \sigma_{ijm}^* \delta p_m^*, \text{ and} \tag{2-6}$$

$$\delta u_i^* = (u_{im}^* + u_{ic}^r C_{cm})\delta p_m^*, \tag{2-7}$$

where the superscript r stands for rigid-body displacements, C_{cm} are arbitrary constants, and the index m represents the point of application of δp_m^*. Using the definitions of eqs. (2-6) and (2-7), and taking into account the fact that σ_{ij}^* satisfies eq. (2-2), eq. (2-5) may be written as

$$\begin{aligned} u_m = &\int_\Omega b_i \delta u_{im}^* \mathrm{d}\Omega - \int_\Gamma \delta\sigma_{jim}^* \eta_j u_i \mathrm{d}\Gamma + \int_\Gamma t_i \delta u_{im}^* \mathrm{d}\Gamma \\ &+ C_{cm}\left(\int_\Gamma t_i u_{ic}^r \mathrm{d}\Gamma + \int_\Omega b_i u_{ic}^r \mathrm{d}\Omega\right) \end{aligned} \tag{2-8}$$

which one recognises as Somigliana's identity, except for the rigid-body terms, which are void for forces in equilibrium, although this is not the case when approximations are used. This consideration is one of the differences of the CBEM when compared with the BEM.

Now let the displacements and traction forces be approximated, on the boundary Γ as

$$u_i = u_{in}d_n, \text{ and} \tag{2-9}$$

$$t_i = t_{i\ell}t_\ell \tag{2-10}$$

where d_n and t_ℓ are the vectors of nodal displacements and traction forces, and u_{in} and $t_{i\ell}$ are interpolation functions with local support, usually taken as the same for both geometry and displacement interpolations, which constitutes an isoparametric formulation.

The discretisation of eq. (2-8) using eqs. (2-9) and (2-10) leads to [17]

$$
\left(\int_\Gamma \sigma^*_{jim} \eta_j u_{in} \mathrm{d}\Gamma + \delta_{mn} \right) d_n =
$$
$$
\int_\Omega b_i \delta u^*_{im} \mathrm{d}\Omega + \int_\Gamma t_{i\ell} \delta u^*_{im} \mathrm{d}\Gamma t_\ell + C_{cm} \left(\int_\Gamma t_{i\ell} t_\ell u^r_{ic} \mathrm{d}\Gamma + \int_\Omega b_i u^r_{ic} \mathrm{d}\Omega \right).
\tag{2-11}
$$

Writing eq. (2-11) in matrix notation, one recognises the conventional BEM equation, except for the rigid-body vector $\boldsymbol{\epsilon}$

$$
\mathbf{Hd} = \mathbf{Gt} + \mathbf{b} + \boldsymbol{\epsilon},
\tag{2-12}
$$

with

$$
\mathbf{H} = H_{sf} = \int_\Gamma \sigma^*_{jis} \eta_j u_{if} \mathrm{d}\Gamma + \delta_{sf},
\tag{2-13}
$$

$$
\mathbf{G} = G_{s\ell} = \int_\Gamma t_{i\ell} \delta u^*_{is} \mathrm{d}\Gamma,
\tag{2-14}
$$

$$
\mathbf{b} = b_s = \int_\Omega b_i \delta u^*_{is} \mathrm{d}\Omega, \text{ and}
\tag{2-15}
$$

$$
\boldsymbol{\epsilon} = \epsilon_s = C_{cs} \left(\int_\Gamma t_{i\ell} t_\ell u^r_{ic} \mathrm{d}\Gamma + \int_\Omega b_i u^r_{ic} \mathrm{d}\Omega \right),
\tag{2-16}
$$

where indices m and n have been replaced with s and f, respectively, to point out that they refer to *source* and *field* points.

One major improvement introduced by Dumont [17] is the consistent interpolation of traction forces, given in eq. (2-10). In fact, Dumont proposes to interpolate traction forces as

$$
t_i = \frac{|J|_{\mathrm{at}\ \ell}}{|J|} u_{i\ell} t_\ell.
\tag{2-17}
$$

The only difference between eqs. (2-10) and (2-17) is the consideration of the Jacobian J of coordinates transformation. Besides its conceptual consistency [28], this definition proves to be very handy, as in eq. (2-14) the term $\mathrm{d}\Gamma = |J|\,\mathrm{d}\xi$ is simplified by the division of $|J|$ in the traction interpolation functions.

This feature is key for the implementation of the General Fast Multipole Method together with high order elements, as shown in Section 5. This is due to the fact that the integrations of the single-layer potential matrix in eq. (2-14) become polynomial, and may be evaluated analytically, thus diminishing the overall numerical errors.

2.2
Hybrid Boundary Element Method

The Hybrid Boundary Element Method (HBEM), differently from the CBEM, is based on an energy formulation. Dumont [24] proposes to use the Hellinger-Reissner potential

$$-\pi_R(u_i, \sigma_{ij}) = \int_{\Omega} \left[U_0^C(\sigma_{ij}) + (\sigma_{ij,j} + b_i)u_i \right] \mathrm{d}\Omega -$$
$$\int_{\Gamma} u_i \sigma_{ij} \eta_j \mathrm{d}\Gamma + \int_{\Gamma_\sigma} u_i \bar{t}_i \mathrm{d}\Gamma + C = stationary, \tag{2-18}$$

where U_0^C is the complementary strain energy, and C is an arbitrary constant. This formulation depends on the assumption of a stress field (σ_{ij}) in the domain,

$$\sigma_{ij} = \sigma_{ij}^* + \sigma_{ij}^p \tag{2-19}$$

from which a displacement field is obtained:

$$u_i = u_i^* + u_i^p. \tag{2-20}$$

The quantities σ_{ij}^* and u_i^* are taken as the underlying problem's fundamental solutions, eq. (2-2), while σ_{ij}^p and u_i^p are some particular solutions. Using eqs. (2-6), (2-7) and (2-20), the potential of eq. (2-18) may be written as

$$\pi_R(u_i, \sigma_{ij}) = \int_{\Gamma} \left(\sigma_{ij}\eta_j - \bar{t}_i \right) u_i \mathrm{d}\Gamma - \int_{\Omega} U_0^C(\sigma_{ij}) \mathrm{d}\Omega. \tag{2-21}$$

Using eq. (2-1), integrating by parts and applying Green's theorem, the term $\int_{\Omega} U_0^C(\sigma_{ij})\mathrm{d}\Omega$ may be written as

$$\int_{\Omega} U_0^C(\sigma_{ij})\mathrm{d}\Omega = \frac{1}{2}p_m^* F_{mn} p_n^* + p_m^* b_m + \frac{1}{2}\int_{\Omega} \sigma_{ij}^p u_{i,j}^p \mathrm{d}\Omega, \tag{2-22}$$

where $F_{mn} = \int_{\Gamma} \sigma_{jim}^* u_{in}^* \eta_j \mathrm{d}\Gamma$ and $b_m = \int_{\Gamma} \sigma_{jim}^* u_{in}^p \eta_j \mathrm{d}\Gamma$.

The remaining term of eq. (2-21) may also be rewritten by taking into account eqs. (2-19), (2-9) and (2-11) as

$$\int_{\Gamma} \left(\sigma_{ij}\eta_j - \bar{t}_i \right) u_i \mathrm{d}\Gamma = p_n d_n - H_{mn} p_m^* d_n - p_n^p d_n, \tag{2-23}$$

with

$$\mathbf{H} = H_{mn} = \int_{\Gamma} \sigma_{jim}^* \eta_j u_{in}^n \mathrm{d}\Gamma, \tag{2-24}$$

$$\mathbf{p} = p_n = \int_{\Gamma} \bar{t}_i u_i^n \mathrm{d}\Gamma, \text{ and} \tag{2-25}$$

$$\mathbf{p}^p = p_n^p = \int_{\Gamma} \sigma_{ij}^p \eta_j u_{in}^n \mathrm{d}\Gamma. \tag{2-26}$$

By writing the potential of eq. (2-21) in terms of eqs. (2-22) and (2-23), and considering that the potential is stationary, one obtains, in matrix notation,

$$\begin{cases} \mathbf{F}\mathbf{p}^* = \mathbf{H}(\mathbf{d} - \mathbf{d}^\mathbf{p}) \\ \mathbf{H}^\mathbf{T}\mathbf{p}^* = \mathbf{p} - \mathbf{p}^\mathbf{p}. \end{cases} \tag{2-27}$$

where $\mathbf{H}$ is the same kinematic transformation (double-layer potential) matrix defined in eq. (2-13) for the CBEM, and $\mathbf{F}$ is a flexibility matrix, which is symmetric by construction but computationally expensive to evaluate.

For an in-depth assessment of the HBEM, the author recommends Dumont's works [24] and [26], and Oliveira's Master's thesis [27].

2.3
Simplified Hybrid Boundary Element Method

As remarked at the end of the previous section, the evaluation of the flexibility matrix $\mathbf{F}$ in the HBEM is computationally expensive. The Simplified Hybrid Boundary Element Method (SHBEM) [22, 21] was proposed as a means to circumvent the evaluation of the flexibility matrix $\mathbf{F}$, as defined in eq. (2-22).

Instead of the first of eq. (2-27), mechanical consistency assessments justify the use of

$$\mathbf{U}^*\mathbf{p}^* = \mathbf{d} - \mathbf{d}^{\mathbf{p}}, \tag{2-28}$$

where $\mathbf{U}^* = U_{sf}$ is the matrix given by the direct evaluation of the fundamental solutions u^* at the field points. No integrations are required to evaluate the elements of $\mathbf{U}^*$, which is also symmetric.

One apparent drawback of eq. (2-28) is that the fundamental solutions cannot be directly evaluated when s and f refer to the same nodal point. Although such a mathematical impossibility has a sound mechanical basis, the undefined values of $\mathbf{U}^*$ can be obtained indirectly by imposing that, if eq. (2-28) holds, then, by a hybrid contragradient statement [21],

$$\mathbf{U}^{*\mathbf{T}}\mathbf{P} = \mathbf{H}\mathbf{D}, \tag{2-29}$$

where $\mathbf{D}$ and $\mathbf{P}$ are matrices whose columns are respectively nodal displacements and equivalent nodal forces corresponding to some simple solution of the problem (linear displacement or potencial field, for instance). There are usually more equations than unknowns in such an evaluation of the undefined values of $\mathbf{U}^*$ in eq. (2-29), so that least squares is used.

3
The Fast Multipole Method

In 1987, Greengard and Rokhlin [1] presented a novel method to be applied to N-body problems, such as the ones of Coulombic and gravitational fields. The Fast Multipole Method (FMM) turned out to be useful in reducing the amount of floating-point operations in the frame of an iterative solver from $\mathcal{O}(N^2)$ to $\mathcal{O}(N \log N)$ and even $\mathcal{O}(N)$ [1, 3].

The FMM was firstly proposed for the approximation of the potential interaction between two distant particles in terms of a Taylor series expansion about a pole that is close to one of the particles.

Dongarra and Sullivan [2] acknowledged the FMM as one of the top 10 algorithms of the 20th century, alongside other well-known methods such as the Krylov Subspace Iteration and the Fast Fourier Transform. Board and Schulten [29] argue that the reduction in computational effort allowed simulations to be done with a manageable approximation error without disregarding far-field contributions and not at the expense of brute force.

The FMM began to be applied to the BEM in the mid-1990s [30, 31], by recognizing the BEM as an N-body problem. In fact, the evaluation of the matrix products **Hd** and **Gt** in eq. (2-12) have become feasible with the FMM for the application of the BEM to very large scale problems [3].

3.1
The FMM applied to the BEM

This section briefly introduces the developments of the application of the BEM for 2D potential problems, as described by Liu [3].

As already mentioned, the BEM formulation delivers matrices that are dense and nonsymmetric. When a simulation scales up to tens of thousands of degrees of freedom, the storage required by them is enormous, thus limiting the possibility of mesh refinement, or the accurate description of the geometry of the problem. For example, a square matrix with 20000 rows requires approximately 3.0 GB for 64-bit numbers in double precision format. It is then easy to see that the amount of storage for a larger problem rapidly becomes impractical, not mentioning the required calculations to solve an equation system with such a large matrix.

Instead of using direct solvers, one must resort to iterative solvers, aiming to decrease the computational time of the solution, but not circumventing the storage issue. Matrix-free iterative solvers could prove useful in this case, except that they would require the integration of the matrices $\mathbf{H}$ and $\mathbf{G}$ at every step.

None of the previous strategies deals with both storage and computational effort at the same time. The FMM allows the use of matrix-free iterative solvers, such as the Generalized Minimum Residual Method (GMRES) [32], while drastically reducing the number of numerical evaluations. This is achieved by the speedup of the matrix-vector multiplication that is required by the solver, usually written as

$$\mathbf{A}\mathbf{x} = \mathbf{b}, \qquad (3\text{-}1)$$

with the coefficients matrix $\mathbf{A}$ assembled from the rows and columns of matrices $\mathbf{H}$ and $\mathbf{G}$, a right-hand side vector $\mathbf{b}$ of known entries, and the unknowns cast in vector $\mathbf{x}$. Although eq. (3-1) is used to illustrate the problem, it is important to point out once again that matrix $\mathbf{A}$ is never assembled in a matrix-free scheme, such as the one used herein.

Liu [3] starts the developments by defining the fundamental solution u^* used for 2D potential problems, in complex coordinates, as

$$u^*(z, z_0) = -\frac{\ln |z - z_0|}{2\pi}, \qquad (3\text{-}2)$$

with normal fluxes

$$q^*(z, z_0) = \frac{\partial u^*(z, z_0)}{\partial \eta}, \qquad (3\text{-}3)$$

where z and z_0 are a field and a source point, respectively, which will be defined shortly in the next section.

3.1.1
Typical FMM expansions

Let the source point z_0 – together with its expansion poles $z_{L'}$ and z_L –, and the field point z – together with its expansion poles z_c and $z_{c'}$– be given as in fig. 3.1.

The fundamental solution of eq. (3-2) is expanded by a Taylor series about the pole z_c as

$$u^*(z, z_0) = -\frac{1}{2\pi} \left(\ln |z - z_c| - \sum_{i=0}^{\infty} i^{-1} \left(\frac{z - z_c}{z_0 - z_c} \right)^i \right). \qquad (3\text{-}4)$$

While in the original equation eq. (3-2) the source and field points were linked by the natural logarithm, they are now decoupled in eq. (3-4). Equation (3-4) may be conveniently rewritten as
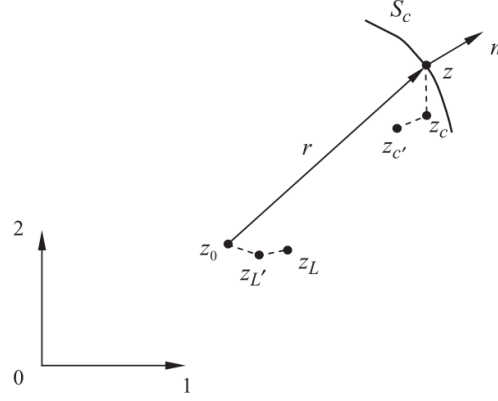
Figure 3.1: Definition of points and expansion poles by Liu [3].

$$u^*(z, z_0) = \frac{1}{2\pi} \sum_{i=0}^{\infty} O_i(z_0 - z_c) I_i(z - z_c), \qquad (3\text{-}5)$$

with the definition of vectors

$$I_i(z) = \frac{z^i}{i!} \text{ for } i \geq 0, \qquad (3\text{-}6)$$

$$O_i(z) = \frac{(i-1)!}{z^i} \text{ for } i \geq 1, \text{ and } O_0 = -\ln z. \qquad (3\text{-}7)$$

Equation (3-5) may be used to evaluate $\mathbf{G}$ in eq. (2-14) as

$$G_{s\ell} = \int_\Gamma u_s^* q_\ell \mathrm{d}\Gamma = \frac{1}{2\pi} \sum_{i=0}^{\infty} O_i(z_{0_s} - z_c) M_i(z_c), \qquad (3\text{-}8)$$

with the introduction of the *moments* about the field pole z_c

$$M_i(z_c) = \int_\Gamma I_i(z - z_c) q_\ell \mathrm{d}\Gamma, \quad i = 0, 1, 2, \ldots, n. \qquad (3\text{-}9)$$

In eqs. (3-8) and (3-9), one may notice that only the vector I requires boundary integration. This is a remarkable feature, as it allows the evaluation of integrals over the boundary, without specifying a source point. Liu [3] also provides the possibility of carrying out the expansion about successive poles. By expanding eq. (3-9), one defines the *moment-to-moment* (M2M) translations as

$$M_i(z_{c'}) = \sum_{j=0}^{i} I_i(z_c - z_{c'}) M_j(z_c). \qquad (3\text{-}10)$$

In order to generalise the expansion possibilities, one may also define the expansions about source poles, the so-called *moment-to-local* (M2L) translations. Expanding eq. (3-8) about a pole z_L, one has that

$$G_{s\ell} = \frac{1}{2\pi} \sum_{i=0}^{\infty} L_i(z_L) I_i(z_{0_s} - z_L), \qquad (3\text{-}11)$$

with

$$L_i(z_L) = (-1)^j \sum_{j=0}^{\infty} O_{i+j}(z_L - z_c)M_j(z_c). \tag{3-12}$$

As in the case of successive expansions about field poles, there is also the possibility of expansions about successive source poles. These are defined as

$$L_j = (z_{L'}) = \sum_{k=0}^{n-j} I_k(z_{L'} - z_L)L_{j+k}(z_L), \tag{3-13}$$

the *local-to-local* (L2L) translations, where n is the number of terms of the desired expansion.

