

**FACULDADE DE ENGENHARIA DA UNIVERSIDADE DO
PORTO**

Departamento de Engenharia Civil

**SIMULAÇÃO NUMÉRICA DA TRANSFERÊNCIA DE
HUMIDADE EM MATERIAIS DE CONSTRUÇÃO**

Maria Clara de Carvalho Pimenta do Vale

Licenciada em Arquitectura pela Faculdade de Arquitectura da Universidade do
Porto

Dissertação submetida para a satisfação parcial dos
requisitos do grau de mestre
em
Construção de Edifícios

Dissertação realizada sob a supervisão de

Professor Doutor Vasco Manuel Araújo Peixoto de Freitas
do Departamento de Engenharia Civil
da Faculdade de Engenharia da Universidade do Porto

Porto, Maio 1999

*Ao meu marido, Nelson,
à minha filha, Raquel,
aos meus Pais, Joaquim e Maria José,
e aos meus irmãos, Zé e Ricardo*

AGRADECIMENTOS

Gostaria, em primeiro lugar, agradecer ao Prof. Vasco Freitas, pelo apoio e disponibilidade sempre demonstrada. Pelos seus conselhos, pela sua atenção e pela sua confiança o meu mais sentido agradecimento

Gostaria também de agradecer ao Nelson, meu marido, pela crítica construtiva e pelo apoio sempre dado, especialmente na aprendizagem de programação, à minha filha, por ter compreendido a importância deste trabalho, e me ter dispensado muitas vezes do meu papel de mãe. Gostaria ainda de agradecer ao Arq. Guilherme Guimarães pela disponibilização do equipamento informático da empresa e pela compreensão demonstrada.

Por último, uma nota de gratidão aos membros da minha família, nomeadamente aos meus pais, meus irmãos e meus sogros, pelo apoio que me concederam neste período atarefado e conturbado da minha vida.

RESUMO

O estudo da transferência conjunta de calor e umidade em meios porosos é uma tarefa importante para melhor conhecer e compreender o comportamento das paredes e dos materiais aplicados nos edifícios ao longo do tempo. A simulação numérica destes fenómenos é um bom auxiliar pois possibilita um conhecimento maior com um menor dispêndio de tempo, e conseqüentemente com menores custos envolvidos.

Neste trabalho, o modelo de Luikov e Philip - De Vries, de transferência de calor e umidade em meios porosos, é tomado como referência, associada à formulação matemática definida por V. Peixoto de Freitas, para a elaboração de um programa de simulação numérica do comportamento de materiais quando sujeitas as gradientes de temperatura e umidade, para funcionar num computador pessoal em Ambiente *Windows*.

Nesta aplicação apostou-se fundamentalmente nos sistemas de comunicação com o utilizador, quer ao nível da entrada de dados, que se pretende o mais intuitiva possível, quer ao nível da visualização dos resultados, com recurso a gráficos que permitem, num relance, abranger quase a totalidade dos elementos envolvidos na simulação.

Foram feitas diversas simulações com vista à validação do programa, por comparação com casos disponíveis em bibliografia. Esta comparação permite considerar o programa *TrHum 98* como uma ferramenta de simulação numérica da transferência de calor e umidade em materiais de construção, vocacionada para a análise de paredes. Tem ainda restrições ao nível da quantidade de materiais disponíveis e números de camadas da parede, aspectos que poderão ser ampliados em versões futuras.

Palavras chave: Umidade; Transferência; Modelo; Programa

ABSTRACT

Moisture and heat transfer in porous media studies are an important task to better know and understand the behaviour of building envelopes during time. Numerical analyses of this phenomena can be very helpful because make possible to know more about it with less time.

In this work, we took Luikov and De Vries model of heat and moisture transfer, with the algorithm already established by V. Freitas on is computer program Trhumidade, and we made a Windows based new program for numerical simulation of construction materials behaviour under moisture and temperature gradients.

This is an user friendly input program, with an graphical interface that gives an entire overview of the simulations conditions, with a single look.

We made some simulations, first to establish the parameter sensibility factors, and then to study some real based situations.

RESUME

L'étude des transferts couplés de chaleur et d'humidité en milieu poreux, c'est un travail important pour mieux connaître et comprendre le comportement des murs des édifices. La simulation numérique de ce phénomène c'est une bonne aide parce qu'il nous donne la possibilité de mieux connaître avec moins de temps nécessaire.

Dans ce travail nous prenons le modèle de Luikov et De Vries, de transferts de chaleur et d'humidité en milieu poreux, avec la formulation définie par V. Freitas, dans son programme d'ordinateur Trhumidade, et on a écrit un nouveau programme, pour Windows, pour la simulation numérique du comportement des murs des édifices, quand sont soumis à gradients de température et d'humidité.

On a fait des simulations, en premier lieu, pour établir la sensibilité des paramètres, et en suite, pour étudier des situations réelles.

ÍNDICES

ÍNDICE GERAL DO TRABALHO

AGRADECIMENTOS	III
RESUMO	IV
ABSTRACT	V
RÉSUMÉ.....	VI
ÍNDICES	VII
ÍNDICE GERAL DO TRABALHO	VII
ÍNDICE DE FIGURAS	X
ÍNDICE DE QUADROS	XV
1 INTRODUÇÃO.....	2
1.1 ENQUADRAMENTO DO TRABALHO	2
1.2 OBJECTIVO DO TRABALHO.....	3
1.3 MOTIVAÇÃO	4
1.4 METODOLOGIA, ORGANIZAÇÃO E ESTRUTURA DO TEXTO	5
2 MODELIZAÇÃO DA TRANSFERÊNCIA DE CALOR E HUMIDADE EM MATERIAIS DE CONSTRUÇÃO	8
2.1 MODELOS TEÓRICOS DE TRANSFERÊNCIA	8
2.1.1 <i>Modelo de Glaser</i>	10
2.1.2 <i>Modelo de Krischer-Vos</i>	10
2.1.3 <i>Modelo de Luikov e Philip /De Vries</i>	11
2.2 DESCRIÇÃO MATEMÁTICA DO MODELO DE LUIKOV E PHILIP /DE VRIES.....	11
2.3 PROGRAMAS DE CÁLCULO EXISTENTES	14
2.4 EVOLUÇÃO DO PROGRAMA TRHUMIDADE	17
2.4.1 <i>Trhumidade</i>	17
2.4.2 <i>TrHum 1.0 / TrHum 96</i>	19
2.5 TRHUM 98	21
3 CONCEPÇÃO DO PROGRAMA.....	24
3.1 OBJECTIVOS.....	24
3.2 ESTRUTURA DO PROGRAMA	26
3.2.1 <i>Funções de entrada, edição e visualização de informação</i>	27
3.2.2 <i>Funções de manuseamento de ficheiros</i>	27
3.2.3 <i>Funções de Cálculo</i>	27

3.3	CONCEPÇÃO DA INTERFACE COM O UTILIZADOR	28
3.3.1	<i>Entrada de dados</i>	28
3.3.2	<i>Apresentação de resultados</i>	32
3.4	MODELO MATEMÁTICO	33
3.4.1	<i>Estrutura dos módulos de cálculo</i>	35
3.4.2	<i>Resolução do sistema de equações</i>	35
3.4.3	<i>Cálculo do teor de humidade em função da humidade relativa</i>	37
4	DESCRIÇÃO DO PROGRAMA.....	40
4.1	CARACTERIZAÇÃO GERAL.....	40
4.2	MÓDULO 1 - SIMULAÇÃO NUMÉRICA DA TRANSFERÊNCIA DE CALOR E HUMIDADE	40
4.2.1	<i>Dados necessários</i>	40
4.2.2	<i>Definição das condições iniciais</i>	47
4.3	MÓDULO 2 – CÁLCULO DA PRESSÃO SATURAÇÃO.....	59
4.4	MÓDULO 3 – CÁLCULO DO TEOR DE HUMIDADE DE UM MATERIAL EM FUNÇÃO DA HUMIDADE RELATIVA	61
4.5	MÓDULO 4 – CÁLCULO DA RESISTÊNCIA TÉRMICA DE UM MATERIAL EM FUNÇÃO DO SEU TEOR DE HUMIDADE.....	62
4.6	MÓDULO 5 – CÁLCULO DOS PERFIS DE TEMPERATURA ESTABILIZADOS PARA PAREDES COM DIVERSAS CAMADAS.....	63
4.7	MÓDULO 6 – CÁLCULO DA TEMPERATURA SUPERFICIAL DE UMA PAREDE EM FUNÇÃO DA LOCALIZAÇÃO, RADIAÇÃO E VENTO.....	64
4.8	MÓDULO 7 – CONVERSÃO DE UNIDADES.....	67
4.9	POTENCIALIDADES E LIMITAÇÕES DO PROGRAMA	68
5	SIMULAÇÃO NUMÉRICA	72
5.1	ANÁLISE DE SENSIBILIDADE.....	72
5.1.1	<i>Parâmetros fixos</i>	72
5.1.2	<i>Análise do passo da simulação</i>	73
5.1.3	<i>Análise das Condições Iniciais</i>	79
5.1.4	<i>Análise das Condicionantes climáticas</i>	88
5.1.5	<i>Simulação de situações de referência</i>	95
5.1.6	<i>Análise dos resultados da simulação</i>	100
6	CONCLUSÕES.....	102
6.1	CONSIDERAÇÕES FINAIS	102
6.2	SÍNTESE CRÍTICA DOS RESULTADOS	102
6.3	DESENVOLVIMENTO FUTURO DA INVESTIGAÇÃO NESTE DOMÍNIO.....	103
7	LISTA DE REFERÊNCIAS BIBLIOGRÁFICAS.....	105
	ANEXO A FICHEIROS DO PROGRAMA TRHUM 98.....	109

A.1	FICHEIROS DE PROJECTO – (*.HUM).....	109
A.2	FICHEIROS DE RESULTADOS - (*.RES).....	115
A.3	FICHEIROS DE RESULTADOS - (*.CSV).....	119
A.4	FICHEIROS DA BASE DE DADOS DE MATERIAIS (*.DBK)	120
A.5	FICHEIROS DE CLIMA (*.WET)	125
ANEXO B LISTAGEM PARCIAL DO PROGRAMA.....		127
B.1	DEFINIÇÃO DO PROGRAMA.....	127
B.2	LISTAGEM DE “MODULES”	128
B.2.1.	Módulo “trh98.bas”	128
B.2.2.	Módulo “functions.bas”	130
B.2.3.	Módulo “n_functions.bas”	135
B.2.4.	Módulo “o_functions.bas”	140
B.2.5.	Módulo “DataFunctions.bas”	147
B.2.6.	Módulo “conv.bas”	158
B.3	LISTAGEM DE “FORMS”	161
B.3.1.	“Form início”	162
B.3.2.	“Form frmmodules “	163
B.3.3.	“Form frmsat”	164
B.3.4.	“Form frmmoisture”	166
B.3.5.	“Form frmheat”	168
B.3.6.	“Form frmtemp”	170
B.3.7.	“Form frmsol”	171
B.3.8.	“Form frmconvert”	173
B.3.9.	“Form frmmain”	176
B.3.10.	Sequência de menus do “form frmmain”	179
B.3.11.	“Form frmmainwindow”	182
B.3.12.	“Form frmobjectname”	189
B.3.13.	“Form frmobjectdefinition”	190
B.3.14.	“Form frmtime-r”	194
B.3.15.	“Form frminitialc”	196
B.3.16.	“Form frminitialprofil”	197
B.3.17.	“Form frmmainweather”	200
B.3.18.	“Form frmsteadyweather”	202
B.3.19.	“Form frmdynamicweather”	203
B.3.20.	“Form frmimport”	207
B.3.21.	“Form material”	209
B.3.22.	“Form frmreview”	210
B.3.23.	“Form frmresults”	214

ÍNDICE DE FIGURAS

Fig. 1 - Diagrama psicrométrico.....	9
Fig. 2 - Organigrama do programa Trhumidade [1].....	18
Fig. 3 - Janela de entrada do programa TrHum 1. 0 [5].....	19
Fig. 4 – Apresentação de resultados do TrHum 96 2.0 [5].....	20
Fig. 5 – Ajuda Automática	21
Fig. 6 - Janela de entrada do programa TrHum 98	22
Fig. 7 - Janela de escolha múltipla	26
Fig. 8 – Sistema de menus	26
Fig. 9 – Exemplo de uma janela encadeada.....	29
Fig. 10 - Janela de tarefas	30
Fig. 11 – Sistema de menus e sub-menus e definição de atalhos.....	31
Fig. 12 – Vista do gráfico de saída de resultados	32
Fig. 13 – Relação entre os diversos módulos do programa	40
Fig. 14 – Relação da janela de tarefas com as diversas janelas de entrada de dados no programa.....	42
Fig. 15 - Janela de definição do projecto.....	44
Fig. 16 - Janela de definição do objecto – situação de 3 sub-camadas	45
Fig. 17 - Janela de definição do objecto – situação de 1 só sub-camada	46
Fig. 18 - Janela de definição do tempo de simulação	47
Fig. 19 - Janela de definição de condições constantes.....	48
Fig. 20 - Janela de definição dos perfis iniciais da amostra.....	49
Fig. 21 - Janela de definição do tipo de ficheiro de clima	50
Fig. 22 - Janela de definição de clima constante ao longo da simulação.....	50
Fig. 23 - Janela da definição de clima variável.....	51
Fig. 24 – Janela de Acesso às diferentes funções relacionadas com o manuseamento de bases de dados de características dos materiais.	52
Fig. 25 – Janela de ligação às funções relacionadas com o clima.	53
Fig. 26 – Janela de verificação dos dados.....	54

Fig. 27 – Janela de tarefas	55
Fig. 28 – Janela de mensagem – Informação da conclusão do cálculo.....	56
Fig. 29 – Janela de visualização de resultados.....	56
Fig. 30 – Janela de visualização de resultados – primeira forma de apresentação	57
Fig. 31 – Janela de visualização de resultados – segunda forma de apresentação.....	57
Fig. 32 – Janela de visualização de resultados – terceira forma de apresentação	58
Fig. 33 – Janela inicial do cálculo da pressão de saturação	59
Fig. 34 – Pressão de vapor para várias humidades relativas.....	59
Fig. 35 – Diagrama psicrométrico para o intervalo de temperaturas fornecido.....	60
Fig. 36 – Teor de humidade em função da humidade relativa.....	61
Fig. 37 – Resistência térmica em função do teor de humidade do material.....	62
Fig. 38 – Janela de definição das diferentes camadas de uma parede	63
Fig. 39 – Janela inicial – cálculo para uma hora específica.....	64
Fig. 40 – Janela com gráfico da variação ao longo do tempo.....	65
Fig. 41 – Janela de conversão de unidades	67
Fig. 42 – Possibilidade de aplicação do programa TrHum 98 - Secagem [1]	68
Fig. 43 - Possibilidade de aplicação do programa TrHum 98 - Embebição [1]	69
Fig. 44 - Possibilidade de aplicação do programa TrHum 98 – na análise do fenómeno das condensações internas [1].....	70
Fig. 45 – Evolução dos perfis do teor de humidade (passo de simulação 0.001 horas).....	74
Fig. 46 – Análise comparativa do teor de humidade do material em função do passo da simulação (perfil efectuado a 3 cm do exterior do material).....	75
Fig. 47 – Evolução dos perfis do teor de humidade para uma simulação de 1440 horas (passo de simulação 0.05 horas)	76
Fig. 48 – Localização do erro entre simulações de diferente passo (0,5 e 0,05 horas).....	77
Fig. 49 - Variação da temperatura do betão celular ao longo de 48 horas - temperatura inicial 10°C....	80
Fig. 50 - Variação da temperatura do betão celular ao longo de 48 horas - temperatura inicial 40°C....	81
Fig. 51 - Variação do teor de humidade do betão celular ao longo de 240 horas – teor de humidade inicial de 0.6 kg/kg – Humidade relativa da ambiência de 50%	82
Fig. 52 - Variação do teor de humidade do betão celular ao longo de 240 horas – teor de humidade inicial de 0.2 kg/kg – Humidade relativa da ambiência de 50%	83

Fig. 53- Variação do teor de humidade ao longo do tempo - 480 horas – teor de humidade inicial de 0.6 kg/kg – humidade relativa 90 %	83
Fig. 54- Variação do teor de humidade ao longo do tempo - 480 horas – teor de humidade inicial de 0.6 kg/kg – humidade relativa 10 %	84
Fig. 55 – Variação do teor de humidade da amostra – Teor inicial 0.0225 kg/kg – Humidade relativa 10%	85
Fig. 56 – Variação do teor de humidade da amostra – Teor inicial 0.0525 kg/kg – Humidade relativa 70%	85
Fig. 57 – Variação do teor de humidade da amostra – Teor inicial 0.1125 kg/kg – Humidade relativa 90%	86
Fig. 58 – Variação do teor de humidade da amostra – Teor inicial 0.215 – Humidade relativa 10%.....	86
Fig. 59 – Variação do teor de humidade da amostra – Teor inicial 0.235 kg/kg – Humidade relativa 70%	87
Fig. 60 – Variação do teor de humidade da amostra – Teor inicial 0.275 – Humidade relativa 90%.....	88
Fig. 61 - Influência da Humidade relativa na secagem do betão – ponto a 3 cm do exterior do material	89
Fig. 62- Secagem diferencial do betão celular em função da humidade relativa (1%-99%)	90
Fig. 63- Secagem diferencial do betão celular em função da humidade relativa (10%-50%)	90
Fig. 64- Secagem do betão celular em função da humidade relativa (10%-10%)	91
Fig. 65- Secagem do betão celular em função da humidade relativa (10%-90%)	91
Fig. 66- Secagem do betão celular em função da humidade relativa (90%-90%)	92
Fig. 67- Secagem diferencial do barro vermelho em função da humidade relativa (10%-90%)	92
Fig. 68 – Deslocamento dos perfis do teor de humidade para o interior do material em consequência do efeito da radiação	93
Fig. 69 – Humidificação do betão celular por chuva incidente (humidade relativa interior 50%)	94
Fig. 70 – Humidificação do betão celular por chuva incidente (humidade relativa interior 95%)	94
Fig. 71 – Secagem diferencial do betão celular sob gradiente de temperaturas (5-45°C) - Trhumidade [1].....	95
Fig. 72 – Secagem diferencial do betão celular sob gradiente de temperaturas (5-45°C) – TrHum 98..	95
Fig. 73 – Embebição diferencial do betão celular em função da humidade relativa (10%-90%) e da temperatura (0.2°C – 45°C) – Trhumidade [1]	96
Fig. 74 – Embebição diferencial do betão celular em função da humidade relativa (10%-90%) e da temperatura (0.2°C – 45°C) – TrHum 98	96

Fig. 75 – Humidificação do barro vermelho por chuva incidente – Trhumidade [1]	97
Fig. 76 – Humidificação do barro vermelho por chuva incidente – TrHum 98	97
Fig. 77 – Processo de embebição e secagem – Trhumidade [1]	98
Fig. 78 – Processo de embebição e secagem – TrHum 98.....	98
Fig. 79 – Humedecimento por condensação junto à superfície fria – Trhumidade [1].....	99
Fig. 80 – Humedecimento por condensação junto à superfície fria – TrHum 98	99
Fig. 81 - Esquema de ligações entre os diferentes “forms”	161
Fig. 82 – “Form início”	162
Fig. 83 – “Form frmmodules”	163
Fig. 84 – “Form frmsat”	164
Fig. 85 – “Form frmmoisture”	166
Fig. 86 – “Form frmheat”	168
Fig. 87 – “Form frmtemp”	170
Fig. 88 – “Form frmsol”	171
Fig. 89 – “Form frmconvert”	173
Fig. 90 - Esquema de ligações entre "forms"	175
Fig. 91 – “Form frmmain”	176
Fig. 92 – menu “File”	179
Fig. 93 – menu “Define”	179
Fig. 94 – menu “Weather”	180
Fig. 95 – menu “Material – Standard data Bank”	180
Fig. 96 – menu “Material – Current Project Data Bank”	180
Fig. 97 – menu “View”	181
Fig. 98 – menu “Calculate”	181
Fig. 99 – menu “Utility”	181
Fig. 100 – menu “Help”	182
Fig. 101 – “Form frmmainindow”	182
Fig. 102 – “Form frmobjectname”	189
Fig. 103 – “Form frmobjectdefinition”	190
Fig. 104 – “Form frmtime-r”	194

Fig. 105 – “Form frinitialc”	196
Fig. 106 – “Form frinitialprofil”	197
Fig. 107 – “Form frmmainweather”	200
Fig. 108 – “Form frmsteadyweather”	202
Fig. 109 – “Form frmdynamicweather”	203
Fig. 110 – “Form frmimport”	207
Fig. 111 – “Form material”	209
Fig. 112 – “Form frmreview”	210
Fig. 113 – “Form frmresults”	214

ÍNDICE DE QUADROS

Quadro 1 - Resumo do Inquérito -Annex 24 - Janeiro de 1993 [7].....	16
Quadro 2 – Diagrama de blocos base do TrHum 98.....	25
Quadro 3 – Definição dos parâmetros de caracterização do teste da variação do intervalo de tempo....	74
Quadro 4 – Comparação entre resultados e respectivos desvios para diferentes passos de simulação...76	
Quadro 5 – Comparação entre o passo da simulação e o tempo de simulação	77
Quadro 6 – Comparação entre diferentes processadores	78
Quadro 7 –Definição dos parâmetros de caracterização do teste da distribuição inicial de temperaturas	80
Quadro 8 –Definição dos parâmetros de caracterização no teste da distribuição inicial de teores de humidade.....	82
Quadro 9 –Valores do teor de humidade em função da humidade relativa	84

1 - INTRODUÇÃO 2

1.1 -	ENQUADRAMENTO DO TRABALHO.....	2
1.2 -	OBJECTIVO DO TRABALHO.....	3
1.3 -	MOTIVAÇÃO.....	4
1.4 -	METODOLOGIA, ORGANIZAÇÃO E ESTRUTURA DO TEXTO	5

1 Introdução

1.1 Enquadramento do trabalho

Nas últimas décadas, os computadores passaram de máquinas de utilização quase exclusiva de cientistas e técnicos, pela complexidade e custo que o seu uso envolvia, a ferramentas de uso generalizado e com uma acessibilidade independente das habilitações ou faixa etária de quem os utiliza.

Para tal muito contribuíram, por um lado o aparecimento dos computadores pessoais, vulgo PC, com dimensões e preços compatíveis com uma utilização mais ampla, e por outro lado, a evolução dos programas (*software*), num sentido de privilegiarem mais a relação com o utilizador.

Os programas desenvolvidos para operarem em ambiente *Windows*, tiveram uma influência determinante na implantação dos meios informáticos como ferramentas de trabalho e mesmo como meio de lazer e entretenimento.

O grande salto dá-se sobretudo com o *Windows 95*, que marcou a passagem entre programas que embora já com algum cuidado no interface gráfico, trabalhavam ainda sobre um sistema operativo baseado numa consola de texto, e uma renovada geração de sistemas operativos gráficos, onde todos os comandos e acções são mostradas ao utilizador por meio de símbolos e ícones.

Esta transformação permitiu a rápida entrada dos computadores na nossa realidade, no nosso sistema de referências. A diferença entre um écran “negro” onde todas as instruções tem que ser dadas por códigos numa linha de comando, e um espaço de trabalho que pode ser arrumado ao nosso gosto e em que conseguimos visualizar acções que de outro modo seriam impenetráveis, é abismal.

Foi inserindo-se e acompanhando esta tendência que o programa *TrHum 98* foi desenvolvido, adaptado a sistemas informáticos de uso generalizado, e com uma aposta forte na interface gráfica, no sentido de auxiliar (e conquistar), o utilizador no uso de uma ferramenta que poderá revelar-se importante na sua actividade profissional. O utilizador poderá ser alguém fora da esfera da investigação, sem grandes conhecimentos científicos sobre esta área, sem grande perícia na utilização dos sistemas informáticos, mas que usará o

programa para necessidades práticas da sua vida profissional, por exemplo, para saber quanto tempo deverá deixar secar um reboco antes de aplicar uma pintura.

O programa foi escrito em *Visual Basic 5.0*, para trabalhar num computador pessoal, sobre sistema operativo *Windows 95*. A escolha desta plataforma prendeu-se com o objectivo atrás referido de tornar este programa uma ferramenta de fácil acesso e de uso generalizado, pelo que o mesmo deveria funcionar em sistemas informáticos de uso corrente.

Recomenda-se a utilização de um computador com processador *Pentium* a 200 MHz, com 64 MB de memória *RAM*, e uma resolução gráfica de 1024x768 . No entanto o programa foi testado com outras configurações inferiores que não limitaram o seu funcionamento.

O programa funciona sobre sistema operativo *Windows*; tendo sido desenvolvido para *Windows 95*, mas poderá igualmente ser executado sobre *Windows 98* ou *Windows NT*.

1.2 Objectivo do Trabalho

O deficiente conhecimento dos mecanismos de embebição e secagem dos diversos materiais de construção usados correntemente na construção, por um lado, e a necessidade de encurtar prazos de obra, por outro, torna o estudo da embebição e secagem dos materiais de construção uma área de estudo a desenvolver e na qual se deve apostar, com vista ao estabelecimento de informação sistematizada que possibilite a garantia de um mínimo de qualidade na execução de diversos trabalhos de construção civil.

É importante estudar os mecanismos de transferência de humidade entre os materiais de construção e a envolvente, quer essa humidade resulte da água necessária à trabalhabilidade do material na fase de construção, quer resulte de infiltrações das águas das chuvas durante a vida útil do mesmo, ou então de condensações que ocorram no seu interior, motivadas pelos gradientes de pressão e temperatura a que estão sujeitos.

A existência de diversos modelos teóricos de transferência de humidade em materiais, e a exploração e validação já feitos dos mesmos, particularmente através do Programa Trhumidade, elaborado por V. Peixoto de Freitas [1], foi um passo importante neste sentido. O desenvolvimento de uma nova aplicação de simulação (TrHum 98) pretende dar mais uma contribuição para possibilitar um estudo sistemático e consequente dos fenómenos associados a ciclos de humedecimento e secagem de diversos materiais de construção.

1.3 Motivação

Para quem lida com a construção no seu dia a dia, quer a nível de concepção dos espaços e sistemas construtivos, quer ao nível da própria construção, no seguimento das obras, ou mesmo da recuperação de construções antigas, os problemas da qualidade, da durabilidade, da manutenção, do aspecto, quer dos edifícios, quer de elementos construtivos está sobremaneira presente.

Um dos factores mais determinante no comportamento dos elementos construtivos, e das edificações, é o conjunto de fenómenos associados à humidade. Um estudo mais aprofundado e sistemático dos mesmos poderá, e deverá, traduzir-se num aumento da qualidade efectiva das construções.

Deve-se reconhecer que a investigação que é feita em diversos campos, muitas vezes não tem tradução directa na construção que se faz, ou quando a tem, ela processa-se com um desfaseamento considerável. Isto sucede por muitas razões, que não vale a pena apontar, mas também pelo facto de não se conseguir fazer a passagem eficiente dos conhecimentos e das ferramentas entre os dois meios, o da investigação e o da concretização.

O presente trabalho pretende fazer essa ponte, na proporção que lhe compete. Isto é, disponibilizar uma ferramenta de investigação e análise a técnicos ligados à construção, sob um meio actual e a forma de um instrumento de fácil utilização.

Poder simular, ainda que parcialmente, o comportamento em embebição e secagem ao longo do tempo, de materiais de construção poderá permitir uma escolha de soluções de forma mais consciente e informada, com ganhos evidentes, na qualidade do conjunto e na sua durabilidade, e implicitamente no seu custo global.

Poder simular os fenómenos de secagem e embebição dos materiais de construção, em função de determinado clima, permite-nos o estabelecimento de prazos mais correctos para o faseamento sequencial das diversas actividades de uma obra.

Com base em modelos teóricos de comportamento de materiais porosos sujeitos a gradientes de temperatura e humidade, já testados no confronto com a análise de situações reais, e válidos dentro de determinadas fronteiras, é possível especificar e desenvolver algoritmos que dão origem a programas de computador, que simulam esses comportamentos ao longo do tempo, e em determinadas condições.

O interesse por este assunto tem vindo a crescer desde a década de 50, em vários países do mundo, tendo-se progressivamente realizado cada vez mais estudos sobre o mesmo. Como

ferramenta importante, e quase indispensável, nesses estudos surgem os programas de computador de simulação, mais ou menos potentes, mais ou menos restritos. Nesta altura existem já várias aplicações distintas para simulação do comportamento de materiais de construção, com características e aplicações diversas. Contudo a maior parte deles não está disponível numa base comercial, e os que estão, tem custos bastantes elevados.

1.4 Metodologia, Organização e Estrutura do Texto

Este trabalho, uma vez que se baseia num modelo teórico, já definido e validado, orientou-se fundamentalmente para a definição do programa, principalmente no que concerne ao estabelecimento do interface com o utilizador.

Os programas anteriormente realizados foram desenvolvidos para a plataforma *MS-DOS*, com tudo o que isso implica a nível de funcionamento e imagem . O presente programa foi desenvolvido em *Visual Basic 5.0* para plataforma *Windows 95*, permitindo uma melhor relação do utilizador comum com o mesmo, de forma mais intuitiva, e com sistemas de ajuda interactiva. Este programa irá auxiliar uma sistematização do estudo do comportamento dos materiais, embora com consciência que existe um longo caminho a percorrer, quer na ampliação das bases de dados de materiais, ainda muito restritas, quer na ampliação das opções disponíveis dentro do próprio programa.

O presente trabalho estrutura-se da seguinte forma; uma primeira parte de enquadramento na modelização da transferência de calor e humidade em materiais de construção; uma segunda referente à concepção geral do programa, uma terceira parte descritiva da funcionalidade do mesmo e uma quarta parte de teste, análise e interpretação dos resultados. Algumas conclusões e apontamentos sobre perspectivas futuras podem ser encontradas na última parte. O Anexo B contém listagens parciais do programa, das rotinas mais importantes para futuros estudos. Uma listagem total do programa não foi julgada conveniente pela grande quantidade de linhas de código do mesmo.

O capítulo 2 apresenta a perspectiva global de diversos modelos existentes de transferência de humidade e calor em materiais, das leis pelas quais se regem e dos programas para simulação, suas características e restrições. É particularizada a análise do modelo de Luikov / Philip - De Vries, uma vez que foi o modelo seguido no presente estudo. Seguidamente é

apresentado o programa *Trhumidade*, a evolução formal e funcional do mesmo, e o presente programa *TrHum 98*.

No capítulo 3 esclarecem-se aspectos referentes à concepção e estrutura do programa *TrHum 98*, ao nível dos módulos e funções de cálculo e de interface gráfica.

A descrição mais exaustiva do funcionamento do programa encontra-se descrita no capítulo 4, com a definição detalhada dos elementos necessários, com indicações das várias opções disponíveis, do modo de funcionamento, bem como da forma de apresentação dos resultados.

No capítulo 5 são apresentados os elementos relativos à validação do presente programa, por comparação com resultados de simulações já validadas no confronto com o real. São estabelecidas as várias condições iniciais, e são apresentados os resultados e as bases de comparação. A apresentação gráfica dos resultados é sempre valorizada para uma rápida apreensão da totalidade da situação. È apresentado um estudo baseado em análises da sensibilidade de alguns parâmetros envolvidos no cálculo, quer ao nível do rigor das soluções, quer do desempenho do programa.

O programa *TrHum 98* pretende ser uma ferramenta de análise de uso generalizado, que possa contribuir de forma clara para a passagem de uma análise qualitativa do comportamento dos materiais face à humidade para uma análise quantitativa.

2. MODELIZAÇÃO DA TRANSFERÊNCIA DE CALOR E HUMIDADE EM MATERIAIS DE CONSTRUÇÃO8

2.1	MODELOS TEÓRICOS DE TRANSFERÊNCIA	8
2.1.1	<i>Modelo de Glaser</i>	10
2.1.2	<i>Modelo de Krischer-Vos</i>	10
2.1.3	<i>Modelo de Luikov e Philip /De Vries</i>	11
2.2	DESCRIÇÃO MATEMÁTICA DO MODELO DE LUIKOV E PHILIP /DE VRIES	11
2.3	PROGRAMAS DE CÁLCULO EXISTENTES	14
2.4	EVOLUÇÃO DO PROGRAMA TRHUMIDADE	17
2.4.1	<i>Trhumidade</i>	17
2.4.2	<i>TrHum 1.0 / TrHum 96</i>	19
2.4.3	<i>TrHum 98</i>	21

2 Modelização da Transferência de Calor e Humidade em Materiais de Construção

2.1 Modelos Teóricos de Transferência

Os materiais utilizados correntemente na construção, são, quase sem excepção, materiais com uma estrutura porosa; isto é, materiais que na sua estrutura, possuem vazios, espaços (poros) que embora pertencendo à forma do material, não se encontram preenchidos pelo mesmo. Os meios porosos são extremamente complexos porque podem envolver elementos em três fases, sólida, líquida e gasosa.[2]

A fixação e o transporte de humidades em materiais porosos, resulta essencialmente de três mecanismos físicos fundamentais; a adsorção, a condensação e a capilaridade [1]. Estes três fenómenos são responsáveis, na generalidade dos casos, pela justificação da variação do teor de humidade de um material poroso.

O comportamento higroscópico dos materiais caracteriza-se pela variação do seu teor de humidade em função da humidade relativa do ambiente em que estão colocados. A generalidade dos materiais utilizados correntemente em construção civil são higroscópicos. A fixação da humidade processa-se em três fases, primeiramente ocorre a fixação de uma única camada de moléculas de água no interior dos poros, *adsorção monomolecular*, seguidamente, assiste-se à deposição de camadas sucessivas de moléculas, *adsorção plurimolecular*. Quando a quantidade de moléculas depositadas é suficientemente grande ou o diâmetro do poro relativamente pequeno, as diversas camadas de moléculas que envolvem as paredes do poro juntam-se, estando nessa altura configurado o fenómeno de condensação capilar.

A condensação ocorre sempre que a pressão instalada seja igual à pressão de saturação. Uma vez que, quer a pressão de saturação, quer a pressão instalada, são influenciadas pela temperatura, uma das principais razões para a ocorrência de condensações é a diminuição da temperatura e conseqüentemente a diminuição da pressão de saturação. No momento que a pressão de vapor instalada é igualada pela pressão de saturação (humidade relativa igual a 100%) verifica-se a condensação do vapor.

