Grups d'Estudi de Matemàtica i Tecnologia

Barcelona, July 2008

Edited by

Aureli Alabert (UAB)
Enric Fossas (UPC)
Joan Solà-Morales (UPC)

fMε FACULTAT de MATEMÀTIQUES i ESTADÍSTICA

UPC UNIVERSITAT POLITÈCNICA DE CATALUNYA

CRM^R CENTRE DE RECERCA MATEMÀTICA

i·math ingenio matematica consolider ingenio2010

Presentation

This booklet collects the problems that were proposed at the *Grups d'Estudi de Matemàtica i Tecnologia* (Study Groups of Mathematics and Technology, GEMT 2008), and the reports on the solutions to these problems. The problems belong to different areas of Mathematics and were proposed by companies and institutions in various ways. The event took place in the premises of the Facultat de Matemàtiques i Estadística of the Universitat Politècnica de Catalunya, Barcelona, from July 8 to 10, 2008.

We believe that the results are very satisfactory. Of course, with more time for work and discussions the results could have been more complete. However, it should be kept in mind that the goal of these Study Groups is just to make a first contact with the problems, that can continue after the event in many ways.

The participation was free of any cost for companies and institutions and also for participants. We are especially grateful to the two institutional co-organizers, the Facultat de Matemàtiques i Estadística of the Universitat Politècnica de Catalunya (FME) and the Centre de Recerca Matemàtica (CRM). They both contributed by strongly supporting these Study Groups. We also thank the Universitat Politècnica de Catalunya (UPC) and Ingenio Mathematica (i-MATH) for their financial support. Additionally, we have worked in coordination with the *Jornadas de Consulta Matemática para Empresas e Instituciones*, organized by CESGA in Santiago de Compostela, an event with the same aims as ours.

Finally, we thank very much all the participants: the companies and institutions that presented the problems and also the researchers, in a number of around thirty, who contributed to the discussions and wrote the final reports.

Barcelona, December 2008

Aureli Alabert, Enric Fossas, and Joan Solà-Morales

Statement of the Problems

Improvements in the resolution of emission images for Nuclear Medicine obtained using a gamma camera

by Santiago Aguadé Bruix, Nuclear Medicine,
Hospital Universitari Vall d'Hebron, Barcelona, Spain

1.1   Introduction to Nuclear Medicine

For centuries, discoveries of the principles of physics have been applied to knowledge of human beings. Three important findings marked the beginning of the radiological age (use of radioactivity for the first time for scientific and/or medical ends) at the end of the 19th century:

- the discovery of X-rays by Röntgen in 1895,

- the discovery of ionizing uranium radiation and its effect on the skin by Becquerel in 1896, and

- the discovery of natural radioactivity by Marie Curie in 1898.

Later, successive physical developments and major technological advances enabled the appearance and development of new specialities related to the use of nuclear energy and ionizing radiation in industrial and medical applications.

1.2   The bases of image detection systems in Nuclear Medicine

These systems are based on the phenomenon of solid scintillation, which is used to detect the gamma radiation emitted by the subject of study, following the introduction of a gamma emitter together with a molecular vehicle (from a radiotracer) into the organism of the subject (normally via intravenous injection).

In the "in vivo" diagnosis this is carried out using the gamma radiation of the radioactive isotope, habitually using a solid scintillation crystal made of sodium iodine activated with thallium (Na I(Tl)). The high atomic number for iodine (53) makes it a good absorber of gamma rays and its optical transparency is very good. The activation with thallium means that the energy levels give rise to jumps of some 3 eV (visible ultraviolet). The quantity of light photons is proportional to the energy deposited in the crystal.

The light produced in the crystal is captured by a photonic amplifier (the first element of the photomultiplier tube = photodetector which converts a photon to electron), giving rise to an electrical impulse in the photomultiplier that becomes higher as more light is emitted in the detection process. Therefore, the energy deposited by the gamma ray is proportional to the height of the electric impulse generated.

We can therefore distinguish between photons in Compton Collision (with partial release of energy) and those experiencing photoelectrical effects (maximum height impulses that represent the total energy of the gamma ray). We can also differentiate among the different energy types in incident rays, and therefore the different gamma emitting radioactive isotopes that can be detected at the same time. Now it is a simple case of counting (understood as integrating in time) the number of impulses from each energy peak (Z) or defining electronic windows that enable only impulses of a given height to be counted (energy window).

There remains the problem of the spatial situation of the photons.

1.3   Gamma camera

To obtain Nuclear Medicine images, the most frequently used "in vivo" detection system are the Anger gamma cameras (1958). These have provided a qualitative leap in nuclear medical explorations since they allow morphological and functional images to be obtained from the detection of radiation from the tracer from inside the organism.

Gamma cameras (Figure 1.1) consist of a special arrangement of a solid scintillator detector with a filter to direct the photons, a large diameter Na I(Tl) crystal with a thickness of between $\frac{1}{4}$ and $\frac{3}{8}$ inches, circular or rectangular, which is connected to a conglomerate of photomultiplier tubes (of 19, 37, 61, or more tubes) each fitted with a circuit for pre-amplification (Figures 1.2(a) and 1.3).

When the radiation falls on the crystal after passing through the lead collimator that defines the direction of the registered photons (the solid angle for each individual pinhole), a light scintillation is produced scattering light

through the crystal. The fraction of this light that reaches the photocathode in each photomultiplier varies inversely with the distance from the photon interaction point, meaning that the impulse of the greatest amplitude is registered in the photomultipliers nearest the place where the light struck and that of the smallest in the further points.
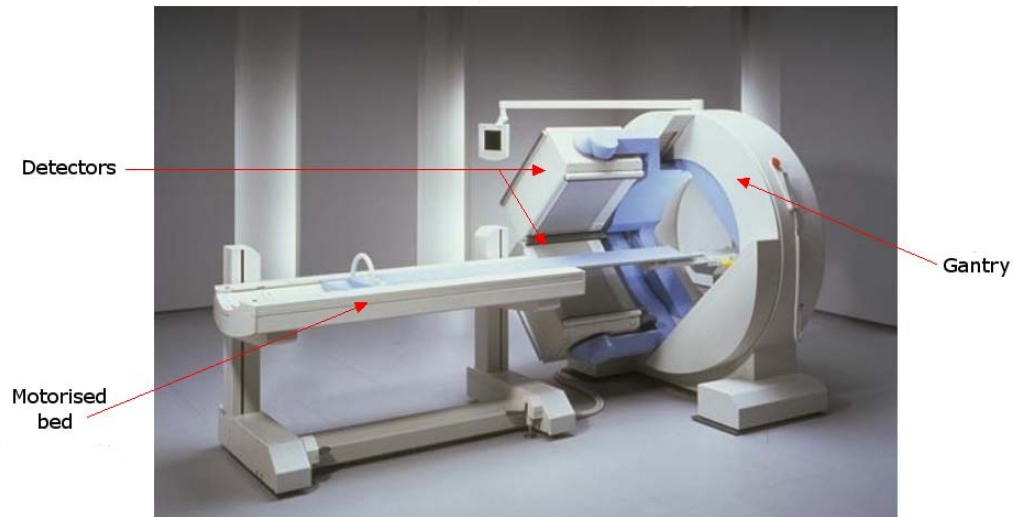

Figure 1.1: Gamma camera.

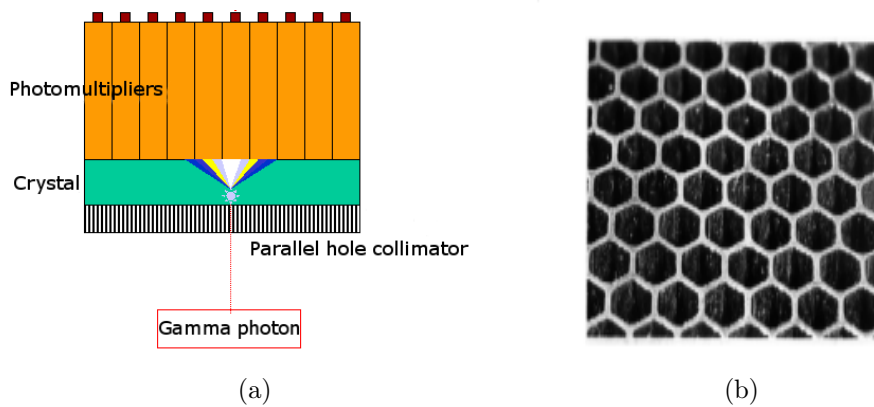(a)   (b)

Figure 1.2: (a) Elements of the Gamma camera. (b) Parallel hole collimator.

The distribution of the amplitude pulses of the photomultipliers contains information relative to the geometrical position in a system of Cartesian
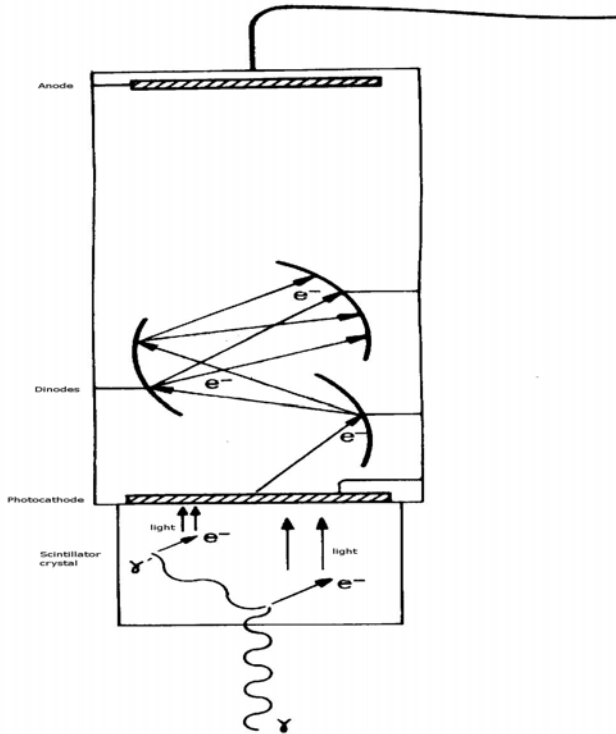
Figure 1.3: Pre-amplification.

coordinates centered on the detection field ($X-/X+$ and $Y-/Y+$, see Figure 1.4(a)), with the spatial resolution being therefore much greater than the number of photomultipliers and clearly marked by the acquisition matrix of the computer image (with certain limitations caused by the system as a whole).
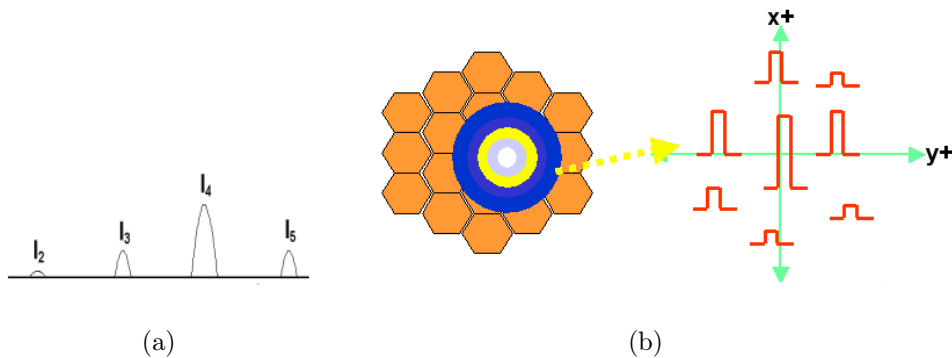

(a)   (b)

Figure 1.4: (a) Signal pulses. (b) Photomultiplier matrix and positioning.

The energy of interaction is defined by the amplitude of pulse Z, which is obtained as the sum of the all photomultiplier pulses and is proportional to the energy released by the photon. The final information on the position of the interaction on the crystal is obtained from the greater of the X/Z and Y/Z signal relationships, and these are then digitalized using an analogue-digital converter for computer processing and storage.

Carrying out this same process with each detected photon and assigning each one a place on the image gives us the radiotracer distribution map for the organs being studied in the detection plane (planar image, Figure 1.5).

In order to obtain volumetric information about the organ being studied, planar images may be taken from different angles of the body (rotating the detector head) and mathematically reconstructing the point from which each emission comes, normally by filtered retro-projection, thus obtaining sectional images (transverse, sagittal and coronal planes) and producing three-dimensional information (Single Photon Emission Computed Tomography, SPECT).

1.4   Questions

Improvements in planar images, which suffer from the problems of all the elements of which the gamma camera is made, such as

- collimator (size of hole, thickness of the septum, the depth of the hole, the total number of holes covering the detector);

- crystal (thickness of the crystal, density of the crystal, light transparency, recovery speed);

- photomultiplier (sensitivity of the photodetector, number of amplification stages, level of the exit signal);

- electronics: preamplifiers, integrators, CAD, positioners (currently, the digital signal (x, y, z) for every event, delay transfer time and electronic delays),

would give a better definition of the planar images, and if they are taken using SPECT this would also generate improvements in the images of sections obtained from the organs being studied.

Improvements are therefore necessary in the areas of

- collimator: corrections by solid angle, correction by the distance of the organ to the collimator;
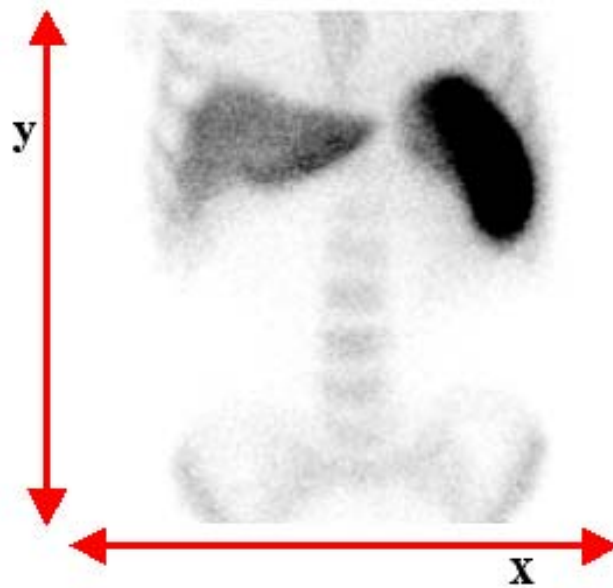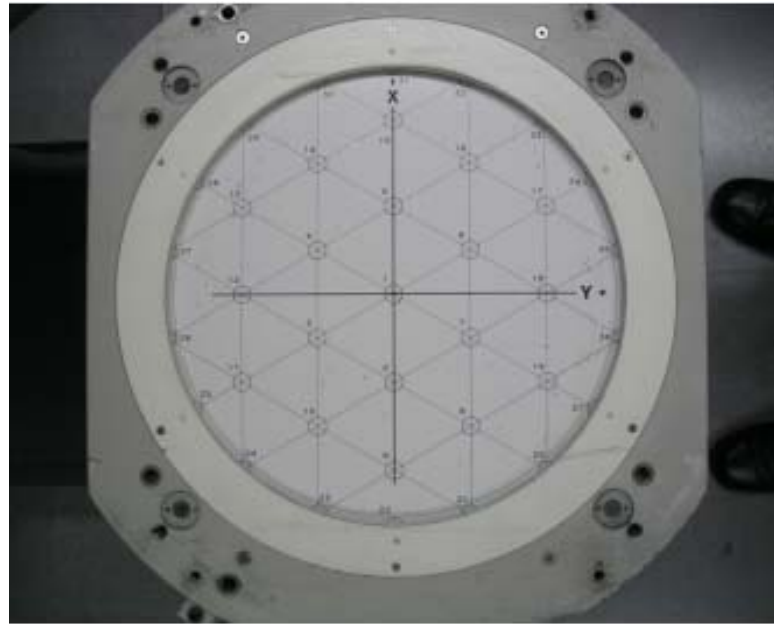
Figure 1.5: Planar image.