The translations in eqs. (3-9), (3-10), (3-12), and (3-13) are defined for the single-layer matrix **G** only. The developments for translations of the double-layer matrix **H** are presented in [3] and [4].

Liu [3] also provides an algorithm for the solution of a BEM problem using the proposed FMM developments, as carefully described in Section 3.2.8 of his book.

4
The General Fast Multipole Method (GFMM)

A generalisation of the FMM is presented in this section. The extension of the FMM developments to different fundamental solutions, as given by Liu [3], is not straightforward.

The General Fast Multipole Method (GFMM), as proposed by Peixoto [20], and expanded as in Novelino [18], aims to implement a FMM formulation that is as much as possible independent of the fundamental solution. Moreover, different notation and terminology – other than the ones given in the classical literature on FMM – are introduced in order to arrive at a more compact formulation that is also directed to a computational implementation.

4.1
GFMM formulation

The following basic definitions are used in the present developments to represent a general function in the complex domain $f(z)$:

- $z - z_s =$ distance between the source point z_s and the field point z.
- z_{c^k}, $k = 0, 1, 2, \ldots, n_c$: hierarchical levels of poles about which the fundamental solution will be successively expanded for the field point z (then, by definition, $z_{c^0} \equiv z$).
- z_{L^l}, $l = 0, 1, 2, \ldots, n_L$: hierarchical levels of poles about which the fundamental solution will be successively expanded for the source point z_s (by definition, $z_{L^0} \equiv z_s$).

The above definitions of a pole z_{c^k} that is *close* (lower case c) to the field point z and of a pole z_{L^l} that is *local* (upper case L) to the source point z_s follow the notation introduced by Liu [3], depicted in fig. 3.1. In the following developments, each *close* pole z_{c^k} and each *local* pole z_{L^l} are actually array representations of different hierarchical levels of poles, as illustrated in fig. 4.1, where the attached superscripts (here omitted, for simplicity) denote an individual pole in the array.

The expression of a generic fundamental solution $f(z)$ for 2D problems is initially expanded about the *close* pole $z_{c^{n_c}}$ (of highest level, as developed next) using n terms:

$$f(z - z_s) = \sum_{i=0}^{n} \frac{1}{i!}(z - z_{c^{n_c}})^i D^{(i)} f(z_{c^{n_c}} - z_s) + O(z - z_{c^{n_c}})^{n+1} \qquad (4\text{-}1)$$

where $D^{(0)} f(z) = f(z)$ and $D^{(i)} f(z) = \partial^i f(z)/\partial z^i$.

The truncated form of eq. (4-1) is conveniently written as

$$f(z - z_s) = \sum_{i=1}^{n+1} \frac{1}{(i-1)!} P_i(z - z_{c^{n_c}}) Q_i(z_{c^{n_c}} - z_s) \qquad (4\text{-}2)$$

for truncation order $O(z - z_{c^{n_c}})^{n+1}$, proportional to the distance between z and z_s and with the arrays of functions $P(z)$ and $Q(z)$ defined for a generic argument z as

$$P(z) = \left\{ \begin{array}{ccccccc} 1 & z & z^2 & z^3 & \cdots & z^{n+1} \end{array} \right\}, \text{ and} \qquad (4\text{-}3)$$

$$Q(z) = \left\{ \begin{array}{ccccc} f(z) & \frac{\partial f(z)}{\partial z} & \frac{\partial f^2(z)}{\partial z^2} & \frac{\partial f^3(z)}{\partial z^3} & \cdots & \frac{\partial f^{n+1}(z)}{\partial z^{n+1}} \end{array} \right\}. \qquad (4\text{-}4)$$

Expansions about the source point are also possible. Let the derivatives $D^{(i)} f(z_{c^{n_c}} - z_s)$ in eq. (4-1) be also expanded for the source point z_s about the *local* point $z_{L^{n_L}}$ (of highest level, as to be also shown subsequently) using m terms:

$$D^{(i)} f(z_{c^{n_c}} - z_s) =$$
$$\sum_{j=0}^{m} \frac{1}{j!}(z_{L^{n_L}} - z_s)^j D^{(i+j)} f(z_{c^{n_c}} - z_{L^{n_L}}) + O(z_{L^{n_L}} - z_s)^{m+1}. \qquad (4\text{-}5)$$

Substituting for $D^{(i)} f(z_{c^{n_c}} - z_s)$ in eq. (4-1) according to above results in

$$f(z - z_s) = \sum_{i=0}^{n} \frac{1}{i!}(z - z_{c^{n_c}})^i \sum_{j=0}^{m} \frac{1}{j!}(z_{L^{n_L}} - z_s)^j D^{(i+j)} f(z_{c^{n_c}} - z_{L^{n_L}})$$
$$+ O(z - z_{c^{n_c}})^{n+1} + O(z_{L^{n_L}} - z_s)^{m+1}. \qquad (4\text{-}6)$$

The truncated form of eq. (4-6) is conveniently written as

$$f(z - z_s) =$$
$$\sum_{i=1}^{n+1} \frac{1}{(i-1)!} P_i(z - z_{c^{n_c}}) \sum_{j=1}^{m+1} \frac{1}{(j-1)!} P_j(z_{L^{n_L}} - z_s) Q_{i+j-1}(z_{c^{n_c}} - z_{L^{n_L}}) \qquad (4\text{-}7)$$

for truncation order given by

$$\max\left(|(z - z_{c^{n_c}})/(z - z_s)|^{n+1}, |(z_{L^{n_L}} - z_s)/(z - z_s)|^{m+1} \right).$$

Equation (4-2) is undoubtedly more economical than eq. (4-7) and seems to be the simplest and fastest way of handling the expansions. However, eq. (4-7) is given as the starting point for a general procedure that leads

to a computationally fast and economical evaluation of a given fundamental solution $f(z - z_s)$ for a very large number of source points z_s and of field points z by means of an approximate expression that can be as accurate as required. The expansion about successive levels of poles is dealt with in the next section.

As a matter of illustration, three expansion schemes are shown in fig. 4.1. They depict the cases where several field points are expanded about a pole z_c (upper left), field point expansions are undertaken about three successive layers of poles z_c (lower left) and, moreover, expansions of source points are carried out about two layers of poles z_L (upper right). The green lines represent analytical integrations of the BEM matrices for the matrix-vector products, as proposed in Peixoto [20].
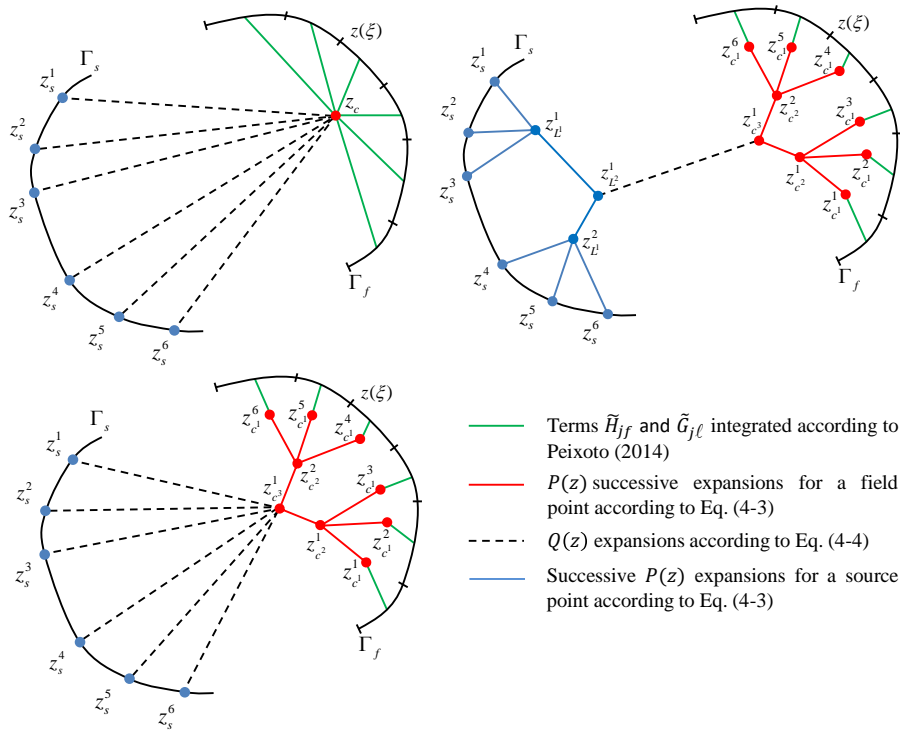
Figure 4.1: Schematic representations of expansions about field and source poles.

The definition of the functions $P_i(z)$ and $Q_i(z)$ – in eqs. (4-3) and (4-4) – follows closely the definitions in eqs. (3-6) and (3-7) of I and O by Liu [3], although without the factorials that are present there. Observe the mnemonic appeal of P (for polynomials, actually binomials) and Q (for quotients, as in the case of the expansion of a logarithm – the simplest fundamental solution conceivable [28, 33]).

Although these latter functions may be computationally intensive to evaluate, they are only needed for the array of poles represented by z_{L^l} interacting with the array of poles represented by z_{c^k}. Then, the evaluation

of $Q_i(z_{L^{n_l}} - z_{c^{n_k}})$ ends up, depending on the numerical implementation, orders of magnitude less intensive than the direct evaluation of $f(z - z_s)$ for all source and field points – and this is the basis of the Fast Multipole Method.

4.1.1
Expansions about successive poles

The next developments are somewhat analogous to the M2M and L2L translations presented by Liu [3] – discussed in Section 3.1 –, and shall lead to the complete generalization of the fast multipole technique. Figure 4.1, in its upper-right corner, shows three levels of close poles (with one, two and four one poles on each level) and two levels of local poles (with one and two poles on each level). This generalisation is at least in part required because, for a closed boundary, $z_{c^3}^1$ may be sufficiently far only from a certain subboundary of Γ_s. Then, hierarchical levels of poles must be implemented, which are sufficiently far from certain subboundaries while building up the contribution for higher-level poles.

The superscripts n_c and n_L in eq. (4-7) turn out to be the highest levels of field and source poles for an expansion to be carried out hierarchically about successive layers of poles. Moreover, the binomials $P_i(\cdot)$ introduced in eq. (4-7) are actually to be directly evaluated according to their definition in eq. (4-3) only when the argument refers to a coordinate difference for two consecutive levels of poles, such as $z_{c^{k-1}} - z_{c^k}$, $k = 1, \cdots, n_c$ in the definition of $P_i(z - z_{c^k})$, or $z_{L^l} - z_{L^{l-1}}$, $l = 1, \cdots, n_L$ in the definition of $P_j(z_{L^{n_L}} - z_s)$. When the argument of $P_i(\cdot)$ refers to poles that are not on consecutive layers, it is expanded in terms of a lower-level pole $z_{c^{l-1}}$ exactly as

$$P_i(z - z_{c^l}) = \sum_{j=1}^{i} C_{j,i+1-j} P_j(z - z_{c^{l-1}}) P_{i+1-j}(z_{c^{l-1}} - z_{c^l}) \qquad (4\text{-}8)$$

where

$$C_{ij} = \begin{cases} 1 & \text{if } i = 1 \text{ or } j = 1 \\ C_{i-1,j} + C_{i,j-1} & \text{else} \end{cases} \qquad (4\text{-}9)$$

In eq. (4-8), $P_{i+1-j}(z_{c^{l-1}} - z_{c^l})$ is defined as in eq. (4-3), since it refers to two consecutive poles. On the other hand, $P_j(z - z_{c^{l-1}})$ is recursively evaluated according to the same eq. (4-8) until the lowest level $P_j(z - z_{c^1})$ is reached and eq. (4-3) becomes directly applicable. With this recursive approach, the binomials $P_i(z - z_{c^{n_k}})$ in eq. (4-7) – and, similarly, $P_j(z_{L^{n_l}} - z_s)$ – are ultimately expressed in terms of arguments given as differences of poles on two consecutive levels.

As shown, $f(z - z_s)$ is expanded in terms of successive arrays of source poles z_{L^l} as well as field poles z_{c^k}. The indicated poles z_{L^l} and z_{c^k} are

representative of arrays of poles corresponding to hierarchical expansion levels l and k. The expansion ends up with a series of products of binomials $P_i(z - z_{c^{n_k}})$ and $P_i(z_s - z_{L^{n_l}})$ – which are independent from the complexity of a problem's fundamental solution – multiplied by functions $Q_i(z_{L^{n_l}} - z_{c^{n_k}})$ that, although eventually cumbersome to obtain numerically, refer to the least number of higher-order pairs of pole arrays $z_{L^{n_l}} - z_{c^{n_k}}$. In the illustration on the upper right of fig. 4.1, there is only one pair of poles $z_{L^2}^1 - z_{c^3}^1$ for which, in the case, $Q_i(z_{L^2}^1 - z_{c^3}^1)$ needs to be evaluated.

The hierarchical expansions described above are next implemented in a boundary element code. It has turned out that the hierarchical development of source poles is not of advantage, at least as presently implemented (Novelino, 2015), so that the scheme shown on the lower left of fig. 4.1 is the actual procedure to be assessed in the numerical examples.

4.2
Hierarchical mesh refinement

The expansions described in the last section give the basic outline for the GFMM formulation. However, as pointed-out in eqs. (4-1) and (4-6), the accuracy of GFMM's expansions is proportional to the distance between source and field points themselves, and their expansion poles. Thus, a strategy to determine adjacency is of utmost importance in order to implement a generally applicable FMM algorithm.

The literature presents many possibilities to handle adjacencies. Liu [3] resorts to a tree structure – a quad-tree in 2D – which is fairly able to assess the relative distances between boundary nodes, and their expansion poles, when dealing with constant elements. Yasuda and Sakuma [16] propose a hierarchical cell structure to be embedded in a conventional BEM algorithm, which also relies on a tree structure, and, again is particularised to constant elements.

These elements are geometrically represented as line segments, but are actually defined by the segment's midpoint. Thus, any adjacency search strategy that deals with points is able to deliver the neighbourhood information required. However, should the order of the elements be higher than constant, these algorithms demand adaptations, which are not straightforward, or may even be not applicable.

As one of the goals of the GFMM is to be as general as possible, it demands an adjacency search able to be applied to elements of any order. The applicability of the GFMM to linear and quadratic elements has been shown both by Peixoto [20] and Novelino [18], and the generalisation to higher-order elements should be simple. A hierarchical mesh refinement and topological

numbering of the elements is proposed in [18, 34] for high order elements, as long as the assessed domain is convex.

This mesh refinement is based on a scheme for splitting elements into two others of the same order and the same geometrical shape. Thus, a refined element inherits from its parent element all geometrical properties under the assumption that the initial refinement level already describes the boundary's shape, although additional degrees of freedom are still required to achieve an accurate solution.