As condensações ocorrem quando existe um forte gradiente de temperaturas, e não são tomados os cuidados para impedir o estabelecimento de condições próximas da pressão de

saturação, ou quando, apesar do gradiente de temperatura não ser importante, estamos perante humidades relativas elevadas, como se pode verificar pelo desenho e espaçamento das curvas do diagrama psicrométrico da Fig. 1.

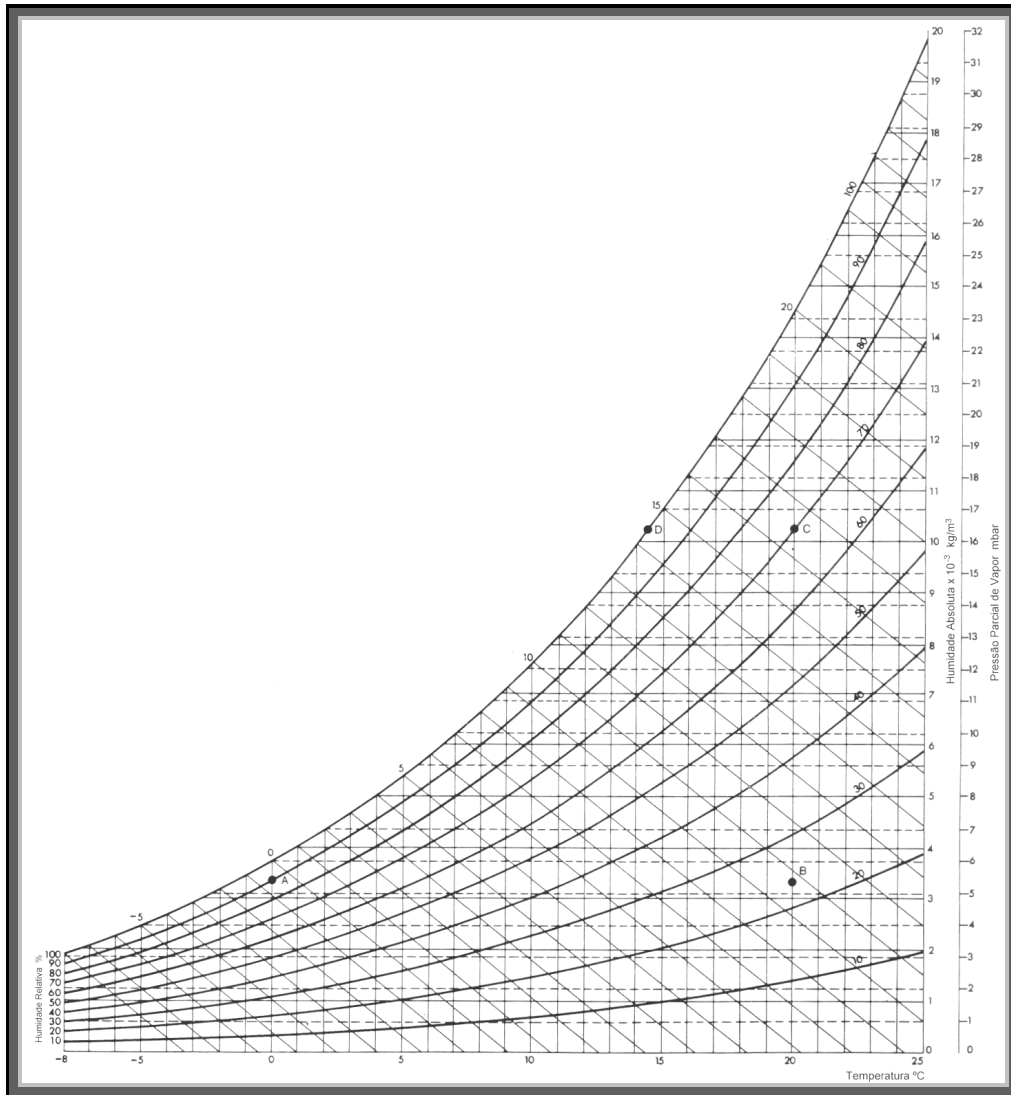


Fig. 1 - Diagrama psicrométrico.

Se as condensações ocorrem no interior de um material são designadas por condensações internas, se ocorrem na superfície do mesmo são apelidadas de condensações superficiais.

A capilaridade resulta das forças de atracção que se desenvolvem entre um corpo no estado sólido e um líquido, e podem anular ou inverter outras forças, como por exemplo as da gravidade. A capilaridade é responsável pela humedificação e embebição de sólidos que se encontram em contacto directo com um líquido, funcionando a estrutura porosa do mesmo

como um sistema de vasos capilares. A pressão capilar é função da temperatura, da humidade e da dimensão da estrutura porosa do material.

A pressão capilar varia na função inversa do teor de humidade, anulando-se para um teor de humidade igual ao teor de humidade máximo [1] .

Para além dos fenómenos atrás descritos, os trabalhos de construção têm uma importante contribuição no humedecimento dos materiais. Os materiais de construção ficam muitas vezes, no final da construção, com elevados teores de humidade, resultado dos métodos de confecção que requerem a presença da água, e da própria precipitação a que ficam invariavelmente expostos.

O processo de secagem dos materiais de construção porosos desenvolve-se em três fases [3] distintas. A primeira caracteriza-se pela evaporação da água superficial dos materiais a um ritmo constante, enquanto a humidade relativa na superfície do material é de 100%. A segunda fase caracteriza-se pela evaporação a um ritmo decrescente. Finalmente, a terceira fase é um processo extremamente lento, que pode decorrer ao longo de vários meses ou anos, até se atingir o equilíbrio higroscópico. Muitas vezes são aplicados acabamentos em elementos construtivos, quando os materiais se encontram aparentemente secos, mas com elevados teores de humidade em profundidade o que pode dar lugar a patologias graves.

A necessidade de caracterizar as situações e os mecanismos de transferência de humidade levou ao aparecimento de modelos matemáticos que pretendem definir e reproduzir esses fenómenos.

2.1.1 Modelo de Glaser

O modelo matemático de simulação da termodifusão de Glaser é baseado na lei de Fick, sendo unicamente válido para condições estacionárias e permanentes.

Permite, pela sua simplicidade, um rápido e fácil cálculo dos riscos de condensações no interior dos elementos de construção, através de resolução gráfica do problema, contudo não contempla qualquer movimento posterior da fase líquida resultante das condensações. Tem servido como instrumento para previsão do comportamento à humidade de paredes e como base para definição de regras de qualidade de elementos construtivos.

2.1.2 Modelo de Krischer-Vos

Este modelo continua a ser um método bastante simplificado dos mecanismos de transferência de humidade em materiais porosos.

A evolução relativamente ao método de Glaser surge no facto de considerar, para além do vapor de água, a água em fase líquida, e a sua distribuição, e o efeito dos gradientes de temperatura na transferência de humidade. [11] [12] .

2.1.3 Modelo de Luikov e Philip /De Vries

Os modelos matemáticos desenvolvidos, independentemente e quase simultaneamente por Luikov (1954) [13] e Philip/ De Vries (1957) [14] [15] [16] são semelhantes. Estes modelos permitem o cálculo do transporte de humidade simultaneamente na fase líquida e de vapor, bem como a transferência de calor [1] .

2.2 Descrição Matemática do modelo de Luikov e Philip /De Vries

O modelo proposto por Luikov e Philip /De Vries baseia-se nas Leis de difusão da massa de Fick e Darcy e na Lei da difusão de calor de Fourier, e é o que será utilizado neste trabalho, com as hipóteses simplificativas já estabelecidas por V. Peixoto de Freitas [1] .

As equações de conservação de calor e de massa para meios macroscopicamente homogéneos são as seguintes:

Conservação da massa:

$$\frac{d}{dt}(\rho_i \theta_i) = -\nabla(\dot{m}_{ic} + \dot{m}_{id}) + M_i \quad (2-1)$$

Conservação da energia

$$\frac{d}{dt}(\rho_i h_i + \rho_w h_w \theta_i) = -\nabla(\dot{q} + h_w (\dot{m}_{ic} + \dot{m}_{id})) \quad (2-2)$$

em que:

ρ_i, ρ_w	[kg/m ³]	Densidade do material e da água
$\dot{m}_{ic}, \dot{m}_{id}$	[kg/m ² s]	Densidade de fluxo de massa transportada por convecção e difusão
\dot{q}	[J/m ² s]	Densidade do fluxo de calor transportado por condução
M_i	[kg/m ³ s]	Termo traduzindo a condensação – evaporação
h_i, h_w	[Jm ³ /kg]	Entalpias mássicas da água e do material
θ	[m ³ /m ³]	Teor de humidade

Assumindo-se que existe conservação da massa global do sistema, o somatório dos termos que traduzem a condensação – evaporação (M_i) deverá ser igual a zero. É também admitido que $\dot{m}_{ic}^p = 0$, considerando que os materiais de construção não são permeáveis ao ar. [1]

O movimento da humidade pode ser dividido em movimento na fase líquida e movimento na fase vapor.

Movimento na fase líquida

$$\dot{m}_{ld}^p = -\rho_l(D\theta_l \cdot \nabla \theta_l + D_{tl} \cdot \nabla T) + \rho_l K \cdot \vec{K} \quad (2-3)$$

Movimento na Fase Vapor

$$\dot{m}_{vd}^p = -\rho_v(D\theta_v \cdot \nabla \theta_l + D_{tv} \cdot \nabla T) \quad (2-4)$$

O potencial de gravidade $\rho_l K \cdot \vec{K}$ poderá ser desprezado, uma vez que iremos estudar transferência de humidade em paredes verticais submetidas a um fluxo unidireccional horizontal. Foi demonstrado pelo estudo experimental elaborado por V. Peixoto de Freitas [1] que o efeito da gravidade é desprezável.

Partindo ainda dos seguintes pressupostos simplificativos:

- o meio poroso se encontra submetido a pequenos gradientes de temperatura;
- a capacidade calorífica varia pouco com a temperatura;
- a água é incompressível;

podemos escrever as equações generalizadas do modelo de Philip/De Vries.

Massa

$$(1 + \alpha) \frac{\partial \theta_l}{\partial t} + x \frac{\partial T}{\partial t} = \nabla(D\theta \cdot \nabla \theta_l + D_{tl} \cdot \nabla T) \quad (2-5)$$

Energia

$$\begin{aligned} (\rho^* C^*) \frac{\partial T}{\partial t} = & \nabla(\lambda^* \nabla T) + \rho_l C_l (D\theta_l \cdot \nabla \theta_l + D_{tl} \cdot \nabla T) \cdot \nabla T + \\ & + \rho_l \rho_v (D\theta_v \cdot \nabla \theta_l + D_{tv} \cdot \nabla T) \cdot \nabla T + \\ & + \rho_l L (\nabla(D\theta_v \cdot \nabla \theta_l + D_{tv} \cdot \nabla T)) - \\ & - \alpha \frac{\partial \theta_l}{\partial t} - x \frac{\partial T}{\partial t} \end{aligned} \quad (2-6)$$

A versão simplificada das equações anteriores é obtida, admitindo que o transporte de energia devido ao fluxo de massa é negligenciável, assim como a variação temporal do teor de vapor de água condensada.

Massa

$$\frac{\partial \theta_l}{\partial t} = \nabla(D\theta \cdot \nabla \theta_l + Dt \cdot \nabla T) \quad (2-7)$$

Energia

$$(\rho * C *) \frac{\partial T}{\partial t} = \nabla(\lambda * \nabla T) + \rho_l L \nabla(D\theta_v \cdot \nabla \theta_l + DT_v \cdot \nabla T) \quad (2-8)$$

Em resumo foram considerados os seguintes aspectos, como pressupostos simplificativos do modelo:

- os materiais porosos encontram-se submetidos a fracos gradientes de temperatura;
- os materiais de construção não são permeáveis ao ar;
- a influência da parcela referente ao transporte de calor por fenómenos de evaporação - condensação é pouco significativa face ao transporte por difusão, pelo que pode ser desprezada;
- os materiais usados em construção civil são homogéneos e macroscopicamente isotrópicos;
- os materiais deverão ter estrutura indeformável;
- não existe qualquer reacção química entre a água e os componentes do material de construção;
- considera-se negligenciável o potencial da gravidade;
- o modelo não contempla nem o congelamento da água, nem a ebulição, nem o degelo, pelo que devemos compreender as temperaturas dos materiais entre os 4°C e os 99°C;
- o teor de humidade da fase líquida é muito mais elevado do que o da fase vapor.

O sistema de equações diferenciais que serviu de base ao programa de cálculo automático *TrHum 98*, com todas as simplificações atrás referidas que descrevem o comportamento das

paredes de edifícios à transferência conjunta de calor e humidade, em regime variável e para um fluxo unidireccional, são as seguintes:

Massa

$$\frac{\partial w}{\partial t} = \frac{\partial}{\partial x} \left(D_w \frac{\partial w}{\partial x} + D_t \frac{\partial T}{\partial x} \right) \quad (2-9)$$

Energia

$$(\rho^* C^*) \frac{\partial w}{\partial t} = \frac{\partial}{\partial x} \left(\lambda^* \frac{\partial T}{\partial x} \right) \quad (2-10)$$

Em que:

ρ^*	[kg/m ³]	Massa volúmica equivalente
C^*	[J/kg.K]	Capacidade calorífica equivalente
λ^*	[W/mK]	Condutibilidade térmica aparente
T	[°C]	Temperatura
D_w	[m ² /s]	Coefficiente de difusividade hídrica
D_t	[m ² /sK]	Coefficiente de difusividade higrótérmica
θ	[m ³ /m ³]	Teor de humidade

A teoria de Philip/De Vries tem sido utilizada por diversos investigadores para previsão ou simulação do comportamento de elementos porosos quando sujeitos à transferência conjunta de humidade e calor, tendo, na generalidade, existido uma boa concordância dos valores experimentais com os valores obtidos pelo modelo teórico.

Estas verificações do modelo de Philip/De Vries permitem afirmar que, apesar de poderem permanecer algumas dúvidas quanto à definição de alguns parâmetros experimentais, o mesmo já foi suficientemente validado e poderá servir de base a estudos de simulação numérica.

2.3 Programas de Cálculo Existentes

A "International Energy Agency" [7] foi criada em 1975 com o propósito de aumentar a segurança energética através da conservação de energia e do desenvolvimento de fontes de energia alternativas. Uma das áreas de intervenção desta comissão é a conservação de energia em edifícios.

Na sequência dos trabalhos da comissão, foi elaborado um inquérito aos modelos de simulação e monitorização de edifícios existentes, no respeitante aos fenómenos de transferências de calor, ar e humidade, em elementos de construção. Este inquérito foi realizado nos diversos países que faziam parte da comissão: Bélgica, Canadá, Dinamarca, Alemanha, Finlândia, Grécia, Itália, Japão, Holanda, Nova Zelândia, Noruega, Suécia, Suíça, Turquia, Reino Unido e Estados Unidos, definindo uma panorâmica do estado de desenvolvimento desta área científica, naquela altura, e do interesse suscitado por este ramo de investigação nos diversos países. É de salientar que Portugal não fez parte dessa Comissão, razão pela qual o Programa *Trhumidade*, realizado por V. Peixoto de Freitas [1], em 1992, não aparece mencionado.

Os trabalhos da comissão ("Annex 24") tinham os seguintes objectivos estratégicos:

- Desenvolvimento do modelo e algoritmo - implementação, melhoria na modelização bem como teste de modelos simplificados com potencialidade de utilização na simulação;
- Caracterização das condições ambientais interiores e exteriores – escolha dos parâmetros ambientais, metodologia de análise e estabelecimento de bases de dados de condicionantes climáticas;
- Determinação das propriedades dos materiais – determinação e recolha das propriedades dos materiais com relevância para o estudo;
- Verificação experimental – ensaios de transferência de humidade, calor e ar, e comparação com valores obtidos pelo modelo;
- Prática e desempenho – transposição do conhecimento adquirido para uma correcta concepção e execução de elementos construtivos.

Foi feito um inquérito, junto dos países participantes, entre 1991 e 1993, que originou uma listagem comparativa dos modelos matemáticos de simulação da transferência de humidade, ar e calor e correspondentes programas.

No Quadro 1 apresenta-se um resumo do inquérito sobre os modelos de transferência de calor, ar e humidade, elaborado pela comissão.

Os programas listados são encarados fundamentalmente como instrumentos de investigação para a equipa que os elaborou e que domina o seu uso. Não existe, na maioria deles uma aposta na divulgação dos mesmos com vista à generalização do seu uso. Dos 29 programas a que o "Annex 24" faz referência apenas 6 estão disponíveis comercialmente.

Quadro 1 - Resumo do Inquérito -Annex 24 - Janeiro de 1993 [7]

Nome do Modelo	Tipo de transferência	Tipo de Modelo	País	Instituto
WALLDRY	1D - Calor + Ar + Humidade	Tipo 7	Canadá	CMHC
WALLFEM	1D - Calor + Ar + Humidade	Tipo 9	Canadá	CMHC
Wand	1D - Calor + Humidade	Tipo 1	Bélgica	KU - Leuven, LB
Glasta	1D - Calor + Humidade	Tipo 2	Bélgica	Physibel
HAMPI	1D - Calor + Humidade	Tipo 3	Canadá	Universidade de Saskatchewan
MATCH	1D - Calor + Humidade	Tipo 4	Dinamarca	TU - Dinamarca Therm. Insul. Lab
LTMB	1D - Calor + Humidade	Tipo 4	França	INSA, Dep Genie-Civil, Toulouse
V30	1D - Calor + Humidade	Tipo 4	França	CSTB
WFTK	1D - Calor + Humidade	Tipo 4	Alemanha	Fraunhofer Institut fur Bauphysik
JOKE	1D - Calor + Humidade	Tipo 4	Alemanha	TU - Cottbus Fachber. Physik + Werkstoffe
COND	1D - Calor + Humidade	Tipo 4	Alemanha	TU - Cottbus Fachber. Physik + Werkstoffe
HYGRO	1D - Calor + Humidade	Tipo 1	Holanda	TNO - Bouw, afdeling BBI
P1200A	1D - Calor + Humidade	Tipo 4	Suécia	SP
FUKT74:6	1D - Calor + Humidade	Tipo 4	Suécia	Gullfiber AB
BRECON2	1D - Calor + Humidade	Tipo 1	Reino Unido	BRE Scottish Laboratory
MOIST	1D - Calor + Humidade	Tipo 4	Estados Unidos	National Institute of Standards and Technology
NatKon	2D - Calor + Ar	Tipo 5	Bélgica	KU - Leuven, LB
AHCONP, ANHCONP	2D - Calor + Ar	Tipo 5	Suécia	Lund University
EMPTEDD	2D - Calor + Ar + Humidade	Tipo 6	Canadá	TROW _ CMHC
TRATMO2	2D - Calor + Ar + Humidade	Tipo 9	Finlândia	VTT - Lab. Of Heating and Ventilation
TCCC2D	2D - Calor + Ar + Humidade	Tipo 8	Finlândia	VTT - Lab. Of Heating and Ventilation
LATENITE	2D - Calor + Humidade	Tipo 4	Canadá	NRC , IRC
CHEoH	2D - Calor + Humidade	Tipo 4	França	Intitut de Mécanique des Fluides, Toulouse
TONY	2D - Calor + Humidade	Tipo 4	França	Intitut de Mécanique des Fluides, Toulouse
V320	2D - Calor + Humidade	Tipo 4	França	CSTB
WUFIZ	2D - Calor + Humidade	Tipo 4	Alemanha	Fraunhofer Institut fur Bauphysik
VADAU	2D - Calor + Humidade	Tipo 4	Suécia	Chalmers University of Technology
WISH - 3D	3D - Calor + Ar	Tipo 5	Holanda	TNO - Bouw, afdeling BBI
Konvek	3D - Calor + Ar + Humidade	Tipo 6	Bélgica	KU - Leuven, LB

Desde 1993 o panorama alterou-se ligeiramente, por um lado pela evolução natural dos programas existentes, com novas versões, e funcionando com interfaces mais intuitivas para o utilizador, e por outro, pela maior divulgação dos programas disponíveis, num grande veículo de informação que é a *Internet*. Assistiu-se também a um interesse crescente na

modelização das transferências de calor e humidade por parte de outras áreas científicas e tecnológicas.

De referir que o sector bio-alimentar, o sector agro-alimentar, a hidrologia, e as energias renováveis e alguns outros tem merecido a atenção dos investigadores desta área. Têm-se verificado adaptações dos programas e modelos existentes, a estas áreas, bem como desenvolvimento de modelos e programas específicos.

Tem-se vindo a verificar também uma maior divulgação dos programas, quer ao nível institucional, por colaboração entre centros de investigação e laboratórios, quer ao nível comercial pela disponibilização pública de alguns dos programas, em versões de uso restrito.

2.4 Evolução do Programa *Trhumidade*

2.4.1 *Trhumidade*

O programa *Trhumidade* foi elaborado por V. Peixoto de Freitas [1] em 1992, na sequência de trabalho realizado para a obtenção do grau de Doutor, utilizando a formulação proposta por Luikov e Philip /De Vries para simular a transferência de calor e humidade. O programa, tal como foi concebido nessa altura, aplicava-se a paredes de múltiplas camadas e estudava especialmente o fenómeno da interface entre diversos materiais e as diferentes situações de continuidade.

Foi desenvolvido em *Fortran 77* e assentava na resolução das equações de transferência de calor e humidade pelo método das diferenças finitas. A complexidade do cálculo, associada à linguagem utilizada, exigia computadores com capacidade elevada para a época, no caso um Eclipse MV 2000 – Data General. O *Trhumidade* foi organizado internamente, ao nível do código, no programa principal e em oito sub-rotinas.

O organigrama da Fig. 2 apresenta o algoritmo do programa e os métodos de cálculo utilizados, para obter aos valores do teor de humidade e das temperaturas instaladas ao longo do tempo.

O facto do programa *Trhumidade* necessitar, para ser executado, de um computador com requisitos especiais restringiu-lhe logo à partida a capacidade de ser utilizado fora do âmbito da investigação científica.

A forma de introdução de dados no programa revelava também alguma complexidade, pelo facto de serem necessários vários ficheiros diferentes para uma única simulação. Para os dados iniciais eram necessários três ficheiros, para a definição do clima um quarto e seis por

cada material diferente, isto para além dos ficheiros de escrita dos resultados. A inexistência de qualquer um desses ficheiros implicava o não funcionamento do programa.

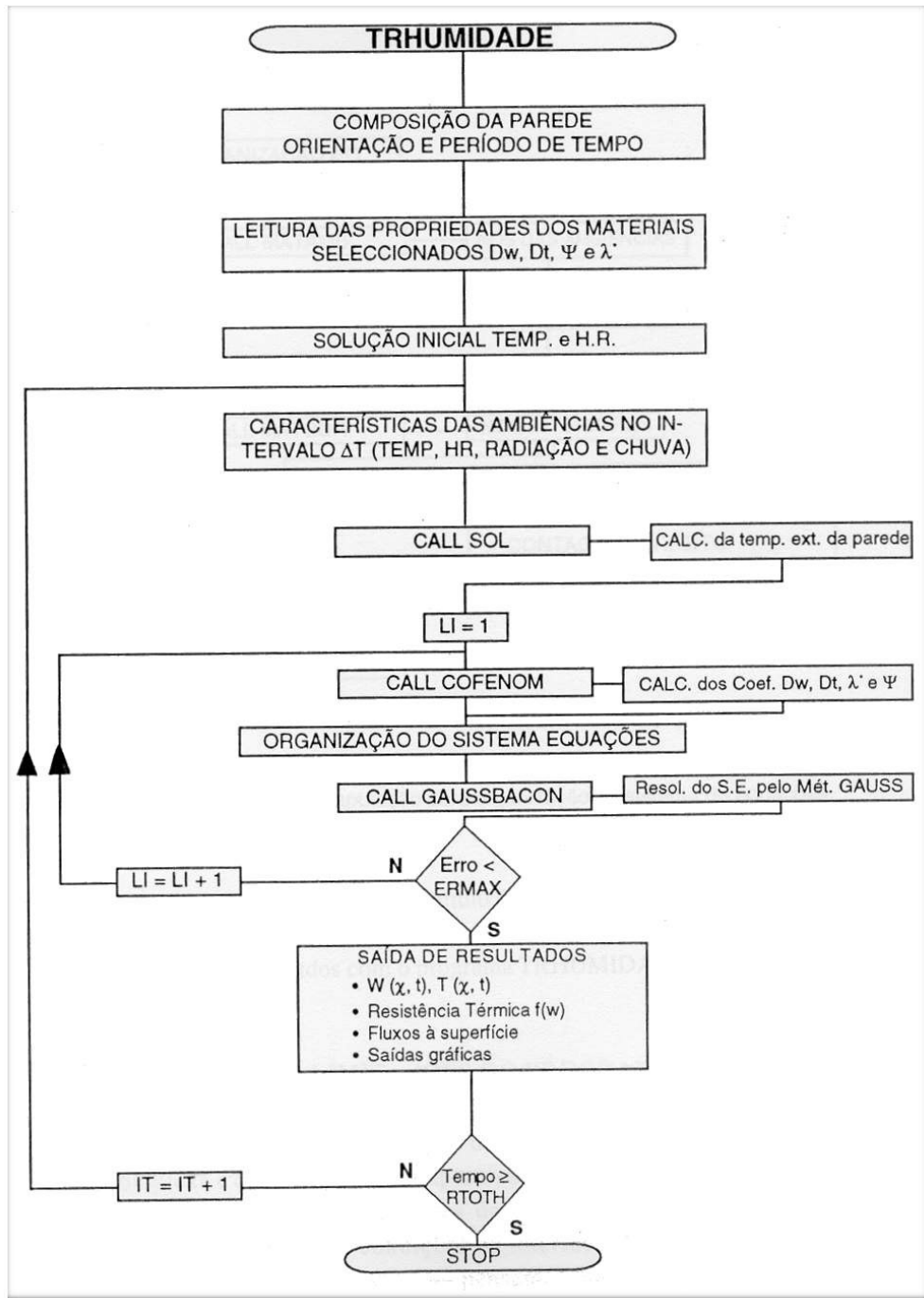


Fig. 2 - Organigrama do programa Trhumidade [1]

2.4.2 TrHum 1.0 / TrHum 96

Em 1995, A.Holm [5] fez a primeira adaptação para *PC* (computador pessoal), do programa *Trhumidade*. Procurou também fazer algumas alterações ao próprio âmbito do mesmo, a mais importante das quais foi a restrição da possibilidade de simulação a paredes compostas por uma única camada homogénea, isto é, sem necessidade de entrar em conta com os fenómenos de continuidade entre diversos materiais.

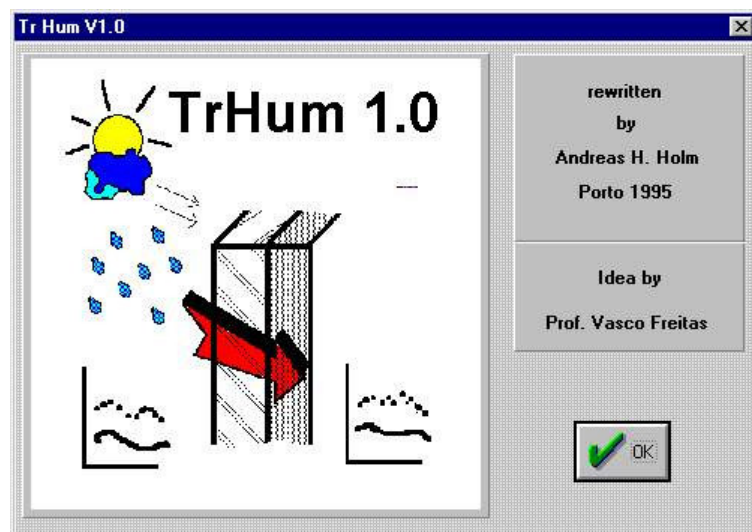


Fig. 3 - Janela de entrada do programa TrHum 1.0 [5]

O programa foi escrito, numa fase inicial em *Borland Pascal* para *Windows*, versão que nunca ficou totalmente operacional, pela morosidade que as linguagens de programação ditas visuais implicam [ver Fig. 3]

Posteriormente foi feita uma nova versão, em *Turbo Pascal*, para funcionar sobre sistema operativo *MS-DOS* ou afim. Esta versão, [ver Fig. 4] visualmente, é muito menos interessante e apelativa que a versão *Windows*, mas ficou quase totalmente operacional, com algumas restrições relativas à efectuação de simulações com sol e com humidades relativas superiores a 90%.

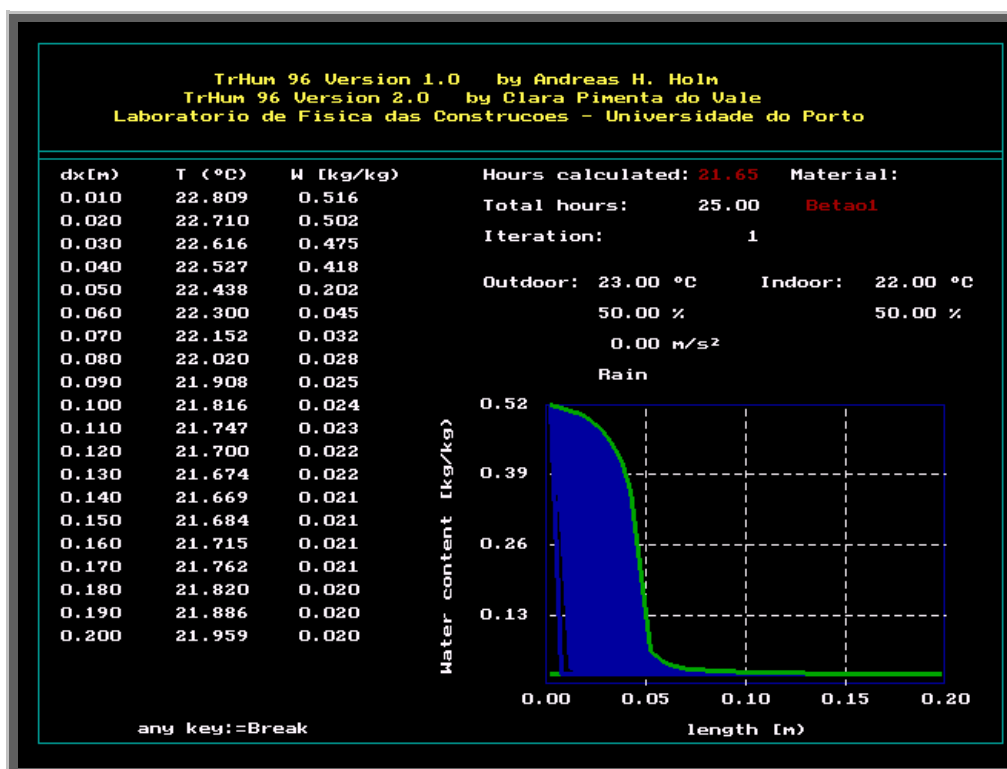


Fig. 4 – Apresentação de resultados do TrHum 96 2.0 [5]

A estrutura do programa manteve-se sensivelmente idêntica à do *Trhumidade*, com as reduções inerentes ao cálculo de uma única camada, e as simplificações necessárias ao aligeiramento do programa, para que o mesmo pudesse funcionar num computador de uso pessoal. Foi contudo mais privilegiada a relação com o utilizador comum, por uma reformulação do modo de entrada de dados.

A entrada de dados no *TrHum 96* continuava a ser feita a partir de ficheiros de texto com informação, contudo o seu número foi reduzido, e foram criadas linhas de informação ao utilizador. Nesta versão são necessários quatro ficheiros para efectuar uma simulação, um com os valores referentes à definição do clima, o segundo, uma base de dados com parâmetros de caracterização de materiais de construção, um terceiro com as definições relativas ao elemento a estudar, sua localização e características, e com definições relativas ao projecto em si, erro máximo admissível, números e hora dos perfis de temperatura e teores de humidade a guardar, e, por ultimo, um ficheiro para a escrita dos resultados.

2.5 TrHum 98

O objectivo do presente trabalho foi o de elaborar um novo programa de simulação numérica do comportamento dos materiais de construção em embebição e secagem, designado por *TrHum 98*. Desenvolvido em *Visual Basic 5.0*, tem como objectivo principal melhorar significativamente a relação com o utilizador, tornar o uso do *software* muito mais intuitivo e fácil, quer pela utilização do Ambiente *Windows*, já com uma larga penetração no mercado da informática, quer pela inclusão de ajuda automática, isto é pelo aparecimento de conselhos de utilização e outras informações, sobre determinado comando, quando o curso do rato de aproxima do mesmo [Fig. 5].

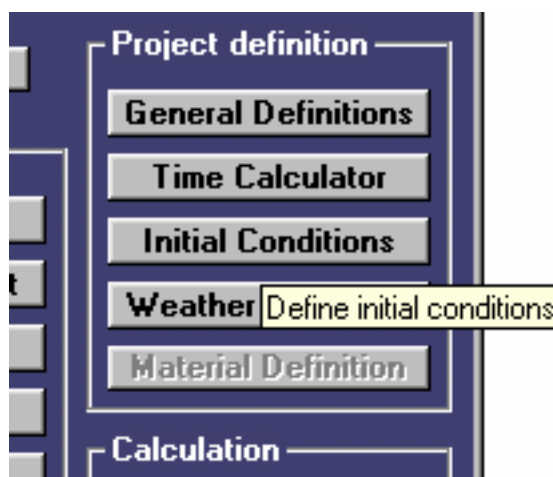


Fig. 5 – Ajuda Automática

O *Windows*, como sistema operativo, ou como ambiente de trabalho, estabeleceu um *standard* que permite que os programas que o utilizam, sejam facilmente usados por pessoas sem formação específica nos mesmos. O *TrHum 98*, ao utilizar o ambiente *Windows* irá beneficiar desta facilidade.