- crystal: a very thick crystal diffuses the image, but it also increases the number of gamma photons that release all the energy to the crystal itself, the dispersion of the light, reflection and refraction which increases the area of visible impact;
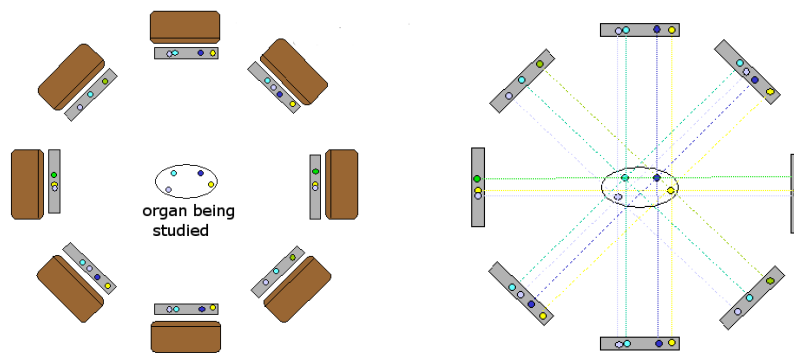
Figure 1.6: Acquisition of planar images from different angles and reconstruction.
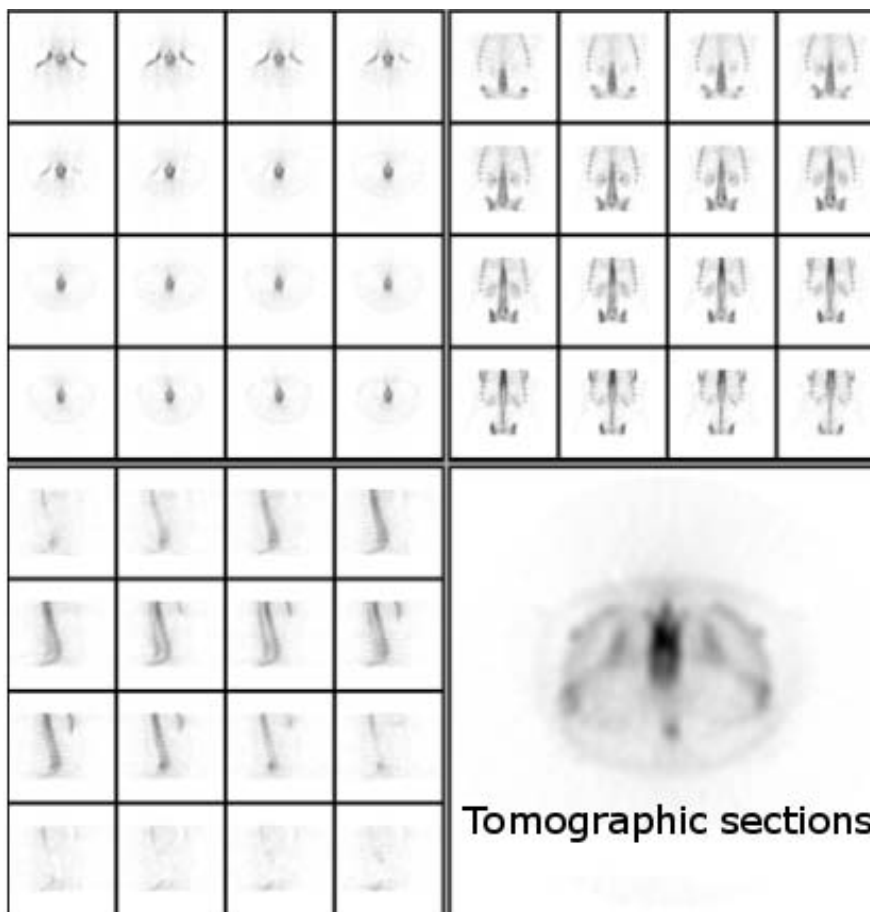

Figure 1.7: Images of the tomographic sections of a SPECT of bones in the thoracoabdominal region.

- photomultiplier: changing the number or distribution of the photomultipliers;

- electronics: reduction of the electronic delay time, improvement in positioning with weighted algorithms of the X/Y position.

In order to make these improvements, one could either redesign the equipment (which is not feasible) or apply the necessary corrections to each image to overcome the deficiencies in obtaining them (recommended solution that is broadly applicable).

From a practical point of view, the images obtained are not those seen directly by the detector. They already include default corrections for the following characteristics:

Energy Each gamma-emitting isotope has the energy of its photons defined. The most widely used in nuclear medicine is ^{99m}Tc (technetium 99 meta-stable) which emits gamma photons at 140 KeV. The detector obtains a map of this energy for all the photomultipliers (individually as an energy curve and number of events) and thereby maintains the energetic uniformity of the whole detector.

Linearity The use of a special collimator made from a lead matrix with holes aligned at a known distance, images of linearity (as in Figure 1.8(a)) are obtained which is how as the detector views a flat, homogeneous emission source. This image, similar to that used for the correction of eyeglass stigmatism, is used to generate a corrective matrix and restore the linearity of the detector. The effect of greatest density can be seen in the center of each photomultiplier.

Homogeneity To ensure that the detector has the same efficiency at any point, an adjustment is made of the detection uniformity of a homogeneous flat source, always after the application of two initial corrections. The image obtained must be visually homogeneous and the numerical calculations for the minimum and maximum differences in the field of vision and in the center of the field must be below 5%.

Despite these adjustments, the overall quality of the images is low as a result of the detection system itself and the random nature of the radioactive emissions, which are toxic in large quantities and therefore the amounts administered to patients are very small.

Considering that the radioactive dose unit is the Becquerel (equivalent to 1 disintegration per second, or one atom emitting 1 gamma photon per
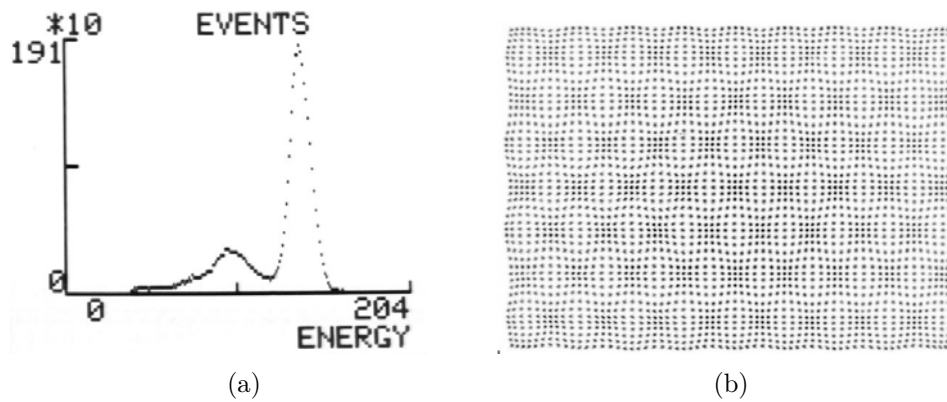
Figure 1.8: (a) Energy. (b) Linearity.

Figure 1.9: Homogeneity.

second), the number of ^{99m}Tc atoms administered in a routine study may be from 37 to 900 MBq (mega-Becquerels), strictly controlled by legislation that limits these values.

This involves a low photon flux, or a weak detection signal, which despite the high level of detector efficiency, gives images with high levels of noise (increase in the background signal) and poor contrast (very little difference between maximum and minimum) and these are the images on which this type of correction is really necessary.

Bibliography

[1] *IAEA Quality Control Atlas for Scintillation Camera Systems* (PDF).

[2] B. M. W. Tsui, *Radioactivity, Nuclear Medicine Imaging and Emission Computed Tomography*, Physics in Nuclear Medicine (PDF) (sections 3.3 to 3.5).

[3] M. F. Kijewski, *Image Quality*, Physics in Nuclear Medicine (PDF).

[4] F. D. Rollo, J. A. Patton, *Freeman and Johnson's Clinical Radionuclide Imaging*, Grune and Stratton Inc., Orlando, 1984, 203–259.

[5] J. Pavía, *Instrumentación para la obtención de imágenes en medicina nuclear. Introducción a la bioingeniería*, 12, Barcelona, Boixareu, 1988, 159–165.

[6] J. Pavía, D. Ros, *Procesado de imágenes biomédicas: Medicina Nuclear. Introducción a la bioingeniería*, 18, Barcelona, Boixareu, 1988, 251–257.

[7] *NEMA Standards Publications NU 1–1986, Performance Measurements of Scintillation Cameras*, NEMA, Washington DC, 2007.

[8] R. J. English, S. E. Brown (eds.), *Single Photon Emission Computed Tomography: A Premier*, New York, The Society of Nuclear Medicine, 1986.

[9] R. Puchal (ed.), *Filtros de imagen en Medicina Nuclear*, Nycomed Amersham Ed., Madrid, 1997.

Routing and dimensioning problems in transport networks

by Alpár Jüttner, Ericsson Research, Budapest, Hungary

2.1 On Resource Constrained Optimization Problems

Here we show a method that has long been used to solve Resource Constrained Optimization Problems and has been found extremely effective in practice. It is effective in the theoretical sense as well; it is proved to be strongly polynomial. In the special case of Resource Constrained Shortest Path Problems, a better running time estimation is also presented. In order to define the *Resource Constrained Optimization Problem* in general, first let us consider an underlying set E, a cost function $c\colon E \to \mathbb{R}^+$ and an abstract optimization problem

$$\min\left\{ \sum_{e \in P} c(e) \colon\ P \in \mathcal{P} \right\}, \tag{2.1}$$

where $\mathcal{P} \in 2^E$ denotes the set of feasible solutions. We refer to this problem as a *basic problem* in this chapter.

The corresponding *Constrained Optimization Problem* is the following. Let $d\colon E \to \mathbb{R}_+$ be another given weighting, called *delay*, and $\Delta \in \mathbb{R}_+$ a given constant called *delay constraint*. With these notations we are looking for the value

$$\min\left\{ \sum_{e \in P} c(e) \colon\ P \in \mathcal{P},\ \sum_{e \in P} d(e) \leq \Delta \right\}. \tag{2.2}$$

An important example for this is the *Constrained Shortest Path Problem*. Assume that a network is given as a directed, connected graph $G = (V, E)$, where V represents the set of nodes, and E represents the set of directed links. Each link $e \in E$ is characterized by two nonnegative values, a cost $c(e)$ and a delay $d(e)$. With a given delay constraint $\Delta \in \mathbb{R}_+$ and two given

nodes $s, t \in V$, the task is to find a least cost path P between s and t with the side constraint that the delay of the path is less then Δ.

One can define the *Constrained Minimum Cost Perfect Matching Problem* and the *Constrained Minimum Cost Spanning Tree Problem* in the same way.

Although their unconstrained versions are easy to solve, the three problems mentioned above are $\mathcal{NP}$-hard (see e.g. [1]). A common way to find near optimal solutions to these problems is to get rid of the additional constraint by using Lagrangian relaxation. In this way the constrained problem turns into a maximization of a one-dimensional concave function (see Section 2.2).

A simple way to find the optimum of the relaxed problem is to use binary search, which is polynomial for integer costs and delays (see [2]).

Instead of using binary search, another simple and practically even more effective method —described in Section 2.3— has been found and applied independently by several authors. After [3] it is sometimes called the *Handler–Zang method*. The same method was used in [4] leading some people to call it the *BM method*. Other papers aim at further improvement either on the running time in practical cases [5] or on the quality of the found solutions [6].

Although this method has turned out to be particularly efficient in practical applications, the worst case running time of this method was unknown for a long time.

Mehlhorn and Ziegelmann showed that for the Constrained Shortest Path Problem the Handler–Zang algorithm is polynomial for integer costs and delays. If $c(e) \in [0, \ldots, C]$ and $d(e) \in [0, \ldots, R]$ for each $e \in E$, then it will terminate after $O\big(\log(|V|RC)\big)$ iterations. They also presented examples showing that this running time is tight for small costs and delays (i.e. if $R \le |V|$ and $C \le |V|$).

One may observe that this problem can be transformed to an extension of the LCFO (Least Cost Fractional Optimization) problem discussed by Radzik in [7]. Moreover, the strongly polynomial solution method proposed in [7] turns out to be equivalent to the Handler–Zang algorithm in this case, showing that the number of the iterations made by the Handler–Zang algorithm does not depend on the range of the cost and the delay functions, so this method is actually *strongly polynomial* if the basic problem can be solved in strongly polynomial time. This also shows that Mehlhorn's and Ziegelmann's bound on the running time is not tight for large costs or delays.

Here we show that the Handler–Zang algorithm takes $O\big(|E|^2 \log(|E|)\big)$ iterations for arbitrary constrained optimization problem.

For the Constrained Shortest Path Problem an even better bound can be shown. The number of iterations is proved to be $O\big(|E| \log^2(|E|)\big)$ in this case (see Theorem 7). The proofs are based on the nice technique proposed by T. Radzik, see [8].

2.2 Lagrangian relaxation

Lagrange relaxation [9, 10] is a common technique for calculating lower bounds, and finding good solutions to hard optimization problems. This section shows how it can be applied to the Resource Constrained Optimization Problem.

From now on we assume that there exists an algorithm $\mathcal{A}(c)$ which runs in time T and solves Problem 2.1 for any given cost function c.

Intuitively the presented Lagrangian relaxation method is based on the heuristic of minimizing the modified cost function $c_\lambda = c + \lambda \cdot d$ over $\mathcal{P}$ for an appropriate λ. For a given (fixed) λ we can easily calculate the minimal solution $p_\lambda \in \mathcal{P}$ of the basic problem. If for $\lambda = 0$ we get that $d(p_\lambda) \le \Lambda$, then we found an optimal solution for the constrained problem as well. If $d(p_\lambda) > \Delta$, we must increase λ in order to increase the dominance of delay in the modified cost function until the optimal solution with respect to c_λ suits the delay requirements.

Now, we show how Lagrangian relaxation helps us find the value of λ that gives the best result. Moreover, the algorithm will give an upper bound on the badness of the solution as a byproduct, based on the following well known claim.