The refinement shown in fig. 4.2 illustrates the numbering rule for successively refined elements and nodes starting from a given macro element. This numbering scheme is the basis of a topological structure that enables the identification of adjacent elements at each refinement level, as stored during runtime only for each uppermost macroelement, which drastically saves storage allocation. Thus, the distance between elements is merely described by how far their numbering is from each other in a given refinement level.
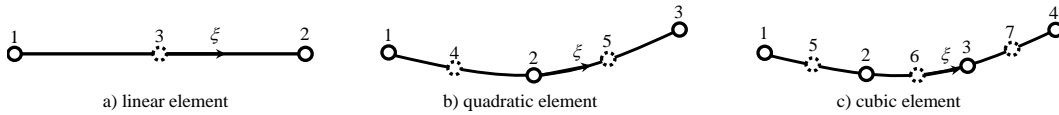
Figure 4.2: Schemes for splitting a general element into two sub-elements [18].

The concept of refinement levels is illustrated in fig. 4.3, which also shows three out of many possibilities for the number of split elements. This strategy naturally creates the near expansion poles required by the FMM algorithm, as the elements on a given refinement level are always linked to its parent element on the previous level.

In order to simulate multiply-connected or irregularly shaped domains, it is proposed an extension of the hierarchical mesh refinement strategy given in [18]. Aiming to avoid unnecessary distance evaluations, this strategy takes advantage of the adjacency information from previous refinement levels, and only assesses the distance between elements with common adjacent parents.

Figure 4.4 shows a square domain with a hole to be assessed at two different refinement levels. If the topological adjacency were to be considered in such a domain for the refinement level $k = 0$, on the left, element 5 would not be detected as adjacent to element 1, as they are 4 elements apart. This illustrates a case that requires a geometry-based assessment. Using the hierarchical refinement, it is possible to deliver the adjacency information of a given element at level k to its children, split elements at level $k + 1$. This information is used to reduce the number of possible adjacent nodes, and therefore, the number of distance evaluations.
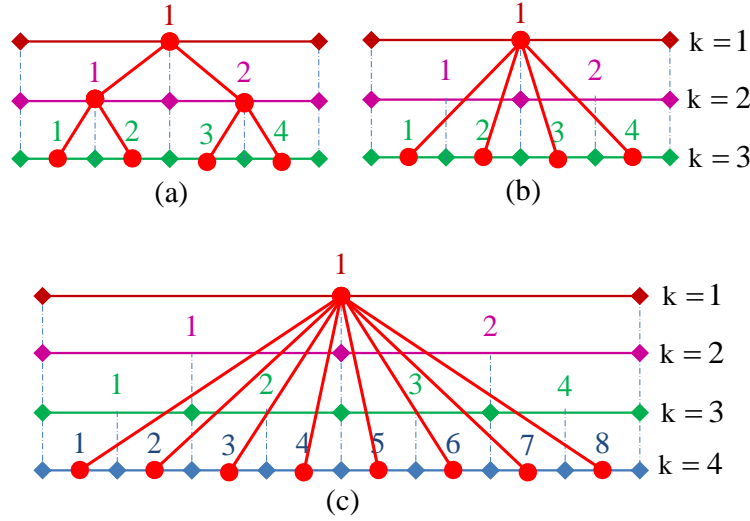
Figure 4.3: Schematic pole expansions using numbers of children poles $n_c = 2, 4$ or 8 (constant elements).

As illustrated on the left of fig. 4.4 for the level $k = 0$, a search is carried out for element 1 using the two green circles centered on its nodes, and it comes out that the nodes marked as blue (corners of the square hole) are adjacent. Then, any element that contains at least one of theses nodes is considered adjacent to element 1. The search radius is $AdjTOL \times L$, where L is the length of the reference element. Good numerical results have been obtained in the frame of the implemented fast multipole algorithm using $0.7 \leq AdjTOL \leq 2$.
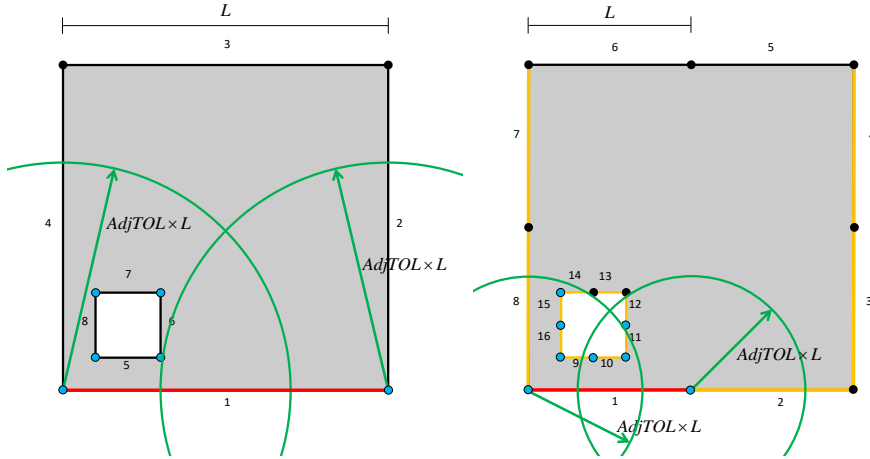
Figure 4.4: Schemes for the adjacency search at the coarser refinement level $k = 0$ (left) and at the next refinement level $k = 1$ (right).

The red element on the right of fig. 4.4 corresponds to a level $k = 1$ and has been generated from element 1 at level $k = 0$. Since the adjacent elements of element 1 at level $k = 0$, on the left figure, are already known, they are the candidates to have adjacent child elements at level $k = 1$, as shown in yellow.

Once more, search circles with radii proportional to the element length are drawn and nodes inside them are marked as adjacent.

An element's adjacency list is built as the hierarchical mesh refinement proceeds up to the highest level. This list is generated and stored for just one element at a given refinement level.

5
GFMM application to the CBEM and SHBEM

This chapter presents the mathematical developments required to apply the GFMM to both potential and elasticity problems with the CBEM and the SHBEM. As eqs. (4-2) and (4-7) are defined for functions in the complex plane, while the definitions in Section 2 are developed for Cartesian coordinates, it is necessary to define complex counterparts of the main equations of the BEM methods to be studied in a GFMM framework.

5.1
Developments for potential problems

Herein is formulated the application of the GFMM to both the CBEM and the SHBEM for high order elements, instead of the usual constant elements seen in the literature [3, 4].

5.1.1
Complex counterpart of BEM fundamental solutions for potential problems

In the following equations z_s is the reference coordinates of a source point (as conceptualised in the frame of a boundary element development) and the difference of a field point z to the source point is $z - z_s$ [35]. However, just for the sake of notation simplicity we temporarily assume $z_s = 0$ as the reference coordinates origin.

The fundamental solution for a potential problem may be written in complex notation as

$$\theta_s^* = \Re\left(\frac{-1}{2\pi k}\ln(z)\right) \equiv \Re\left(\theta_s^C\right) \tag{5-1}$$

with $z = x + iy$ and including a material property k, such as a conductivity parameter in a stationary heat propagation analysis for a homogeneous and isotropic domain. One checks that

$$\frac{-1}{2\pi k}\ln(z) = \frac{-1}{2\pi k}\ln(r) + i\arctan(y, x), \tag{5-2}$$

where $r = \sqrt{x^2 + y^2}$ is the distance between source and field points. Since

$$\frac{\mathrm{d}\ln(z)}{\mathrm{d}z} = \frac{1}{z} \quad \equiv \quad \frac{1}{x+iy} = \frac{x}{r^2} - i\frac{y}{r^2}, \tag{5-3}$$

the potential flux in the Cartesian coordinates is defined as

$$
\begin{aligned}
q_x &= -k\frac{\partial\theta_s^*}{\partial x} = -\frac{x}{r^2} \equiv -\Re\left(k\frac{d\theta_s^C}{dz}\right) \\
q_y &= -k\frac{\partial\theta_s^*}{\partial y} = -\frac{y}{r^2} \equiv -\Im\left(k\frac{d\theta_s^C}{dz}\right),
\end{aligned}
\tag{5-4}
$$

or, in complex notation,

$$q = q_x + iq_y \equiv -k\frac{d\theta_s^C}{dz}. \tag{5-5}$$

The complex expression of the boundary unit vector is

$$\eta = \eta_x + i\,\eta_y \equiv \frac{1}{|J|}\left(\frac{dy}{d\xi} - i\frac{dx}{d\xi}\right) \tag{5-6}$$

and the expression of the boundary normal flux q_n becomes

$$q_n = -q_x\eta_x - q_y\eta_y = \frac{\partial\theta_s^*}{\partial x}\eta_x + \frac{\partial\theta_s^*}{\partial y}\eta_y \equiv \Re\left(\frac{d\theta_s^C}{dz}\eta\right). \tag{5-7}$$

5.1.2
Complex counterpart of BEM matrices for potential problems

The double-layer and single-layer potential matrices **H** and **G** are

$$H_{sf} = \Re\left(H_{sf}^C\right), \text{ and} \tag{5-8}$$

$$G_{s\ell} = \Re\left(G_{s\ell}^C\right), \tag{5-9}$$

with

$$
\begin{aligned}
H_{sf}^C &= k\int_\Gamma \frac{\partial\theta_s^C(z_f - z_s)}{\partial z(\xi)}\eta(\xi)u_f(\xi)\mathrm{d}\Gamma(\xi) \\
&= \frac{-1}{2\pi}\int_\Gamma \frac{1}{z_f - z_s}\eta(\xi)u_f(\xi)\mathrm{d}\Gamma(\xi)
\end{aligned}
\tag{5-10}
$$

$$G_{s\ell}^C = \int_\Gamma \theta_s^C(z_\ell - z_s)q_\ell(\xi)\mathrm{d}\Gamma(\xi) = \frac{-1}{2\pi k}\int_\Gamma \ln(z_\ell - z_s)q_\ell(\xi)\mathrm{d}\Gamma(\xi). \tag{5-11}$$

Equations (5-10) and (5-11) are the complex counterparts of eqs. (2-13) and (2-14) for potential problems.

Integration over a general segment in the frame of the CBEM is carried out considering that both potential θ and normal flux q are interpolated along a boundary segment according to the real functions

$$\theta = u_f(\xi)d_f, \text{ and} \tag{5-12}$$

$$q = t_\ell(\xi)q_\ell, \qquad (5\text{-}13)$$

where

$$t_\ell = u_\ell|J|_{at\ \ell}/|J|, \qquad (5\text{-}14)$$

with $u_f(\xi)$ and $u_\ell(\xi)$ standing for the same type of real polynomial interpolation functions of any order (constant, linear and quadratic functions are implemented in our code).

Since $u_f(\xi)$ and $u_\ell(\xi)$ have local support, there is no need to make explicit that the integrations indicated in the evaluation of H_{sf} and $G_{s\ell}$ must be carried out segment by segment along the boundary.

5.1.3
Formulation for the CBEM

The application of the GFMM to 2D potential problems using the CBEM is developed in [20, 18, 34] and will be briefly outlined in this section.

Let eq. (4-2) be applied to eq. (5-9) with $f(z - z_s) = \theta_s^C$, then

$$G_{s\ell}^C = \frac{-1}{2\pi k} \int_\Gamma q_\ell(\xi) \sum_{i=1}^{n+1} \frac{1}{(i-1)!} P_i(z(\xi) - z_{c^{nc}}) Q_i(z_{c^{nc}} - z_s) \mathrm{d}\Gamma(\xi), \qquad (5\text{-}15)$$

without the need to define *a priori* the vector Q. It must be specified only when the fundamental solution of the potential problem is determined. Thus, eq. (5-15) describes any potential problem in 2D, as long as its fundamental solution θ^C has a valid series representation.

Taking into account that in eq. (5-15) Q is constant on Γ – as its arguments are expansion poles which do not lay on the domain's boundary, one may further write

$$G_{s\ell}^C = \frac{-1}{2\pi k} \sum_{i=1}^{n+1} \frac{1}{(i-1)!} Q_i(z_{c^{nc}} - z_s) \int_\Gamma q_\ell(\xi) P_i(z(\xi) - z_{c^{nc}}) \mathrm{d}\Gamma(\xi). \qquad (5\text{-}16)$$

The integrals in eq. (5-16) may be analytically evaluated, as they are merely the integration of a multiplication of polynomials, if the fluxes q_ℓ are interpolated as proposed in eq. (5-13). Moreover, eq. (5-15) also shows that integrations are not required for vector Q, that usually is composed of transcendental functions, which are not easily integrated by numerical means. Also, the integrations do not depend on a source point z_s, and may be evaluated for a cluster of field points z ultimately expanded about a pole $z_{c^{nc}}$.

The same GFMM expansion of eq. (4-2) applied to eq. (5-10) leads to

$$H_{sf}^C =$$
$$\frac{-1}{2\pi} \int_\Gamma \eta(\xi) u_f(\xi) \sum_{i=1}^{n+1} \frac{1}{(i-1)!} P_i(z(\xi) - z_{c^{nc}}) Q_{i+1}(z_{c^{nc}} - z_s) \mathrm{d}\Gamma(\xi), \qquad (5\text{-}17)$$

which can be written as

$$H_{sf}^C =$$
$$\frac{-1}{2\pi} \sum_{i=1}^{n+1} \frac{1}{(i-1)!} Q_{i+1}(z_{c^{nc}} - z_s) \int_\Gamma \eta(\xi) u_f(\xi) P_i(z(\xi) - z_{c^{nc}}) d\Gamma(\xi), \tag{5-18}$$

by once again taking advantage of the fact that Q is constant over Γ. As in the case of eq. (5-16), these integrals may be analytically evaluated as they are composed by polynomials. Notice that the indices of vector Q in eq. (5-18) are shifted by 1 when compared to eq. (5-16). This is due to the fact that the function to be expanded, in the case of the double-layer potential matrix, is actually the first derivative of the fundamental solution θ^C, which is represented by the second component in the definition of vector Q in eq. (4-4).

All boundary integrals may be obtained analytically in the proposed GFMM framework for generally curved, high order elements [20, 18] – near-field evaluations are carried out as usually in the conventional CBEM [36].

5.1.4
Formulation for the SHBEM

As discussed in Section 2.3, the SHBEM relies on the same double-layer potential matrix $\mathbf{H}$, presented in eq. (2-27), and on the matrix $\mathbf{U}^* = U_{sf}^* = \theta_{sf}^*$. In order to be able to apply the GFMM to the transpose of $\mathbf{H}$, it will be required to apply eq. (4-7), instead of eq. (4-2) as done in the last Section (eq. (5-17)), leading to

$$H_{fs}^C = \frac{-1}{2\pi} \int_\Gamma \eta(\xi) u_f(\xi) \sum_{i=1}^{n+1} \frac{1}{(i-1)!} P_i(z - z_{c^{nc}})$$
$$\sum_{j=1}^{m+1} \frac{1}{(j-1)!} P_j(z_{L^{n_L}} - z_s) Q_{i+j-1}(z_{c^{nc}} - z_{L^{n_L}}) d\Gamma(\xi), \tag{5-19}$$

which can be written as,

$$H_{fs}^C = \frac{-1}{2\pi} \sum_{j=1}^{m+1} \frac{1}{(j-1)!} P_j(z_{L^{n_L}} - z_s) Q_{i+j-1}(z_{c^{nc}} - z_{L^{n_L}})$$
$$\sum_{i=1}^{n+1} \frac{1}{(i-1)!} \int_\Gamma \eta(\xi) u_f(\xi) P_i(z - z_{c^{nc}}) d\Gamma(\xi). \tag{5-20}$$

The integrals in eq. (5-20) may be analytically integrated and are exactly the same as in eq. (5-18) for the CBEM.