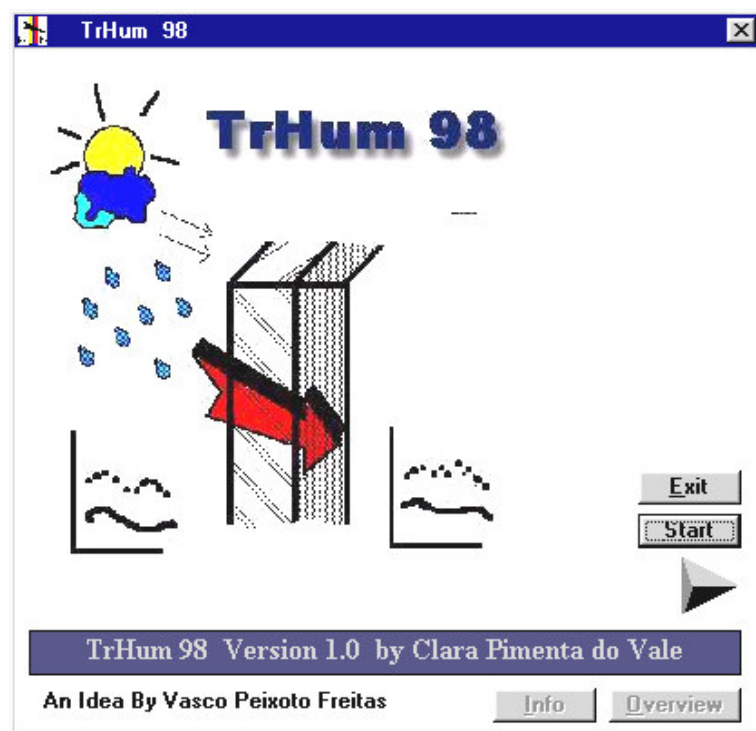


Fig. 6 - Janela de entrada do programa TrHum 98

A aposta principal foi efectivamente na melhoria do aspecto visual e na facilidade de utilização do programa.

3.	CONCEPÇÃO DO PROGRAMA	24
3.1	OBJECTIVOS	24
3.2	ESTRUTURA DO PROGRAMA	26
3.2.1	<i>Funções de entrada, edição e visualização de dados e resultados</i>	27
3.2.2	<i>Funções de manuseamento de ficheiros</i>	27
3.2.3	<i>Funções de Cálculo</i>	27
3.3	CONCEPÇÃO DA INTERFACE COM O UTILIZADOR	28
3.3.1	<i>Entrada de dados</i>	28
3.3.2	<i>Apresentação de resultados</i>	32
3.4	MODELO MATEMÁTICO	33
3.4.1	<i>Estrutura dos módulos de Cálculo</i>	35
3.4.2	<i>Resolução do sistema de equações</i>	35
3.4.3	<i>Cálculo do teor de humidade em função da humidade relativa</i>	37

3 Concepção do Programa

3.1 Objectivos

O desenvolvimento do programa *TrHum 98* dividiu-se em duas vertentes, que funcionaram paralelamente, mas nem sempre com o mesmo grau de importância; a análise e definição do que deveria ser a estrutura do programa na sua relação com o utilizador (sistemas de entradas e armazenamento de dados, sistemas de saídas e visualização de resultados); e a definição dos módulos de cálculo matemático, com base nos algoritmos já definidos nos anteriores programas. Os algoritmos foram optimizados tendo em conta o sistema operativo e a linguagem de programação utilizados, a par com a definição de uma estrutura que permita uma distribuição livre do programa, aumentando a base de dados dos materiais, sem ser necessário alterar o código do programa. Pretendeu-se também condensar o número de ficheiros necessários para uma simulação, para que o programa se tornasse mais operativo, mais manobrável.

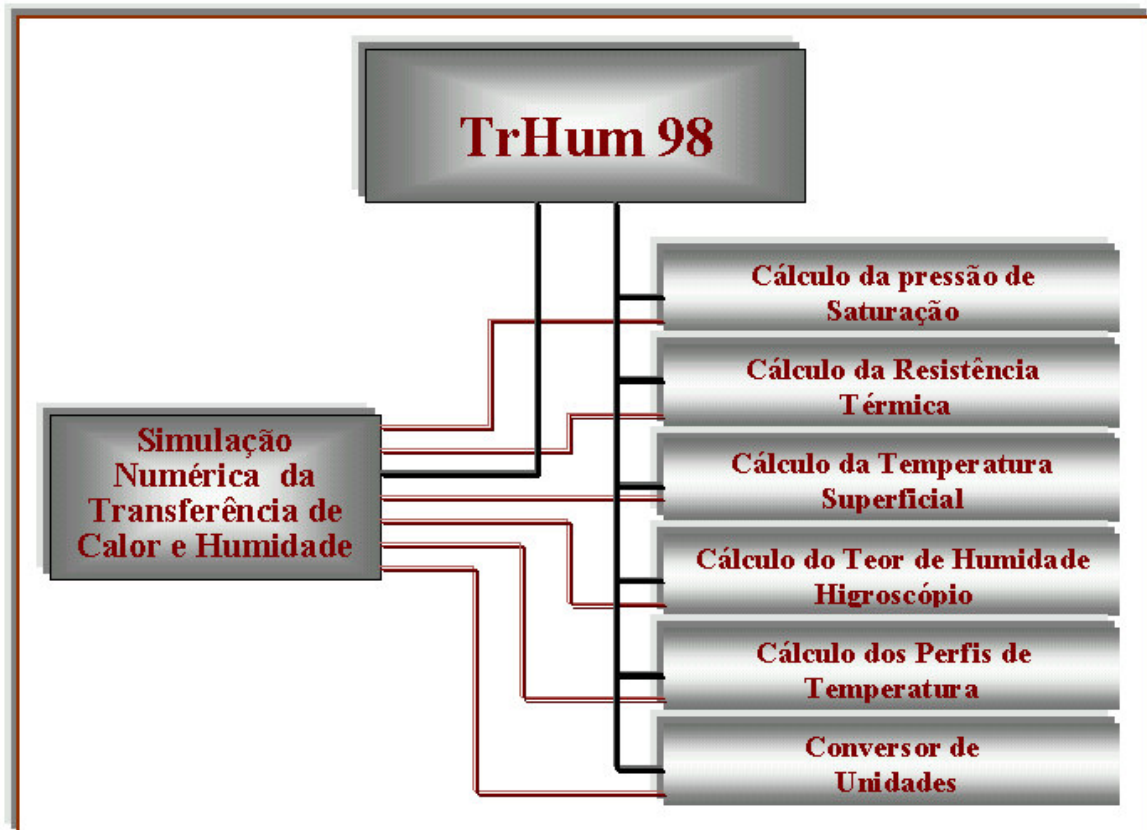
O programa foi inicialmente desenvolvido como um instrumento de simulação do comportamento de materiais e elementos de construção à transferência conjunta de calor e humidade, mas foi estruturado de uma forma modular, no sentido de integrar vários módulos, que no conjunto ajudam a consolidar a aplicabilidade do programa, em várias situações e a vários níveis.

Estes módulos tem diversas funções, que auxiliam uma determinação mais rápida, rigorosa e fácil de parâmetros com influência na caracterização dos comportamentos higrótérmicos dos materiais. Funcionam como sub-programas que podem ser utilizados de uma forma autónoma, ou então podem ser chamados dentro do programa principal, funcionando como auxiliares na definição de condições de simulação.

Foram definidos seis módulos adicionais ao módulo principal, ficando assim o programa dividido em sete componentes. Qualquer um dos módulos adicionais pode ser instalado isoladamente, como um pequeno utilitário, com a vantagem de poderem funcionar em computadores sem requisitos especiais. Tal como o programa principal, os módulos funcionam sobre sistema operativo *Windows*.

As funções dos novos módulos são as indicadas no diagrama de blocos seguinte, e resultam na sua maioria de funções do programa principal que foram autonomizadas. Serão descritos com detalhe mais adiante.

Quadro 2 – Diagrama de blocos base do TrHum 98



O acesso a qualquer destes módulos é feito de uma forma fácil, logo no arranque do programa através de uma janela de escolha múltipla.

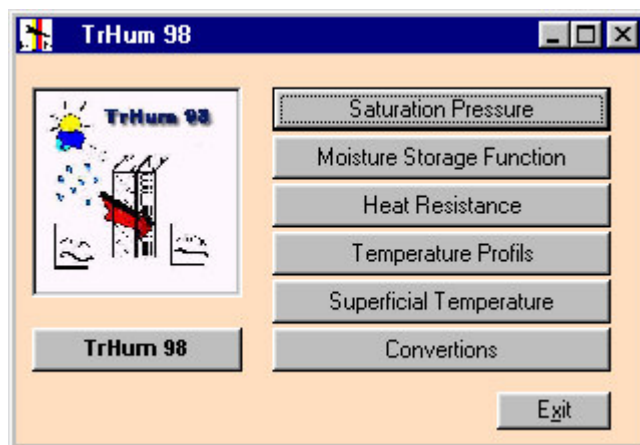


Fig. 7 - Janela de escolha múltipla

Durante a utilização do módulo principal, é possível aceder aos módulos adicionais através do sistema de menus.

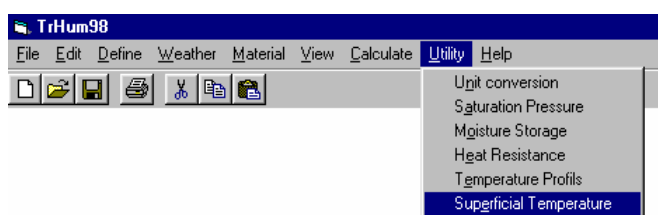


Fig. 8 – Sistema de menus

3.2 Estrutura do programa

O *Visual Basic* é uma linguagem de programação orientada aos objectos, em que as sequências de acontecimentos são desencadeadas por acções definidas. Cada janela do programa é constituída por um conjunto de objectos, com características determinadas, e sobre os quais é possível ter determinado tipo de acções, por exemplo, a colocação do cursor do rato sobre o objecto, ou o carregar no comando *enter*. O programa é constituído por um conjunto mais ou menos extenso de pequenas funções que se encadeiam em resultado das acções do utilizador sobre objectos do programa.

Numa aplicação escrita numa linguagem visual, as funções relacionadas com o interface gráfico tem um peso substancial na programação. Todas as facilidades concedidas ao utilizador no uso do programa correspondem sempre a um acréscimo substancial de código que é necessário executar.

Neste programa as funções relacionadas com o interface gráfico excedem largamente as funções específicas de cálculo, quer ao nível de quantidade de linhas de código, quer ao nível do tempo de execução das mesmas.

O programa tem fundamentalmente três tipos distintos de funções: funções relacionadas com a entrada, edição e visualização de dados e resultados; funções de manuseamento de ficheiros e funções de cálculo. Esta estrutura verifica-se tanto no módulo principal como nos módulos adicionais.

3.2.1 Funções de entrada, edição e visualização de informação

Este tipo de funções está relacionado directamente com a parte visível do programa, e com as acções directas do utilizador. São um conjunto extenso de pequenas funções, respostas directas a acções sobre os elementos gráficos do programa. Definem toda a estrutura de funcionamento do mesmo e o modo como as diversas acções se encadeiam. Ao contrário das funções de cálculo, não são específicas nem reféns do programa e ajudam a consolidar a utilização intuitiva que se pretende.

3.2.2 Funções de manuseamento de ficheiros

Outro nível de funcionalidade do programa tem a ver com o manuseamento de dados e dos ficheiros para o seu armazenamento. Constituem um grupo relativamente grande, quer pelo tipo e quantidade de dados necessários à simulação que determinam ficheiros extensos; quer pela funcionalidade do programa, que ao apostar na compatibilidade com os programas anteriores, torna necessário a existência de várias funções para fazer a importação dos mesmos.

O programa guarda a totalidade dos dados necessários à simulação num ficheiro de projecto, ficheiro de texto com extensão “.hum”. A utilização de um ficheiro de texto como método de armazenagem do projecto tem a vantagem de permitir a leitura e edição dos mesmos em programas facilmente disponíveis no mercado. Permite ainda manter a compatibilidade vertical com futuras versões do *TrHum* 98.

3.2.3 Funções de Cálculo

O terceiro nível na estrutura funcional do programa é o cálculo. O que noutras linguagens seria o programa em si, nas linguagens visuais, funciona apenas como mais um conjunto de

instruções, e que em muitos casos é difícil de separar dos conjuntos atrás descritos, por causa das interferências ou interdependências entre uns e outros.

3.3 Concepção da Interface com o Utilizador

Como já foi referido, foi dado grande importância ao aspecto formal do programa, ao modo como a informação é solicitada e apresentada, à forma como se faz o encadeamento das diversas acções, e às indicações que se dão ao utilizador para o uso do mesmo.

O programa foi desenvolvido em inglês, tanto ao nível do interface com o utilizador, como ao nível dos comentários às linhas de código, por duas razões: primeiro para promover a internacionalização do programa, quer para o seu teste e validação, quer para a sua divulgação junto dos profissionais da construção; em segundo lugar, porque quer os contactos do Laboratório de Física das Construções com organismos congéneres estrangeiros, quer os Mestrados Internacionais promovidos pela FEUP, poderão levar a uma continuação deste trabalho por um investigador não nacional, à semelhança do que já aconteceu no passado.

O inglês, como língua com maior divulgação internacional, é a escolha lógica para o desenvolvimento do programa, sem contudo por de parte a hipótese de uma versão em Português, bem como noutros idiomas.

3.3.1 Entrada de dados

A definição dos mecanismos disponíveis para entrada de dados, e a informação dada sobre o modo como esse fornecimento deve ser feito, são aspectos importantes para a determinação do nível de facilidade de utilização de um programa, e para o seu sucesso.

Para melhorar a funcionalidade e a simplicidade da interface gráfica do programa, optou-se por considerar múltiplas formas de inserir os dados, para que cada utilizador o possa fazer do modo mais adaptado às suas necessidades e ao seu hábito. Tomou-se como referência várias aplicações da família *windows*, pois pela divulgação e aceitação de que dispõem, constituem um *standard* ao nível da definição de interface gráfico de *software*.

O *TrHum 98* considera 4 tipos possíveis de entrada de dados. A entrada da informação poderá ser feita a partir de menus e sub-menus, poderá ser feita por atalhos pré-definidos, através da janela de tarefas [ver Fig. 10] e pelas janelas encadeadas [ver Fig. 9].

3.3.1.1 Janelas encadeadas

Este sistema de entrada de dados é destinado preferencialmente a utilizadores pouco familiarizados, quer com a informática, quer com o programa, pois a sequência de acções necessárias está pré-definida à partida, e é executada automaticamente pelo computador, após acordo do utilizador.

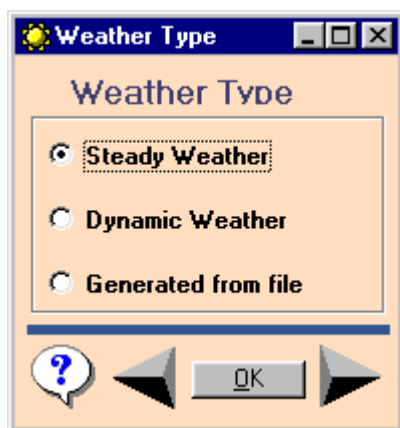


Fig. 9 – Exemplo de uma janela encadeada

Dois botões em forma de setas, com sentidos opostos [ver Fig. 9], permitem ao utilizador a passagem directa à janela seguinte ou à janela anterior, correndo a totalidade da sequência de janelas necessárias à completa definição dos dados necessários à simulação.

Este tipo de entrada de dados impõe, como condição, a passagem por todas as janelas antes de efectuar o cálculo, o que pode ser moroso, quando se pretende unicamente alterar uma ou outra condição de simulação. Será um tipo de entrada de dados que será tendencialmente abandonado por utilizadores mais experientes do programa.

3.3.1.2 Janela de tarefas

Os dados podem ser introduzidos no programa através da janela de tarefas [ver Fig. 10], que mostra todas as acções necessárias à definição das condições de simulação.

Esta entrada de dados destina-se a utilizadores com mais conhecimento do programa, pois requer que os mesmos saibam quais os passos que deverão executar para a entrada de todos os dados necessários.

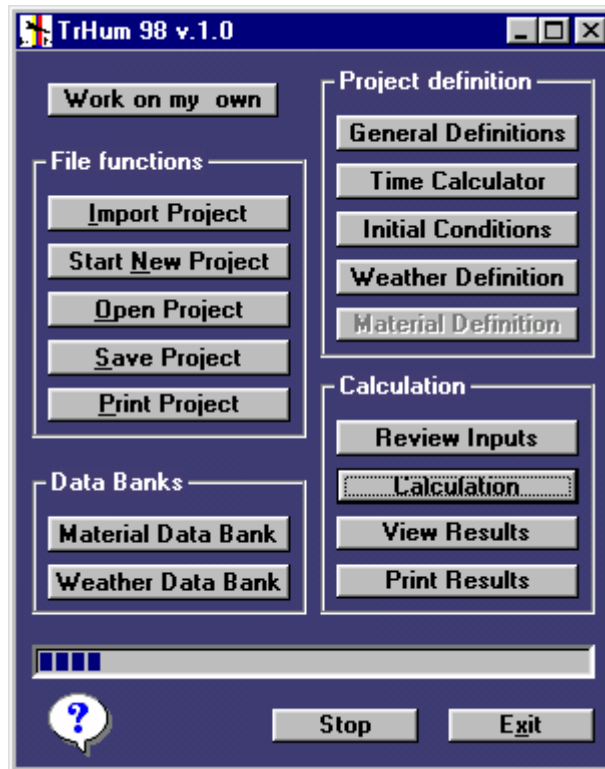


Fig. 10 - Janela de tarefas

Esta janela, contudo, tem os botões de comando organizados em grupos, pelo tipo de instruções que executam, sendo fácil a determinação de quais os comandos que deverão ser executados para a definição do projecto. Não implica o encadeamento de janelas, pelo que permite ao utilizador escolher quais os dados que pretende fornecer em cada momento, e ordem pela qual o deseja fazer. Tem ainda uma utilização preferencial na edição de projectos já definidos, pois permite a alteração de um só parâmetro sem ter que verificar todos os outros, e no cálculo, permite fazer a verificação de todos os parâmetros entrados. A partir da janelas de tarefas tem-se acesso a todas as funções normais de manipulação de ficheiros, tais como o gravar, o abrir, o iniciar e o imprimir.

3.3.1.3 Menus e sub-menus.

Todas as instruções existentes no programa estão disponíveis através de menus e sub-menus [ver Fig. 11]. O sistema de menus é algo com que os utilizadores de programas de computador estão suficientemente familiarizados. Se existir o cuidado de seguir uma lógica de agrupamento de comandos coerente, e mais próxima possível da dos programas de grande divulgação, pelo menos no tocante a funções genéricas, temos uma primeira garantia de facilidades de utilização. Um utilizador habituado a utilizar programas de computador,

mesmo inconscientemente, saberá onde encontrar, por exemplo, a opção de gravar, ou de imprimir. Tem como inconveniente o facto de as funções dos menus não estarem sempre visíveis, e ser necessário um esforço de memória e aprofundamento da aprendizagem para que determinado tipo de utilizadores passe a dominar o programa.

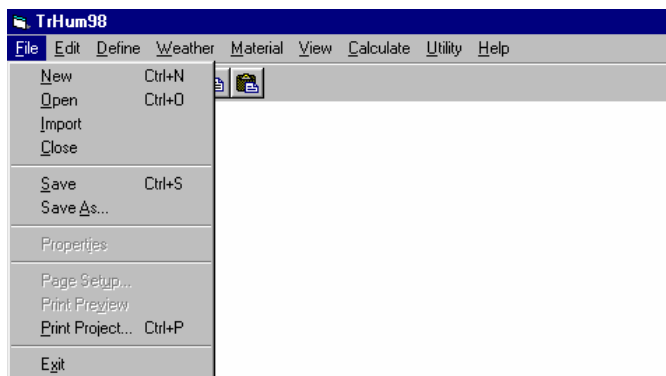


Fig. 11 – Sistema de menus e sub-menus e definição de atalhos

Pelos menus é possível aceder a funções que não estão disponíveis nem na janela de tarefas nem por janelas encadeadas. Estas funções, na sua maior parte, são pequenos utilitários criados para auxiliar o utilizador em pequenas tarefas, como consulta de dados ou conversão de unidades, funcionando também como módulos independentes, como foi já apresentado.

3.3.1.4 Sistema de atalhos

Muitos utilizadores de computadores, essencialmente “utilizadores pré *Windows*”, ainda estão habituados a usar muito mais o teclado do que o rato para a navegação dentro do programa. Para esse utilizadores, foi criado o sistema de atalhos, que lhes permite navegarem pelos menus e janelas, sem necessitarem de pegar no rato [ver Fig. 11]. O sistema de acesso é o habitual em programas “*Windows standard*”, a conjugação da tecla *alt* ou da tecla *ctrl* com uma letra ou sequência de letras, permite aceder às janelas do programa e executar os comandos. Poderá ser a forma mais rápida de fazer o preenchimento de todas as janelas, mas implica um esforço de memorização por parte do utilizador, para recordar a totalidade dos atalhos.

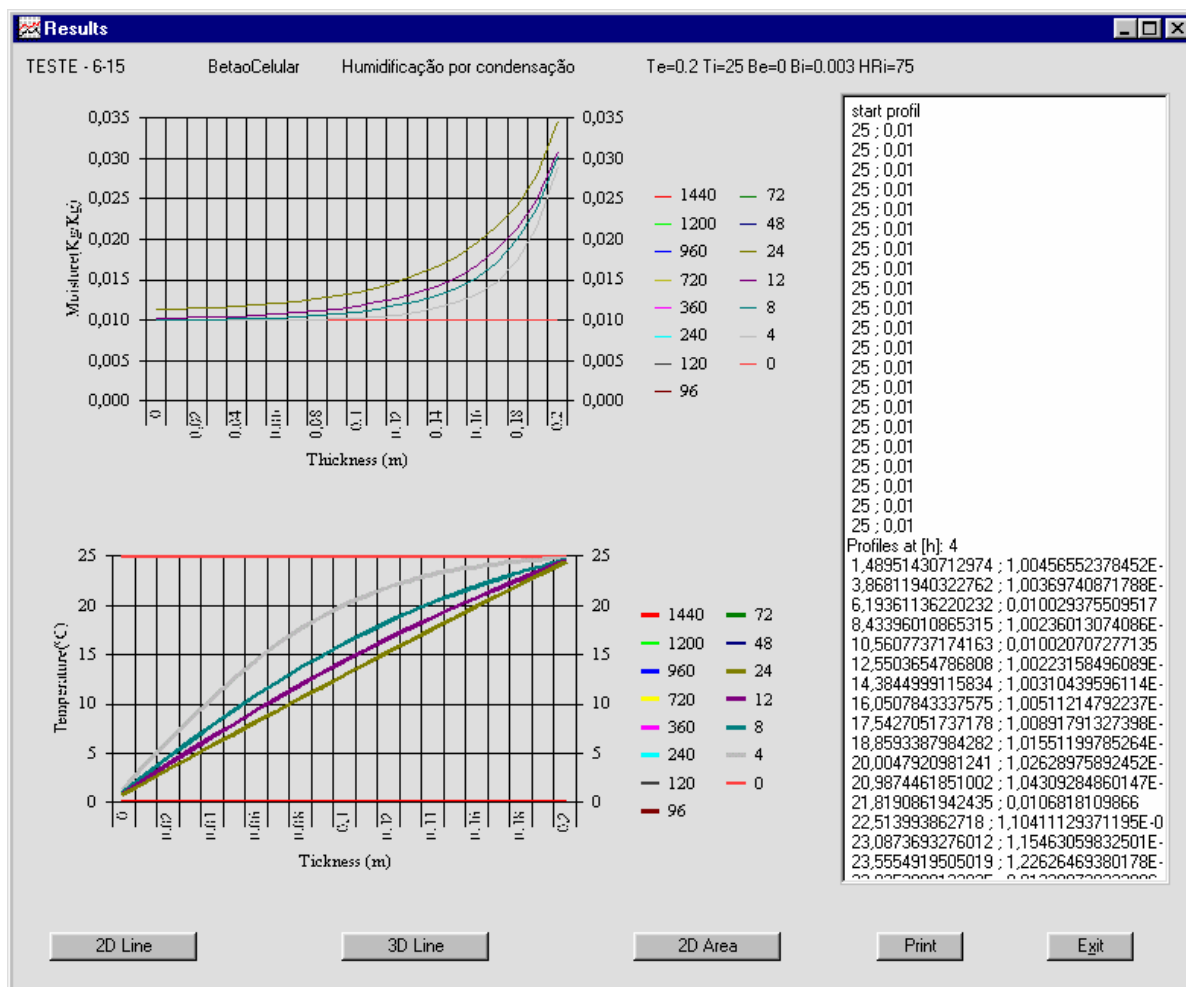


Fig. 12 – Vista do gráfico de saída de resultados

3.3.2 Apresentação de resultados

Outro aspecto importante na definição do programa é a forma como os resultados são apresentados ao utilizador. Os resultados de uma simulação poderão ser demasiado extensos para que possamos ter uma fácil percepção dos mesmos. É pois importante, por um lado, fazer a selecção dos valores que deverão ser transmitidos ao utilizador para que ele possa facilmente aperceber-se do seu significado e, segundo, tornar essa transferência o mais operativa possível com, por exemplo, recurso a gráficos que num relance nos dão a mesma informação que várias páginas de resultados.

O programa, após terminar o cálculo, permite o acesso à janela de visualização dos resultados, onde os mesmos são apresentados sob a forma gráfica e analítica [ver Fig. 12]. É possível a visualização dos perfis dos teores de humidade e dos perfis de temperatura, nos

momentos pré definidos, em gráficos bi-dimensionais. O utilizador poderá sempre escolher outro tipo de gráfico, dentro das opções disponíveis. Tem ainda a possibilidade de ver os valores dos resultados numa janela de texto.

São ainda guardados dois ficheiros de resultados, com os valores do teor de humidade e da temperatura, nos momentos definidos, que podem ser visualizados no interior do programa, ou por programas externos, tais como o *Wordpad* ou o *Excel*. Um dos ficheiros foi especificamente concebido para ser importado para o *Excel*, para possibilitar a análise dos resultados de uma forma mais sistemática, permitindo a comparação directa entre simulação, ou a elaboração de outro tipo de gráficos mais complexos, que não estejam disponíveis no programa.

3.4 Modelo Matemático

O programa *TrHum 98*, inspira-se no programa *Trhumidade [1]*, quanto à formulação matemática, embora com algumas diferenças porque os objectivos agora são diferentes. O *Trhumidade* foi elaborado como uma ferramenta de análise do fenómeno da transferência de humidade através das paredes do edifício e da influência da interface entre as diversas camadas. O *TrHum 98*, pretende estudar os mecanismos de embebição e secagem de materiais de construção, pelo que não foi considerado na formulação matemática a possibilidade de elementos com mais de uma camada. Consubstancia uma aposta na possibilidade da generalização do uso do mesmo a um público de formação não específica.

O programa baseia-se na resolução numérica das equações de transferência conjunta de humidade e calor em meios porosos constituídos por um único material, macroscopicamente homogéneo e isotrópico (de acordo com a formulação proposta por Luikov e Philip /De Vries).

As equações simplificadas (2-9) (2-10) descritas no Capítulo 2 são resolvidas pelo método das diferenças finitas.

A função teor de humidade $W(x, t)$ e a função Temperatura $T(x, t)$, que satisfazem as equações, deverão ser deriváveis uma vez em relação ao tempo, no intervalo $[1, +\infty]$ e duas vezes em relação ao espaço no intervalo $[0, e]$.

O método das diferenças finitas consiste em primeiro lugar na substituição, numa equação diferencial, das derivadas pelas derivadas aproximadas.

Assim:

$$\frac{\partial y}{\partial x} = \frac{1}{\Delta x} (y_{i+1} - y_i) \quad (3-1)$$

$$\frac{\partial^2 y}{\partial x^2} = \frac{1}{\Delta x^2} (y_{i-1} - 2y_i + y_{i+1}) \quad (3-2)$$

Utilizando o método implícito centrado de Crank-Nicholson, obtemos a linearização do sistema pela avaliação dos coeficientes no instante $(m+\theta) \Delta t$.

$$x^{(m+\theta)} = \theta \cdot x^{(m+1)} - (1-\theta) \cdot x^m \quad (3-3)$$

Com $\theta = 1/2$

Assim, a determinação dos coeficientes (D_w , D_T , λ^* , C^* , ρ^*) no instante $(m + 1/2) \Delta t$, faz-se do seguinte modo:

$$a(w, T)^{m+1/2} = 1/2 [a(w^{m+1}, T^{m+1}) + (w^m, T^m)] \quad (3-4)$$

Tomando A e B pelas expressões:

$$A = \left[D_w \frac{\partial w}{\partial x} + D_T \frac{\partial T}{\partial x} \right] \quad (3-5)$$

$$B = \left[\lambda^* \frac{\partial T}{\partial x} \right] \quad (3-6)$$

Pode-se escrever as equações de conservação de massa e energia da seguinte forma:

Massa

$$W_k^{m+1} - W_k^m = \frac{\theta}{\Delta x} [A_{k+1/2}^{m+1} - A_{k-1/2}^{m+1}] + \frac{1-\theta}{\Delta x} [A_{k+1/2}^m - A_{k-1/2}^m] \quad (3-7)$$

Energia

$$\rho^* C_k^{*m+1} \frac{T_k^{m+1} - T_k^m}{\Delta t} = \frac{\theta}{\Delta x} [B_{k+1/2}^{m+1} - B_{k-1/2}^{m+1}] + \frac{1-\theta}{\Delta x} [B_{k+1/2}^m - B_{k-1/2}^m] \quad (3-8)$$

Obtém-se assim, em cada nó interior duas equações de transferência e seis incógnitas; W_{k-1} , W_k , W_{k+1} , T_{k-1} , T_k , T_{k+1} , correspondentes aos valores da temperatura e do teor de humidade, no nó, no nó anterior e no nó seguinte.

Isto permite o estabelecimento de um sistema de $2*N$ equações com $2*N$, incógnitas, sendo N o número de nós da amostra em questão.

A resolução do sistema de equações é feita pelo método de Gauss.

3.4.1 Estrutura dos módulos de cálculo

A parte responsável pelo cálculo matemático necessário às simulações assenta num conjunto de funções que são accionadas pela parte gráfica do programa.

As funções mais importantes são as seguintes:

- Sol* - Cálculo da temperatura fictícia ar-sol
- Cofenon* - Cálculo dos coeficientes fenomenológicos para a temperatura e teor de humidade instalados
- Matamb* - Cálculo das equações dos nós em contacto com as ambiências
- Matcor* - Cálculo das equações nos nós interiores
- Psatura* - Cálculo da pressão de saturação
- Hurelat2* - Cálculo da Humidade relativa em função do teor de humidade
- Higroscop2* - Cálculo do teor de humidade em função da humidade relativa
- Gauss* - Resolução de um sistema linear de equações pelo método de condensação de matrizes de Gauss

3.4.2 Resolução do sistema de equações

A resolução do sistema de equações é um dos pontos fundamentais no programa porque determina o grau de rigor do mesmo. Foram feitos alguns estudos sobre diferentes formas de resolução, todos baseados no método de Gauss. Fizeram-se comparações entre resolução de grandes matrizes por diferentes métodos e chegou-se à conclusão que a diferença de resultados é sempre inferior a $1E-14$, o que é considerado aceitável para este tipo de cálculo.

A matriz que resulta da resolução das equações de transferência nos nós pelo método das diferenças finitas é uma matriz banda. O programa *Trhumidade* utilizava um algoritmo de resolução da matriz, pelo método de Gauss, desenvolvido pelo *Institut de Mécanique des*

Fluides de Toulouse, em que o cálculo só se efectuava dentro da banda, reduzindo o número de operações necessárias o que tem evidentes vantagens a nível do tempo despendido no cálculo, o que era importante naquele programa, fundamentalmente pelas limitações de velocidade do computador e do compilador *Fortran 77*.

Foi desenvolvido um módulo de resolução de matrizes pelo método de Gauss, baseado no algoritmo geral [9] [10] de resolução por condensação da matriz, para ser utilizado no programa *TrHum 98*.

```

Sub Gauss()

Dim m As Double
Dim m1 As Double
Dim m2 As Double
Dim nn As Integer
Dim k As Integer
Dim ii As Double
Dim j As Integer
Dim FoundPivot As Boolean
Dim iii As Double

nn = 2 * gTotalNodes

For ii = 1 To nn
  For j = 1 To nn
    AA(ii, j) = AA(ii, j) * 100000000
  Next
  b(ii) = BB(ii) * 100000000
Next

'pivots e swap verification
For ii = 1 To nn
  If AA(ii, ii) = 0 Then
    If AA(ii + 1, ii) <> 0 Then
      For j = 1 To nn
        swap AA(ii, j), AA(ii + 1, j)
      Next j
      swap b(ii), b(ii + 1)
    Else
      FoundPivot = False
      j = 0
      Do
        j = j + 1
        If j > nn Then Print #3, "error": Stop
        If AA(j, ii) <> 0 Then
          For k = 1 To nn
            swap AA(ii, k), AA(j, k)
          Next k
          swap b(ii), b(j)
          FoundPivot = True
        End If
      Loop Until (FoundPivot)
    End If
  End If
Next

For k = 1 To nn - 1
  For ii = k + 1 To nn
    m1 = AA(ii, k)
    m2 = AA(k, k)
    m = m1 / m2
    For j = k To nn
      AA(ii, j) = AA(ii, j) - m1 / m2 * AA(k, j)
    Next
    b(ii) = b(ii) - m1 / m2 * b(k)
  Next
Next

Dim Helpx As Double
Dim x(50) As Double

```

```

x(nn) = b(nn) / AA(nn, nn)
For k = nn - 1 To 1 Step -1
  Helpx = 0
  For j = k + 1 To nn
    Helpx = Helpx + AA(k, j) * x(j)
  Next
  x(k) = (b(k) - Helpx) / AA(k, k)
Next

For iii = 1 To nn
  b(iii) = x(iii)
Next iii

End Sub

```

A versão final do *TrHum 98* utiliza esta função. O motivo principal desta escolha prende-se com o facto de se pôr a hipótese de avançar num alargamento das funções do programa, em versões futuras, e onde poderão surgir matrizes sem desenvolvimento em banda para resolução de outros problemas. Ao utilizarmos o algoritmo geral, temos já a função definida para a resolução de qualquer tipo de matriz. A diferença de velocidade de computação das duas funções, nos computadores em que o programa foi testado (*Pentium 166MHz*, *200MHz*, *350 MHz* e *Pentium II 350 MHz*) não é sensível.

