

**Modeling a phase partition problem to assess the
appearance of NAPL in polluted soils**

Álvaro Pérez Lada

Thesis to obtain the Master of Science Degree in

Engenharia Geológica e de Minas

Orientadores:

Prof Maria João Correia Colunas Pereira

Prof Fernando Barrio Parra

Júri

Presidente/ Chairperson: Prof. Maria Matilde Mourão de Oliveira Carvalho Horta
Costa e Silva

Orientador/ Supervisor: Prof Maria João Correia Colunas Pereira

Vogal(is)/ Members of the Committe: Prof Leonardo Azevedo Guerra Raposo Pereira

December 2020

I hereby declare that this document is an original work of mine and complies with all the requirements of the Code of Conduct and Good Practices of the University of Lisbon.

ABSTRACT

NAPL is a well-known term used to describe those non aqueous phase liquids. These non-aqueous liquids appear under certain edaphic conditions and the aim of this project is to truly understand how these conditions can affect the appearance of the NAPLs and in which measure.

NAPLs represent a major threat for the living organisms in the fields (so for the humans who consume contaminated vegetables for instance) and they appear in a situation where some hydrocarbons have been leaked to the ground. It becomes important to quantify the appearance of NAPL when the ground has reached the saturation point where no more liquids can be absorbed because that's when the NAPLs can move through the ground.

A mathematical model has been built, the series of equations describe the procedures needed to see whether there is NAPLs under the initial both edaphic and chemical conditions. A soil is described, characterizing the main parameters as the pollutant concentration in the soil, the total porosity, the water filled porosity, the air filled porosity, the organic carbon fraction etc. Afterwards the chemical inputs for a certain contaminant are also characterized. The programme build with Matlab code, allows to quantify the NAPL in the soil.

It is important to evaluate the already built programme, by varying some of the inputs to see whether the appearance of NAPL changes. The porosity inputs turned not to be a determinant for the appearance of the NAPLs under the initial conditions for the case of study. Evaluating the variation of the contaminant concentration in soil has led to the creation of a new programme that finds the limit for the pollutant concentration which verifies the appearance of NAPL.

Keywords: saturated ground, porosity, groundwater contamination.

RESUMO

NAPL é um termo bem conhecido usado para descrever os líquidos de fase não aquosa dos solos. Esses líquidos não aquosos aparecem sob certas condições edáficas e o objetivo deste projeto é compreender como essas condições podem afetar o surgimento dos NAPLs e em que medida.

Os NAPLs representam uma grande ameaça para os organismos vivos (por exemplo, para os humanos que consomem vegetais contaminados) e aparecem em situações em que hidrocarbonetos vazaram para o solo. Torna-se importante quantificar o aparecimento de NAPL quando o solo atingiu o ponto de saturação onde nenhum líquido pode ser absorvido, porque é quando os NAPLs se podem mover pelo solo.

Construiu-se um modelo matemático a partir de equações que verificam se sob as condições edáficas e químicas iniciais poderá ocorrer NAPLs. Um solo é descrito, caracterizando os principais parâmetros como a concentração de poluente no solo, a porosidade total, a proporção de água e ar dos poros, a fração de carbono orgânico, entre outros parâmetros. Em seguida são caracterizados as características químicas para um determinado contaminante. O programa construído com código Matlab, permite quantificar o NAPL no solo.

Por último foi realizada uma análise de sensibilidade do modelo para avaliar a importância de algumas variáveis de input na variação de NAPL, tendo-se verificado que nos casos de estudo analisados a proporção de ar e água não foram determinantes para o surgimento dos NAPLs nas condições iniciais. A avaliação da variação da concentração do contaminante no solo levou à criação de um novo programa que encontra o limite de concentração do poluente que verifica o surgimento de NAPL.

Palavras-chave: NAPL, solo saturado, porosidade do solo, contaminação de águas subterrâneas,

Tabla de contenido

ABSTRACT	ii
RESUMO	iv
1. Introduction	1
2. State of the art	2
2.1 Soils	2
2.1.1 Ground horizons	4
2.1.2 Water and air phases in the soil	5
2.1.3 Mineral matter in the soil	6
2.1.4 Soil parameters	6
2.1.5 Soil organic matter	8
2.1.6 Organic carbon fraction	9
2.2 NAPL	10
2.2.1 DNAPL	10
2.2.2 LNAPL	13
2.3 Phase distribution estimation	16
3. Methods	17
3.1 NAPL free phase calculation	17
3.1.1 Molar fraction	18
3.1.2 Density of the mix	18
3.1.3 Effective solubility	19
3.1.4 Effective vapor pressure	19
3.1.5 Theoretical concentration in aqueous pore	20
3.1.6 Saturation limit for soil	21
3.1.7 Volumetric saturation limit	21
3.1.8 Concentration in aqueous pore	21
3.1.9 Concentration in steam pore	21
3.1.10 Volumetric concentration in steam pore	22
3.1.11 Concentration absorbed by the solid	23
3.1.12 Mass of contaminant in aqueous pore	23
3.1.13 Mass of pollutant in air pore	23
3.1.14 Absorbed mass of pollutant in the soil	24
3.1.15 NAPL mass for compound	25
3.1.16 Mass balance for the compound	25
3.1.17 Total mass of NAPL in the sample	25
3.1.18 NAPL residual saturation percentage	26
3.1.19 Volumetric content of NAPL in the sample	26
3.1.20 Volumetric content of NAPL in air	26
3.1.21 Volumetric water content if the air content were $a = 0$	27
3.1.22 Volumetric content of solid in the sample	27
4. Results	28
4.1 Evaluation of pollutants	28

4.1.1	Trichloroethylene	28
4.1.2	Arochlor 1260	31
4.2	Sensitivity analysis.....	32
4.2.1	Edaphic parameters	32
4.2.2	Chemical parameters	42
4.2.3	Discussion.....	46
5.	Conclussion	47
6.	Bibliography.....	48
7.	Annexed	50
7.1	Code.....	50

Table of tables

Table 1-Chemical parameters	28
Table 2-Chemical parameters for Trichloroethylene	29
Table 3-Obtained data.....	29
Table 4-Chemical parameters for Arochlor 1260	31
Table 5-Obtained data.....	31

Table of figures

Figure 1- Outline of the NAPL s migration problem. Source (Charles J. Newell)	1
Figure 2- LNAPL leak model. (Source: https://www.ienvi.com.au/INAPL /)	13
Figure 3- LNAPL thickness depending on the LNAPL saturation. (Source: LNAPL Part 1: A deeper understanding of the behavior of LNAPL in the subsoil)	14
Figure 4-Matlab code	33
Figure 5- Test_sumW	34
Figure 6-Command window Matlab	35
Figure 7-Matlab NAPL_sumA code	36
Figure 8-Matlab code TEST_sumA	37
Figure 9-Command window Matlab	38
Figure 10-NAPL_nt code	39
Figure 11-TEST_nt.....	40
Figure 12-Command window Matlab	41
Figure 13-NAPL_Cti	42
Figure 14-TEST_Cti	43
Figure 15-Last iterations of TEST_Cti.....	44
Figure 16-Workspace	44
Figure 17-Results for the last iteration	45

1. INTRODUCTION

NAPL s (Non-aqueous phase liquids) are hydrocarbons that exist as an immiscible and separate phase when in contact with water and / or air.

The problem arises when there is a leak of hydrocarbons to a terrain, these leak into it and it moves, due to the force of gravity, downwards. If the area is not saturated, the NAPL s (Non-aqueous phase liquids) are retained in the pores until their movement stops.

On the other hand, when the area is saturated, that is, there are no more free pores without liquid, the NAPL (Non-aqueous phase liquids) moves as a free phase, guided by the movement governed by the force of gravity and capillary forces.

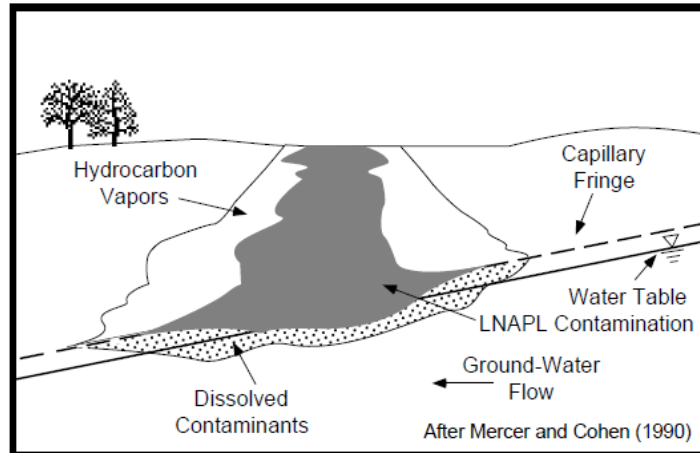


Figure 1- Outline of the NAPL s migration problem. Source (Charles J. Newell)

To characterize the problem, the following factors must basically be considered:

- Ground.
- NAPL s.
- Phase distribution.

Once these factors have been described, which form the Genesis of our problem, we will have to define the practical case at hand, that is, the construction of a mathematical model that describes and calculates the distribution in phases of the NAPL during their migration through saturated soil. For this, the MATLAB mathematical program will be used.

2. STATE OF THE ART

In this chapter the aforementioned fundamental factors for the problem that concerns this master's thesis will be described.

2.1 SOILS

There are many definitions of soil depending on the field in which you are working, the geological and biological considerations that are being made. The United States Department of Agriculture (USDA) offers many, including the following:

- The unconsolidated mineral or organic material in the Earth's surface layer that serves as a natural environment for plant growth.
- Unconsolidated mineral or organic material from the Earth's surface layer that has been subjected to and shows the effects of genetic and environmental factors: climate (including the effects of water and temperature), and macro and microorganisms, conditioned by the relief, acting on the precursor material for a period of time. The soil produced differs from the material from which it has been derived in physical, chemical, biological and morphological properties and characteristics.

But there is a definition that perfectly includes aspects related to composition, structure and origin (genesis), which can condition the interaction that occurs between the soil and the contaminants that are incorporated into it. (Ambiente et al., 2015):

- Soil is a complex mixture of air, aqueous solution, organic matter and disaggregated mineral matter, with a more or less developed horizontal or sub-horizontal layered structure, resulting from the weathering of rocks on the Earth's surface.

The soil formation starts from the base of the bedrock, which, when weathered by meteorological agents, remains exactly in the same place where it has been previously weathered, giving rise to a nutrient-rich mantle that provides the necessary conditions for development of the life.

In this mantle microorganisms, plants and insects are established, whose biological activity favors the appearance of the so-called horizons, that is, differentiated areas within the subsoil.

The main precursor agents for bedrock weathering and subsequent soil formation are:

- **Originating material:** The precursor material represents the original rocks, sediments, lava or pyroclastic accumulations on which a soil develops. Soil can develop in situ by weathering the primary material ("residual" soils) or from materials transported from other areas by wind, rivers, or glaciers. (Ambiente et al., 2015)

- **Weather:** The characteristics of a soil vary depending on the climate of the region in which it has developed. The temperature and rainfall regime determine the weathering and leaching pattern, and the wind redistributes the mineral particles, essentially in arid regions. The amount, intensity, periodicity and type of precipitation influences the formation of the soil by dissolving the materials and allowing them to change phase and facilitating biological activity. Similarly, seasonal and daily changes in temperature affect effective humidity, biological activity, rates of chemical reactions, and vegetation types. Generally speaking, high values of temperature and precipitation cause more intense weathering and appearance of better developed soils. (Ambiente et al., 2015)
- **Topography and relief:** Soils formed on steep topographies are subject to greater erosion stress than those formed on gentle slopes, with the slope of the terrain exerting a negative influence on soil formation. The erosion produced, among other factors by the surface runoff that occurs in areas of high slope, continuously removes the surface layer of soil that may be generated. In these locations, most of the rainwater forms surface runoff since the transit times do not allow its infiltration into the ground. The vegetation that develops in these soils is sparse and the soil that is formed is relatively shallow. In contrast, soils formed on smoother slopes usually have better drainage and therefore tend to be more powerful and better developed. (Ambiente et al., 2015)
- **Biological factors:** The soil hosts different forms of life, from microorganisms to vascular plants. These life forms have two important effects on soil formation: they help the disaggregation of the original material (through the opening of channels) and provide organic matter for more biological activity. The leaves of plants and other plant debris fall on the surface and decompose in the soil. Organisms break down these debris and mix it with the upper soil. Trees and shrubs have large roots that can grow to considerable depths. (USDA sf). These roots open pathways through thick layers facilitating the percolation of rainwater. The microorganisms affect the chemical exchanges that take place between the roots and the soil. (Ambiente et al., 2015)
- **Weather:** The time during which the above factors interact with the soil is also a factor that intervenes in its formation. Soil formation is a slow and continuous process, which can take hundreds to thousands of years, depending on how favorable the factors listed above are. Typically an old surface will have more possibilities of having developed a soil, presenting characteristics that reflect the factors that have intervened in its formation, than the surfaces exposed more recently. (Ambiente et al., 2015)

2.1.1 Ground horizons

Soil horizons are the different regions differentiable by their geophysical, geochemical and biological characteristics.

These horizons are formed by the processes described in the previous section, and almost always involve a movement of matter vertically, downwards or upwards, depending on the balance of matter between precipitation and evaporation. (Ambiente et al., 2015)

The horizons are as follows:

- Horizon O: It is the most superficial layer and is made up of decomposing leaves in the forests, remains of branches, moss.
- Horizon H: It is a layer formed by organic material without decomposing, but saturated with water for long periods of time.
- Horizon A: Its organic matter is humified and integrated with the mineral fraction of the horizon. Typically dark gray, more or less black, but when it contains little organic matter (frequent case in cultivated soils) it can be light. Crumble and granular structure.
- Horizon E: Strong wash horizon. Typically located between A and B. With less clay, Fe and Al oxides, and organic matter than the hor. A and that the horizon B. Very sandy and very light colors (high values), sometimes white. Structure with a very low degree of development (the laminar is typical of this horizon) (Edafología, 2006)
- Horizon B: Horizon of enrichment in clay (illuvial or in situ), Fe and Al oxides (iluvial or in situ) or organic matter (only if it is of illuvial origin; not in situ), or also by evidence of washing of carbonates (if present in the rock), or residual concentration of sesquioxides. It has brown and red colors. With development of edaphic structure (typically in angular, subangular, prismatic blocks). (Edafología, 2006)
- Horizon C: Original material, hard, cannot be dug, can show degree of weathering but can never be edaphized.
- Horizon R: It is the bedrock, it cannot be dug, hard material.

2.1.2 Water and air phases in the soil

For the perfect understanding of the problem, it is necessary to differentiate two areas of the subsoil, in each of them the fluids migrate in a different way:

- **Vado or unsaturated zone:** in it the water moves due to the gravitational forces. The pores are occupied by two phases (water and air). The air when in contact with the atmospheric air has a very similar composition, although the air contained in the pores of the subsoil presents a slight increase in CO₂ and other gases, due to the biological activity present in that horizon of the terrain. This gaseous enrichment turns the water present into much more acid due to its dissolution in it.
- **Saturated zone:** it is the zone in which all the pores are occupied, that is to say, all the possible volume for the storage of fluids is occupied, reason why they move in relation to hydraulic gradient forces. The composition of the water present in the soil is strongly conditioned by the solid matrix (organic and inorganic) with which it is in contact. The soil solution contains different solutes (cations and / or anions of mineral origin) as well as colloids in suspension, both organic (macromolecules) and inorganic (typically clay particles). (Ambiente et al., 2015). These media are the most conducive to retention of NAPL -type contaminants.

In addition, there is another zone, the transition zone between the vadose and the saturated zone, called the capillary strip, in which the pore water content is high, and this is moved by capillary forces.

2.1.3 Mineral matter in the soil

The constituent minerals of the soil can be classified into two groups according to their genesis:

- **Disrupting minerals:** this is the name given to the minerals formed by chemical associations and reactions when the primitive minerals of the bedrock are weatherized and by processes (ie metasomatics) they have changed their composition becoming other minerals. Among them are mainly salts, oxides and hydroxides.
- **Residual minerals:** they are simply those that are weathered from the bedrock, but without undergoing any significant change in their composition. The most important residuals are silicates, since they constitute 98% of the continental terrestrial crust. Among them special mention must be made of the phyllosilicates subclass, and within them the clays. The basic structural unit of the phyllosilicates are silicon and oxygen tetrahedra, SiO_4^{4-} , which unite by sharing three of their four oxygens with other neighbors, forming infinite extension layers and formula $(\text{Si}_2\text{O}_5)^{2-}$ (tetrahedral layer). In them the tetrahedrons are distributed forming hexagons. The minerals in the clay behave like colloids in the soil and given their structure and high specific surface they are very active minerals from the point of view of the retention of contaminants (Ambiente et al., 2015).

2.1.4 Soil parameters

There are certain characteristics of the soil, intrinsic to it and that are related to a greater or lesser extent with its formation or the weathering processes that it has undergone, which are very important to characterize it correctly.

- **Total porosity:** it is the quotient between the volume of pores that presents a fraction of the soil referred to the total volume of the same, expressed as a percentage. Although it is possible to measure the total porosity directly, it is rarely done, since to determine the pore volume, the sample must be totally saturated, and experimentally it is difficult to do so, so it is almost always obtained indirectly from the expression.

$$(1) \quad n = 1 - \frac{\rho_b}{\rho_p}$$

- n = Total porosity
- ρ_b = Apparent density of the soil
- ρ_p = Particle density

- **Apparent density:** this is the density of the soil, which has been previously dried. It is the quotient between the mass of the soil dried at 105°C and its volume. The low apparent density is equivalent to more porosity and greater aggregates to the soil.

$$(2) \rho_b = \frac{M_{o.d}}{V_t}$$

- $M_{o.d}$ = Mass of dry soil
- V_t = Sample volume

- **Particle density:** this is the connection between the mass of dry soil and the volume of solids.

$$(3) \rho_p = \frac{M_{o.d}}{V_s}$$

- V_s = Volume of solids

- **Effective porosity:** refers exclusively to the part of the holes that is useful for the free circulation of water. Therefore, it relates the volume of removable water in the medium to the total volume of material. (Rebollo & Hidrogeología A, nd)

- **Water saturated porosity:** it is the fraction of the total volume of pores occupied by water. There are several ways to express it. The first of these is the gravimetric content of water that is estimated by drying (at 105 ° C) a soil sample and estimating the water mass by losing the weight of the sample after it. The method does not remove structural water from the sample.

$$(4) \theta_g = \frac{M_{agua}}{M_{o.d}}$$

- θ_g = Gravimetric content of water
- M_{agua} = Body of water in the sample

The volumetric content in water is estimated in a similar way to the gravimetric content in water but is obtained from the volumes of water and soil. Considering that the density of water is 1Kg / L, the volumetric content in water can be expressed according to the following equation:

$$(5) \theta_v = \frac{V_w}{V_T}$$

- V_w = volume of water

The saturation ratio is the ratio of the pore volume occupied by water to the total porosity.

$$(6) R_s = \frac{V_w}{V_p} = \frac{\theta_v}{n}$$

- **Air saturated porosity:** it is the volume of air that the sample contains in relation to the total volume of the soil.

$$(7) \epsilon = \frac{V_a}{V_T}$$

- V_a = Total volume of air

2.1.5 Soil organic matter

Soil organic matter (SOM) is the name given to organic molecules that result from the partial disintegration and decomposition of plant remains (and, to a lesser extent, animals) due to the action, fundamentally, of soil microorganisms, as well as the compounds synthesized by these microorganisms during the decomposition process (Ambiente et al., 2015). It varies between 0.1% and 10% and most of its content (around 70%) is made up of humic substances, which represent the most stable fraction of the organic matter contained in the soil.