The application of the GFMM to the matrix $\mathbf{U}^*$ is straightforward and consists only in directly using any of eqs. (4-2) or (4-6) while taking $f(z - z_s) = \theta_s^C$, as

$$U_{sf}^* = \sum_{i=1}^{n+1} \frac{1}{(i-1)!} P_i(z_f - z_{c^{nc}}) Q_{i+1}(z_{c^{nc}} - z_s), \tag{5-21}$$

or

$$U_{sf}^* = \sum_{i=1}^{n+1} \frac{1}{(i-1)!} P_i(z_f - z_{c^{nc}})$$
$$\sum_{j=1}^{m+1} \frac{1}{(j-1)!} P_j(z_{L^{n_L}} - z_s) Q_{i+j-1}(z_{c^{nc}} - z_{L^{n_L}}), \tag{5-22}$$

respectively.

5.1.5
Results at internal points

The formulation developed up to now refers to a source point located on the boundary. In the frame of the CBEM, potential results at internal points are obtained by applying the Somigliana's identity [17], as presented in Section 5.2.5) for elasticity. For potential problems, it is written as

$$\theta_m = \int_\Gamma q_i \theta_{im}^* \mathrm{d}\Gamma - \int_\Gamma q_{im}^* \eta_i u_i \mathrm{d}\Gamma, \tag{5-23}$$

where m are the internal points of interest, and i are the degrees of freedom along the boundary, disregarding eventual body sources.

Since the integrals in eq. (5-23) are the same ones of matrices H_{sf} and $G_{s\ell}$ presented in eqs. (5-10) and (5-11), the previous developments are directly applicable by just replacing source points s with internal points m, and field points f with degrees of freedom i. The application of the GFMM to eq. (5-23) is then straightforward and requires only the adjustment of indices in eqs. (5-16) and (5-18).

However simple this application may seem, it actually poses as a very efficient application of the GFMM, as Somigliana's identity requires integration over all boundary elements for each internal point. Internal points that are sufficiently far from a cluster of elements of the boundary, may take advantage of a single Fast Multipole integration for all of them.

5.2
Developments for elasticity problems

The application of the GFMM for elasticity problems for both the CBEM and the SHBEM is discussed in this subsection. In principle, elements of arbitrary order (quadratic, cubic, and so on) may be used in the context of this formulation.

5.2.1
Complex counterpart of BEM fundamental solutions for elasticity

The following developments are classical in the literature and thoroughly covered in [35, 37, 38].

Let the displacements $u(z) \equiv u_x(x, y) + i\, u_y(x, y)$ be expressed as a function of the complex potentials $\varphi(z)$ and $\psi(z)$ as

$$u^*(z) = \frac{1}{2G}\left((3 - 4\nu)\varphi(z) - z\overline{\varphi'(z)} - \overline{\psi(z)}\right). \tag{5-24}$$

The stresses are obtained from

$$\sigma^*_{xx} + \sigma^*_{yy} = 4\Re\varphi'(z), \text{ and} \tag{5-25}$$

$$\sigma^*_{yy} - \sigma^*_{xx} + 2i\sigma^*_{xy} = 2\left(\bar{z}\varphi''(z) + \psi'(z)\right), \tag{5-26}$$

which leads to

$$\sigma^*_{xx} = \Re\left(2\varphi'(z) - \bar{z}\varphi''(z) - \psi'(z)\right), \tag{5-27}$$

$$\sigma^*_{yy} = \Re\left(2\varphi'(z) + \bar{z}\varphi''(z) + \psi'(z)\right), \text{ and} \tag{5-28}$$

$$\sigma^*_{xy} = \Im\left(\bar{z}\varphi''(z) + \psi'(z)\right). \tag{5-29}$$

Given the boundary unit vector η, defined in eq. (5-6), the boundary traction forces are

$$\begin{aligned}
T_x &= \sigma^*_{xx}\eta_x + \sigma^*_{xy}\eta_y \equiv 2\Re\left(\varphi'(z)\right)\Re\left(\eta\right) - \Re\left(\left(\bar{z}\varphi''(z) + \psi'(z)\right)\eta\right)\\
T_y &= \sigma_{xy}\eta_x + \sigma_{yy}\eta_y \equiv 2\Re\left(\varphi'(z)\right)\Im\left(\eta\right) + \Im\left(\left(\bar{z}\varphi''(z) + \psi'(z)\right)\eta\right),
\end{aligned} \tag{5-30}$$

which may be compactly written as

$$T \equiv T_x + i\, T_y = 2\Re\left(\varphi'(z)\right)\eta - \overline{\left(\left(\bar{z}\varphi''(z) + \psi'(z)\right)\eta\right)}. \tag{5-31}$$

For the specific case of Kelvin's fundamental solution, the potential functions $\varphi(z)$ and $\psi(z)$ are

$$\varphi(z) = \frac{-p^*}{8\pi\left(1 - \nu\right)}\ln\left(z\right) \tag{5-32}$$

$$\psi(z) = \frac{\left(3 - 4\nu\right)\bar{p}^*}{8\pi\left(1 - \nu\right)}\ln\left(z\right) \tag{5-33}$$

where $p^* = p^*_x + ip^*_y$ is a point force applied at a source point z_s. Applying these potentials to eq. (5-24) leads to

$$u^* = \frac{-1}{8\pi G\left(1 - \nu\right)}\left[\left(3 - 4\nu\right)p^*\Re\ln\left(z\right) - \frac{1}{2}\overline{\left(p^*\frac{\bar{z}}{z}\right)}\right] \tag{5-34}$$

which coincides exactly with the solution arrived at in a displacement formulation using a real potential function of the distance r from a field point to

the source, force point. The only negligible difference consists in the explicit inclusion of a rigid-body term in the displacement expressions, coming from eqs. (5-32) and (5-33):

$$\begin{Bmatrix} u_x^* \\ u_y^* \end{Bmatrix} = \frac{-1}{8\pi G(1-\nu)} \left[\frac{3-4\nu}{2}\ln(r^2)\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} - \frac{1}{r^2}\begin{bmatrix} x^2 & xy \\ xy & y^2 \end{bmatrix} + \frac{1}{2}\begin{bmatrix} 1 & 0 \\ 0 & 1 \end{bmatrix} \right] \begin{Bmatrix} p_x^* \\ p_y^* \end{Bmatrix}. \tag{5-35}$$

Dumont [39] presents a thorough assessment of these rigid-body terms using a consistent BEM formulation and shows that they do not influence the numerical results in the frame of a consistent formulation.

Moreover, applying eqs. (5-32) and (5-33) to the tractions in eq. (5-31) one obtains the complex expression

$$T^* = \frac{-1}{8\pi(1-\nu)}\left[2\Re\left(\frac{p^*}{z}\right)\eta + \overline{\left(\frac{3-4\nu}{z}\bar{p}^* + p^*\frac{\bar{z}}{z^2}\right)\eta}\right] \tag{5-36}$$

5.2.2
Complex counterpart of BEM matrices for elasticity

The general expressions of both **H** and **G** in Cartesian coordinates are

$$H_{sf} = \int_\Gamma T_{js}^* u_{jf}(\xi)\mathrm{d}\Gamma(\xi), \text{ and} \tag{5-37}$$

$$G_{s\ell} = \int_\Gamma u_{js}^* T_{j\ell}(\xi)\mathrm{d}\Gamma(\xi) \tag{5-38}$$

where T_{js}^* and u_{js}^* are the Kelvin's fundamental solution expressed for point force parameters p_s^*, with

$$T_{js}^* = \frac{-1}{4\pi(1-\nu)r}\left\{ \left[(1-2\nu)\delta_{js} + 2r,_j r,_s\right]\frac{\partial r}{\partial \eta} + \right.$$
$$\left. (1-2\nu)(r,_j \eta_s - \eta_j r,_s) \right\}, \text{ and} \tag{5-39}$$

$$u_{js}^* = \frac{-1}{8\pi G(1-\nu)}\left((3-4\nu)\ln(r)\delta_{js} - r,_j r,_s + C\delta_{js}\right), \tag{5-40}$$

for $j = 1, 2$ indicating the Cartesian coordinate directions x and y. It is understood that, for notation simplicity, $r \equiv \sqrt{(x_f - x_s)^2 + (y_f - y_s)^2}$ in eq. (5-39) and $r \equiv \sqrt{(x_\ell - x_s)^2 + (y_\ell - y_s)^2}$ in eq. (5-40).

While the matrices **H** and **G** for a potential problem have the simple complex representation of eqs. (5-8) and (5-9), the application to elasticity problems is slightly more involved. To start with, it is advisable to rewrite the displacement and boundary traction expressions in eqs. (5-34) and (5-36) in

terms of functions that multiply p^* and $\bar{p}^*$ explicitly as

$$
\begin{aligned}
u^* &\equiv \frac{-1}{8\pi G\,(1-\nu)}\left\langle (3-4\nu)\,\Re\ln(z) \quad -\tfrac{1}{2}\overline{\left(\tfrac{\bar z}{z}\right)}\right\rangle \begin{Bmatrix} p^* \\ \bar{p}^* \end{Bmatrix} \\
&\equiv \left\langle u_1^* \quad u_2^* \right\rangle \begin{Bmatrix} p^* \\ \bar{p}^* \end{Bmatrix}, \text{ and}
\end{aligned}
\tag{5-41}
$$

$$
T^* \equiv \frac{-1}{8\pi\,(1-\nu)}\left\langle \tfrac{\eta}{z}+\overline{\tfrac{3-4\nu}{z}\eta} \quad \overline{\tfrac{\bar\eta}{z}}+\tfrac{\bar z}{z^2}\eta \right\rangle \begin{Bmatrix} p^* \\ \bar{p}^* \end{Bmatrix} \equiv \left\langle T_1^* \quad T_2^* \right\rangle \begin{Bmatrix} p^* \\ \bar{p}^* \end{Bmatrix}
\tag{5-42}
$$

observing that $\Re\left(\tfrac{p^*}{z}\right)=\tfrac{1}{2}\left(\tfrac{p^*}{z}+\tfrac{\bar{p}^*}{\bar z}\right)$.

A more convenient expression of displacements is

$$
\begin{aligned}
\begin{Bmatrix} u^* \\ \bar{u}^* \end{Bmatrix} &= \frac{-1}{8\pi G\,(1-\nu)}\begin{bmatrix} (3-4\nu)\,\Re\ln(z) & -\tfrac{1}{2}\overline{\left(\tfrac{\bar z}{z}\right)} \\ -\tfrac{1}{2}\left(\tfrac{\bar z}{z}\right) & (3-4\nu)\,\Re\ln(z) \end{bmatrix}\begin{Bmatrix} p^* \\ \bar{p}^* \end{Bmatrix} \\
&\equiv \begin{bmatrix} u_{11}^* & u_{12}^* \\ u_{21}^* & u_{22}^* \end{bmatrix}\begin{Bmatrix} p^* \\ \bar{p}^* \end{Bmatrix} \equiv \begin{bmatrix} u_{11}^* & \bar{u}_{21}^* \\ u_{21}^* & u_{11}^* \end{bmatrix}\begin{Bmatrix} p^* \\ \bar{p}^* \end{Bmatrix}.
\end{aligned}
\tag{5-43}
$$

Observe the introduction of a Hermitian matrix in the above linear transformation. Similarly, the boundary tractions are better represented as

$$
\begin{aligned}
\begin{Bmatrix} T^* \\ \bar{T}^* \end{Bmatrix} &= \frac{-1}{8\pi\,(1-\nu)}\begin{bmatrix} \tfrac{\eta}{z}+\overline{\tfrac{3-4\nu}{z}\eta} & \overline{\tfrac{\bar\eta}{z}}+\tfrac{\bar z}{z^2}\eta \\ \tfrac{\bar\eta}{z}+\tfrac{\bar z}{z^2}\eta & \overline{\left(\tfrac{\eta}{z}\right)}+\tfrac{3-4\nu}{z}\eta \end{bmatrix}\begin{Bmatrix} p^* \\ \bar{p}^* \end{Bmatrix} \\
&\equiv \begin{bmatrix} T_{11}^* & T_{12}^* \\ T_{21}^* & T_{22}^* \end{bmatrix}\begin{Bmatrix} p^* \\ \bar{p}^* \end{Bmatrix} \equiv \begin{bmatrix} T_{11}^* & \bar{T}_{21}^* \\ T_{21}^* & \bar{T}_{11}^* \end{bmatrix}\begin{Bmatrix} p^* \\ \bar{p}^* \end{Bmatrix}.
\end{aligned}
\tag{5-44}
$$

This linear transformation matrix is the sum of a Hermitian and a diagonal matrix. This property is possibly more relevant than the fact that the corresponding transformation matrix for the real-variable formulation, as shown in eq. (5-39), is the sum of a symmetric and a skew-symmetric matrix.

Finally, integration over a general segment in the frame of the CBEM is carried out considering that displacements u_j and boundary tractions T_j are interpolated along a boundary segment according to the complex representation

$$
\begin{Bmatrix} u \\ \bar{u} \end{Bmatrix} = u_f(\xi)\begin{Bmatrix} d \\ \bar{d} \end{Bmatrix}_f, \text{ and } \begin{Bmatrix} T \\ \bar{T} \end{Bmatrix} = t_\ell(\xi)\begin{Bmatrix} t \\ \bar{t} \end{Bmatrix}_\ell,
\tag{5-45}
$$

where $u_f \in \Re$ and $t_\ell = u_\ell |J|_{at\ \ell}/|J|$.

The interpolation function u_ℓ is the same one introduced in eq. (2-17). As a result of the above developments, the double- and the single-layer matrices

for elasticity problems are represented as

$$H_{sf} = \int_{\Gamma} \begin{bmatrix} \overline{T}^*_{11} & \overline{T}^*_{21} \\ T^*_{21} & T^*_{11} \end{bmatrix}_s u_f \mathrm{d}\Gamma, \text{ and} \tag{5-46}$$

$$G_{s\ell} = \int_{\Gamma} \begin{bmatrix} u^*_{11} & \overline{u}^*_{21} \\ u^*_{21} & u^*_{11} \end{bmatrix}_s t_\ell \mathrm{d}\Gamma. \tag{5-47}$$

5.2.3
Formulation for the CBEM

Let the single-layer potential matrix $G_{s\ell}$ presented in eq. (5-47) be multiplied by the vector of traction forces as in eq. (2-12), then

$$G_{s\ell} T_\ell = \int_{\Gamma} \begin{bmatrix} u^*_{11} & \overline{u}^*_{21} \\ u^*_{21} & u^*_{11} \end{bmatrix}_s u_\ell \frac{|J|_\ell}{|J|} \mathrm{d}\Gamma \begin{Bmatrix} T \\ \overline{T} \end{Bmatrix}_\ell, \tag{5-48}$$

using the interpolation of tractions defined in eq. (5-45). As the first line of the matrix-vector multiplication $G_{s\ell} T_\ell$ is the complex conjugate of the second (u^*_{11} is a real number), only one of them needs to be evaluated. The second component of $G_{s\ell} T_\ell$ in eq. (5-48) is

$$\begin{aligned}
\{G_{s\ell} T_\ell\}_2 &= \int_{\Gamma} \left(u^*_{21_s} T_\ell + u^*_{11_s} \overline{T}_\ell \right) u_\ell \frac{|J|_{at\,\ell}}{|J|} \mathrm{d}\Gamma \\
&= \int_{\Gamma} u^*_{21_s} T_\ell u_\ell \frac{|J|_{at\,\ell}}{|J|} \mathrm{d}\Gamma + \int_{\Gamma} u^*_{11_s} \overline{T}_\ell u_\ell \frac{|J|_{at\,\ell}}{|J|} \mathrm{d}\Gamma.
\end{aligned} \tag{5-49}$$

Now let the definition of u^* in eq. (5-43) be applied to eq. (5-49), then

$$\begin{aligned}
\{G_{s\ell} T_\ell\}_2 &= \frac{1}{16\pi G (1-\nu)} \int_{\Gamma} \frac{\overline{z - z_s}}{z - z_s} T_\ell u_\ell \frac{|J|_{at\,\ell}}{|J|} \mathrm{d}\Gamma - \\
&\quad \frac{(3-4\nu)}{8\pi G (1-\nu)} \int_{\Gamma} \Re \ln (z - z_s) \overline{T}_\ell u_\ell \frac{|J|_{at\,\ell}}{|J|} \mathrm{d}\Gamma.
\end{aligned} \tag{5-50}$$