3.4.3 Cálculo do teor de humidade em função da humidade relativa

As funções *Hurelate2* e *Higroscop2* destinam-se a calcular o teor de humidade do material de equilíbrio higroscópico, para determinada humidade relativa do ambiente, e vice-versa. Correspondem a uma evolução funcional do presente programa, em relação ao *Trhumidade*, no sentido de permitir que os utilizadores possam adicionar materiais à base de dados sem terem que alterar o código do programa.

No *Trhumidade* as funções que descreviam o comportamento higroscópico dos materiais eram definidas no interior do programa. Adicionar um novo material à base de dados implicava necessariamente a alteração das subrotinas respectivas, e a compilação do programa.

No presente programa, o comportamento higroscópico é descrito por um conjunto de pares de valores, teor de humidade – humidade relativa, que estão definidos na base de dados do material. Estas duas funções, por interpolação linear, definem um conjunto de polinómios de grau 1 que caracterizam o comportamento do material.

```

Function Higroscop2(h)
Dim i As Integer
Dim w
i = 0
Do
i = i + 1
If gSSorpHR(i) > h And gSSorpHR(i - 1) < h Then
w = LinearInterpolation(h, gSSorpHR(i - 1), gSSorpWHR(i - 1), gSSorpHR(i),
gSSorpWHR(i))
ElseIf gSSorpHR(i) = h Then
w = gSSorpWHR(i)
End If
Loop While i < gSSorpCount
Higroscop2 = w
End Function

```

```

Function Hurelat2(w)
Dim i As Integer
Dim h
i = 0
Do
i = i + 1
If gSSorpWHR(i) > w And gSSorpWHR(i - 1) < w Then
h = LinearInterpolation(w, gSSorpWHR(i - 1), gSSorpHR(i - 1), gSSorpWHR(i),
gSSorpHR(i))
ElseIf gSSorpWHR(i) = w Then
h = gSSorpHR(i)
End If
Loop While i < gSSorpCount
If w > gSSorpWHR(gSSorpCount) Then h = 100
Hurelat2 = h
End Function

```

```

Function LinearInterpolation(X, X1, Y1, X2, Y2)
LinearInterpolation = Y1 + (X - X1) * (Y1 - Y2) / (X1 - X2)
End Function

```

O resultado encontrado corresponde à coordenada do ponto da recta para o valor da ordenada fornecido.

Outras rotinas utilizadas no programa foram incluídas no Anexo B para consulta.

4 DESCRIÇÃO DO PROGRAMA 40

4.1	CARACTERIZAÇÃO GERAL.....	40
4.2	MÓDULO 1 - SIMULAÇÃO NUMÉRICA DA TRANSFERÊNCIA DE CALOR E HUMIDADE	40
4.2.1	<i>Dados necessários</i>	40
4.2.2	<i>Definição das condições iniciais</i>	47
4.3	MÓDULO 2 – CÁLCULO DA PRESSÃO SATURAÇÃO.....	59
4.4	MÓDULO 3 – CÁLCULO DO TEOR DE HUMIDADE DE UM MATERIAL EM FUNÇÃO DA HUMIDADE RELATIVA 61	
4.5	MÓDULO 4 – CÁLCULO DA RESISTÊNCIA TÉRMICA DE UM MATERIAL EM FUNÇÃO DO SEU TEOR DE HUMIDADE.....	62
4.6	MÓDULO 5 – CÁLCULO DOS PERFIS DE TEMPERATURA ESTABILIZADOS PARA PAREDES COM DIVERSAS CAMADAS.	63
4.7	MÓDULO 6 – CÁLCULO DA TEMPERATURA SUPERFICIAL DE UMA PAREDE EM FUNÇÃO DA LOCALIZAÇÃO, RADIAÇÃO E VENTO.	64
4.8	MÓDULO 7 – CONVERSÃO DE UNIDADES.	67
4.9	POTENCIALIDADES E LIMITAÇÕES DO PROGRAMA	68

4 Descrição do programa

4.1 Caracterização geral

Como foi já referido o programa é composto por sete módulos com funções distintas mas interrelacionáveis. Esses módulos estão disponíveis, no arranque do programa, através de uma janela de escolha múltipla, que permite a navegação entre as diversas funções que constituem o programa.

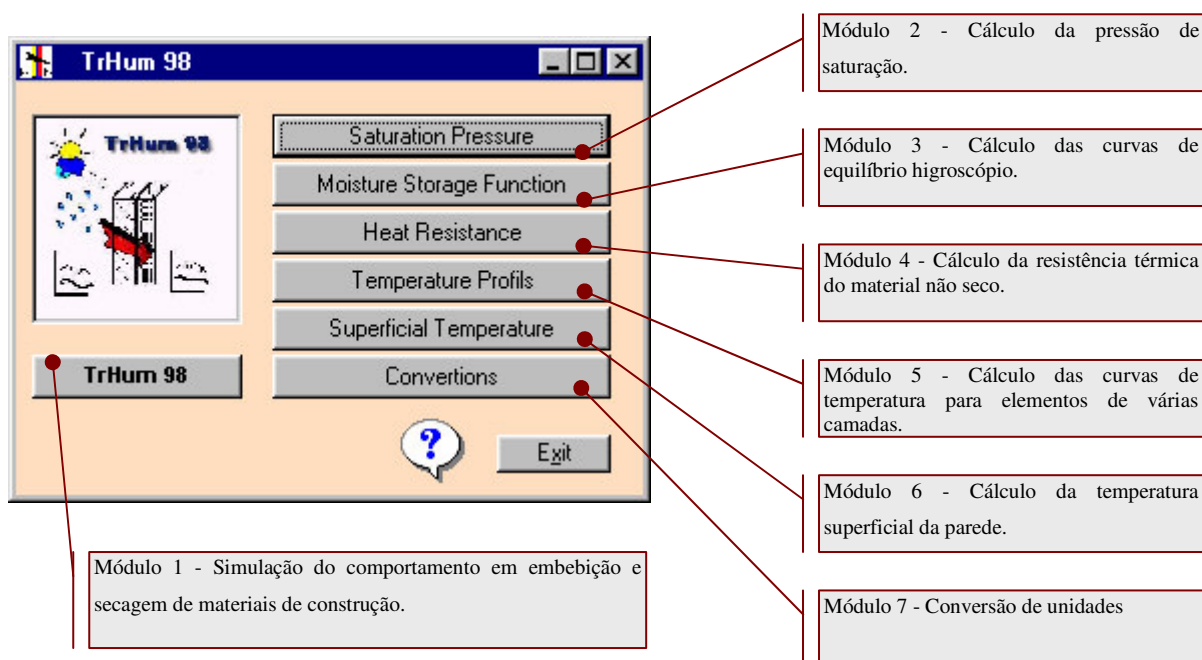


Fig. 13 – Relação entre os diversos módulos do programa

4.2 Módulo 1 - Simulação Numérica da transferência de calor e humidade

4.2.1 Dados necessários

A simulação numérica da secagem ou embebição de materiais de construção, pela sua complexidade, necessita da definição, e da conseqüente fornecimento ao programa, de um conjunto de dados de caracterização, referentes às condições iniciais, às características dos materiais e às condições de simulação. A quantidade de dados a fornecer ao programa é

bastante extensa, pelo que, para facilitar e tornar mais cómoda e atraente a entrada desses dados, optou-se por fazer o seu agrupamento e a visualização em várias janelas.

Nestas janelas existem campos que tanto servem para o preenchimento de dados, como para a visualização e edição dos dados já definidos. Estes campos são, na sua quase totalidade, campos de preenchimento obrigatório, sem o qual o programa não funciona.

O sistema de variáveis do programa está estruturado a dois níveis, variáveis locais e variáveis globais. Quando é feito o preenchimento de um campo, o valor é armazenado numa variável local, que só será transmitida à variável global respectiva no momento em que o utilizador passe a uma outra janela, com um comando de aceitação, ou através das setas de navegação, ou do botão de comando *exit*. Se a saída for feita pelo botão de comando *cancel*, as alterações feitas às variáveis locais não serão transmitidas às variáveis globais, preservando assim os valores anteriormente entrados.

Da mesma forma, quando é aberta uma janela, são atribuídos à variáveis locais os valores das variáveis globais, armazenados em memória nesse momento, permitindo assim a visualização de dados já definidos.

Todos os valores, quer de variáveis, quer de resultados, usados numa simulação estão guardados na memória do computador, não sendo necessário o acesso a disco para leitura ou escrita dos mesmos, durante o cálculo. Isto permite que o cálculo seja mais rápido, tem contudo o inconveniente de poder implicar um computador com mais memória. Não foi estabelecido o mínimo de memória *RAM* necessário, fizemos contudo testes com computadores Pentium 160 MHz com 64 Mb de *RAM* (a configuração mais baixa a que tivemos acesso) e o programa funcionou, mesmo para simulações mais dilatadas no tempo.

Para guardar definitivamente um projecto, deverá ser escolhida a opção *Save Project*, para que a totalidade dos parâmetros de caracterização sejam escritos no ficheiro de projecto. Todas as variáveis e todos os valores calculados são limpos quando o programa termina.

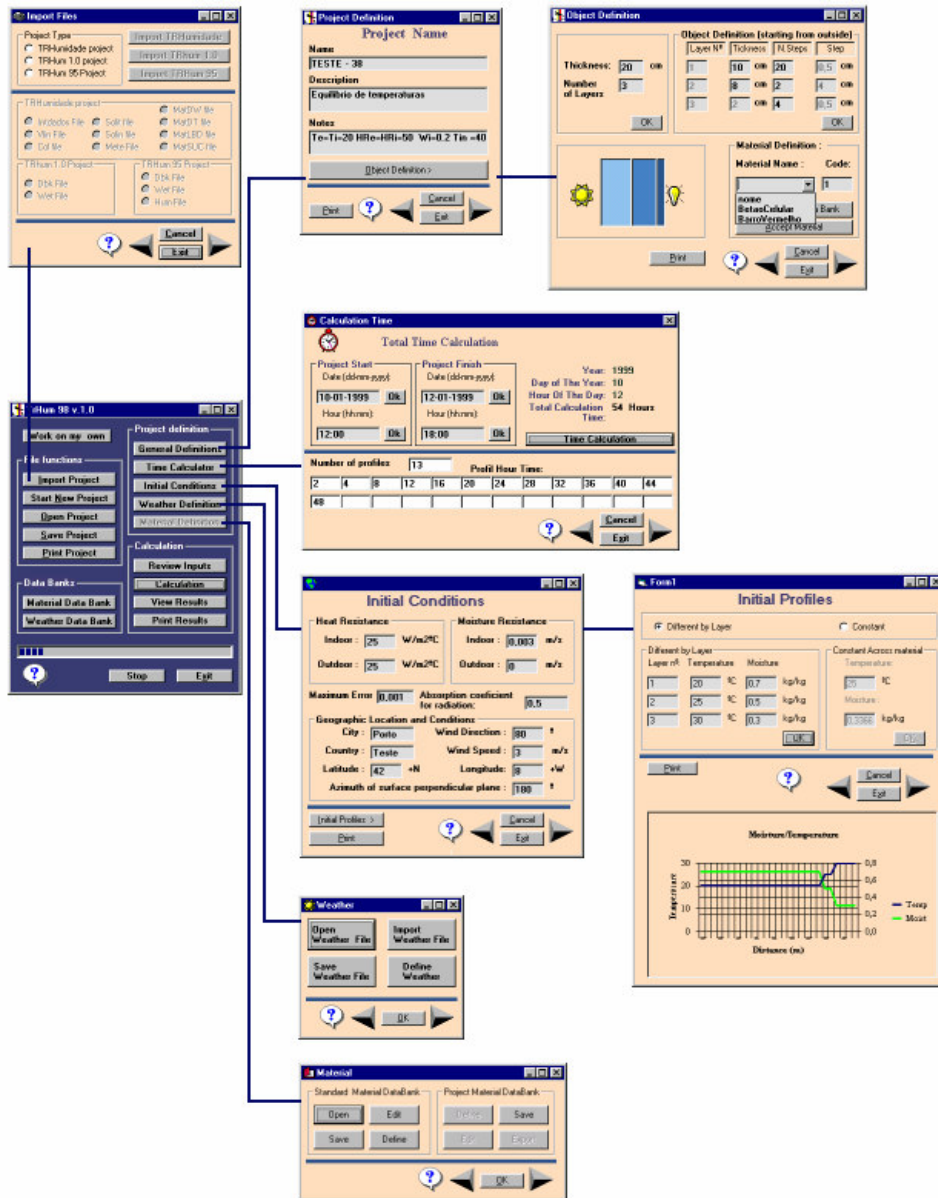


Fig. 14 – Relação da janela de tarefas com as diversas janelas de entrada de dados no programa

A estrutura base de entrada de dados no programa é constituída pelo seguinte conjunto de janelas:

- Definição do projecto [Fig. 15] – Elementos de identificação geral da simulação
- Definição do objecto [Fig. 16] – Caracterização da amostra
- Cálculo do tempo [Fig. 18] – Definição das datas de início e fim da simulação, e das horas dos perfis
- Condições iniciais [Fig. 19] – Definição dos dados de caracterização que são constantes ao longo da simulação
- Perfis iniciais [Fig. 20] – Definição dos perfis iniciais de temperatura e teor de humidade instalados na amostra
- Clima [Fig. 21, Fig. 22, Fig. 23] – Caracterização do clima
- Caracterização dos materiais [Fig. 24] – Escolha ou definição do material e suas características

A janela de tarefas, pela sua organização, permite uma melhor estruturação e definição da forma como os dados devem ser inseridos no programa [Fig. 14].

4.2.1.1 Dados genéricos sobre o Projecto

A janela de definição do projecto [Fig. 15] é a primeira na sequência de janelas encadeadas, e a única cujo total preenchimento não é obrigatório. Serve essencialmente para guardar notas sobre o projecto em questão, tais como o nome, que o identifica, que poderá ser igual ou diferente do nome do ficheiro onde ele está arquivado, uma pequena descrição e algumas notas que o utilizador julgue convenientes, para uma definição maior das condicionantes do projecto.

Estes dados serão apresentados, após a simulação, na janela de resultados, juntamente com os perfis de temperatura e teor de humidade, servindo para a fácil identificação do projecto.

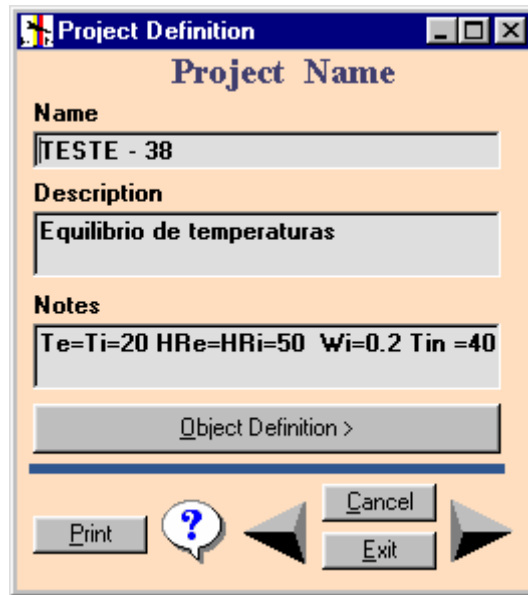


Fig. 15 - Janela de definição do projecto

Temos sempre a possibilidade de solicitar ao programa informações adicionais sobre o funcionamento do mesmo e sobre o tipo e forma de introdução de dados, opção disponível no ícone de ajuda. Podemos ainda imprimir cada uma das janelas, com a informação entrada, para registo mais operativo do projecto.

Quando se sai desta janela passa-se à janela de definição do objecto. [Fig. 16]

4.2.1.2 Definição do objecto

Na janela de definição do objecto [Fig. 16] todos os campos são de preenchimento obrigatório, e servem para caracterizar a amostra virtual que irá ser alvo da simulação.

Os dados a fornecer deverão ser os seguintes:

- Designação do material - O material deverá ser escolhido dentre os disponíveis na base de dados, e deverá ser aceite para que todos os parâmetros de caracterização desse material passem da base de dados para o sistema de variáveis globais e possam ser gravados no ficheiro do projecto. Deverá ser escolhida uma base de dados e carregada no programa para que os materiais estejam disponíveis para utilização. Poderá optar-se por não seleccionar o material nesta altura, podendo tal ser feito na janela de definição do material.

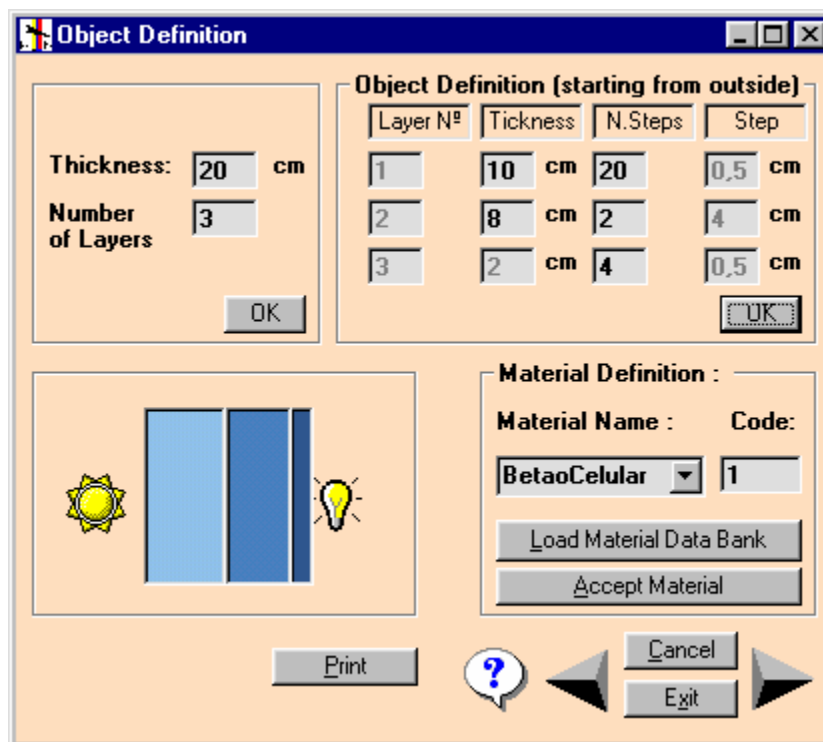


Fig. 16 - Janela de definição do objecto – situação de 3 sub-camadas

- Espessura - Espessura total da amostra, em cm.
- Número de sub-camadas – Número de sub-camadas teóricas em que se divide o material, para estabelecimento de espessura da malha para cálculo. Esta subdivisão permite fazer a diferenciação da espessura da malha, reduzindo-a em zonas onde seja necessário uma maior definição, e aumentando-a noutras zonas onde tal não se verifique. O número máximo de sub-camadas possível na presente versão do programa é de três.
- Número e espessura da sub-camada, número de passos e espaçamento da malha. Definições relativas a cada uma das sub-camada consideradas. Identificação da sub-camada pelo número, definição da sua espessura e do número de passos utilizados para a definição do espaçamento da malha, valores que são interdependentes

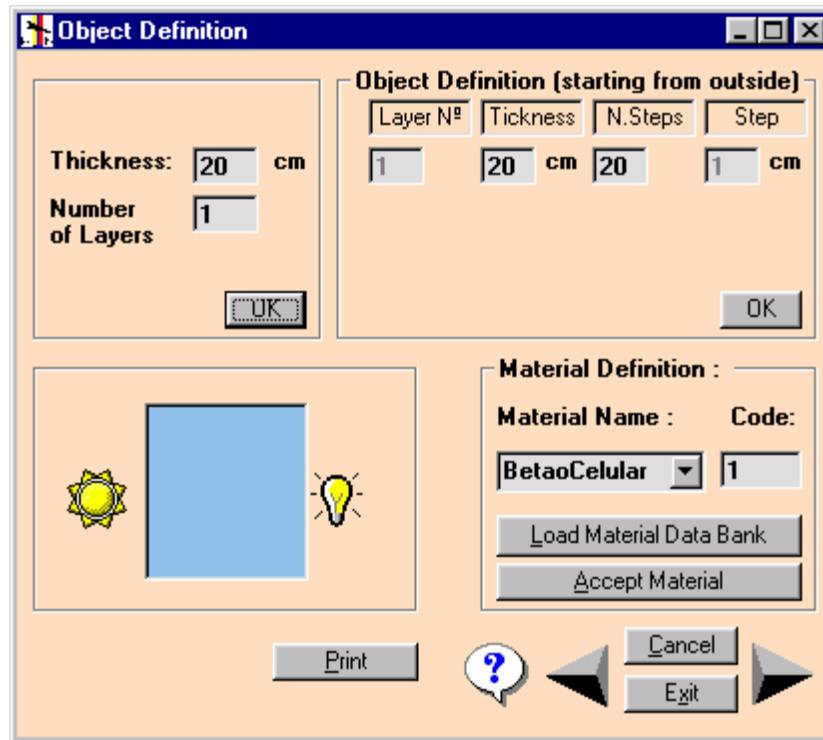


Fig. 17 - Janela de definição do objecto – situação de 1só sub-camada

É necessário o preenchimento da totalidade dos campos visíveis e acessíveis. O programa tem algumas funções automatizadas, para por um lado retirar algum trabalho ao utilizador mas essencialmente para limitar os erros de preenchimento.

Assim, só é possível o preenchimento das características das camadas até ao número de camadas definidas na parte superior da janela, ficando os restantes campos invisíveis [Fig. 17]. O estabelecimento do espaçamento da malha é feito automaticamente em função do número de passos considerados, por divisão da espessura da sub-camada pelo número de passos. A espessura das sub-camadas é verificada e corrigida em função da espessura total da amostra, e sempre que se verifica alguma alteração.

4.2.1.3 Definição do tempo de simulação

O programa necessita que lhe sejam fornecidas as datas de início e fim da simulação, calculando automaticamente a duração da mesma.

A definição das datas de início e fim é importante, e determinante, quando se simula em condições reais, sob a influência da radiação e do vento.

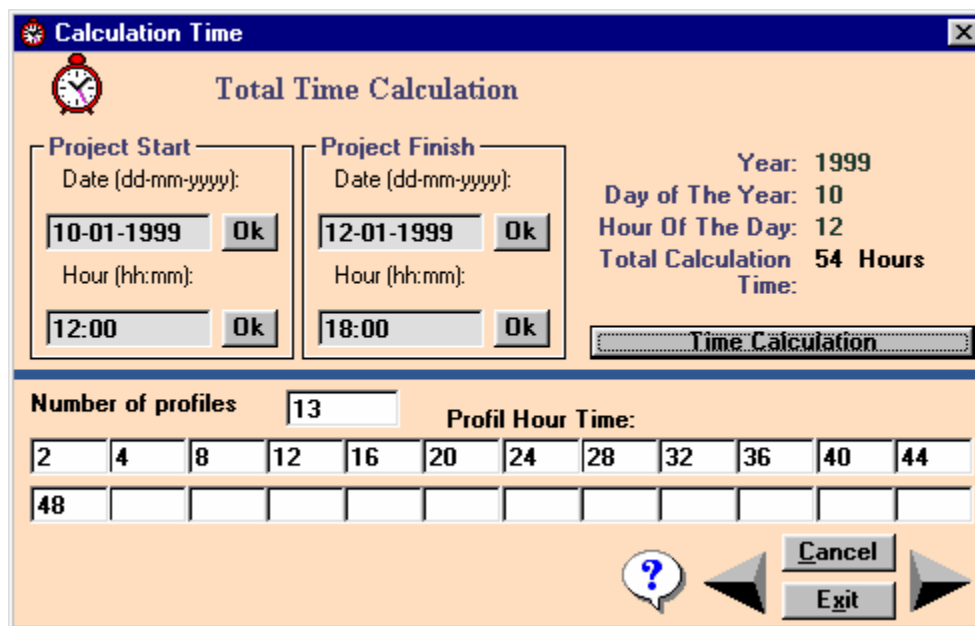


Fig. 18 - Janela de definição do tempo de simulação

O programa calcula as variáveis internas que definem o número de anos, os dias desde o início do ano e as horas desde o início do dia, dados que vão ser utilizados na função *Sol*, que calcula a temperatura fictícia ar-sol.

Nesta janela [Fig. 18] o utilizador define também o número de perfis pretendidos, bem como as respectivas horas. Estes perfis serão apresentados na janelas de resultados, em forma gráfica, e serão escritos num ficheiro de resultados, sob forma analítica. A necessidade de estabelecer o número de perfis pretendidos prende-se quer com a impossibilidade de dar os resultados em contínuo, quer pela dificuldade da sua interpretação, se assim fossem fornecidos. Foram feitos testes nesse sentido, para simulações de poucas horas, e verificou-se que o ficheiro de resultados assumia proporções não compatíveis nem com a gestão do disco duro do computador, nem com a maior parte dos programas que poderiam fazer a leitura do mesmo. Um ficheiro de resultados de uma simulação de 24 horas, com um passo de 6 minutos, atingiu os 200 Mb.

4.2.2 Definição das condições iniciais

A janela da Fig. 19 é referente às condições iniciais e parâmetros gerais de caracterização do material e da simulação. Deverão ser definidos os coeficientes de condutância térmica

superficial, interior e exterior (h_i ; h_e) e os coeficiente de transferência superficial de humidade, interior e exterior (β_i ; β_e).

Superficial conductance		Moisture Coefficient	
Indoor :	25 W/m ² °C	Indoor :	0,003 m/s
Outdoor :	25 W/m ² °C	Outdoor :	0,003 m/s

Maximum Error: 0,001 Absorption coefficient for radiation: 0,5

Geographic Location and Conditions			
City :	Porto	Wind Direction :	80 °
Country :	Portugal	Wind Speed :	3 m/s
Latitude :	42 +N	Longitude :	8 +W
Azimuth of surface perpendicular plane :		180 °	

Buttons: Initial Profiles >, Print, ? (Help), Cancel, Exit

Fig. 19 - Janela de definição de condições constantes

Deverá também ser fornecido ao programa a precisão pretendida no processo iterativo, utilizado para a definição da convergência do resultado, isto é, o erro máximo admissível entre duas iterações.

O programa necessita ainda que sejam definidas as latitude e longitude do lugar, a direcção e velocidade do vento, e o azimute da perpendicular à fachada, para a definição da orientação da amostra.

Os campos referentes à cidade e ao país são de preenchimento facultativo e funcionam sobretudo como um auxílio informativo ao utilizador.

Numa versão futura pretendemos implementar um sistema gráfico na determinação de coordenadas geográficas, a partir de mapas de zonas do território.

4.2.2.1 Definição dos perfis iniciais de temperatura e teor de humidade

O presente programa permite que se divida o material em estudo em diferentes camadas e se especifiquem valores diferentes de espaçamento da malha e de condições iniciais de humidade e temperatura. O objectivo é o de possibilitar simulações de materiais com condições iniciais variáveis de temperatura e teor de humidade.

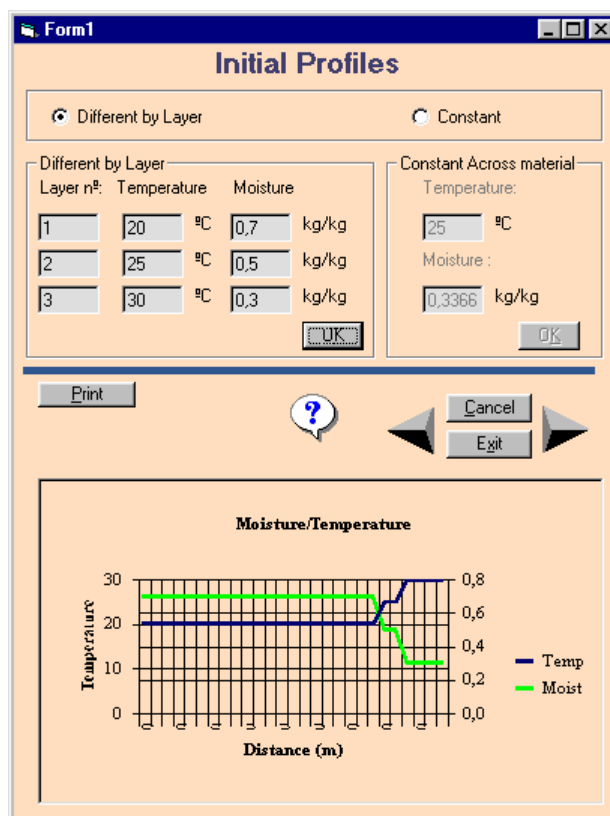


Fig. 20 - Janela de definição dos perfis iniciais da amostra

O utilizador poderá escolher entre perfis iniciais constantes em todo o material, ou variáveis por sub-camada, devendo em qualquer um dos casos definir a temperatura e os teores de humidade iniciais.

Os perfis de temperatura e humidades entrados são apresentados em forma gráfica [Fig. 20], para uma melhor percepção das condições iniciais.

4.2.2.2 Definição do ficheiro de clima

O *TrHum 98* permite o estabelecimento de várias definições diferentes do clima, adaptadas às características da simulação pretendida. Os parâmetros de caracterização do clima poderá ser definido na altura ou lido a partir de um ficheiro pré existente.

O programa permite a definição de clima constante ou variável, a partir da definição individual de todos os parâmetros de caracterização.

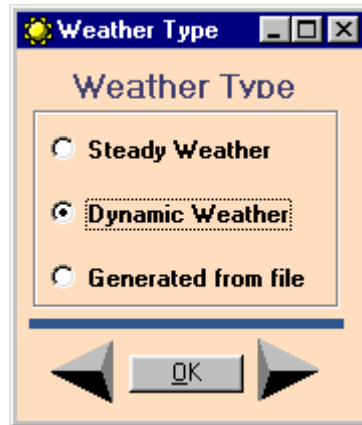


Fig. 21 - Janela de definição do tipo de ficheiro de clima

Tem ainda a hipótese de gerar o clima a partir das bases de dados existentes no programa, ou a partir de ficheiros de duração inferior ao tempo de simulação, por repetição do mesmo várias vezes até atingir ou ultrapassar o tempo necessário.

O utilizador deverá escolher [Fig. 21] que tipo geração de clima pretende para que o programa passe à janela respectiva.

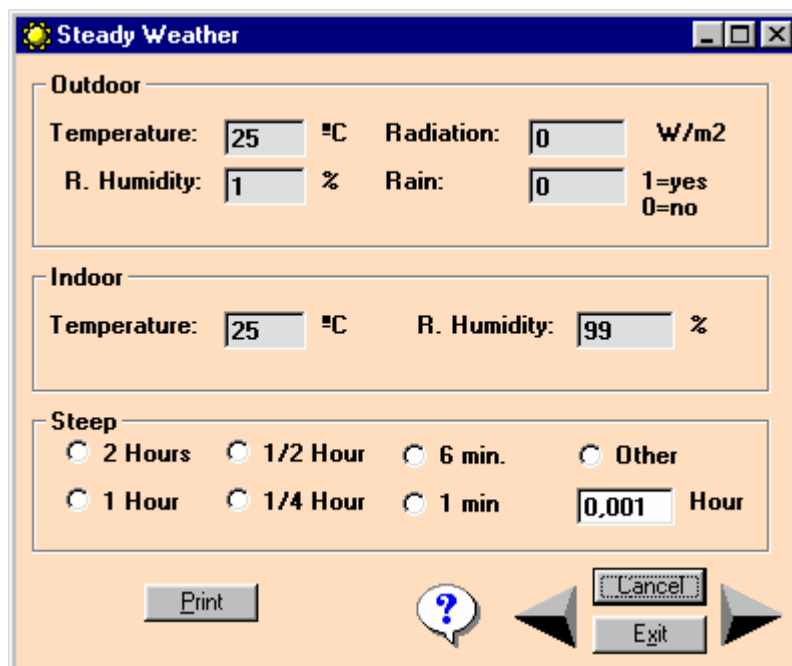


Fig. 22 - Janela de definição de clima constante ao longo da simulação

Os dados de caracterização do clima exterior necessários à simulação são, para cada instante considerado; a temperatura, a humidade relativa, a radiação e a chuva. No que se refere ao interior deve fornecer-se a temperatura e a humidade relativa.

Esses valores poderão ser constantes ao longo de toda a simulação, ou podem ser variáveis tentando fazer a aproximação a uma situação real determinada.

No caso da definição de um clima constante, o utilizador necessita apenas de escolher qual é o passo temporal pretendido para a simulação. Quando a definição do clima é variável, para além do passo temporal da simulação, o utilizador necessita também de definir o intervalo de tempo em que as condições se mantêm constantes.

Deve ainda ser fornecido o passo temporal da simulação, e o número de horas em que as condições se mantêm constantes.

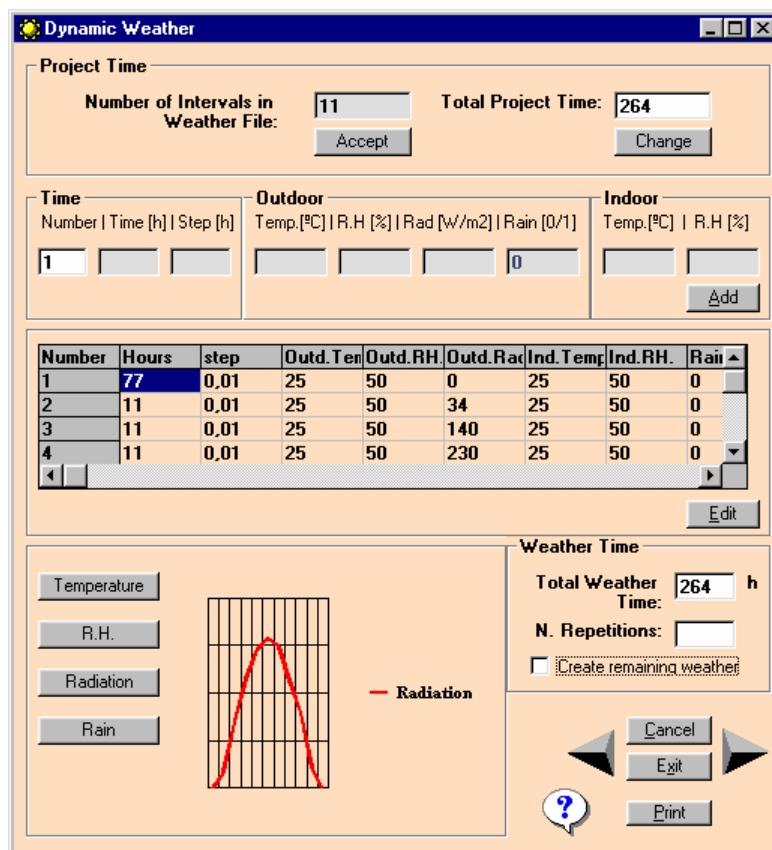


Fig. 23 - Janela da definição de clima variável

O programa calcula os novos perfis de temperatura e teores de humidade com intervalos de tempo determinados. Esses intervalos de tempo são definidos pelo passo do ficheiro de clima. Quanto menor for o passo considerado, maior será a morosidade do cálculo mas também maior será o rigor da simulação. Foram feitas análises da performance do programa, na sua relação com o passo de simulação, que serão apresentadas no capítulo 5.

