SAPIENZA
UNIVERSITÀ DI ROMA

MASTER DI II LIVELLO IN "CALCOLO SCIENTIFICO"

A.A. 2013-2014

High Intensity Focused Ultrasound (HIFU): computing tools for medical applications.

Candidata
Silvia Pozzi

Tutor Interno
Prof. Maurizio Falcone

Tutor Esterno (ISS)
Dr. Barbara Caccia

Contents

Introduction

For significant steps forward, it is crucial in medicine, as well as in other fields, to integrate specific experiences with the power of computing tools. In the particular case of this thesis work involved in the HIFU therapy of cancer, the support of a numerical simulation can improve the safety and effectiveness of the treatment and the comfort of the patients, and help to decrease the probability of side effects.

This work has been carried on at the *Istituto Superiore di Sanità* (ISS) which is the National Health Institute with the mission to guarantee the safety and the efficacy of the therapies in order to protect public health. At the same time ISS has the precise duty to promote public health and thus to favor the translation of new treatments, as the HIFU, to patients who need them.

HIFU stands for High Intensity Focused Ultrasound. It is a noninvasive treatment for either benign or malignant tumor lesions based on the power of a focused ultrasound beam to locally heat biological tissues over a necrotic level. The use of this kind of therapy can drastically reduce the time needed for the treatment to few hours, and there is no need for the patient to be hospitalized. It has also the advantage that can be applied for the ablation of tumors impossible to be remove surgically and to patients not suitable for the general anesthesia required for surgery. Thus it may lead to a sensible reduction of social and economic costs.

However, specific tissues laying on the trajectory of the beam, can make the HIFU application impossible or, at most, do not allow a complete ablation of the target lesion.

Hence, numerical simulation is an important tool to predict the effects

of the application of HIFU in order to reduce the probability of damages due to skin-burns or nerve injuries, *etc.*

This work concerns the use of High Intensity Focused Ultrasound guided by magnetic resonance imaging (MRgFUS) in cancer treatment, within a collaboration with the hospital *Policlinico Umberto I* of Rome, where uterine and prostate fibroid tumors are currently routine in the experimental environment, and bone metastasis have been recently introduced in the experimental treatment protocol.

This work covers several aspects of the therapy planning and the simulation required for the HIFU translation to the clinical application. The work includes a Graphic User Interface (GUI) created to allow the physicians to manage the simulation for a treatment plan. An already existent model has been then extended in order to simulate a more realistic situation. The ultimate goal of the numerical simulation will be to take into account all the media, artificial and biological, traversed by the ultrasounds, by reading the biological information from the scan taken before the treatment.

The results of the simulation are given in form of temperature distribution and thermal dose delivered into the target area. A validation of this work has been done by comparing those results with the ones provided by the original algorithm. Eventually experimental data were provided by two different clinical centers: *Policlinico Umberto I* (University "La Sapienza", Rome) and *IBFM CNR-LATO, Cefalù (Palermo)*[1].

This thesis is structured as follows: **Chapter 1** introduces HIFU and its applications; **Chapter 2** briefly recalls the equation of wave propagation, in one dimension; **Chapter 3** describes some of the physics behind HIFU; in **Chapter 4** the algorithm and the GUI are presented; **Chapter 5** shows the validation of the code with respect to the original script and the experimental data; finally **Chapter 6** reports the conclusions and the future prospects.

[1] IBFM stands for *Istituto di Bioimmagini e Fisica Molecolare*,
LATO stands for *LAboratorio di Tecnologie Oncologiche* at Ospedale San Raffaele Giglio, Cefalù (Palermo)

Chapter 1

HIFU

In this chapter there will be a brief review on how ultrasound therapy is used in medicine.

The interest in HIFU oncological application lies in the fact that the very high intensity of the ultrasound beam and the focusing allow to quickly heat small regions of tissues without damages for the surrounding tissues. It is a noninvasive treatment, it does not require hospitalization, and can be applied to unresectable lesions or to patients that cannot undergo surgery. HIFU is usually delivered under imaging guidance for a real-time treatment monitoring. The imaging guidance can be either magnetic resonance (MR) or diagnostic ultrasound (US).

1.1 Ultrasounds in Medicine

Ultrasounds are widely known for their use in medicine for imaging. However there is a relatively recent development in their use in the therapeutic field. Either at experimental stage or approved as standard clinical treatment, ultrasound is used to treat a wide range of pathologies. Exploiting the several phenomena associated, ultrasound is used in repairing injuries of the musculoskeletal system, in breaking down blood clots and stones in kidney, urethra, and gallbladder, as well as in tumor treatment with or without the association with chemo- or radiotherapy.

Regarding Italy, the periodic report produced by the Ministry of Health

to the Government [2] describes a quite wide distribution of HIFU equipments on the national territory with 29 active facilities at the date of the report, 2011. It also indicates HIFU as a very promising cancer treatment.

HIFU ablation is also known as focused ultrasound ablation, or focused ultrasound surgery (FUS). The imaging guidance can be both performed with ultrasound (USgFUS) and with magnetic resonance (MRgFUS).

MR imaging offers high anatomic resolution and temperature sensitivity [3, 4], it has proven capabilities to create temperature maps [5],but the temporal resolution is limited due to the slow image acquisition frame rate [6], and it is not suited for patients with metallic implants. USgFUS has not the possibility to create quantitative temperature maps during the treatment but many studies have demonstrated that US grey-scale change is reliable for monitoring the temperature response to HIFU treatment in real-time. While the temporal resolution is limited in MRgFUS, USgFUS guidance provides real-time imaging at a relatively low cost, although with a limited field of view, spatial resolution, and contrast resolution. In addition, it is the only solution when the location of the region to be treated makes it difficult for the patient to be positioned into the MR device because of the small size of its bore [7].

MRgFUS ablation in oncology is applied to both benign and malignant tumor lesions, and to the palliative therapy of bone pain due to metastasis, thus improving the quality of life for patients with advanced disease [8, 9]. Although it is successfully applied for the resection of tumors located in several anatomical districts, the main indication for MRgFUS is uterine fibroids [10, 11].

HIFU treatments are usually carried out in a single session, often as a day case procedure, with the patient either fully conscious, lightly sedated or under light general anesthesia. The application of HIFU is called *sonication*. It consists of several short pulses followed by brief intervals of cooling down. This sequence of pulses is then followed by a longer period of cooling down.

Even though it was believed that ultrasound energy could not penetrate the bone tissue enough to perform ablation, in a pilot study patients were successfully treated for osteosarcoma [12].

In order to undergo coagulative necrosis, and thus to be thermally ablated, the tissue needs to be largely heated, at temperatures of about $56°C$ [13].

The death of tumor cells is thought to be caused by the denaturation of cell proteins, induced by the heating [14]. In addition, in the region surrounding the target area there is an increase of the blood flow which facilitates the accumulation of chemotherapeutic agents. As consequence of the rise of blood flow, there is also increase in oxygenation that produces a larger sensibility of the tumor tissue to radiation [15]. This is the reason for the association of chemo- and radiotherapy with HIFU. A scheme of this is shown in Fig. 1.1.

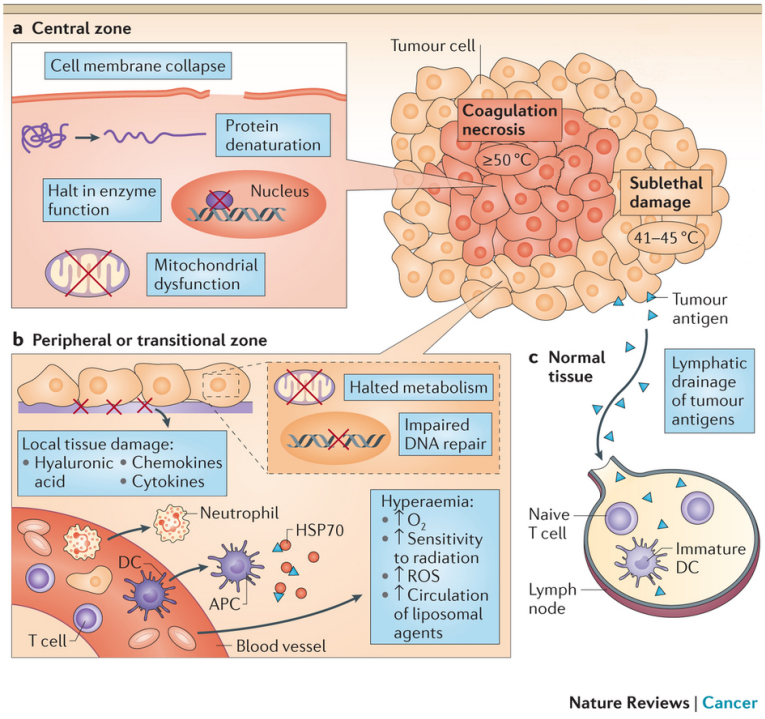

Figure 1.1: Zones in hyperthermic ablation [15].

The *thermal dose* concept has been proposed to quantify the relationship between treatment efficacy and the temperature of the target as a function of time. In Fig. 1.2 the relation between exposure time and temperature on tissues is depicted. The thermal dose is usually defined using an *equivalent*

minutes calculations [16] which was derived from an *in vitro* cell survival experiment during the application of temperatures of hyperthermia between $41°C$ and $50°C$ [17]. However, it has been shown that this model gives good estimations of the thermal lesion for the higher temperatures caused by HIFU. When the temperature changes with time it is used the thermal dose in the form of Cumulative Equivalent Minutes (CEM) of exposure at $43°C$ [16]

$$CEM_{43} = \int_0^{t_f} R^{(43-T(t))} dt, \quad R = \begin{cases} 0.25, & T \leq 43°C \\ 0.5, & T > 43°C \end{cases} \tag{1.1}$$

where t_f is the duration of the exposure. The commonly accepted threshold for protein denaturation is 240 CEM_{43} [18].

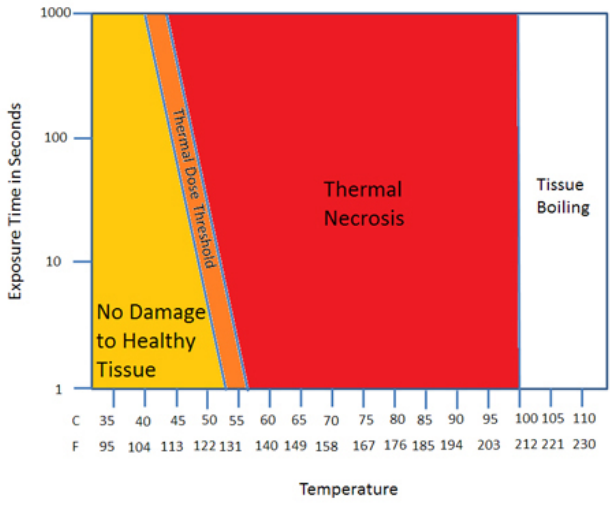

Figure 1.2: Relation between exposure time and temperature on tissues.

Chapter 2

The wave equation

This chapter is dedicated to recall the mathematical aspects behind HIFU.

High Intensity Focused Ultrasound therapy is based on the use of acoustic waves of frequencies laying in a non audible range for humans. HIFU waves are longitudinal pressure waves of finite amplitude and are subject to nonlinear effects. In order to solve the complete equation that models their propagation, the approximation of plane waves, that is wave propagating in only one direction, can be made.

Hence, here is illustrated the wave equation in one dimension and the derivation of the pressure wave equation.

2.1 The wave equation in one dimension

To understand the functioning of HIFU it can be started with the description of the wave equation modeling the propagation of waves in one dimension.

The description of a wave propagating along the z direction with velocity c is given by

$$u_{tt} - c^2 u_{zz} = 0 \tag{2.1}$$

where $u = u(z,t)\colon U \times \mathbb{R}_+ \to \mathbb{R}^n$, and U is an open set of $\mathbb{R}^n$. This wave equation is a simplified model for a vibrating string of infinite length or, like in this context, a plane wave propagating in the z direction. The function

$u(z,t)$ represents the displacement of the point z at time $t \geq 0$.

In order to completely determine the solution, some boundary conditions are needed. Thus the problem to be solved is

$$\begin{cases} u_{tt} - c^2 u_{zz} = 0 & \text{in } \mathbb{R} \times (0, \infty) \\ u = g & \text{on } \mathbb{R} \times \{t = 0\} \\ u_t = h & \text{on } \mathbb{R} \times \{t = 0\} \end{cases} \tag{2.2}$$

where g and h are given functions.