The second term of the previous expression is the same one of the potential problem, defined in eq. (5-16), except for a multiplication constant. Thus, no additional attention is required for this term, as it has already been dealt with. On the other hand, dealing with the first term of eq. (5-50) may be challenging, as the series expansion of the function $\frac{\overline{z - z_s}}{z - z_s}$ is not straightforward. Let a function $g(z)$ be defined as

$$g(z) = \overline{z - z_s} f(z) \equiv (\bar{z} - \bar{z}_s) f(z), \text{ with } f(z) = \frac{1}{z - z_s}. \tag{5-51}$$

Introducing the latter definition in the first term of the right-hand side of eq. (5-50), one obtains

$$\int_{\Gamma} \frac{\overline{z - z_s}}{z - z_s} T_\ell u_\ell \frac{|J|_{at\,\ell}}{|J|} \mathrm{d}\Gamma = \int_{\Gamma} \overline{z - z_s} f(z) T_\ell u_\ell \frac{|J|_{at\,\ell}}{|J|} \mathrm{d}\Gamma, \tag{5-52}$$

which may be further expanded by making use of eq. (4-2) as

$$
\int_\Gamma \frac{\overline{z - z_s}}{z - z_s} T_\ell u_\ell \frac{|J|_{at\,\ell}}{|J|} \mathrm{d}\Gamma =
$$
$$
\left[\int_\Gamma \overline{z} \sum_{i=2}^{n+2} fac_{i-1} P_{i-1}\left(z - z_c\right) Q_i\left(z_c - z_s\right) T_\ell u_\ell \frac{|J|_{at\,\ell}}{|J|} \mathrm{d}\Gamma \right.
$$
$$
\left. - \overline{z_s} \int_\Gamma \sum_{i=2}^{n+2} fac_{i-1} P_{i-1}\left(z - z_c\right) Q_i\left(z_c - z_s\right) T_\ell u_\ell \frac{|J|_{at\,\ell}}{|J|} \mathrm{d}\Gamma \right]. \tag{5-53}
$$

Taking the constant parts out of the integrals, one finally obtains

$$
\int_\Gamma \frac{\overline{z - z_s}}{z - z_s} T_\ell u_\ell \frac{|J|_{at\,\ell}}{|J|} \mathrm{d}\Gamma =
$$
$$
T_\ell |J|_{at\,\ell} \left[\sum_{i=2}^{n+2} fac_{i-1} Q_i\left(z_c - z_s\right) \int_\Gamma \overline{z} P_{i-1}\left(z - z_c\right) u_\ell \frac{1}{|J|} \mathrm{d}\Gamma \right.
$$
$$
\left. - \overline{z_s} \sum_{i=2}^{n+2} fac_{i-1} Q_i\left(z_c - z_s\right) \int_\Gamma P_{i-1}\left(z - z_c\right) u_\ell \frac{1}{|J|} \mathrm{d}\Gamma \right]. \tag{5-54}
$$

Then, introducing the vectors

$$
\widetilde{Q}_i = \overline{z_s} Q_i, \text{ and } \widetilde{P}_i = \overline{z} P_i, \tag{5-55}
$$

eq. (5-54) may be written compactly as

$$
\int_\Gamma \frac{\overline{z - z_s}}{z - z_s} T_\ell u_\ell \frac{|J|_{at\,\ell}}{|J|} \mathrm{d}\Gamma =
$$
$$
T_\ell |J|_{at\,\ell} \left[\sum_{i=2}^{n+2} fac_{i-1} Q_i\left(z_c - z_s\right) \int_\Gamma \widetilde{P}_{i-1}\left(z - z_c\right) u_\ell \frac{1}{|J|} \mathrm{d}\Gamma \right.
$$
$$
\left. - \sum_{i=2}^{n+2} fac_{i-1} \widetilde{Q}_i\left(z_c - z_s\right) \int_\Gamma P_{i-1}\left(z - z_c\right) u_\ell \frac{1}{|J|} \mathrm{d}\Gamma \right]. \tag{5-56}
$$

Discretisation of the boundary Γ in the above expression leads to

$$
\int_\Gamma \frac{\overline{z - z_s}}{z - z_s} T_\ell u_\ell \frac{|J|_{at\,\ell}}{|J|} \mathrm{d}\Gamma =
$$
$$
T_\ell |J|_{at\,\ell} \left[\sum_{i=2}^{n+2} fac_{i-1} Q_i\left(z_c - z_s\right) \int_0^1 \widetilde{P}_{i-1}\left(z(\xi) - z_c\right) N(\xi)_\ell \mathrm{d}\xi \right.
$$
$$
\left. - \sum_{i=2}^{n+2} fac_{i-1} \widetilde{Q}_i\left(z_c - z_s\right) \int_0^1 P_{i-1}\left(z(\xi) - z_c\right) N(\xi)_\ell \mathrm{d}\xi \right], \tag{5-57}
$$

where $N(\xi)$ are shape functions.

Both integrals in eq. (5-57) may be evaluated analytically, as they are as simple as the multiplication of polynomials. One of these terms, which uses vector P, has been analytically evaluated and organized in tables by Peixoto [20] for potential problems. The second term is a polynomial of only one order higher and may be evaluated in the same way.

Now, let the double-layer potential matrix H_{sf}, as defined in eq. (5-46),

be multiplied by the vector of displacements d_f as in eq. (2-12) – in a similar assessment as previously carried out for the single-layer potential matrix. Then,

$$H_{sf}d_f = \int_\Gamma \begin{bmatrix} \overline{T}^*_{11} & \overline{T}^*_{21} \\ T^*_{21} & T^*_{11} \end{bmatrix}_s u_f \mathrm{d}\Gamma \begin{Bmatrix} d \\ \overline{d} \end{Bmatrix}_f. \tag{5-58}$$

Once again, as in eq. (5-48), the first line of eq. (5-58) is the complex conjugate of the second. The evaluation of the second line leads to

$$\{H_{sf}d_f\}_2 = \int_\Gamma \left(T^*_{21_s} d_f + T^*_{11_s} \overline{d}_f \right) u_f \mathrm{d}\Gamma$$
$$= \int_\Gamma T^*_{21_s} u_f d_f \mathrm{d}\Gamma + \int_\Gamma T^*_{11_s} u_f \overline{d}_f \mathrm{d}\Gamma. \tag{5-59}$$

Expanding this expression using the definition of T^* in eq. (5-44), one obtains

$$\{H_{sf}d_f\}_2 = \int_\Gamma \left[\frac{\overline{\eta(z)}}{z - z_s} + \frac{\overline{z - z_s}}{(z - z_s)^2} \eta(z) \right] u_f d_f \mathrm{d}\Gamma +$$
$$\int_\Gamma \left[\frac{\eta(z)}{z - z_s} + \frac{\overline{3 - 4\nu}}{z - z_s} \eta(z) \right] u_f \overline{d}_f \mathrm{d}\Gamma. \tag{5-60}$$

The first term in each integral of eq. (5-60) is the same one of eq. (5-10) for the matrix H^C_{sf} of potential problems – except for the complex conjugate of the normal vector η. Then, its evaluation is carried out according to eq. (5-18) and only paying attention to the conjugate term.

On the other hand, the terms multiplied by $\frac{\overline{z - z_s}}{(z - z_s)^2}$ and $\frac{1}{z - z_s}$ must be more carefully assessed. Let the former term be written as

$$\int_\Gamma \frac{\overline{z - z_s}}{(z - z_s)^2} \eta(z) u_f d_f \mathrm{d}\Gamma = \int_\Gamma \frac{\overline{z}}{(z - z_s)^2} \eta(z) u_f d_f \mathrm{d}\Gamma -$$
$$\int_\Gamma \frac{\overline{z}_s}{(z - z_s)^2} \eta(z) u_f d_f \mathrm{d}\Gamma, \tag{5-61}$$

which presents similar terms as in eq. (5-53) except for the higher order power of the binomial $z - z_s$. By observing the definition of the vector Q in eq. (4-4) for the fundamental solution, one identifies that eq. (4-2) may be readily applied to the first term of eq. (5-61) as

$$\int_\Gamma \frac{\overline{z}}{(z - z_s)^2} \eta(z) u_f d_f \mathrm{d}\Gamma =$$
$$\sum_{i=3}^{n+3} fac_{i-2} \widetilde{P}_{i-2}(z - z_c) Q_i(z_c - z_s)\eta(z) u_f d_f \mathrm{d}\Gamma, \tag{5-62}$$

also using eq. (5-55). Moving the constants out of the integral leads to

$$\int_\Gamma \frac{\overline{z}}{(z - z_s)^2} \eta(z) u_f d_f \mathrm{d}\Gamma =$$
$$\sum_{i=3}^{n+3} fac_{i-2} Q_i(z_c - z_s) \int_\Gamma \widetilde{P}_{i-2}(z - z_c)\eta(z) u_f d_f \mathrm{d}\Gamma. \tag{5-63}$$

By a similar manipulation, the second term in eq. (5-61) results in

$$
\int_\Gamma \frac{\overline{z_s}}{(z - z_s)^2} \eta(z) u_f d_f \mathrm{d}\Gamma =
$$
$$
\sum_{i=3}^{n+3} fac_{i-2} \widetilde{Q}_i (z_c - z_s) \int_\Gamma P_{i-2} (z - z_c) \eta(z) u_f d_f \mathrm{d}\Gamma. \tag{5-64}
$$

Finally, the term in eq. (5-60) that is multiplied by $\frac{1}{\overline{z - z_s}}$ may be written as

$$
\int_\Gamma \frac{\overline{3 - 4\nu}}{z - z_s} \eta(z) u_f \overline{d}_f \mathrm{d}\Gamma = \int_\Gamma \frac{3 - 4\nu}{\overline{z - z_s}} \overline{\eta}(z) u_f \overline{d}_f \mathrm{d}\Gamma. \tag{5-65}
$$

Then, application of the GFMM series expansion of eq. (4-2) leads to

$$
\int_\Gamma \frac{3 - 4\nu}{\overline{z - z_s}} \overline{\eta}(z) u_f \overline{d}_f \mathrm{d}\Gamma =
$$
$$
\int_\Gamma (3 - 4\nu) \sum_{i=2}^{n+2} \overline{fac_{i-1} P_{i-1} (z - z_c) Q_i (z_c - z_s)} \overline{\eta}(z) u_f \overline{d}_f \mathrm{d}\Gamma = \tag{5-66}
$$
$$
(3 - 4\nu) \sum_{i=2}^{n+2} fac_{i-1} \overline{Q}_i (z_c - z_s) \int_\Gamma \overline{P}_{i-1} (z - z_c) \overline{\eta}(z) u_f \overline{d}_f \mathrm{d}\Gamma.
$$

5.2.4
Formulation for the SHBEM

As it has already been remarked in Section 5.1.4, the SHBEM uses the conjugate transpose of the same matrix **H** of the CBEM, as indicated in the matrix product in the second line of eq. (2-27). Moreover, the SHBEM makes use of the Hermitian matrix of fundamental solutions of eq. (5-43),

$$
\mathbf{U}^* = U_{sf}^* = \begin{bmatrix} u_{11}^* & \overline{u}_{21}^* \\ u_{21}^* & u_{11}^* \end{bmatrix}_{z=z_f-z_s}. \tag{5-67}
$$

In order to evaluate the product of the conjugate transpose matrix of H_{sf} in a GFMM context, it is necessary to use the full expansion presented in eq. (4-7). The product of the conjugate transpose matrix H_{sf} by the vector of equivalent nodal forces p_f is, from eq. (5-58),

$$
H_{fs} p_f = \overline{H}_{sf} p_s = \int_\Gamma \begin{bmatrix} T_{11}^* & \overline{T}_{21}^* \\ T_{21}^* & \overline{T}_{11}^* \end{bmatrix}_s u_f \mathrm{d}\Gamma \begin{Bmatrix} p \\ \overline{p} \end{Bmatrix}_s. \tag{5-68}
$$

As done in the last session, one can write for the second line of equations above

$$
\left\{ \overline{H}_{sf} p_s \right\}_2 = \int_\Gamma \left(T_{21_s}^* p_s + \overline{T}_{11_s}^* \overline{p}_s \right) u_f \mathrm{d}\Gamma
$$
$$
= \int_\Gamma T_{21_s}^* u_f p_s \mathrm{d}\Gamma + \int_\Gamma \overline{T}_{11_s}^* u_f \overline{p}_s \mathrm{d}\Gamma, \tag{5-69}
$$

which can be expanded by applying the definition of T^* in eq. (5-44) as

$$\left\{\overline{H}_{sf}p_s\right\}_2 = \int_\Gamma \left[\frac{\overline{\eta(z)}}{z-z_s} + \frac{\overline{z-z_s}}{(z-z_s)^2}\eta(z)\right]u_f p_s \mathrm{d}\Gamma +$$
$$\int_\Gamma \left[\frac{\overline{\eta(z)}}{z-z_s} + \frac{3-4\nu}{z-z_s}\eta(z)\right]u_f \overline{p}_s \mathrm{d}\Gamma. \tag{5-70}$$

As a matter of illustration of the application of the full GFMM expansion of eq. (4-7), the first integral in the above expression can be written, with $f(z-z_0) = \frac{1}{z-z_s}$, as

$$\int_\Gamma \frac{\overline{\eta(z)}}{z-z_s}u_f p_s \mathrm{d}\Gamma =$$
$$\int_\Gamma \sum_{i=1}^{n+1} fac_i P_i(z-z_{c^{nc}}) \sum_{j=1}^{m+1} fac_j P_j(z_{L^{nL}}-z_s)Q_{i+j}(z_{c^{nc}}-z_{L^{nL}})\overline{\eta(z)}u_s p_f \mathrm{d}\Gamma =$$
$$\sum_{i=1}^{n+1} fac_i \int_\Gamma P_i(z-z_{L^{nL}})\overline{\eta(z)}u_s p_f \mathrm{d}\Gamma \sum_{j=1}^{m+1} fac_j P_j(z_{c^{nc}}-z_f)Q_{i+j}(z_{L^{nL}}-z_{c^{nc}}), \tag{5-71}$$

which holds for the other terms multiplied by $\frac{1}{z-z_s}$ in eq. (5-70). These integrals are the same ones solved analytically by Peixoto [20].

The remaining integral in eq. (5-70) may be written as a sum of integrals – aiming to take advantage of the definitions of eq. (5-55) – as

$$\int_\Gamma \frac{\overline{z-z_s}}{(z-z_s)^2}\eta(z)u_f p_s \mathrm{d}\Gamma =$$
$$\int_\Gamma \frac{\overline{z}}{(z-z_s)^2}\eta(z)u_f p_s \mathrm{d}\Gamma - \int_\Gamma \frac{\overline{z_s}}{(z-z_s)^2}\eta(z)u_f p_s \mathrm{d}\Gamma. \tag{5-72}$$

Applying the GFMM expansion of eq. (4-7) to the first term at the right-hand side of eq. (5-72), one obtains

$$\int_\Gamma \frac{\overline{z}}{(z-z_s)^2}\eta(z)u_f p_s \mathrm{d}\Gamma =$$
$$\int_\Gamma \sum_{i=1}^{n+1} fac_i \widetilde{P}_i(z-z_{c^{nc}}) \sum_{j=1}^{m+1} fac_j P_j(z_{L^{nL}}-z_s)Q_{i+j+1}(z_{c^{nc}}-z_{L^{nL}})\eta(z)u_f p_s \mathrm{d}\Gamma =$$
$$\sum_{i=1}^{n+1} fac_i \int_\Gamma \widetilde{P}_i(z-z_{c^{nc}})u_f p_s \mathrm{d}\Gamma \sum_{j=1}^{m+1} fac_j P_j(z_{L^{nL}}-z_s)Q_{i+j+1}(z_{c^{nc}}-z_{L^{nL}})\eta(z), \tag{5-73}$$

which may be discretised and integrated with the same strategies of the ones presented in the previous Section.