Claim 1 *Let*

$$L(\lambda) = \min\{c\lambda(p) : \ p \in \mathcal{P}\} - \lambda\Delta. \tag{2.3}$$

Then $L(\lambda)$ is a lower bound to Problem 2.2 for any $\lambda \ge 0$. ☐

To obtain the best lower bound we need to maximize the function $L(\lambda)$; that is, we are looking for the value

$$L^* = \max_{\lambda \ge 0} L(\lambda), \tag{2.4}$$

and the maximizing λ^*. Some more properties of the function $L(\lambda)$ are given.

Claim 2 *L is a concave piecewise linear function, namely the minimum of the linear functions $c(p) + \lambda(d(p) - \Delta)$ for all $p \in \mathcal{P}$.* ☐

Claim 3 *For any $\lambda \ge 0$ and c_λ-minimal solution $p_\lambda \in \mathcal{P}$, $d(p_\lambda) - \Delta$ is a subgradient of L in the point λ.* ☐

This gives us the following:

Claim 4 *If $\lambda < \lambda^*$, then $d(p\lambda) \ge \Delta$, and if $\lambda > \lambda^*$, then $d(p_\lambda) \le \Delta$ for each c_λ-minimal solution p_λ.* ☐

Claim 5 *A value λ maximizes the function $L(\lambda)$ if and only if there are solutions p_c and p_d which are both c_λ^*-minimal and for which $d(p_c) \geq \Delta$ and $d(p_d) \leq \Delta$. (If p_c and p_d coincide, then $d(p_d) = d(p_c) = \Delta$.)* $\square$

The Handler–Zang algorithm will give these solutions along with λ^*.

Claim 6 *Let $0 \leq \lambda_1 < \lambda_2$, and let $p_{\lambda_1}, p_{\lambda_2} \in \mathcal{P}$ be λ_1-minimal and λ_2-minimal solutions. Then $c(p_{\lambda_1}) \leq c(p_{\lambda_2})$ and $d(p_{\lambda_1}) \geq d(p_{\lambda_2})$.* $\square$

These two latter claims together mean that the λ^* that maximizes the function $L(\lambda)$ gives the best modified cost function, that is λ^* is the smallest value for which there exists a c_λ^*-minimal solution p_d which satisfies the delay constraint.

2.3   The Handler–Zang method

In this section the Handler–Zang algorithm is described.

- In the first step the algorithm sets $\lambda = 0$. It calculates an optimal solution with respect to the modified cost function c_λ. This means that the algorithm finds the c-minimal solution. If this solution meets the delay requirement Δ, it is an optimal solution of (2.2), and the algorithm stops.

- Otherwise the algorithm stores the solution as the best solution that does not satisfy the delay requirement Δ (it is denoted by p_c in what follows), and checks whether an appropriate solution exists or not: calculates the d-minimal solution. If the obtained solution suits the delay requirement, a proper solution exists, so the algorithm stores it as the best feasible solution found up to now (denoted by p_d). Otherwise there is no solution that fulfils the delay requirement, so the algorithm stops.

  In the next steps we obtain the optimal λ, by repeatedly updating either p_c or p_d with a new solution.

- Let us see the current solutions p_c and p_d. If, for a certain λ, both p_c and p_d are c_λ-minimal, then using Claim 5, this λ maximizes $L(\lambda)$. But it can be true only if $c_\lambda(p_c) = c_\lambda(p_d)$, from which we obtain that the only possible λ is

$$\lambda = \frac{c(p_c) - c(p_d)}{d(p_d) - d(p_c)}. \tag{2.5}$$

So, we set it as the new candidate for the optimal solution. Then we find a c_λ-minimal solution r. If $c_\lambda(r) = c_\lambda(p_c)$, then p_c and p_d are also c_λ-minimal, so we are done by setting $\lambda^* = \lambda$ and returning p_d. If $c_\lambda(r) < c_\lambda(p_c)$, then we replace either p_c or p_d with r according to whether it fails or fulfils the delay constraint, and repeat this step.

2.4 Running time of the algorithm

First of all, obviously

$$d(p_c^1) \geq d(p_c^2) \geq d(p_c^3) \geq \cdots > \Delta \tag{2.6}$$

and

$$d(p_c^1) \leq d(p_c^2) \leq d(p_c^3) \leq \cdots \leq \Delta, \tag{2.7}$$

and either $d(p_c^i) > d(p_c^{i+1})$ or $d(p_d^i) < d(p_d^{i+1})$ for any i. Since there is only a finite number of different solutions, the algorithm finds the optimal λ in a finite number of steps.

The following stronger result can be proved following the idea of the proof of the strong polynomiality of the so-called Newton method for fractional optimization problems [8].

Theorem 7 *The Handler–Zang algorithm terminates after $O(|E|^2 \log |E|)$ iterations, so the running time of the algorithm is $O(T|E|^2 \log |E|)$.* $\qquad\square$

In the case of the Constrained Shortest Path Problem, a better bound can actually be proved:

Theorem 8 *In the case of the Constrained Shortest Path Problem, the Handler–Zang algorithm terminates after $O(m \log^2 m)$ iterations, thus the full running time of the algorithm is $O(m^2 \log^2 m + mn \log^3 m)$.* $\qquad\square$

2.5 An open question: Handling multiple (two) resource constraints

The presence of two resource constraints is not as clear as the single constraint version, mainly because even the finding of a feasible solution is $\mathcal{NP}$-hard in this case. However, the Lagrange relaxation approach can also be applied here, and if the required constraints are not hard, the resulting solution is reasonably good. Even the Handler–Zang method extends to this case quite straightforwardly, but nothing is known about its running time (except that it is finite). Thus the question is: *Is the Handler–Zang method (strongly) polynomial in the case of two resource constraints?*

Bibliography

[1] M. R. Garey and D. S. Johnson, *Computers and Intractability – A Guide to the Theory of NP-Completeness,* W. H. Freeman and Company, New York, 1979.

[2] Mark Ziegelmann, *Constrained Shortest Paths and Related Problems,* PhD thesis, Universität des Saarlandes, July 2001.

[3] G. Handler and I. Zang, *A dual algorithm for the constrained shortest path problem,* Networks, **10**:293–310, 1980.

[4] David Blokh and George Gutin, *An approximation algorithm for combinatorial optimization problems with two parameters,* Australasian J. Combin., **14**:157–164, 1996.

[5] Alpár Jüttner, Balázs Szviatovszki, Ildikó Mécs, and Zsolt Rajkó, *Lagrange relaxation based method for the QoS routing problem,* In: Infocom, IEEE, April 2001.

[6] K. Mehlhorn and M. Ziegelmann, *Resource constrained shortest paths,* In: 8th Annual European Symposium on Algorithms (ESA), pages 326–337, LNCS 1879, 2000.

[7] T. Radzik, *Parametric flows, weighted means of cuts, and fractional combinatorial optimization,* In: P. Pardalos (ed.), Complexity in Numerical Optimization, pages 351–386, World Scientific, 1993.

[8] T. Radzik, *Fractional combinatorial optimization,* In: DingZhu Du and Panos Pardalos (eds.), Handbook of Combinatorial Optimization, Kluwer Academic Publishers, Dec. 1998.

[9] M. Held and R. M. Karp, *The traveling salesman problem and minimum spanning trees,* Operations Research, **18**:1138–1162, 1970.

[10] M. Held and R. M. Karp, *The traveling salesman problem and minimum spanning trees: Part II,* Mathematical Programming, **1**:6–25, 1971.

Ideal MHD equilibria of tokamak plasmas

by Alfredo Portone, Fusion for Energy, Barcelona, Spain

3.1 Background

Plasma MHD equilibrium and stability play a key role in tokamak physics and engineering since they have far reaching consequences in the design and operation of present and future fusion devices. As a result of this, the computation of axi-symmetric plasma equilibrium configurations and their stability analysis —particularly against $n = 0$ (i.e. axi-symmetric) displacement modes— has been widely investigated in the past years and is presently performed using a large range of computer codes [1]–[6].

The plasma equilibrium problem is often divided in two broad classes. Firstly —mainly for the Poloidal Field (PF) system design purposes— it is required to compute the currents in the PF coils to keep in equilibrium a plasma with the boundary as close as possible to a pre-defined flux line $\partial\Omega$ (Figure 3.1), total current I_p and current density profile (typically the plasma poloidal beta β_p and internal inductance l_i are assigned). We will refer to this as *inverse equilibrium problem* [7].

The position and size of each PF coil is generally known. However, in some cases it may be requested to *optimize* the position and shape of these coils to maintain the desired plasma equilibrium. This problem is generally a *constrained* optimization problem and will not be considered here.

Secondly —mainly for the analysis of transient electro-magnetic phenomena— it is necessary to compute the evolution of the plasma current, shape and current profile parameters, in the presence of assigned external currents, flowing in the PF coils (equilibrium currents) and in the surrounding metallic structures (eddy currents). This problem is the so-called *direct equilibrium problem* [3]–[4].

In between these two classes stands a third type of problem —most relevant for magnetic diagnostics purposes— in which the currents flowing in the

PF coils are specified together with the poloidal flux and/or field at some specific locations. In that case it is requested to compute I_p, the plasma boundary $\partial\Omega$ and, possibly, β_p and l_i (*plasma identification problem*) [1].

All these problems are closely related and their solution is based on the solution of the Grad–Shafranov equation [6] —with different boundary conditions— inside an a-priori unknown domain Ω_p whose determination is part of the problem itself.

The presence of Ω_p as an additional unknown complicates the solution of the problem that, in essence, is a fairly standard elliptic type problem with a (mild) non-linearity deriving from the fairly smooth dependence of the source term upon the flux function Ψ.

The aim of this study should be to find an efficient algorithm for the determination of the plasma region Ω_p which is the key unknown of the free boundary equilibrium problem.

3.2   Free boundary equilibrium problem

The basic equilibrium equation for any toroidal axi-symmetric plasma is obtained starting from the single fluid MHD equilibrium equation and quasi-static Maxwell laws [7]:

$$\nabla \times \mathbf{B} \;=\; \mu_0 \mathbf{j} \tag{3.8}$$

$$\nabla \cdot \mathbf{B} \;=\; 0 \tag{3.9}$$

$$\mathbf{j} \times \mathbf{B} \;=\; \nabla p \tag{3.10}$$

In the set above, (3.8) holds in the volume occupied by the plasma (V_p) whereas (3.9–3.10) hold in the whole space ($\mathbf{R}^3$). By writing (3.8–3.9) in cylindrical (r, φ, z) co-ordinates under the assumption of axi-symmetry $\partial/\partial\varphi = 0$, $\mathbf{B}$ and $\mathbf{j}$ can be expressed as:

$$\mathbf{B} \;=\; \mathbf{B}_p + \mathbf{B}_\varphi = \nabla\psi \times \nabla\varphi + \mu_0 T \nabla\varphi \tag{3.11}$$

$$\mathbf{j} \;=\; \mathbf{j}_p + \mathbf{j}_\varphi = \nabla T \times \nabla\varphi + L(\psi/\mu_0)\nabla\varphi$$

In (3.11), $\psi(r, z) = \int_0^r r' B_z(r', z)dr'$ is the poloidal magnetic flux per radian, p is the plasma kinetic pressure (zero outside V_p), $T \equiv r B_\varphi/\mu_0$ is the poloidal current per radian with B_φ the toroidal field and L is an elliptic operator (see below).

We can then formulate the free boundary equilibrium problem (3.8–3.10) as:

$$-\frac{\partial}{\partial r}\left(\frac{1}{r}\frac{\partial\psi}{\partial r}\right) - \frac{\partial}{\partial z}\left(\frac{1}{r}\frac{\partial\psi}{\partial z}\right) \equiv L\psi = \mu_0 j_\varphi \tag{3.12}$$
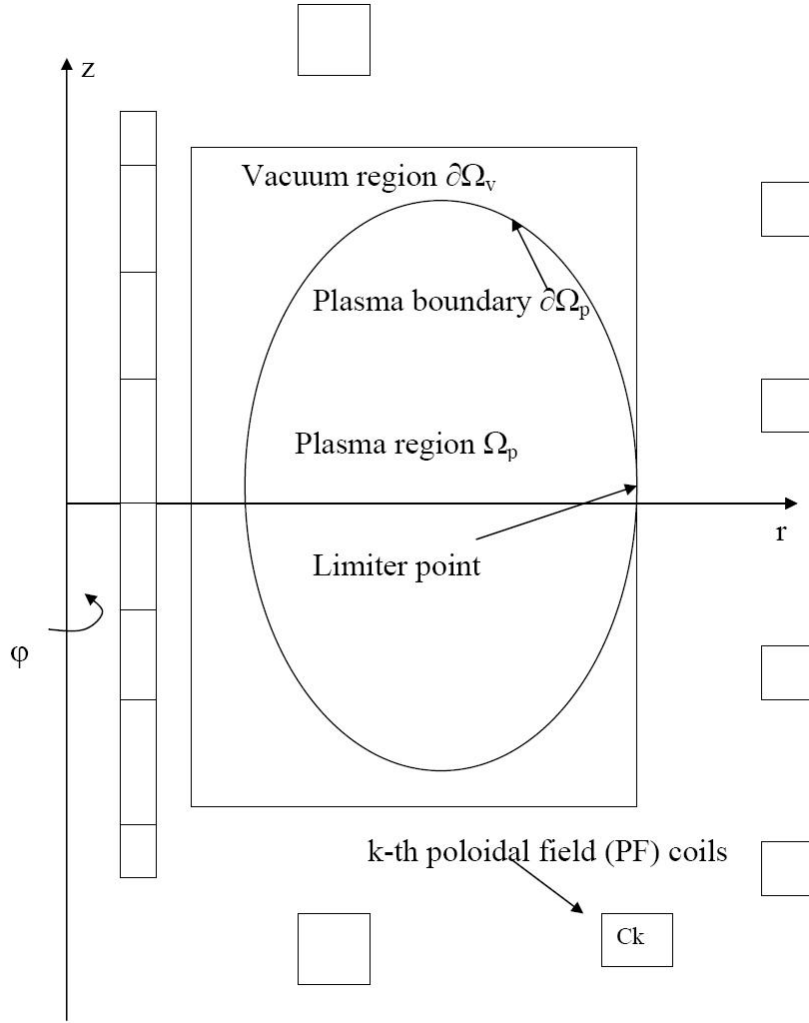
Figure 3.1: Poloidal plane.

where

$$
j_\varphi =
\begin{cases}
rp' + \left(\dfrac{\mu_0}{r}\right) TT' & \text{in } \Omega_p \\[2ex]
C_k & \text{in } \Omega_k \\
0 & \text{in } \Omega_v
\end{cases}
\tag{3.13}
$$

In (3.13), Ω_p is the *unknown* plasma cross-section in the poloidal (r, z) plane (Figure 3.1), Ω_v is the vacuum region surrounding Ω_p, where there is no current flow and Ω_k is the generic region with prescribed toroidal current density C_k. In Ω_p, p and T depend only upon ψ and the prime denotes the derivative with respect to ψ.