The equation $u_{tt} - c^2 u_{zz} = 0$ in (2.2) can be factorized as

$$(\partial_t + \partial_z)(\partial_t - \partial_z) u = 0 \tag{2.3}$$

where ∂_z stands for ∂/∂_z. Then the following variable transformation is suggested

$$\begin{cases} \xi = z + ct \\ \eta = z - ct\,, \end{cases} \tag{2.4}$$

with its inverse transformation given by

$$\begin{cases} z = \frac{\xi - \eta}{2} \\ ct = \frac{\xi - \eta}{2}\,. \end{cases} \tag{2.5}$$

For a function $v = v(z,t)$, defined as $\tilde{v}(\xi,\eta) := v(z(\xi,\eta), t(\xi,\eta))$ one has

$$\begin{aligned} \partial_\xi \tilde{v} &= \partial_z v \partial_\xi z + \partial_t v \partial_\xi t = \frac{1}{2}(\partial_t + \partial_z) v \\ \partial_\eta \tilde{v} &= \partial_z v \partial_\eta z + \partial_t v \partial_\eta t = -\frac{1}{2}(\partial_t - \partial_z) v \end{aligned} \tag{2.6}$$

so that $\partial_\xi \partial_\eta \tilde{u} = -\frac{1}{4}(\partial_t + \partial_z)(\partial_t - \partial_z) u = 0$. This means that $\partial_\eta \tilde{u}$ does not depend on ξ, and thus it is

$$\partial_\eta \tilde{u} = g(\eta) \tag{2.7}$$

for some function g. Integrating the previous equation over η leads to

$$\tilde{u}(\xi,\eta) = F(\xi) + \int_\eta g(s)\, ds =: F(\xi) + G(\eta) \tag{2.8}$$

for some function F. Going back to the initial variables z, t it is

$$u(z,t) = F(z+ct) + G(z-ct)\,. \tag{2.9}$$

In order to solve the problem (2.2), the functions F and G need to be determined by imposing the boundary conditions to the generic solution (2.9). Given $u_t(z,t) = F'(z+ct) - G'(z-ct)$, from the boundary conditions it follows that the functions F and G must satisfy

$$\begin{cases} F(z) + G(z) = g(z) \\ F'(z) - G'(z) = h(z)\,. \end{cases} \tag{2.10}$$

Deriving the first equation in Eq. (2.10) and adding it to the second, one obtains

$$F(z) = \frac{g(z)}{2} + \frac{1}{2}\int_0^z h(s)\,ds + k_1 \tag{2.11}$$

and thus, being $G(z) = g(z) - F(z)$

$$G(z) = \frac{g(z)}{2} - \frac{1}{2}\int_0^z h(s)\,ds + k_2 \tag{2.12}$$

where k_1 and k_2 are arbitrary constants. Substituting Eqs. (2.11) and (2.12) into Eq. (2.9), leads to

$$u(z,t) = \frac{1}{2}\left(g(z-ct) + g(z+ct)\right) + \frac{1}{2}\int_{z-ct}^{z+ct} h(y)\,dy \tag{2.13}$$

that represents the solution of the problem in Eq. (2.2), *i.e.* the solution of the wave equation with Cauchy boundary conditions in one dimension. Eq.(2.13) is then the sum of a progressive wave and a retrograde wave moving along the z direction with velocities c and $-c$, respectively.

It will be seen in the following section that it models the propagation of sound waves, that is pressure waves.

2.2 The sound wave equation

To build the equation of propagation for pressure waves, three important equations must be considered. They are the Euler equation, the continuity equation, and the state equation.

An ideal compressible fluid is moving within a region $E \subset \mathbb{R}^3$. Let $\rho(x,t)$ and $v(x,t)$ denote its density and velocity at $x \in E$ at the instant t. Each x can be regarded as being in motion along the trajectory $t \to x(t)$ with velocity $\dot{x}(t)$. Therefore, denoting by $v_i(x,t)$ the components of v along the x_i-axes, then

$$\dot{x}_i(t) = v_i(x(t),t)\,, \quad i = 1,2,3. \tag{2.14}$$

The acceleration has components

$$\ddot{x}_i = \frac{\partial v_i}{\partial t} + \frac{\partial v_i}{\partial x_j}\dot{x}_j = \frac{\partial v_i}{\partial t} + (\mathbf{v}\cdot\nabla)v_i \tag{2.15}$$

where ∇ denotes the gradient with respect to the space variables only. Consider a region $G_0 \subset E$ with boundary ∂G_0 of class C^1. Since G_0 is instantaneously in equilibrium, the balance of forces acting on G_0 must be zero.

The inertial forces due to acceleration, per unit of mass, are

$$\int_{G_0} \rho\,(\mathbf{v}_t + (\mathbf{v}\cdot\nabla)\mathbf{v})\,dx\,. \tag{2.16}$$

Let $p(x,t)$ be the pressure at $x \in E$ at time t. Then the internal forces due to pressure on G_0 are

$$\int_{\partial G_0} p\nu\,d\sigma\,, \tag{2.17}$$

where ν is the unit vector normal to the surface ∂G_0 and directed outward.

Therefore for the forces to be balanced it must be

$$\int_{G_0} \rho\,(\mathbf{v}_t + (\mathbf{v}\cdot\nabla)\mathbf{v})\,dx = -\int_{\partial G_0} p\nu\,d\sigma\,. \tag{2.18}$$

By Green's theorem

$$\int_{\partial G_0} p\nu\,d\sigma = \int_{G_0} \nabla p\,dx\,, \tag{2.19}$$

and thus by the arbitrariness of $G_0 \subset E$

$$\rho\left[\mathbf{v}_t + (\mathbf{v}\cdot\nabla)\, v\right] = \nabla p \quad \text{in } E\times\mathbb{R}\,. \tag{2.20}$$

Assume the following physical, modeling assumptions: (a) the fluid moves with small relative velocity and small time variations of density. Therefore second-order terms of the type $v_i v_{j,x_h}$ and $\rho_t v_i$ are negligible with respect to first order terms; (b) heat transfer is slower than pressure drops, *i.e.*, the process is adiabatic and $\rho = h(\rho)$ for some $h \in C^2(\mathbb{R})$.

Expanding the function h about the equilibrium pressure p_0, renormalized to be zero, gives

$$\rho = a_0 p + a_1 p^2 + \cdots\,. \tag{2.21}$$

Assume that the pressure is close to the equilibrium pressure, so that all terms of higher order are negligible compared to $a_0 p$. Including all these assumptions in Eq. (2.20) yields

$$\frac{\partial}{\partial t}(\rho\mathbf{v}) = -\nabla p \quad \text{in } E\times\mathbb{R}\,. \tag{2.22}$$

Taking the divergence of both sides leads to

$$\frac{\partial}{\partial t} div(\rho\mathbf{v}) = -\nabla^2 p \quad \text{in } E\times\mathbb{R}\,. \tag{2.23}$$

and from the continuity equation one has

$$div(\rho\mathbf{v}) = \rho_t = a_0 p_t\,. \tag{2.24}$$

Combining equations (2.22)-(2.24) gives the equation of the pressure in the propagation of sound waves in a fluid, in the form

$$\frac{\partial^2 p}{\partial t^2} - c^2\nabla^2 p = 0 \quad \text{in } E\times\mathbb{R}\,. \tag{2.25}$$

Eq. (2.25) is the equation of a pressure wave propagating with velocity c, in the three dimensional space. Taking into account only one dimension

leads to

$$\frac{\partial^2 p}{\partial t^2} - c^2 \frac{\partial^2 p}{\partial z^2} = 0 \tag{2.26}$$

that is the equation of the problem in Eq. (2.2) treated in the previous section, with solution given by Eq. (2.13). Considering only the general solution for the progressive wave, this has the form

$$p(z,t) = f(z - ct)\,. \tag{2.27}$$

A possible example of a progressive wave is a sinusoidal wave of amplitude $\bar{p}$

$$p(z,t) = \bar{p}\sin\left[k\left(z - ct\right) + \phi_0\right] = \bar{p}\sin\left(kz - \omega t + \phi_0\right) \tag{2.28}$$

where c is the wave velocity, ω is the angular frequency, $k = \omega/c$ is the wave number, and ϕ_0 is a constant phase.

In the model here considered, only progressive plane waves are taken into account.

Chapter 3

Physics of ultrasound

HIFU is particularly interesting in oncology for its power of heating small regions over the necrotic level. This feature is the result of the interaction of pressure waves with the medium. When the amplitude of the ultrasound is large, the pressure wave propagation is accompanied by nonlinear effects. While the intensities used in diagnostics are of the order of $100\,mW/cm^2$, HIFU intensities are of about $10\,kW/cm^2$.

For these reasons in this chapter it will be illustrated some of the physics behind the propagation of ultrasound.

3.1 Ultrasound wave

Sound waves are due to the propagation of oscillations, about their mean position, of the particles in a medium. They do not propagate in vacuum. Acoustic waves are longitudinal waves, *i.e.* the particles oscillate about their mean position in the direction of the wave propagation. On the contrary, transverse waves are produced by the oscillations of particles in a plane perpendicular to the direction of propagation, like sea waves. Longitudinal and transverse waves are schematically represented in Fig. 3.1. When a sound wave passes, even though it creates a perturbation in the strength of forces binding atoms and molecules, the medium can be thought of as a continuum. The reason is that its single constituents can be discerned only when the wavelength of the perturbation is of the same order of magnitude of

the interatomic or intermolecular distances, and this happens at extremely high frequencies, of the order of $100\,GHz$. Thus, the term *particle* can be generally intended as a volume of the medium, which is very small compared to the wavelength and very large compared to the interatomic and intermolecular distances.

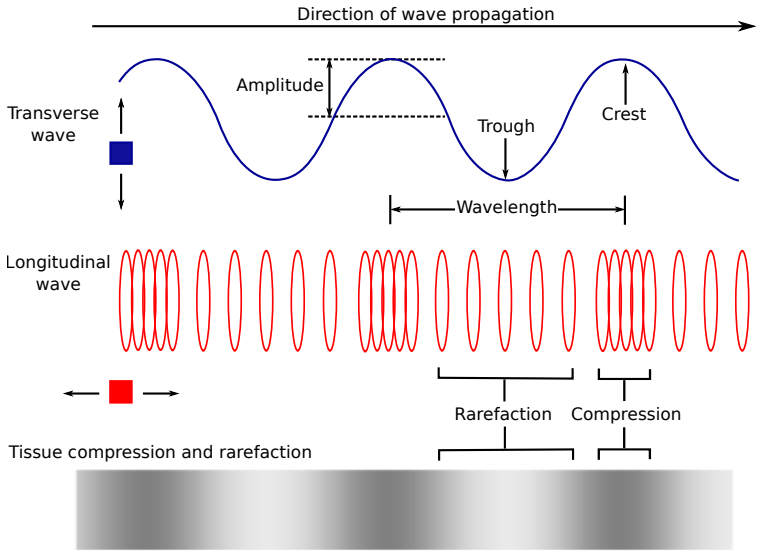

Figure 3.1: Longitudinal and transverse waves.

When an acoustic wave propagates, the particles oscillate about their mean position with a *particle velocity*, different from the wave propagation velocity, the *sound velocity*. The particle oscillations induce changes in density in the neighboring area, that in turn leads to a variation of pressure, positive (compression) and negative (rarefaction), with respect to the pressure at the equilibrium. This variation is the *acoustic pressure*, and it is what is perceived by the human ear, when the frequency of the wave falls into the audible range, Fig. 3.2, or by transducers.

From the Euler equation, Eq. (2.20), with the (a) and (b) approximations subsequently stated, in one dimension it is

$$\frac{\partial u_z}{\partial t} = -\frac{1}{\rho_0}\frac{\partial p}{\partial z} \tag{3.1}$$

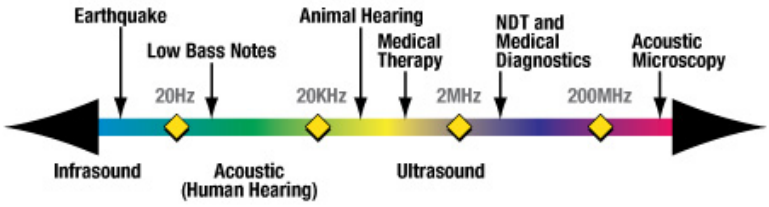

Figure 3.2: Sound spectrum.

where u_z is the z component of the particle velocity u.

Considering the pressure p as in Eq. (2.28)

$$p(z,t) = \bar{p}\sin\left(kz - \omega t + \phi_0\right) \tag{3.2}$$

the previous equation yields to

$$u_z = \frac{1}{\rho_0 c} p\,. \tag{3.3}$$

That means that the particle velocity u and the acoustic pressure p are in phase. The factor of proportionality between the two, $1/\rho_0 c$ is defined as the *characteristic acoustic impedance* Z_0, that is a quantity specific of the medium.

The most important effect of the interaction of ultrasound with a medium is the deposition of heat in a given volume. Thus the definition of intensity is needed.

The intensity of an acoustic wave is the amount of energy E that flows, in the time unit dt, through a unitary surface dS perpendicular to the direction of the wave propagation. That is

$$I = \frac{E}{dSdt} = \frac{Fdz}{dSdt} = pu_z \tag{3.4}$$

where f is the force acting on the surface dS and u_z is the z component of the particle velocity u. And thus

$$I = \bar{p}\bar{u}\sin^2\left(kz - \omega t + \phi_0\right) \tag{3.5}$$

is the intensity of the acoustic progressive wave propagating along z. What is usually of interest is the average of the intensity over a period T, and then

$$\langle I \rangle = \frac{1}{T}\bar{p}\bar{u}\int_0^T \sin^2(kz - \omega t + \phi_0) = \frac{1}{2}\bar{p}\bar{u} = \frac{\bar{p}^2}{2\rho_0 c} = \frac{\rho_0 c}{2}\bar{u}^2 \tag{3.6}$$

where Eq. (3.3) has been used.