Humic substances are classified according to their solubility in acidic and basic media:

- **Fulvic acids:** They are a set of solid or semi-solid compounds of light colors and colloidal in nature, easily dispersible in water and soluble in acids and bases. Aromatic compounds account for approximately 25% of their composition, and are smaller molecules with greater polarity than humic acids. The presence of charge in these molecules generates repulsion forces so that the structures of fulvic acids are more linear than those of humic acids. (Ambiente et al., 2015)
- **Humic acids:** They are compounds that appear as dark brown amorphous solids, soluble in bases but not in acids. They are composed of 35% -45% of aromatic groups, mainly single-ring structures with carboxyl (-COOH) and hydroxyl (-OH) groups. These aromatic structures appear linked to each other by long aliphatic chains, also with carboxyl groups, and which cause the hydrophobicity of humic acids. (Ambiente et al., 2015)
- **Humines:** They constitute a group of substances relatively different from each other, whose origin can be inherited (inherited humine) or of new formation. The inherited humines are retained in the aggregates of the heavy fraction of the soil, they occur in soils with little biodegradable vegetation and they are linked to the clayey fraction of the soils. Neof ormation humins include removable insolubilization humins, comparable in nature to humic and fulvic acids, but irreversibly linked to the mineral fraction. (Ambiente et al., 2015)

2.1.6 Organic carbon fraction

It is the quantification of organic matter capable of absorbing pollutants as in NAPL . It is normally assumed to be 58% of the total organic matter in the soil.

As indicative ranges, a fertile soil is considered to have 6% SOM, so $6 \times 0.58 = 3.42\%$ ($foc = 0.0342 \text{ g / g}$). In deeper soils, the organic matter content is much lower. A conservative value of foc is 0.001(Ambiente et al., 2015)

2.2 NAPL

NAPLs (Non-aqueous phase liquids) are immiscible hydrocarbons in the subsoil, in the non-aqueous phase, and although their immiscibility characteristic has been mentioned, they are capable of mixing with water at very slow speeds.

Considering the density of the NAPLs they can be classified into:

- LNAPL : acronym for light NAPL , that is, a light NAPL , which has less density than water. Among them the most prominent are gasoline, diesel, and crude.
- DNAPL : Dense NAPL , are named for having a higher density than water. A good example is chlorinated solvents, or one-component products.

The soils have a great capacity of retention of non-organic hydrocarbons, therefore, when NAPL is leaking, the need to characterize the way in which they are moving and associating becomes evident.

The problem that exists with NAPLs is that many of them, such as benzene, are carcinogens, however, the possible leakage of oil into the soil can produce ecotoxicological problems, due to the toxic, carcinogenic and mutagenic potential.

2.2.1 DNAPL

The discharge of large quantities of DNAPL means that when infiltrating the subsoil, they reach stability, placing themselves in impermeable layers. This fact, together with its great toxicity and its high migratory potential, makes it of utmost importance to characterize its leaks.

2.2.1.1 Factors that control infiltration capacity

2.2.1.1.1 Relative mobility coefficient

One of the essential factors to control their mobility is the relative mobility coefficient, which measures the infiltration capacity of a liquid. The greater its relative mobility, the farther the liquid can penetrate into the subsoil (Agència de Residus de Catalunya, nd)

$$(8) M_r = \frac{\rho_a \mu_w}{\rho_w \mu_d}$$

- ρ_a = DNAPL density
- μ_d = Dynamic DNAPL viscosity

- The density of DNAPL varies between 1.01 and 1.65.
- Dynamic viscosity measures the internal friction derived from the molecular cohesion that takes place in a fluid when this fluid tries to resist its own movement.

2.2.1.1.2 Solubility

Solubility is defined as the maximum concentration of DNAPL liquid that can be dissolved in water at a given temperature and pressure (typically 20 ° C and 1 atm). The usual symbol for solubility is the capital letter S and is expressed in units of mass per unit volume of water, that is, in units of concentration: milligrams per liter (mg / l) or micrograms per liter ($\mu\text{g} / \text{l}$), where $1\text{mg} / \text{l} = 103 \mu\text{g} / \text{l}$ (Agència de Residus de Catalunya, nd).

In general terms, it is understood that a concentration of more than 1% with respect to the effective solubility (that is, the actual solubility), may indicate the possible existence of a DNAPL in the subsoil.

The mixture of chemical substances in the same compound makes the solubility of each element of the compound less than the solubility of each chemical element analyzed independently. This, together with the heterogeneity of the medium, the transport and degradation processes, the erratic distribution of the DNAPL phase in the focus of contamination, the mixing of groundwater in the observation wells and the slow dissolution of the DNAPL phase in water (Agència de Residus de Catalunya, nd) makes a difference between solubility and measured concentrations in the field.

2.2.1.1.3 Vapor pressure

Vapor pressure describes the interaction between soil vapors and the liquid phase DNAPL in an equilibrium state. Vapor pressure is defined as the partial pressure of the gas that is obtained by volatilizing a DNAPL liquid under equilibrium conditions at a certain temperature (20 ° C). The usual symbol for vapor pressure is P_{vap} and is expressed in pressure units, usually atmospheres (atm) or millimeters of mercury (mm Hg).

The units ratio is $1 \text{ mm Hg} = 0.00131578947 \text{ atm}$. In general, vapor pressure values greater than 1 mm Hg (0.0013 atm) indicate that the DNAPL liquid is volatile and therefore can generate vapors. (Agència de Residus de Catalunya, nd)

When a liquid is made up of enough compounds, each one of them volatilizes in order to reach its characteristic partial pressure, which can be calculated using Raoult's law:

$$(9) p = p_a(1 - x_b) + p_b x_b$$

This law allows, since the sum of mole fractions must always be equal to unity, to express the total pressure as a function of the partial pressures.

2.2.1.1.4 Henry's constant

Henry's constant describes the equilibrium interaction between the vapors of the gas phase and the aqueous phase. It is defined as the relationship between the partial pressure of the chemical in the gas phase (atm) and its concentration in water (mol / m³) in a state of equilibrium and at a certain temperature (25 ° C). The usual symbol for the Henry constant is K_H and is expressed in units of atm · m³ / mol. Values of the Henry's constant greater than $10^{-5} \text{ atm} \cdot \text{m}^3 / \text{mol}$ indicate that the compound is volatile once dissolved in water. (Agència de Residus de Catalunya, nd)

This is the relationship between DNAPL dissolved in the aqueous phase and that present in the gas phase.

$$(10) K_h = \frac{p_g}{c_w}$$

- p_g = partial pressure of the gas
- c_w = concentration in water

It can also be used to express the relationship between the gaseous concentration of a compound and its concentration in water, in this case it is dimensionless, and it is simply the quotient between the two quantities

2.2.1.1.5 Distribution coefficient

The distribution coefficient describes the interaction between the DNAPL liquid dissolved in water and the solid phase of the subsoil in a state of equilibrium. It is defined as the ratio between the total amount of DNAPL that is retained in the solid soil material (mg / kg) and the concentration of DNAPL dissolved in water (mg / l). The usual symbol for the distribution coefficient is K_d and is expressed in units of l / kg (liters per kilogram of soil) or ml / g (milliliters per gram of soil). (Agència de Residus de Catalunya, nd)

$$(11) K_d = \frac{C_s}{C_w}$$

- C_s = amount retained per unit mass of soil
- C_w = concentration in water

Most pollutants in the subsoil are adsorbed by the hydrophobic effect, that is, the effect that establishes a tendency for water (polar) molecules to exclude nonpolar molecules. This fact leads to the adsorption of organic compounds (apolar) in the organic matter (also apolar) that is part of the subsoil.

As the amount of organic matter in the soil increases, the effect becomes more evident and becomes more important, so the distribution or distribution coefficient has been directly related to the amount of organic matter existing in the soil.

$$(12) K_d = f_{oc} K_{oc}$$

- f_{oc} = organic matter fraction
- k_{oc} = organic carbon partition coefficient

2.2.2 LNAPL

Specifically, LNAPLs are the subject of study and concern because most of them are explosive and flammable compounds. In solution they form a cloud of soluble components that move through the subsoil, and their very intrusion into the vapor materializes in the form of volatile components.

There is no clear regulatory framework for LNAPL-contaminated sites where most of the hydraulically recoverable LNAPL has already been removed. For LUST sites (underground storage tanks), the 40 CFR 280.64 standard requires: "... remove spilled (free) product to the maximum level possible by the agency in charge ..." (Technology et al., Nd).

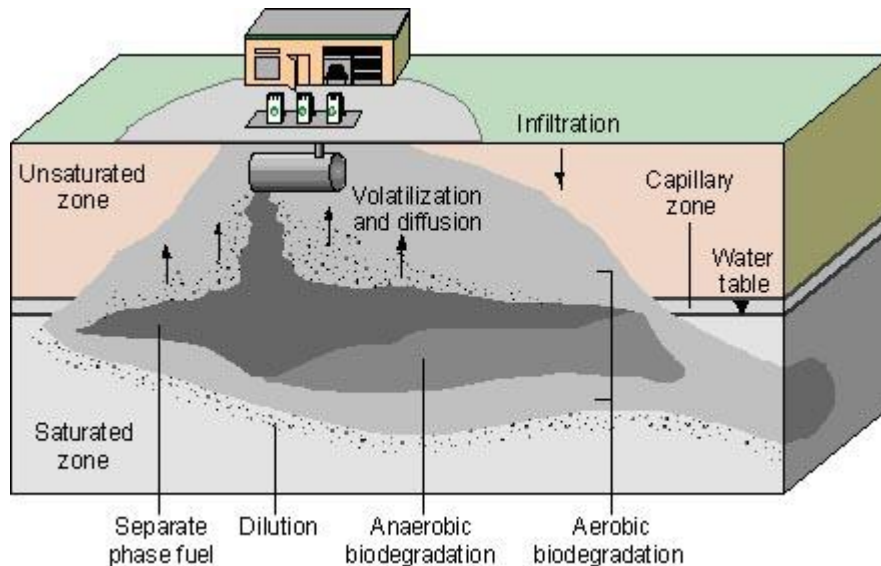


Figure 2- LNAPL leak model. (Source: <https://www.ienvi.com.au/INAPL/>)

For an LNAPL to infiltrate the soil, it must displace the fluid that is occupying the soil pore, so it is much easier for the LNAPL to displace air than to displace water. To enter a pore with water, it has to overcome the inlet pressure, so it can be distributed vertically and migrate laterally.

The difficulty for an LNAPL to enter a pore with water can be characterized with the expression of the load:

$$(13) \quad h_{Nc} = \frac{2\sigma \cos\theta}{r(\rho_w - \rho_o)g}$$

Thus, the more difficult it is for the LNAPL to enter the pore. From the point of view of the type of soil that presents more difficulties or facilities: h_{Nc}

- **Clay-** Water becomes trapped in your pores with force, making it relatively difficult for an LNAPL to displace water from its pores.
- **Sand:** Holds water out of your pores with much less force, making it easier for an LNAPL to displace water.

2.2.2.1 LNAPL mobility

To consider a potentially mobile LNAPL, it must have exceeded the residual saturation, that is, the saturation reached after an infinite number of displacement fluid pore volumes have flowed through a particular portion of the reservoir rock.

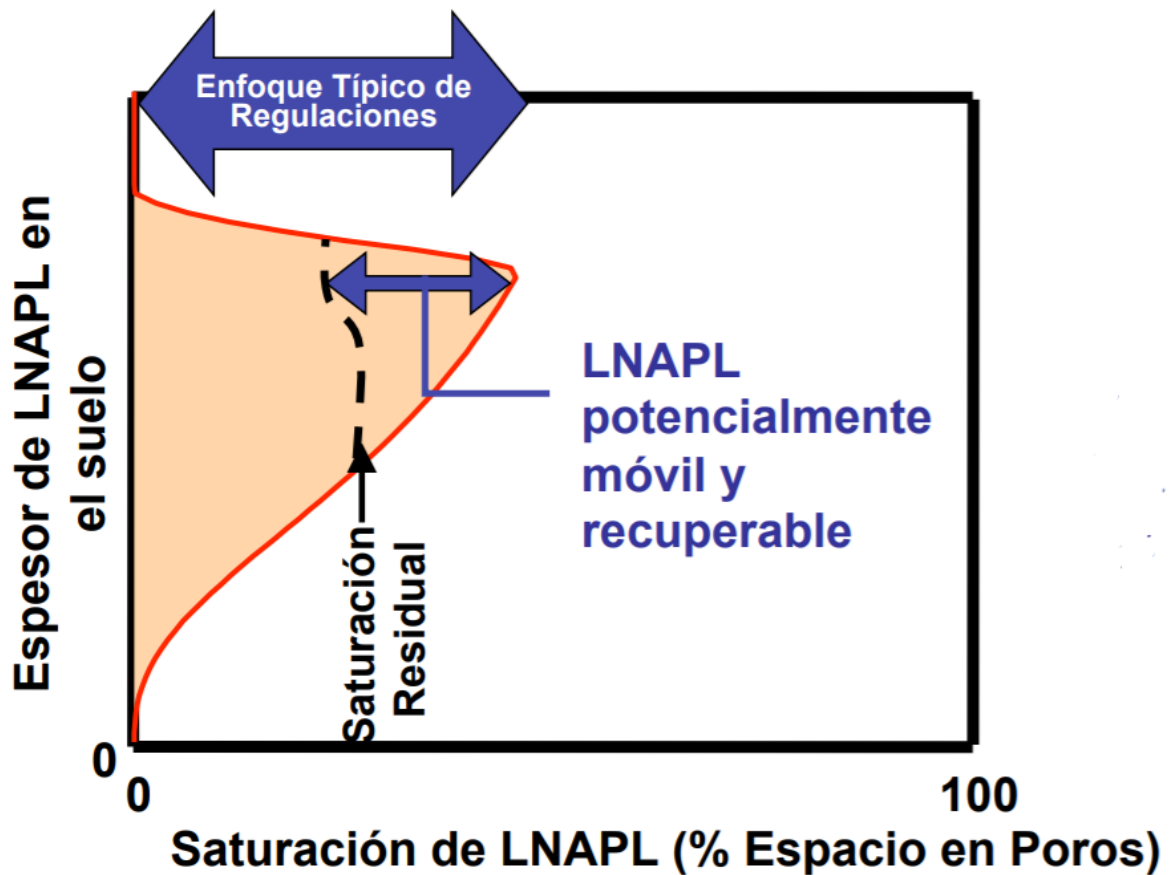


Figure 3- LNAPL thickness depending on the LNAPL saturation. (Source: LNAPL Part 1: A deeper understanding of the behavior of LNAPL in the subsoil)

Thus, the coexisting water and LNAPL in the subsoil pores will be governed by Darcy's law, but its application must be considered independently for each fluid.

For LNAPL, the following expression of Darcy's law is used as a function of its conductivity.

$$(14) K_o = \frac{k_{ro} K_w \rho_o \mu_w}{\rho_w \mu_o}$$

- k_{ro} = relative permeability of LNAPL
- K_w = saturated hydraulic conductivity
- ρ_o = LNAPL density
- μ_w = viscosity of water
- ρ_w = water density
- μ_o = LNAPL viscosity

The following conclusions can be obtained from this expression:

- For a certain LNAPL saturation, a higher LNAPL viscosity implies a lower LNAPL conductivity.
- For a given LNAPL viscosity, a higher LNAPL saturation implies a lower LNAPL viscosity.

There are other interesting considerations to be made regarding the mobility of LNAPL s, and observation experience in the field:

- Initially an LNAPL can be spread faster than groundwater velocity by large hydraulic loads at the time of the spill.(Technology et al., Nd)
- An LNAPL can spread in the opposite direction to that of water flow (radial spread). (Technology et al., Nd)
- After an LNAPL spill stops, the LNAPL body reaches a stable configuration in a relatively short time.(Technology et al., n.d.)

2.3 PHASE DISTRIBUTION ESTIMATION

To make an estimate of the distribution in phases, we start from the mass balance equation.

$$(15) M_t = M_w + M_A + M_s + M_N$$

- M_t = total mass of pollutant
- M_w = mass of pollutant contained in aqueous phase
- M_A = mass of pollutant contained in the gas phase
- M_s = mass of pollutant contained in solid phase
- M_N = mass of free phase pollutant

It is based on certain data such as the concentration of a pollutant in the soil, it is possible to carry out this mass balance. It is of fundamental importance to be able to obtain it, in order to characterize the free phase that is generated in saturated soil by filtering these hydrocarbons.

The development of equations, which allows reaching the mass balance, to characterize the distribution, will be described below, in the study of the practical case that this work occupies.

3. METHODS

In this chapter, the procedure to calculate the NAPL (Non-aqueous phase liquids) free phase will be described as the practical case of the project.

3.1 NAPL FREE PHASE CALCULATION

The following method describes the execution and order of operations necessary to calculate the NAPL free phase, for a given pollutant based on data and under specific conditions.

The necessary data are as follows and the subscript "i" refers to a hypothetical pollutant "i", being thus I have these data from the same pollutant:

- MW_i molecular weight (g / mol).
- Solubility limit Yes (mg / L)
- Vapor pressure VP_i (mmHg)
- Henry H_i's law constant (dimensionless)
- K_{oc} organic carbon partition coefficient (L / kg)
- Density d_i (kg / L)
- Contaminant mass in the C_{ti} sample (mg / Kg)
- Total porosity n_t (%)
- Volumetric water content w (%)
- Volumetric air content a (%)
- Fraction of organic carbon F_{oc} (%)
- Bulk density P_b (kg / L)
- Gas constant R_i (dimensionless)
- Temperature T (K)
- Pressure (mmHg)

These are the necessary data to be able to start applying the method since the algorithm will ask for data successively.

It is worth commenting that the case of the apparent density P_b, although it can be calculated theoretically by means of an equation, has been assumed a value, calculated experimentally, which in the practical case will be described as an average of values provided by the professor.