3.3 Numerical solution

The set (3.12–3.13) is usually solved by re-casting it as a non-linear eigenvalue problem [9]:

$$
\begin{aligned}
L\psi &= \lambda f\left(r, \overline{\psi}\right) \qquad \text{in } \Omega_p \\
\lambda \int_{\Omega_p} f\, d\Omega &= \mu_0 I_p
\end{aligned}
\tag{3.14}
$$

where the eigenvalue λ plays the role of a scaling factor for the plasma current density to match the total plasma current I_p [7] and the auxiliary variables are defined as:

$$
\begin{aligned}
\overline{\psi} &\equiv (\psi - \psi_a)/(\psi_b - \psi_a) \\
\psi_a &\equiv \sup_{\Omega} \psi \\
\psi_b &\equiv \sup_{\Omega \cup \partial\Omega} (\psi_L, \psi_X)
\end{aligned}
\tag{3.15}
$$

In (3.14), ψ_a is the maximum value of the poloidal flux ψ inside the whole computational domain $\Omega = \Omega_p \cup \Omega_v$, corresponding to the flux at the position of the plasma magnetic axis and ψ_b is the maximum value of the poloidal flux ψ among all the possible *limiter* (ψ_L) and *saddle* points (ψ_X).

By definition, a *limiter point* is a point on $\partial\Omega$ where the corresponding iso-flux line for the poloidal flux ψ is tangent to $\partial\Omega$ and completely enclosed inside Ω.

On the other hand, a *saddle* point is a point in $\Omega \cup \partial\Omega$ containing a stagnation (or X-point) for the flux function ψ and the corresponding iso-flux line is completely enclosed inside Ω. At a saddle point the gradient of ψ vanishes and the Hessian is neither positive nor negative defined.

The free boundary equilibrium problem is solved numerically by re-formulating the equations in weak form and solving them by the finite element method. The solution is sought among the linear combinations of finite element basis functions by expanding ψ on the finite element basis functions (ξ, ζ):

$$
\psi(r, z) = \sum_{i=1}^{N} \xi_i(r, z)\psi_i + \sum_{i=1}^{M} \zeta_i(r, z)\varphi_i = \xi^{\mathbf{T}} \cdot \psi + \zeta^{\mathbf{T}} \cdot \varphi.
\tag{3.16}
$$

In (3.16), ψ and ξ denote the vector of nodal values and basis functions associated with the N nodes inside $\Omega = \Omega_p \cup \Omega_v$, whereas φ and ζ refer to

the M nodes on the boundary $\partial\Omega$. In discrete form, the set to be solved during each iteration is:

$$
\begin{aligned}
A^{(n)}\psi^{(n+1)} &\equiv b^{(n)} \\
\psi_a^{(n+1)} &\equiv \sup_\Omega \psi^{(n+1)} \\
\psi_b^{(n+1)} &\equiv \sup_{\Omega\cup\partial\Omega} \left(\psi_L^{(n+1)}, \psi_X^{(n+1)}\right)
\end{aligned}
\tag{3.17}
$$

3.4 Goal of the study

The aim of this study should be to find an *efficient* and *robust* algorithm to evaluate $\psi_a^{(n+1)}$ and $\psi_b^{(n+1)}$ with the corresponding magnetic axis and boundary iso-flux line positions.

Such an algorithm will start from the knowledge of $\psi^{(\mathbf{n+1})}$ in the grid points inside Ω and of $\varphi^{(\mathbf{n+1})}$ on $\partial\Omega$ as well as the functional dependences of ψ through the finite element basis and it will evaluate $\psi_a^{(n+1)}$, $\psi_b^{(n+1)}$ and $\partial\Omega_p^{(n+1)}$.

Bibliography

[1] L. L. Lao et al., *Reconstruction of current profile parameters and plasma shapes in tokamaks*, Nuclear Fusion **25**, no. 11 (1985), 1611–1622.

[2] J. A. Croatinger et al., Rep. UCRL-ID-126284, LLNL, CA (1997).

[3] S. C. Jardin, N. Pomphrey and J. De Lucia, *Dynamic modeling of transport and positional control of tokamaks*, J. Comput. Phys. **66** (1986), 481–507.

[4] R. R. Khayrutdinov and V. E. Lukash, *Studies of plasma equilibrium and transport in a tokamak fusion device with the inverse-variable technique*, J. Comput. Phys. **109** (1993), 193–201.

[5] R. Gruber et al., *HERA and other extensions of ERATO*, Computer Physics Communications **21** (1981), 363–376.

[6] S. W. Haney et al., *Vertical stability analysis of tokamaks using a variational procedure*, Plasma Physics Reports **23**, no. 9 (1997), 798–804.

[7] K. Lackner, *Computation of ideal MHD equilibria*, Computer Physics Communications **12** (1976), 33–44.

[8] T. Takeda and S. Tokuda, *Computation of MHD equilibrium of tokamak plasma,* J. Comput. Phys. **93** (1991), 1–107.

[9] R. Temam, *A non-linear eigenvalue problem: The shape at equilibrium of a confined plasma,* Arch. Rat. Mech. Anal. **60** (1975), 51–73.

Answers to the Problems

Improvements in the resolution of emission images for Nuclear Medicine obtained using a gamma camera

by Sonia Fernández-Méndez and Jordi Saludes

4.1   Introduction

A gamma camera is a device used to image gamma radiation emitted by radioisotopes [1], through a technique known as scintigraphy. The applications of scintigraphy include nuclear medical imaging, to view and analyse selected tissues of the human body. Radionuclides are injected, inhaled or ingested by the patient, and fixed to the organ of interest, emitting gamma rays. The gamma camera receives these gamma rays to produce an image.

A gamma camera consists of one or more crystal planes, optically coupled to an array of photomultiplier tubes. The crystal scintillates in response to incident gamma radiation, producing a faint flash of light (called *event*), which is detected by the photomultipliers. The sum of events for every photomultiplier gives a bidimensional image with the relative spatial count density.

Nevertheless, to obtain reliable spatial information about the gamma emissions a method for correlating each detected photon with its point of origin is required. The conventional method is to use a *collimator*, which is a thick sheet of lead with thousands of adjacent holes. This plate is placed just in front of the detection crystal, blocking photons that do not come from a given cone of directions. Finally, the reconstructed image reflects the distribution and relative concentration of radioactive tracer elements present in the organs and tissues.

To get 3D volumetric information of the organ of interest, several planar images from different angles of view may be taken, leading to the so called SPECT (Single Photon Emission Computed Tomography). Usually two or three detectors are slowly rotated around the patient's torso, providing planar images for about 90 equally spaced angles. Using these planar images, a
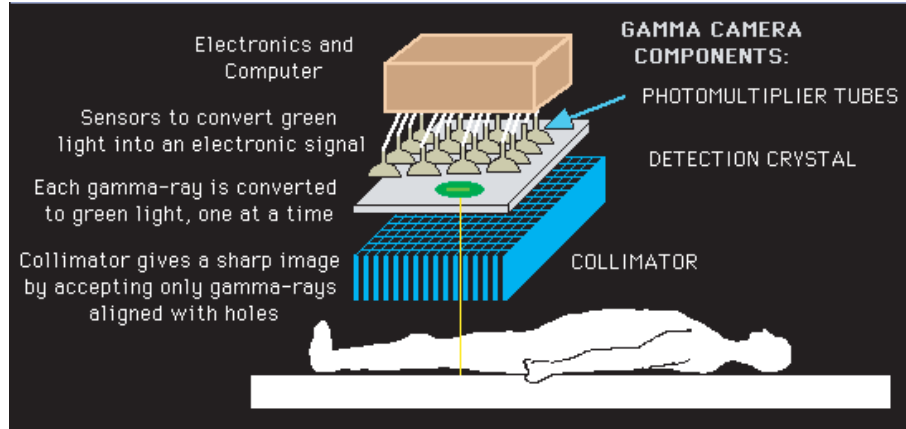
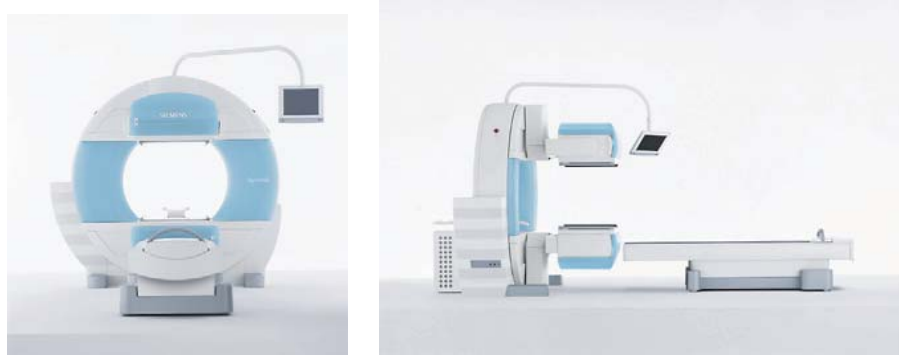Figure 4.1: Gamma camera graphical representation [4].

Figure 4.2: Siemens gamma camera [5].

tomographic reconstruction [2, 3], based on a filtered back projection, can then be computed to get cross-sectional images of patients.

Although gamma cameras are nowadays widely used in image diagnostics, for many applications the images obtained do not provide conclusive information, due to their poor resolution and contrast. For instance, the thickness of a normal heart is about 1.2 cm and most of the left ventricle muscle is minimum about 0.8 cm, whereas a common gamma camera can differentiate two separate sources of gamma photons located at a minimum of 1.8 cm apart, at 5 cm away from the camera face. Furthermore, the spatial resolution decreases rapidly at increasing distances from the camera face. Thus, the need for improvement in this technology is apparent.

The poor resolution of the images is due to several factors such as the low levels of radiation permitted for human patients, the influence of the collimator, the properties of the crystal and the electronic/computer postprocessing. In particular the collimator plays an important role:

- A normal collimator attenuates more than 99% of incident photons, and limits the sensitivity of the camera system;

- The size of the collimator holes affects the result (larger holes lead to more blurred images, but smaller holes limit the sensitivity);

- There can be some crosstalk between holes, blurring the result.

In addition, the images obtained from the gamma camera are not exactly the images received by the device. Some corrections are applied, as a postprocess of the image, such as the linearity or homogenization corrections. Due to the effect of the discrete holes in the collimator, the image obtained from a regular square grid is far from being regular. The linearity correction is close to a stigmatism correction to reduce this effect and try to reproduce straight lines with uniform intensity. The homogeneity correction aims at making the detection of radiation from a planar source uniform. Both are "make-up" techniques to improve the obtained image, but which unfortunately reduce the reliability of the data.

4.1.1   General remarks

After a presentation by Dr. Santiago Aguadé from *Hospital Universitari de la Vall d'Hebron*, describing the main issues related to gamma camera imaging, the GEMT devoted two days to exploring different strategies for the improvement of gamma camera images. The conclusions of this work are summarized in this manuscript, and come from discussions during the meeting, with contributions by Xavier Alameda, Sonia Fernández, Enric Fossas, Xavier Gràcia, Pablo Gutiérrez, Fernando Martínez, Josep Masdemont, Xevi Roca, Carmen Safont, Jordi Saludes, and Yolanda Vidal.

Two different approaches were adopted. The first approach aims at improving the images irrespective of their origin, using image improving techniques such as non-linear filters, super-resolution algorithms and MS-VST techniques (see Section 4.2).

The second approach proposes a simple model for the collimator component (see Section 4.3). Modelling of the scintillator or the photomultiplier sensor was not attempted.

4.2   Image improving techniques

4.2.1   Non-linear diffusion

Inspired in the use of partial differential equations, non-linear diffusion techniques have been extensively studied [6, 7, 8]. Convolving with a Gaussian

kernel gives an elementary denoising technique that corresponds to the solution of the linear heat equation with the image as initial datum. Other diffusion equation could be used for denoising, such as

$$\frac{d}{dt}I = \nabla(c\nabla I),$$

where $c(x,y)$ is the *diffusivity function*. In the case of constant c, we obtain the classical diffusion equation with homogeneous diffusivity: all features of the image are smoothed equally in this case.

This is not what we want, since we expect the edges to remain crisp. An improvement would be to make $c(x,y)$ sensitive to the local features of the image and, thus, make the equation one of non-linear diffusion type. For instance, using a function c that depends entirely on $|\nabla I|$, Perona and Malik showed in [6] that one can avoid diffusing the image in the neighborhood of an edge, while allowing the diffusion process to smear high-frequency noise in regions far from the edges. Since the diffusivity function c is scalar, the process is called *isotropic non-linear diffusion*. Another possibility is to consider *anisotropic diffusion* schemes like

$$\frac{d}{dt}I = \nabla(C\nabla I),$$

where C is now a matrix steering the diffusion process in directions other than the image gradient. This case and the use of the diffusion tensor has not been considered in this paper.

Figure 4.3 shows the application of the Perona–Malik non-linear filter to a gamma camera image. The processed images have been obtained with a Matlab code available in [9]. Results are shown for $10, 20, 30, 40, 50$ iterations (time steps), for a diffusion given by $c = 1/(1 + (|\nabla I|/K)^2)$, with edge threshold parameter $K = 1$, and time step $\Delta t = 0.2$. The gradient in the diffusion coefficient formula is computed through a convolution with a Gaussian with $\sigma^2 = 1$. The smoothing, but shape preserving, effect of the non-linear filter can be clearly observed. The image is clearly improved after just 10 iterations, with a much clearer edge definition. After 30 iterations the image almost reaches the stationary image (it is very close to the image with 40 or 50 iterations). This technique may be useful for applications in which edge (or any boundary detection) of organs is of interest. Nevertheless, a proper tuning (depending on the particular example and result of interest) of the parameters and number of time steps is mandatory.
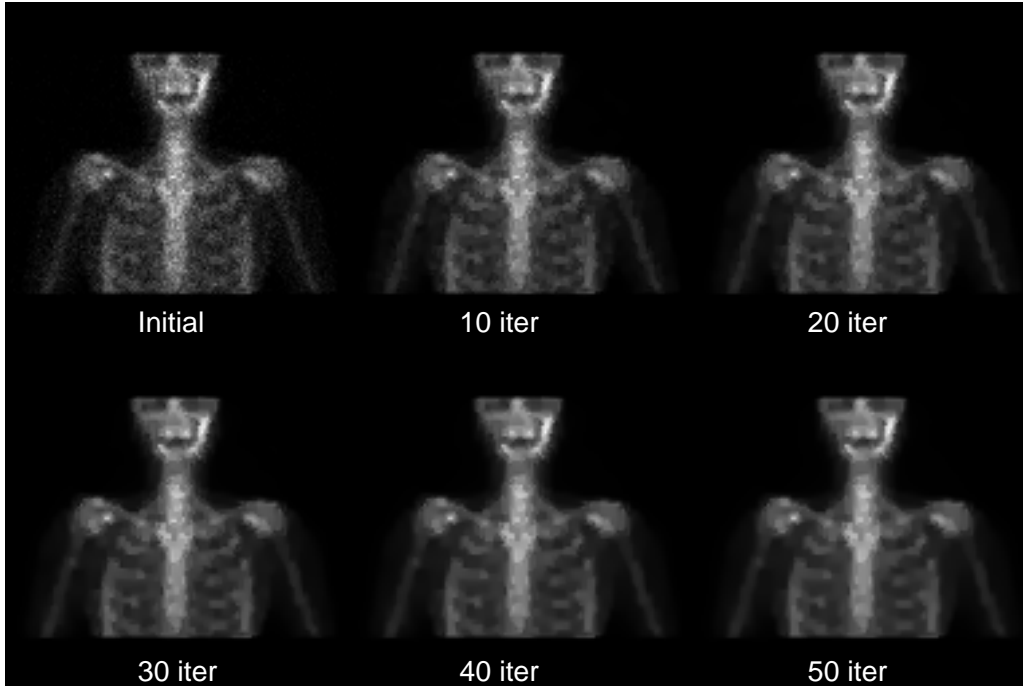
Figure 4.3: Perona–Malik non-linear filter with $K = 1$, $\Delta t = 0.2$, $\sigma = 1$, using the Matlab files in [9].