When a propagating acoustic wave passes from a medium to another, it can be partly transmitted or reflected depending on the interface and the physical properties of the two media.

A pressure plane wave of amplitude $\bar{p}$ traveling in the direction of the positive z, from a medium with characteristic acoustic impedance $Z_{01} = \rho_1 c_1$ to a medium with characteristic acoustic impedance $Z_{02} = \rho_2 c_2$, and impinging perpendicularly on a plane surface separating the two media, will be partly transmitted and partly reflected.

The *reflection* and *transmission* coefficients are defined as the fraction of the incident amplitude that is reflected back and the fraction that passes through the surface

$$r = \frac{\bar{p}_r}{\bar{p}_i} \qquad \tau = \frac{\bar{p}_t}{\bar{p}_i} \tag{3.7}$$

where the subscripts i, r, and t indicate the incident, reflected, and transmitted quantity respectively, as shown in Fig. 3.3.

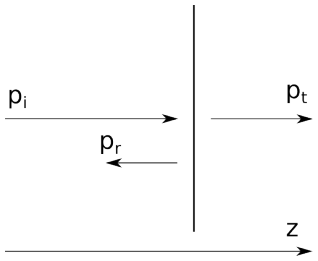

Figure 3.3: Incident, reflected, and transmitted pressure.

Given the relation in Eq. (3.3), the pressure can be expressed in terms of the velocity and thus, in order to have a net null force acting on the interface it must be

$$\frac{\bar{p}_i - \bar{p}_r}{Z_{01}} = \frac{\bar{p}_t}{Z_{02}} \tag{3.8}$$

where it has been taken into account that the reflected wave travels toward the negative z. The transmission and reflection coefficients of the pressure for a plane wave in terms of the acoustic impedance are then

$$r = \frac{Z_{02} - Z_{01}}{Z_{02} + Z_{01}} = \frac{1 - Z_{01}/Z_{02}}{1 + Z_{01}/Z_{02}} \tag{3.9}$$

$$\tau = \frac{2Z_{02}}{Z_{02} + Z_{01}} = \frac{2}{1 + Z_{01}/Z_{02}}\,. \tag{3.10}$$

Those coefficients have been explicitly mentioned because they are crucial for the simulation.

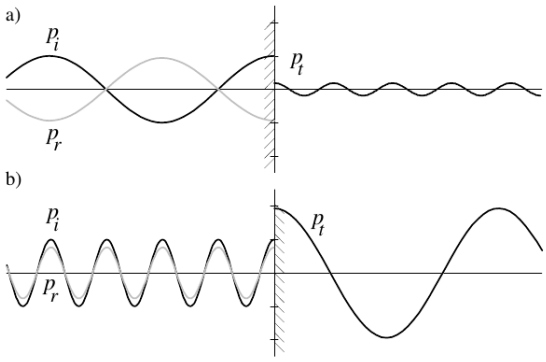

Figure 3.4: Incident, reflected, and transmitted pressure waves with: a) $r = 1$ and $\tau = 2$; b) $r = 1$ and $\tau = 1/2$

A plus sign of the reflection coefficient means that the reflected wave is in phase with the incident wave, while a minus sign means phase opposition. Thus, to a compression of the incident wave corresponds a rarefaction in the reflected one. On the other hand the transmission coefficient is always positive, meaning that at the interface the pressure of the transmitted wave is always in phase with the incident one. From (3.10) follows that for $Z_{02} > Z_{01}$ it is $\tau > 1$. In Fig. 3.4 are depicted pressure waves with $r = 1$ and $\tau = 2$, a), and $r = -1$ and $\tau = 0.5$, b). However the conservation law applies to energy, and both wave amplitude and wave velocity determine the energy

flow through a surface in the time unit, *i.e.* the intensity. Then, in terms of intensity there should be a net balance. Eq. (3.6) together with (3.9) leads to the coefficients of reflection and transmission for the intensity

$$\tau_I = \frac{I_t}{I_i} = \left(\frac{Z_{01}}{Z_{01}}\right)\tau^2 \qquad r_I = \frac{I_r}{I_i} = r^2 \tag{3.11}$$

that is

$$r = \left(\frac{Z_{02} - Z_{01}}{Z_{02} + Z_{01}}\right)^2 = \left(\frac{1 - Z_{01}/Z_{02}}{1 + Z_{01}/Z_{02}}\right)^2 \tag{3.12}$$

$$\tau = \frac{4Z_{01}Z_{02}}{(Z_{02} + Z_{01})^2} = \frac{4Z_{01}/Z_{02}}{(1 + Z_{01}/Z_{02})^2}\,. \tag{3.13}$$

from which it can be seen that in both cases $Z_{02} << Z_{01}$ and $Z_{02} >> Z_{01}$ the transmitted acoustic intensity (energy) tends toward zero.

At interfaces where there is little difference in acoustic properties, the transmission coefficient is close to unity. With the exception of fat, air, and bone, most tissues in the human body have acoustic properties similar to those of water. Aqueous media are therefore optimal for transmitting ultrasound energy from the transducer into the body, and reflections at tissue interfaces are generally weak.

3.2 Interaction of ultrasound with a medium

The sound wave that propagates through a medium progressively loses energy. Basically there are two causes: the absorption, that is the conversion of the mechanical energy carried by the wave into heat, and scattering, that determines a partial diffusion of the energy along directions different from the direction of propagation. In total, the density of energy lost in propagating through a medium of thickness dz is

$$dI = -2\alpha I(z)dz \tag{3.14}$$

which is

$$I(z) = I_0 e^{-2\alpha z} \tag{3.15}$$

where α is the *attenuation* coefficient and I_0 is the intensity at the origin $z = 0$.

For most tissues, the attenuation coefficient is related to the ultrasound frequency via a power law of the form

$$\alpha = \alpha_0 f^{\eta} \tag{3.16}$$

where α_0 and η are tissue-specific constants [19].

In addition, in the ultrasound region and especially at the frequencies used in medicine, there are non-negligible dissipative effects. These effects, specific for the considered medium, are mainly due to viscous and thermal effects.

The heat produced per unit of volume by the wave that loses its energy according to (3.14), neglecting the scattering, is

$$H = -\frac{dI}{dz} = 2\alpha I \tag{3.17}$$

where α is the attenuation coefficient introduced in Eq. (3.14). Assuming no heat loss, the local rise of temperature would be

$$\frac{\partial T}{\partial t} = \frac{H}{\rho_0 C_p} = \frac{2\alpha I}{\rho_0 C_p} \tag{3.18}$$

where C_p is the specific heat capacity at constant pressure. The above equation is certainly too naive to describe what happens in a real biological tissue. The local increase of temperature with respect to the environment can induce non-negligible heat transfers toward regions at a lower temperature. One of the responsibilities of the vascular system is to distribute heat transferring it from warmer to cooler regions, with the process of *perfusion*. Pennes' equation [20] describes the heat transfer in a biological medium with a heat source and a cooling process, taking into account for both thermal diffusion and perfusion

$$\frac{\partial T}{\partial t} = \frac{k}{\rho_0 C_p}\nabla^2 T - \frac{w}{\rho_0} T + \frac{H}{\rho_0 C_p} \tag{3.19}$$

where k is the thermal conductivity ($W\,m^{-1}\,K^{-1}$), C_p is again the specific heat capacity ($J\,kg^{-1}K^{-1}$), and w is the perfusion rate ($kg\,m^{-3}s^{-1}$).

Thus the Pennes' or bioheat transfer (BHT) equation, Eq. (3.19), will be solved in Section 4.1.2 in order to compute the temperature reached in the target region to induce the thermal ablation of the tumor tissues.

3.3 Propagation of wave with finite amplitude

When the amplitude of the acoustic wave is small enough, the particle velocity and the density changes are negligible with respect to the sound velocity and the density at the equilibrium. The main practical consequence is that the velocity of propagation is constant in the medium. However, when the wave amplitude (and thus the intensity) is large, the changes in particle velocity and density are not negligible and some non-linear effects need to be taken into account. These phenomena introduce mathematical complications that are bypassed by treating special cases, like a progressive plane wave.

Given the solution of the linear equation (2.25) for the acoustic pressure $p = p_0 f(z-ct)$ found in Eq. (2.27), it can be demonstrated that the solution for a nonlinear progressive plane wave is given by

$$p = p_0 f\left(z - (c + \beta u)t\right) \tag{3.20}$$

where β is the nonlinear coefficient. There are two physical effects determined by the term βu in (3.20): a *convective nonlinearity* and a nonlinearity due to the intrinsic properties of the medium, both of which determine different regions of the wave to propagate with different velocities. Namely, peaks move faster than troughs. The result is a distortion of the initial sinusoidal wave that increases with the distance traveled into the medium, as shown in Fig. 3.5. The distorted wave has a markedly different frequency content than the initial sine wave. The curve c) represents the maximum distortion, which corresponds to a quasi-continuous frequency spectrum. Beyond the distance where the maximum distortion occurs, the *shock distance*, the energy absorption from the medium progressively reduces the

amplitude of the wave until the initial wave form is reached. At this point the nonlinear distortion becomes negligible again.

The contributions of the two afore mentioned nonlinear effects result in a propagation velocity, in a liquid, given by

$$c = c_0 + \left(1 + \frac{B}{2A}\right) u \tag{3.21}$$

where A and B are the coefficients of the first and second order terms of the Taylor series expansion of the equation relating the pressure of the material to its density, and c_0 is the small-signal sound velocity, *i.e.* the velocity of a sound wave of infinitesimal amplitude. Introducing the nonlinear coefficient

$$\beta = 1 + \frac{B}{2A} \tag{3.22}$$

Eq. (3.21) becomes

$$c = c_0 + \beta u\,. \tag{3.23}$$

The shock distance (or *discontinuity length*), is given by [21]

$$L_{shock} = \frac{1}{\beta \varepsilon k} = \frac{c_0^2}{2\pi} \frac{1}{\beta u f} = \left(\frac{\rho_0 c_0^3}{2\pi}\right) \frac{1}{\beta p_0 f} \tag{3.24}$$

where $\varepsilon = u/c_0$ is the Mach number, $k = \omega/c_0 = 2\pi f$ is the wave number, f is the wave frequency, and p_0 is the amplitude of the acoustic pressure at the source.

The Khokhlov-Zabolotskaya-Kunetsov (KZK) equation [22, 23] models the propagation of a pressure wave of finite amplitude taking into account the nonlinear effects mentioned in Section 3.2.

A sound beam of finite amplitude radiated by an axisymmetric source in the plane $z = 0$ and propagating through a thermoviscous fluid, is described by the KZK equation in cylindrical coordinates as

$$\frac{\partial^2 p}{\partial z \partial t'} = \frac{c_0}{2}\left(\frac{\partial^2 p}{\partial r^2} + \frac{1}{r}\frac{\partial p}{\partial r}\right) + \frac{\delta}{2c_0^3}\frac{\partial^3 p}{\partial t'^3} + \frac{\beta}{2\rho_0 c_0^3}\frac{\partial^2 p^2}{\partial t'^2} \tag{3.25}$$

where p is the sound pressure, z and r the axial and transverse radial coor-

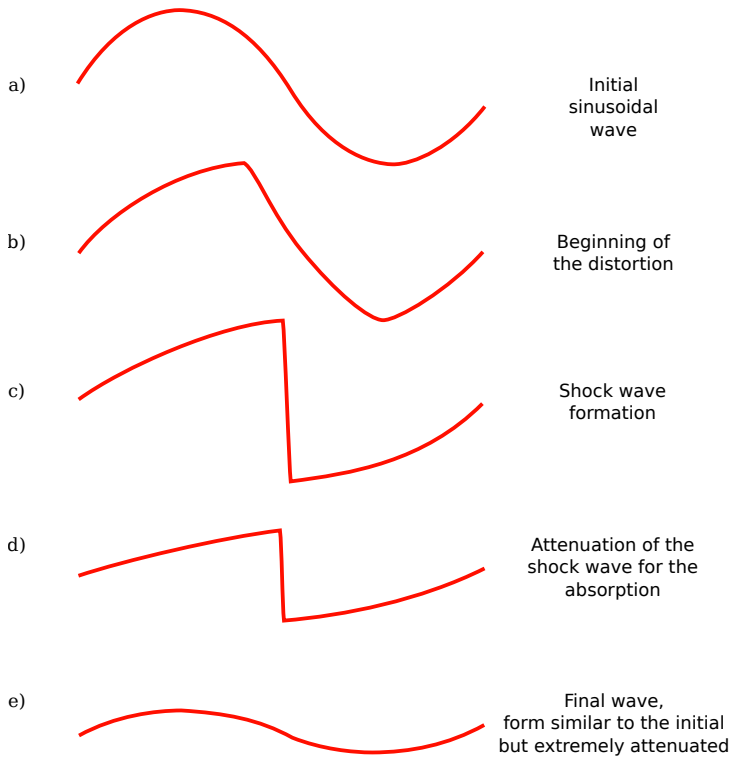

Figure 3.5: Shock wave formation.

dinates, $\partial^2/\partial r^2 + (1/r)\,\partial/\partial r = \nabla_\perp^2$ is the transverse Laplacian, $t' = t - z/c_0$ is the retarded time, c_0 is the small-signal sound velocity, δ is the diffusivity of sound in a thermoviscous fluid, β is the coefficient of nonlinearity, and ρ_0 is the equilibrium density of the fluid.