3.1.1 Molar fraction

First thing about the procedure is to calculate the molar fraction of the pollutant according to the whole pollutant mass:

$$(16) W_{ti} = C_{ti} / M_t$$

W_{ti} = mass% for compound i (mg / mg)

C_{ti} = Mass of pollutant in the soil (mg / kg)

M_t = $\sum C_{ti}$, sum of the masses of all pollutants (mg / kg)

$$(17) M_{ci} = W_{ti} / MW_i$$

M_{ci} = Molar mass of compound i (mol / g)

MW_i = Molecular weight of compound i (g / mol)

$$(18) X_i = M_{ci} / \sum M_{ci}$$

X_i = Molar fraction of compound i (%)

$\sum M_{ci}$ = Sum of all the molar masses of the compounds present (mol / g)

M_{ci} = Molar mass of compound i (mol / g)

3.1.2 Density of the mix

The density of the mix of pollutants need to be calculated so that some parameters can be calculated afterwards:

$$(19) d = \sum (X_i \times d_i)$$

d = Density of the mixture (kg / L)

X_i = Molar fraction of compound i (%)

d_i = Density of compound i (kg / L)

3.1.3 Effective solubility

The effective solubility is used to calculate C_{sat} . C_{sat} is a theoretical limit that indicates when the concentration in soil-pore air and soil pore water and the sorptive limits of the soil particles have been reached

$$(20) SE_i = X_i \times S_i$$

SE_i = Effective solubility of compound i in the mixture (mg / L)

X_i = Molar fraction of compound i (%)

S_i = pure aqueous phase solubility for compound i (mg / L)

$$(21) S_t = \sum SE_i$$

S_t = Solubility of the mixture (mg / L)

3.1.4 Effective vapor pressure

The total vapor pressure of the mixture and the effective vapor pressure of each compound in the mixture is calculated.

$$(22) VPE_i = (X_i \times VPI)$$

VPE_i = effective vapor pressure for compound i in the mixture (mmHg)

X_i = Molar fraction of compound i (%)

VPI = Pure vapor pressure of pollutant i (mmHg)

$$(23) VP_t = \sum VPE_i$$

VP_t = Total vapor pressure for the mixture (mmHg)

3.1.5 Theoretical concentration in aqueous pore

A theoretical pore-water concentration (IC_{wi}) is calculated. This value is compared to the effective solubility (SE_i) or the solubility limit (S). The software selects the lesser of the effective solubility, solubility limit, or the pore-water concentration to calculate media phase concentrations and masses.

$$(24) IC_{wi} = (C_{ti} \times P_b) / (K_{di} \times P_b + \sum w + H'_i \times \sum a)$$

IC_{wi} = Theoretical concentration in aqueous pore for compound i (mg / L)

C_{ti} = Mass of pollutant i in the sample (mg / kg)

P_b = Bulk density (kg / L)

K_{di} = Soil-water partition coefficient (f_{oc} x K_{oci}) (L / kg)

K_{oc} = Soil-organic carbon partition coefficient (L / kg)

w = percentage of pores occupied by water (%)

H'_i = Henry's constant for compound i (dimensionless)

a = percentage of pores occupied by air (%)

3.1.6 Saturation limit for soil

Csat is calculated using the effective solubility, solubility limit, or pore-water concentration for each compound. This would be the most important parameter to quantify whether there is NAPL free phase

$$(25) C_{sati} = (SE_i / P_b) * (K_{di} * P_b + \omega + H'_i * a)$$

Csati = Limit of soil saturation for compound i (mg / kg)

SEi = Effective solubility for compound i (mg / L)

Pb = Bulk density (kg / L)

Kdi = Soil-water partition coefficient (foc x Koci) (L / kg)

w = pores occupied by water (%)

H'i = Henry's constant for compound i (dimensionless)

a = pores occupied by air (%)

3.1.7 Volumetric saturation limit

Volumetric saturation limit is calculated as the saturation limit, multiplied by the wet bulk density.

$$(26) C_{coni} = C_{sati} \times P_b$$

Cconi = Volumetric Soil Saturation Limit (mg / L)

Csati = Limit of soil saturation for compound i (mg / kg)

Pb = Bulk density (kg / L)

3.1.8 Concentration in aqueous pore

The equilibrium concentration for each compound is calculated for soil pore-water and soil pore-vapor with either the effective solubility limit, or the theoretical porewater concentration of each compound (ICwi). The software automatically uses a logic function to select the lower of the values S, SEi or ICwi

$$(27) C_{wi} = IC_{wi} \text{ or } S_i \text{ or } SE_i$$

ICwi = Theoretical concentration in aqueous pore for compound i (mg / L)

SEi = Effective solubility for compound i (mg / L)

Si = Solubility limit for compound i (mg / L)

Cwi is calculated with a logical test, so if:

- $X_i = 1$, S_i is chosen.
- $IC_{wi} > SE_i$, SE_i will be selected as the soil-pore water concentration.

3.1.9 Concentration in steam pore

Concentration in steam pore can be easily calculated with the previous concentration and the Henry's constant for each compound.

$$(28) C_{ai} = C_{wi} \times H'_i$$

C_{ai} = Vapor pore concentration for compound i (mg / L)

C_{wi} = concentration in aqueous pore for compound i (mg / L)

H'_i = Henry's constant for compound i (dimensionless)

3.1.10 Volumetric concentration in steam pore

The previous results lead to the calculation of the volumetric concentration in steam pore, that would be the previous parameter multiplied by the universal gas constant, a conversion factor, the temperature and divided by the pressure and the molecular weight of compound i.

$$(29) C_{avi} = C_{ai} \times R \times CF \times T / (P \times MW_i)$$

C_{avi} = Volumetric concentration in vapor pore for compound i (ppmv)

C_{ai} = Vapor pore concentration for compound i (mg / L)

R = Universal gas constant (8.31451 Pa m³ / (K * mol))

CF = Conversion factor (1E + 6 mg / L)

T = Temperature (K)

P = Pressure (Pa)

MW_i = Molecular weight of compound i (g / mol)

3.1.11 Concentration absorbed by the solid

This concentration represents the concentration of pollutant i , sorbed by the solid matter in the soil.

$$(30) C_{si} = K_{di} \times C_{wi}$$

C_{si} = Concentration absorbed by the solid of compound i (mg / kg)

K_{di} = Soil-water partition coefficient ($f_{oc} \times K_{oc}$) (L / kg)

C_{wi} = concentration in aqueous pore for compound i (mg / L)

3.1.12 Mass of contaminant in aqueous pore

The calculation of the mass of pollutant in the aqueous pore is calculated with the next formula, knowing the concentration in aqueous pore (C_{wi})

$$(31) M_{wi} = (C_{wi} \times \sum w) / P_b$$

M_{wi} = Mass of contaminant in solid for compound i (mg / kg)

C_{wi} = concentration in aqueous pore for compound i (mg / L)

w = pores occupied by water (%)

P_b = Bulk density (kg / L)

3.1.13 Mass of pollutant in air pore

The calculation of the mass of pollutant in the air pore is calculated with the next formula, knowing the concentration in aqueous pore (C_{wi})

$$(32) M_{ai} = (C_{wi} \cdot a \cdot H'_i) / P_b$$

M_{ai} = mass of pollutant in air pore for compound i (mg / kg)

C_{wi} = concentration in aqueous pore for compound i (mg / L)

a = pores occupied by air (%)

H'_i = Henry's constant for compound i (dimensionless)

P_b = Bulk density (kg / L)

3.1.14 Absorbed mass of pollutant in the soil

The last mass calculation, that corresponds with the amount of pollutant absorbed by the organic matter in the soil.

$$(33) M_{si} = C_{wi} \times K_{di}$$

M_{si} = Mass absorbed by the soil of compound i (mg / kg)

C_{wi} = concentration in aqueous pore for compound i (mg / L)

K_{di} = Soil-water partition coefficient ($f_{oc} \times K_{oci}$) (L / kg)

3.1.15 NAPL mass for compound

The most important equation of the model, it describes and gives the necessary condition for the model to have formation of NAPL free phase. If this condition ($C_{ti} > C_{sati}$) does not verify, there will not be NAPL free phase.

$$(34) \text{ NAPL } i = C_{ti} - C_{sati}, \text{ if } C_{ti} > C_{sati}$$

NAPL i = NAPL mass of compound i in the sample (mg / kg)

C_{ti} = Mass concentration of pollutant i (mg / kg)

C_{sati} = Limit of soil saturation for compound i (mg / kg)

3.1.16 Mass balance for the compound

The mass per unit volume (M_{ti}) for each compound is calculated for soil porewater, soil pore-vapor, soil pore-water, and NAPL, if any, based upon the C_{wi} .

$$(35) M_{ti} = (\sum M_{ai} + M_{wi} + M_{si} + \text{MNAPL } i)$$

M_{ti} = Total mass of pollutant i (mg / kg)

M_{ai} = mass of pollutant in air pore for compound i (mg / kg)

M_{wi} = Mass of solid contaminant for compound i (mg / kg)

M_{si} = Mass absorbed by the soil of compound i (mg / kg)

$\text{MNAPL } i$ = NAPL mass of compound i in the sample (mg / kg)

3.1.17 Total mass of NAPL in the sample

It quantify the amount of NAPL free phase, so the sum of the NAPL mass for each compound.

$$(36) \text{ NAPL } t = \sum \text{NAPL } i$$

NAPL t = Total mass of NAPL in the sample (mg / kg)

NAPL i = NAPL mass of compound i in the sample (mg / kg)

3.1.18 NAPL residual saturation percentage

The Percent Residual NAPL Saturation (S_r) is calculated. This is in the form of a volumetric calculation of either the volume of NAPL occupying air-porosity volume in unsaturated samples or the volume of NAPL occupying total porosity volume for saturated samples.

$$(37) S_r = (\text{NAPL } t \times P_b / d) / n_t$$

S_r = NAPL residual saturation (%)

NAPL t = Total mass of NAPL in the sample (mg / kg)

P_b = Bulk density (kg / L)

d = Density of the NAPL mixture (kg / L)

n_t = Total porosity of the soil (%)

3.1.19 Volumetric content of NAPL in the sample

The volumetric content of NAPL in the sample is calculated with the total mass of NAPL in the sample, the wet bulk density and the density of NAPL mixture.

$$(38) N_v = (\text{NAPL } t \times P_b) / P_{\text{NAPL}}$$

N_v = NAPL Volume (L / L)

NAPL t = Total mass of NAPL in the sample (mg / kg)

P_b = Bulk density (kg / L)

P_{NAPL} = Density of the NAPL mixture (mg / L)

3.1.20 Volumetric content of NAPL in air

The volumetric content of NAPL in the air is calculated with the percentage of pores occupied by air and the volumetric content of NAPL in the sample.

$$(39) A_v = \sum a \cdot N_v$$

A_v = Air volume in the sample (% , L / L)

a = percentage of pores occupied by air (%)

N_v = NAPL Volume (L / L)

3.1.21 Volumetric water content if the air content were $a = 0$

The volumetric content of NAPL in the water is calculated with the percentage of pores occupied by water and the volumetric content of NAPL in the sample, assuming air content = 0.

$$(40) V_w = \sum w - N_v$$

V_w = Air volume in the sample (% , L / L)

w = pores occupied by water (%)

N_v = NAPL Volume (L / L)

3.1.22 Volumetric content of solid in the sample

The volumetric content of NAPL in the solid is calculated with the percentage of pores occupied by air, water and the volumetric content of NAPL in the sample.

$$(41) S_v = 1 - (\sum w + \sum a + N_v)$$

S_v = Volume of solid (% , L / L)

w = pores occupied by water (%)

a = pores occupied by air (%)

N_v = NAPL Volume (L / L)

4. RESULTS

The aim of the practical case is to evaluate whether a NAPL free phase is obtained

In this chapter the procedure described is evaluated by building a mathematical program using MATLAB.

The programming will be carried out with object-oriented programming, which will allow greater flexibility regarding the execution of functions implemented in the program.

The program will read an Excel file with the data of a pollutant and the study soil in question. Subsequently, and based on these initial data, the necessary parameters will be calculated for the calculation of the NAPL free phase. This calculation will be sequential and will have the same order as in the process described above.

The evaluation of the programme will be executed in two phases:

- Evaluation of two pollutants assuming they are in the soil described above, with the edaphic parameters mentioned.
- Sensitivity analysis of the programme, done by changing some parameter to see whether this variation affects the appearance of NAPL free phase and its quantity.

Every example in the practical case will have the same properties, talking about edaphic and geotechnical properties.

The following geotechnical inputs correspond to a soil examined by the Professor Fernando Barrio in the Universidad Politecnica de Madrid laboratory. He did the experimental analysis to calculate parameters as total porosity, organic carbon fraction and contaminant concentration.

Table 1-Chemical parameters for soil 1

Chemical	Cti (mg/Kg)	nt (%)	sumW (%)	sumA (%)	Foc (%)
Trichloroethylene	100	40	30	10	0,0001
Arochlor 1260	60	40	30	10	0,0001

The data above establish the initial and most important conditions for the study of these two contaminants and the sensitivity analysis, all the study is going to rely on this data.

4.1 EVALUATION OF POLLUTANTS

In this section a soil with two pollutants will be assessed to see if there is free NAPL phase obtained due to the infiltration of hydrocarbons in the soil.

4.1.1 Trichloroethylene

Trichloroethylene breaks down very slowly in soil and water and is removed mainly through evaporation into the air. It is likely to remain in the groundwater for a long time because it cannot evaporate.

4.1.1.1 Previous data

The following data was used for the evaluation of the mathematical model for trichloroethylene.

Table 2-Chemical parameters for Trichloroethylene

Name	Mwi	Si	Vpi	Hi	Koci	di
Trichloroethylene	131.29	1100	74.31	0.3649	126	1.46

In addition to these characteristic data, the following assumptions will be made.

- Pressure = 101325 mmHg
- Temperature = 293.15 K
- Gas constant for pollutant = 8.31451
- Bulk density = 1.8903 kg / L and has been obtained as the arithmetic mean of values resulting from a study carried out by the professor of a similar field, otherwise it would not be a way to calculate it but the experimental way.

With these data it is already possible to run the program and evaluate its performance.

4.1.1.2 Evaluation with the program

By executing the program, the following data is obtained:

Table 3-Obtained data

Xi	1,0000
d	1,4600
Sei	1100,0000
Sti	1100,0000
Vpei	74,3100
Kdi	0,0013
lcwi	430,7181
Csat	255,3875
Cconi	482,7590
Cwi	1100,0000
Cai	401,3900
Cavi	735440000,0000
Csi	1,3860
Mwii	232,7673
Mai	21,2342
Msi	1,3860
NAPL i	'N/A'
Mti	N/A'
NAPL t	N/A'
Sr	N/A'
Nv	'N/A'
Av	'N/A'
Vw	'N/A'
Sv	'N/A'

It should be noted that the molar fraction X_i is the unit because it is the study of a single contaminant in the soil.

When calculating the total mass of NAPL, and the following parameters, N / A is obtained, which means NOT APPLICABLE, this occurs because for that soil and that contaminant it has been obtained:

$$C_{sati} > C_{ti}$$

$$255.3875 > 100$$

This means that the condition that the concentration of that pollutant is bigger than the soil saturation limit for the same pollutant is not fulfilled, therefore there is no free phase.

The following operations are also represented with N / A since it does not make sense to calculate these parameters with the mass of NAPL = 0.

4.1.2 Arochlor 1260

This is a compound formed by Polychlorinated biphenyl mixture with 60% chlorine. It represents, as well as trichloroethylene, an enormous hazard towards the living organisms if free NAPL phase is obtained.

4.1.2.1 Previous data

The data used to run the programme is summed up in the following table.

Table 4-Chemical parameters for Arochlor 1260

Name	Mwi	Si	Vpi	Hi	Koci	di
Arochlor 1260	372	0,0144	0,000011	0,01025	6700000	1,566

4.1.2.2 Evaluation with the program

By introducing the data shown above and run the program, the next results are obtained:

Table 5-Obtained data

Xi	1,0000
d	1.5660
Sei	0.0144
Sti	0.0144
Vpei	1.1000e-05
Kdi	670
lcwi	0.0895
Csat	9.6511
Cconi	18.2434
Cwi	0.0144
Cai	1.4760e-04
Cavi	0.0095
Csi	9.6480
Mwii	0.0030
Mai	7.8083e-06
Msi	9.6480
NAPL i	50.3489
Mti	60
NAPL t	50.3489
Sr	1.5969
Nv	63.8756
Av	'N/A'
Vw	'N/A'
Sv	'N/A'

Arochlor 126 in this soil does not produce NAPL free phase under these conditions, as it is shown in the results.

There is NAPL free phase because $C_{sati} < C_{ti}$, so that the saturation limit concentration for this pollutant under the soil condition exceeds the value of the total concentration of this pollutant in this soil, so free phase is formed, with the following concentration.

$$\text{NAPL} = 50.3489 \text{ mg / kg}$$

4.2 SENSITIVITY ANALYSIS

The aim of this second part of the evaluation of the program is to test how would the appearance of NAPL free phase change by varying some parameter of the model.

The test is going to be separated in two phases regarding the nature of the parameters studied in the test:

- Edaphic parameters: in which we will study the variation of the most important parameters that describe the characteristics of the study terrain.
- Chemical parameters: in which we will study the variation of the most important parameters that describe the characteristics of the pollutant.

4.2.1 Edaphic parameters

The method that I am going to follow the test is changing only one parameter and maintaining the rest of them, both edaphic and chemical. The chemical parameters considered are the Trichloroethylene's and the edaphic parameters are the Field 1's.

The parameters needed to be changed are:

- Soil water porosity.
- Soil air porosity.
- Total porosity.

4.2.1.1 Soil water porosity

In order to evaluate the existence of NAPL free phase by varying the proportion of soil water porosity percentage, a new script is going to be made. This scrip is going to be very similar to the NAPL .m but introducing the already done functions that calculate each parameter into the Matlab object, so that a function can be called whenever, no matter what the algorithms order is.

Afterwards the edaphic parameters will stay the same as the previous model, but the NAPL new function that it is going to be called NAPL_sumW (sumW stands for sum of Water filled pores) is going to relay on only one parameter: sumW.

To make the analysis consistent and real, a porosity limit is going to be introduced, and it will be 60% which is a generous limit knowing that the average total porosity is 42% for sandy-clayey soils, but this limit is introduced just to magnify the range of values suitable for the appearance of NAPL free phase so that the model can be tested.


```

function obj = NAPL_sumW(sumW)
    % Definición del objeto

    obj.Mwi      = xlsread(obj.FileName, 'B2:B2');
    obj.Si       = xlsread(obj.FileName, 'C2:C2');
    obj.Vpi      = xlsread(obj.FileName, 'D2:D2');
    obj.Hi       = xlsread(obj.FileName, 'E2:E2');
    obj.Koci     = xlsread(obj.FileName, 'F2:F2');
    obj.di       = xlsread(obj.FileName, 'G2:G2');
    obj.Cti      = xlsread(obj.FileName, 'H2:H2');
    obj.nt       = 0.6 ;
    obj.sumW     = sumW ;
    obj.sumA     = xlsread(obj.FileName, 'K2:K2');
    obj.Foc      = xlsread(obj.FileName, 'L2:L2');

    obj.Mt       = obj.getMt;
    obj.Xi       = obj.getXi;
    obj.d        = obj.getd;
    obj.Sei      = obj.getSei ;
    obj.d        = obj.getd;
    obj.Sti      = obj.getSti;
    obj.Vpei     = obj.getVpei;
    obj.Kdi      = obj.getKdi;
    obj.ICwi     = obj.getICwi;

```

Figure 4-Matlab code.