4.2.2 Multiple-frame super-resolution techniques

Super-resolution techniques aim to enhance the resolution of an imaging system, usually by acquiring more graphic information (considering several images of the same object), or by reconstructing the details lost due to the deficiencies of the optical system [11]. The most widely used super-resolution techniques are the multiple-frame algorithms, because they can be applied in any situation, without profound knowledge of the optical system. By fusing together several low-resolution images, one enhanced-resolution image is formed; see for instance [10, 12].

Figure 4.4 shows an example of super-resolution image processing using the PhotoAcute software, available in [10]. Five consecutive images, corresponding to rotations of the gamma camera with an angle increment of 4 degrees, are considered to clearly improve the resolution of the third image. This example illustrates how standard multiple-frame algorithms can be successfully applied in the context of gamma camera images.

Nevertheless, super-resolution algorithms based on the knowledge of the gamma camera device faultiness, or on the back-projected reconstruction [13], are also very promising and merit further investigation.
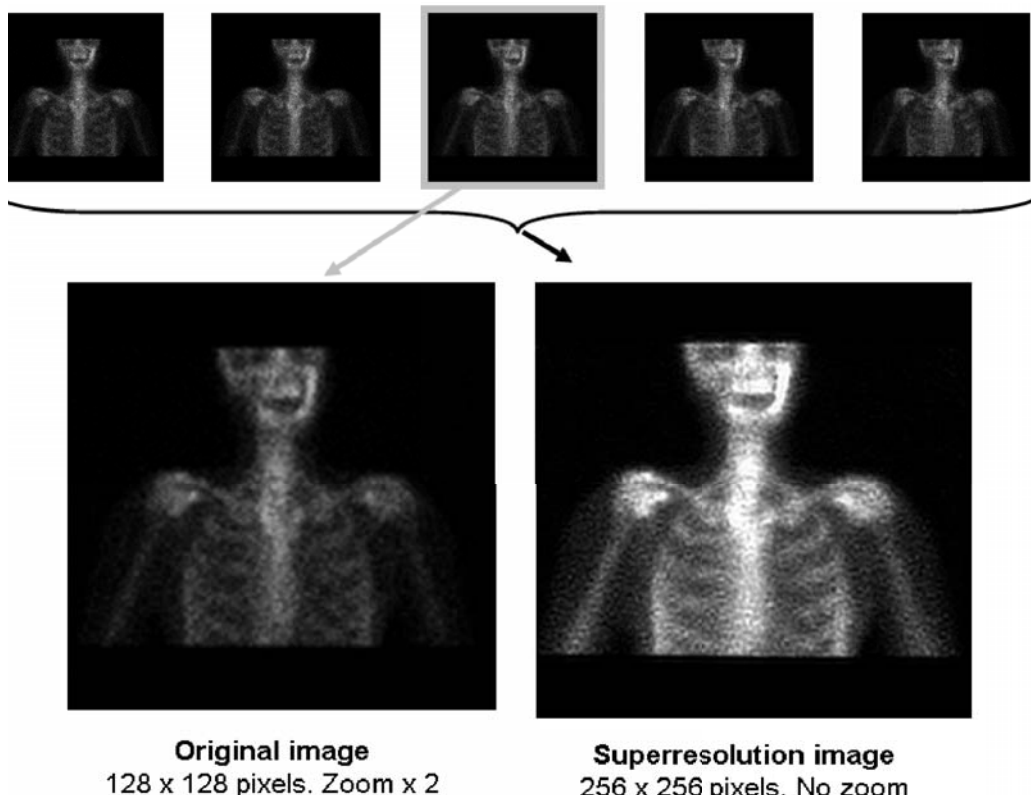
Figure 4.4: Super-resolution using PhotoAcute software [10].

4.2.3   MS-VST

Denoising techniques based on preprocessing the image using *variance stabilization transform* (VST) have been used. The approach of [14] describes a way to combine it with multiscale analysis via filter banks. This technique has proved valuable in fields like astronomy and astrophysics, which deal with extremely faint signals (modelled after Poisson processes) arranged on a grid and can be considered an extension of the Anscombe VST [15, 16]. The definition of *ridgelets* as 1-D wavelet transform to the slices of the Radon transform is effective on finding global lines of the image [17].

Figure 4.5 shows the application of the multiple-scale variance stabilization transform (MS-VST) to a gamma-camera image. The quality of the original image is too poor to obtain a satisfactory result. In Figure 4.6 a simple pre-process of the original image has been considered: the closer available images, corresponding to a gamma camera rotation of -4 and 4 degrees, have been translated -1 and 1 pixels respectively, and overlapped to the original image. The application of MS-VST to the overlapped image is shown.
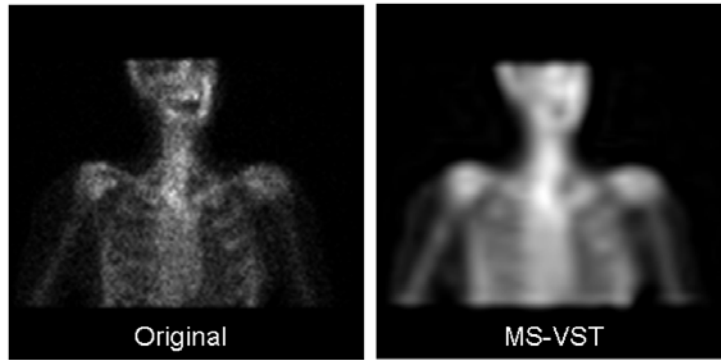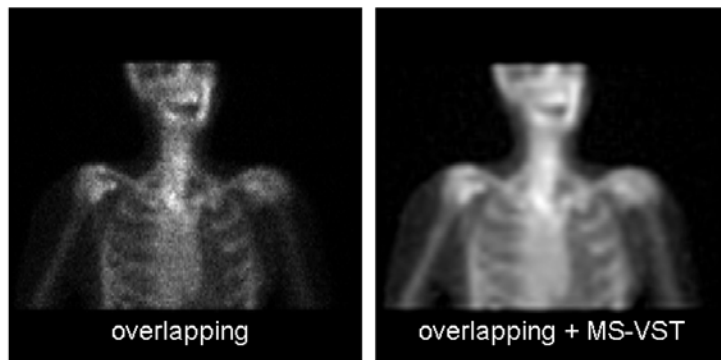
Figure 4.5: MS-VTS image processing.

Figure 4.6: MS-VTS image processing applied to a superposition of three consecutive images.

4.3 Collimator model

4.3.1 Modelling the imaging process without absorption

The collimator unit aims at blocking any ray which does not come from a direction perpendicular to the imaging plane. This is achieved by an array of long perforations on a lead layer positioned directly on top of the image sensors. In this way, the collimator effect can be modelled by

$$r(k) = \int_{H_k} \int_{\Omega} c\left(\frac{x - y}{|x - y|}\right) g(x)\, dx\, dy$$

where $g(x)$ denotes the emission rate of point $x \in \Omega \subset \mathbb{R}^3$, $y \in \Pi$, where Ω denotes the compact support of g, and Π is the imaging plane. The map $c\colon \mathbb{S}^2 \longrightarrow \mathbb{R}_+$ describes the absorption of the collimator walls and H_k denotes

the sensitive area of the k sensor. Ideally, c will be a Dirac's delta centered on the direction perpendicular to Π but in real devices there is a fair amount of crosstalk between adjacent sensors (see Figure 4.7).
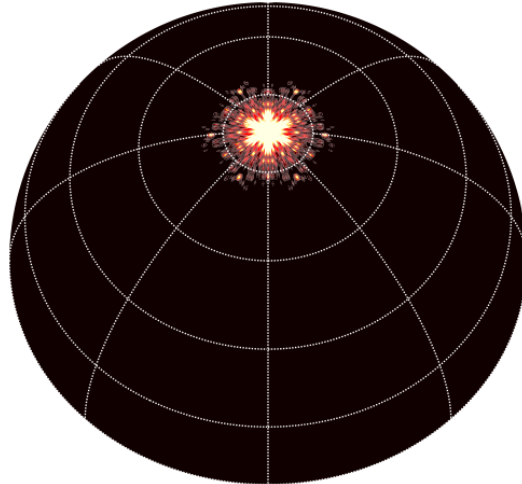

Figure 4.7: Simulation of collimator effect on the incidence direction of the gamma ray for a honeycomb grid consisting of hexagonal perforations with aspect ratio 50.
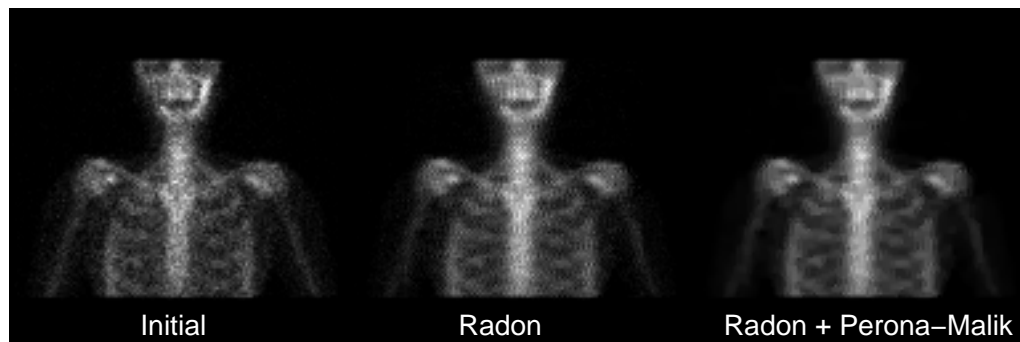

Figure 4.8: Radon processing (inverse Radon transform + Radon transform) and Perona–Malik non-linear filter.

Figure 4.8 shows a preliminary test of image improving using the standard Radon transform. Matlab library functions are used for the Radon transform (`radon`) and for the back-projection (`iradon`). The back-projection is computed from all the images, corresponding to a 360° rotation of the gamma camera. Then, the resulting 3D object (containing information of all the available images) is projected with the Radon transform to get new planar

images. The goal of this approach is to use all the available information to get a better image for a given angle. It can be interpreted as adding information from the events counted in other images to improve the current image. The processed image seems to have better quality than the initial one. For instance, a better definition of the ribs is observed. The non-linear Perona–Malik filter is also applied to improve the image further.

It is worth mentioning that this is a preliminary test. A proper definition of the Radon transform, and the corresponding back-projection, taking into account the absorption of the tissues has to be considered. The presence of the back ribs appearing in the front image evidences the negative effect of obviating the absorption of the tissues.

4.3.2 Modelling with absorption

However, if we are to take into account the effect of body absorption while considering the collimator ideal, the tool of choice will be the *attenuated Radon transform* [18]

$$\mathbf{R}_\mu g(\theta, s) = \int_{x \cdot \theta = s} \exp(-\mathbf{D}\mu(x, \theta^\perp)) g(x)\, dx;$$

where $\theta^\perp$ is the unit vector perpendicular to $\theta \in \mathbb{S}^1$ counterclockwise from θ and μ is the absorption per unit length at the point and $\mathbf{D}$ is the *divergent beam transform*:

$$\mathbf{D}f(a, \theta) = \int_0^\infty f(a + t\theta)\, dt.$$

The following cases should be considered:

- $\mu = \mu_0$ constant in a compact convex domain K containing the support of f;

- μ locally constant.

Attenuated Radon transform is a special case of *general Radon transform*

$$\mathbf{R}_\Phi(\theta, s) = \int_{x \cdot \theta = s} \Phi(x, \theta) f(x)\, dx,$$

for which conditions for inversion had been found. See e.g. [19, 20, 21, 22].

4.4   Concluding remarks

The two days meeting of the GEMT group opened and explored several possibilities for the improvement of gamma camera images. The preliminary results, reflected in this manuscript, encourage further effort in several research lines such as super-resolution algorithms, image overlapping and back-projected reconstruction using proper Radon transforms. Note that all these strategies take advantage of a particular property of gamma camera images: several low resolution images of the same object are available, and may be combined to get a higher resolution image.

Examples confirm that general purpose super-resolution algorithms are successful tools for image improving, with high availability, that can be easily applied to gamma camera images. However, simpler algorithms especially designed for gamma camera images could be easily developed. For instance, given the radius of rotation of the gamma camera, and the increment in the angle between consecutive images, a linear transformation could be applied to neighboring images in order to overlap them to the image of interest.

Bibliography

[1] http://en.wikipedia.org/wiki/Gamma_camera

[2] http://en.wikipedia.org/wiki/Tomographic_reconstruction

[3] http://www.owlnet.rice.edu/ elec539/Projects97/cult/report.html

[4] http://www.med.harvard.edu/JPNM/physics/didactics/basics.html

[5] http://www.medical.siemens.com

[6] P. Perona and J. Malik, *Scale-space and edge detection using anisotropic diffusion,* IEEE Transactions on Pattern Analysis and Machine Intelligence **12** (1990).

[7] Black et al., *Robust anisotropic diffusion,* Image Processing, 1998.

[8] Weickert et al., *Efficient and reliable schemes for nonlinear diffusion filtering,* Image Processing, 1998.

[9] http://www.math.ucla.edu/ gilboa/PDE-filt/diffusions.html

[10] http://www.photoacute.com

[11] http://en.wikipedia.org/wiki/Super-resolution

[12] M. Irani and S. Peleg, *Super resolution from image sequences*, ICPR **2** (June 1990), 115–120.

[13] G. T. Clement, J. Huttunen, and K. Hynynen, *Superresolution ultrasound imaging using back-projected reconstruction,* J. Acoustical Soc. Amer. **118**, no. 6 (2005), 3953–3960.

[14] Bo Zhang, J. M. Fadili and J.-L. Starck, *Wavelets, ridgelets, and curvelets for Poisson noise removal*, Image Processing, 2008.

[15] F. J. Anscombe, *The transformation of Poisson, binomial and negative-binomial data*, Biometrika **35** (1984), 246–254.

[16] D. L. Donoho, *Nonlinear wavelet methods for recovery of signals, densities and spectra from indirect and noisy data,* in Proc. Symp. Applied Mathematics: Different Perspectives on Wavelets **47** (1993), 173–205.

[17] E. Candès and D. L. Donoho, *Ridgelets: The key to high dimensional intermittency?* Philos. Trans. R. Soc. London A **357** (1999), 2495–2509.