Na janela de definição de clima variável o utilizador deverá definir linha a linha os parâmetros de caracterização, escolhendo depois a opção de adicionar para que a mesma passe a fazer parte do ficheiro de clima

Na situação de clima variável [Fig. 23] é possível ver num gráfico com a distribuição ao longo do tempo do parâmetro de caracterização pretendido, temperatura, humidade relativa, radiação ou precipitação, para uma leitura mais directa das condicionantes climáticas. Nesta janela é ainda possível alterar a duração total do projecto.

4.2.2.3 Manuseamento de Bases de dados

O programa dispõe de bases de dados de caracterização de materiais e de caracterização de clima, podendo ser sempre adicionadas mais por parte do utilizador, ou em ficheiros individuais ou aglutinados a ficheiros existentes.

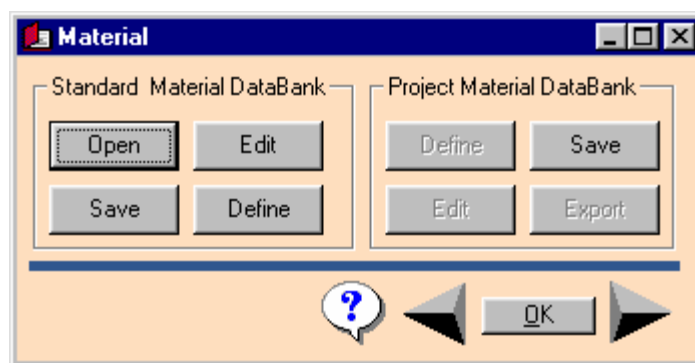


Fig. 24 – Janela de Acesso às diferentes funções relacionadas com o manuseamento de bases de dados de características dos materiais.

No caso dos ficheiros de clima o programa aceita também bases de dados executadas em *Access*, que ficam disponíveis através do *TrHum 98* para consulta, e edição se necessário.

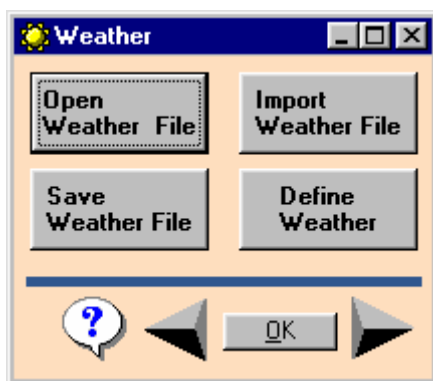


Fig. 25 – Janela de ligação às funções relacionadas com o clima.

4.2.2.4 Revisão dos dados entrados e cálculo

A Fig. 26 representa a janela de revisão dos valores entrados, com o resumo da totalidade dos parâmetros definidos. Funciona como um meio de verificação de toda a informação relativa ao projecto, conjuntamente com as janelas de entrada de dados.

Após a entrada, ou a leitura, de todos os dados, é finalmente possível iniciar o cálculo. Quando o utilizador escolhe a opção calcular, o programa vai verificar se todos os valores necessários à simulação foram fornecidos, e vai fazer a validação individual de todos os valores, confirmando se o valor entrado era apropriado ao campo em que se inseria. Por exemplo, se determinado campo tem que ser preenchido com uma data, o computador verifica se o valor entrado tem significado como data. Quando algum dos valores não foi fornecido, ou foi fornecido com erro, o programa pede ao utilizador para verificar os dados entrados, indicando a janela em que o erro se verifica, para uma maior comodidade e simplicidade.

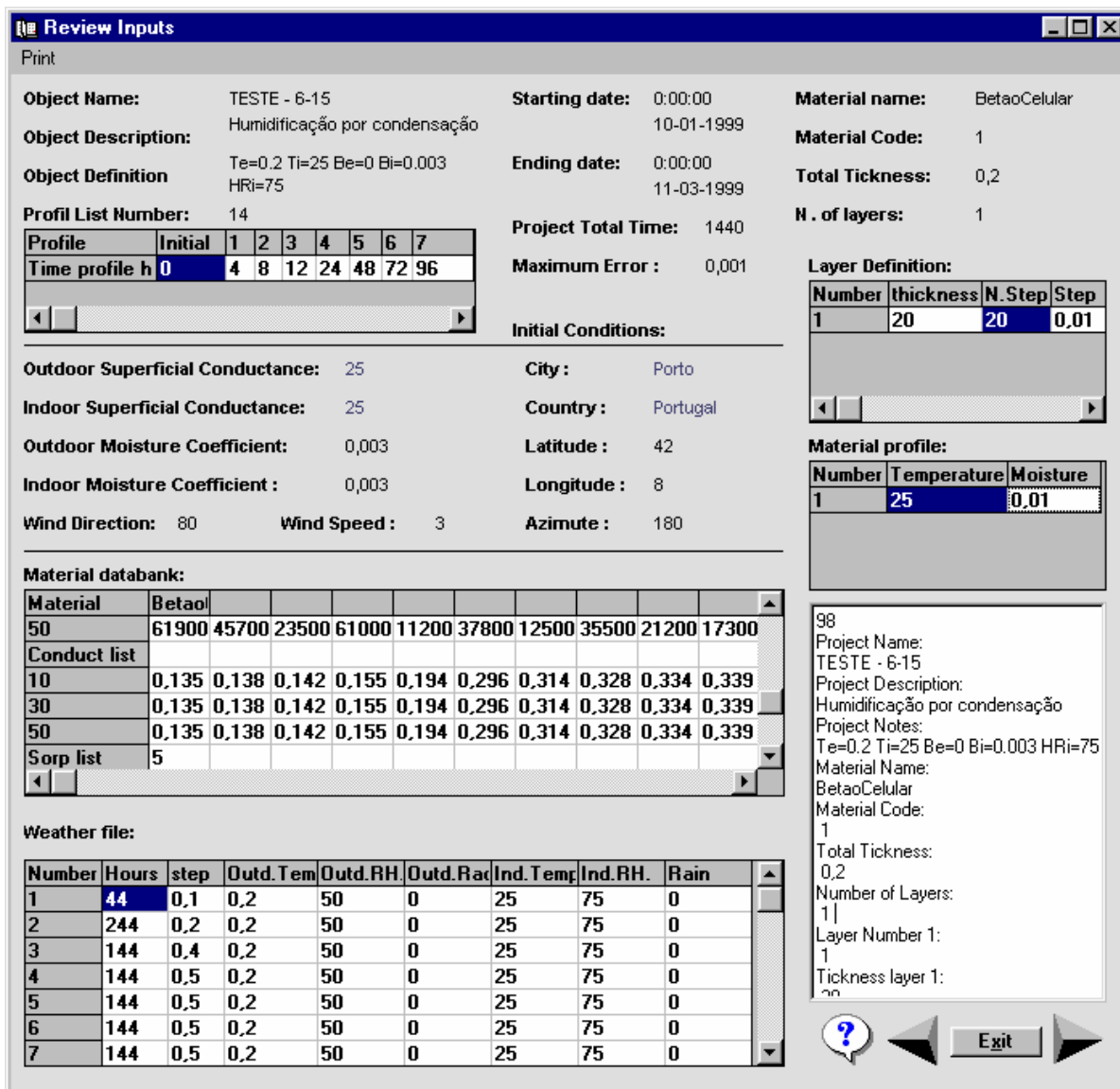


Fig. 26 – Janela de verificação dos dados

Após esta verificação pode-se então proceder à simulação do comportamento do elemento à transferência conjunta de calor e humidade, de acordo com os parâmetros definidos. O programa permite a saída dos resultados em forma de ficheiro de texto, ou sob a aparência de gráfico bidimensional de linhas, que nos apresenta quer a variação da temperatura, quer a variação do teor de humidade, nos momentos definidos pelo utilizador e ao longo da espessura da amostra. Permite ainda a visualização dos resultados noutro tipo de gráficos sempre que o utilizador o solicite.

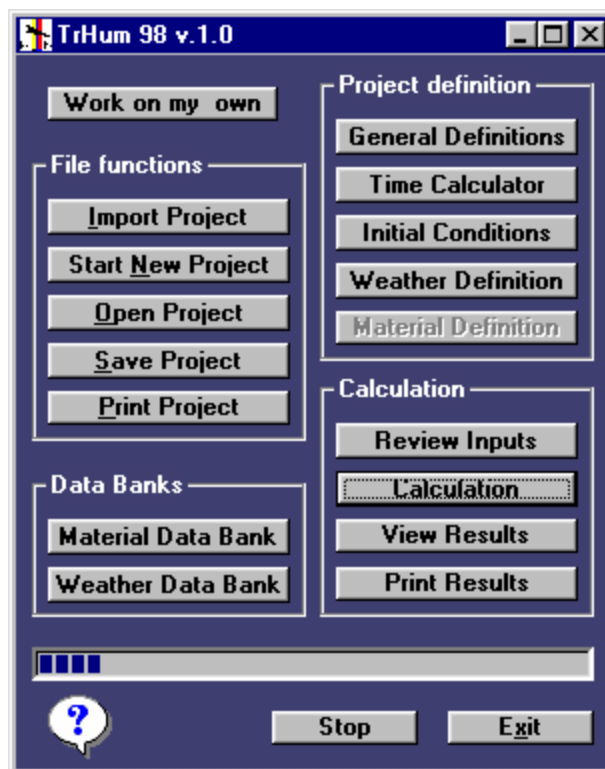


Fig. 27 – Janela de tarefas

Quando o cálculo termina, o programa apresenta uma pequena janela de mensagem, informando o utilizador [Fig. 27]. Durante o tempo em que o computador efectua o cálculo, é mostrada uma barra de progresso parcial [Fig. 28]. Pelo seu movimento é possível, em primeiro lugar, saber que o computador se encontra efectivamente a trabalhar, pois a longa duração de algumas simulações poderiam levar o utilizador a pensar que o computador tinha encravado. O número de células acendidas durante o cálculo dá uma ideia ao utilizador do número de iterações que estão a ser necessárias para a convergência do resultado dentro dos parâmetros de erro definidos, uma vez que se fez depender o movimento da barra do número de iterações. Se a barra for totalmente preenchida por células, a totalidade das iterações definidas no programa (20) estão a ser usadas e o cálculo é interrompido pela ultrapassagem desse valor. Por outro lado, se a barra só for parcialmente preenchida significa que o programa não necessita de efectuar as iterações todas para atingir a convergência.

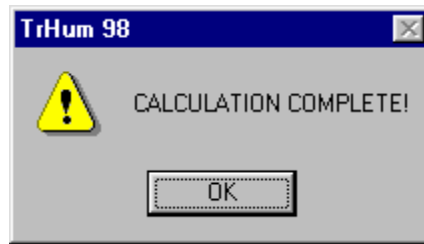


Fig. 28 – Janela de mensagem – Informação da conclusão do cálculo.

Para ver e imprimir os resultados da simulação, o utilizador deve escolher essa opção na janela de tarefas. A janela de visualização de resultados será mostrada [Fig. 29].

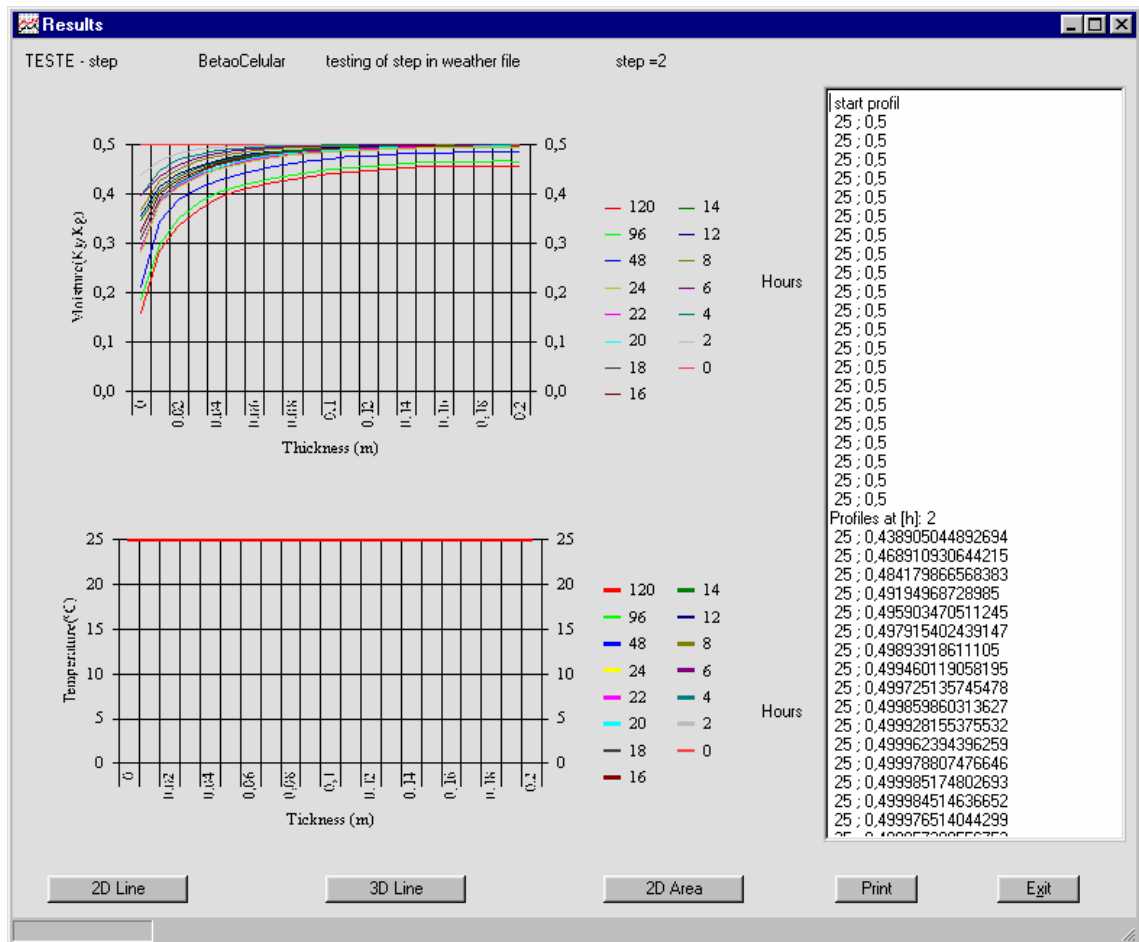


Fig. 29 – Janela de visualização de resultados

O programa permite a visualização dos resultados em três tipos de gráfico para que o utilizador possa escolher o que mais se adapta às suas preferências. A saída de resultados

que o programa apresenta por defeito é uma gráfico bidimensional de linhas [ver Fig. 30]. O utilizador poderá escolher a visualização num gráfico tridimensional [Fig. 31] ou num gráfico bidimensional de áreas [Fig. 32].

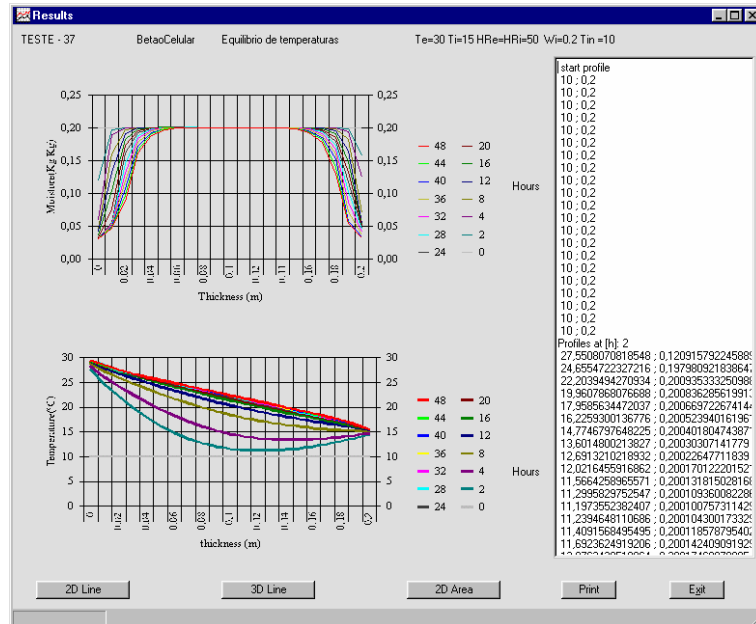


Fig. 30 – Janela de visualização de resultados – primeira forma de apresentação

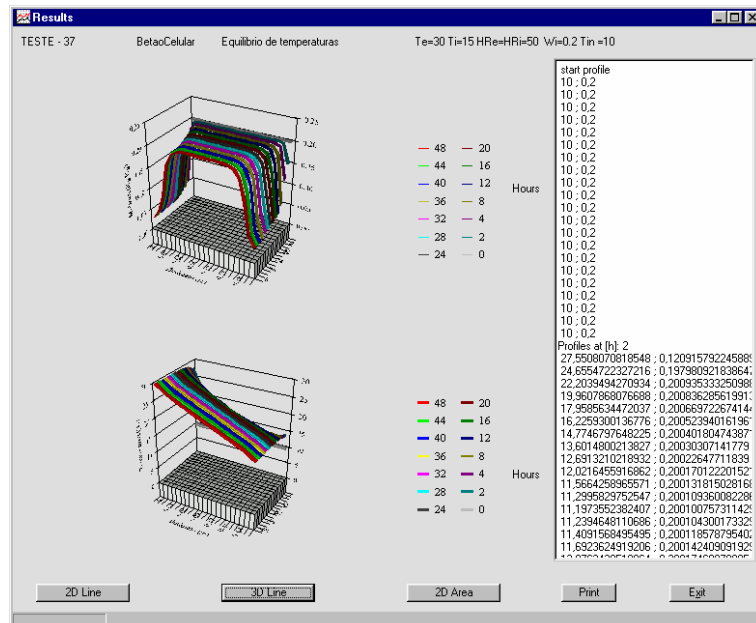


Fig. 31 – Janela de visualização de resultados – segunda forma de apresentação

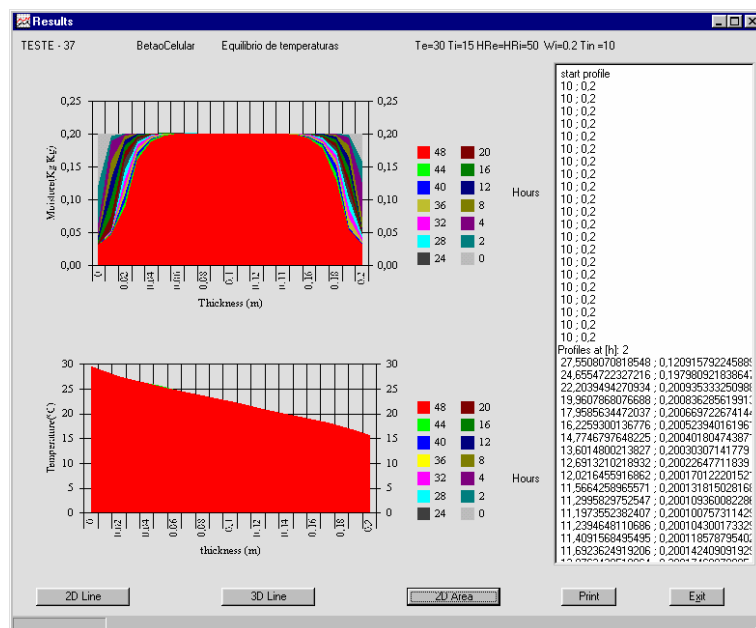


Fig. 32 – Janela de visualização de resultados – terceira forma de apresentação

Os resultados da simulação são sempre guardados em dois ficheiros, com o mesmo nome do ficheiro de projecto (se outro não for especificamente escolhido) mas com extensões diferentes, “.res” para o primeiro ficheiro de resultados, que vai sendo escrito ao longo do cálculo, e “.csv”, para o segundo ficheiro, escrito só no final da simulação, e que equivale a uma transformação do ficheiro inicial e se destina essencialmente ao tratamento dos resultados da simulação por outros programas, designadamente o *Excel* [ver Anexo A].

4.3 Módulo 2 – Cálculo da pressão saturação

Tal como foi referido no capítulo anterior, o programa *TrHum 98* comporta vários módulos de funcionamento independente. Estes módulos partilham funções internas e muitas vezes o sistema de variáveis globais.

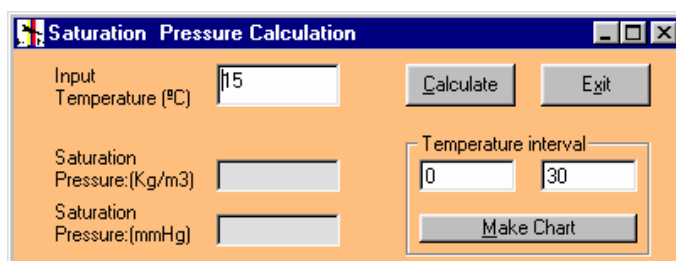


Fig. 33 – Janela inicial do cálculo da pressão de saturação

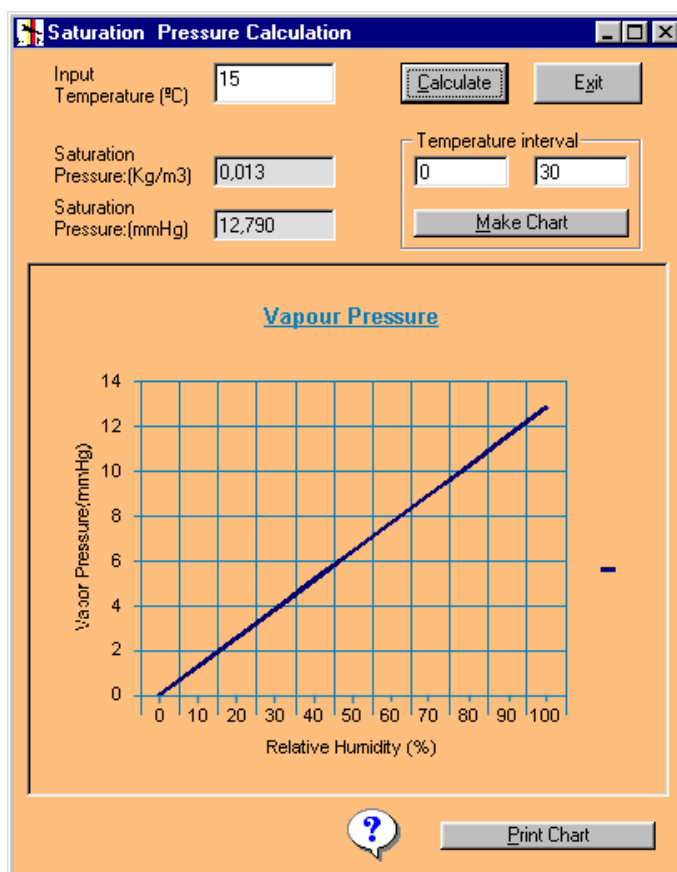


Fig. 34 – Pressão de vapor para várias humidades relativas

Um desses módulos é o do cálculo da pressão de saturação que nos permite a rápida determinação da pressão de saturação a uma dada temperatura, em dois sistemas de unidades; kg/m³ e mmHg. Dá-nos ainda, de uma forma gráfica, a pressão de vapor, a essa temperatura, para várias humidades relativas.

Permite também a determinação da pressão de saturação ao longo de determinado intervalo, escolhido pelo utilizador, e a visualização do diagrama psicrométrico para esse mesmo intervalo, e para alguns patamares de humidades relativas., com a possibilidade da sua impressão.

Foi individualizado num pequeno utilitário.

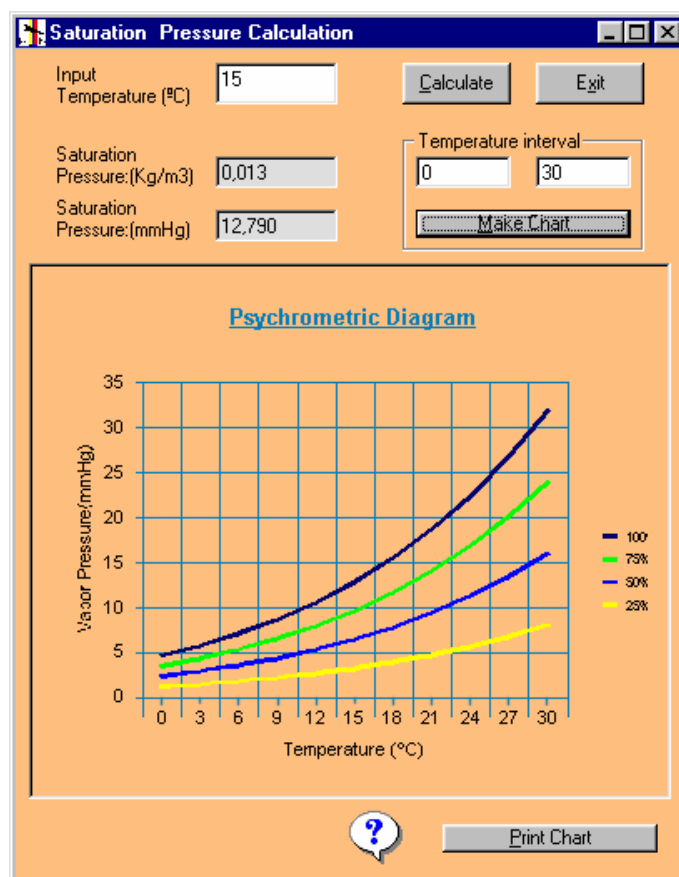


Fig. 35 – Diagrama psicrométrico para o intervalo de temperaturas fornecido

4.4 Módulo 3 – Cálculo do teor de humidade de um material em função da humidade relativa

Este módulo permite-nos, para qualquer humidade relativa do ambiente, saber qual é o teor de humidade higroscópico, para os materiais constantes da base de dados. Tem uma utilização preferencial no estudo e monitorização dos comportamentos ambientais, e na sua relação com o aparecimento de patologias em elementos de construção. Faz ainda um gráfico da relação entre humidade relativa e teor de humidade para uma apreensão mais rápida das características e comportamento higroscópios de um material.

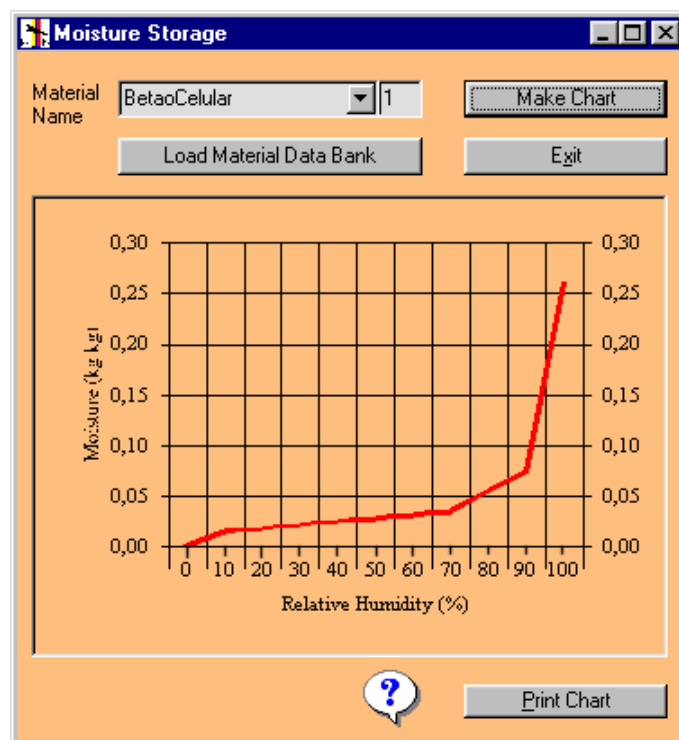


Fig. 36 – Teor de humidade em função da humidade relativa

4.5 Módulo 4 – Cálculo da resistência térmica de um material em função do seu teor de humidade.

A resistência térmica de um material diminui quando o seu teor de humidade aumenta. A determinação desta diminuição é importante pois pode marcar por exemplo, a diferença entre ter ou não condensações superficiais num elemento de construção.

Este utilitário permite, para os materiais constantes da base de dados, a determinação da resistência térmica possibilitando assim a correcção de coeficientes de condutibilidade térmica e resistências térmicas de materiais não secos.

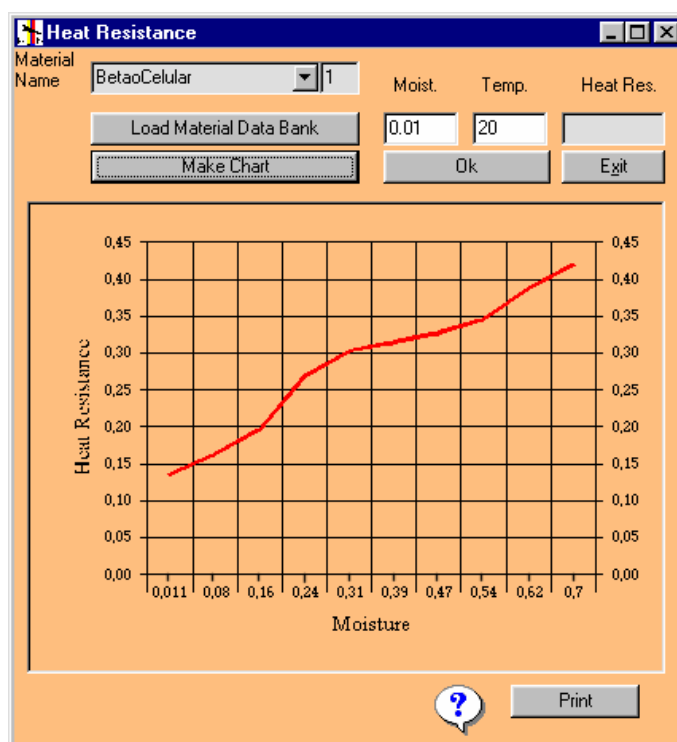


Fig. 37 – Resistência térmica em função do teor de humidade do material

Nesta altura o seu uso está restringido pela dimensão da base de dados do programa, pelo que a ampliação da mesma, no sentido de englobar os materiais mais correntes de construção, permitiria tornar este modulo uma ferramenta verdadeiramente efectiva na análise e verificação do comportamento de elementos de construção.

4.6 Módulo 5 – Cálculo dos perfis de temperatura estabilizados para paredes com diversas camadas.

As diferentes resistências térmicas dos diversos elementos que constituem uma parede, e as suas diferentes espessuras, condicionam os perfis de temperaturas instalados. A sua determinação é tanto mais complexa quanto maior número de camadas estiverem envolvidos.

Este utilitário permite rapidamente visualizar os perfis de temperatura instalados numa paredes. Uma das utilizações que lhe advínhamos será a de auxiliar na análise de elementos de construção para determinar onde se dão as grandes variações de temperatura, que podem danificar os materiais, ou para perceber se existe necessidade ou não de colocação de barreiras pára-vapor.

Poderão ser utilizados materiais não constantes da base de dados, desde que seja fornecido o valor do coeficiente de condutibilidade térmica. Deverá ser também fornecida a espessura relativa de cada camada. O programa aceita até 5 camadas diferentes.

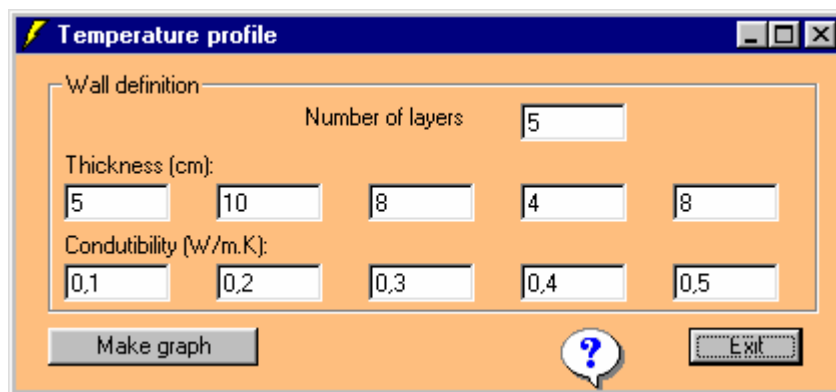


Fig. 38 – Janela de definição das diferentes camadas de uma parede

4.7 Módulo 6 – Cálculo da temperatura superficial de uma parede em função da localização, radiação e vento.

A radiação solar incidente sobre uma parede vai influenciar a temperatura da mesma. Esta rotina permite o cálculo da temperatura fictícia ar-sol que a parede atingiria em condições de localização, vento e radiação específicas. O algoritmo de resolução utilizado foi desenvolvido por V. Abrantes [17], com as adaptações introduzidas por V. Peixoto de Freitas [1].

Os valores de radiação disponibilizados pelas estações meteorológicas referem-se a radiação sobre superfície horizontal, tornando-se pois necessário calcular a radiação que efectivamente atinge o elemento de construção em estudo, um paramento vertical, para a orientação definida.