In the figure above, the new function is written, and as it is seen in the code, its only variable is sumW. The total porosity value is written as an constant number, and the rest of the values are taken from the Excel document.

The functions that allow the new code to interact in different order are also shown in Figure 4.

Construction of a new script to evaluate the appearance of NAPL

Once the new code is implemented, a new script in which the values of the new variable sumW should change has to be written. This new script is going to be called TEST-sumW

To construct this new script, I am going to change the values of the new variable and then evaluate the programme NAPL_sumW to see whether or not there is NAPL free phase.

```

Filename = 'parametrosentrada.xlsx';
Mess= 'N/A';
sumW=0;
ex= NAPL_sumW(sumW);
max.sumW= ex.nt - ex.sumA;

f = NAPL_sumW (sumW);
f.NAPLi;
while sumW < max.sumW
    sumW = sumW + 0.05;
    f = NAPL_sumW (sumW);
    Res = f.NAPLi
    sumW
end

```

Figure 5- Test_sumW.

In the figure above the Matlab code is shown and it works like as follows:

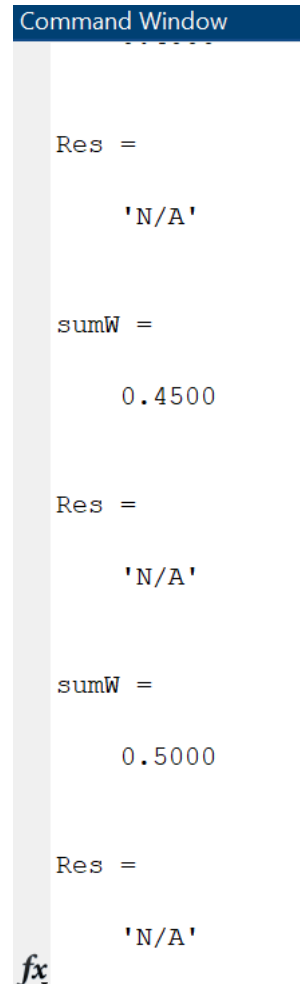
1. The filename 'parametrosentrada.xlsx' is called to have the parameters (except sumW and total porosity (nt)) in the test script.
2. Message 'N/A' is introduced. As mentioned before, it is the message that the programme gives when there is not NAPL free phase.
3. The new variable sumW starts with the value 0.
4. A new maximum is defined for variable sumW which corresponds with the difference between the total porosity and the air-filled porosity. Otherwise it will have no sense, as the total porosity is equal to the soil water porosity plus the air-filled porosity.
5. The function NAPL_sumW is called.
6. The result of the variable NAPL i which tells whether there is NAPL free phase and quantifies it is saved in the program.
7. A condition is established: while sumW is less than the maximum of sumW (variable calculated above), a value is added to sumW to evaluate NAPL_sumW again and see whether there is NAPL free phase by saving the variable NAPL i.
8. When sumW plus sumA (Soil air porosity) is equal to total porosity, the programme stops.

Results

The program runs and outputs some results:

- If there is NAPL free phase and it quantifies it.
- It prints the value of sumW associated with the appearance of NAPL free phase.

By executing the program, it is shown that for Trichloroethylene and Field 1 soil, there is no NAPL free phase even with the variation of the parameter sumW



```
Command Window

Res =

    'N/A'

sumW =

    0.4500

Res =

    'N/A'

sumW =

    0.5000

Res =

    'N/A'
```

Figure 6-Command window Matlab.

In the figure 6 it is shown the last two iterations that the program made before reaching the maximum sumW limit. The variable Res prints the result of NAPL i (the variable that tells us the quantification of NAPL free phase) and it is 'N/A' which means not applicable, as it does not exist.

4.2.1.2 Soil air porosity

In order to evaluate the existence of NAPL free phase by varying parameters percentage, a new script is made, very similar to the NAPL_sumW but making the variable sumW permanent, and the variable sumA (Soil air porosity) changeable. The total porosity remains the same as for the previous test. So the edaphic parameters will stay the same as the previous model, but the NAPL new function that it is going to be called NAPL_sumA is going to depend on one parameter: sumA.

```
function obj = NAPL_sumA(sumA)
    % Definición del objeto

    obj.Mwi      = xlsread(obj.FileName, 'B2:B2');
    obj.Si       = xlsread(obj.FileName, 'C2:C2');
    obj.Vpi      = xlsread(obj.FileName, 'D2:D2');
    obj.Hi       = xlsread(obj.FileName, 'E2:E2');
    obj.Koci     = xlsread(obj.FileName, 'F2:F2');
    obj.di       = xlsread(obj.FileName, 'G2:G2');
    obj.Cti      = xlsread(obj.FileName, 'H2:H2');
    obj.nt       = 0.6 ;
    obj.sumW     = xlsread(obj.FileName, 'J2:J2');
    obj.sumA     = sumA ;
    obj.Foc      = xlsread(obj.FileName, 'L2:L2');

    obj.Mt       = obj.getMt;
    obj.Xi       = obj.getXi;
    obj.d        = obj.getd;
    obj.Sei      = obj.getSei ;
    obj.d        = obj.getd;
    obj.Sti      = obj.getSti;
    obj.Vpei     = obj.getVpei;
    obj.Kdi      = obj.getKdi;
    obj.ICwi     = obj.getICwi;
```

Figure 7-Matlab NAPL_sumA code

In the figure 7, the new function is written, and as it is seen in the code, its only variable is sumA.

The total porosity value is written as a constant number, and the rest of the values are taken from the Excel document.

The functions that allow the new code to interact in different order are also shown in Figure 7.

Construction of a new script to evaluate the appearance of NAPL

Once the new code is implemented, a new script in which the values of the new variable sumA should change has to be written. This new script is going to be called TEST_sumA

To construct this new script, the values of the new variable are changed and then evaluated by the program NAPL_sumW to see whether there is NAPL free phase.

```
Filename = 'parametrosentrada.xlsx';  
Mess= 'N/A';  
sumA=0;  
ex= NAPL_sumA(sumA);  
max.sumA= ex.nt - ex.sumW;
```

```
f = NAPL_sumA (sumA);  
f.NAPLi;  
while sumA < max.sumA  
    sumA = sumA + 0.05;  
    f = NAPL_sumA (sumA);  
    Res = f.NAPLi  
    sumA  
end
```

Figure 8-Matlab code TEST_sumA

In the figure 8 the Matlab code is shown and it works like that:

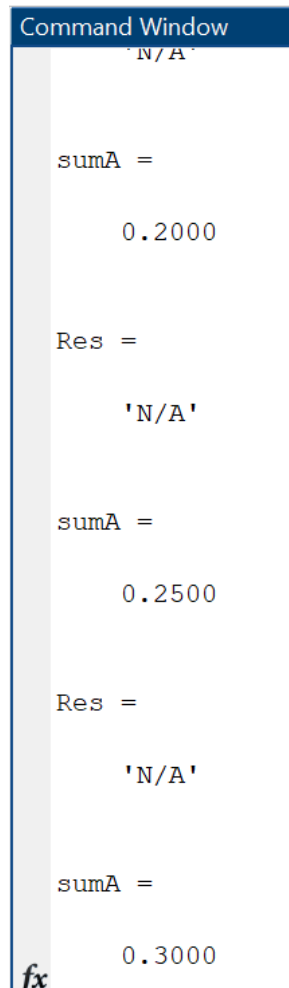
1. The filename 'parametrosentrada.xlsx' is called to have the parameters (except sumA and total porosity (nt)) in the test script.
2. Message 'N/A' is introduced. As mentioned before, it is the message that the programme gives when there is not NAPL free phase.
3. The new variable sumA starts with the value 0.
4. A new maximum is defined for variable sumA which corresponds with the difference between the total porosity and the soil water porosity. Otherwise it will have no sense, as the total porosity is equal to the soil water porosity plus the air-filled porosity.
5. The function NAPL_sumA is called.
6. The result of the variable NAPL i which tells whether there is NAPL free phase and quantifies it is saved in the program.
7. A condition is established: while sumA is less than the maximum of sumA (variable calculated above), a value is added to sumA to evaluate NAPL_sumA again and see whether there is NAPL free phase by saving the variable NAPL i.
8. When sumA plus sumW (soil water porosity) is equal to total porosity, the programme stops.

Results

The program runs and it outputs the results:

- If there is NAPL free phase and it quantifies it.
- It is going to print the value of sumA associated with the appearance of NAPL free phase.

By executing the program, it is shown that for Trichloroethylene and Field 1 soil, there is no NAPL free phase even with the variation of the parameter sumA.



```
Command Window
'N/A'
sumA =
    0.2000
Res =
    'N/A'
sumA =
    0.2500
Res =
    'N/A'
sumA =
    0.3000
fx
```

Figure 9-Command window Matlab

In the figure 9 it is shown the last two iteration that the program made before reaching the maximum sumA limit. The variable Res prints the result of NAPL i (the variable that tells us the quantification of NAPL free phase) and it is 'N/A' which means not applicable, as it does not exist.

4.2.1.3 Total porosity

In order to evaluate the existence of NAPL free phase by varying the total porosity percentage, I am going to do a new script, very similar to the NAPL_sumW and the NAPL_sumA but making the variables sumW and sumA static, and the variable nt (total porosity) changeable. The total porosity is going to change from 0.05 to 0.6.

So the edaphic parameters will stay the same as the previous model, but the NAPL new function that it is going to be called NAPL_nt is going to rely on only one parameter: nt.

```
methods

##### FUNCIÓN PARA OBTENER TODOS LOS DATOS DEL ARCHIVO
function obj = NAPL_nt(nt)
    % Definición del objeto

    obj.Mwi      = xlsread(obj.FileName, 'B2:B2');
    obj.Si       = xlsread(obj.FileName, 'C2:C2');
    obj.Vpi      = xlsread(obj.FileName, 'D2:D2');
    obj.Hi       = xlsread(obj.FileName, 'E2:E2');
    obj.Koci     = xlsread(obj.FileName, 'F2:F2');
    obj.di       = xlsread(obj.FileName, 'G2:G2');
    obj.Cti      = xlsread(obj.FileName, 'H2:H2');
    obj.nt       = nt;
    obj.sumW     = xlsread(obj.FileName, 'K2:K2');
    obj.sumA     = xlsread(obj.FileName, 'K2:K2');
    obj.Foc      = xlsread(obj.FileName, 'L2:L2');

    obj.Mt       = obj.getMt;
    obj.Xi       = obj.getXi;
    obj.d        = obj.getd;
    obj.Sei      = obj.getSei;
    obj.d        = obj.getd;
    obj.Sti      = obj.getSti;
```

Figure 10-NAPL_nt code

In the figure above, the new function is written, and as it is seen in the code, its only variable is nt. The functions that allow the new code to interact in different order are also shown in Figure 10.

Construction of a new script to evaluate the appearance of NAPL

Once the new code is implemented, a new script in which the values of the new variable nt should change has to be written. This new script is going to be called TEST_nt.

To construct this new script, the values of the new variable should be changed and then evaluated by the programme NAPL_nt to see whether there is NAPL free phase.

```
Filename = 'parametrosentrada.xlsx';
Mess= 'N/A';
nt=0.05;

f = NAPL_nt (nt);
f.NAPLi;
while nt <= 0.6
    nt = nt + 0.05;
    f = NAPL_nt (nt);
    Res = f.NAPLi
    nt
end
```

Figure 11-TEST_nt

In the figure above the Matlab code is shown and it works like as follows:

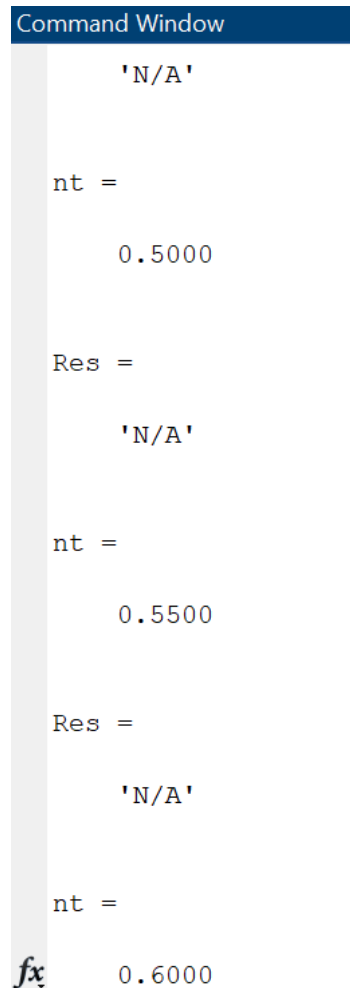
1. The filename 'parametrosentrada.xlsx' is called to have the parameters (except nt) in the test script.
2. Message 'N/A' is introduced. As mentioned before, it is the message that the programme gives when there is not NAPL free phase.
3. The new variable nt starts with the value 0.05
4. The function NAPL_nt is called.
5. The result of the variable NAPL i which tells whether there is NAPL free phase and quantifies it is saved in the program.
6. A condition is established: while nt is less than the maximum of nt (0.6), a value is added to nt to evaluate NAPL_nt again and see whether there is NAPL free phase by saving the variable NAPL i.
7. When nt is equal to total porosity maximum (0.6), the programme stops.

Results

The program runs and outputs the results:

- It tells if there is NAPL free phase and it quantifies it.
- It prints the value of nt associated with the appearance of NAPL free phase.

By executing the program, it is shown that for Trichloroethylene and Field 1 soil, there is no NAPL free phase even with the variation of the parameter nt .



```
Command Window
'N/A'
nt =
    0.5000
Res =
'N/A'
nt =
    0.5500
Res =
'N/A'
nt =
    0.6000
fx
```

Figure 12-Command window Matlab

In the figure 12 it is shown the last two iteration that the program made before reaching the maximum nt limit. The variable Res prints the result of NAPL i (the variable that tells us the quantification of NAPL free phase) and it is 'N/A' which means not applicable, as it does not exist.

4.2.2 Chemical parameters

In this chapter, Cti (pollutant concentration in soil) will be evaluated. The procedures in the construction of the analysis will be different than the ones for the edaphic parameters, as an evaluation and finding for the first value for each parameter that verifies the fact that NAPL free phase is obtained will be made.

4.2.2.1 Cti-Pollutant concentration in soil

Maybe the most interesting part of the sensitivity analysis, in which the evaluation of the appearance of NAPL free phase relating on the concentration is going to be done.

The implemented code is going to be slightly different from the ones before that were evaluating the appearance of NAPL by varying some parameter in certain range, but whit these analysis I will find the first value that verifies the condition of the appearance of NAPL free phase.

The first part is to construct the NAPL_Cti code which is similar to the NAPL_sumW and these ones, but in this one the only variable of the model will be Cti. Regarding the previous models, I am going to introduce the functions in the definition of the object to improve the speed and the calculations.

```
##### FUNCIÓN PARA OBTENER TODOS LOS DATOS DEL ARCHIVO
function obj = NAPL_Cti(Cti)
    % Definición del objeto

    obj.Mwi      = xlsread(obj.FileName, 'B2:B2');
    obj.Si       = xlsread(obj.FileName, 'C2:C2');
    obj.Vpi      = xlsread(obj.FileName, 'D2:D2');
    obj.Hi       = xlsread(obj.FileName, 'E2:E2');
    obj.Koci     = xlsread(obj.FileName, 'F2:F2');
    obj.di       = xlsread(obj.FileName, 'G2:G2');
    obj.Cti      = Cti; %Masa de contaminante en m
    obj.nt       = xlsread(obj.FileName, 'I2:I2');
    obj.sumW     = xlsread(obj.FileName, 'J2:J2');
    obj.sumA     = xlsread(obj.FileName, 'K2:K2');
    obj.Foc      = xlsread(obj.FileName, 'L2:L2');

    obj.Mt       = obj.getMt;
    obj.Xi       = obj.getXi;
    obj.d        = obj.getd;
    obj.Sei      = obj.getSei ;
    obj.d        = obj.getd;
    obj.Sti      = obj.getSti;
    obj.Vpei     = obj.getVpei;
```

Figure 13-NAPL _Cti

Construction of a new script to evaluate the appearance of NAPL

The construction of a new script is mandatory to evaluate the value of NAPL i during the variation of Cti.

The new script is going to be called TEST_Cti

```
%% Entrada de datos

Filename = 'parametrosentrada.xlsx';
Cti      = 100;
Mess     = 'N/A';
```

```
%% Ejecución
f = NAPL_Cti (Cti);
f.NAPLi;
while f.NAPLi == Mess

    Cti = Cti + 10;
    f = NAPL_Cti (Cti);
    Res = f.NAPLi

end
```

Figure 14-TEST_Cti

In the figure above the Matlab code is shown and it works like as follows:

1. The filename 'parametrosentrada.xlsx' is called to have the parameters (except nt) in the test script.
2. The variable Cti is created with the initial value of 100 mg/kg as it is the initial concentration for the pollutant in the soil.
3. The function NAPL_Cti is called.
4. The result of the variable NAPL i which tells whether there is NAPL free phase and quantifies it is saved in the program.
5. A condition is established: while the value of NAPL i is equal to 'N/A', a value is added to Cti to evaluate NAPL_Cti again and see whether there is NAPL free phase by saving the variable NAPL i .
6. When NAPL i is different from 'N/A' the programme stops and gives the value of Cti and NAPL i .

Results

The program runs and outputs some results:

```
Command Window

Res =

    'N/A'

Res =

    'N/A'

Res =

    'N/A'

Res =

    'N/A'

Res =

    2.1385

fx >> |
```

Figure 15-Last iterations of TEST_Cti

The program has found that after some iterations, the first value for NAPL i different from 'N/A' is 2.1385. The program also opens a chapter in the workspace in which the result of the evaluation of NAPL_Cti for the last iteration is shown.

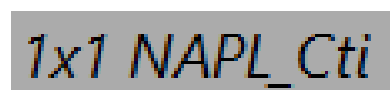


Figure 16-Workspace

In this chapter of the workspace, all the parameters are going to be shown, but the one we are looking for is Cti.

The following picture shows the exact results for the last iteration.

Si	1100
Vpi	74.3100
Hi	0.3649
Koci	126
di	1.4600
Cti	270
nt	0.4000
sumW	0.3000
sumA	0.1000
Foc	1.0000e-04
Wv	1
dw	1
a	0.1000
da	0.0013
w	0.4000
ds	2.6500
PNAPL	1.4900
Mess	'N/A'
Pb	1.8903
Ri	8.3145
CFi	1000000
Ti	293.1500
Pi	101325

Figure 17-Results for the last iteration

The value for Cti which is used to obtain a NAPL free phase concentration of 2.1385 mg/Kg is 270 mg/kg of Trichloroethylene.