The second term at the right-hand side of eq. (5-72) becomes, by making use of eq. (4-7),

$$\int_\Gamma \frac{\overline{z_s}}{(z-z_s)^2}\eta(z)u_f p_s \mathrm{d}\Gamma =$$

$$\sum_{i=1}^{n+1} fac_i \int_\Gamma P_i(z-z_{c^{n_c}})u_f p_s \mathrm{d}\Gamma \sum_{j=1}^{m+1} fac_j P_j(z_{L^{n_L}}-z_s)\widetilde{Q}_{i+j+1}(z_{c^{n_c}}-z_{L^{n_L}})\eta(z).$$

$$(5\text{-}74)$$

The application of the GFMM expansions to the matrix U_{sf} of fundamental solutions is as straightforward as possible, as it requires no integrations at all. Moreover, the expressions of the fundamental solution u^*, defined in eq. (5-43), are almost the same ones of potential problems, except for the multiplication by $\overline{z}$ in the off-diagonal terms, which can be dealt with according to the definitions given in eq. (5-55).

5.2.5
Results at internal points

The evaluation of results at internal points for elasticity follows the same strategy as for potential problems in Section 5.1.5. The Somigliana identity reads, as presented by Dumont [17],

$$u_m = \int_\Gamma u_{im}^* t_i \mathrm{d}\Gamma - \int_\Gamma \sigma_{jim}^* \eta_j u_i \mathrm{d}\Gamma, \qquad (5\text{-}75)$$

with m internal points, disregarding rigid-body displacements and body forces.

As for potential problems, eq. (5-75) presents two terms that are very similar to matrices H_{sf} and $G_{s\ell}$, for displacements at an internal point and degree of freedom m and considering that the boundary data are known in terms of nodal tractions t_i and displacements u_i.

6
Computational aspects

Herein the computational aspects of an algorithm to the formulation presented in Section 5 are discussed. For the sake of simplicity and generality, the algorithms are presented as flowcharts and code independent. They are valid for both potential and elasticity problems, as well as for the CBEM and the SHBEM. Thus, they may be implemented in any suitable language. The numerical assessments of Section 7 are based on an implementation in C++. The number of code lines is actually very small, as the algorithm resorts to recursion in two different instances.

There are four major procedures: Main, Adjacencies, Source and PoleExpansion [28, 40]. As the recursive routine PoleExpansion is used inside the also recursive routine Adjacencies, the written explanation of such a convoluted algorithm becomes difficult. Aiming to overcome this issue, flowcharts are used to present the basics of each procedure.

The procedure Main shown in fig. 6.1 loads the input data, generates the hierarchical mesh according to the concepts discussed in Section 4.2 and defines the vector Q from eq. (4-4), which is the only kernel-dependent part of the whole algorithm. Then, it executes a small loop over all elements of the first refinement level ($k = 0$) – herein referred to as macro-elements i_e – in order to create the first adjacency structure (in routine Adjacencies0), carrying out at the same time all possible field evaluations for the child elements of element i_e.

The routine Adjacencies, presented in fig. 6.2 and called in the Main procedure after the initialising routine Adjacencies0, assembles the adjacency structure for all children elements of i_e in a recursive loop until the most refined level is reached.

The routine Source is called in the routine Adjacencies when the most refined level ($k = n_v$) is reached. This routine, shown in fig. 6.3, handles integrations of both the near (routine BEMAdj) and far field elements and child elements of i_e. The procedure BEMAdj is a BEM algorithm for matrix-vector products handled by conventional numerical integration (see Dumont [36]), without Fast Multipole considerations, but taking into account the interpolation of tractions of eq. (2-17). Singularity and quasi-singularity issues must
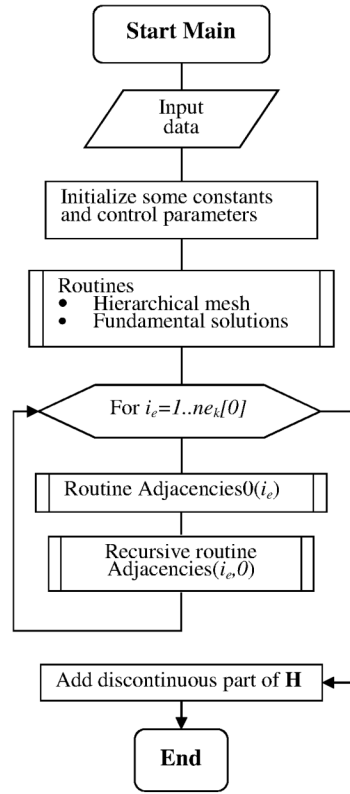
Figure 6.1: Flowchart for the routine Main.

be handled carefully by BEMAdj, which is never the case in the frame of the GFMM *per se*. The FMM integrations for non-adjacent elements – already analytically pre-evaluated – are assembled in the Routine Source, which deals – as indicated in eq. (5-16), for instance – with the corresponding information passed to the n_c child elements of element ie_p.
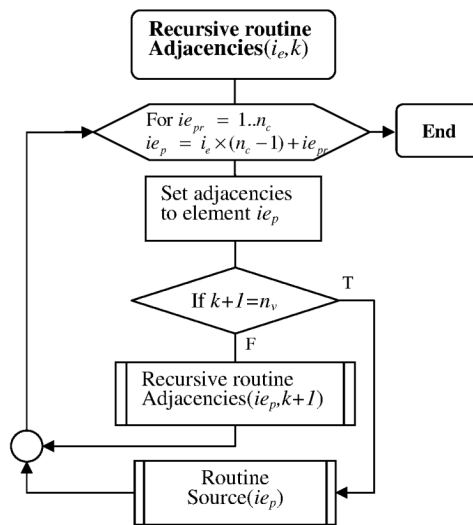

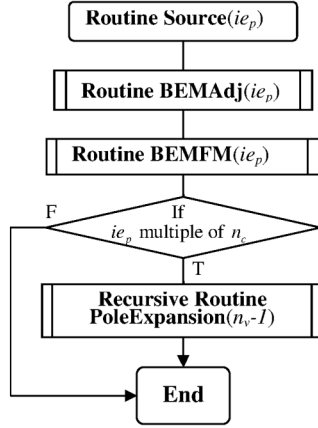Figure 6.2: Flowchart for the routine Adjacencies.

Figure 6.3: Flowchart for the routine Source.

At last, the expansion data are passed to the non-adjacent points by means of the recursive routine PoleExpansion. This routine (fig. 6.4), passes the Fast Multipole integrated P vectors to either distant source or internal points by calling the routine Qvector, which evaluates the array of kernel expansions given in eq. (4-4) after evaluating one of the expansions series in eqs. (4-2) or (4-7). It also checks if the level $k = k_{exp}$ has been reached, as successive multipole expansions (eq. (4-8)) stop at this level, indicating that the P data vectors clustered so far are directly passed to the remaining source or internal points. For $k < k_{exp}$, the routine expands the P vectors – clustered up to this level – to one level higher $(k+1)$ by applying eq. (4-8), which is represented in fig. 6.4 by routine Pvector. When all elements of the current level k have been processed, PoleExpansion calls itself recursively to proceed to the immediately higher refinement level of the hierarchical structure.

The definition of k_{exp} is actually user-dependent, as it, in principle, does not influence the accuracy of the method. In the assessments of Chapter 7, it is always taken as $k_{exp} = 2$, which allows expansions up to a somewhat coarse mesh refinement level.

6.1
Solution of the equations system

All applications of the GFMM up to this point consist in carrying out a single matrix-vector multiplication. In order to solve the linear equation system of the BEM while using the GFMM, one must use iterative solvers that rely on this multiplication. The Generalized Minimum Residual Method (GMRES) [32] is the most common iterative solver used in the BEM literature for FMM implementations [3, 4, 41].

The major feature of the GMRES is that it requires only one matrix-
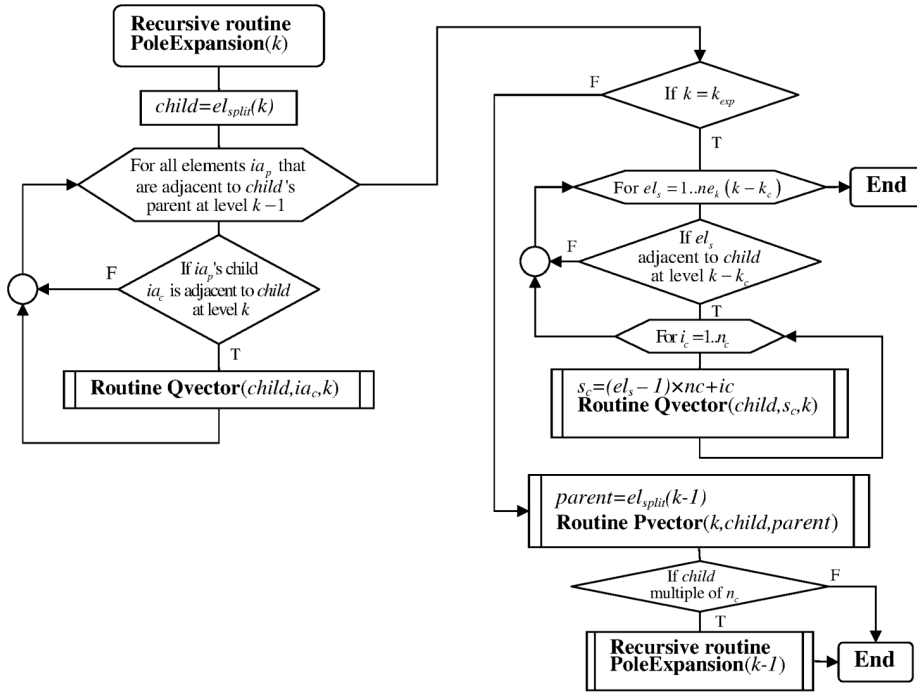
Figure 6.4: Flowchart for the routine Source.

vector multiplication per iteration. This is very interesting for the FMM, as a problem's number of degrees of freedom may be very large. Thus, the matrix-vector products are, usually, the bottleneck of the solution by iterative solvers. Furthermore, the GMRES is also capable of handling nonsymmetric systems of equations, such as the ones from the BEM.

A fast, optimised version of GMRES, as the one used by Liu [3], is available online in the open-source library SLATEC (<[http://www.netlib.org/slatec/>) in FORTRAN language. In this thesis, a C++ version of the optimised SLATEC GMRES package is used (<http://www.netlib.org/templates/cpp/>). All the results presented herein which use the GMRES are obtained without the need of preconditioners. Should these be needed in order to grant convergence, Liu [3] proposes the usage Jacobi block-diagonal preconditioners [42].

7
Some numerical assessments and examples

This section attempts to numerically validate the developments formulated in the previous chapters by means of some error assessments of results that can also be evaluated analytically. Moreover, the computational time of the GFMM algorithms is given for different calculations and particularly for comparison with the time required when running the corresponding conventional implementation. All codes have been implemented by the author in C++ and were compiled with VC++ 2015 with O2 optimisation flag, based on the topics discussed in Section 6 as an extension of the codes initially proposed by Peixoto [20] and Novelino [18].

All numerical tests were executed in a desktop computer with an i7™-4770 CPU 3.4 GHz, 16 GB RAM on Windows® 7. None of the following examples used parallel execution, despite the fact that the processor provided 8 threads for concurrent execution in a shared memory environment. Both algorithms for the conventional methods and their GFMM counterparts follow the same logic and data structure, aiming to compare results as consistently as possible, without taking into account some eventual code optimisation that might contribute to arrive at smaller computational times.

7.1
Cut-out tests

Let some analytical solution of the differential equation that governs the problem to be assessed be defined in the open domain. One may draw an arbitrary boundary and evaluate displacements $\mathbf{d}$ and traction forces $\mathbf{t}$, for an elasticity problem, or potentials $\boldsymbol{\theta}$ and normal fluxes $\mathbf{q}$, for a potential problem, along points of a discretised mesh and evaluate eq. (2-12) in terms of the Euclidean error norm

$$\epsilon = \frac{|\mathbf{Hd} - \mathbf{Gt}|}{|\mathbf{Gt}|}, \tag{7-1}$$

which configures a cut-out test. These are useful in order to assess the numerical errors introduced by the methods used, when compared to analytical solutions.

7.1.1
Evaluations for potential problems

Some results for potential problems have been already discussed by Novelino [18] for convex problems with both constant and curved elements. Novelino [18] has shown that the GFMM indeed leads to a running time complexity below $\mathcal{O}(N \log N)$ – where N is the number of degrees of freedom – for constant elements with a small loss of precision even for a number of expansion terms as low as $n = 5$. Novelino [18] also presents an accuracy assessment of the GFMM for curved quadratic elements, which demonstrates its capability of arriving at good results when compared against the conventional method.

Peixoto and Dumont [33] present results for the multiply-connected domain with a boundary full of reentrancies defined by the nodes numbered from 1 to 20 shown in fig. 7.1, which are studied for linear and quadratic elements, refined according to the developments discussed in Section 4.2. Despite the fact that the domain has a very irregular boundary, excellent results could be obtained with the application of the GFMM in the frame of the CBEM. The study is carried out for a series of mesh discretizations in terms of quadratic elements with up to $5 \times 2^{18} = 1,310,720$ degrees of freedom submitted to a logarithmic field $\theta = \ln|z - z_{s_1}| + \ln|z - z_{s_2}|$, where $z = x + iy$ is a domain point and $z_{s_1} = 12.5 + 14i$ and $z_{s_2} = 20 + 10i$ are source points, represented by $(*)$ in fig. 7.1.

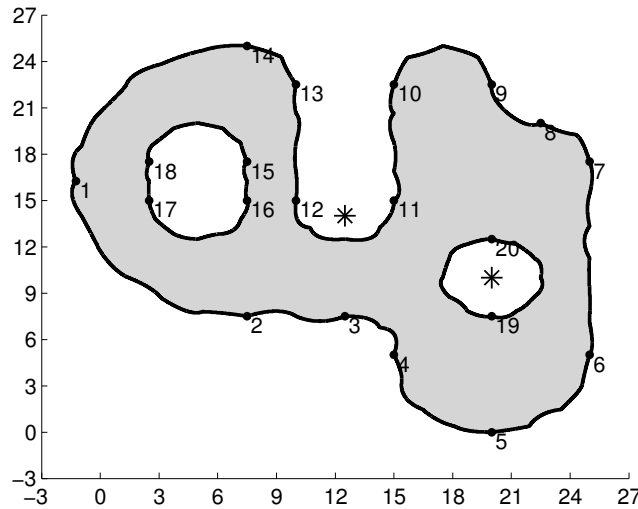
Figure 7.1: Curved domain discretized with quadratic elements and submitted to a logarithmic potential field due to two point sources (marked as *) [33].

The execution times and errors are given in fig. 7.2 for several levels of mesh refinement. The solid black circles refer to the error in eq. (7-1) for the conventional CBEM without GFMM improvements. It is shown on the left of fig. 7.2 that the CBEM algorithm demands running time proportional

to N^2, the expected complexity. The blue squares mark the execution time and error delivered by a CBEM execution for $N = 5,120$. Moreover, several simulations using the GFMM are shown for an increasing number of series terms in eq. (4-2). As expected, results for $n = 1$ are the fastest in terms of execution time and also lead to the highest errors. Results for $n = 17$ and $N = 5,120$ are marked with red circles for the sake of comparison with the CBEM results for the same N. For the same level of accuracy, around 10^{-7}, the GFMM requires approximately 0.1% of the computational time for the conventional implementation. As a matter of fact, a dashed line proportional to $N \log N$ and a dotted line proportional to N are also drawn on the left figure to show that the computational time required by the GFMM algorithm grows rather proportional to N, which is the theoretical best time performance expected [1, 3].
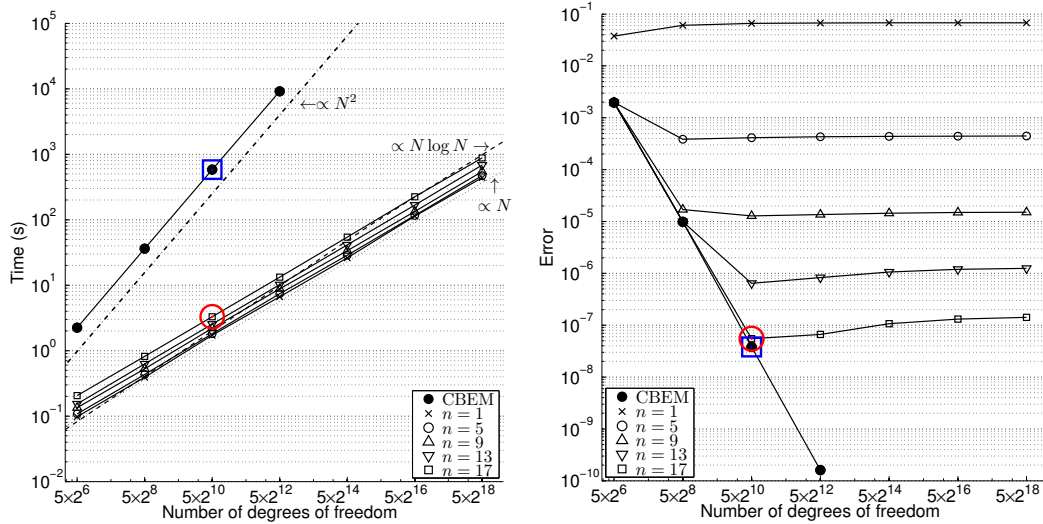
Figure 7.2: Execution times and error measures for the irregular domain of fig. 7.1 discretized with quadratic elements [33].