[18] F. Natterer, *The Mathematics of Computerized Tomography,* Teubner, 1986.

[19] E. T. Quinto, *The invertibility of rotation invariant Radon transforms,* J. Math. Anal. Appl. **91** (1983), 510–522.

[20] J. Boman, *Uniqueness theorems for generalized Radon transforms,* Preprint, Dept. of Mathematics, Univ. of Stockholm.

[21] A. Markoe and E. T. Quinto, *An elementary proof of the local invertibility for generalized and attenuated Radon transforms*, SIAM J. Math. Anal. **16** (1985), 1114–1119.

[22] A. Hertle, *On the injectivity of the attenuated Radon transform,* Proc. Amer. Math. Soc. **92** (1984), 201–206.

Routing and dimensioning problems in transport networks

by Roc Alabern and Aureli Alabert

5.1   Introduction

We study here the problem of handling two or more constraints in the Resource Constrained Shortest Path Problem. We produce a counterexample showing that the Handler–Zang method, which is a good algorithm in the one-constraint case, cannot be extended in a straightforward manner to the general case, so that the complexity of the problem remains open.

In Section 2, we describe the problem and the corresponding extension of the Handler–Zang method. In Section 3, the immediate problems and questions raised by the method are exposed. Section 4 contains a paradigmatic very simple counterexample, where the pitfalls in which the algorithm may incur are clearly seen, therefore suggesting the need for further research in this direction. Section 5 summarises the open questions.

This report is based on contributions from the discussions held at the Grups d'Estudi, which included the following people: Roc Alabern, Aureli Alabert, Anna Bosch, Adrián Galdrán, Xavier Muñoz, Aleix Ruiz de Villa, Oriol Serra, Vaida Spokaite, Noèlia Viles.

5.2   Handling multiple (two) resource constraints

The problem we will study in the following lines is a generalisation of the Single Constraint Shortest Path Problem, for which an efficient algorithm (namely, the Handler–Zang method) exists. Specifically, the new *Constrained Optimization Problem* is the following:

Let $d_1\colon E \to \mathbb{R}_+$, $d_2\colon E \to \mathbb{R}_+$, be two different weighting functions, and $\Delta_1 \in \mathbb{R}_+$, $\Delta_2 \in \mathbb{R}_+$ two given constants. The cost function to be minimized

is $c\colon E \to \mathbb{R}_+$, and $\mathcal{P} \in 2^E$ denotes the set of feasible paths. We are then trying to solve the following minimization problem:

$$\min\left\{\sum_{e\in P} c(e) :\ P \in \mathcal{P},\ \sum_{e\in P} d_1(e) \le \Delta_1,\ \sum_{e\in P} d_2(e) \le \Delta_2\right\}.$$

Of course, this problem is harder than the one with a single constraint, but we can try to apply *Lagrangian relaxation* again. So, instead of minimizing c under the given constraints, we will try to minimize

$$\min\left\{\sum_{e\in P} c(e) + \lambda_1 d_1(e) + \lambda_2 d_2(e) :\ P \in \mathcal{P}\right\},$$

where λ_1, λ_2 are two fixed real numbers.

We will write this relaxed problem, for reasons that will be clear later, in the following way:

$$\min\left\{\sum_{e\in P} c(e) + \lambda_1(d_1(e) - \Delta_1) + \lambda_2(d_2(e) - \Delta_2) :\ P \in \mathcal{P}\right\}.$$

Choosing λ_1, λ_2 equal to zero we will find a solution that minimizes the cost. Choosing λ_1 large enough and λ_2 equal to zero we will find a solution that satisfies $\sum_{e\in P} d_1(e) \le \Delta_1$ if such a solution exists. On the other hand choosing λ_2 large enough and λ_1 equal to zero we will find a solution that satisfies $\sum_{e\in P} d_2(e) \le \Delta_2$ if it exists. After these observations a first question comes up easily: Do we have any control over how to find a solution that satisfies both constraints at the same time? Even in the case when feasible solutions exist, we cannot choose any easy value for λ_1, λ_2 to find just one feasible solution, whereas in the one-dimensional case, i.e., only one constraint, it was a straightforward task.

In the case of one constraint it has been important that the region

$$\left\{(x, y) \in \mathbb{R}^2 :\ y = \sum_{e\in P} c(e) + x(d(e) - \Delta)\right\}$$

where

$$P = \operatorname{argmin}\left\{\sum_{e\in P} c(e) + \lambda d(e) :\ P \in \mathcal{P}\right\}$$

is convex and has a unique maximum which splits $\mathbb{R}^+$ into two regions $(0, \alpha)$ and $[\alpha, \infty)$. The first region can be characterised by the values of λ for which $\operatorname{argmin}\left\{\sum_{e\in P} c(e) + \lambda d(e) :\ P \in \mathcal{P}\right\}$ is not feasible, and the second corresponds to those λ for which it is feasible.

We can rewrite them in a way that is useful for generalisation. Let λ_0 be some fixed positive value. If

$$P_0 = \operatorname{argmin}\left\{ \sum_{e \in P} c(e) + \lambda_0 d(e) : \ P \in \mathcal{P} \right\},$$

then P_0 is feasible if and only if $\sum_{e \in P_0} d(e) \leq \Delta$; in other words, the line $\sum_{e \in P_0} c(e) + \lambda(d(e) - \Delta)$ is non-decreasing with respect to λ. In the same way, P_0 is not feasible if and only if $\sum_{e \in P_0} d(e) > \Delta$, meaning that the line $\sum_{e \in P_0} c(e) + \lambda(d(e) - \Delta)$ is decreasing with respect to λ. Define

$$f_{\lambda_0}(\lambda) = \sum_{e \in P_0} c(e) + \lambda(d(e) - \Delta).$$

Then the solution that we can find for λ_0 is feasible if and only if $\frac{\partial}{\partial \lambda} f_{\lambda_0} \leq 0$.

In the case of two constraints, we have to define, for some fixed λ_0 and β_0,

$$P_0 = \operatorname{argmin}\left\{ \sum_{e \in P} c(e) + \lambda_0(d(e) - \Delta_1) + \beta_0(d(e) - \Delta_2) : \ P \in \mathcal{P} \right\},$$

and a corresponding function f_{λ_0, β_0} with two variables as

$$f_{\lambda_0, \beta_0}(\lambda, \beta) = \sum_{e \in P_0} c(e) + \lambda(d(e) - \Delta_1) + \beta(d(e) - \Delta_2).$$

With these notations, we find that a solution P_0 does not satisfy any of the two constraints if $\frac{\partial}{\partial \lambda} f_{\lambda_0, \beta_0} > 0$, $\frac{\partial}{\partial \beta} f_{\lambda_0, \beta_0} > 0$; it satisfies the first one, but not the second one if $\frac{\partial}{\partial \lambda} f_{\lambda_0, \beta_0} \leq 0$ and $\frac{\partial}{\partial \beta} f_{\lambda_0, \beta_0} > 0$; it satisfies the second one but not the first one if $\frac{\partial}{\partial \beta} f_{\lambda_0, \beta_0} \leq 0$, $\frac{\partial}{\partial \lambda} f_{\lambda_0, \beta_0} > 0$; and finally, it is feasible and satisfies both at the same time if $\frac{\partial}{\partial \lambda} f_{\lambda_0, \beta_0} \leq 0$, $\frac{\partial}{\partial \beta} f_{\lambda_0, \beta_0} \leq 0$. These four different cases define four different regions for the values (λ_0, β_0). Let us call them A_1, A_2, A_3, A_4 using the same order in which they have been presented.

In any case, the generalisation of the Handler–Zang algorithm must rely on using three different planes instead of two lines. Those three planes, which we will call $f_{\lambda_0^1, \beta_0^1}$, $f_{\lambda_0^2, \beta_0^2}$, $f_{\lambda_0^3, \beta_0^3}$, intersect in one point $(\lambda_0', \beta_0', f_{\lambda_0^1, \beta_0^1}(\lambda_0', \beta_0'))$. The minimization of the relaxed problem with values λ_0', β_0' will give us a new plane $f_{\lambda_0', \beta_0'}$ that could replace any one of the three old planes.

In Figure 5.1 we can see three points which are coloured in blue, yellow and green. Let us assume that they are the starting points (λ_0^i, β_0^i) for $i = 1, 2, 3$. Each of these points has an associated plane $f_{\lambda_0^i, \beta_0^i}$. The three planes intersect at one point, which is depicted in orange. And this point has also an associated plane. This new plane, passing through the red point
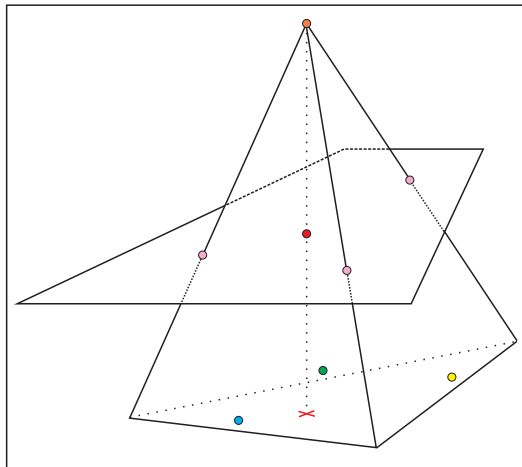
Figure 5.1: Handler–Zang method with two restrictions.

in the figure, will replace one of the old three planes. So the next point in this process will be one of the pink points. No matter which of the pink points is selected, it will lie below the orange point. Hence the sequence of those points is decreasing, thus ensuring the convergence of the process, since cycles will not be possible, and the number of planes is finite.

5.3   Problems and questions arising

Choosing three combinations of (λ_0, β_0) we will have three different initial planes. Choosing, for instance, both coordinates equal to zero, the first one only equal to zero, and the second one only equal to zero, respectively, will give us three different planes each one on the regions A_1, A_2, A_3, and none of them on A_4 if we are unlucky. So we cannot ensure a way to start the algorithm with one plane that belongs to the feasible domain.

Starting with those three planes we can find the intersection and the plane associated to that point. If the plane associated to that point is one of those three previous planes, the algorithm cannot go further. But if it is different, then we can replace one of them by this new plane, ensuring that still a convex surface is defined by taking the minimum of the chosen planes. In this way it is clear that there are no possible cycles and it can be proven that the sequence of points (intersections of three planes) gives rise to a decreasing sequence that converges in a finite number of steps. So the natural question is: Is the final solution given by this algorithm feasible? Unfortunately this is not always the case, as can be seen in some examples.

5.4 Some counterexamples to the convergence of the algorithm

With the following example we will show that the algorithm may converge to a non-feasible solution. Consider a graph with the values (c, d_1, d_2) of each arc as drawn in the Figure 5.2 and represented in tabular form in Figure 5.3.
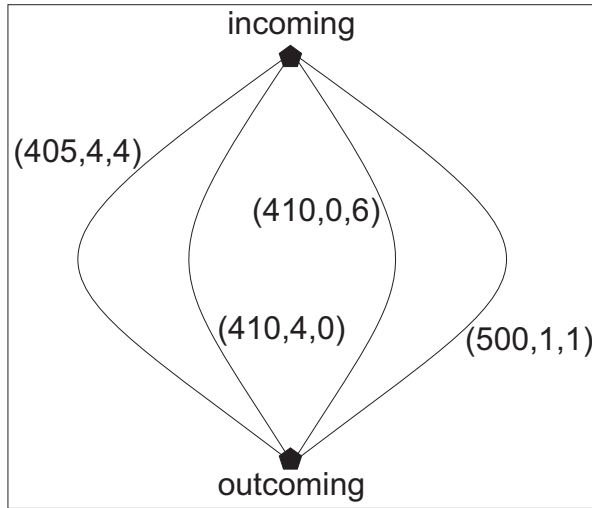

Figure 5.2: Counterexample to convergence with two constraints.

	c	d_1	d_2
P_1	405	4	4
P_2	410	4	0
P_3	410	0	6
P_4	500	1	1

Figure 5.3: The example of Figure 5.2 in tabular form.

There are only four different solutions, so that there are only four planes in this problem. Each plane has a different sign for their partial derivatives, so whenever a plane is minimum, it defines at that point what kind of region we have. Remember the four regions defined before: A_1, A_2, A_3, A_4. The four regions in this example are as shown in Figures 5.4(a), 5.4(b) and 5.4(c).

If we draw the graph of the minimum of those four planes in the example we will see something similar to Figure 5.4(d). And it is easy to see that with this example, or any other with slightly changed numbers, the algorithm may converge to a maximum which is non-feasible, or converge to some point that is not a maximum, and maybe even not feasible.
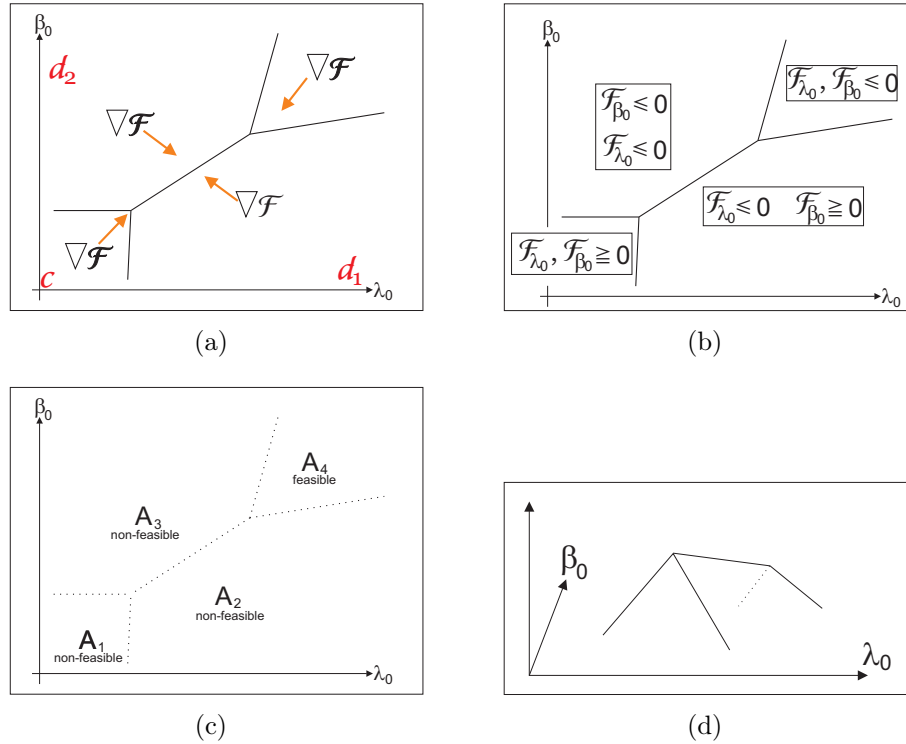
Figure 5.4: Four regions and graph for the example in Figure 5.2.