4.2.3 Discussion

The program allows to the calculation of the NAPL free phase, which in the first pollutant of study it isn't formatted for these soil conditions, however in the second case of study, there is NAPL free phase due to the condition of $C_{sati} < C_{ti}$.

So, in a hypothetical case that these two pollutants are found contaminating a soil under these conditions, some environmental assessment would need to be done to reduce the risk of living beings to be contaminated.

The sensitivity analysis could help me to evaluate how good did the program respond towards any change in the variation of some parameters.

1. The porosity parameters are not that important regarding the variation of NAPL free phase under these conditions.
2. I could programme a code that finds the minimum value for C_{ti} (contaminant concentration in soil) that verifies the appearance of NAPL free phase, so that is the limit for the concentration.

5. CONCLUSION

The main factor in controlling the NAPL free phase formation is the solubility limit C_{sat} because it is used to determine if NAPL is in the sample because it is used to perform the equilibrium mass balance. Any contaminant mass above the C_{sat} value is NAPL by definition ($C_{ti} - C_{sati} = \text{NAPL}_i$, if $C_{ti} > C_{sati}$).

This software provides a powerful and robust method of assessing the magnitude of NAPL contamination at a variety of waste sites in all media phases. Use of this model should save both time and money by eliminating the tedious calculations and lookup values needed to perform this analysis. Additionally, the extent of contamination at a waste site may be better estimated using this software. Use of the software in conjunction with the characterization data can result in a detailed development of the subsurface volume of contamination. This will allow the establishment of a baseline against which a remedial system can be designed and its performance assessed.

6. BIBLIOGRAPHY

- Agència de Residus de Catalunya. (n.d.). *Guía técnica para la evaluación de la problemática del subsuelo asociada a compuestos organoclorados*.
- Ambiente, M., Miguel, E. De, & Barrio, F. (2015). *CONTAMINANTES*.
- Rebollo, L. F., & Hidrogeología A, H. (n.d.). *Tema 2 PROPIEDADES HIDRÁULICAS ELEMENTALES DE LOS MEDIOS SÓLIDOS*. 1–19. https://portal.uah.es/portal/page/portal/GP_EPD/PG-MA-ASIG/PG-ASIG-67044/TAB42351/T2-Propiedades_hidr%EAulicas_elementales.pdf
- Technology, I., Council, R., Epa, U. S., Up, C., & Network, I. (n.d.). *LNAPL Parte 1 : Un conocimiento más profundo del comportamiento de los LNAPL en el subsuelo Estado de la Ciencia vs Estado de la Práctica Cuestiones Técnicas*.

7. ANNEXED

7.1 CODE

classdef NAPL 22

%% Parámetros Considerados

properties

Filename = 'parametroentrada.xlsx'; %nombre del fichero excel

Data = xlsread('parametroentrada.xlsx'); %Datos contenidos en el fichero

Mwi; % Peso molecular (g/mol)

Si; %Limite de solubilidad (mg/l)

Vpi; %Presion de vapor (mmHg)

Hi; %Constante de Henry

Koci; %Coeficiente de carbon organico (L/Kg)

di; %Densidad (Kg/L)

Cti; %Masa de contaminante en muestra (mg/Kg)

nt; %Porosidad total

sumW; %contenido volumetrico de agua

sumA; %contenido volumetrico de aire

Foc; %Fraccion carbon organico

end

properties (Constant)

name = 'Trichloroethylene';

Wv = 1.0;

dw = 1.0;

a = 0.1;

da = 0.0013;

w = 0.4;

ds = 2.65;

PNAPL = 1.0;

Mess= 'N/A'

Pb = 1.8903; %Densidad aparente, el valor se corresponde con el promedio de valores de un ensayo experimental aportado por el profesor.

Ri = 8.31451; %constante de los gases

CFi = 1.0*10^6; %factor de conversion

Ti = 293.15; %temperatura

Pi = 101325; %Presion

end

properties (Dependent)

%ACUMULATIVOS DE TODOS LOS CONTAMINANTES

Mt; %suma de masas de contaminantes

%##### DATOS A OBTENER SOBRE UN CONTAMINANTE #####

Contaminante; %Nombre del Contaminante

Xi; %Fraccion molar de contaminantes $Xi = Mci \cdot Mci$ $Mci = Wti/MWi$ $Wti = Cti/Mt$ (Mwi,Cti)

d; %Densidad de la mezcla $d = \sum (Xi \cdot di)$

Sei; %Solubilidad efectiva $Sei = Xi \cdot Si$

Sti; %Solubilidad de la mezcla

Vpei; %Presión efectiva de vapor $VPei = (Xi \cdot VPi)$

Kdi; %coeficiente de particion

%##### DATOS A OBTENER SOBRE SUELO DE ESTUDIO #####

ICwi; %Concentración en poro acuoso en el suelo

Csat; %Limite de saturación en el suelo

Cconi; %Limite de saturación volumétrico

Cwi; %Concentración contaminante en los poros acuosos

Cai; %Concentración Contaminante en vapor de los poros

Cavi; %Concentración Volumetrica contaminante en vapor poro

Csj; %Concentración Contaminante de sorción en solido

DATOS A OBTENER CÁLCULOS DE MASAS DE CONTAMINANTES

Mwii; %Masa de contaminante en los poros ocupados por agua
Mai; %Masa de contaminante en los poros ocupado por aire
Msi; %Masa de contaminante de sorción en el suelo
NAPL i; %Masa NAPL de contaminante
Mti; %Balance de masa para contaminante $Mti=Mai+Msi+Mwi+NAPL i$
NAPL t; %Masa total de NAPL s
Sr; %Porcentaje residual saturado de NAPL

DATOS A OBTENER CÁLCULOS DE VOLÚMENES

Nv; %Contenido en volumen NAPL
Av; %Contenido volumétrico de aire en la muestra
Vw; %Contenido en Volumen de Agua en la muestra
Sv; %Contenido en Volumen de suelo en la muestra

end

%% Métodos

methods

FUNCIÓN PARA OBTENER TODOS LOS DATOS DEL ARCHIVO

```
function obj = NAPL 22()  
    % Definición del objeto  
  
    obj.Mwi = xlsread(obj.FileName,'B2:B2'); % Peso molecular (g/mol)  
    obj.Si = xlsread(obj.FileName,'C2:C2'); %Limite de solubilidad (mg/l)  
    obj.Vpi = xlsread(obj.FileName,'D2:D2'); %Presion de vapor (mmHg)  
    obj.Hi = xlsread(obj.FileName,'E2:E2'); %Constante de Henry  
    obj.Koci = xlsread(obj.FileName,'F2:F2'); %Coeficiente de carbon organico (L/Kg)  
    obj.di = xlsread(obj.FileName,'G2:G2'); %Densidad (Kg/L)  
    obj.Cti = xlsread(obj.FileName,'H2:H2'); %Masa de contaminante en muestra (mg/Kg)  
    obj.nt = xlsread(obj.FileName,'I2:I2'); %Porosidad total  
    obj.sumW = xlsread(obj.FileName,'J2:J2'); %contenido volumetrico de agua  
    obj.sumA = xlsread(obj.FileName,'K2:K2'); %contenido volumetrico de aire  
    obj.Foc = xlsread(obj.FileName,'L2:L2'); %Fraccion carbon organico
```

end

%Funcion para calcular la suma de masas de contaminantes

```
function Mt= get.Mt(obj)  
    Mt=sum(obj.Cti);  
end
```

FUNCIONES PARA OBTENER TODOS LOS DATOS DEL CONTAMINANTE

% Nombre del contaminante

```
function Contaminante = get.Contaminante(obj)  
  
    Contaminante = obj.name;  
    disp('Contaminante');  
    disp(Contaminante);
```

end

% 1. Fraccion molar de contaminantes

%Función que calcula la fraccion molar de contaminante en el total
%de masa de contaminantes del suelo.

```
function Xi = get.Xi(obj)
```

```

sumMwi= sum(obj.Mwi) ;
Wti= obj.Cti/sumMwi;
Mci = Wti/obj.Mwi;
Xi=Mci/abs(Mci);

end

% 2. Densidad de la mezcla
%Función que calcula la densidad de la mezcla
function d = get.d(obj)

d= obj.Xi*obj.di;

end

% 3. Solubilidad efectiva
%Función que calcula la solubilidad efectiva del compuesto
function Sei = get.Sei(obj)

Sei=obj.Xi*obj.Si;

end

% 4. Solubilidad de la mezcla
%Función que suma las solubilidades de los compuestos
function Sti = get.Sti(obj)
Sti= sum(obj.Sei);

end

% 5. Presión efectiva de vapor
%Funcion que calcula la presión efectiva de vapor, como la fracción
%molar del contaminante por la presion de vapor en fase pura del
%compuesto
function Vpei = get.Vpei(obj)

Vpei= obj.Xi*obj.Vpi;

end

%6. Coeficiente de partición
%Funcion que calcula el coeficiente de partición como la fraccion
%de carbono organico por el coeficiente de particion del suelo
function Kdi = get.Kdi(obj)

Kdi= obj.Foc*obj.Koci;

end

##### FUNCIONES PARA OBTENER TODOS LOS DATOS REFERENTES AL SUELO DE
ESTUDIO #####

% 1. Concentración en poro acuoso en el suelo
%Función que permite calcular la concentracion teorica del
%contaminante en los poros del suelo.
function ICwi = get.ICwi(obj)

```

```
ar14= obj.Cti*obj.Pb;  
ar15 = obj.Kdi*obj.Pb;  
ar16 = abs(obj.w);  
ar17 = obj.Hi*abs(obj.a);  
sum3 = ar15+ar16+ar17;
```

```
ICwi = ar14/sum3;
```

```
end
```

```
% 2. Limite de saturación en el suelo  
%Función que permite calcular el limite de saturación en el suelo  
%para el contaminante en cuestion.
```

```
function Csat = get.Csat(obj)
```

```
ar1 = obj.Sei/obj.Pb;  
ar2 = obj.Kdi*obj.Pb;  
ar3 = abs(obj.w);  
ar4 = abs(obj.a);  
ar5 = obj.Hi*ar4;  
sum = ar2+ar3+ar5;
```

```
Csat = ar1*sum;
```

```
end
```

```
% 3. Limite de saturación volumétrico  
%Funcion que permite calcular el limite de saturacion volumetrico,  
% expresado en mg/L en vez de mg/kg, es decir, se trata del Csati  
% multiplicado por Pb (densidad aparente)
```

```
function Cconi = get.Cconi(obj)
```

```
Cconi = obj.Csat*obj.Pb;
```

```
end
```

```
% 4. Concentración contaminante en los poros acuosos  
%Función que asigna un valor a la concentracion de contaminante en  
%los poros ocupados por agua
```

```
function Cwi = get.Cwi(obj)
```

```
Cwi = obj.Si;
```

```
end
```

```
% 5. Concentración Contaminante en vapor de los poros  
%Funcion que calcula la concentracion de contaminante en vapor en  
%los poros
```

```
function Cai = get.Cai(obj)
```

```
Cai = obj.Cwi*obj.Hi;
```

```
end
```

```
% 6. Concentración Volumetrica contaminante en vapor poro  
%Funcion que calcula la concentracion volumetrica en el vapor de un  
%contaminante
```

```
function Cavi = get.Cavi(obj)
```

```
ar18 = obj.Cai*obj.Ri;
ar19 = obj.CFi*obj.Ti;
ar20 = ar18*ar19;
ar21 = obj.Pi*obj.Mwi;
Cavi = ar20/ar21;
```

```
end
```

```
% 7. Concentración Contaminante de sorción en solido
%Funcion que permite calcular la concentracion de contaminante
%absorbido por el material solido.
```

```
function Csi = get.Csi(obj)
    Csi = obj.Kdi*obj.Cwi;
```

```
end
```

```
%##### FUNCIONES PARA OBTENER TODOS LOS DATOS REFERENTES A LAS MASAS
CONTAMINANTES #####
```

```
% 1. Masa de contaminante en los poros ocupado por agua
%Funcion, que a partir de la concentracion de contaminante en poros
%acuosos, permite calcular la masa de dicho contaminante contenida en
%los citados poros.
```

```
function Mwii= get.Mwii(obj)
```

```
    Mwii= (obj.Cwi * obj.w)/obj.Pb;
```

```
end
```

```
% 2. Masa de contaminante en los poros ocupado por aire
%Funcion, que a partir de la concentracion de contaminante en poros
%en aire, permite calcular la masa de dicho contaminante contenida en
%los citados poros.
```

```
function Mai = get.Mai(obj)
```

```
ar22 = obj.Cwi;
ar23 = abs(obj.a);
ar24 = obj.Hi;
ar25 = ar22*ar23*ar24;
Mai = ar25/obj.Pb;
```

```
end
```

```
% 3. Masa de contaminante de sorción en el suelo
%Funcion, que a partir de la concentracion de contaminante
%absorbido por el solido
%permite calcular la masa de dicho contaminante contenida en
%los citados poros.
```

```
function Msi = get.Msi(obj)
```

```

Msi = obj.Cwi*obj.Kdi;

end

% 4. Masa NAPL de contaminante
%Función que permite calcular la masa total de NAPL en fase libre
%del contaminante. Si Cti (masa del contaminante en el suelo) es
%mayor que Csati (limite de saturacion del suelo), no habrá NAPL en
%la muestra, y el programa devolverá 1
function NAPL i = get.NAPL i(obj)
ar1 = obj.Sei/obj.Pb;
ar2 = obj.Kdi*obj.Pb;
ar3 = abs(obj.w);
ar4 = abs(obj.a);
ar5 = obj.Hi*ar4;
sum = ar2+ar3+ar5;

Csati2 = ar1*sum;

if obj.Cti>Csati2
    NAPL i = obj.Cti - Csati2;
else
    NAPL i= obj.Mess ;
end

end

% 4. Balance de masa para contaminante Mti=Mai+Msi+Mwi+NAPL i
function Mti = get.Mti(obj)
    if obj.NAPL i== obj.Mess
        Mti=obj.Mess;
    else

Mti=obj.Mai+obj.Msi+obj.Mwi+obj.NAPL i;
    end

end

% 5. Masa total de NAPL s
function NAPL t = get.NAPL t(obj)
    if obj.Mti==obj.Mess
        NAPL t=obj.Mess;
    else

NAPL t=abs(obj.NAPL i);
    end

end

% 6. Porcentaje residual saturado de NAPL
function Sr = get.Sr(obj)
    if obj.NAPL t==obj.Mess
        Sr=obj.Mess;
    else

```

```
ar26 = obj.Pb/obj.d;  
ar27 = obj.NAPL t*ar26;  
Sr = ar27/obj.nt;  
end
```

```
end
```

```
%##### FUNCIONES PARA OBTENER TODOS LOS VOLÚMENES #####
```

```
% 1. Contenido en volumen NAPL
```

```
function Nv = get.Nv(obj)  
if obj.Sr==obj.Mess  
Nv=obj.Mess;  
else
```

```
ar26 = obj.NAPL t * obj.Pb;  
ar27 = obj.PNAPL ;  
Nv = ar26/ar27;  
end
```

```
end
```

```
% 2. Contenido volumétrico de aire en la muestra
```

```
function Av = get.Av(obj)  
if obj.Nv==obj.Mess  
Av=obj.Mess;  
else
```

```
Av= abs(obj.a)-obj.Nv;  
end
```

```
end
```

```
% 3. Contenido en Volumen de Agua en la muestra
```

```
function Vw = get.Vw(obj)  
if obj.Av==obj.Mess  
Vw=obj.Mess;  
else
```

```
Vw = abs(obj.w)-obj.Nv;  
end
```

```
end
```

```
% 4. Contenido en Volumen de suelo en la muestra
```

```
function Sv = get.Sv(obj)  
if obj.Vw==obj.Mess  
Sv=obj.Mess;  
else
```

```
ar28 = abs(obj.w)+abs(obj.a)+obj.Nv;  
Sv = 1.0-ar28;  
end
```

```
end
```


end

end

classdef NAPL_Cti

%% Parámetros Considerados

properties

Filename = 'parametrosentrada.xlsx' ;%nombre del fichero excel

Data % = xlsread('parametrosentrada.xlsx'); %Datos contenidos en el fichero

Mwi; % Peso molecular (g/mol)

Si; %Limite de solubilidad (mg/l)

Vpi; %Presion de vapor (mmHg)

Hi; %Constante de Henry

Koci; %Coeficiente de carbon organico (L/Kg)

di; %Densidad (Kg/L)

Cti; %Masa de contaminante en muestra (mg/Kg)

nt; %Porosidad total

sumW; %contenido volumetrico de agua

sumA; %contenido volumetrico de aire

Foc; %Fraccion carbon organico

end

properties

Wv = 1.0;

dw = 1.0;

a = 0.1;

da = 0.0013;

w = 0.4;

ds = 2.65;

PNAPL = 1.49;

Mess= 'N/A'

Pb = 1.8903; %Densidad aparente, el valor se corresponde con el promedio de valores de un ensayo experimental aportado por el profesor.