It is convenient to reason in terms of variations of pressure relative to the pressure at the equilibrium, and thus to introduce

$$p' = p - p_0\,. \tag{3.26}$$

Applying the following substitutions [24]

$$P = \frac{p'}{p_0} \quad \sigma = \frac{z}{d} \quad \rho = \frac{r}{a} \quad \tau = \omega_0 t' = 2\pi t' \tag{3.27}$$

and integrating in time, Eq. (3.25) becomes the dimensionless

$$\frac{\partial P}{\partial \sigma} = \frac{1}{4G} \int \nabla_{\perp}^2 P \, d\tau + A \frac{\partial^2 P}{\partial \tau^2} + NP \frac{\partial P}{\partial \tau} \tag{3.28}$$

where $A = \alpha_0 d$ is the absorption parameter, $N = d/\bar{z}$ is the nonlinearity parameter, and $G = z_0/d$ is the *linear focusing gain* at angular frequency ω_0. $\alpha_0 = \delta\omega_0^2/2c_0^3$ is the thermoviscous coefficient, $\bar{z} = \rho_0 c_0^3/\beta\omega_0 p_0$ is the plane wave shock distance at angular frequency $\omega_0 = 2\pi f$ defined in Eq. (3.24), $z_0 = \omega_0 a^2/2c_0$ is the *Rayleigh distance* at angular frequency ω_0, and a is the source radius. The easiest way to describe an acoustic field produced by a circular transducer is to consider it as a plane wave of the same diameter of the transducer in the near field and then as an expanding spherical wave in the far field, the Fresnel and the Fraunhofer regions in Fig. 3.6). The transition occurs at the Rayleigh distance [25].

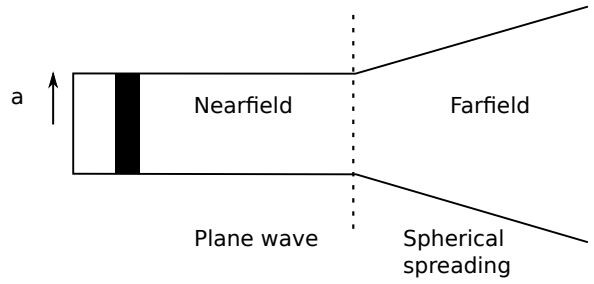

Figure 3.6: Schematic view of the axial section of the acoustic field produced by a circular transducer of diameter $2a$.

Thus the solution of Eq. (3.28) (KZK) is the purpose of the code that will be described in Chapter 4.

Chapter 4

GUI, Script & Co.

As already mentioned in the Introduction, the long term goal of the project I worked on is to provide the physicians experimenting HIFU therapy with a simulation tool able to estimate the heating effects of a planned treatment, in order to minimize possible damages to the patient's body.

As part of this project, the first step of this thesis work concerned the development of a graphic user interface (GUI) to allow a management of the simulation input parameters with no need to access the code.

Secondly, in order to better reproduce the real experimental systems, the original model [26,27] has been extended. Instead of just the water in which the transducer is immersed the extended model may take into account an arbitrary number of media and a thickness of *equivalent tissue*, as depicted in Fig. 4.1.

Finally, a validation of this multi-layered model has been performed by comparing the results obtained with the original model and with experimental data.

4.1 The model

The code I worked on is an open source package released by Dr. Soneson [26] which is described in [27]. It consists of a collection of scripts and functions based on Matlab. They simulate HIFU produced by a continuous wave transducer, and its heating effects.

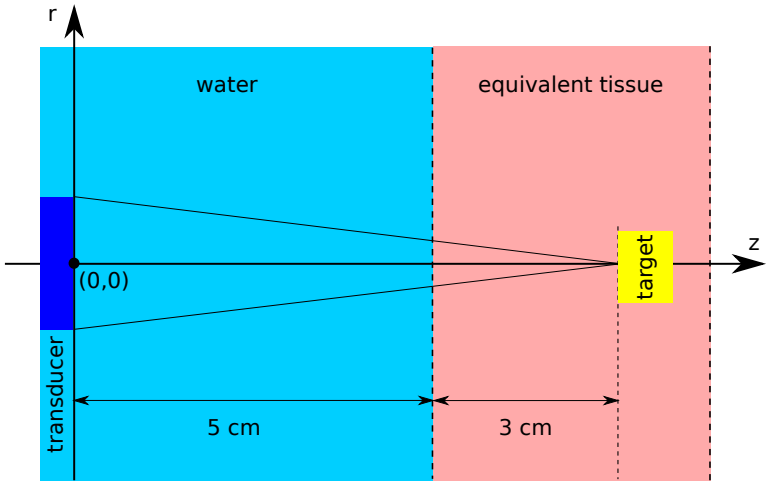

Figure 4.1: Sketch of the lateral view of the simulated original apparatus. From the left: in dark blue the annular transducer, in light blue the water in which it is immersed, in light red the equivalent tissue, in yellow the target. The distance between the transducer and the interface water/equivalent tissue is $5\,cm$ and the distance between the interface and the target is $3\,cm$. The origin of the axes lies in the center of the transducer and at its surface.

The two main scripts are devoted to the computation of a numerical solution for the KZK and the BHT equations. The output obtained by the first is the density of energy lost by the pressure wave and transferred to the traversed media, H. As mentioned in Chapter 3, this quantity is an input parameter for the solution of the BHT equation which models the changes of temperature with time during a sonication.

4.1.1 KZK

As reported in literature [29], the KZK equation

$$\frac{\partial P}{\partial \sigma} = \frac{1}{4G}\int \nabla_{\perp}^{2} P d\tau + A\frac{\partial^{2} P}{\partial \tau^{2}} + NP\frac{\partial P}{\partial \tau}\,. \tag{4.1}$$

is solved by discretizing $P(\rho, \sigma, \tau)$ both in space and time. The source condition is written in the form

$$p = p_0 F(r,t) \quad \text{at } z = 0\,, \tag{4.2}$$

where p_0 is the characteristic source pressure and $F(r,t)$ defines the time dependence as a function of radial position. The initial condition for the pressure of the sound wave at the transducer surface is an uniform distribution with an initial phase shift corresponding to a quadratic approximation of a spherically converging wave [30].

The main loop

The sound pressure is expanded as a Fourier series

$$P(\sigma, r, \tau) = \sum_{k=-K}^{K} \hat{p}_k(\sigma, r) e^{-ik\tau} \tag{4.3}$$

where $\hat{p}_k$ is the complex amplitude of the k-th harmonic, the maximum number of harmonics to be taken into account in the expansion, K, is an input parameter, so that the accuracy can be tuned according to the required speed of computation.

Substituting Eq. (4.3) into the linear part of Eq. (4.1) leads to a set of coupled equations

$$\frac{\partial \hat{p}_k}{\partial \sigma} + \frac{ic_0}{2} \nabla_\perp^2 \hat{p}_k + \alpha_k \hat{p}_k = 0 \,, \tag{4.4}$$

solved numerically. In Eq. (4.4) c_0 is the small-signal speed of sound, r is the radial coordinate, $\nabla_\perp^2 = \partial^2/\partial r^2 + (1/r)\,\partial/\partial r$ is the transverse Laplacian, σ is the axial coordinate introduced in Eq. (3.27), and $Re(\alpha_k)$ and $Im(\alpha_k)$ are the absorption and dispersion coefficients. The intensity of the computed solution of Eq. (4.1) with the dimensionless pressure P expanded in the Fourier series can be written as

$$I(r, z) = \frac{1}{2\rho_0 c_0} \sum_{k=1}^{K} |\hat{p}_k|^2 \tag{4.5}$$

and the corresponding density of energy lost, defined in Eq. (3.17) is

$$H(r, z) = \frac{1}{\rho_0 c_0} \sum_{k=1}^{K} |\hat{p}_k|^2 \tag{4.6}$$

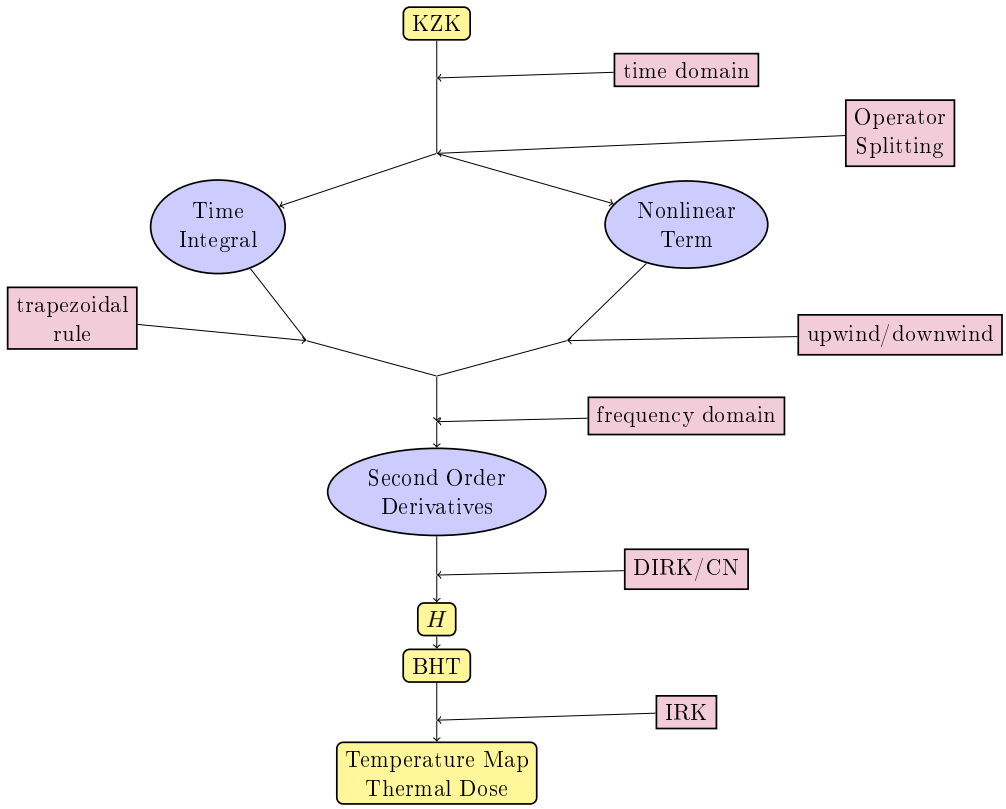

Figure 4.2: Flowchart of the algorithm.

Where:

$$\text{KZK:} \quad \frac{\partial P}{\partial \sigma} = \frac{1}{4G}\int \nabla_\perp^2 P\, d\tau + A\frac{\partial^2 P}{\partial \tau^2} + NP\frac{\partial P}{\partial \tau}$$

$$\text{Time Integral:} \quad \frac{\partial P}{\partial \sigma} = \frac{1}{4G}\int P\, d\tau$$

$$\text{Nonlinear Term:} \quad \frac{\partial P}{\partial \sigma} = \frac{\partial P}{\partial \sigma} = NP\frac{\partial P}{\partial \tau}$$

$$\text{Second Order Derivatives:} \quad \frac{\partial P}{\partial \sigma} = \nabla_\perp^2\left(\frac{1}{4G}\int P\, d\tau\right)$$

$$\text{BHT:} \quad \frac{\partial T}{\partial t} = \frac{k}{\rho_0 C_p}\nabla^2 T - \frac{w}{\rho_0}T + \frac{H}{\rho_0 C_p}$$

The flow chart reported in Fig. 4.2 schematically describes the phases of the actual computation.

Assuming that the integration steps are sufficiently small, the linear and

nonlinear effects are decoupled and thus they can be solved with different and more appropriate techniques (operator splitting [29]).

The computational grid is defined on the base of the focal length of the system, the radius of the transducer, and the maximum of the linear gain G of the materials, mentioned at the end of Section 3.3. The number of axial nodes in each material is in turn given by the ratio of the layer depth and the focal length times the step size.

For each axial node, a loop over the radial nodes is entered where the pressure P is expressed as a Fourier expansion in the time domain with the inverse Fast Fourier Transform. Then both the integral over time

$$\frac{\partial P}{\partial \sigma} = \frac{1}{4G} \int \nabla_{\perp}^2 P d\tau \tag{4.7}$$

and the nonlinear term

$$\frac{\partial P}{\partial \sigma} = NP \frac{\partial P}{\partial \tau} \tag{4.8}$$

are computed. The term in Eq. (4.7), exploiting that space and time are disentangled in the Fourier expansion, is first integrated over time and afterward the Laplacian is computed.

The integration is performed with the trapezoidal method. The nonlinear term in Eq. (4.8), being a perturbation with respect to the linear terms, is computed only when the amplitude of the solution is not negligibly small. In this case the amplitude itself is used to determine the number of steps required to respect the stability condition and keep the upwind/downwind method stable.