5.5   Open questions

The open questions are about the way to solve the problems that we have seen in the last example. The idea that came up in the discussion was to try to find a new good algorithm in some sense, and that, starting with a feasible plane, keeps at least one feasible plane at each step. The first problem that comes up with this idea is how to start with a feasible plane. To find a non-optimal feasible solution in the case of two constraints seems by itself a good problem to study. One possibility that was discussed was to develop some numerical algorithm to find an optimal solution for large values of both variables λ_0 and β_0. But how large one of them must be with respect to the other is not clear, and there seems to be no easy way to study this relation in order to find a feasible solution.

Ideal MHD equilibria of tokamak plasmas

by J. Blasco, P. Díez, M. Pellicer, and A. Rodríguez-Ferran

6.1 Introduction

A tokamak is a device that confines a plasma by means of a toroidal magnetic field (see `http://en.wikipedia.org/wiki/Tokamak`). The ultimate goal is the generation of energy through nuclear fusion. Several experimental tokamaks are currently in operation. A world-wide consortium (including, among others, the European Union, USA, Russia, Japan, and China) has launched the ITER initiative (Figure 6.1), "aimed at demonstrating that this energy source [fusion] can be used to produce electricity in a safe and environmentally benign way, with abundant fuel resources, to meet the needs of a growing world population" (see `http://www.iter.org`).
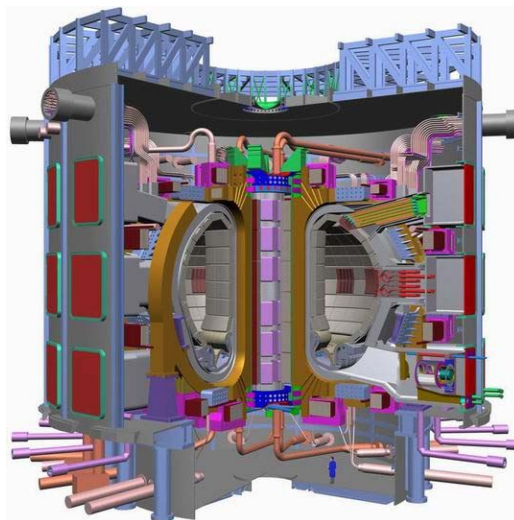
Figure 6.1: The ITER device (from `http://www.iter.org`).

In a tokamak, the confining magnetic field is generated by means of external coils (the term "tokamak", of Russian origin, stands for "toroidal chamber with magnetic coils"). For stable equilibria, azimuthal (i.e., rotational) symmetry is required.

6.1.1 General problem

A well established result in the field of magnetohydrodynamics (MHD) is that plasma equilibrium under such axi-symmetric conditions can be modelled by means of the so-called Grad–Shafranov equation (see, for instance, [2] and the references therein), which reads

$$L\Psi = j \quad \text{with} \quad j = \begin{cases} j(\Psi) & \text{in } \Omega_P \\ j_C & \text{in } \Omega_C \\ 0 & \text{in } \Omega_V \cup \Omega_A \end{cases} \tag{6.18}$$

In Equation (6.18), Ψ is the flux function, L is the 2D Laplacian-type operator in cylindrical coordinates,

$$L\Psi = -\frac{\partial}{\partial r}\left(\frac{1}{\mu_0 r}\frac{\partial \Psi}{\partial r}\right) - \frac{\partial}{\partial z}\left(\frac{1}{\mu_0 r}\frac{\partial \Psi}{\partial z}\right),$$

with μ_0 the magnetic permeability, and j is the electric current, which depends on the flux function in the plasma Ω_P, is constant in the coils Ω_C and is zero in Ω_V (void inside the chamber) and Ω_A (outer domain); see Figure 6.2. This partial differential equation is complemented with appropriate boundary conditions.

The Grad–Shafranov equation is nonlinear due to the flux-dependent current in the plasma. This nonlinearity, however, is mild and not a significant difficulty. The main issue is that plasma equilibrium is a free-boundary problem: the plasma domain Ω_P is unknown a priori, and part of the solution.

Our problem is hence the determination of the plasma domain Ω_P and the flux function Ψ that satisfy Equation (6.18), by means of numerical methods, such as the finite element method or more advanced discretisation techniques.

6.1.2 Specific problem

Based on the underlying physics, one can argue that the boundary $\partial\Omega_P$ of the plasma (i.e., the interface between Ω_P and Ω_V) is the minimal isoline of the flux function Ψ:

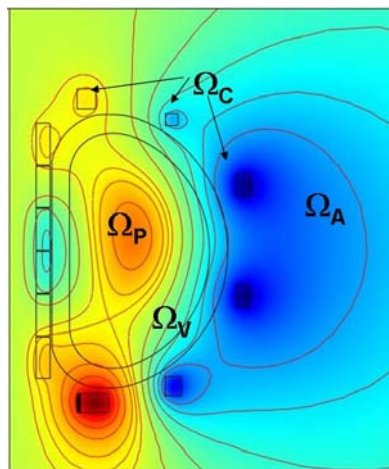$$\Psi(\partial\Omega_P) = \text{constant, minimal.}$$

Figure 6.2: Problem domain.

Two main types of such critical isolines are possible, characterised by two different critical points; see Figure 6.3. First, a simple closed isoline tangent to the chamber wall in a limiter point (i.e., a local minimum of Ψ). Second, a self-intersecting isoline at an X-point (i.e., a saddle point of Ψ). The resulting equilibrium configurations are termed "limited plasma" and "diverted plasma" respectively.
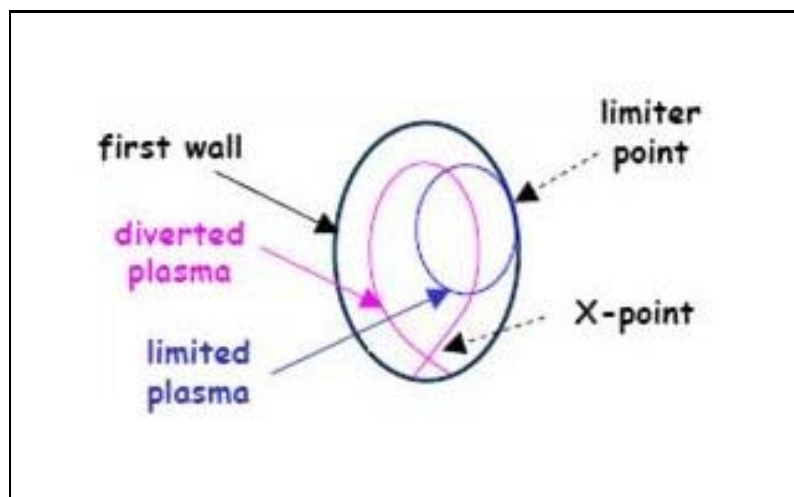

Figure 6.3: Critical isolines.

The envisaged approach is based on the iterative determination of the critical isoline: one solves the Grad–Shafranov equation with an initial guess for Ω_P, computes the critical isoline (i.e., a new guess for Ω_P) as a postprocess of the solution and repeats this process up to convergence.

In this context, our specific problem can be stated as follows:

> Given a finite element solution Ψ_H of the Grad–Shafranov equation, characterised by its nodal values, find the critical isoline.

In proposing an algorithm for this purpose, it is important to bear in mind that linear finite elements are the preferred option, so the Hessian matrix of the flux function Ψ is *not* readily available. Alternative procedures for the classification of critical points are required.

This report is based on contributions from the discussions held at GEMT, which included the following people: Marino Arroyo, Jordi Blasco, Josep M. Burgués, Pedro Díez, Xavier Cabré, Miquel Noguera, Marta Pellicer, Antonio Rodríguez-Ferran, Josep Sarrate, Joan Solà-Morales.

6.2   Classification of critical points via the index of a critical point

6.2.1   Motivation

In this section we propose an alternative and efficient way of facing the problem of classification of critical points of a vector field: the *index* of a *critical point*.

Essentially, we are thinking of the *winding number* of the gradient field along a closed, simple and small enough curve around the critical point. These notions can be found in Ch. V, § 36 of [1] or in Appendix II of [3], and we include them here for the self-containedness of the present report.

6.2.2   Informal definition

In order to get an intuition, let us first give an informal point of view of this approach. Let us consider a vector field $\overrightarrow{V} = \overrightarrow{V}(x, y)$ on the oriented Euclidean plane. Consider also a closed curve γ without any singular points of the vector field on it. Choose an initial point $P = (x_0, y_0)$ on the curve. Observe that we can associate to this point P the vector $\overrightarrow{V}(P) = \overrightarrow{V}(x_0, y_0)$ of the vector field. If we now move the point P along the curve, the associated vector $\overrightarrow{V}(P)$ will rotate. We can count the number of complete

rotations of this vector until P reaches its initial position again. We count the revolution as positive if $\overrightarrow{V}(P)$ rotates in an anticlockwise direction, and negative otherwise.

So, the total number of rotations of the vector on the curve is called the *winding number of $\overrightarrow{V}$ on γ* or the *index of γ on $\overrightarrow{V}$*.

Several remarks can be made. The first is noticing that the index does not change under small deformations of the curve, whenever the perturbed new curve does not pass through a critical point of the field. Also, it can be seen that a curve with a non-zero index encloses —at least— one singular point of the vector field in its interior.

We now define the *index of a critical point of a vector field* as the index of a sufficiently small and closed, positively oriented curve, that encloses this point but no other singularities of the vector field in its interior. Observe that the previous remarks imply that this definition does not depend on the chosen curve, if it is small enough. For instance, we can consider a small positively oriented circumference.

Essentially, this index *counts* the number of turns that a vector field is making around the singularity. In Figure 6.4 we can see two examples for the index of a critical point. We have represented a vector field that is the gradient of a function with a maximum (at the top) and one with a saddle point (at the bottom). We want to compute the index of each point, so we choose a closed curve (in green) that encloses only this singularity for each case. In both pictures we can see the rotation of a vector of the vector field on the green curve around the singularity. An anticlockwise rotation results in index 1 for the maximum, while a clockwise rotation gives index -1 for the saddle point.

If the curve (for instance the oriented boundary of a certain region) encloses several singularities, it also makes sense to compute the corresponding index. In this case, we have the following result.

Theorem 1. *Let us consider a domain Ω with $\gamma = \partial\Omega$ as the oriented curve giving the boundary. Suppose that the domain contains several singularities of the vector field in its interior (but not on the boundary). Then the index of the curve γ is equal to the sum of the indices of the singular points that are inside the domain Ω.*

The index of the region will be referred to as the *global index*. We can see an example of how this index is computed in Figure 6.5. In it, we have three maxima (index $+1$), two minima (index $+1$) and three saddle points (index -1). So, the global index of the domain Ω can computed as

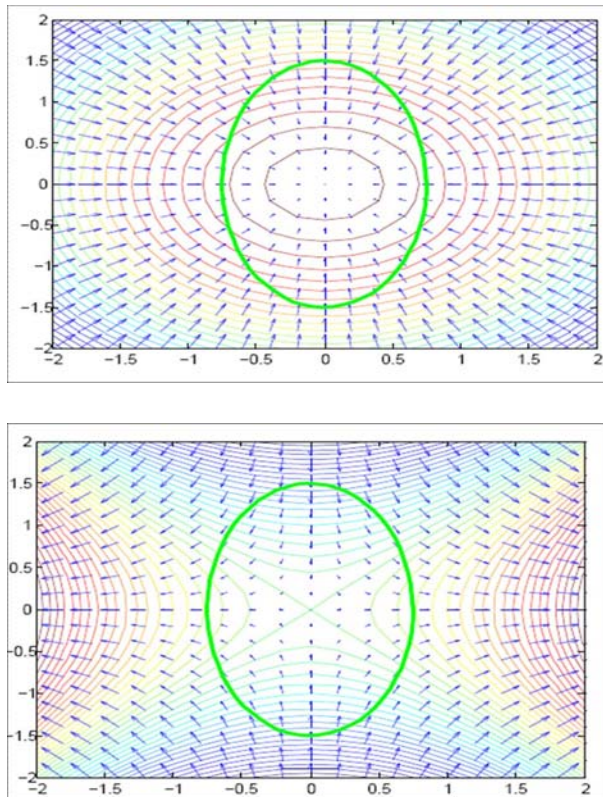$$\mathrm{Ind}_\Omega = 3 \cdot (+1) + 2 \cdot (+1) + 3 \cdot (-1) = 2.$$

Figure 6.4: Vector field of a maximum (top) and of a saddle point (bottom).

6.2.3   Connection with the tokamak plasma configuration

What is the use of the index in our problem, which is to find the critical isoline of Ψ that gives the physical configuration of the plasma inside the chamber? The first thing to notice is that, for the tokamak plasma, only a few configurations are possible. Essentially, knowing the kind of critical isoline is possible in our case when we know the number of maxima and saddle points of the vector field $\overrightarrow{V} = \nabla\Psi$ that are inside the chamber. This is possible using the global index of $\Omega_V \cup \Omega_P$ and, if necessary, the local index of each singularity.

For instance, as we have seen in Figure 6.3 of Section 1, we have two possible plasmas: a *diverted* one (the case of a maximum and a saddle point in its interior) and a *limited* one (only a maximum in the interior). Observe that the global index allows us to distinguish between these two possible configurations, as it is 0 for the diverted plasma and $+1$ for the limited one.
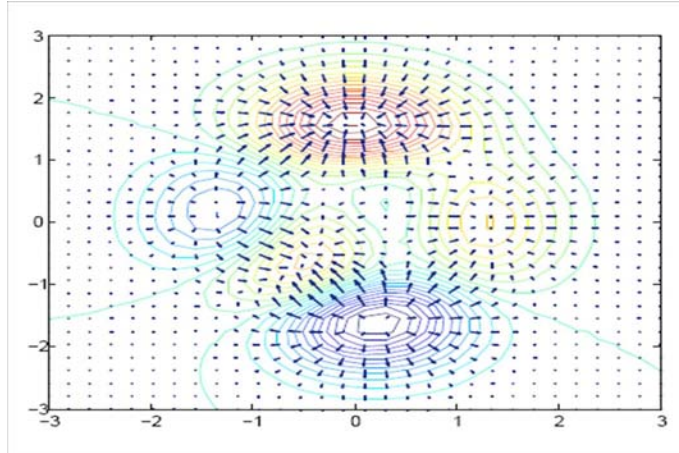
Figure 6.5: For many singularities the indices are added up. In the figure, the global index of the region is 2.

In Section 6.3 we propose an algorithm in which the index is used to distinguish between the most usual cases for the plasma configuration.

6.2.4   Formal definition in a continuous setup

A rigorous definition of the index can be given (again, see [1] and [3] as references). Let us consider a vector field $\overrightarrow{V} : \Omega \longrightarrow S^1$ with norm 1, given in polar coordinates. That is, $\overrightarrow{V}(x, y) = (r(x, y), \theta(x, y))$, where r stands for the radius and θ for the angle. Let us now consider a closed and oriented curve on this field, $\gamma : S^1 \longrightarrow \Omega$.

Definition 2. The *index of γ on $\overrightarrow{V}$* is

$$\mathrm{Ind}_\gamma = \frac{1}{2\pi} \oint_\gamma d\theta.$$

As $d\theta$ is the variation of the angle and we are integrating it along the curve, it is clear that the previous definition corresponds to the informal one for the index given above. This definition allows us to rigourously prove all the *informal* results given in this section. Actually, all this relates to the so-called *topological degree theory*.