Ri = 8.31451; %constante de los gases

CFi = 1.0*10^6; %factor de conversion

Ti = 293.15; %temperatura

Pi = 101325; %Presion

end

properties

%ACUMULATIVOS DE TODOS LOS CONTAMINANTES

Mt; %suma de masas de contaminantes

%##### DATOS A OBTENER SOBRE UN CONTAMINANTE #####

Contaminante; %Nombre del Contaminante

Xi; %Fraccion molar de contaminantes $Xi = Mci \cdot Mci$ $Mci = Wti/MWi$ $Wti = Cti/Mt$ (Mwi,Cti)

d; %Densidad de la mezcla $d = \sum (Xi \cdot di)$

Sei; %Solubilidad efectiva $Sei = Xi \cdot Si$

```
Sti; %Solubilidad de la mezcla
Vpei; %Presión efectiva de vapor  $V_{Pei} = (X_i * V_{Pi})$ 
Kdi; %coeficiente de particion
```

```
%##### DATOS A OBTENER SOBRE SUELO DE ESTUDIO #####
```

```
ICwi; %Concentración en poro acuoso en el suelo
Csat; %Limite de saturación en el suelo
Cconi; %Limite de saturación volumétrico
Cwi; %Concentración contaminante en los poros acuosos
Cai; %Concentración Contaminante en vapor de los poros
Cavi; %Concentración Volumetrica contaminante en vapor poro
Csi; %Concentración Contaminante de sorción en solido
```

```
%##### DATOS A OBTENER CÁLCULOS DE MASAS DE CONTAMINANTES #####
```

```
Mwii; %Masa de contaminante en los poros ocupados por agua
Mai; %Masa de contaminante en los poros ocupado por aire
Msi; %Masa de contaminante de sorción en el suelo
NAPLi; %Masa NAPL de contaminante
Mti; %Balance de masa para contaminante  $M_{ti} = M_{ai} + M_{si} + M_{wi} + NAPL_i$ 
NAPL_t; %Masa total de NAPLs
Sr; %Porcentaje residual saturado de NAPL
```

```
%##### DATOS A OBTENER CÁLCULOS DE VOLÚMENES #####
```

```
Nv; %Contenido en volumen NAPL
Av; %Contenido volumétrico de aire en la muestra
Vw; %Contenido en Volumen de Agua en la muestra
Sv; %Contenido en Volumen de suelo en la muestra
```

```
end
```

```
%% Métodos
methods
```

```
%##### FUNCIÓN PARA OBTENER TODOS LOS DATOS DEL ARCHIVO #####
```

```
function obj = NAPL_Cti(Cti)
% Definición del objeto
```

```
obj.Mwi = xlsread(obj.FileName, 'B2:B2'); % Peso molecular (g/mol)
obj.Si = xlsread(obj.FileName, 'C2:C2'); %Limite de solubilidad (mg/l)
obj.Vpi = xlsread(obj.FileName, 'D2:D2'); %Presion de vapor (mmHg)
obj.Hi = xlsread(obj.FileName, 'E2:E2'); %Constante de Henry
obj.Koci = xlsread(obj.FileName, 'F2:F2'); %Coeficiente de carbon organico (L/Kg)
obj.di = xlsread(obj.FileName, 'G2:G2'); %Densidad (Kg/L)
obj.Cti = Cti; %Masa de contaminante en muestra (mg/Kg)
obj.nt = xlsread(obj.FileName, 'I2:I2');
obj.sumW = xlsread(obj.FileName, 'J2:J2');
obj.sumA = xlsread(obj.FileName, 'K2:K2'); %contenido volumetrico de aire
obj.Foc = xlsread(obj.FileName, 'L2:L2'); %Fraccion carbon organico
```

```
obj.Mt = obj.getMt;
obj.Xi = obj.getXi;
obj.d = obj.getd;
obj.Sei = obj.getSei;
obj.d = obj.getd;
obj.Sti = obj.getSti;
obj.Vpei = obj.getVpei;
obj.Kdi = obj.getKdi;
obj.ICwi = obj.getICwi;
obj.Csat = obj.getCsat;
obj.Cconi = obj.getCconi;
obj.Cwi = obj.getCwi;
obj.Cai = obj.getCai;
```

```

obj.Cavi      = obj.getCavi;
obj.Csi      = obj.getCsi;
obj.Mwii     = obj.getMwii;
obj.Mai      = obj.getMai;
obj.Msi      = obj.getMsi;
obj.NAPLi    = obj.getNAPLi;
obj.Mti      = obj.getMti;
obj.NAPLt    = obj.getNAPLt;
obj.Sr       = obj.getSr;
obj.Nv       = obj.getNv;
obj.Av       = obj.getAv;
obj.Vw       = obj.getVw;
obj.Sv       = obj.getSv;

```

```
end
```

```
%Funcion para calcular la suma de masas de contaminantes
```

```
function Mt= getMt(obj)
```

```
    Mt=sum(obj.Cti);
```

```
end
```

```
%##### FUNCIONES PARA OBTENER TODOS LOS DATOS DEL CONTAMINANTE #####
```

```
% 1. Fraccion molar de contaminantes
```

```
%Función que calcula la fraccion molar de contaminante en el total
```

```
%de masa de contaminantes del suelo.
```

```
function Xi = getXi(obj)
```

```
    sumMwi= sum(obj.Mwi) ;
```

```
    Wti= obj.Cti/sumMwi;
```

```
    Mci = Wti/obj.Mwi;
```

```
    Xi=Mci/abs(Mci);
```

```
end
```

```
% 2. Densidad de la mezcla
```

```
%Función que calcula la densidad de la mezcla
```

```
function d = getd(obj)
```

```
    d= obj.getXi*obj.di;
```

```
end
```

```
% 3. Solubilidad efectiva
```

```
%Función que calcula la solubilidad efectiva del compuesto
```

```
function Sei = getSei(obj)
```

```
    Sei=obj.getXi*obj.Si;
```

```
end
```

```
% 4. Solubilidad de la mezcla
```

```
%Función que suma las solubilidades de los compuestos
```

```
function Sti = getSti(obj)
    Sti= sum(obj.getSei);
```

```
end
```

```
% 5. Presión efectiva de vapor
%Funcion que calcula la presión efectiva de vapor, como la fracción
%molar del contaminante por la presión de vapor en fase pura del
%compuesto
```

```
function Vpei = getVpei(obj)
```

```
Vpei= obj.Xi*obj.Vpi;
```

```
end
```

```
%6. Coeficiente de partición
%Funcion que calcula el coeficiente de partición como la fracción
%de carbono organico por el coeficiente de particion del suelo
```

```
function Kdi = getKdi(obj)
```

```
Kdi= obj.Foc*obj.Koci;
```

```
end
```

```
%##### FUNCIONES PARA OBTENER TODOS LOS DATOS REFERENTES AL SUELO DE
ESTUDIO #####
```

```
% 1. Concentración en poro acuoso en el suelo
%Función que permite calcular la concentración teorica del
%contaminante en los poros del suelo.
```

```
function ICwi = getICwi(obj)
```

```
ar14= obj.Cti*obj.Pb;
ar15 = obj.getKdi*obj.Pb;
ar16 = abs(obj.w);
ar17 = obj.Hi*abs(obj.a);
sum3 = ar15+ar16+ar17;
```

```
ICwi = ar14/sum3;
```

```
end
```

```
% 2. Limite de saturación en el suelo
%Función que permite calcular el limite de saturación en el suelo
%para el contaminante en cuestion.
```

```
function Csat = getCsat(obj)
```

```
ar1 = obj.Sei/obj.Pb;
ar2 = obj.Kdi*obj.Pb;
ar3 = abs(obj.w);
ar4 = abs(obj.a);
ar5 = obj.Hi*ar4;
sum = ar2+ar3+ar5;
```

```
Csat = ar1*sum;
```

```
end
```

```

% 3. Limite de saturación volumétrico
%Funcion que permite calcular el limite de saturacion volumetrico,
% expresado en mg/L en vez de mg/kg, es decir, se trata del Csati
% multiplicado por Pb (densidad aparente)
function Cconi = getCconi(obj)

```

```

    Cconi = obj.getCsat*obj.Pb;

```

```

end

```

```

% 4. Concentración contaminante en los poros acuosos
%Función que asigna un valor a la concentracion de contaminante en
%los poros ocupados por agua
function Cwi = getCwi(obj)

```

```

    Cwi = obj.Si;

```

```

end

```

```

% 5. Concentración Contaminante en vapor de los poros
%Funcion que calcula la concentracion de contaminante en vapor en
%los poros
function Cai = getCai(obj)

```

```

    Cai = obj.getCwi*obj.Hi;

```

```

end

```

```

% 6. Concentración Volumetrica contaminante en vapor poro
%Funcion que calcula la concentracion volumetrica en el vapor de un
%contaminante
function Cavi = getCavi(obj)

```

```

    ar18 = obj.Cai*obj.Ri;
    ar19 = obj.CFi*obj.Ti;
    ar20 = ar18*ar19;
    ar21 = obj.Pi*obj.Mwi;
    Cavi = ar20/ar21;

```

```

end

```

```

% 7. Concentración Contaminante de sorción en solido
%Funcion que permite calcular la concentracion de contaminante
%absorbido por el material solido.

```

```

function Csi = getCsi(obj)
    Csi = obj.getKdi*obj.getCwi;

```

```

end

```

```

##### FUNCIONES PARA OBTENER TODOS LOS DATOS REFERENTES A LAS MASAS
CONTAMINANTES #####

```

```

% 1. Masa de contaminante en los poros ocupado por agua
%Funcion, que a partir de la concentracion de contaminante en poros
%acuoso, permite calcular la masa de dicho contaminante contenida en
%los citados poros.

```

```
function Mwii= getMwii(obj)
```

```
    Mwii= (obj.getCwi * obj.w)/obj.Pb;
```

```
end
```

```
% 2. Masa de contaminante en los poros ocupado por aire  
%Funcion, que a partir de la concentracion de contaminante en poros  
%en aire, permite calcular la masa de dicho contaminante contenida en  
%los citados poros.
```

```
function Mai = getMai(obj)
```

```
    ar22 = obj.getCwi;  
    ar23 = abs(obj.a);  
    ar24 = obj.Hi;  
    ar25 = ar22*ar23*ar24;  
    Mai = ar25/obj.Pb;
```

```
end
```

```
% 3. Masa de contaminante de sorción en el suelo  
%Funcion, que a partir de la concentracion de contaminante  
%absorbido por el solido  
%permite calcular la masa de dicho contaminante contenida en  
%los citados poros.
```

```
function Msi = getMsi(obj)
```

```
    Msi = obj.getCwi*obj.getKdi;
```

```
end
```

```
% 4. Masa NAPL de contaminante  
%Función que permite calcular la masa total de NAPL en fase libre  
%del contaminante. Si Cti (masa del contaminante en el suelo) es  
%mayor que Csati (limite de saturacion del suelo), no habrá NAPL en  
%la muestra, y el programa devolverá 1
```

```
function NAPLi = getNAPLi(obj)
```

```
    ar1 = obj.getSei/obj.Pb;  
    ar2 = obj.getKdi*obj.Pb;  
    ar3 = abs(obj.w);  
    ar4 = abs(obj.a);  
    ar5 = obj.Hi*ar4;  
    sum = ar2+ar3+ar5;
```

```
    Csati2 = ar1*sum;
```

```
if obj.Cti>Csati2  
    NAPLi = obj.Cti - Csati2;
```

```
else
```

```
    NAPLi= obj.Mess ;
```

```
end
```

```
end
```

```
% 4. Balance de masa para contaminante Mti=Mai+Msi+Mwi+NAPLi
```

```
function Mti = getMti(obj)
    if obj.NAPLi==obj.Mess
        Mti=obj.Mess;
    else

    Mti=obj.Mai+obj.Msi+obj.Mwii+obj.NAPLi;
    end
```

```
end
```

```
% 5. Masa total de NAPLs
```

```
function NAPLt = getNAPLt(obj)
    if obj.Mti==obj.Mess
        NAPLt=obj.Mess;
    else

    NAPLt=sum(obj.NAPLi);
    end
```

```
end
```

```
% 6. Porcentaje residual saturado de NAPL
```

```
function Sr = getSr(obj)
    if obj.NAPLt==obj.Mess
        Sr=obj.Mess;
    else

    ar26 = obj.Pb/obj.PNAPL;
    ar27 = obj.NAPLt*ar26;
    Sr = ar27/(obj.nt*100);
    end
```

```
end
```

```
%##### FUNCIONES PARA OBTENER TODOS LOS VOLÚMENES #####
```

```
% 1. Contenido en volumen NAPL
```

```
function Nv = getNv(obj)
    if obj.Sr==obj.Mess
        Nv=obj.Mess;
    else

    ar26 = obj.getNAPLt * obj.Pb;
    ar27 = obj.PNAPL;
    Nv = ar26/ar27;
    end
```

```
end
```

```
% 2. Contenido volumétrico de aire en la muestra
```

```

function Av = getAv(obj)
    if obj.Nv==obj.Mess
        Av=obj.Mess;
    else

        Av= abs(100*obj.a)-obj.getNv;
        if Av<1
            Av=obj.Mess;
        else
            end
        end

    end

    end

% 3. Contenido en Volumen de Agua en la muestra
function Vw = getVw(obj)
    if obj.getAv==obj.Mess
        Vw=obj.Mess;
    else

        Vw = abs(100*obj.w)-obj.getNv;
        if Vw<1
            Vw=obj.Mess;
        else

            end
        end

    end

    end

% 4. Contenido en Volumen de suelo en la muestra
function Sv = getSv(obj)
    if obj.Vw==obj.Mess
        Sv=obj.Mess;
    else

        ar28 = abs(obj.w)+abs(obj.a)+obj.getNv;
        Sv = 1.0-ar28;
        end

    end

    end

end

end

end

%% Entrada de datos

Filename = 'parametrosentrada.xlsx';
Cti     = 100;
Mess    = 'N/A';

%% Ejecución
f = NAPL_Cti (Cti);
f.NAPLi;
while f.NAPLi == Mess

```



```

Cti = Cti + 10;
f = NAPL_Cti (Cti);
Res = f.NAPLi

```

```
end
```

```
classdef NAPL_nt
```

```
%% Parámetros Considerados
```

```
properties
```

```

Filename = 'parametroentrada.xlsx' ;%nombre del fichero excel
Data % = xlsread('parametroentrada.xlsx'); %Datos contenidos en el fichero

Mwi; % Peso molecular (g/mol)
Si; %Limite de solubilidad (mg/l)
Vpi; %Presion de vapor (mmHg)
Hi; %Constante de Henry
Koci; %Coeficiente de carbon organico (L/Kg)
di; %Densidad (Kg/L)
Cti; %Masa de contaminante en muestra (mg/Kg)
nt; %Porosidad total
sumW; %contenido volumetrico de agua
sumA; %contenido volumetrico de aire
Foc; %Fraccion carbon organico

```

```
end
```

```
properties
```

```

Wv = 1.0;
dw = 1.0;
a = 0.1;
da = 0.0013;
w = 0.4;
ds = 2.65;
PNAPL = 1.49;
Mess= 'N/A'
Pb = 1.8903; %Densidad aparente, el valor se corresponde con el promedio de valores de un
ensayo experimental aportado por el profesor.
Ri = 8.31451; %constante de los gases
CFi = 1.0*10^6; %factor de conversion
Ti = 293.15; %temperatura
Pi = 101325; %Presion

```

```
end
```

```
properties
```

```
%ACUMULATIVOS DE TODOS LOS CONTAMINANTES
```

```
Mt; %suma de masas de contaminantes
```

```
%##### DATOS A OBTENER SOBRE UN CONTAMINANTE #####
```

```

Contaminante; %Nombre del Contaminante
Xi; %Fraccion molar de contaminantes  $X_i = \frac{M_{ci}}{M_t}$   $M_{ci} = \frac{W_{ti}}{M_{wi}}$   $W_{ti} = \frac{C_{ti}}{M_t}$  (Mwi,Cti)
d; %Densidad de la mezcla  $d = \sum (X_i * d_i)$ 
Sei; %Solubilidad efectiva  $Se_i = X_i * S_i$ 
Sti; %Solubilidad de la mezcla
Vpei; %Presión efectiva de vapor  $VP_{ei} = (X_i * VP_i)$ 
Kdi; %coeficiente de particion

```

```
%##### DATOS A OBTENER SOBRE SUELO DE ESTUDIO #####
```

```

ICwi; %Concentración en poro acuoso en el suelo
Csat; %Limite de saturación en el suelo

```

Cconi; %Limite de saturación volumétrico
 Cwi; %Concentración contaminante en los poros acuosos
 Cai; %Concentración Contaminante en vapor de los poros
 Cavi; %Concentración Volumetrica contaminante en vapor poro
 Csi; %Concentración Contaminante de sorción en solido

DATOS A OBTENER CÁLCULOS DE MASAS DE CONTAMINANTES

Mwii; %Masa de contaminante en los poros ocupados por agua
 Mai; %Masa de contaminante en los poros ocupado por aire
 Msi; %Masa de contaminante de sorción en el suelo
 NAPLi; %Masa NAPL de contaminante
 Mti; %Balance de masa para contaminante $Mti=Mai+Msi+Mwi+NAPLi$
 NAPLt; %Masa total de NAPLs
 Sr; %Porcentaje residual saturado de NAPL

DATOS A OBTENER CÁLCULOS DE VOLÚMENES

Nv; %Contenido en volumen NAPL
 Av; %Contenido volumétrico de aire en la muestra
 Vw; %Contenido en Volumen de Agua en la muestra
 Sv; %Contenido en Volumen de suelo en la muestra

end

%% Métodos

methods

FUNCIÓN PARA OBTENER TODOS LOS DATOS DEL ARCHIVO

function obj = NAPL_nt(nt)

 % Definición del objeto

obj.Mwi = xlsread(obj.FileName, 'B2:B2'); % Peso molecular (g/mol)
 obj.Si = xlsread(obj.FileName, 'C2:C2'); %Limite de solubilidad (mg/l)
 obj.Vpi = xlsread(obj.FileName, 'D2:D2'); %Presion de vapor (mmHg)
 obj.Hi = xlsread(obj.FileName, 'E2:E2'); %Constante de Henry
 obj.Koci = xlsread(obj.FileName, 'F2:F2'); %Coeficiente de carbon organico (L/Kg)
 obj.di = xlsread(obj.FileName, 'G2:G2'); %Densidad (Kg/L)
 obj.Cti = xlsread(obj.FileName, 'H2:H2'); %Masa de contaminante en muestra (mg/Kg)
 obj.nt = nt ; %Porosidad Total
 obj.sumW = xlsread(obj.FileName, 'K2:K2'); %Contenido Volumetrico de agua
 obj.sumA = xlsread(obj.FileName, 'K2:K2'); %contenido volumetrico de aire
 obj.Foc = xlsread(obj.FileName, 'L2:L2'); %Fraccion carbon organico