The nonlinear term can be seen as an inviscid mono dimensional Burger's equation in which the spatial and temporal partial derivatives are interchanged

$$u_t + u u_x = 0 \quad \rightarrow \quad u u_t + u_x = 0\,. \tag{4.9}$$

The finite difference methods used to solve this equation are based on the idea of writing it in the form

$$u_t + \partial_x \left(\frac{u^2}{2} \right) = 0 \tag{4.10}$$

and so the upwind/downwind scheme at the first order can be written as

$$u_j^{n+1} = u_j^n - \frac{\Delta t}{\Delta x}\frac{1}{2}\left((u^2)_{j+1}^n - (u^2)_j^n\right) \tag{4.11}$$

$$u_j^{n+1} = u_j^n - \frac{\Delta t}{\Delta x}\frac{1}{2}\left((u^2)_j^n - (u^2)_{j-1}^n\right) \tag{4.12}$$

and the stability condition is given by

$$\left|\frac{\Delta t}{\Delta x}u_{max}\right| \leq 1\,. \tag{4.13}$$

[31].

At this point the solution is transformed back to the frequency domain, the loop over the radial nodes is over, and then the actual computation at the current axial node is performed. According to [29], two different methods are used to compute the Laplacian in Eq. (4.7) for the region near the transducer surface and beyond it. The reason is that in the near field, because of diffraction effects, the solution is rapidly oscillating. It is then used a second order diagonally implicit Runge-Kutta method (DIRK) [32] that allows larger integration steps with respect to the backward-Euler scheme traditionally used. In this way a reasonable speed in computation is allowed while preserving the stability. Beyond the near field the Crank-Nicolson (CN) method is used. Consistently, the same two methods, DIRK and CN, are used to solve the second order time derivatives in the absorption term.

In a temporary matrix the quantity is stored

$$\sum_{k=0}^{K} Re(\alpha_k)\,|\hat{p}_k|^2 \tag{4.14}$$

where $\hat{p}_k$ are the coefficients of the Fourier expansion in Eq. (4.3). Thus Eq. (4.14) is proportional to the density of energy lost reported in Eq. (4.6) through the normalization factor $1/\rho_0 c_0$, for each axial node reached in the computation. Its rows and columns correspond to the radial and axial nodes, respectively.

The above description refers to the calculation that occurs at each axial node. This calculation is then repeated until the last axial node, *i.e.* the

whole focal length, is reached.

The original code models an annular transducer immersed in water followed by a phantom gel, as sketched in Fig. 4.1. Thus during the computation of the solution on the axial nodes, the interface water/gel is encountered. As mentioned in Section 3.1 only the fraction of amplitude transmitted has to be taken into account in the following computation. The main loop, in this case of two layers, is divided into two loops accounting for the amount of axial nodes contained in the layer of water and the amount of axial nodes contained in the phantom volume.

Reminding that the KZK equation, Eq. (3.28) for which the solution has just been numerically computed, is dimensionless

$$P = \frac{p}{p_0}\,, \tag{4.15}$$

the calculated pressure has to be normalized for the value of the pressure at the first axial node.

In order to compute the density of energy lost, H, Eq. (4.14) needs to be multiplied by the material-dependent parameters $1/\rho_0 c_0$. However, from Eq. (4.15), the additional factor p_0^2 is needed.

4.1.2 BHT

The time evolution of the temperature is modeled by the Pennes' bioheat transfer equation (BHT), as mentioned in Section 3.2,

$$\frac{\partial T}{\partial t} = \frac{k}{\rho_0 C_p}\nabla^2 T - \frac{w}{\rho_0}T + \frac{H}{\rho_0 C_p} \tag{4.16}$$

where k is the thermal conductivity ($W\,m^{-1}\,K^{-1}$) C_p is the specific heat capacity ($J\,kg^{-1}K^{-1}$), and w is the perfusion rate ($kg\,m^{-3}s^{-1}$). The source term is then the output of the previous script, H, that is the density of energy lost by the pressure wave.

The BHT equation is solved by a second order discretization with Dirichlet boundary conditions, and the scheme used to evolve the solution in time is a second-order implicit Runge-Kutta method [27].

The time step used in the scheme is defined on the basis of the sonication protocol adopted by the user. A single sonication is actually made of several short pulses followed by brief intervals of cooling down. This sequence of pulses is then followed by several seconds of cooling down. Generally, a check of the ablation resulting from the sonication is made with the imaging system adopted.

The output of this script is then a curve of temperature *vs* time, a temperature map in the focal region, and the thermal dose in the focal region.

4.2 The extension of the model

The goal of this phase of the work was the extension of the original model [26, 27] in order to make a step further toward the simulation of a progressively more realistic model.

After a sufficient comprehension of the problem and of the code at hand, it was possible to decide how to extend the model.

The first step was the introduction of an arbitrary number of layers of different materials that the pressure wave needs to traverse before reaching the target area. However, it was chosen a safe approach preserving the structure of the code. In order to repeat the same computation several times, the linearity of the code was broken and parts of it were set into external functions that can be called at the proper time. In particular, the main loop described above in Section 4.1.1 needs to be globally repeated for a number of times equal to the number of layers in which the focal length is divided. The total number of axial nodes is calculated on the base of the maximum value of the linear amplitude gain parameter G of all the materials composing the layers, like in the original model. In addition, the fraction of nodes laying in the single layer depends on the fraction of the focal length represented by the thickness of the layer itself.

As already mentioned, the pressure amplitude that is the amplitude of the computed solution in a dimensionless form, transmitted at an interface is related to the amplitude of the impinging wave through the transmission

coefficient defined in Eq. (3.7) and expressed in Eq. (3.10) as function of the acoustic impedances of the two media forming the interface. Thus the solution computed at the last node of a material must be multiplied by the corresponding transmission coefficient before being used as starting point for the loop over the axial nodes of the next layer.

As a consequence of the segmentation of the thickness of water into several layers, in renormalizing the temporary matrix of Eq. (4.14) one has to take into account the different values of the parameters for the different materials, namely the density ρ and the small-signal sound velocity c_0.

The BHT script was quite simple to extend because it was sufficient to divide the columns of the matrix H corresponding to each layer for the appropriate constants ρ_0 and C_p.

4.3 GUI

The GUI has been developed with GUIDE (Graphic User Interface Development Environment) Matlab tool. Its purpose is to easily change the simulation inputs according to the apparatus at hand.

In order to give an idea of its functioning, a picture of the GUI is shown in Fig. 4.3 with several phases of the simulation. To make this statement even more clear, its sequence is here described.

At first, a magnetic resonance (MR) scan of the patient is loaded to compare the output thermal dose with the input scan (this function will be later implemented). Referring to Fig. 4.3, the system menu icon at the top of the window allows to load the desired image, which is shown in the bottom right corner of the window. In principle this does not need to be done as first step, but it has been written in this order keeping in mind the long term goal of the project: the use of the patient's scan to automatically get information about the amount and type of layers the HIFU beam will pass through to reach the target region.

The input parameters are entered in the dedicated frames, at the top left, divided in parameters related to experimental geometry (*i.e.* the focal distance, and the inner and outer radii of the transducer) and in parameters

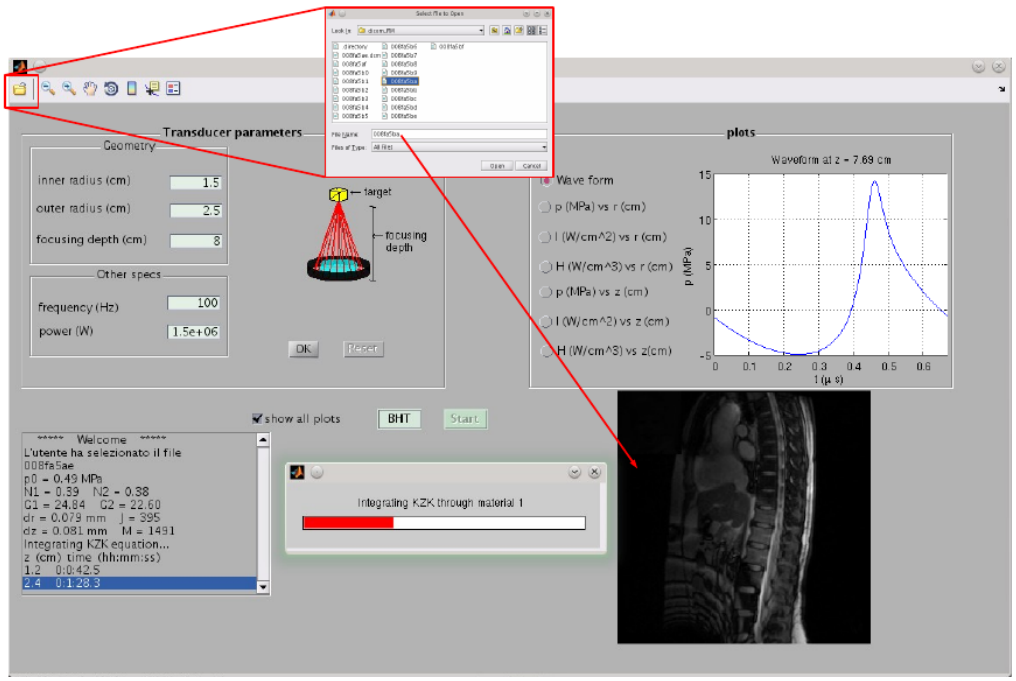

Figure 4.3: Snapshot of the GUI with several phases of the simulation at the same time. In red: the system menu to choose the MR image, shown in the bottom right corner. Top left: the frame *Transducer parameters* with the input parameters related to the sound wave source: the inner and outer radii, and the focal depth (all in *cm*), with the corresponding sketch, and the power and frequency parameters. Bottom left: a scroll box for the printout messages. Middle bottom: a progress bar - in a separate window – shows the percentage of computation performed, for each layer. Bottom right: the loaded MR image. Top right: a frame containing the output plots. With toggle buttons one can choose the plot to be visualized and toggle among all the others. Center: check-box and buttons for other options.

regarding the transducer itself (*i.e.* the power applied and the frequency of the produced sound wave). As in the future more geometries of transducers might be introduced, a double check has been created by showing a scaled sketch of the geometry of the entered parameters. This sketch is visualized by clicking on the OK button in the transducer frame. Clearly the button RESET resets all the filled fields.

The START button is bond to the main script that manages the calls to the KZK and BHT calculation. The BHT button sets a flag that is read in the just mentioned main script, and the BHT simulation is called.

Ticking the check-box enables the visualization of all the output plots

in the plot frame. It is possible to toggle from one to another by using the appropriate toggle button.

The progress of the calculation is shown in the progress bar. This has been placed in an external window in order to attract more attention from the user since the main window occupies the whole screen. The length of the progress bar increases together with the fraction of axial nodes where the computation just occurred, for each layer.

All the printout messages are progressively stored in an array of strings whose elements are shown at the same time. The choice of this kind of variable was dictated by the importance of keeping track of those messages. At the moment the messages shown are the size of the axial and radial node steps in cm, the linear gain and nonlinear coefficients, and the MR image file name. In addition, during the computation of the KZK equation, the time needed to advance of a certain spatial distance toward the target is visualized. Afterward, it is shown the time needed to advance in the computation of the solution of the BHT equation.

Chapter 5

Validation

A first step in validating the extended multi-layer model here developed, is a comparison of data with the output of the original algorithm.

Although not yet completed, a comparison between the multi-layer model and experimental data is also carried on.

5.1 Multilayer vs Original

In order to validate the changes with respect to the original algorithm, a comparison of data is made between the two models. The comparison is both qualitative and quantitative.

For a qualitative comparison the plots of relevant physical quantities are here shown, coupled between the original (left) and the multi-layer model (right). The plots are obtained under the following conditions:

original model: one layer of water $5\,cm$ thick; then a layer of gel mimicking an *equivalent tissue* $3\,cm$ thick. The total number of axial nodes is 1491, that is considered a good compromise between computational speed and accuracy.

multi-layer model: three layers of water of thickness of 2, 2, and $1\,cm$, for a total of $5\,cm$; then a layer of gel mimicking an *equivalent tissue* $3\,cm$ thick. The total number of axial nodes is the same, 1491, to be consistent and carry on the comparison.

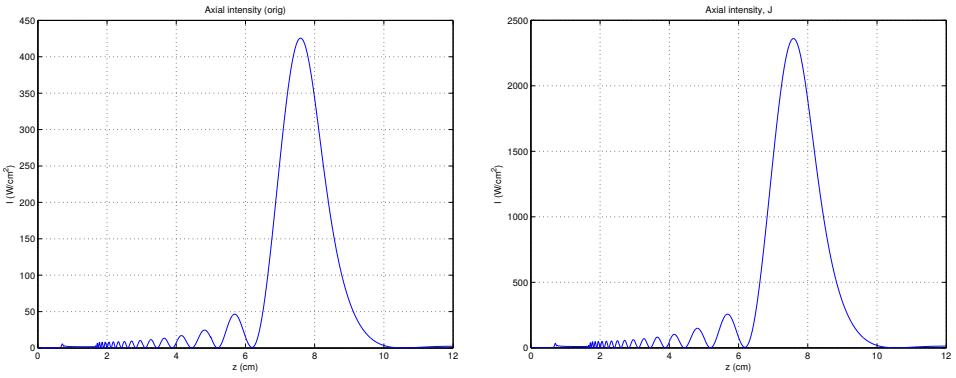

Figure 5.1: Comparison of the density of energy lost by the wave in the time unit, in the original (left) and the extended (right) models.