Hour	Temp.	Radia.	Wall Temp
8	26	34	
9	27	140	
10	28	230	
11	29	292	
12	30	316	
13	30	297	
14	29	226	
15	28	159	
16	27	44	

Fig. 39 – Janela inicial – cálculo para uma hora específica

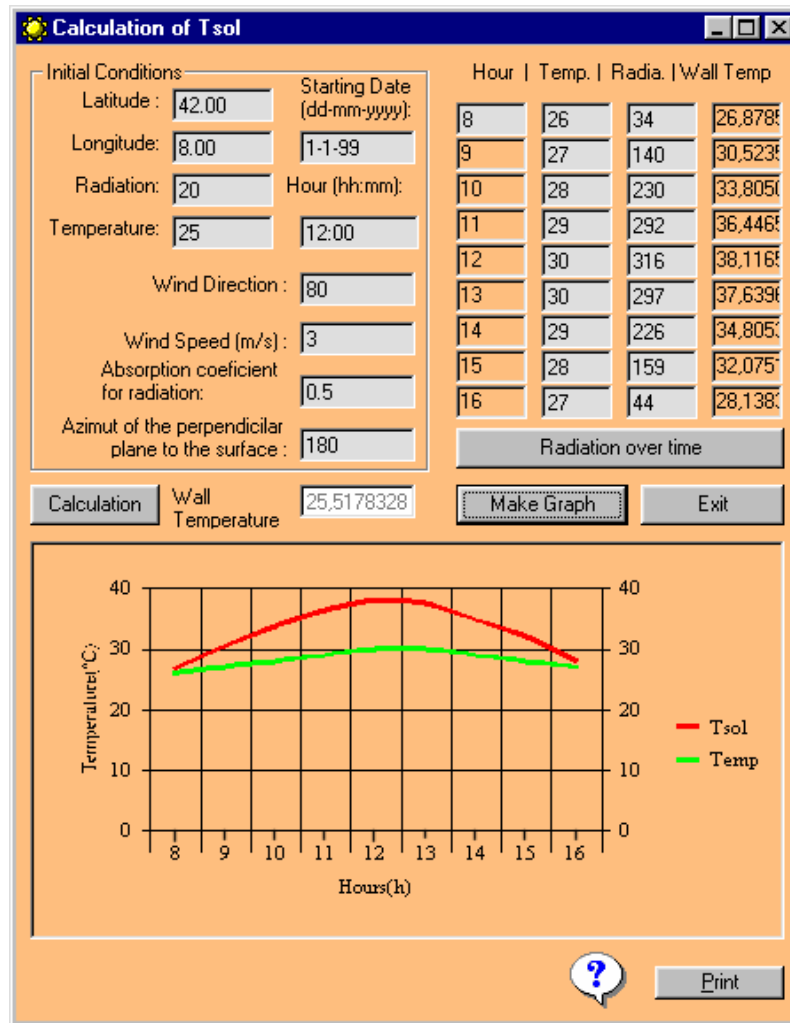


Fig. 40 – Janela com gráfico da variação ao longo do tempo

Os dados necessários ao cálculo deverão ser fornecidos pelo utilizador, tendo contudo o programa, no arranque e por defeito, valores normais para uma situação de inverno.

Fig. 39) permite fazer o cálculo isolado para uma hora determinada. Permite também, ao repetir o cálculo hora a hora, ao longo de um período do dia (definido por defeito no arranque entre as 8 e 16 horas), perceber a variação do efeito da radiação solar sobre superfícies verticais. As horas indicadas são horas solares.

Para o cálculo deverão ser fornecidos os seguintes dados;

- Latitude e longitude do lugar
- Data e hora do teste
- Temperatura exterior, para o momento de cálculo, ou as diversas temperaturas às diversas horas, para a situação de intervalo de tempo.
- Radiação solar horizontal para o momento de cálculo, ou para as diversas horas.
- Coeficiente de absorção do elemento

Quando é feita o cálculo ao longo das diversas horas do dia, para além da saída analítica dos resultados, é possível também visualizar num gráfico as curvas da temperatura do ar exterior e a temperatura superficial da parede [Fig. 40].

4.8 Módulo 7 – Conversão de unidades.

Este módulo surge como um utilitário de apoio aos outros módulos e ao programa geral, tanto no que se refere à introdução de dados como à sua interpretação. Permite fazer a conversão de algumas unidades, entre sistemas usualmente utilizados, para uma melhor e mais eficaz comparação de características ou de resultados.

- Conversão de unidades de pressão
- Conversão de unidades de energia
- Conversão de temperaturas
- Conversão de coeficientes de condutibilidade térmica.
- Conversão de unidades de comprimento (sistema internacional e britânico)

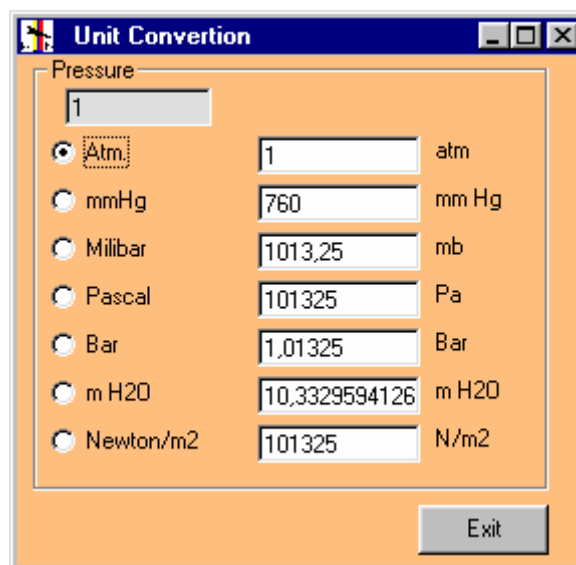


Fig. 41 – Janela de conversão de unidades

Deu ainda origem a um pequeno utilitário, que pode ser instalado num computador, independente dos outros módulos, funciona como uma calculadora, que pode ser acedida através de um ícone na área de trabalho(*desktop*).

4.9 Potencialidades e limitações do Programa

Uma simulação é uma parametrização do real, mas dada a quantidade de elementos em jogo, uma simulação é necessariamente imprecisa, com alguma variação relativamente ao fenómeno real que lhe serviu de base.

As simulações de transferências de calor e humidade em meios porosos não são excepção, não descrevem exactamente e fielmente o fenómeno real, por um lado pelas simplificações do modelo teórico necessárias ao seu cálculo em tempo e com meios acessíveis, e por outro, pela dificuldade de caracterizar univocamente todos os parâmetros que definem uma situação real. Servem contudo para identificar comportamentos, e para analisar comparativamente diversas situações.

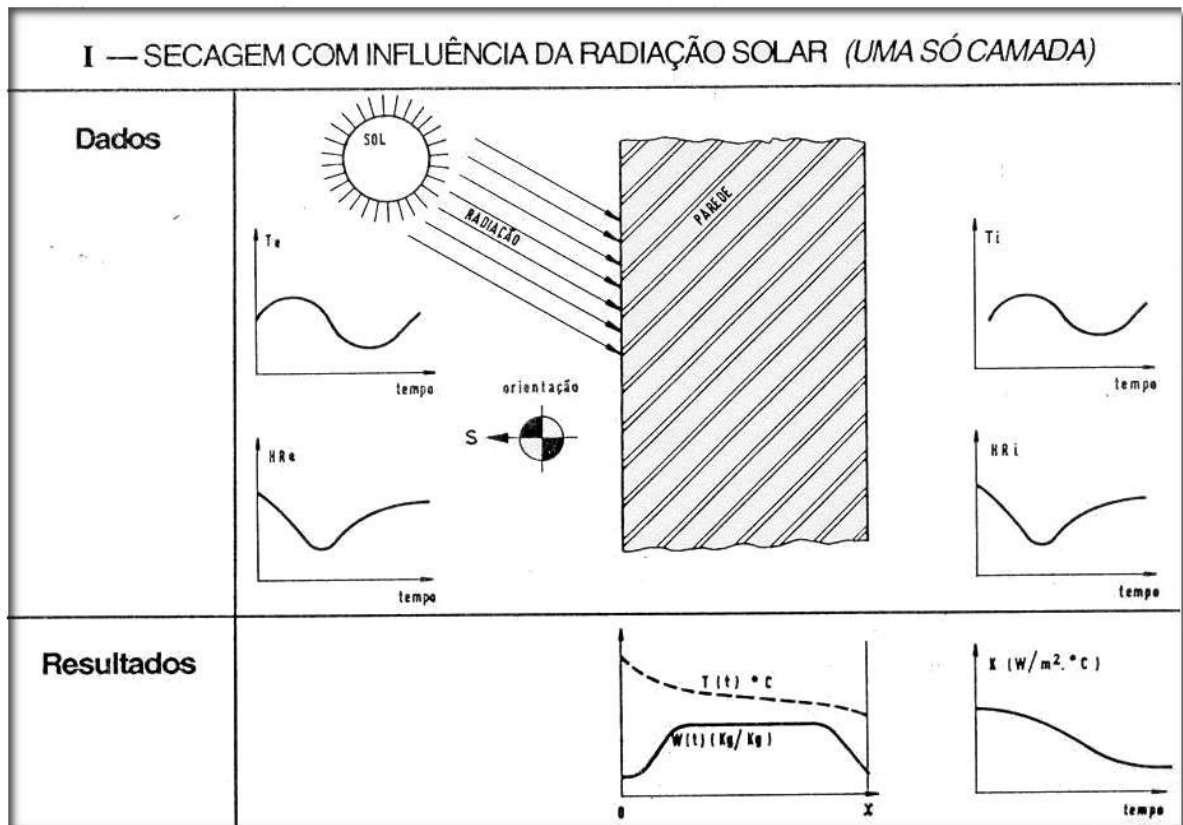


Fig. 42 – Possibilidade de aplicação do programa TrHum 98 - Secagem [1]

Sabemos que os perfis hídricos determinados por simulação não são rigorosamente coincidentes, em valor, com os obtidos experimentalmente, mas existe uma boa concordância na forma da distribuição dos mesmos.

O programa *TrHum 98* é um instrumento importante para a análise de situações existentes ou para auxílio na concepção de novas situações, pois permite, sem grandes esforço ou necessidade de tempo ter a percepção do comportamento do elemento quando sujeito a determinadas condições. Acresce-se o facto de ser um programa de fácil utilização, e sem requisitos especiais para o sistema informático, pelo que poderá vir a ser amplamente utilizado, não só num ambiente de investigação, mas também ligado à actividade da construção civil.

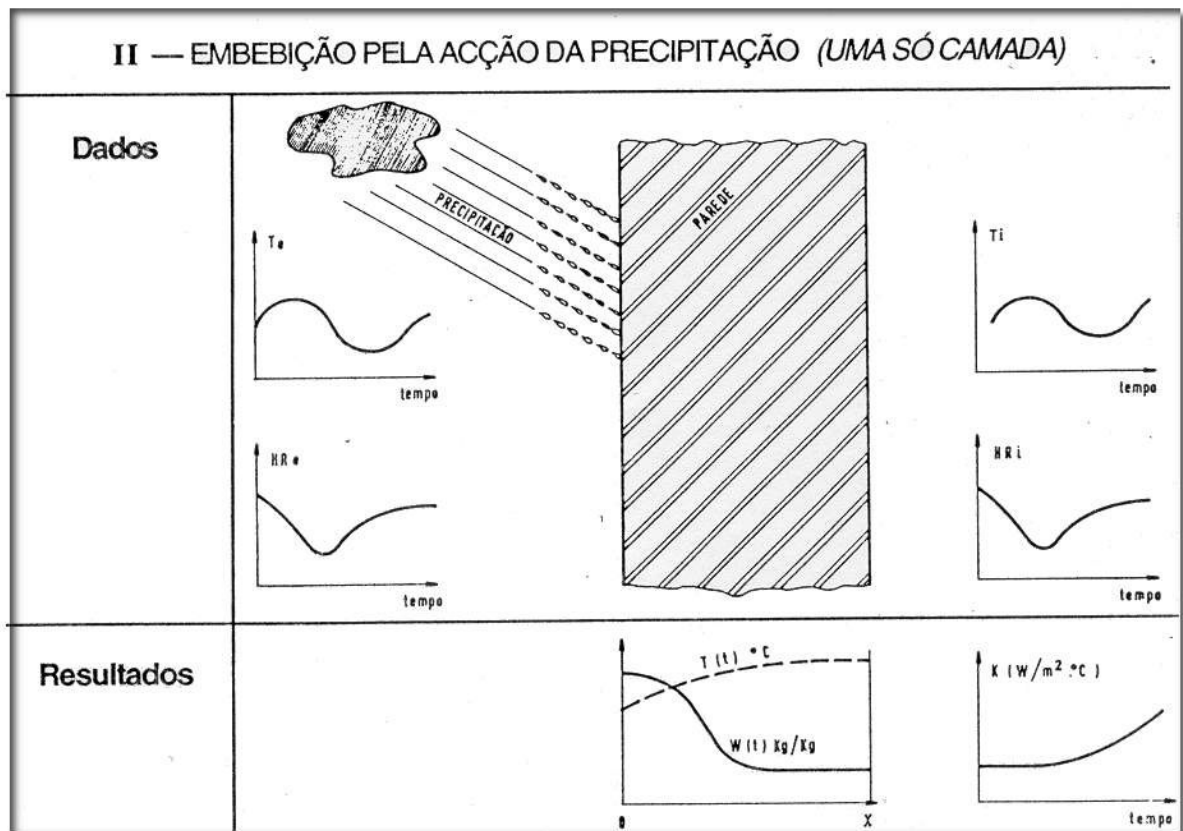


Fig. 43 - Possibilidade de aplicação do programa *TrHum 98* - Embebição [1]

Como limitação principal, para além das atrás referidas, o facto de só permitir a simulação com elementos macroscopicamente homogéneos, isto é, constituídos por uma única camada. O programa permite um conjunto vasto de simulações com interesse na construção, designadamente estudos de embebição de elementos, por chuva incidente ou por condensações internas, estudo de secagem por efeito da radiação, da temperatura, ou da humidade relativa, estudos de ciclos embebição e secagem, determinação dos tempos

necessários para determinado teor de humidade ser atingido, ou mesmo, a determinação da variação da resistência térmica dos elementos em função da sua humedificação.

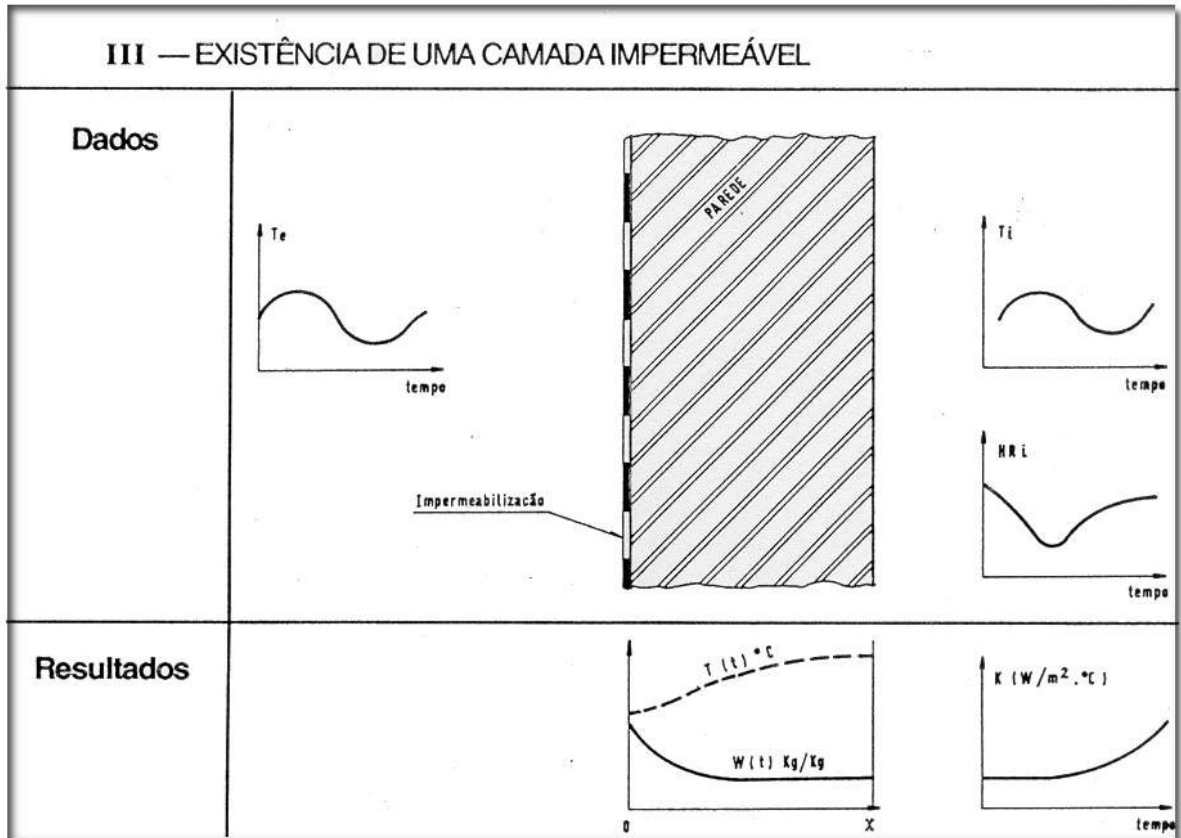


Fig. 44 - Possibilidade de aplicação do programa TrHum 98 – na análise do fenómeno das condensações internas [1]

5 SIMULAÇÃO NUMÉRICA 72

5.1	ANÁLISE DE SENSIBILIDADE.....	72
5.1.1	<i>Parâmetros fixos</i>	72
5.1.2	<i>Análise do passo da simulação</i>	73
5.1.3	<i>Análise das Condições Iniciais</i>	79
5.1.3.1	Temperatura	79
5.1.3.2	Humidade	81
5.1.4	<i>Análise das Condicionantes climáticas</i>	88
5.1.4.1	Temperaturas interior e exterior	88
5.1.4.2	Humidade relativa interior e exterior.....	89
5.1.4.3	Radiação	93
5.1.4.4	Chuva	93
5.1.5	<i>Simulação de situações de referência</i>	95

5 Simulação Numérica

5.1 *Análise de sensibilidade*

O estabelecimento mais efectivo das condições de simulação requer o conhecimento de quais os parâmetros de caracterização do modelo que são verdadeiramente determinantes nos resultados. Para tal foram realizadas algumas simulações com vista à determinação do grau de influência de alguns dos parâmetros nos resultados finais.

Foram executadas análises de sensibilidade aos seguintes factores:

- Intervalo de tempo da simulação
- Condições iniciais
- Condicionantes climáticas

Os restantes parâmetros foram mantidos fixos para que a análise fosse mais expedita.

5.1.1 *Parâmetros fixos*

- Localização

O programa permite, desde que lhe sejam fornecidos os dados climáticos necessários, fazer a simulação de situações localizadas em qualquer ponto do globo. No presente conjunto de simulações foram unicamente utilizadas as coordenadas geográficas do Porto.

- Orientação

As orientações dos elementos em estudos correspondem ao ângulo que a perpendicular à superfície em análise faz com o Norte. O presente programa destina-se à simulação numérica do comportamento em embebição e secagem de elementos verticais, dentro desta restrição todas as orientações são possíveis. Nos testes de sensibilidade, fizeram-se simulações para elementos orientados a sul.

- Materiais

Os materiais disponíveis na base de dados do programa, são os seguintes:

- Betão celular
- Barro Vermelho

Estes materiais correspondem a dois materiais normalmente utilizados na construção civil corrente, como elementos de suporte, como elementos de fronteira e como bases para

receber revestimentos. Nos testes de sensibilidade foi principalmente utilizado o betão celular.

- Parâmetros de caracterização do clima

Para simulação de situações de aproximação à realidade, os dados climáticos devem aproximar-se o mais possível de situações já observadas ou passíveis de serem observadas, pelo que os dados climáticos para esse tipo de estudos deverão ser baseados em condições reais. Poderão ser utilizadas leituras passadas, ou podem ser criados novos ficheiros baseados nos estudos de probabilidade de ocorrência.

O programa possuiu uma base de dados com alguns ficheiros de definição de clima que foram utilizados nas simulações. Correspondem essencialmente à definição de climas constantes (utilizados nos testes de sensibilidade) ou de climas variáveis com a duração de um dia, para a sua repetição consecutiva até atingir o tempo total da simulação [ver Anexo A]. Poderão ser sempre adicionados ou gerados mais.

Os parâmetros de caracterização do clima utilizados nas simulações foram definidos em função do objectivo das mesmas. Assim, para análises de sensibilidade de passo e de condicionantes iniciais foram considerados climas estáticos, isto é, climas em que todos os parâmetros de caracterização se mantêm constantes ao longo de todo o tempo de simulação.

Para efectuar comparações entre diferentes simulações é necessário a escolha de um ponto determinado, por razões operativas. O ponto normalmente escolhido para comparação situa-se a 3 cm do limite exterior do material, pois, por um lado, não é tão fortemente condicionado pelos parâmetros de caracterização exterior, como um ponto da superfície, e por outro lado não revela a inércia de um ponto localizado na zona central do material. Na comparação são utilizados os valores, ao longo do tempo e nas diferentes simulações, naquele ponto específico.

5.1.2 Análise do passo da simulação

O intervalo de tempo escolhido na simulação tem influência directa sobre o tempo que o computador efectivamente leva a fazer o cálculo. Por necessidades operativas relativamente ao tempo despendido na simulação seria desejável que o intervalo de tempo fosse relativamente grande. Em contrapartida, o rigor e a necessidade de aproximação ao real, determinam que esse passo seja o menor possível.

Fizemos um estudo comparativo entre diversos passos de simulação, para avaliarmos, numa relação de custo / benefício, as implicações que os mesmos terão sobre o resultado final da simulação, e sobre a operacionalidade do programa.

Quadro 3 – Definição dos parâmetros de caracterização do teste da variação do intervalo de tempo

Espessura do material	20 cm	Latitude	42 Norte
Espaçamento da malha	1 cm	Longitude	8 Este
$h_e; h_i$	25 W/m ² .°C	Azimute	180 °
$\beta_e; \beta_i$	0.003 m/s	Direcção do Vento	80 °
Erro máximo:	0.001	Velocidade do vento	3 m/s
Material	Betão Celular	Temperatura exterior	25 °C
Tempo de simulação	120 horas (5 dias)	Humidade Relativa Exterior	1 %
Temperatura inicial	25 °C (constante)	Radiação	0 W/m ²
Teor de humidade inicial	0.5 kg/kg (constante)	Chuva	0
Temperatura Interior	25 °C	Humidade relativa interior	99%

Passos de simulação	2 h	1 h	0,5 h	0,2 h	0.1 h	0.015 h	0.001 h
---------------------	-----	-----	-------	-------	-------	---------	---------

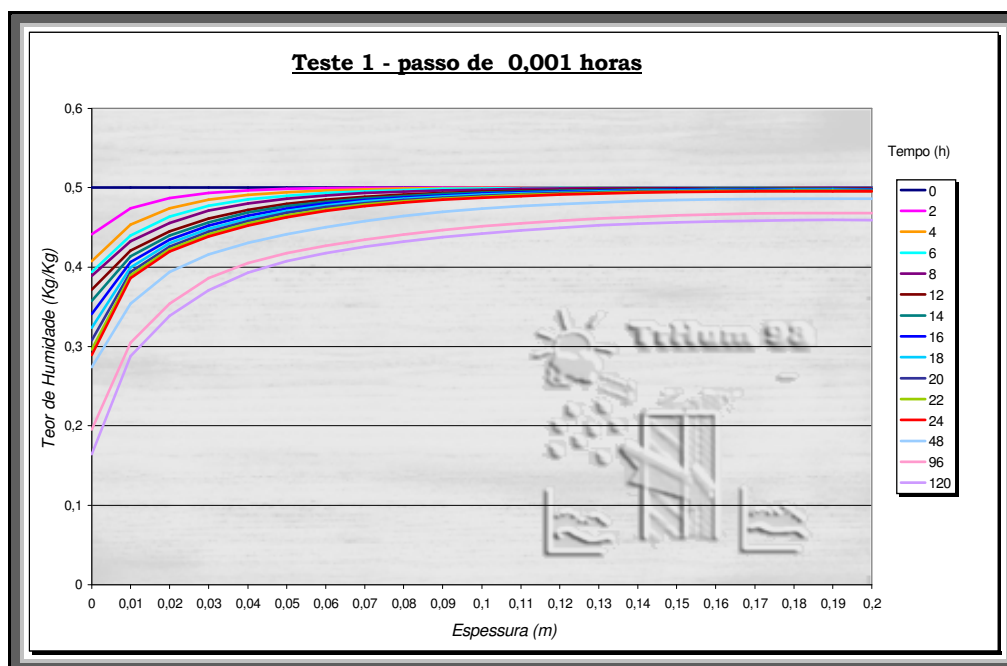


Fig. 45 – Evolução dos perfis do teor de humidade (passo de simulação 0.001 horas)

Efectuaram-se várias séries de simulações idênticas fazendo variar unicamente o intervalo de tempo entre 0,001 hora (3,6 segundos) [Fig. 45] e 2 horas, para tentar determinar a diferença nos resultados finais, bem como a variação do tempo de cálculo.

A simulação foi efectuada partindo de um teor de humidade e temperatura constantes e colocando o material em diferentes condições interiores e exteriores, de modo a poder avaliar a relação entre o passo do cálculo e a velocidade de estabilização do material.

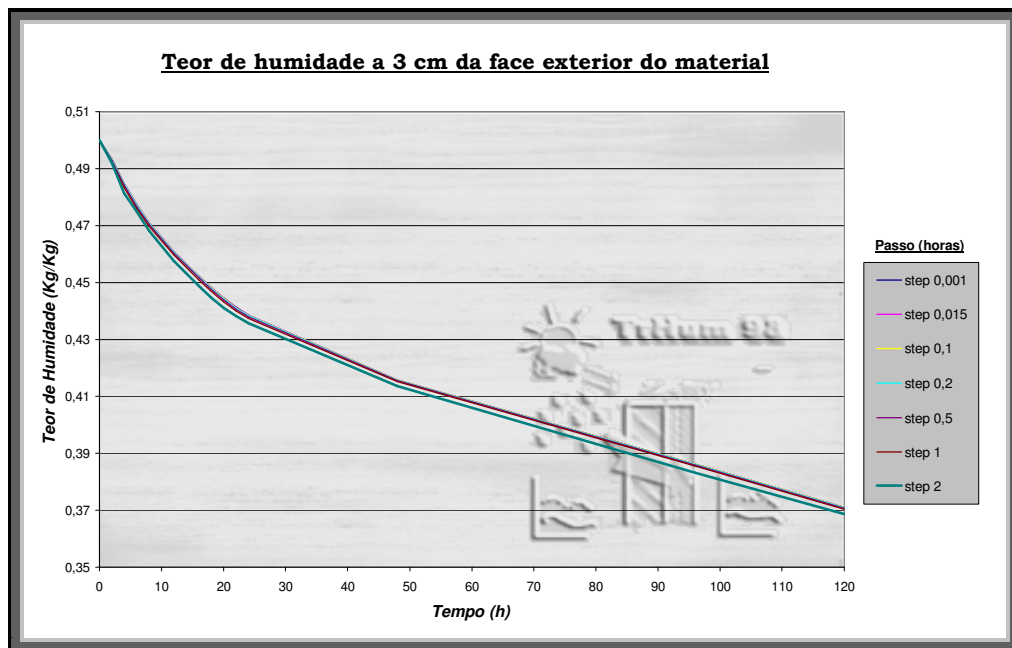


Fig. 46 – Análise comparativa do teor de humidade do material em função do passo da simulação (perfil efectuado a 3 cm do exterior do material)

Como podemos observar na Fig. 46, a alteração do intervalo de tempo da simulação tem influência na evolução do teor de humidade. Essa variação faz-se sentir predominantemente para passos de simulação já elevados.

Comparamos os resultados das diferentes simulações ao fim de 120 horas, em quatro pontos distintos do material, na superfície exterior, a 3cm da superfície exterior, na superfície interior e no ponto médio do mesmo. Como podemos verificar no Quadro 4, o desvio dos resultados em relação à simulação de passo menor varia consideravelmente. Esse desvio é tanto maior quanto maior é o desequilíbrio do material com a envolvente.

Quadro 4 – Comparação entre resultados e respectivos desvios para diferentes passos de simulação.

	Ponto exterior do material		Ponto a 3 cm do exterior		Ponto médio do material		Ponto interior do material	
	Valor Absoluto	Desvio Percentual	Valor Absoluto	Desvio Percentual	Valor Absoluto	Desvio Percentual	Valor Absoluto	Desvio Percentual
Passo 0,001 h	0,164334	0,0000%	0,371079	0,0000%	0,4423384	0,0000%	0,4588237	0,0000%
Passo 0,015 h	0,1643147	0,0118%	0,371069	0,0026%	0,4423328	0,0013%	0,4588185	0,0011%
Passo 0,1 h	0,1641975	0,0831%	0,371009	0,0187%	0,4422987	0,0090%	0,4587868	0,0081%
Passo 0,2 h	0,1640613	0,1660%	0,37094	0,0373%	0,4422592	0,0179%	0,45875	0,0161%
Passo 0,5 h	0,1636688	0,4048%	0,370742	0,0906%	0,4421459	0,0435%	0,4586451	0,0389%
Passo 1 h	0,1629395	0,8486%	0,370368	0,1915%	0,4419385	0,0904%	0,4584553	0,0803%
Passo 2 h	0,1597909	2,7646%	0,368654	0,6534%	0,4412011	0,2571%	0,4578781	0,2061%

Fizeram-se simulações mais dilatadas no tempo para avaliar a forma como a diferença do passo pode influenciar o resultado final [Fig. 47]. Pudemos constatar que existe uma tendência para a convergência entre os diversos passos, correspondendo essa convergência à fase em que o material atinge o equilíbrio.

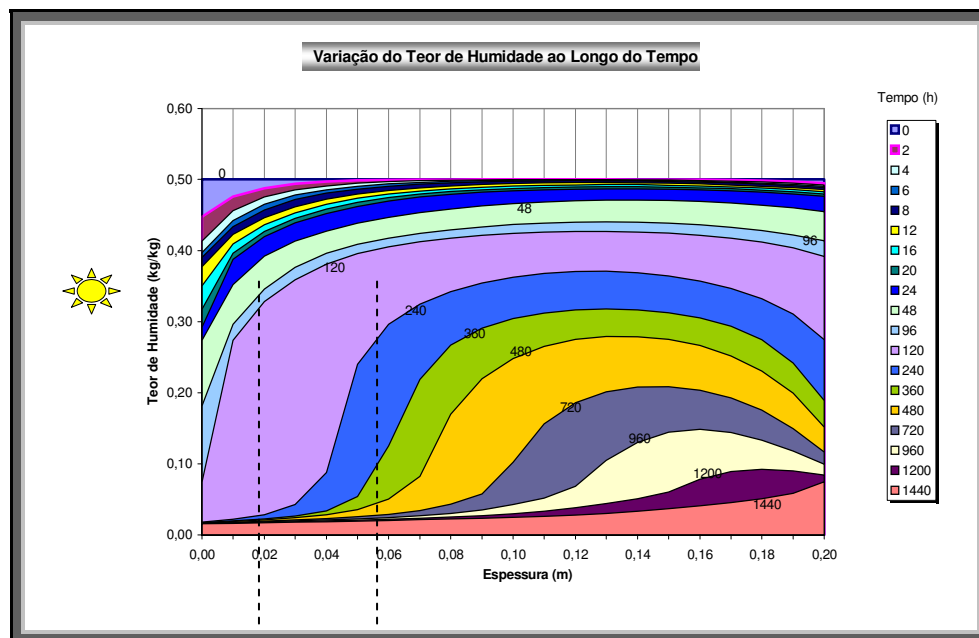


Fig. 47 – Evolução dos perfis do teor de umidade para uma simulação de 1440 horas (passo de simulação 0.05 horas)

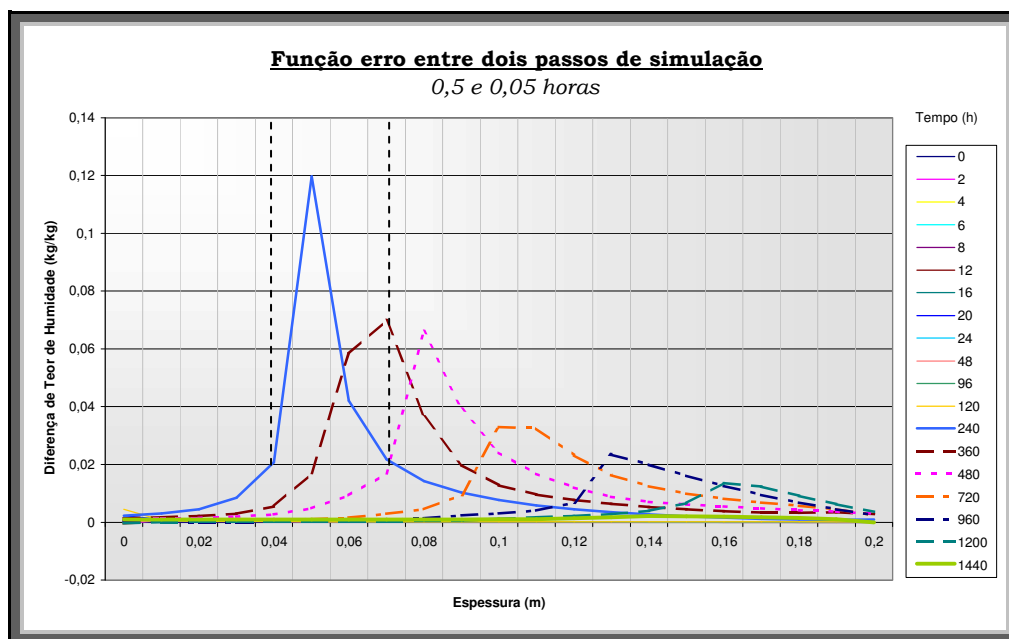


Fig. 48 – Localização do erro entre simulações de diferente passo (0,5 e 0,05 horas)

O gráfico da Fig. 48 corresponde ao valor da diferença entre os valores do teor de humidade de simulações de dois diferentes passos, 0,5 e 0,05 horas. O erro é considerável em zonas específicas que correspondem aos pontos de inflexão das curvas do teor de humidade, tal como se pode verificar na comparação da Fig. 47 com a Fig. 48, sendo diminuto nos restantes pontos.