obj.Mt = obj.getMt;
 obj.Xi = obj.getXi;
 obj.d = obj.getd;
 obj.Sei = obj.getSei ;
 obj.d = obj.getd;
 obj.Sti = obj.getSti;
 obj.Vpei = obj.getVpei;
 obj.Kdi = obj.getKdi;
 obj.lCwi = obj.getlCwi;
 obj.Csat = obj.getCsat;
 obj.Cconi = obj.getCconi;
 obj.Cwi = obj.getCwi;
 obj.Cai = obj.getCai;
 obj.Cavi = obj.getCavi;
 obj.Csi = obj.getCsi;
 obj.Mwii = obj.getMwii;
 obj.Mai = obj.getMai;
 obj.Msi = obj.getMsi;
 obj.NAPLi = obj.getNAPLi;
 obj.Mti = obj.getMti;

```
obj.NAPLt      = obj.getNAPLt;
obj.Sr         = obj.getSr;
obj.Nv         = obj.getNv;
obj.Av         = obj.getAv;
obj.Vw         = obj.getVw;
obj.Sv         = obj.getSv;
```

```
end
```

```
%Funcion para calcular la suma de masas de contaminantes
```

```
function Mt= getMt(obj)
    Mt=sum(obj.Cti);
end
```

```
%#### FUNCIONES PARA OBTENER TODOS LOS DATOS DEL CONTAMINANTE ####
```

```
% 1. Fraccion molar de contaminantes
```

```
%Función que calcula la fraccion molar de contaminante en el total  
%de masa de contaminantes del suelo.
```

```
function Xi = getXi(obj)
```

```
    sumMwi= sum(obj.Mwi) ;
    Wti= obj.Cti/sumMwi;
    Mci = Wti/obj.Mwi;
    Xi=Mci/abs(Mci);
```

```
end
```

```
% 2. Densidad de la mezcla
```

```
%Función que calcula la densidad de la mezcla
```

```
function d = getd(obj)
```

```
    d= obj.getXi*obj.di;
```

```
end
```

```
% 3. Solubilidad efectiva
```

```
%Función que calcula la solubilidad efectiva del compuesto
```

```
function Sei = getSei(obj)
```

```
    Sei=obj.getXi*obj.Si;
```

```
end
```

```
% 4. Solubilidad de la mezcla
```

```
%Función que suma las solubilidades de los compuestos
```

```
function Sti = getSti(obj)
```

```
    Sti= sum(obj.getSei);
```

```
end
```

```
% 5. Presión efectiva de vapor
%Funcion que calcula la presión efectiva de vapor, como la fracción
% molar del contaminante por la presión de vapor en fase pura del
% compuesto
```

```
function Vpei = getVpei(obj)
```

```
Vpei= obj.Xi*obj.Vpi;
```

```
end
```

```
%6. Coeficiente de partición
%Funcion que calcula el coeficiente de partición como la fracción
% de carbono organico por el coeficiente de particion del suelo
```

```
function Kdi = getKdi(obj)
```

```
Kdi= obj.Foc*obj.Koci;
```

```
end
```

```
##### FUNCIONES PARA OBTENER TODOS LOS DATOS REFERENTES AL SUELO DE
ESTUDIO #####
```

```
% 1. Concentración en poro acuoso en el suelo
%Función que permite calcular la concentración teorica del
% contaminante en los poros del suelo.
```

```
function ICwi = getICwi(obj)
```

```
ar14= obj.Cti*obj.Pb;
ar15 = obj.getKdi*obj.Pb;
ar16 = abs(obj.w);
ar17 = obj.Hi*abs(obj.a);
sum3 = ar15+ar16+ar17;
```

```
ICwi = ar14/sum3;
```

```
end
```

```
% 2. Limite de saturación en el suelo
%Función que permite calcular el limite de saturación en el suelo
% para el contaminante en cuestion.
```

```
function Csat = getCsat(obj)
```

```
ar1 = obj.Sei/obj.Pb;
ar2 = obj.Kdi*obj.Pb;
ar3 = abs(obj.w);
ar4 = abs(obj.a);
ar5 = obj.Hi*ar4;
sum = ar2+ar3+ar5;
```

```
Csat = ar1*sum;
```

```
end
```

```
% 3. Limite de saturación volumétrico
%Funcion que permite calcular el limite de saturacion volumetrico,
% expresado en mg/L en vez de mg/kg, es decir, se trata del Csati
% multiplicado por Pb (densidad aparente)
```

```
function Cconi = getCconi(obj)
```

```
Cconi = obj.getCsat*obj.Pb;
```

```
end
```

```
% 4. Concentración contaminante en los poros acuosos  
%Función que asigna un valor a la concentración de contaminante en  
%los poros ocupados por agua
```

```
function Cwi = getCwi(obj)
```

```
    Cwi = obj.Si;
```

```
end
```

```
% 5. Concentración Contaminante en vapor de los poros  
%Funcion que calcula la concentracion de contaminante en vapor en  
%los poros
```

```
function Cai = getCai(obj)
```

```
    Cai = obj.getCwi*obj.Hi;
```

```
end
```

```
% 6. Concentración Volumetrica contaminante en vapor poro  
%Funcion que calcula la concentracion volumetrica en el vapor de un  
%contaminante
```

```
function Cavi = getCavi(obj)
```

```
    ar18 = obj.Cai*obj.Ri;
```

```
    ar19 = obj.CFi*obj.Ti;
```

```
    ar20 = ar18*ar19;
```

```
    ar21 = obj.Pi*obj.Mwi;
```

```
    Cavi = ar20/ar21;
```

```
end
```

```
% 7. Concentración Contaminante de sorción en solido  
%Funcion que permite calcular la concentracion de contaminante  
%absorbido por el material solido.
```

```
function Csi = getCsi(obj)
```

```
    Csi = obj.getKdi*obj.getCwi;
```

```
end
```

```
##### FUNCIONES PARA OBTENER TODOS LOS DATOS REFERENTES A LAS MASAS  
CONTAMINANTES #####
```

```
% 1. Masa de contaminante en los poros ocupado por agua  
%Funcion, que a partir de la concentracion de contaminante en poros  
%acuosos, permite calcular la masa de dicho contaminante contenida en  
%los citados poros.
```

```
function Mwii= getMwii(obj)
```

```
    Mwii= (obj.getCwi * obj.w)/obj.Pb;
```

```
end
```

```
% 2. Masa de contaminante en los poros ocupado por aire
%Funcion, que a partir de la concentracion de contaminante en poros
%en aire, permite calcular la masa de dicho contaminante contenida en
%los citados poros.
```

```
function Mai = getMai(obj)
```

```
ar22 = obj.getCwi;
ar23 = abs(obj.a);
ar24 = obj.Hi;
ar25 = ar22*ar23*ar24;
Mai = ar25/obj.Pb;
```

```
end
```

```
% 3. Masa de contaminante de sorción en el suelo
%Funcion, que a partir de la concentracion de contaminante
%absorbido por el solido
%permite calcular la masa de dicho contaminante contenida en
%los citados poros.
```

```
function Msi = getMsi(obj)
```

```
Msi = obj.getCwi*obj.getKdi;
```

```
end
```

```
% 4. Masa NAPL de contaminante
%Función que permite calcular la masa total de NAPL en fase libre
%del contaminante. Si Cti (masa del contaminante en el suelo) es
%mayor que Csati (limite de saturacion del suelo), no habrá NAPL en
%la muestra, y el programa devolverá 1
```

```
function NAPLi = getNAPLi(obj)
```

```
ar1 = obj.getSei/obj.Pb;
ar2 = obj.getKdi*obj.Pb;
ar3 = abs(obj.w);
ar4 = abs(obj.a);
ar5 = obj.Hi*ar4;
sum = ar2+ar3+ar5;
```

```
Csati2 = ar1*sum;
```

```
if obj.Cti>Csati2
```

```
    NAPLi = obj.Cti - Csati2;
```

```
else
```

```
    NAPLi= obj.Mess ;
```

```
end
```

```
end
```

```
% 4. Balance de masa para contaminante Mti=Mai+Msi+Mwi+NAPLi
```

```
function Mti = getMti(obj)
```

```
    if obj.NAPLi== obj.Mess
```

```

        Mti=obj.Mess;
    else

    Mti=obj.Mai+obj.Msi+obj.Mwii+obj.NAPLi;
    end

```

```
end
```

```
% 5. Masa total de NAPLs
```

```
function NAPLt = getNAPLt(obj)
    if obj.Mti==obj.Mess
        NAPLt=obj.Mess;
    else

    NAPLt=sum(obj.NAPLi);
    end

```

```
end
```

```
% 6. Porcentaje residual saturado de NAPL
```

```
function Sr = getSr(obj)
    if obj.NAPLt==obj.Mess
        Sr=obj.Mess;
    else

    ar26 = obj.Pb/obj.PNAPL;
    ar27 = obj.NAPLt*ar26;
    Sr = ar27/(obj.nt*100);
    end

```

```
end
```

```
%##### FUNCIONES PARA OBTENER TODOS LOS VOLÚMENES #####
```

```
% 1. Contenido en volumen NAPL
```

```
function Nv = getNv(obj)
    if obj.Sr==obj.Mess
        Nv=obj.Mess;
    else

    ar26 = obj.getNAPLt * obj.Pb;
    ar27 = obj.PNAPL;
    Nv = ar26/ar27;
    end

```

```
end
```

```
% 2. Contenido volumétrico de aire en la muestra
```

```
function Av = getAv(obj)
    if obj.Nv==obj.Mess
        Av=obj.Mess;
    else

    Av= abs(100*obj.a)-obj.getNv;
    if Av<1

```

```

        Av=obj.Mess;
    else
    end
end

end

% 3. Contenido en Volumen de Agua en la muestra
function Vw = getVw(obj)
    if obj.getAv==obj.Mess
        Vw=obj.Mess;
    else

        Vw = abs(100*obj.w)-obj.getNv;
        if Vw<1
            Vw=obj.Mess;
        else

        end
    end

end

% 4. Contenido en Volumen de suelo en la muestra
function Sv = getSv(obj)
    if obj.Vw==obj.Mess
        Sv=obj.Mess;
    else

        ar28 = abs(obj.w)+abs(obj.a)+obj.getNv;
        Sv = 1.0-ar28;
    end

end

end

end

end
Filename = 'parametrosentrada.xlsx';
Mess= 'N/A';
nt=0.05;

f = NAPL_nt (nt);
f.NAPLi;
while nt <= 0.6
    nt = nt + 0.05;
    f = NAPL_nt (nt);
    Res = f.NAPLi
    nt

end

classdef NAPL_sumA
%% Parámetros Considerados
properties

```



```
Filename = 'parametrosentrada.xlsx' ;%nombre del fichero excel  
Data % = xlsread('parametrosentrada.xlsx'); %Datos contenidos en el fichero
```

```
Mwi; % Peso molecular (g/mol)  
Si; %Limite de solubilidad (mg/l)  
Vpi; %Presion de vapor (mmHg)  
Hi; %Constante de Henry  
Koci; %Coeficiente de carbon organico (L/Kg)  
di; %Densidad (Kg/L)  
Cti; %Masa de contaminante en muestra (mg/Kg)  
nt; %Porosidad total  
sumW; %contenido volumetrico de agua  
sumA; %contenido volumetrico de aire  
Foc; %Fraccion carbon organico
```

```
end
```

```
properties
```

```
Wv = 1.0;  
dw = 1.0;  
a = 0.1;  
da = 0.0013;  
w = 0.4;  
ds = 2.65;  
PNAPL = 1.49;  
Mess = 'N/A'  
Pb = 1.8903; %Densidad aparente, el valor se corresponde con el promedio de valores de un  
ensayo experimental aportado por el profesor.  
Ri = 8.31451; %constante de los gases  
CFi = 1.0*10^6; %factor de conversion  
Ti = 293.15; %temperatura  
Pi = 101325; %Presion
```

```
end
```

```
properties
```

```
%ACUMULATIVOS DE TODOS LOS CONTAMINANTES
```

```
Mt; %suma de masas de contaminantes
```

```
%##### DATOS A OBTENER SOBRE UN CONTAMINANTE #####
```

```
Contaminante; %Nombre del Contaminante  
Xi; %Fraccion molar de contaminantes  $Xi = Mci \cdot Mci$   $Mci = Wti/MWi$   $Wti = Cti/Mt$  (Mwi,Cti)  
d; %Densidad de la mezcla  $d = \sum (Xi \cdot di)$   
Sei; %Solubilidad efectiva  $Sei = Xi \cdot Si$   
Sti; %Solubilidad de la mezcla  
Vpei; %Presión efectiva de vapor  $VPei = (Xi \cdot VPi)$   
Kdi; %coeficiente de particion
```

```
%##### DATOS A OBTENER SOBRE SUELO DE ESTUDIO #####
```

```
ICwi; %Concentración en poro acuoso en el suelo  
Csat; %Limite de saturación en el suelo  
Cconi; %Limite de saturación volumétrico  
Cwi; %Concentración contaminante en los poros acuosos  
Cai; %Concentración Contaminante en vapor de los poros  
Cavi; %Concentración Volumetrica contaminante en vapor poro  
Csi; %Concentración Contaminante de sorción en solido
```

```
%##### DATOS A OBTENER CÁLCULOS DE MASAS DE CONTAMINANTES #####
```

```
Mwii; %Masa de contaminante en los poros ocupados por agua  
Mai; %Masa de contaminante en los poros ocupado por aire  
Msi; %Masa de contaminante de sorción en el suelo
```

```

NAPLi; %Masa NAPL de contaminante
Mti; %Balance de masa para contaminante Mti=Mai+Msi+Mwi+NAPLi
NAPLt; %Masa total de NAPLs
Sr; %Porcentaje residual saturado de NAPL

```

```

%#### DATOS A OBTENER CÁLCULOS DE VOLÚMENES ####

```

```

Nv; %Contenido en volumen NAPL
Av; %Contenido volumétrico de aire en la muestra
Vw; %Contenido en Volumen de Agua en la muestra
Sv; %Contenido en Volumen de suelo en la muestra

```

```

end

```

```

%% Métodos

```

```

methods

```

```

%#### FUNCIÓN PARA OBTENER TODOS LOS DATOS DEL ARCHIVO ####

```

```

function obj = NAPL_sumA(sumA)

```

```

    % Definición del objeto

```

```

obj.Mwi = xlsread(obj.FileName, 'B2:B2'); % Peso molecular (g/mol)
obj.Si = xlsread(obj.FileName, 'C2:C2'); %Limite de solubilidad (mg/l)
obj.Vpi = xlsread(obj.FileName, 'D2:D2'); %Presion de vapor (mmHg)
obj.Hi = xlsread(obj.FileName, 'E2:E2'); %Constante de Henry
obj.Koci = xlsread(obj.FileName, 'F2:F2'); %Coeficiente de carbon organico (L/Kg)
obj.di = xlsread(obj.FileName, 'G2:G2'); %Densidad (Kg/L)
obj.Cti = xlsread(obj.FileName, 'H2:H2'); %Masa de contaminante en muestra (mg/Kg)
obj.nt = 0.6 ; %Porosidad Total
obj.sumW = xlsread(obj.FileName, 'J2:J2'); %Contenido Volumetrico de agua
obj.sumA = sumA ; %contenido volumetrico de aire
obj.Foc = xlsread(obj.FileName, 'L2:L2'); %Fraccion carbon organico

```

```

obj.Mt = obj.getMt;
obj.Xi = obj.getXi;
obj.d = obj.getd;
obj.Sei = obj.getSei ;
obj.d = obj.getd;
obj.Sti = obj.getSti;
obj.Vpei = obj.getVpei;
obj.Kdi = obj.getKdi;
obj.lCwi = obj.getlCwi;
obj.Csat = obj.getCsat;
obj.Cconi = obj.getCconi;
obj.Cwi = obj.getCwi;
obj.Cai = obj.getCai;
obj.Cavi = obj.getCavi;
obj.Csi = obj.getCsi;
obj.Mwii = obj.getMwii;
obj.Mai = obj.getMai;
obj.Msi = obj.getMsi;
obj.NAPLi = obj.getNAPLi;
obj.Mti = obj.getMti;
obj.NAPLt = obj.getNAPLt;
obj.Sr = obj.getSr;
obj.Nv = obj.getNv;
obj.Av = obj.getAv;
obj.Vw = obj.getVw;
obj.Sv = obj.getSv;

```

```

end

```

%Funcion para calcular la suma de masas de contaminantes

```
function Mt= getMt(obj)
    Mt=sum(obj.Cti);
end
```

FUNCIONES PARA OBTENER TODOS LOS DATOS DEL CONTAMINANTE

% 1. Fraccion molar de contaminantes

%Función que calcula la fraccion molar de contaminante en el total
%de masa de contaminantes del suelo.

```
function Xi = getXi(obj)
```

```
    sumMwi= sum(obj.Mwi) ;
    Wti= obj.Cti/sumMwi;
    Mci = Wti/obj.Mwi;
    Xi=Mci/abs(Mci);
```

```
end
```

% 2. Densidad de la mezcla

%Función que calcula la densidad de la mezcla

```
function d = getd(obj)
```

```
    d= obj.getXi*obj.di;
```

```
end
```

% 3. Solubilidad efectiva

%Función que calcula la solubilidad efectiva del compuesto

```
function Sei = getSei(obj)
```

```
    Sei=obj.getXi*obj.Si;
```

```
end
```

% 4. Solubilidad de la mezcla

%Función que suma las solubilidades de los compuestos

```
function Sti = getSti(obj)
```

```
    Sti= sum(obj.getSei);
```

```
end
```

% 5. Presión efectiva de vapor

%Funcion que calcula la presión efectiva de vapor, como la fracción
%molar del contaminante por la presión de vapor en fase pura del
%compuesto

```
function Vpei = getVpei(obj)
```

```
    Vpei= obj.Xi*obj.Vpi;
```

```
end
```

%6. Coeficiente de partición

```

%Funcion que calcula el coeficiente de partición como la fraccion
%de carbono organico por el coeficiente de particion del suelo
function Kdi = getKdi(obj)

    Kdi= obj.Foc*obj.Koci;

end

%##### FUNCIONES PARA OBTENER TODOS LOS DATOS REFERENTES AL SUELO DE
ESTUDIO #####

% 1. Concentración en poro acuoso en el suelo
%Función que permite calcular la concentracion teorica del
%contaminante en los poros del suelo.
function ICwi = getICwi(obj)

    ar14= obj.Cti*obj.Pb;
    ar15 = obj.getKdi*obj.Pb;
    ar16 = abs(obj.w);
    ar17 = obj.Hi*abs(obj.a);
    sum3 = ar15+ar16+ar17;

    ICwi = ar14/sum3;

end

% 2. Limite de saturación en el suelo
%Función que permite calcular el limite de saturación en el suelo
%para el contaminante en cuestion.
function Csat = getCsat(obj)
    ar1 = obj.Sei/obj.Pb;
    ar2 = obj.Kdi*obj.Pb;
    ar3 = abs(obj.w);
    ar4 = abs(obj.a);
    ar5 = obj.Hi*ar4;
    sum = ar2+ar3+ar5;

    Csat = ar1*sum;

end

% 3. Limite de saturación volumétrico
%Funcion que permite calcular el limite de saturacion volumetrico,
% expresado en mg/L en vez de mg/kg, es decir, se trata del Csati
% multiplicado por Pb (densidad aparente)
function Cconi = getCconi(obj)

    Cconi = obj.getCsat*obj.Pb;

end

% 4. Concentración contaminante en los poros acuosos
%Función que asigna un valor a la concentracion de contaminante en
%los poros ocupados por agua
function Cwi = getCwi(obj)
    Cwi = obj.Si;