- Fig. 5.1 shows the plot corresponding to the density of energy lost by the traveling wave, in the time unit.

  1. the peak is located about the focal distance of $8\,cm$, that means inside the phantom object, $3\,cm$ downstream the interface water/gel;
  2. for the acoustic waves water is transparent, as the null plateau up to $5\,cm$ shows;
  3. the small asymmetric peak at $5\,cm$ from the source corresponds to the water/gel interface.

- Pressure wave intensity plots 5.2

  1. the plot represents the intensity of the wave propagating though the medium. The intensity is related to the wave amplitude through Eq. (3.6);
  2. the main peak occurs at the focal length of $8\,cm$, $3\,cm$ inside the phantom object;
  3. as noted about Fig. 5.1, a small jump is present at the water/gel interface. However, the magnitude is smaller by a factor 2α, where α is the absorption coefficient defined in Eq. (3.16).

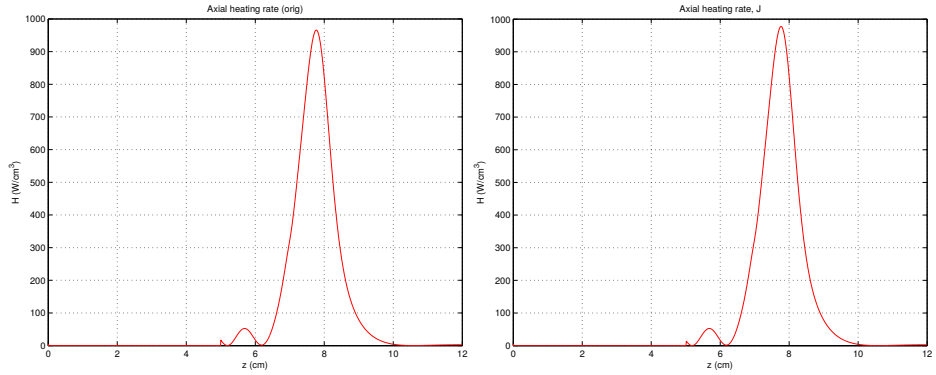

Figure 5.2: Comparison of the wave intensity, in the original (left) and the extended (right) models

Quantity	Model	Peak position (cm)	Max value	FWHM (cm)
H (W/cm^3)	Original	7.77	965.82	1.08
	Multi-layer	7.76	977.504	1.07
I (W/cm^2)	Original	7.59	$2.36x10^3$	1.46
	Multi-layer	7.59	$2.34x10^3$	1.46

Table 5.1: Quantitative comparison the density of energy lost per time unit, H, and pressure wave intensity, I, for the original and the multi-layer models.

For a quantitative comparison, Table 5.1 reports position, height, and full width at half maximum (FWHM) of the heating rate H and intensity I.

Between the two models there are no approximations involved. In fact the solution propagates over the same distance, and the interfaces added by the segmentation of the water layer lie in the same material. Thus, in principle, there is no difference in the transmission coefficients between the two models and actually the plots and the table show good agreement in the computations of the two models.

In Fig. 5.3, 5.4, and 5.5 are shown the output plots of the BHT script, on the left the original model and on the right the extended model. There is a good agreement between the two models.

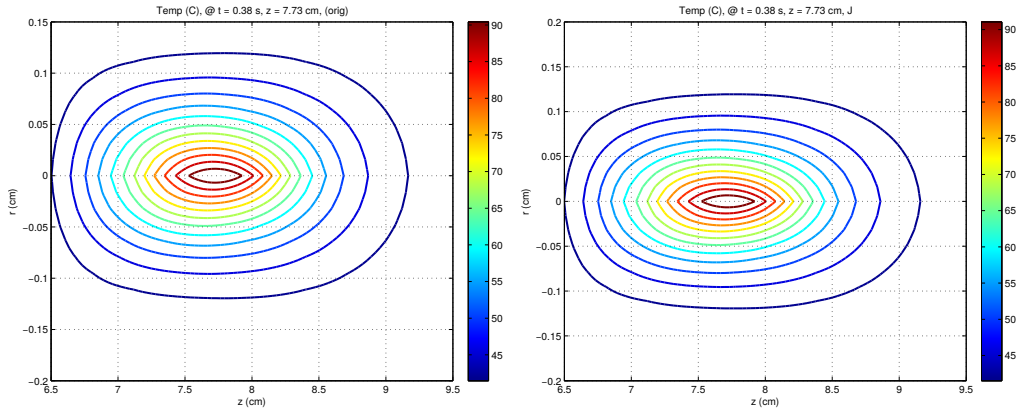

Figure 5.3: Comparison of the contour plots of the temperature reached in the focus, in the original (left) and the extended (right) models.

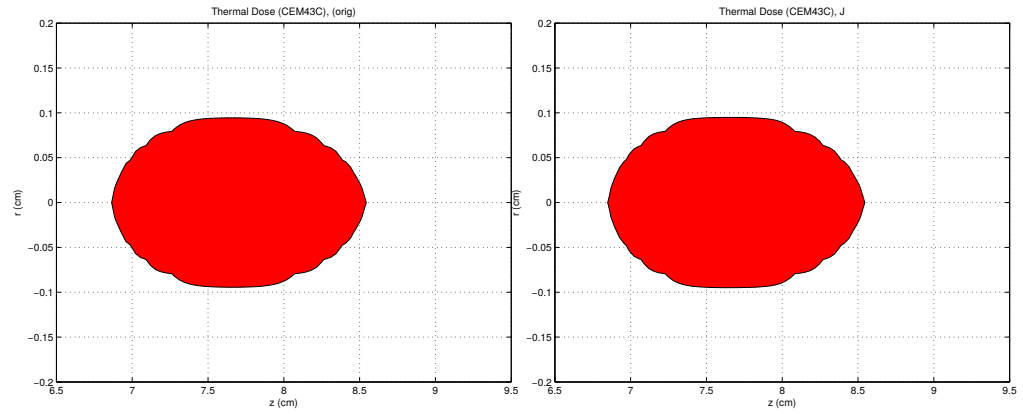

Figure 5.4: Comparison of the thermal dose in the focus, in the original (left) and the extended (right) models.

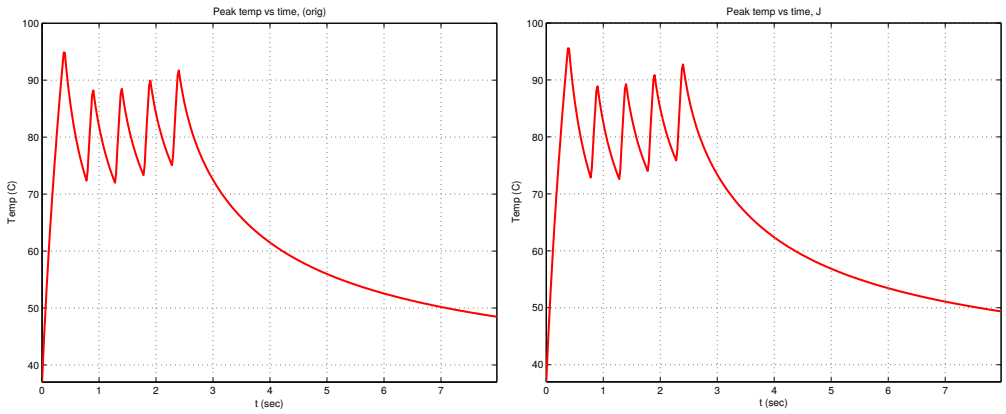

Figure 5.5: Comparison of the temperature vs time, in the original (left) and the extended (right) models.

Stability of the scheme

An additional test is made and involves the stability of the scheme. The computational grid is refined in order to compare the two models as a function of the axial number of nodes, with the radial direction neglected the radial direction.

In Fig. 5.6 the behaviors of H are compared. The plots of H obtained in the two models by multiplying the total number of axial nodes by a factor 2, 3, 4, and 10 are superimposed. To better appreciate the differences, a zoom of the peak region is reported in Fig. 5.7.

The main features of the plots are : a difference that occurs in the peak height in function of the reference grid and the solutions of the two models that tend to a common value, with a step decrease corresponding to the increase of the number of nodes.

The approximation made by a numerical scheme is regarded to tend toward zero with the discrete reaching the continuum and correspondingly, the computed solution to tend toward zero with the discrete reaching the continuum. Correspondingly, the computed solution tends toward the analytical solution. Since the KZK equation, Eq. (3.28), has no analytical solution, it can be assumed that refining the computational grid implies an improvement of the computed solution.

The increments of the peak heights are reported in Fig. 5.8. For both models they show a non divergent behavior.

Within the multi-layer extension, an increase in the segmentation of the water layer that does not affect the solution can be considered a further test for its validation.

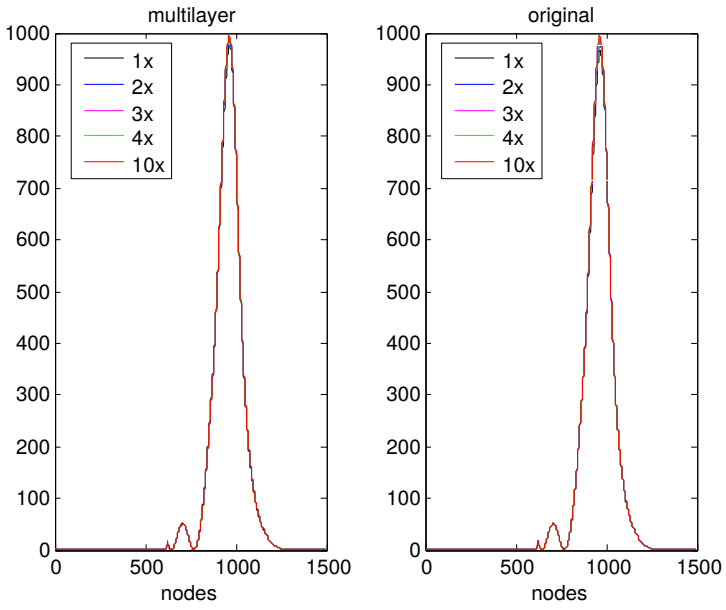

Figure 5.6: Multi-layer model, density of energy refining the computational grid.

5.2 Experimental Data

For a future validation of the code with experimental data, some measurements of sonication have been aqcuired from the *Istituto di Bioimmagini e Fisica Molecolare* of *CNR-LATO*, in Cefalù.

The apparatus consists of the InSightec ExAblate® 2100 which provides HIFU in combination with magnetic resonance imaging (MRIgFUS). The transducer is immersed in a volume of water, on top of which a phantom object lies, as indicated in Fig 5.9.

The characteristics of the materials are shown in Tab. 5.2.

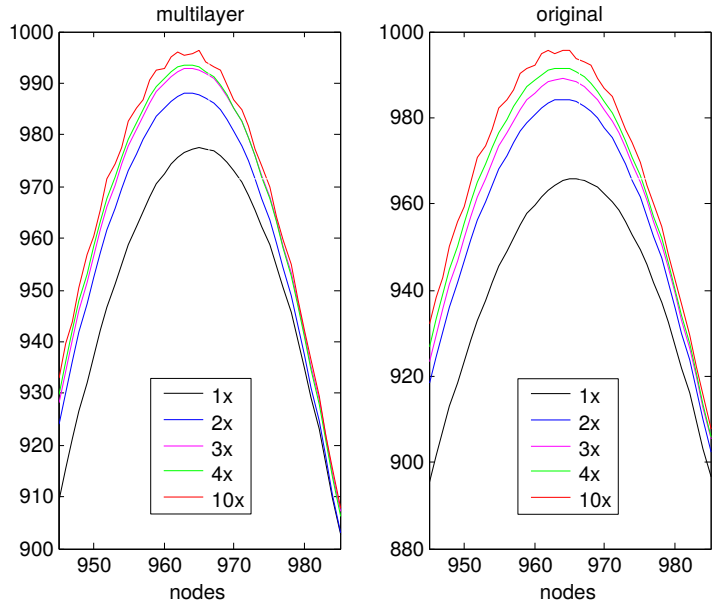

Figure 5.7: Multi-layer model, zoom on the peak of the density of energy refining the computational grid.