Quadro 5 – Comparação entre o passo da simulação e o tempo de simulação

Passo da simulação	N.total de passos necessários	Tempo	Tempo/N passos*1000
0,001 h (3,6 segundos)	$120/0,001=120.000$	03:12:43	0:01:36
0,015 h (apr.1 minuto)	$120/0,015=8.000$	00:12:55	0:01:37
0,1 h (6 minutos)	$120/0,1=1.200$	00:02:02	0:01:42
0,2 h (12 minutos)	$120/0,2=600$	00:01:00	0:01:40
0,5 h (30 minutos)	$120/0,5=240$	00:00:23	0:01:36
1 h	$120/1=120$	00:00:12	0:01:40
2 h	$120/2=60$	00:00:08	0:02:13

A escolha do intervalo de tempo é importante também ao nível da velocidade de cálculo. Foram feitas algumas simulações para o mesmo intervalo de tempo, (5 dias – 120 horas), para passos entre 3,6 segundos e 2 horas, sendo os tempos gastos no cálculo os apresentados no Quadro 5.

Como se pode verificar, o tempo de cálculo de uma simulação é directamente proporcional ao número de passos da mesma. A diferença observada para a simulação de passo de 2 horas é explicável pelo facto da medição dos tempos se ter feito unicamente com precisão de segundos, e a duração ser extremamente curta, verificando-se um fenómeno de acumulação de erro.

A relação entre o passo temporal considerado e tempo de cálculo não é sempre constante, mas sim dependente de todos os parâmetros necessários à simulação. A convergência de resultados entre iterações sucessivas é também factor determinante para o tempo de cálculo, pelo que não é possível prever, à partida e com rigor, qual o tempo necessário para fazer uma determinada simulação.

É todavia possível, após uma primeira simulação executada com passo elevado, determinar com uma boa aproximação o tempo necessário para fazer a mesma simulação com um passo mais reduzido.

Sendo uma simulação essencialmente constituída por operações matemáticas, o tipo de processador do computador influencia também a velocidade da simulação. Os resultados apresentados no Quadro 5 correspondem a testes efectuados num computador *Pentium* a 200 MHz, com 96 Mb de *RAM*. Foram realizados novos testes num computador *Pentium II* a 350 MHz, com 128 Mb de *RAM*, sendo os tempos significativamente menores, até cerca de ¼ dos inicialmente verificados.

Quadro 6 – Comparação entre diferentes processadores

Passo da simulação	Pentium 200		Pentium II 350	
	Tempo	Tempo/N passos*1000	Tempo	Tempo/N passos*1000
36 segundos (0,001 h)	03:12:43	0:01:36	1:20:02	0:00:40
1 minuto (0,015 h)	00:12:55	0:01:37	0:05:28	0:00:41
6 minutos (0,1 h)	00:02:02	0:01:42	0:00:50	0:00:42
12 minutos (0,2 h)	00:01:00	0:01:40	0:00:23	0:00:38
30 minutos (0,5 h)	00:00:23	0:01:36	0:00:09	0:00:37
1 hora	00:00:12	0:01:40	0:00:04	0:00:33
2 horas	00:00:08	0:02:13	0:00:02	0:00:33

Isto permite-nos antever, que com a evolução dos sistemas informáticos, o programa se irá tornar uma ferramenta cada vez mais expedita para a simulação do comportamento em embebição e secagem de materiais de construção.

Por tudo o apresentado é aceitável que em simulações de longa duração, por questões de economia de tempo, se estabeleçam intervalos de tempo variáveis, menores no início, ou quando existe variação das condições climáticas, e intervalos maiores, quando o material está mais próximo do equilíbrio, ou quando as condições climáticas são constantes. Não será conveniente usar intervalos superiores a 0,5 horas, pois a economia de tempo não compensa o aumento de erro.

5.1.3 Análise das Condições Iniciais

Serão as condições iniciais de um material, isto é, a distribuição inicial dos teores de humidade e de temperatura, determinantes ou não nos resultados finais da simulação, bem como no comportamento ao longo do tempo?

Foram feitas várias simulações, variando unicamente a distribuição inicial dos teores de humidade e da temperatura, e colocando o material sob as mesmas restantes condições de simulação, para determinar o efeito dessa variação sobre os resultados finais.

5.1.3.1 Temperatura

A análise das temperaturas realizou-se entre os 10°C e os 35°C, com intervalo de 5°C. Da observação dos resultados constata-se que a distribuição inicial de temperaturas no material deixa de ter qualquer efeito sobre o resultado da simulação, mesmo para períodos relativamente curtos. Num tempo de simulação de sensivelmente 24 horas, independentemente das temperaturas iniciais, o material fica com uma variação de temperaturas que reflecte unicamente a diferença de temperaturas entre interior e exterior [Fig. 49, Fig. 50].

Quadro 7 – Definição dos parâmetros de caracterização do teste da distribuição inicial de temperaturas

Espessura do material	20 cm	Latitude	42 Norte
Espaçamento da malha	1 cm	Longitude	8 Este
$h_e; h_i$	25 W/m ² .°C	Azimute	180 °
$\beta_e; \beta_i$	0.003 m/s	Direcção do Vento	80 °
Erro máximo:	0.001	Velocidade do vento	3 m/s
Material	Betão Celular	Temperatura exterior	25 °C
Tempo de simulação	48 horas (2 dias)	Humidade Relativa Exterior	50 %
Passo da simulação	0.05 h	Radiação	0 W/m ²
Teor de humidade inicial	0.5 kg/kg (constante)	Chuva	0 (não)
Temperatura Interior	25 °C	Humidade relativa interior	50%

Temperatura Inicial (°C)	10	15	20	15	30	35
--------------------------	----	----	----	----	----	----

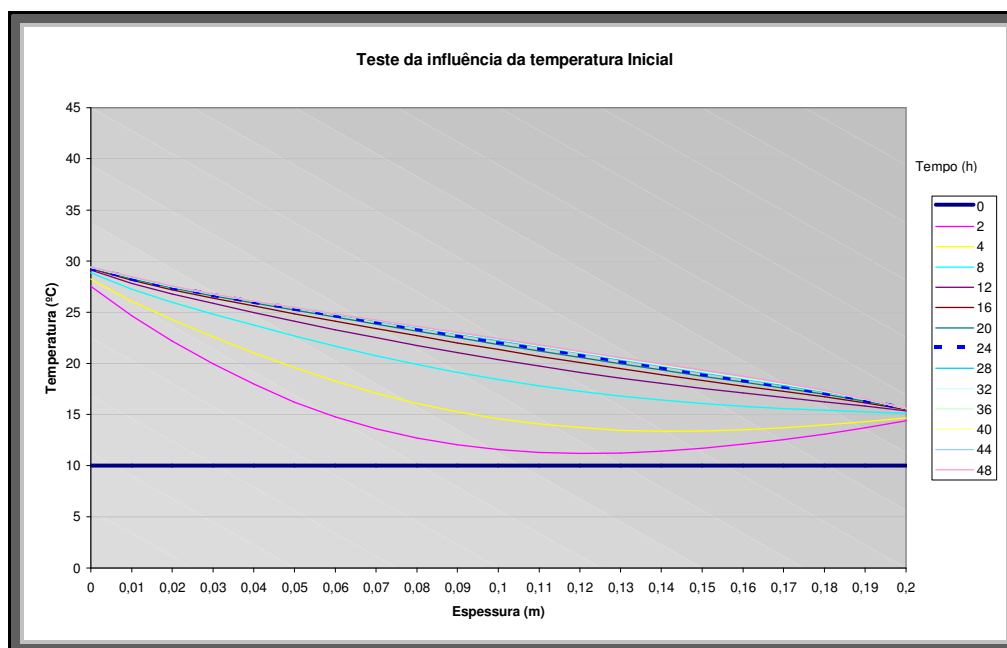


Fig. 49 - Variação da temperatura do betão celular ao longo de 48 horas - temperatura inicial 10°C

De referir que os materiais em análise não são materiais de isolamento térmico. No caso de se tratar de materiais com resistência térmica elevada, será de esperar que o tempo de estabilização térmico seja superior.

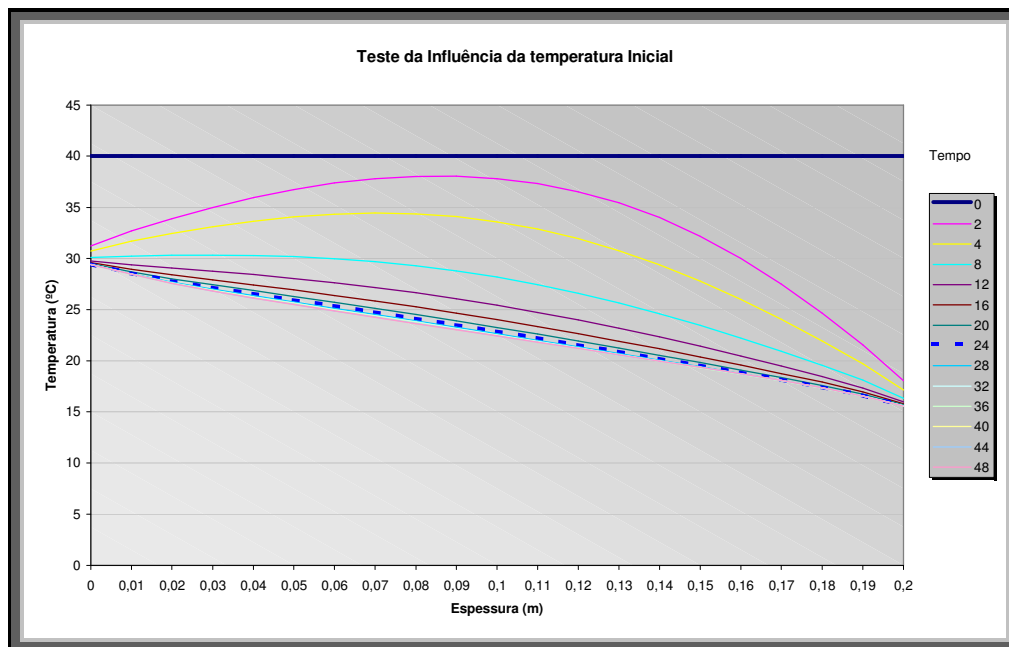


Fig. 50 - Variação da temperatura do betão celular ao longo de 48 horas - temperatura inicial 40°C

5.1.3.2 Humidade

A análise dos teores de humidade realizou-se entre 0.05 e 0.60 kg/kg para o betão celular, e 0.01 e 0.13 kg/kg para o barro vermelho. Os teores de humidade iniciais do material, como esperávamos, são já determinantes na evolução dos perfis ao longo do tempo, fundamentalmente nos processos de secagem do material [Fig. 52, Fig. 53].

A velocidade de secagem é variável em função do maior ou menor teor de humidade inicial do material. Pretendeu-se determinar se o que influenciava a velocidade de secagem do material era o seu teor de humidade em valor absoluto, ou se seria o valor da diferença (valor absoluto ou percentual) entre o teor de humidade inicial e o teor de humidade de equilíbrio higroscópico para a humidade relativa do ambiente.

Quadro 8 – Definição dos parâmetros de caracterização no teste da distribuição inicial de teores de humidade

Espessura do material	20 cm	Latitude	42 Norte
Espaçamento da malha	1 cm	Longitude	8 Este
he; hi	25 W/m ² .°C	Azimute	180 °
βe; βi	0.003 m/s	Direcção do Vento	80 °
Erro máximo:	0.001	Velocidade do vento	3 m/s
Material	Betão Celular	Temperatura exterior	25 °C
Tempo de simulação	480 horas (20 dias)	Humidade Relativa Exterior	50 %
Passo da simulação	0.05 h	Radiação	0 W/m ²
Temperatura inicial	25 °C (constante)	Chuva	0 (não)
Temperatura Interior	25 °C	Humidade relativa interior	50%

Teor de humidade inicial kg/kg (betão celular)	0,2	0,4	0,6
Teor de humidade inicial kg/kg (barro vermelho)	0.01	0.05	0.13

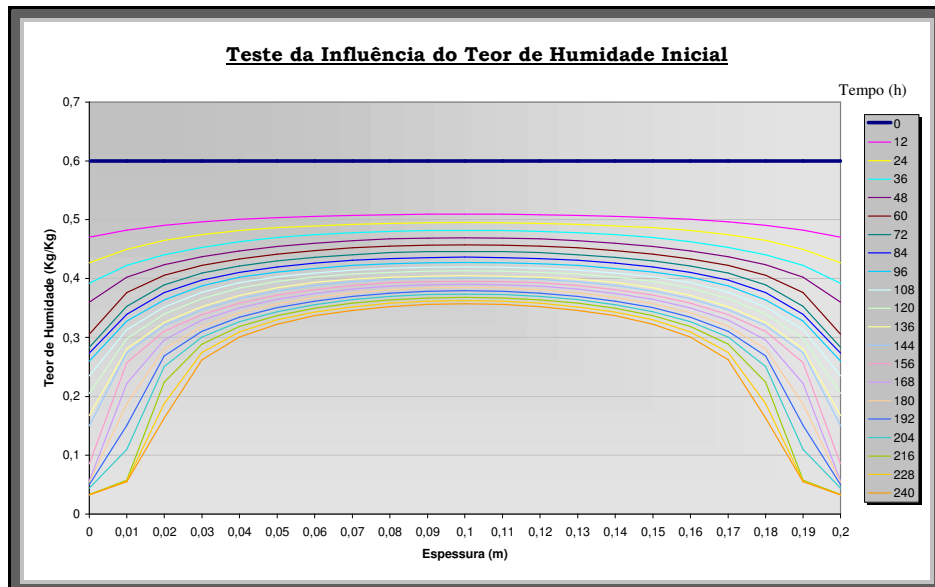


Fig. 51 - Variação do teor de humidade do betão celular ao longo de 240 horas – teor de humidade inicial de 0.6 kg/kg – Humidade relativa da ambiência de 50%

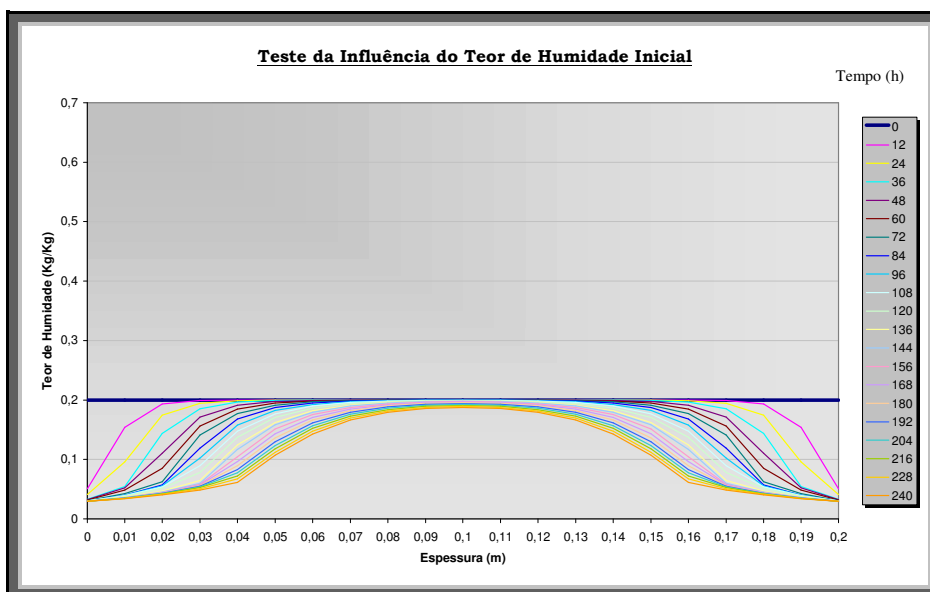


Fig. 52 - Variação do teor de umidade do betão celular ao longo de 240 horas – teor de umidade inicial de 0.2 kg/kg – Umidade relativa da ambiência de 50%

Para tal foram realizadas algumas simulações, em que se variou conjuntamente teor de umidade inicial e umidade relativa da ambiência. Na Fig. 51e Fig. 52 apresentam-se os resultados de duas simulações, partindo de teores de umidade iniciais de 0,6 kg/kg e 0,2 kg/kg, respectivamente. Podemos verificar que o teor de umidade do material, ao fim de 240 horas é fundamentalmente determinado pelo teor de umidade inicial.

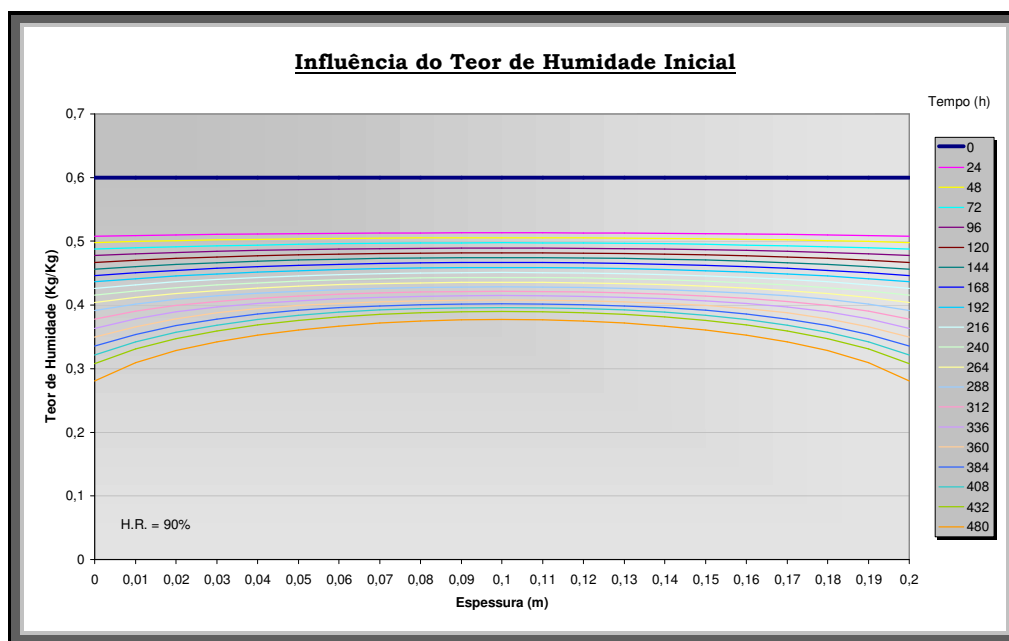


Fig. 53- Variação do teor de umidade ao longo do tempo - 480 horas – teor de umidade inicial de 0.6 kg/kg – umidade relativa 90 %

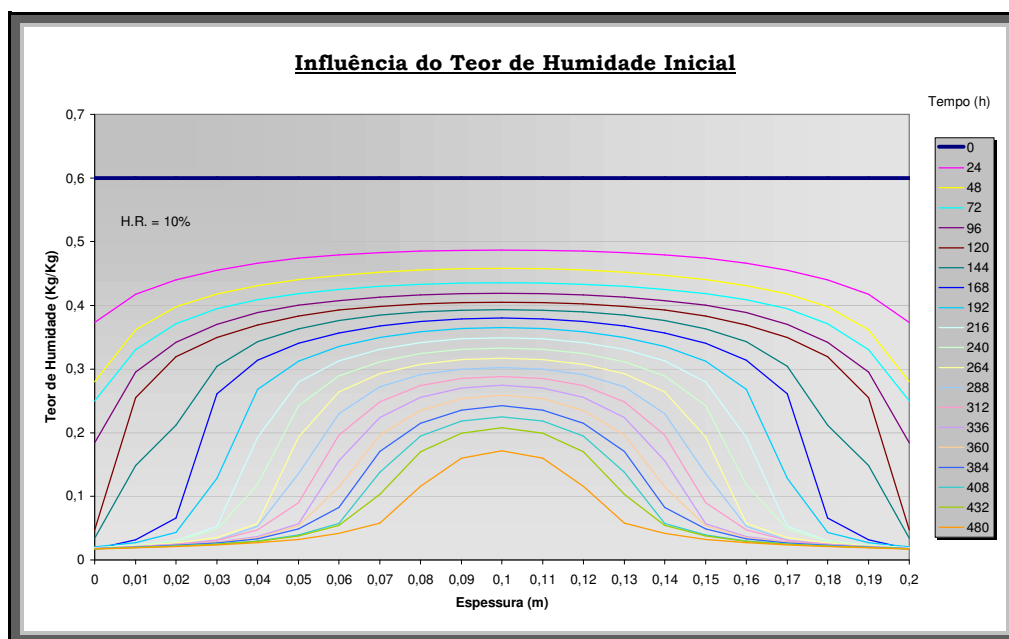


Fig. 54- Variação do teor de humidade ao longo do tempo - 480 horas – teor de humidade inicial de 0.6 kg/kg – humidade relativa 10 %

As Fig. 51, Fig. 53 e Fig. 54 correspondem a simulações partindo de teores de humidade iniciais idênticos (0,6 kg/kg) e colocando o material em ambiências com humidades relativas diferentes (10%, 50% e 90%).

Como se pode constatar na comparação das figuras, a velocidade de secagem é tanto maior quanto menor for a humidade relativa da ambiência.

Quadro 9 –Valores do teor de humidade em função da humidade relativa

Valor da humidade relativa da ambiência	Valor do teor de humidade de equilíbrio higroscópico	Tipo de relação	Valor do teor de humidade inicial
10 %	0.015 kg/kg	Valor absoluto (+ 0.2 kg/kg)	0.215 kg/kg
70%	0.035 kg/kg	Valor absoluto (+ 0.2 kg/kg)	0.235 kg/kg
90%	0.075 kg/kg	Valor absoluto (+ 0.2 kg/kg)	0.275 kg/kg
10 %	0.015 kg/kg	Percentual (+ 50%)	0.0225 kg/kg
70%	0.035 kg/kg	Percentual (+ 50%)	0.0525 kg/kg
90%	0.075 kg/kg	Percentual (+ 50%)	0.1125 kg/kg

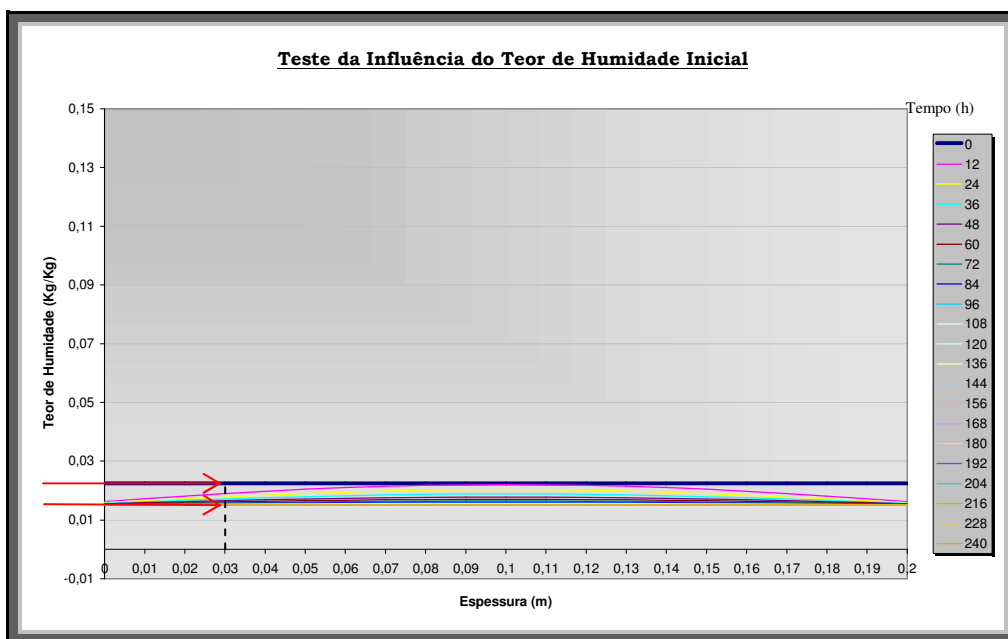


Fig. 55 – Variação do teor de humidade da amostra – Teor inicial 0.0225 kg/kg – Humidade relativa 10%

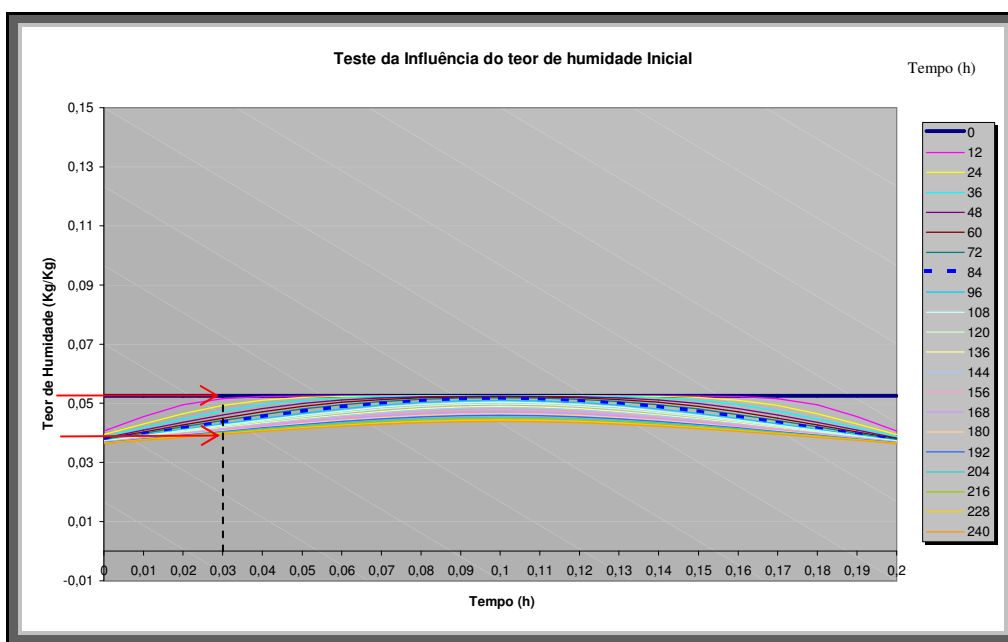


Fig. 56 – Variação do teor de humidade da amostra – Teor inicial 0.0525 kg/kg – Humidade relativa 70%

Fizemos ainda um novo conjunto de simulações [Quadro 9] em que fizemos a relação entre o teor de humidade inicial e o teor de humidade de equilíbrio higroscópico para as diferentes humidades relativas da ambiência. As Fig. 55, Fig. 56 e Fig. 57 correspondem a uma

diferença percentual entre o teor de humidade inicial e o teor de humidade higroscópico, para o betão celular.

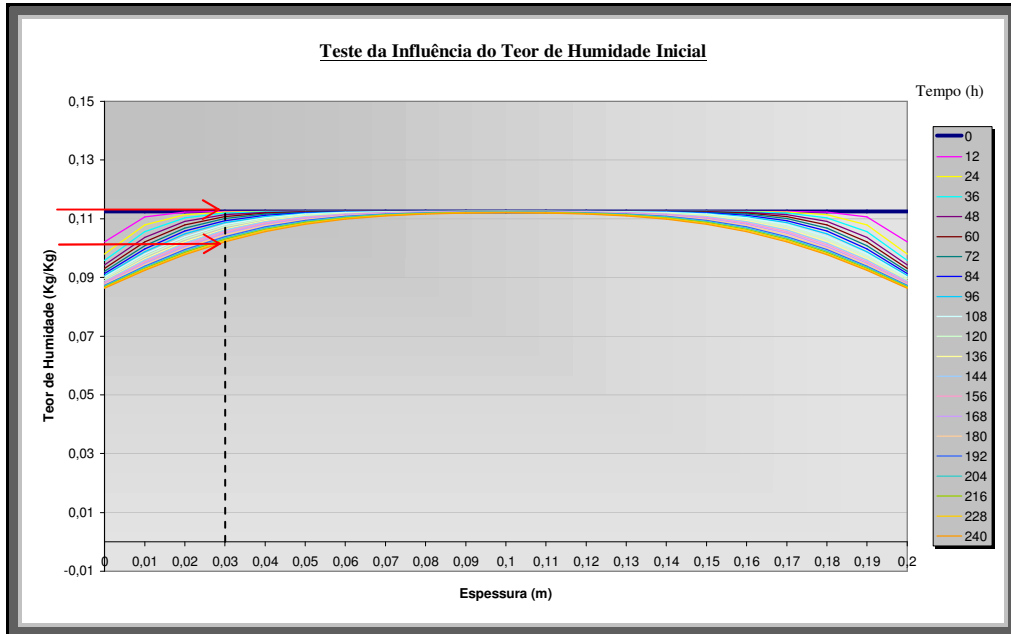


Fig. 57 – Variação do teor de humidade da amostra – Teor inicial 0.1125 kg/kg – Humidade relativa 90%

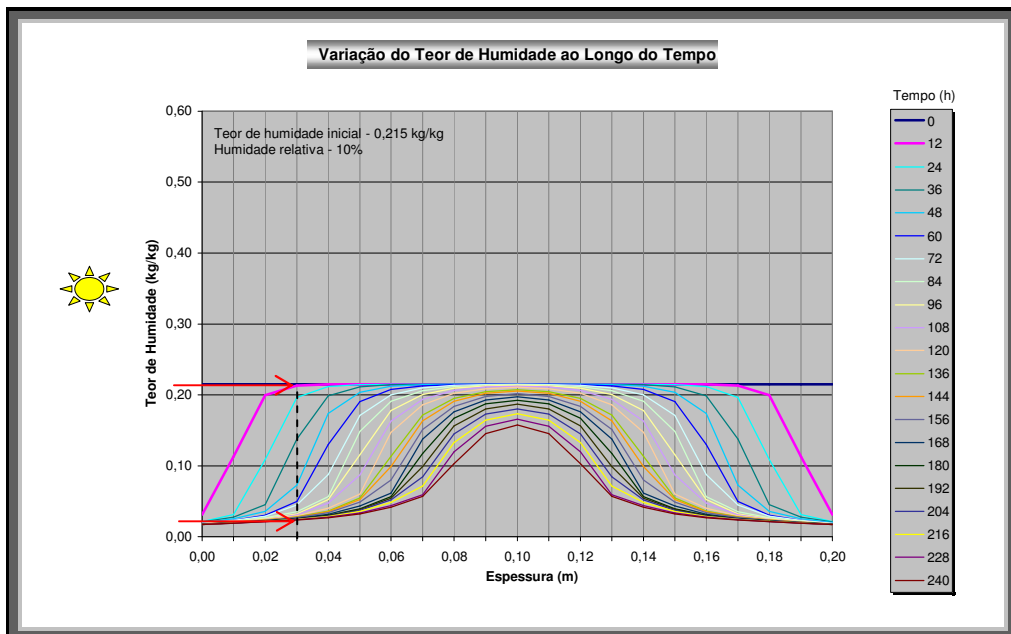


Fig. 58 – Variação do teor de humidade da amostra – Teor inicial 0.215 – Humidade relativa 10%

Nas Fig. 58, Fig. 59, e Fig. 60 apresentamos os gráficos dos teores de humidade para o betão celular, ao longo de 240 horas, para um teor de humidade inicial superior em 0,2 kg/kg ao teor de humidade higroscópico para a humidade relativa da ambiência.

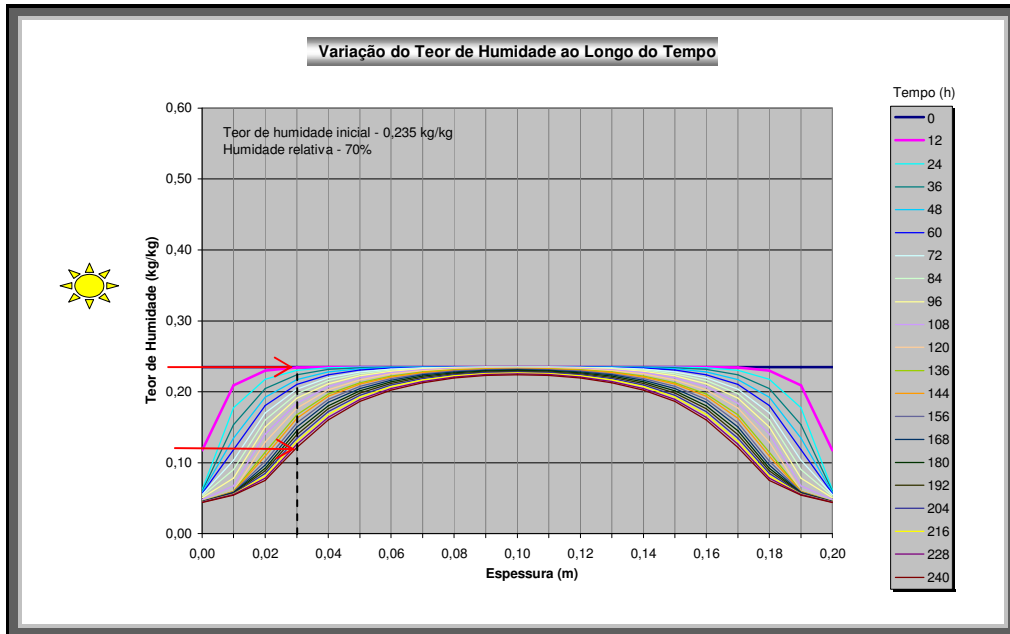


Fig. 59 – Variação do teor de humidade da amostra – Teor inicial 0.235 kg/kg – Humidade relativa 70%

A humidade relativa do ambiente é muito importante na evolução do teor de humidade do material. A secagem do material será feita com maior ou menor velocidade, consoante a diferença entre teor de humidade do material e o teor de humidade higroscópico para a humidade relativa do ambiente.

Tentamos determinar se a velocidade de secagem seria proporcional ao valor absoluto desta diferença, ou seria uma diferença percentual.

Da comparação entre os dois conjuntos de simulações, pudemos constatar que a proporcionalidade exacta não existe, sendo a hipótese de variação percentual a mais aproximada.