Another contribution to the application of the GFMM to potential problems is presented by Dumont and Peixoto [19] for the irregularly-shaped domain in fig. 7.3. It is submitted to the quadratic field $\theta = x^2 + y^2$, which is an analytical solution for the Laplace equation. Its boundary is discretized with constant, linear and quadratic elements with up to $2^{24} = 16,777,216$ degrees of freedom as represented in the horizontal axis of fig. 7.4. It may be noticed that, for a given element type, increasing the number of terms in the series does not lead to a considerable increase in the execution time. On the other hand, the graph on the right of fig. 7.4 shows that the number of expansion terms n considerably affects the numerical accuracy. Since the applied analytical field is quadratic, the conventional CBEM solution for quadratic elements (dashed

lines) is as accurate as the numerical integration and round-off errors make possible. However, when evaluated using the GFMM, there is an intrinsic error due to the series expansions. This error turns out to be an accuracy threshold of the simulations with lower-order elements, as shown in fig. 7.4.
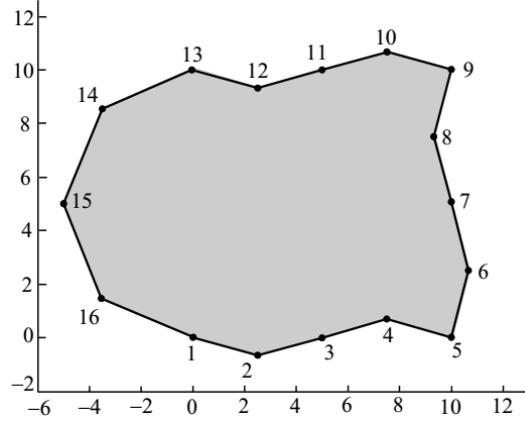

Figure 7.3: Domain shaped by a series of straight segments and submitted to a quadratic potential field used in numerical assessments for potential problems [19].
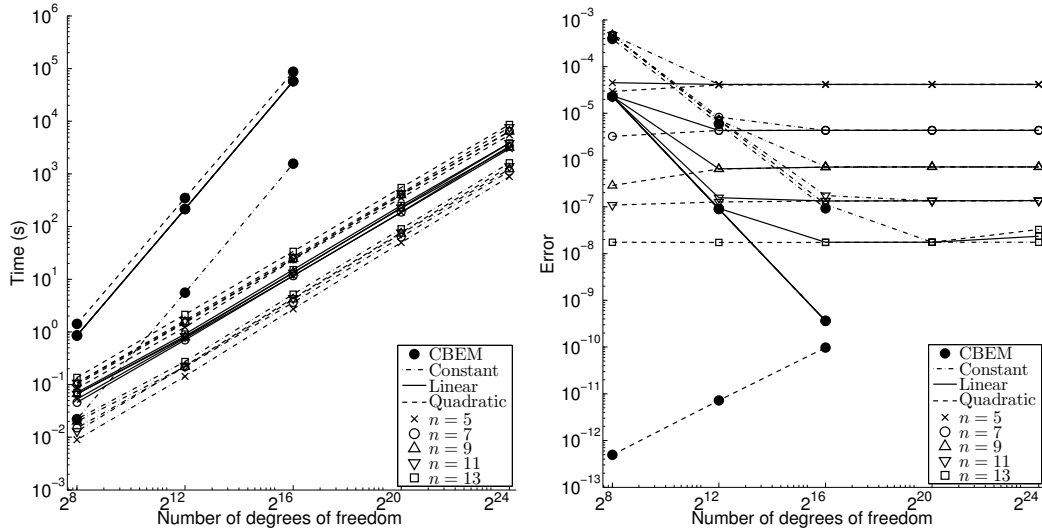

Figure 7.4: Execution times (left) for the domain of fig. 7.3 using constant, linear and quadratic elements, and errors (right) for different numbers n of expansion terms [19].

7.1.2
Elasticity problems

The assessments for elasticity problems follow the same framework as the ones for potential problems. Figure 7.5 shows a multiply connected domain that will undergo a hierarchical mesh refinement and then will be submitted

to a testing field of displacements $u = \frac{1}{-3x^2-3y^2}$, which satisfies equilibrium homogeneously. The displacements **u** and traction forces **t** are obtained along the nodal points of the boundary drawn in the open field, as shown. This cutout test measures the accuracy of the BEM equations for elasticity by using the same Euclidean error norm in eq. (7-1).
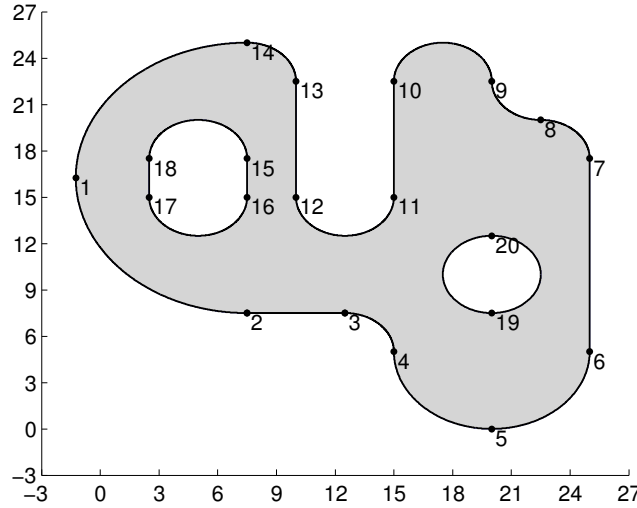

Figure 7.5: Irregularly-shaped, multiply-connected domain.

The domain of fig. 7.5 is discretised with quadratic elements with up to $5 \times 2^{18} = 1,310,720$ degrees of freedom.

One sees on the left of fig. 7.6 that the computational effort increases only slightly as the number n of expansion terms increases. This graph also displays the curves proportional to $\mathcal{O}(N)$ (dotted line), $\mathcal{O}(N \log N)$ (dashed line), and $\mathcal{O}(N^2)$ (dash-dot line). Furthermore, while the implementation of the matrix-vector product in terms of the conventional BEM requires a computational time proportional to $\mathcal{O}(N^2)$, the present fast multipole implementation performs close to $\mathcal{O}(N)$, as already suggested by Liu [3] as an achievable goal.

7.2
Evaluation of results at internal points

The domain in fig. 7.7 is used to demonstrate the capability of the GFMM to rapidly evaluate results at internal points. This multiply-connected domain has 3829 internal points that will be used to obtain, in this case, potentials based on the Somigliana's identity. The nodal values of potentials and normal fluxes are evaluated from the analytical potential field $\theta = .5 \ln(x^2 + 2x + 1 + y^2)$, which is a solution of the Laplace equation.

Figure 7.8 shows on the left the computational time needed to obtain potential results at all internal points of the domain in fig. 7.7. It also displays, on the right, the global error norms for the evaluated potential results as
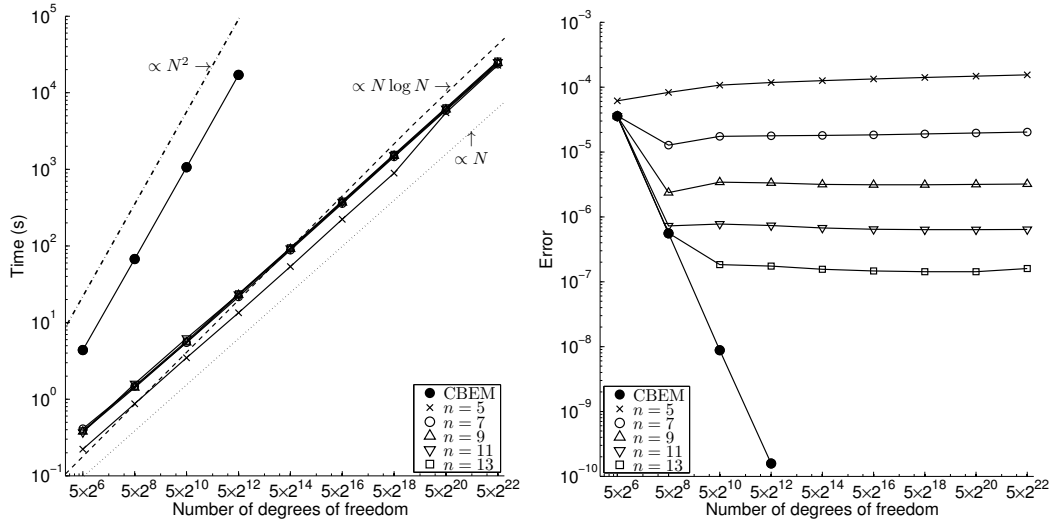
Figure 7.6: Execution times (left) for an elasticity field for the domain on fig. 7.5 with quadratic elements, and accuracy results (right) for different numbers of expansion terms n.
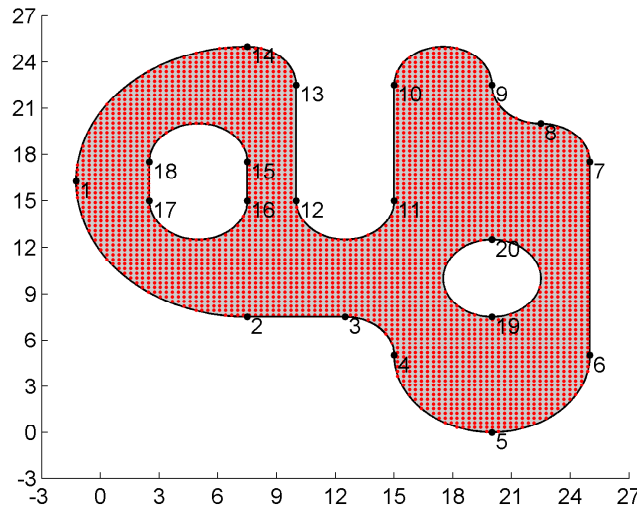

Figure 7.7: Irregular domain with two holes and 3829 internal points marked as dots.

compared with the analytical ones. As expected, the GFMM execution times are faster then the ones using the conventional formulation of the BEM. Moreover, the curves for different numbers of expansion terms still tend to be parallel, but more apart from each other than in the evaluation of the BEM matrix products, which should be expected, as the evaluation of results at internal points does not involve the same code complexity as for the BEM matrices. In fact, the number of internal points is kept constant in all calculations even though the number of boundary segments and nodes increases, which changes the time complexity algorithm for the conventional evaluations from $\mathcal{O}(N^2)$ to $\mathcal{O}(N*M)$, with M being the number of internal

points. For a constant M, it is then expected that the execution time of different discretizations be proportional to N. Then, the time performance with the FMM does not differ as drastically as when matrix products are evaluated.

The error results on the right of fig. 7.8 are not of the same accuracy order than the ones from the previous examples. Observe that the increase of the number of terms n in the series expansion does not increase the precision of the GFMM. This is due to the fact that internal points which are too close to nodes of the parametric representation of the boundary present quasi-singularities, which are not treated in this work, but should be addressed in a future research work, based on the formulation proposed by Dumont [36].
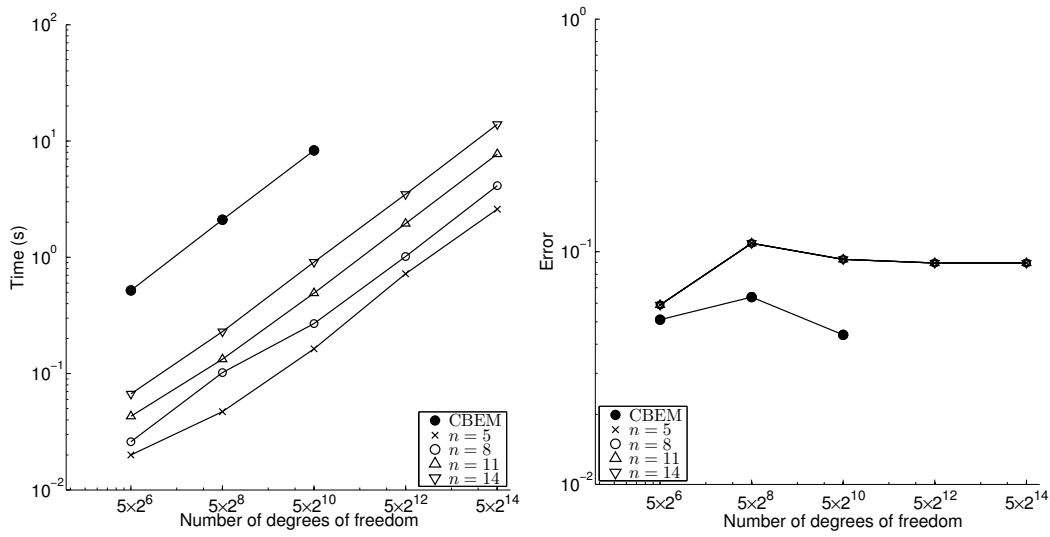
Figure 7.8: Execution times (left) for the evaluation of results at internal points for the domain on fig. 7.7 with quadratic elements, and accuracy results (right) for different numbers of expansion terms n.

7.3
Complete solution

All the assessments above have been done in order to show the GFMM's capability of representing the BEM matrices accurately. A more comprehensive study than the cut-out tests is to couple the GFMM developments with the GMRES iterative solver, aiming to analyse a problem with consistent boundary conditions.

The domain in fig. 7.9 is an annular region submitted to a constant potential $\theta_{inner} = 100$ in the inner boundary, and constant normal flux $q_{outer} = 200$ on the outer boundary. This example is proposed and studied by Liu [3]. For this arrangement, the analytical solution of the problem is

$$\theta(r) = \theta_{inner} + q_{outer}r_{inner} \log \frac{r}{r_{outer}}, \tag{7-2}$$

where $r_{inner} = 1$ is the internal and $r_{outer} = 2$ is the outer radius of the annular domain. Thus, the expected solution for this problem is $\theta = 377.2589$ and $q = -400.0$.
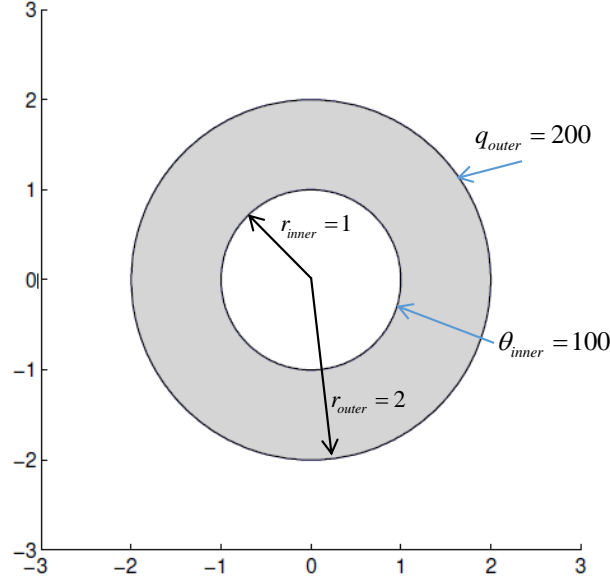

Figure 7.9: Annular domain submitted to a prescribed potential in its inner rim, and a prescribed flux on its outer rim.