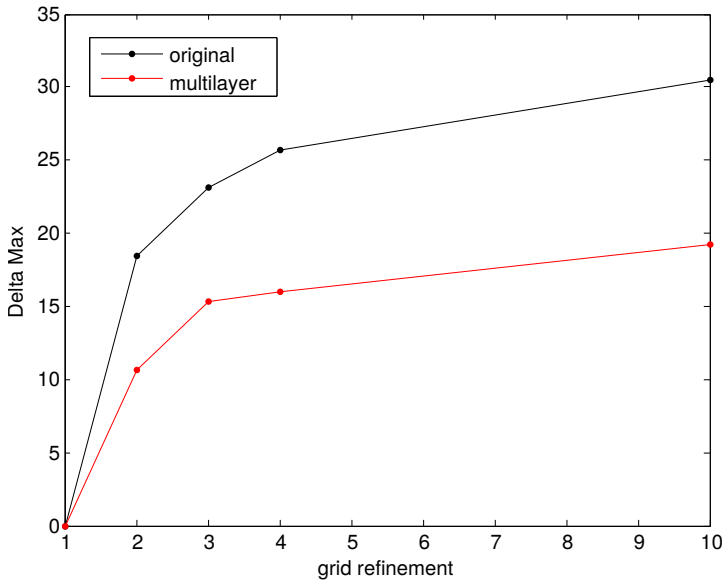

Figure 5.8: Difference between one step and the next in the computational grid refining.

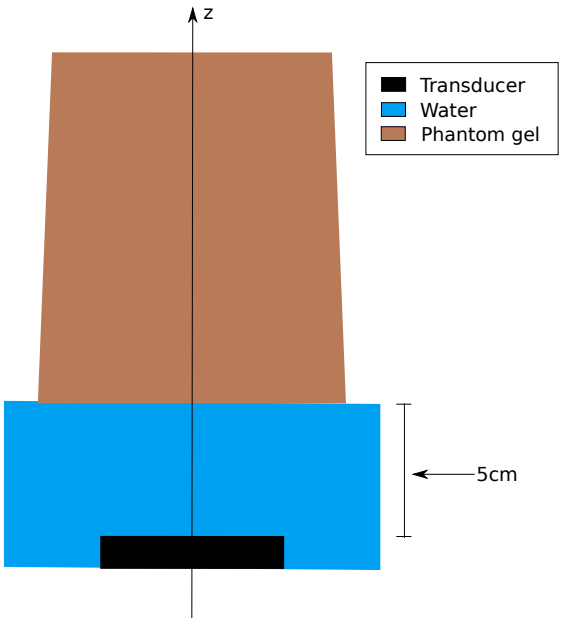

Figure 5.9: Sketch of the experimental setup, axial view. The transducer lies water, then there is the phantom gel.

The layer of water is necessary to completely transmit to the phantom object the pressure wave produced by the transducer, assuring the absence of gas between the two materials. In addition, the water is degassed so that cavitation is avoided.

The only way to measure temperature non-invasively is to extract it from magnetic resonance scans. This kind of scans are fast sequences of images of the same slice. The recorded signal is a complex value and, for each instant, a triplet of the images acquired: the real and imaginary parts, and the absolute value of the signal.

A phase can be defined

$$\Phi = \arctan\left(\frac{Im(x,y)}{Re(x,y)}\right) , \tag{5.1}$$

where $Im(x,y)$ and $Re(x,y)$ are the values of the pixel in the (x,y) position of the real and imaginary images.

The variations of this phase are related to the variations of temperature,

Water	small-signal sound speed	c	1482	$m\,s^{-1}$
	mass density	ρ	1000	$kg\,m^{-3}$
	absorption at $1\,MHz$	α	0.217	$dB\,m^{-1}$
	exponent of absorption vs frequency curve	η	2	———
	nonlinear parameter	β	3.5	———
	specific heat capacity	C_p	4180	$J\,kg^{-1}K^{-1}$
	thermal conductivity	k	0.6	$W\,m^{-1}\,K^{-1}$
	perfusion rate	w	0	$kg\,m^{-3}s^{-1}$
	thickness	z	5	cm
Phantom gel	small-signal sound speed	c	1600	$m\,s^{-1}$
	mass density	ρ	1060	$kg\,m^{-3}$
	absorption at $1\,MHz$	α	68	$dB\,m^{-1}$
	exponent of absorption vs frequency curve	η	1	———
	nonlinear parameter	β	4.8	———
	specific heat capacity	C_p	3850	$J\,kg^{-1}K^{-1}$
	thermal conductivity	k	0.55	$W\,m^{-1}\,K^{-1}$
	perfusion rate	w	18	$kg\,m^{-3}s^{-1}$

Table 5.2: Parameters of the experimental setup at IBFM CNR-LATO.

ΔT, through the relation

$$\Delta\Phi = \alpha\gamma_p B_0 TE\Delta T\,, \tag{5.2}$$

where α is a magnetic constant whose value for water is $\alpha = 0.01 ppm/°C$, $\gamma_p = 2.675 \times 10^8$ is the gyromagnetic ratio of the proton, $B_0 = 1.5T$ is the static magnetic, and $TE = 12.74 \times 10^{-}3$ is the echo time, a parameter depending on the type of magnetic scan performed. The phases of the sequence of a number n of images are subtracted from the first image phase Φ_0, taken as a reference.

The total temperature increase is then

$$\Delta T = \sum_{i=1}^{n-1} \Delta T_i\,, \tag{5.3}$$

where ΔT_i is the temperature difference of the i-th image with respect to

the reference image.

If the initial temperature T_i in known, the final temperature reached is

$$T_f = T_i + \Delta T\,. \tag{5.4}$$

Even though the procedure is quite general and straightforward, it is not trivial to have the correct images.

This analysis has been applied to a sequence of magnetic resonance scans performed to monitor the result of a sonication applied to the experimental setup described above.

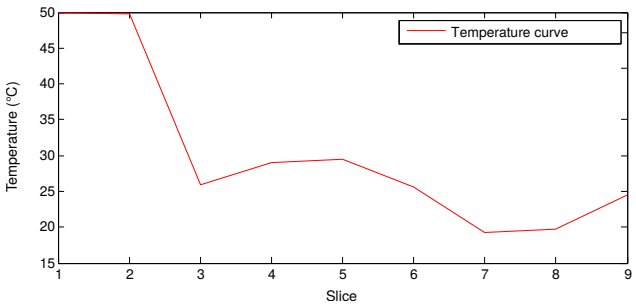

Figure 5.10: The temperature variation.

In Fig. 5.10 there is the plot of the corresponding extracted temperature. The reference temperature is $23.5\,°C$.

The next step is the simulation of the same setup and the comparison of the results.

Chapter 6

Conclusions

This thesis work started with a preliminary study of the fundamentals of nonlinear acoustics and the numerical techniques used to solve the complex equations that model the propagation of pressure waves of finite amplitude. This acquired background allowed the comprehension of the chosen preexisting simulation code, and thus its extension.

The work covered also the study of the medical applications of HIFU, the clinical and experimental setups, and the analysis of the results extracted from the output images.

It was also required the development of a graphic user interface to easily change the simulation inputs according to the apparatus at hand.

Being carried out in the context of a stage at *Istituto Superiore di Sanità*, this work contributed to acquire new knowledges in a frontier field. ISS has to carry on research work so that it can fulfill its mission to protect public health by guaranteeing homogeneity in the quality of treatments offered by the facilities operating with HIFU nationwide.