6.2.5   Definition in a discrete setup

For the discrete case, the computation of the index in the case of a tokamak plasma turns out to be quite simple. As we will be using linear elements, the

gradient (that is, the vector field we are interested in) will be constant on each element. Let us call $\overrightarrow{V_i}$ the normalized gradient vector on the ith-element, that is,

$$\overrightarrow{V_i} = \frac{\overrightarrow{(\nabla\Psi)_i}}{\|\overrightarrow{(\nabla\Psi)_i}\|}.$$

Given a discrete and closed curve γ (for instance, the one in Figure 6.6) we can see that it passes through a finite number of elements (9 in this figure). So, we will have the corresponding $\overrightarrow{V_i}$ for each element. Let us call α_i the angle between $\overrightarrow{V_i}$ and $\overrightarrow{V_{i-1}}$. Then, the *(discrete) index of γ* on the gradient vector field is:

$$\mathrm{Ind}_\gamma = \frac{1}{2\pi}\sum_i \alpha_i.$$
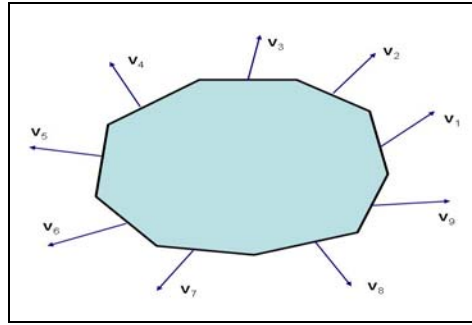
Figure 6.6: Discrete computation of the index.

The global index of a domain Γ is computed in the same way: we just have to consider the boundary of the domain as the curve γ.

6.3   Proposed algorithm

In this section we present the numerical algorithm proposed for the determination of the plasma region Ω_P. The algorithm aims at determining Ω_P from a given solution Ψ represented by a Finite Element nodal field. The overall idea of the proposed scheme consists in locating the singular points of Ψ through the vanishing of the gradient, computing the global and the local indices of $\nabla\Psi$ and classifying the state according to the number of singular points and the values of the indices.

6.3.1 Computation of the gradient and gradient recovery

Step 1 consists in computing $V = \nabla\Psi$ on every element; for linear triangular elements, V is piecewise constant. A gradient recovery technique has then to be employed to have V evaluated at the nodal points. This can be achieved by a weighted average of the elemental values on the elements to which the node belongs (weighted by the element area) or by an L^2 projection onto the Finite Element space.

6.3.2 Singular points detection

Step 2 consists in locating the singular points of Ψ. The norm of the gradient is computed on each node, and singular points are signalled when $||V|| <$ tol$\,||V||_{\max}$, where tol is a suitable tolerance. This yields the singular points P_i for $i = 1, \ldots, N_{\text{sing}}$.

6.3.3 Computation of the index

In Step 3 the global index $\text{Ind}_{\partial\Omega_V}$ of $\partial\Omega_V$ on $\nabla\Psi$ is computed by the discrete algorithm described in Subsection 6.2.5 applied to the nodal representation of $\partial\Omega_V$, the external boundary of the vacuum region Ω_V. The local indices $\text{Ind}_{\partial\Omega_i}$ around each singular point P_i are computed on patches Ω_i of elements surrounding P_i. The conservation of the property $\text{Ind}_{\partial\Omega_V} = \sum_{i=1}^{N_{\text{sing}}} \text{Ind}_{\partial\Omega_i}$ can be a good check for this phase.

6.3.4 Classification of cases

In Step 4 the state of the system is classified according to the number of singular points detected and the values of the global and local indices computed. The most frequent cases are reflected in the Table 6.1. The situation in each of these cases is the following:

1. NON-EXISTENCE OF CLOSED MAGNETIC SURFACES.

2. LIMITED PLASMA. There is a tangent isoline. Its isovalue is found as the maximum of Ψ on the boundary $\partial\Omega_V$, which is achieved at the limiter point.

3. NON-EXISTENCE OF CLOSED MAGNETIC SURFACES. An X-point exists, but none of the two branches of its isoline lies within the chamber, and there is no other critical point in the chamber.

4. A maximum P_1 and an X-point P_2 exist. The isovalue of interest is the value of Ψ in P_2. Open and closed branches of the isoline can be identified by curve tracking. Two subcases are possible, which are characterized by the number of changes of the sign of $(\Psi - \Psi(P_2))$ along the boundary $\partial\Omega_V$:

   4.1 One closed isoline inside the chamber. This is identified by two changes of sign. DIVERTED PLASMA.

   4.2 No closed isolines inside the chamber. This is identified by four changes of sign. NON-EXISTENCE OF CLOSED MAGNETIC SURFACES.

The remaining cases to be identified must be defined by the end-user according to his/her needs.

Case	N_{sing}	$\text{Ind}_{\partial\Omega_V}$	$\text{Ind}_{\partial\Omega_i}$
1	0	0	−
2	1	1	1
3	1	−1	−1
4	2	0	1, −1

Table 6.1: Classification of cases according to the number of singular points and the value of the indices.

6.4 Further considerations

6.4.1 Level set characterization of the critical isoline

In the previous sections, a methodology has been presented to compute the critical isoline from the Finite Element solution Ψ_H, characterized by its nodal values. The isoline is determined by finding the corresponding critical value of Ψ, say Ψ_c. Then, tracing the curve is a simple exercise of computational geometry. For instance, in a mesh of linear triangular elements (3-noded triangles), the iso-curve is a polygonal such that the part of it contained in every element is a straight segment. The elements crossed by the iso-curve are those such that the range of the nodal values comprises Ψ_c.

This characterization of the curve as the isoline of a contour function reminds us the well known Level Set (LS) concept [5]. An LS is an auxiliary function such that the curve to be characterized is precisely the zero iso-value. In this case, the function $\Psi - \Psi_c$ is an LS.

An LS function ϕ is defined such that its sign describes the geometrical domains using the following convention

$$\phi(\mathbf{x}, t) \begin{cases} > 0 & \text{for } \mathbf{x} \text{ in domain 1} \\ = 0 & \text{for } \mathbf{x} \text{ on the interface} \\ < 0 & \text{for } \mathbf{x} \text{ in domain 2} \end{cases} \tag{6.19}$$

where $\mathbf{x}$ stands for a point in the simulation domain and t is the time. The interface position is the set of points where the level set field vanishes (Figure 6.7).

In common LS practice, ϕ is taken as a signed distance to the interface. Far enough from the interface, ϕ is truncated by maximum and minimum cutoff values. The resulting level set function describes the position of the interface independently of the computational mesh. Note that, in this case, ϕ does not coincide with Ψ: ϕ is only a numerical artifact describing the isoline without any precise physical meaning. The only coincidence lies in the fact that both ϕ and $\Psi - \Psi_c$ vanish at the same points.

In practical implementations, ϕ is described (interpolated) with the finite element mesh, and therefore the resolution of the approximated interface depends on the quality of this mesh. Thus, the level set represents interfaces which do not necessarily coincide with the element edges. The same mesh can be used throughout the entire simulation to describe the interface.
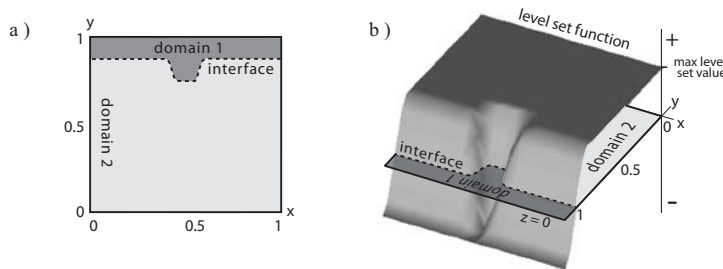
Figure 6.7: a) The two domains (materials) are associated with the sign of the level set function. The dotted line is the interface. b) Surface representation of the level set function.

Describing the curve using the LS ϕ rather than Ψ is not only a matter of taste. It may bring additional advantages related to the possibility of moving the estimated boundary of Ω_P along the iterative process in a more efficient manner.

6.4.2   X-FEM enrichment

The LS method is often used in combination with the so-called X-FEM (eXtended Finite Element Method); see [4, 6].

The LS approach allows the description of interphases that may cross the elements of the Finite Element mesh. Note that the phase changes are associated with discontinuities of the derivatives of the solution. Consequently, in the LS context, these discontinuities are expected to take place inside the elements. And this is something the standard finite elements cannot reproduce.

The X-FEM enriches the Finite Element discretization introducing additional interpolation functions with discontinuous derivatives across the interphase (the so-called ridge functions; see Figure 6.8 for illustration) via the Partition of the Unity Method.
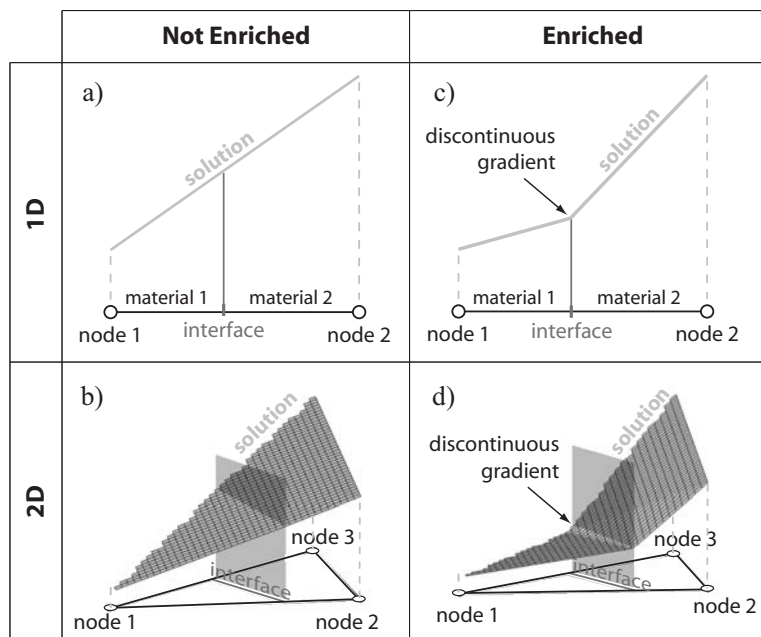
Figure 6.8: Illustration of the X-FEM enrichment both in 1D and 2D for linear elements.

Using the X-FEM would definitively increase the accuracy of the numerical approximation Ψ_H and would allow using much coarser meshes to obtain the same numerical resolution.

6.4.3 Analysis and possible alternative for the iterative scheme

The iterative method currently used consists in starting with an initial guess for Ω_P, say Ω_P^0 (or, equivalently, an initial guess for Ψ, say Ψ_H^0), then computing Ψ_H^1 solving Equation (6.18) numerically. The determination of the critical isoline corresponding to Ψ_H^1 yields also the new guess for Ω_P, say Ω_P^1, which in general does not coincide with Ω_P^0. The same idea is used to iterate from Ψ_H^k to Ψ_H^{k+1} for $k = 1, 2 \ldots$

This iterative scheme is astonishingly simple but, according to the end-users, the observed convergence behavior is fair enough. From a mathematical viewpoint this scheme may typically show an oscillatory behavior, jumping between two solutions (Ψ_H^k leads to Ψ_H^{k+1} and Ψ_H^{k+1} leads back to Ψ_H^k). It should be analyzed in detail in the context of the present physical problem, to understand why it is so efficient for this problem.

If a more sophisticated iterative scheme is used, based on finding the stationary/null point of some functional, it may be very useful describing the boundary of Ω_P by an LS ϕ. This would allow updating Ω_P via ϕ (from ϕ^k to ϕ^{k+1}) using a standard technique as the Hamilton–Jacobi transport equation. The input for this transport equation is the advancing front velocity that has to be given by the selected iterative scheme. For instance, this velocity can be defined as a function of the difference between the critical isovalue in the consecutive iterations, Ψ_c^{k-1} and Ψ_c^k.

Bibliography

[1] V. I. Arnold, *Ordinary Differential Equations*, Springer Textbook, Springer-Verlag, Berlin, 1992.

[2] P. J. Mc Carthy, *Analytical solutions to the Grad–Shafranov equation for tokamak equilibrium with dissimilar source functions*, Physics of Plasmas **6** (1999), 3554–3560.

[3] S. Lefschetz, *Differential Equations: Geometric Theory*, Dover Publications, Inc., New York, 1977.

[4] N. Moës, M. Cloirec, P. Cartraud, and J.-F. Remacle, *A computational approach to handle complex microstructure geometries*, Computer Methods in Applied Mechanics and Engineering **192** (2003), 3163–3177.

[5] J. A. Sethian, *Level Set Methods and Fast Marching Methods*, Cambridge University Press, 1999.

[6] S. Zlotnik, P. Díez, M. Fernández, and J. Vergés, *Numerical modelling of tectonic plates subduction using X–FEM*, Computer Methods in Applied Mechanics and Engineering **196** (2007), 4283–4293.

Participants

Aguadé, Santiago (H. Vall d'Hebron)

Alabern, Roc (DM, UAB)

Alabert Romero, Aureli (DM, UAB)

Alameda, Xavier (CEBSIAM, UPC)

Arroyo Balaguer, Marino (MA3, UPC)

Blasco Llorente, Jordi (MA1, UPC)

Bosch i Camós, Anna (DM, UAB)

Burgués, Josep M. (DM, UAB)

Cabré Vilagut, Xavier (MA1, UPC)

Cavinato, Mario (Fusion for Energy)

Díez Mejía, Pedro (MA3, UPC)

Fernández Méndez, Sonia (MA3, UPC)

Fossas Colet, Enric (IOC, UPC)

Galdrán, Adrián (Univ. València)

Gràcia Sabaté, Xavier (MA4, UPC)

Gutiérrez Barrientos, Pablo (DM, UAB)

Jüttner, Alpár (Ericsson-Hungary)

Martínez Sáez, Fernando (MA2, UPC)

Masdemont Soler, Josep (MA1, UPC)

Muñoz López, Xavier (MA4, UPC)

Noguera Batlle, Miquel (MA2, UPC)

Pellicer Sabadí, Marta (DIMA, UdG)

Portone, Alfredo (Fusion for Energy)

Roca Navarro, Xevi (MA3, UPC)

Rodríguez Ferran, Antonio (MA3, UPC)

Rodríguez González, Elena (DM, UAB)

Ruiz de Villa Robert, Aleix (DM, UAB)

Safont, Carmen (DM, UAB)

Saludes Closa, Jordi (MA2, UPC)

Sarrate Ramos, Josep (MA3, UPC)

Serra Albó, Oriol (MA4, UPC)

Solà-Morales Rubio, Joan (MA1, UPC)

Spokaite, Vaida (DM, UAB)

Vidal Seguí, Yolanda (MA3, UPC)

Viles Cuadros, Noèlia (DM, UAB)

Sebastià Xambó (MA2, UPC)

Contents

Participants at GEMT 2008