As the curved geometry already suggests, a discretisation with quadratic elements is the more suited for this analysis. The results shown in fig. 7.10 present the same behaviour as the ones for cut-out tests already presented in the previous sections, since the GFMM execution times perform close to $\mathcal{O}(N)$, while with the BEM they are proportional to $\mathcal{O}(N^2)$.

Fifteen simulation results are shown, and except for the simulations with $n = 5$ terms in the series expansion, all of them converged to a tolerance of 10^{-6} with just two GMRES iterations. The simulations with $n = 5$ took 4 iterations to converge. None of them required a restart of the GMRES. Moreover, the problems were studied with $AdjTol = 2$, meaning that source points that are more distant than two lengths from a reference element are considered far enough to receive GFMM expansions. This tolerance is required in order to obtain a fairly accurate matrix-vector product inside the GMRES iteration, thus letting the solution converge in 2, or maybe 3, iterations.

The computational time to run the simulation with the largest number of degrees of freedom $(2,097,152)$ for $n = 15$ terms in the series expansion was about 2 hours. The simulation for $16,384$ degrees of freedom with the conventional BEM took about 3 hours to run. Thus, a GFMM execution with
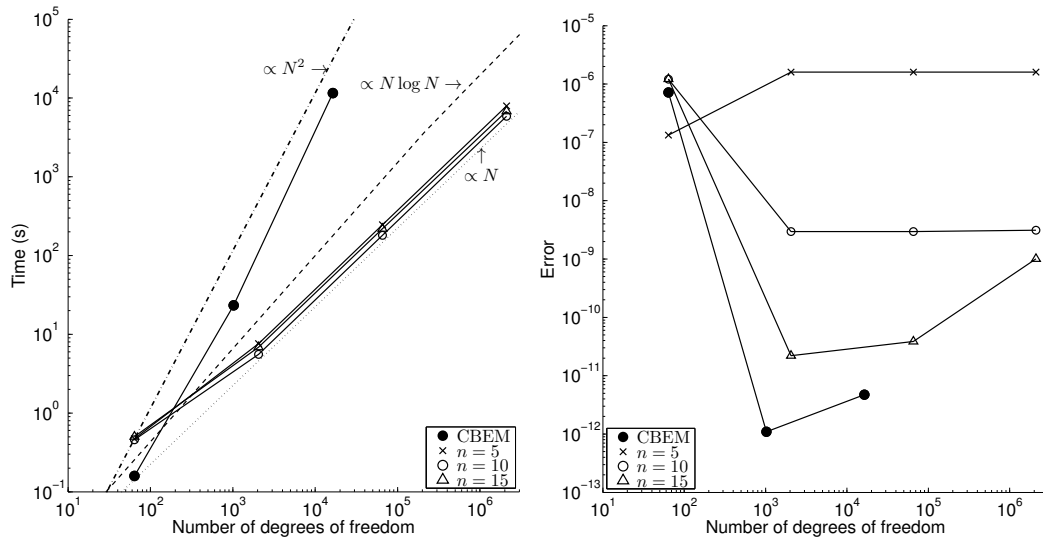
Figure 7.10: Execution times (left) for the solution for the potential problem on the annular domain in fig. 7.9 with quadratic elements, and accuracy results (right) for different numbers of expansion terms n.

128 times the number of degrees of freedom as in the BEM case took 67% of the running time for the same error tolerance.

8
Conclusions

The main goal of this thesis was to propose, develop and validate a kernel-independent Fast Multipole Method applied to both conventional and hybrid Boundary Element Methods, by further extending the author's [20] and Novelino's [18] Master dissertations, which were based on a draft paper proposed by their advisor [43]. Several features and mathematical aspects were still open subjects after these two works, and are now presented in this thesis. Novelino [18] laid the basics of the application of the GFMM – developed by the author [20] – to potential problems in the conventional BEM. This initial work was limited to convex domains, as no strategy to handle geometrical neighbourhood was available in connection with the hierarchical refinement strategy proposed. In this thesis, this limitation is removed by a simple, yet efficient, strategy to assess multiply-connected domains. The results in Section 7 present some complex and challenging domains, such as the one in fig. 7.1, which are successfully handled by the proposed extension of Novelino's developments.

Furthermore, the application of the GFMM to elasticity problems is also shown in this work and can easily be extended, in the framework of the GFMM, to other problems with vector fields, but with different kernels. The Simplified Hybrid BEM is also approached and its application together with the GFMM is formulated for both potential and elasticity problems.

Several numerical assessments are discussed in Section 7 for a variety of domains. It is shown that the Fast Multipole Method is in fact a powerful algorithm as suggested by Dongarra and Sullivan [2]. The GFMM is capable of delivering a matrix-vector multiplication with a running time complexity as low as $\mathcal{O}(N)$, which is the best performance achievable, according to Liu [3]. Almost all results presented in Section 7 support this statement, which puts the GFMM in a competitive position when compared with the conventional FMM, letting aside the major feature of being able to handle general kernels [20, 43].

Results at internal points, evaluated via Somigliana's identity, are also dealt with in this work for both potential and elasticity problems. The evaluation of results for gradients and stresses pose no conceptual difficulties and will be dealt with in a future work. Although the application of the GFMM

to the evaluation of results at internal points reduces the computational effort significantly when compared to conventional BEM methods, the speed-up is not as astonishing as for the matrix-vector product evaluation in the frame of an iterative BEM solution. This is due to the fact that the implementation of the Somigliana's identity has a lower complexity than for assembling a complete BEM problem.

The last numerical example in this thesis illustrated the complete, iterative solution of a BEM problem that uses the matrix-vector multiplication developed in the frame of the GFMM. The GMRES solver [32] was chosen due to its widespread use in the BEM literature such as by Liu [3] and Nishimura [4]. This solver is fairly stable, requires only one matrix-vector product per iteration, and usually converges within an acceptable error tolerance in just a few iterations without the need to use its restart feature – provided that one uses a preconditioner, such as the Jacobi block-diagonal.

Summing up, this work lays down a comprehensive study of the GFMM by separately assessing execution time and accuracy in the evaluation of a matrix-vector product for different problems and topology issues. The iterative solution of a linear equation system is dealt with more as a black box problem.

8.1
Suggestions for future research works

Although much effort has been dedicated to this thesis, time constraint has forced the author to leave significant developments to future works. Several features of the GFMM combined with the conventional and the hybrid boundary element methods are still open to further developments. Some of these research subjects are listed in the following.

– Develop the present GFMM-BEM algorithms for the accurate evaluation of quasi-singular integrals, as the basic literature is already available [36].

– Propose, formulate and validate a 3D version of the GFMM. Nishimura *et al.* [31, 3] uses solid harmonic functions to carry out the expansions. This is already the subject of a Ph.D research work in progress at PUC-Rio, also advised by Prof. Dumont.

– Develop an open-source BEM-GFMM framework in an efficient computational language, such as C++, aiming to disseminate the method's efficiency.

– Propose and implement a parallel version – whether in CPUs or Graphics Processing Units (GPUs) – of the current algorithms, as the GFMM clustering capabilities are naturally suited for this improvement.

– Consistently compare the efficiency of the proposed GFMM and BEM
  implementation with developments available in the literature.

– Propose, formulate and validate the present GFMM developments for
  applications in relation with the isogeometric BEM.

– Attempt to formulate a GFMM-BEM for large topology optimisation
  problems.

– Appplication of the GFMM to classical elasticity problems, as fracture
  mechanics and plate bending.

Bibliography

1 GREENGARD, L.; ROKHLIN, V. A fast algorithm for particle simulations. *Journal of Computational Physics*, v. 73, n. 2, p. 325–348, 1987.

2 DONGARRA, J.; SULLIVAN, F. Guest Editors Introduction to the top 10 algorithms. *Computing in Science & Engineering*, v. 2, n. 1, p. 22–23, 2000.

3 LIU, Y. *Fast Multipole Boundary Element Method*. Cambridge: Cambridge University Press, 2009.

4 NISHIMURA, N. Fast multipole accelerated boundary integral equation methods. *Applied Mechanics Reviews*, v. 55, n. 4, p. 299, 2002.

5 DARVE, E. The Fast Multipole Method: Numerical Implementation. *Journal of Computational Physics*, v. 160, n. 1, p. 195–240, 2000.

6 BREBBIA, C. A.; TELLES, J. C. F.; WROBEL, L. *Boundary Element Techniques*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1984.

7 HARBRECHT, H.; PETERS, M. Comparison of fast boundary element methods on parametric surfaces. *Computer Methods in Applied Mechanics and Engineering*, Elsevier B.V., v. 261-262, p. 39–55, 2013.

8 PHILLIPS, J. R.; WHITE, J. K. A Precorrected-FFT method for electrostatic analysis of complicated 3-d structures. *IEEE Transactions on Computer-Aided Design of Integrated Circuits and Systems*, v. 16, n. 10, p. 1059–1072, 1997.

9 BEBENDORF, M. Approximation of boundary element matrices. *Numerische Mathematik*, v. 86, n. 4, p. 565–589, 2000.

10 KURZ, S.; RAIN, O.; RJASANOW, S. The adaptive cross-approximation technique for the 3D boundary-element method. *IEEE Transactions on Magnetics*, v. 38, n. 2, p. 421–424, 2002.

11 GHOLAMI, A. et al. FFT, FMM, or Multigrid? A comparative Study of State-Of-the-Art Poisson Solvers for Uniform and Nonuniform Grids in the Unit Cube. *SIAM Journal on Scientific Computing*, v. 38, n. 3, 2016.

12 YING, L.; BIROS, G.; ZORIN, D. A kernel-independent adaptive fast multipole algorithm in two and three dimensions. *Journal of Computational Physics*, v. 196, n. 2, p. 591–626, 2004.

13 CAO, Y. et al. A kernel-independent fast multipole BEM for large-scale elastodynamic analysis. *Engineering Computations*, v. 32, n. 8, p. 2391–2418, 2015.

14   LÉTOURNEAU, P.-D.; CECKA, C.; DARVE, E. Generalized fast multipole method. *IOP Conference Series: Materials Science and Engineering*, v. 10, 2010.

15   FONG, W.; DARVE, E. The black-box fast multipole method. *Journal of Computational Physics*, Elsevier Inc., v. 228, n. 23, p. 8712–8725, 2009.

16   YASUDA, Y.; SAKUMA, T. An effective setting of hierarchical cell structure for the Fast Multipole Boundary Element Method. *Journal of Computational Acoustics*, v. 13, n. 01, p. 47–70, 2005.

17   DUMONT, N. A. The boundary element method revisited. *Boundary Elements and Other Mesh Reduction Methods XXXII*, WIT Press, v. 50, p. 227–238, 2010.

18   NOVELINO, L. S. *Application of Fast Multipole Techniques in the Boundary Element Methods*. Thesis (Master's thesis (in portuguese)) — PUC-Rio, 2015.

19   DUMONT, N. A.; PEIXOTO, H. F. C. A fast-multipole implementation of the simplified hybrid boundary element method. In: ALIABADI, M.; TEZER-SEZGIN, M.; KARASÖZEN, B. (Ed.). *Advances in Boundary Element Techniques XVII*. Eastleigh: EC, Ltd, UK, 2016. v. 01, p. 51–58.

20   PEIXOTO, H. F. C. *A study of the fast multipole method applied to boundary element problems*. 63 p. Thesis (Master's thesis (in portuguese)) — PUC-Rio, 2014.

21   DUMONT, N. A.; AGUILAR, C. A. The best of two worlds: The expedite boundary element method. *Engineering Structures*, v. 43, p. 235–244, 2012.

22   CHAVES, R. A. P. *Study of the Hybrid Boundary Element Method and the proposal of a simplified formulation*. Thesis (Master's thesis (in portuguese)) — PUC-Rio, 1999.

23   BREBBIA, C. A. *Boundary Element Method for Engineers*. 1. ed. London: Pentech Press Ltd, 1978.

24   DUMONT, N. A. The Hybrid Boundary Element Method. In: _____. *Mathematical and Computational Aspects*. Berlin, Heidelberg: Springer Berlin Heidelberg, 1987. p. 117–130.

25   AGUILAR, C. A. *An expedite implementation of the hybrid boundary element method for potential and elasticity problems*. 87 p. Thesis (PhD thesis (in portuguese)) — PUC-Rio, 2013.

26   DUMONT, N. A. The Hybrid Boundary Element Method: An Alliance Between Mechanical Consistency and Simplicity. *Applied Mechanics Reviews*, v. 42, n. 11S, p. S54, 1989.

27   OLIVEIRA, M. F. *Conventional, Hybrid and Simplified Boundary Element Methods*. Thesis (Master's thesis (in portuguese)) — PUC-Rio, 2004.

28   DUMONT, N. A.; PEIXOTO, H. F. C. A fast multipole implementation of the simplified hybrid boundary element method: application to 2D potential problems. *European Journal of Computational Mechanics*, 2017.

29   BOARD, J.; SCHULTEN, L. The fast multipole algorithm. *Computing in Science & Engineering*, v. 2, n. 1, p. 76–79, 2000.

30   PEIRCE, A. P.; NAPIER, J. A. L. A spectral multipole method for efficient solution of large-scale boundary element models in elastostatics. *International Journal for Numerical Methods in Engineering*, John Wiley & Sons, Ltd, v. 38, n. 23, p. 4009–4034, 1995.

31   NISHIMURA, N.; YOSHIDA, K.-i.; KOBAYASHI, S. A fast multipole boundary integral equation method for crack problems in 3D. *Engineering Analysis with Boundary Elements*, v. 23, n. 1, p. 97–105, 1999.

32   SAAD, Y.; SCHULTZ, M. H. GMRES: A Generalized Minimal Residual Algorithm for Solving Nonsymmetric Linear Systems. *SIAM Journal on Scientific and Statistical Computing*, v. 7, n. 3, p. 856–869, 1986.

33   PEIXOTO, H. F. C.; DUMONT, N. A. On the fast-multipole implementation of the simplified hybrid boundary element method. *Revista Interdisciplinar de Pesquisa em Engenharia - RIPE*, v. 2, n. 7, p. 167–185, 2016.

34   PEIXOTO, H. F. C.; NOVELINO, L. S.; DUMONT, N. A. A fast-multipole unified technique for the analysis of continuum mechanics problems with the boundary element methods. In: *XXXVI Ibero-Latin American Congress on Computational Methods in Engineering*. [S.l.: s.n.], 2015. p. 16.

35   DUMONT, N. A. *Lecture notes in Complex variable boundary element developments for 2D potential and elasticity problems*. Rio de Janeiro: PUC-Rio, 2016.

36   DUMONT, N. A. The collocation boundary element method revisited: Perfect code for 2D problems. *International Journal of Computational Methods and Experimental Measurements*, v. 6, n. 6, p. 965–975, 2018.

37   MOGILEVSKAYA, S. G.; LINKOV, A. M. Complex fundamental solutions and complex variables boundary element method in elasticity. *Computational Mechanics*, v. 22, n. 1, p. 88–92, 1998.

38   SADD, M. H. *Elasticity: theory, applications, and numerics*. [S.l.: s.n.], 2005. 473 p.

39   DUMONT, N. A. An assessment of the spectral properties of the matrix G used in the boundary element methods. *Computational Mechanics*, v. 22, n. 1, p. 32–41, 1998.

40   DUMONT, N. A.; PEIXOTO, H. F. C. A Fast-Multipole Unified Technique For The Analysis Of Potential Problems With The Boundary Element Methods. *Proceedings of the Indian National Science Academy*, v. 82, n. 2, p. 289–299, 2016.

41   AYACHOUR, E. A fast implementation for GMRES method. *Journal of Computational and Applied Mathematics*, v. 159, n. 2, p. 269–283, 2003.

42 BARRETT, R. et al. *Templates for the Solution of Linear Systems: Building Blocks for Iterative Methods, 2nd Edition*. Philadelphia, PA: SIAM, 1994.

43 DUMONT, N. A. Draft paper: The fast multipole algorithm for a general function. PUC-Rio, Rio de Janeiro, 2012.