Appendix A

GUI code

```
function varargout = second_geom(varargin)
% SECOND_GEOM MATLAB code for second_geom.fig
%   SECOND_GEOM, by itself, creates a new SECOND_GEOM or raises the existing
%   singleton*.
%
%   H = SECOND_GEOM returns the handle to a new SECOND_GEOM or the handle to
%   the existing singleton*.
%
%   SECOND_GEOM('CALLBACK',hObject,eventData,handles,...) calls the local
%   function named CALLBACK in SECOND_GEOM.M with the given input arguments.
%
%   SECOND_GEOM('Property','Value',...) creates a new SECOND_GEOM or raises the
%   existing singleton*.  Starting from the left, property value pairs are
%   applied to the GUI before second_geom_OpeningFcn gets called.  An
%   unrecognized property name or invalid value makes property application
%   stop.  All inputs are passed to second_geom_OpeningFcn via varargin.
%
%   *See GUI Options on GUIDE's Tools menu.  Choose "GUI allows only one
%   instance to run (singleton)".
%
% See also: GUIDE, GUIDATA, GUIHANDLES

% Edit the above text to modify the response to help second_geom

% Last Modified by GUIDE v2.5 31-Mar-2015 18:13:58

% Begin initialization code - DO NOT EDIT
gui_Singleton = 1;
gui_State = struct('gui_Name',       mfilename, ...
                   'gui_Singleton',  gui_Singleton, ...
                   'gui_OpeningFcn', @second_geom_OpeningFcn, ...
                   'gui_OutputFcn',  @second_geom_OutputFcn, ...
                   'gui_LayoutFcn',  [] , ...
                   'gui_Callback',   []);
if nargin && ischar(varargin{1})
    gui_State.gui_Callback = str2func(varargin{1});
end

if nargout
    [varargout{1:nargout}] = gui_mainfcn(gui_State, varargin{:});
else
    gui_mainfcn(gui_State, varargin{:});
end
% End initialization code - DO NOT EDIT
```

```
% --- Executes just before second_geom is made visible.
function second_geom_OpeningFcn(hObject, eventdata, handles, varargin)
% This function has no output args, see OutputFcn.
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
% varargin   command line arguments to second_geom (see VARARGIN)

global string_message;
string_message = {};

path('..', path);

%%% fields' initialization

% transducer's sketch
cla(handles.trans_axes,'reset');
set(handles.trans_axes,'Visible','off');

% text
set(handles.text_message, 'String', '');

% waveform, radial and axial plots
cla(handles.axes_waveform, 'reset');
set(handles.axes_waveform, 'Visible', 'on');
set(handles.axes_waveform, 'Tag', 'axes_waveform');

cla(handles.axes_p_vs_r, 'reset');
set(handles.axes_p_vs_r, 'Visible', 'off');
set(handles.axes_p_vs_r, 'Tag', 'axes_p_vs_r');

cla(handles.axes_i_vs_r, 'reset');
set(handles.axes_i_vs_r, 'Visible', 'off');
set(handles.axes_i_vs_r, 'Tag', 'axes_i_vs_r');

cla(handles.axes_h_vs_r, 'reset');
set(handles.axes_h_vs_r, 'Visible', 'off');
set(handles.axes_h_vs_r, 'Tag', 'axes_h_vs_r');

cla(handles.axes_p_vs_z, 'reset');
set(handles.axes_p_vs_z, 'Visible', 'off');
set(handles.axes_p_vs_z, 'Tag', 'axes_p_vs_z');

cla(handles.axes_i_vs_z, 'reset');
set(handles.axes_i_vs_z, 'Visible', 'off');
set(handles.axes_i_vs_z, 'Tag', 'axes_i_vs_z');

cla(handles.axes_h_vs_z, 'reset');
set(handles.axes_h_vs_z, 'Visible', 'off');
set(handles.axes_h_vs_z, 'Tag', 'axes_h_vs_z');

% plots' radiobuttons
set(handles.waveform, 'Value', 1);
set(handles.p_vs_r, 'Value', 0);
set(handles.i_vs_r, 'Value', 0);
set(handles.h_vs_r, 'Value', 0);
set(handles.p_vs_z, 'Value', 0);
set(handles.i_vs_z, 'Value', 0);
set(handles.h_vs_z, 'Value', 0);

% MRI image
cla(handles.MRI_axes, 'reset');
set(handles.MRI_axes, 'Visible', 'off');

% BHT togglebutton
set(handles.toggleBHT, 'Value', 0);

% all plots checkbutton
set(handles.check_showall, 'Value', 1);
```

```
%
string_message{end+1} = '     *****     Welcome     *****';
set(handles.text_message, 'String', string_message, 'Value', 1);


%%%% TEMPORARY!!! %%%% --> red background
%%%% SOLO PERCHE' MI SCOCCIO A SCRIVERE I NUMERI TUTTE LE VOLTE
set(handles.inner_text, 'String', 1.5);
set(handles.outer_text, 'String', 2.5);
set(handles.focusing_text, 'String', 8);
set(handles.freq_text,  'String', 1.5e6);
set(handles.power_text, 'String', 100);
%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%


% Choose default command line output for second_geom
handles.output = hObject;

% Update handles structure
guidata(hObject, handles);

% UIWAIT makes second_geom wait for user response (see UIRESUME)
% uiwait(handles.figure1);




% --- Executes on button press in pushb_OK.
function pushb_OK_Callback(hObject, eventdata, handles)
% hObject    handle to pushb_OK (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

%path('..', path)
%%soluzioneKZK(handles)
%KZK_parameters(handles);
%%KZK_input_parameters(handles);
%[p0,c1,c2,rho1,rho2,N1,N2,G1,G2,gamma1,gamma2,a,b,d,f,R,Z,z_,K] = KZK_parameters(
    handles);
a = str2num(get(handles.outer_text, 'String'));
b = str2num(get(handles.inner_text, 'String'));
d = str2num(get(handles.focusing_text, 'String'));
f = str2num(get(handles.freq_text,  'String'));
P = str2num(get(handles.power_text, 'String'));
%%% select the correct axes for transducer's figure
axes(handles.trans_axes);
drawCylindricalTransducer(a,b,d,handles);


% --- Executes on button press in pushB_reset.
function pushB_reset_Callback(hObject, eventdata, handles)
% hObject    handle to pushB_reset (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

%set(handles.inner_text, 'String', '');
%set(handles.outer_text, 'String', '');
%set(handles.depth_text, 'String', '');

% --- Executes on button press in pushB_start.
function pushB_start_Callback(hObject, eventdata, handles)
% hObject    handle to pushB_start (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

global string_message;
path('../HIFU/', path);
```

```
%[z,r,H,I,Ppos,Pneg] = soluzioneKZK(handles);
BHTchoice = get(handles.toggleBHT, 'Value');
set(handles.pushB_start, 'Enable', 'off')
mainBHT(BHTchoice,handles);
disp('ancora viva')



% --- Executes when selected object is changed in plot_panel.
function plot_panel_SelectionChangeFcn(hObject, eventdata, handles)
% hObject    handle to the selected object in plot_panel
% eventdata  structure with the following fields (see UIBUTTONGROUP)
%       EventName: string 'SelectionChanged' (read only)
%       OldValue: handle of the previously selected object or empty if none was
    selected
%       NewValue: handle of the currently selected object
% handles    structure with handles and user data (see GUIDATA)

%%% when the radio button is selected only the
%%% corresponding plot is visible

% get the handle to the previously selected button
old = get(eventdata.OldValue,'Tag');
% concatenate strings
ax_old = ['axes_' old];
% get the handle to the object (axes) with the specified tag
hax_old = findobj('Tag', ax_old);
% set the visibility of that object
set(hax_old, 'Visible', 'off');
% get its children (plot(s))
plot = get(hax_old, 'children');
% set the visibility of those children
set(plot, 'Visible', 'off');

% get the handle to the currently selected button
new = get(eventdata.NewValue, 'Tag');
% concatenate strings
ax_new = ['axes_' new];
% get the handle to the object (axes) with the specified tag
hax_new = findobj('Tag', ax_new);
% set the visibility of that object
set(hax_new, 'Visible', 'on');
% get its children (plot(s))
plot = get(hax_new, 'children');
% set the visibility of those children
set(plot, 'Visible', 'on');



% --------------------------------------------------------------------
function uipushtoolOpen_ClickedCallback(hObject, eventdata, handles)
% hObject    handle to uipushtoolOpen (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

global string_message;
%%% esempio da copiare e poi buttare
[filename,pathname] = uigetfile({'*.dcm;','All Image Files';...
    '*.*','All Files' })

if isequal(filename,0)
    string_message{end+1} = sprintf('Non Ã¨ stata selezionata nessuna immagine')
else

    name = removeExtension(filename)
    f = fullfile(pathname, name)

    string_message{end+1} = sprintf('L''utente ha selezionato il file');
    string_message{end+1} = sprintf('%s', name);
```

```
    set(handles.text_message, 'String', string_message,...
        'Value', length(string_message));
    pause(0.7)

    I = dicomread(f);
    set(handles.MRI_axes);
    imshow(I);
    imagesc;

end

% --- Executes on button press in toggleBHT.
function toggleBHT_Callback(hObject, eventdata, handles)
% hObject    handle to toggleBHT (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of toggleBHT

% if BHT is selected, the axisymmetricBHT script needs to be run
choice = get(hObject,'Value');
if choice == 1 % BHT selected

   % emphasize that BHT is selected
   set(hObject, 'FontWeight', 'bold');
else
   set(hObject, 'FontWeight', 'normal');
end


% --- Executes on button press in check_showall.
function check_showall_Callback(hObject, eventdata, handles)
% hObject    handle to check_showall (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hint: get(hObject,'Value') returns toggle state of check_showall

choice = get(hObject,'Value');

if choice == 1 % show all plots selected
    set(handles.plot_panel, 'Visible', 'on')
else
    set(handles.plot_panel, 'Visible', 'off')
end


% --- Executes during object deletion, before destroying properties.
function figure1_DeleteFcn(hObject, eventdata, handles)
% hObject    handle to figure1 (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

clearvars
close all




%%%%%%%%%%%%%%% not used, for the time being %%%%%%%%%%%%%%%
%%%%%%%%%%%%%%% still, do not delete...      %%%%%%%%%%%%%%%


% --- Outputs from this function are returned to the command line.
function varargout = second_geom_OutputFcn(hObject, eventdata, handles)
% varargout  cell array for returning output args (see VARARGOUT);
% hObject    handle to figure
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)
```

```
% Get default command line output from handles structure
varargout{1} = handles.output;


% --- Executes on selection change in text_message.
function text_message_Callback(hObject, eventdata, handles)
% hObject    handle to text_message (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: contents = cellstr(get(hObject,'String')) returns text_message contents as
    cell array
%        contents{get(hObject,'Value')} returns selected item from text_message


% --- Executes during object creation, after setting all properties.
function text_message_CreateFcn(hObject, eventdata, handles)
% hObject    handle to text_message (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: listbox controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'
    defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


function inner_text_Callback(hObject, eventdata, handles)
% hObject    handle to inner_text (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of inner_text as text
%        str2double(get(hObject,'String')) returns contents of inner_text as a double


% --- Executes during object creation, after setting all properties.
function inner_text_CreateFcn(hObject, eventdata, handles)
% hObject    handle to inner_text (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'
    defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


function outer_text_Callback(hObject, eventdata, handles)
% hObject    handle to outer_text (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of outer_text as text
%        str2double(get(hObject,'String')) returns contents of outer_text as a double


% --- Executes during object creation, after setting all properties.
function outer_text_CreateFcn(hObject, eventdata, handles)
% hObject    handle to outer_text (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
```

```
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'
    defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


function focusing_text_Callback(hObject, eventdata, handles)
% hObject    handle to focusing_text (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of focusing_text as text
%        str2double(get(hObject,'String')) returns contents of focusing_text as a
    double


% --- Executes during object creation, after setting all properties.
function focusing_text_CreateFcn(hObject, eventdata, handles)
% hObject    handle to focusing_text (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'
    defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


function power_text_Callback(hObject, eventdata, handles)
% hObject    handle to power_text (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of power_text as text
%        str2double(get(hObject,'String')) returns contents of power_text as a double


% --- Executes during object creation, after setting all properties.
function power_text_CreateFcn(hObject, eventdata, handles)
% hObject    handle to power_text (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called

% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'
    defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end


function freq_text_Callback(hObject, eventdata, handles)
% hObject    handle to freq_text (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    structure with handles and user data (see GUIDATA)

% Hints: get(hObject,'String') returns contents of freq_text as text
%        str2double(get(hObject,'String')) returns contents of freq_text as a double


% --- Executes during object creation, after setting all properties.
function freq_text_CreateFcn(hObject, eventdata, handles)
% hObject    handle to freq_text (see GCBO)
% eventdata  reserved - to be defined in a future version of MATLAB
% handles    empty - handles not created until after all CreateFcns called
```

```
% Hint: edit controls usually have a white background on Windows.
%       See ISPC and COMPUTER.
if ispc && isequal(get(hObject,'BackgroundColor'), get(0,'
    defaultUicontrolBackgroundColor'))
    set(hObject,'BackgroundColor','white');
end

%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%%
```

Bibliography

[1] *Physiscs of thermal therapy*, ed E. G. Moros

[2] *Relazione sullo Stato Sanitario del Paese 2012-2013*, Ministero della Salute

[3] K. Hynynen, C. Damianou, A. Darkazanli, E. Unger, J. F. Schenck, *The feasibility of using MRI to monitor and guide noninvasive ultrasound surgery*, Ultrasound Med. Biol. **19** 91-92 (1993)

[4] K. Hynynen, N. I. Vykhodtseva, A. H. Chung, V. Sorrentino, V. Colucci, F. A. Jolesz, et al., *Thermal effects of focused ultrasound on the brain: determination with MR imaging*, Radiology **206** 247-253 (1997)

[5] C. Bohris, W. G. Schreiber, J. Jenne, I. Simiantonakis, R. Rastert, H. J. Zabel, P. Huber, R. Bader, G. Brix, *Quantitative MR temperature monitoring of high-intensity focused ultrasound therapy*, Magn. Reson. Imaging **17** 603-610 (1999)

[6] F. Kallel, *The feasibility of elastographic visualization o HIFU induced thermal lesion in soft tissues*, Ultrasound in Med. & Biol. **25** 641-647 (1999)

[7] F. Orsi, P. Arnone, W. Chen, L. Zang, *High intensity focused ultrasound ablation: a new therapeutic option for solid tumors*, Journal of Cancer Research and Therapeutics **6** 414-420 (2010)

[8] W. M. Gedroyc, *New clinical applications of magnetic resonance-guided focused ultrasound*, Top. Magn. Reson. Imaging **17** 189-194 (2006)

[9] C. T. Moonen, B. Quesson, R. Salomir, *Thermal therapies in interventional MR imaging: focused ultrasound*, Neuroimaging Clin. N. Am. **11** 737-747 (2001)

[10] E. A. Stewart, W. M. Gedroyc, C. M. Tempany, *Focused ultrasound treatment of uterine fibroid tumors: safety and feasibility of a noninvasive thermoablative technique*, Am. J. Obstet. Gynecol. **189** 49-54 (2003)

[11] C. M. Tempany, E. A. Stewart, N. McDannold, B. J. Quade, F. A. Jolesz, K. Hynynen, et al., *MR imaging-guided focused ultrasound surgery of uterine leiomyomas: a feasibility study*, Radiology **226** 897-905 (2003)

[12] W. Z. Chen, K. Zohu, *High-intensity focused ultrasound ablation: a new strategy to manage primary bone tumors*, Curr. Opin . Orthop. **16** 494-500 (2005)

[13] G. R. ter Haar, *Therapeutic and Surgical Applications*, in Physical Principles of Medical Ultrasonics, eds. C. R. Hill, J. C. Bamber, and G. R. ter Haar, 407-456, John Wiley & Sons

[14] B. Rosenberg, G. Kemeny, R. C. Switzer, T. C. Hamilton, *Quantitative Evidence for Protein Denaturation as the Cause of Thermal Death*, Nature **232** 471-473 (1971)

[15] K. F. Chu, D. E. Dupuy, *Thermal ablation of tumours: biological mechanisms and advances in therapy* Nature Reviews Cancer **14** 199â208 (2014)

[16] S. A. Sapareto, W. C. Dewey, *Thermal dose determination in cancer therapy*, Int. J. Radiat. Oncol Biol. Phys. **10** 787-800 (1984)

[17] M. W. Dewhirst, B. L. Viglianti, M. Lora-Michielis, M. Hanson, P. J. Hoopes, *Basic principles of thermal dosimetry and thermal thresholds for tissue damage from hyperthermia.* Int. J. Hyperthermia **19** 267-294 (2003)

[18] C. A. Damianou, K. Hynynen, *The effect on the size and shape of necrosed tissue volume during ultrasound surgery*, J. Acoust. Soc. Am. **95** 1641-1649 (1994)

[19] J. Bamber, G. R. ter Haar, C. Hill, *Physical Principles of medical Ultrasound*, John Wiley & Sons

[20] H. H. Pennes *Analysis of tissue and arterial blood temperatures in the resting human forearm* J. Appl. Physiol. **1** 93-122 (1948)

[21] D. T. Blackstock, M. F. Hamilton, A. D. Pierce, *Progressive waves in lossless and lossy fluids*, *Nonlinear acoustics*, eds. M. F. Hamilton, D. T. Blackstock, Academic Press

[22] E. A. Zabolotskaya and R. V. Khokhlov, *Quasi-plane waves in the nonlinear acoustics of confined beams*, Sov. Phys. Acoust. **15** 35-40 (1969)

[23] V. P. Kuznetsov, *Equations of nonlinear acoustics*, Sov. Phys. Acoust. **16** 467-470 (1971)

[24] Y-S. Lee, *Numerical solution of the KZK equation for pulsed finite amplitude sound beams in thermoviscous fluids*, PhD dissertation, The University of Texas at Austin (1993)

[25] L. E. Kinsler, A. R. Frey, A. B. Coppens, J. V. Sanders, *Fundamentals of acoustics*, John Wiley & Sons

[26] Mathworks File Exchange

[27] J. E. Soneson, *A User Friendly Software Package for HIFU Simulation*, AIP Conf. Proc. **1113** 165 (2009)

[28] M. F. Hamilton, J. Naze Tjøtta, S. Tjøtta, *Nonlinear effects in the farfield of a directive sound source*, J. Acoust. Soc. Am. **78** 202-216 (1985)

[29] Y.-S. Lee and M. F. Hamilton, *Time-domain modeling of pulsed finite-amplitude sound beams*, J. Acoust. Soc. Am. **97** 906-917 (1995)

[30] M. F. Hamilton, *Sound beams*, *Nonlinear acoustics*, eds. M. F. Hamilton, D. T. Blackstock, Academic Press

[31] D. Anderson, J. Tannehill, R. Pletcher, *Computational fluid mechanics and heat transfer* McGraw-Hill 1984

[32] U. M. Ascher, L. R. Petzold, *Computer Methods for Ordinary differential equations and differential-algebraic equations*, Society for Industrial and Applied Mathematics

[33] L. H. Thomas, *Elliptic problems in linear difference equations over a network*, Watson Sci. Comput. Lab. Rept. (1949)