```

```
end
```

```
% 5. Concentración Contaminante en vapor de los poros  
%Funcion que calcula la concentracion de contaminante en vapor en  
%los poros
```

```
function Cai = getCai(obj)  
    Cai = obj.getCwi*obj.Hi;
```

```
end
```

```
% 6. Concentración Volumetrica contaminante en vapor poro  
%Funcion que calcula la concentracion volumetrica en el vapor de un  
%contaminante
```

```
function Cavi = getCavi(obj)  
    ar18 = obj.Cai*obj.Ri;  
    ar19 = obj.CFi*obj.Ti;  
    ar20 = ar18*ar19;  
    ar21 = obj.Pi*obj.Mwi;  
    Cavi = ar20/ar21;
```

```
end
```

```
% 7. Concentración Contaminante de sorción en solido  
%Funcion que permite calcular la concentracion de contaminante  
%absorbido por el material solido.
```

```
function Csi = getCsi(obj)  
    Csi = obj.getKdi*obj.getCwi;
```

```
end
```

```
##### FUNCIONES PARA OBTENER TODOS LOS DATOS REFERENTES A LAS MASAS  
CONTAMINANTES #####
```

```
% 1. Masa de contaminante en los poros ocupado por agua  
%Funcion, que a partir de la concentracion de contaminante en poros  
%acuosos, permite calcular la masa de dicho contaminante contenida en  
%los citados poros.
```

```
function Mwii= getMwii(obj)
```

```
    Mwii= (obj.getCwi * obj.w)/obj.Pb;
```

```
end
```

```
% 2. Masa de contaminante en los poros ocupado por aire  
%Funcion, que a partir de la concentracion de contaminante en poros  
%en aire, permite calcular la masa de dicho contaminante contenida en  
%los citados poros.
```

```
function Mai = getMai(obj)
```

```

ar22 = obj.getCwi;
ar23 = abs(obj.a);
ar24 = obj.Hi;
ar25 = ar22*ar23*ar24;
Mai = ar25/obj.Pb;

end

% 3. Masa de contaminante de sorción en el suelo
%Funcion, que a partir de la concentracion de contaminante
%absorbido por el solido
%permite calcular la masa de dicho contaminante contenida en
%los citados poros.
function Msi = getMsi(obj)

    Msi = obj.getCwi*obj.getKdi;

end

% 4. Masa NAPL de contaminante
%Función que permite calcular la masa total de NAPL en fase libre
%del contaminante. Si Cti (masa del contaminante en el suelo) es
%mayor que Csati (limite de saturacion del suelo), no habrá NAPL en
%la muestra, y el programa devolverá 1
function NAPLi = getNAPLi(obj)
ar1 = obj.getSei/obj.Pb;
ar2 = obj.getKdi*obj.Pb;
ar3 = abs(obj.w);
ar4 = abs(obj.a);
ar5 = obj.Hi*ar4;
sum = ar2+ar3+ar5;

Csati2 = ar1*sum;

if obj.Cti>Csati2
    NAPLi = obj.Cti - Csati2;
else
    NAPLi= obj.Mess ;
end

end

% 4. Balance de masa para contaminante Mti=Mai+Msi+Mwi+NAPLi
function Mti = getMti(obj)
    if obj.NAPLi== obj.Mess
        Mti=obj.Mess;
    else

        Mti=obj.Mai+obj.Msi+obj.Mwii+obj.NAPLi;
    end

end

% 5. Masa total de NAPLs
function NAPLt = getNAPLt(obj)

```

```

    if obj.Mti==obj.Mess
        NAPLt=obj.Mess;
    else

    NAPLt=sum(obj.NAPLi);
    end

end

```

% 6. Porcentaje residual saturado de NAPL

```

function Sr = getSr(obj)
    if obj.NAPLt==obj.Mess
        Sr=obj.Mess;
    else

    ar26 = obj.Pb/obj.PNAPL;
    ar27 = obj.NAPLt*ar26;
    Sr = ar27/(obj.nt*100);
    end

```

```
end
```

FUNCIONES PARA OBTENER TODOS LOS VOLÚMENES

% 1. Contenido en volumen NAPL

```

function Nv = getNv(obj)
    if obj.Sr==obj.Mess
        Nv=obj.Mess;
    else

    ar26 = obj.getNAPLt * obj.Pb;
    ar27 = obj.PNAPL;
    Nv = ar26/ar27;
    end

```

```
end
```

% 2. Contenido volumétrico de aire en la muestra

```

function Av = getAv(obj)
    if obj.Nv==obj.Mess
        Av=obj.Mess;
    else

    Av= abs(100*obj.a)-obj.getNv;
    if Av<1
        Av=obj.Mess;
    else
    end
    end

```

```
end
```

% 3. Contenido en Volumen de Agua en la muestra

```

function Vw = getVw(obj)
    if obj.getAv==obj.Mess

```

```

        Vw=obj.Mess;
    else

    Vw = abs(100*obj.w)-obj.getNv;
    if Vw<1
        Vw=obj.Mess;
    else

    end
    end

end

% 4. Contenido en Volumen de suelo en la muestra
function Sv = getSv(obj)
    if obj.Vw==obj.Mess
        Sv=obj.Mess;
    else

    ar28 = abs(obj.w)+abs(obj.a)+obj.getNv;
    Sv = 1.0-ar28;
    end

end

end

end
Filename = 'parametrosentrada.xlsx';
Mess= 'N/A';
sumA=0;
ex= NAPL_sumA(sumA);
max.sumA= ex.nt - ex.sumW;

f = NAPL_sumA (sumA);
f.NAPLi;
while sumA < max.sumA
    sumA = sumA + 0.05;
    f = NAPL_sumA (sumA);
    Res = f.NAPLi
    sumA

end

classdef NAPL_sumW
%% Parámetros Considerados
properties
    Filename = 'parametrosentrada.xlsx' ;%nombre del fichero excel
    Data % = xlsread('parametrosentrada.xlsx'); %Datos contenidos en el fichero

    Mwi; % Peso molecular (g/mol)
    Si; %Limite de solubilidad (mg/l)
    Vpi; %Presion de vapor (mmHg)
    Hi; %Constante de Henry
    Koci; %Coeficiente de carbon organico (L/Kg)
    di; %Densidad (Kg/L)
    Cti; %Masa de contaminante en muestra (mg/Kg)
end

```



```

nt; %Porosidad total
sumW; %contenido volumetrico de agua
sumA; %contenido volumetrico de aire
Foc; %Fraccion carbon organico

```

```
end
```

```
properties
```

```

Wv = 1.0;
dw = 1.0;
a = 0.1;
da = 0.0013;
w = 0.4;
ds = 2.65;
PNAPL = 1.49;
Mess= 'N/A'
Pb = 1.8903; %Densidad aparente, el valor se corresponde con el promedio de valores de un
ensayo experimental aportado por el profesor.
Ri = 8.31451; %constante de los gases
CFi = 1.0*10^6; %factor de conversion
Ti = 293.15; %temperatura
Pi = 101325; %Presion

```

```
end
```

```
properties
```

```
%ACUMULATIVOS DE TODOS LOS CONTAMINANTES
```

```
Mt; %suma de masas de contaminantes
```

```
%##### DATOS A OBTENER SOBRE UN CONTAMINANTE #####
```

```

Contaminante; %Nombre del Contaminante
Xi; %Fraccion molar de contaminantes  $X_i = M_{ci} / M_{ci}$   $M_{ci} = W_{ti} / M_{wi}$   $W_{ti} = C_{ti} / M_t$  ( $M_{wi}, C_{ti}$ )
d; %Densidad de la mezcla  $d = \sum (X_i * d_i)$ 
Sei; %Solubilidad efectiva  $Se_i = X_i * S_i$ 
Sti; %Solubilidad de la mezcla
Vpei; %Presión efectiva de vapor  $VP_{ei} = (X_i * VP_i)$ 
Kdi; %coeficiente de particion

```

```
%##### DATOS A OBTENER SOBRE SUELO DE ESTUDIO #####
```

```

ICwi; %Concentración en poro acuoso en el suelo
Csat; %Limite de saturación en el suelo
Cconi; %Limite de saturación volumétrico
Cwi; %Concentración contaminante en los poros acuosos
Cai; %Concentración Contaminante en vapor de los poros
Cavi; %Concentración Volumetrica contaminante en vapor poro
Csi; %Concentración Contaminante de sorción en solido

```

```
%##### DATOS A OBTENER CÁLCULOS DE MASAS DE CONTAMINANTES #####
```

```

Mwii; %Masa de contaminante en los poros ocupados por agua
Mai; %Masa de contaminante en los poros ocupado por aire
Msi; %Masa de contaminante de sorción en el suelo
NAPLi; %Masa NAPL de contaminante
Mti; %Balance de masa para contaminante  $M_{ti} = M_{ai} + M_{si} + M_{wi} + NAPL_i$ 
NAPLt; %Masa total de NAPLs
Sr; %Porcentaje residual saturado de NAPL

```

```
%##### DATOS A OBTENER CÁLCULOS DE VOLÚMENES #####
```

```

Nv; %Contenido en volumen NAPL
Av; %Contenido volumétrico de aire en la muestra
Vw; %Contenido en Volumen de Agua en la muestra
Sv; %Contenido en Volumen de suelo en la muestra

```

end

%% Métodos
methods

FUNCIÓN PARA OBTENER TODOS LOS DATOS DEL ARCHIVO

function obj = NAPL_sumW(sumW)

% Definición del objeto

```
obj.Mwi = xlsread(obj.FileName, 'B2:B2'); % Peso molecular (g/mol)
obj.Si = xlsread(obj.FileName, 'C2:C2'); %Limite de solubilidad (mg/l)
obj.Vpi = xlsread(obj.FileName, 'D2:D2'); %Presion de vapor (mmHg)
obj.Hi = xlsread(obj.FileName, 'E2:E2'); %Constante de Henry
obj.Koci = xlsread(obj.FileName, 'F2:F2'); %Coeficiente de carbon organico (L/Kg)
obj.di = xlsread(obj.FileName, 'G2:G2'); %Densidad (Kg/L)
obj.Cti = xlsread(obj.FileName, 'H2:H2'); %Masa de contaminante en muestra (mg/Kg)
obj.nt = 0.6 ; %Porosidad Total
obj.sumW = sumW ; %Contenido Volumetrico de agua
obj.sumA = xlsread(obj.FileName, 'K2:K2'); %contenido volumetrico de aire
obj.Foc = xlsread(obj.FileName, 'L2:L2'); %Fraccion carbon organico
```

```
obj.Mt = obj.getMt;
obj.Xi = obj.getXi;
obj.d = obj.getd;
obj.Sei = obj.getSei ;
obj.d = obj.getd;
obj.Sti = obj.getSti;
obj.Vpei = obj.getVpei;
obj.Kdi = obj.getKdi;
obj.lCwi = obj.getlCwi;
obj.Csat = obj.getCsat;
obj.Cconi = obj.getCconi;
obj.Cwi = obj.getCwi;
obj.Cai = obj.getCai;
obj.Cavi = obj.getCavi;
obj.Csi = obj.getCsi;
obj.Mwii = obj.getMwii;
obj.Mai = obj.getMai;
obj.Msi = obj.getMsi;
obj.NAPLi = obj.getNAPLi;
obj.Mti = obj.getMti;
obj.NAPLt = obj.getNAPLt;
obj.Sr = obj.getSr;
obj.Nv = obj.getNv;
obj.Av = obj.getAv;
obj.Vw = obj.getVw;
obj.Sv = obj.getSv;
```

end

%Funcion para calcular la suma de masas de contaminantes

function Mt= getMt(obj)

Mt=sum(obj.Cti);

end

FUNCIONES PARA OBTENER TODOS LOS DATOS DEL CONTAMINANTE

```
% 1. Fraccion molar de contaminantes
%Función que calcula la fraccion molar de contaminante en el total
%de masa de contaminantes del suelo.
```

```
function Xi = getXi(obj)
```

```
sumMwi= sum(obj.Mwi) ;
Wti= obj.Cti/sumMwi;
Mci = Wti/obj.Mwi;
Xi=Mci/abs(Mci);
```

```
end
```

```
% 2. Densidad de la mezcla
%Función que calcula la densidad de la mezcla
```

```
function d = getd(obj)
```

```
d= obj.getXi*obj.di;
```

```
end
```

```
% 3. Solubilidad efectiva
%Función que calcula la solubilidad efectiva del compuesto
```

```
function Sei = getSei(obj)
```

```
Sei=obj.getXi*obj.Si;
```

```
end
```

```
% 4. Solubilidad de la mezcla
%Función que suma las solubilidades de los compuestos
```

```
function Sti = getSti(obj)
```

```
Sti= sum(obj.getSei);
```

```
end
```

```
% 5. Presión efectiva de vapor
%Funcion que calcula la presión efectiva de vapor, como la fracción
%molar del contaminante por la presion de vapor en fase pura del
%compuesto
```

```
function Vpei = getVpei(obj)
```

```
Vpei= obj.Xi*obj.Vpi;
```

```
end
```

```
%6. Coeficiente de partición
%Funcion que calcula el coeficiente de partición como la fraccion
%de carbono organico por el coeficiente de particion del suelo
```

```
function Kdi = getKdi(obj)
```

```
Kdi= obj.Foc*obj.Koci;
```

```
end
```

```
%##### FUNCIONES PARA OBTENER TODOS LOS DATOS REFERENTES AL SUELO DE
```

ESTUDIO

```
% 1. Concentración en poro acuoso en el suelo
%Función que permite calcular la concentracion teorica del
%contaminante en los poros del suelo.
function ICwi = getICwi(obj)

    ar14= obj.Cti*obj.Pb;
    ar15 = obj.getKdi*obj.Pb;
    ar16 = abs(obj.w);
    ar17 = obj.Hi*abs(obj.a);
    sum3 = ar15+ar16+ar17;

    ICwi = ar14/sum3;

end

% 2. Limite de saturación en el suelo
%Función que permite calcular el limite de saturación en el suelo
%para el contaminante en cuestion.
function Csat = getCsat(obj)
    ar1 = obj.Sei/obj.Pb;
    ar2 = obj.Kdi*obj.Pb;
    ar3 = abs(obj.w);
    ar4 = abs(obj.a);
    ar5 = obj.Hi*ar4;
    sum = ar2+ar3+ar5;

    Csat = ar1*sum;

end

% 3. Limite de saturación volumétrico
%Funcion que permite calcular el limite de saturacion volumetrico,
% expresado en mg/L en vez de mg/kg, es decir, se trata del Csati
% multiplicado por Pb (densidad aparente)
function Cconi = getCconi(obj)

    Cconi = obj.getCsat*obj.Pb;

end

% 4. Concentración contaminante en los poros acuosos
%Función que asigna un valor a la concentracion de contaminante en
%los poros ocupados por agua
function Cwi = getCwi(obj)
    Cwi = obj.Si;

end

% 5. Concentración Contaminante en vapor de los poros
%Funcion que calcula la concentracion de contaminante en vapor en
%los poros
function Cai = getCai(obj)
    Cai = obj.getCwi*obj.Hi;
```

end

% 6. Concentración Volumetrica contaminante en vapor poro
%Funcion que calcula la concentracion volumetrica en el vapor de un
%contaminante

```
function Cavi = getCavi(obj)
    ar18 = obj.Cai*obj.Ri;
    ar19 = obj.CFi*obj.Ti;
    ar20 = ar18*ar19;
    ar21 = obj.Pi*obj.Mwi;
    Cavi = ar20/ar21;
```

end

% 7. Concentración Contaminante de sorción en solido
%Funcion que permite calcular la concentracion de contaminante
%absorbido por el material solido.

```
function Csi = getCsi(obj)
    Csi = obj.getKdi*obj.getCwi;
```

end

FUNCIONES PARA OBTENER TODOS LOS DATOS REFERENTES A LAS MASAS
CONTAMINANTES #####

% 1. Masa de contaminante en los poros ocupado por agua
%Funcion, que a partir de la concentracion de contaminante en poros
%acuosos, permite calcular la masa de dicho contaminante contenida en
%los citados poros.

```
function Mwii= getMwii(obj)

    Mwii= (obj.getCwi * obj.w)/obj.Pb;
```

end

% 2. Masa de contaminante en los poros ocupado por aire
%Funcion, que a partir de la concentracion de contaminante en poros
%en aire, permite calcular la masa de dicho contaminante contenida en
%los citados poros.

```
function Mai = getMai(obj)

    ar22 = obj.getCwi;
    ar23 = abs(obj.a);
    ar24 = obj.Hi;
    ar25 = ar22*ar23*ar24;
    Mai = ar25/obj.Pb;
```

end

```

% 3. Masa de contaminante de sorción en el suelo
%Funcion, que a partir de la concentracion de contaminante
%absorbido por el solido
%permite calcular la masa de dicho contaminante contenida en
%los citados poros.
function Msi = getMsi(obj)

    Msi = obj.getCwi*obj.getKdi;

end

% 4. Masa NAPL de contaminante
%Función que permite calcular la masa total de NAPL en fase libre
%del contaminante. Si Cti (masa del contaminante en el suelo) es
%mayor que Csati (limite de saturacion del suelo), no habrá NAPL en
%la muestra, y el programa devolverá 1
function NAPLi = getNAPLi(obj)
    ar1 = obj.getSei/obj.Pb;
    ar2 = obj.getKdi*obj.Pb;
    ar3 = abs(obj.w);
    ar4 = abs(obj.a);
    ar5 = obj.Hi*ar4;
    sum = ar2+ar3+ar5;

    Csati2 = ar1*sum;

    if obj.Cti>Csati2
        NAPLi = obj.Cti - Csati2;
    else
        NAPLi= obj.Mess ;
    end

end

% 4. Balance de masa para contaminante Mti=Mai+Msi+Mwi+NAPLi
function Mti = getMti(obj)
    if obj.NAPLi== obj.Mess
        Mti=obj.Mess;
    else

        Mti=obj.Mai+obj.Msi+obj.Mwii+obj.NAPLi;
    end

end

% 5. Masa total de NAPLs
function NAPLt = getNAPLt(obj)
    if obj.Mti==obj.Mess
        NAPLt=obj.Mess;
    else

        NAPLt=sum(obj.NAPLi);
    end

end

```

```
% 6. Porcentaje residual saturado de NAPL
```

```
function Sr = getSr(obj)  
    if obj.NAPLt==obj.Mess  
        Sr=obj.Mess;  
    else
```

```
        ar26 = obj.Pb/obj.PNAPL;  
        ar27 = obj.NAPLt*ar26;  
        Sr = ar27/(obj.nt*100);  
    end
```

```
end
```

```
%#### FUNCIONES PARA OBTENER TODOS LOS VOLÚMENES ####
```

```
% 1. Contenido en volumen NAPL
```

```
function Nv = getNv(obj)  
    if obj.Sr==obj.Mess  
        Nv=obj.Mess;  
    else
```

```
        ar26 = obj.getNAPLt * obj.Pb;  
        ar27 = obj.PNAPL;  
        Nv = ar26/ar27;  
    end
```

```
end
```

```
% 2. Contenido volumétrico de aire en la muestra
```

```
function Av = getAv(obj)  
    if obj.Nv==obj.Mess  
        Av=obj.Mess;  
    else
```

```
        Av= abs(100*obj.a)-obj.getNv;  
        if Av<1  
            Av=obj.Mess;  
        else  
            end  
        end
```

```
end
```

```
% 3. Contenido en Volumen de Agua en la muestra
```

```
function Vw = getVw(obj)  
    if obj.getAv==obj.Mess  
        Vw=obj.Mess;  
    else
```

```
        Vw = abs(100*obj.w)-obj.getNv;  
        if Vw<1  
            Vw=obj.Mess;  
        else
```

```
            end  
        end
```

```

end

% 4. Contenido en Volumen de suelo en la muestra
function Sv = getSv(obj)
    if obj.Vw==obj.Mess
        Sv=obj.Mess;
    else

        ar28 = abs(obj.w)+abs(obj.a)+obj.getNv;
        Sv = 1.0-ar28;
    end

end

end

end

end
Filename = 'parametrosentrada.xlsx';
Mess= 'N/A';
sumW=0;
ex= NAPL_sumW(sumW);
max.sumW= ex.nt - ex.sumA;

f = NAPL_sumW (sumW);
f.NAPLi;
while sumW < max.sumW
    sumW = sumW + 0.05;
    f = NAPL_sumW (sumW);
    Res = f.NAPLi
    sumW

end

```