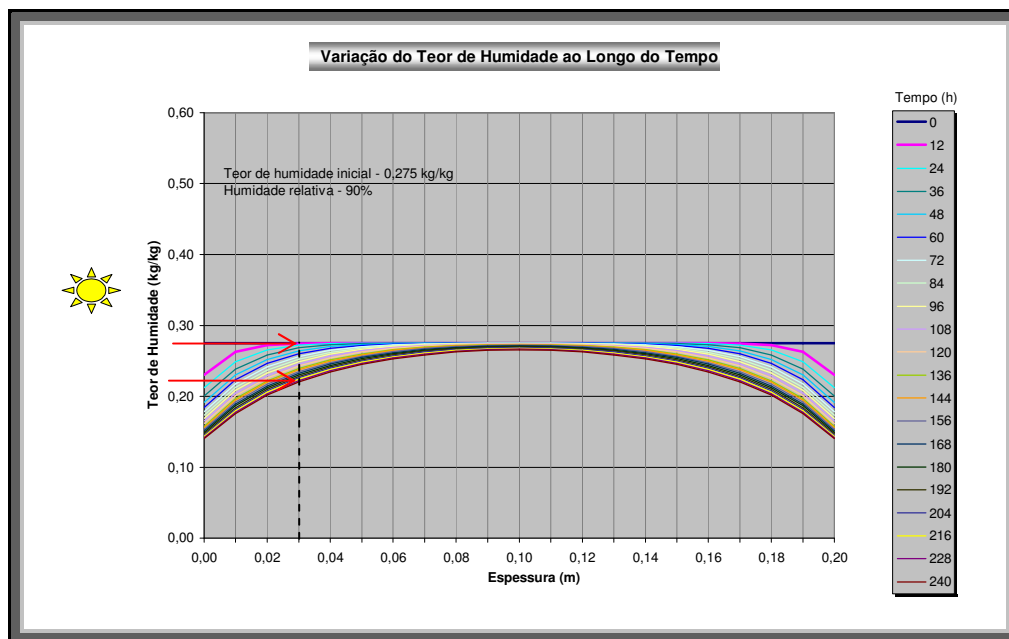


Fig. 60 – Variação do teor de umidade da amostra – Teor inicial 0.275 – Humidade relativa 90%

5.1.4 Análise das Condicionantes climáticas

Fizeram-se também testes da influência da variação dos vários parâmetros de definição do clima, no comportamento do material ao longo do tempo. Esses testes foram feitos primeiramente isolando um dos factores e mantendo todos os restantes constantes para em seguida fazê-lo interagir com um ou mais.

5.1.4.1 Temperaturas interior e exterior

Ao estudar a influência da variação da temperatura devemos ter em conta dois aspectos; em primeiro lugar, devemos considerar a variação da temperatura interior e exterior em si, isto é, a influência do valor absoluto da temperatura no comportamento do material, e segundo lugar devemos ter em conta a influência do gradiente de temperaturas instalado na amostra, em função das diferenças de temperaturas exterior e interior, e o modo como essa variação de temperaturas se desenvolve ao longo da mesma.

Tendo em conta este aspecto realizaram-se varias simulações, cobrindo a quase totalidade de combinações possíveis, com passos de 5°C.

A temperatura, como motor de secagem do material, tem uma influência relativamente pequena , nos valores do teor de humidade ao longo do tempo, como podemos ver pelo gráfico da Fig. 72.

Contudo, essa influência determina um deslocamento da curva de teor de humidade máximo, que deixa de estar localizada no ponto médio da amostra, para passar a localizar-se mais próxima do lado de menor temperatura.

5.1.4.2 Humidade relativa interior e exterior

A humidade relativa interior e exterior foi posta a variar entre 10% e 90%, com iguais valores para interior e exterior, para uma análise mais directa dos resultados.

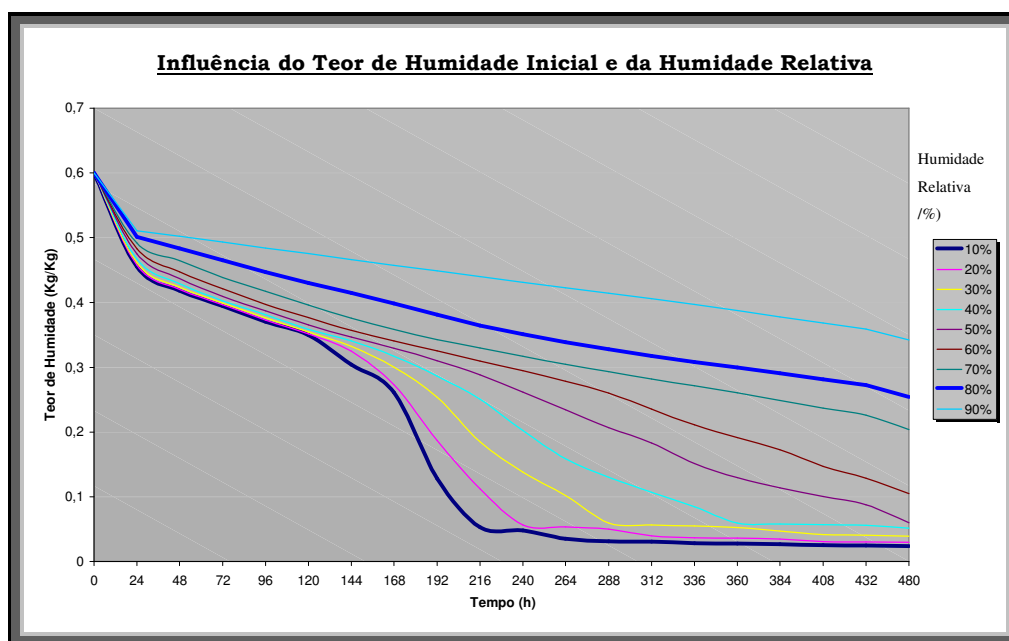


Fig. 61 - Influência da Humidade relativa na secagem do betão – ponto a 3 cm do exterior do material

No gráfico da Fig. 61 mostra o perfil do teor de humidade ao longo do tempo, e nele é facilmente perceptível a influência da humidade relativa da ambiência na velocidade de secagem e no teor de humidade do material. O gráfico representa o teor de humidade de um ponto situado a 3cm da face exterior do material, em diversas simulações em que se fez variar apenas a humidade relativa interior e exterior entre 10% e 90%.

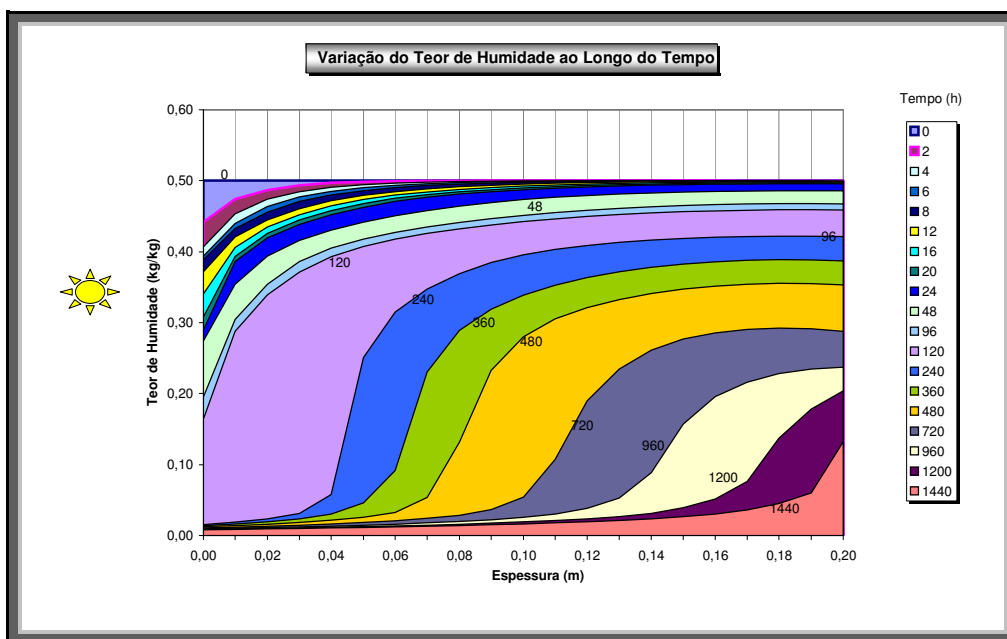


Fig. 62- Secagem diferencial do betão celular em função da humidade relativa (1%-99%)

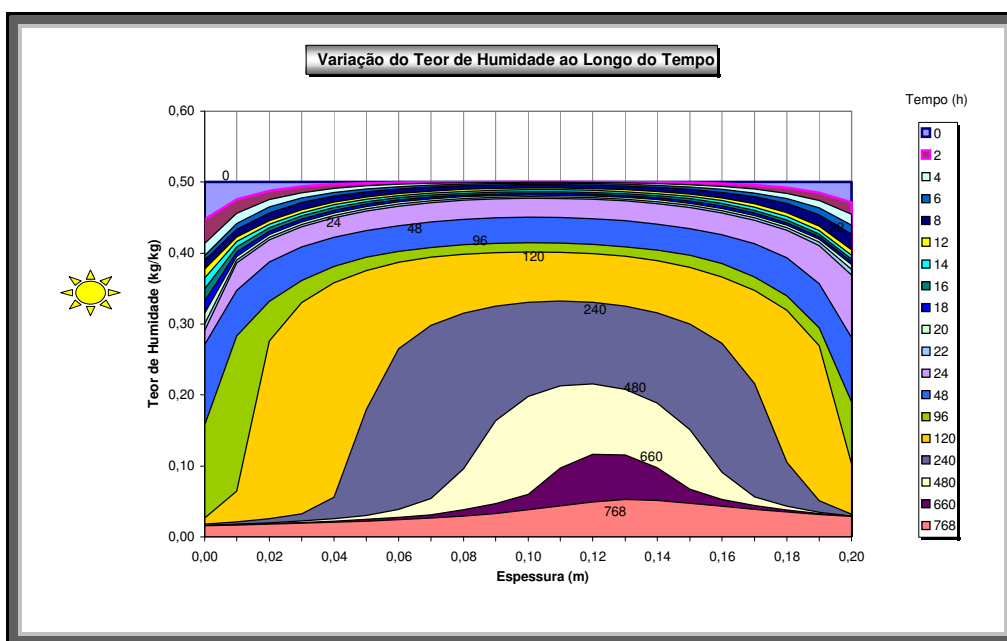


Fig. 63- Secagem diferencial do betão celular em função da humidade relativa (10%-50%)

Fez-se ainda um outro conjunto de simulações, colocando o material sob uma diferença de humidade relativa interior e exterior, para avaliar do modo como essa diferença influenciaria a secagem. A colocação do material numa ambiência com variação de humidade relativa entre as duas faces opostas do mesmo, implica uma distribuição não equilibrada do teor de humidade na espessura do material. Assiste-se assim a um valor mais elevado do teor de

humidade na zona próxima da zona em contacto com a ambiência de humidade relativa mais elevada, tal como se pode observar nas Fig. 62, Fig. 63, Fig. 64, Fig. 65 e Fig. 66.

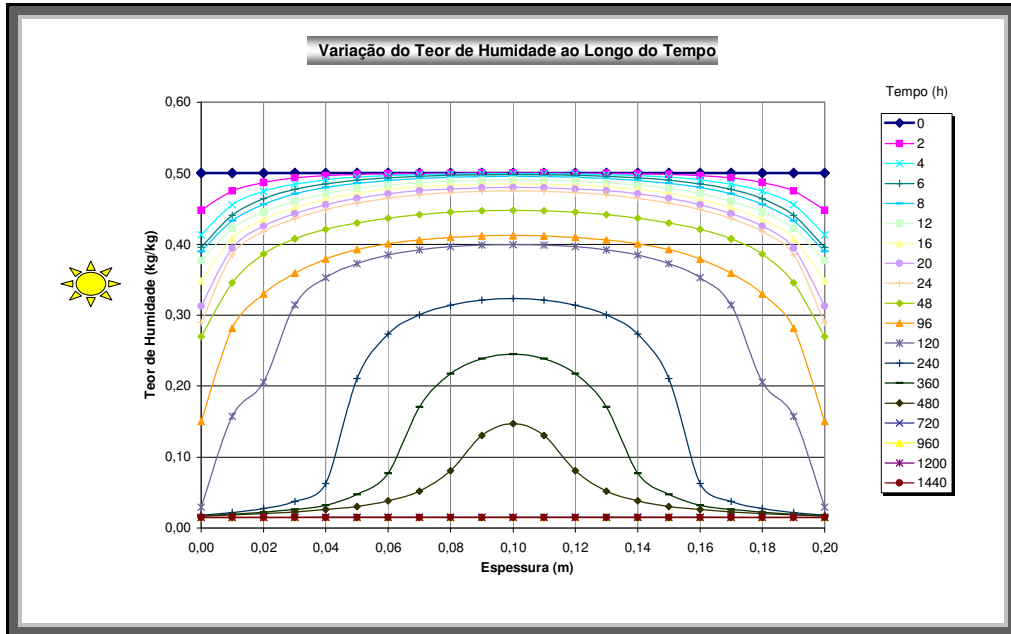


Fig. 64- Secagem do betão celular em função da humidade relativa (10%-10%)

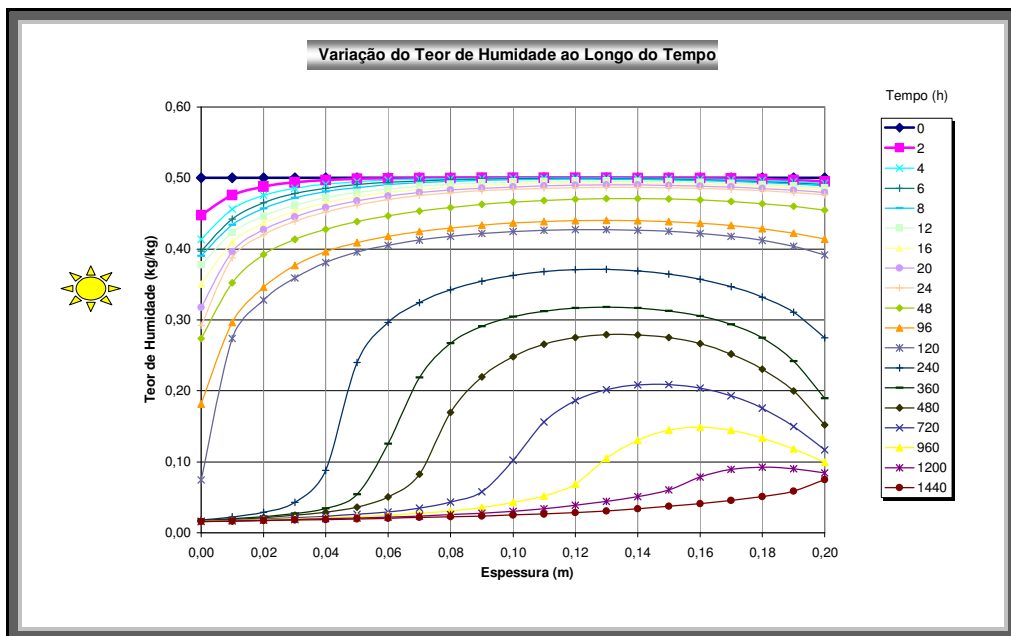


Fig. 65- Secagem do betão celular em função da humidade relativa (10%-90%)

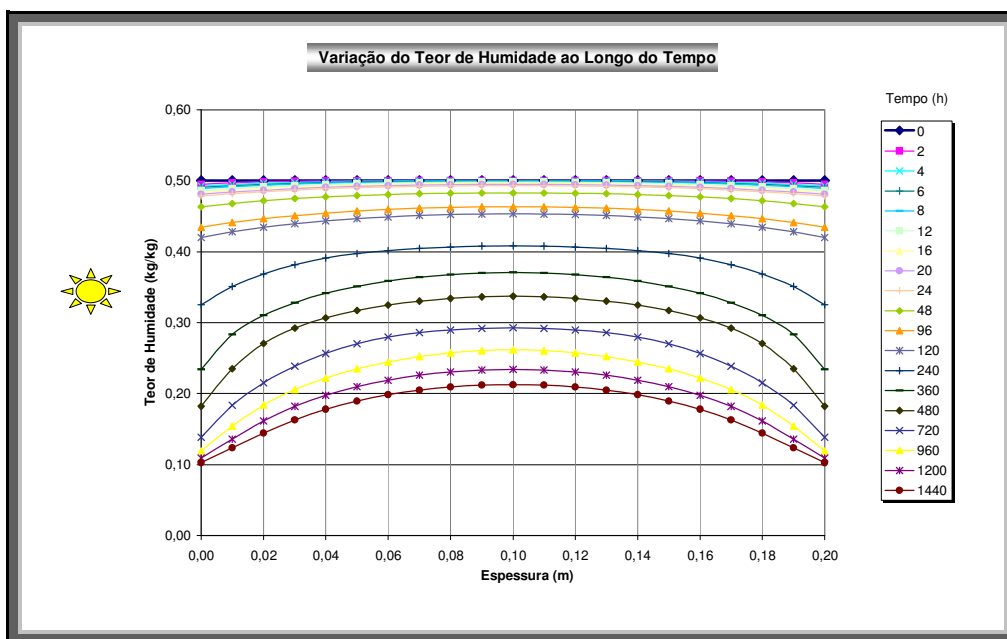


Fig. 66- Secagem do betão celular em função da humidade relativa (90%-90%)

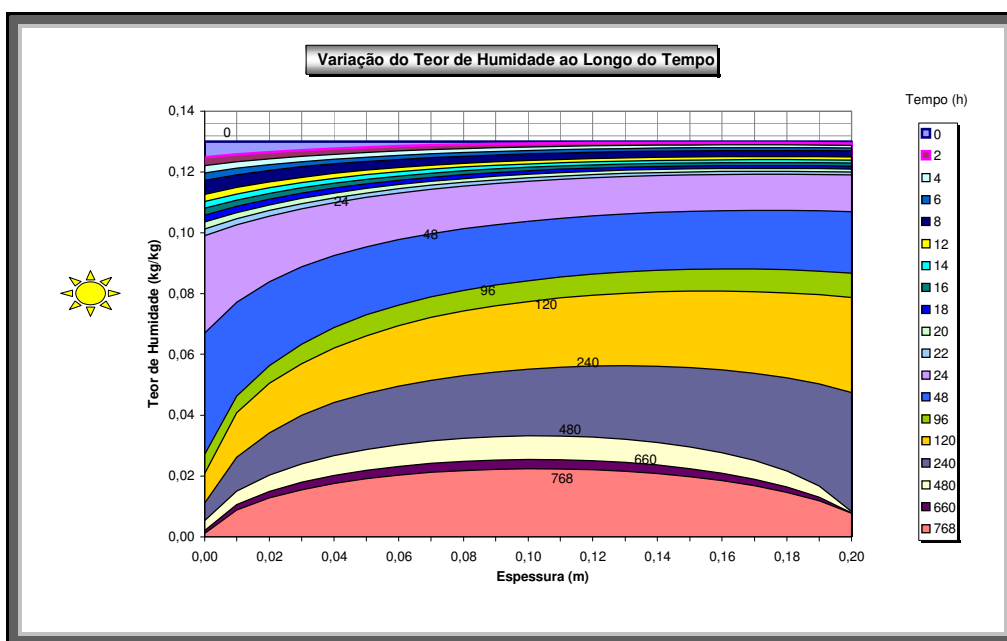


Fig. 67- Secagem diferencial do barro vermelho em função da humidade relativa (10%-90%)

À primeira vista seria de esperar que quão mais próximo o valor do teor de humidade se encontrado do valor do teor de humidade higroscópico, mais rapidamente o material chegaria a uma situação de equilíbrio, mas tal não se verifica. Da comparação dos gráficos Fig. 65 e Fig. 66 podemos constatar que apesar da humidade relativa a 90% se encontrar mais próxima do teor de humidade considerado, o material atinge mais facilmente o

equilíbrio a 10%. Este fenómeno explica-se facilmente pelo facto de uma ambiência praticamente saturada de humidade, constituir uma resistência à secagem do material.

Um maior diferencial entre as duas faces da amostra permite a secagem muito mais rápida da face com menor humidade relativa, determinando também uma secagem mais rápida de todo o material.

5.1.4.3 Radiação

A radiação influiu na velocidade de secagem do material e na forma como a humidade se distribui ao longo da amostra. Na Fig. 68 pode-se verificar que a radiação actua como um motor de secagem, fazendo deslocar o perfil hídrico na direcção do interior.

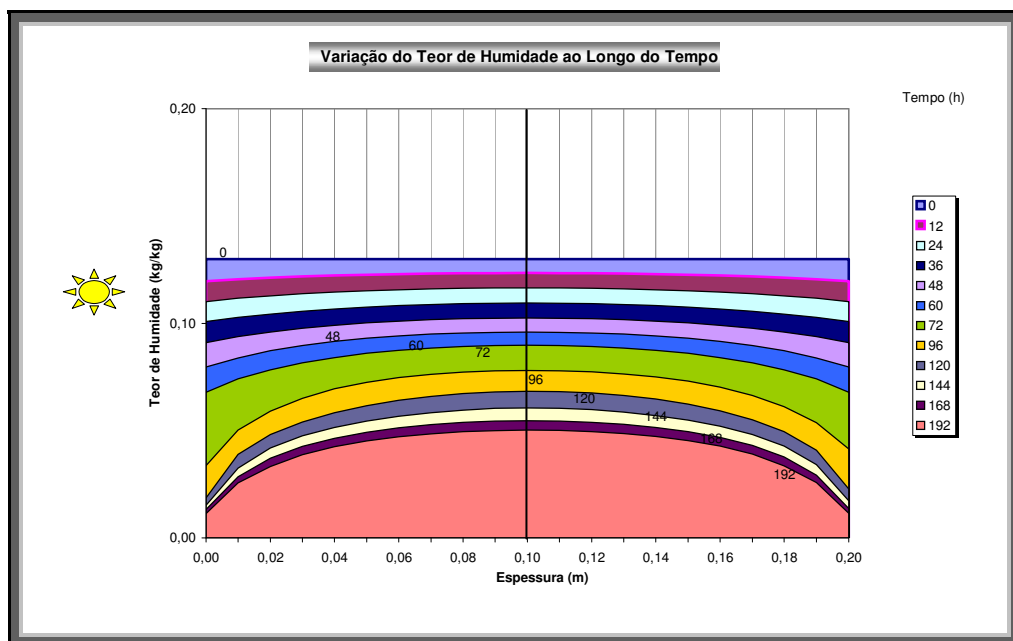


Fig. 68 – Deslocamento dos perfis do teor de humidade para o interior do material em consequência do efeito da radiação

5.1.4.4 Chuva

A chuva funciona como um motor de embebição que faz avançar a frente húmida do exterior para o interior da amostra.

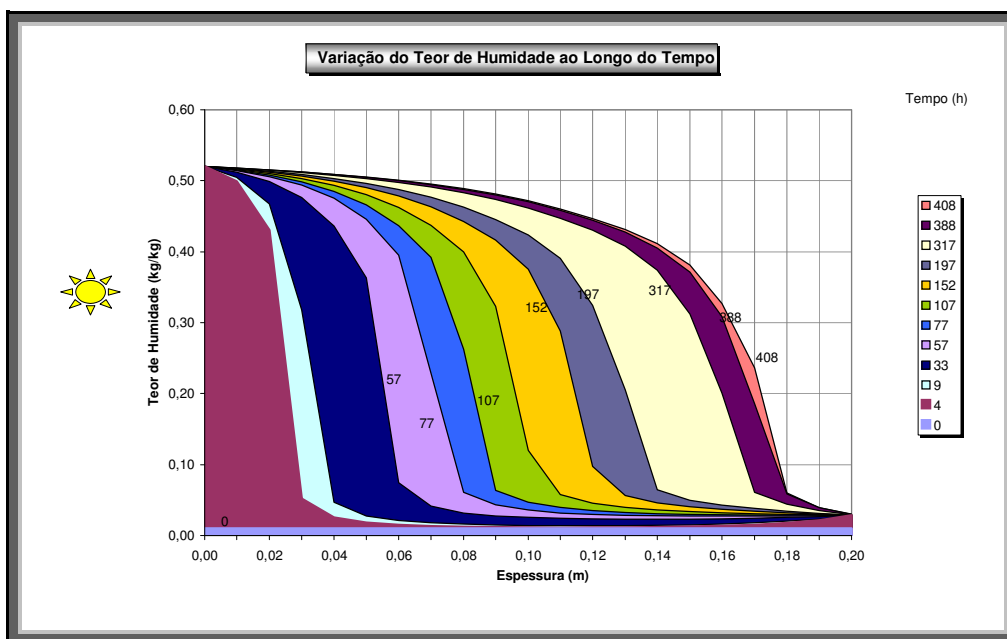


Fig. 69 – Humidificação do betão celular por chuva incidente (humidade relativa interior 50%)

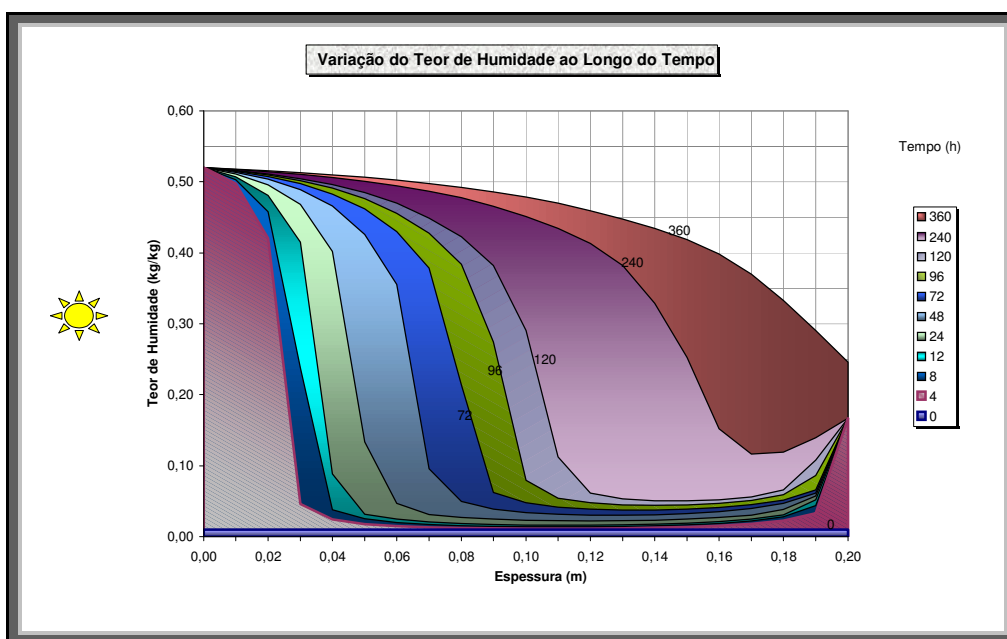


Fig. 70 – Humidificação do betão celular por chuva incidente (humidade relativa interior 95%)

O mecanismo pode ser ampliado pela elevada humidade relativa interior do ambiente, avançando a frente húmida com maior rapidez, como se pode verificar na comparação entre as Fig. 69 e Fig. 70, em que se assiste a um aumento do teor de humidade do material quando se aumenta a humidade relativa da ambiência interior de 50% para 95%.

5.1.5 Simulação de situações de referência

Para validação do programa *TrHum 98*, procedeu-se a uma série de simulações de situações já referenciadas em literatura [1]. Estas simulações permitiram determinar, sem recurso à validação experimental, o nível de concordância com os modelos já estabelecidos.

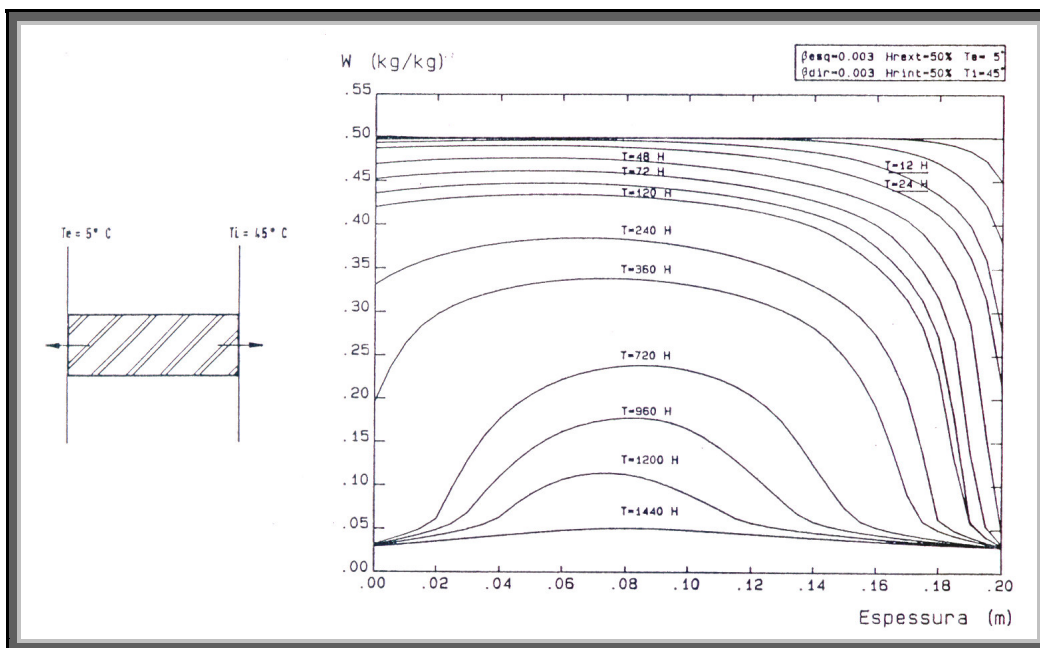


Fig. 71 – Secagem diferencial do betão celular sob gradiente de temperaturas (5-45°C) - Trhumidade [1]

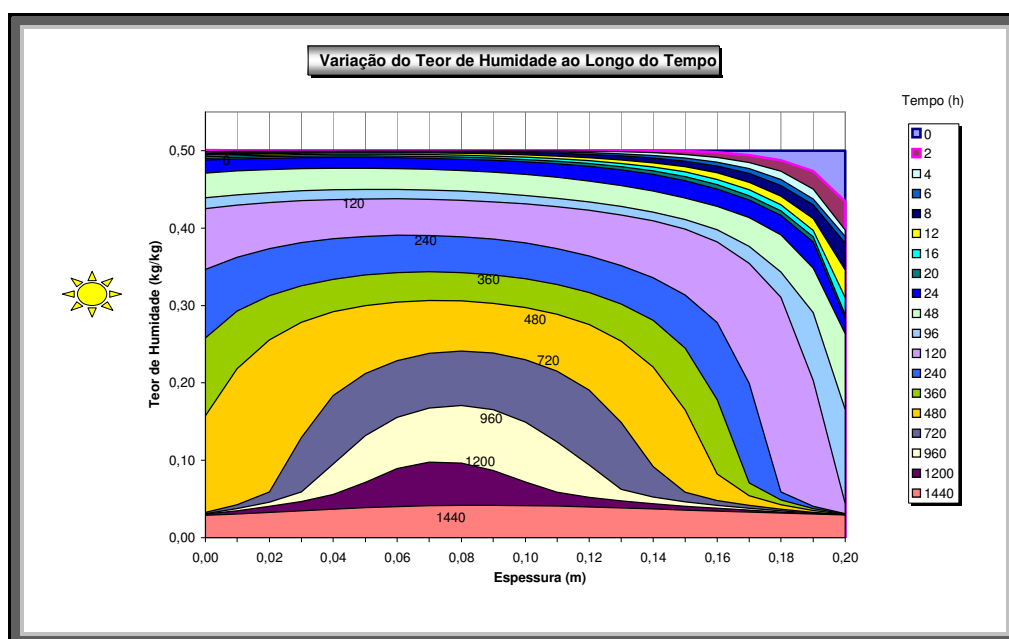


Fig. 72 – Secagem diferencial do betão celular sob gradiente de temperaturas (5-45°C) – TrHum 98

Da comparação entre os gráficos da Fig. 71 e da Fig. 72 podemos verificar os que existe uma boa concordância, entre o resultado das simulações efectuadas no programa *Trhumidade* e no programa *TrHum 98*.

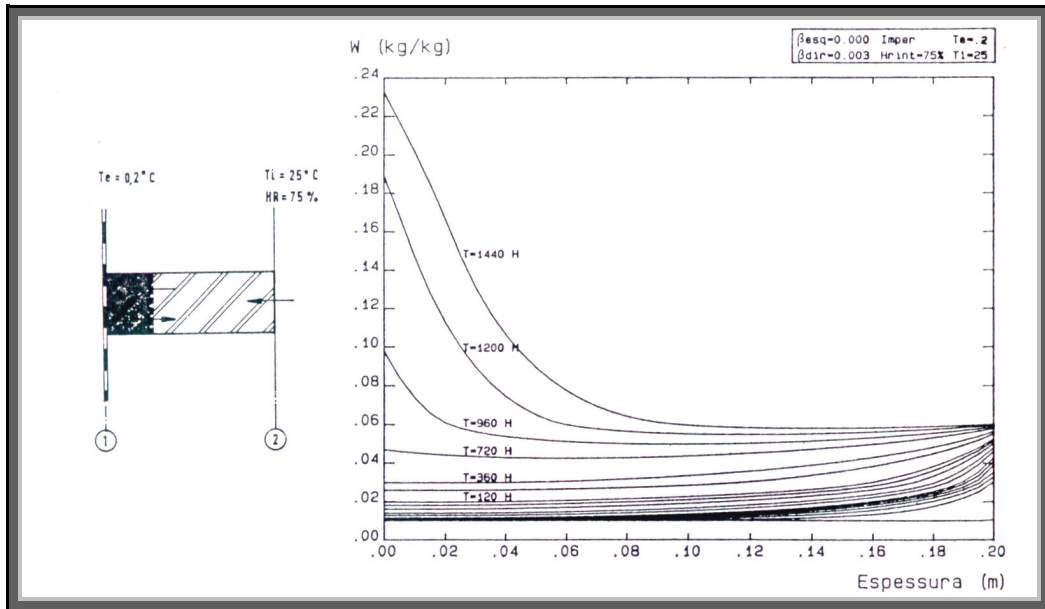


Fig. 73 – Embebição diferencial do betão celular em função da humidade relativa (10%-90%) e da temperatura (0.2°C – 45°C) – Trhumidade [1]

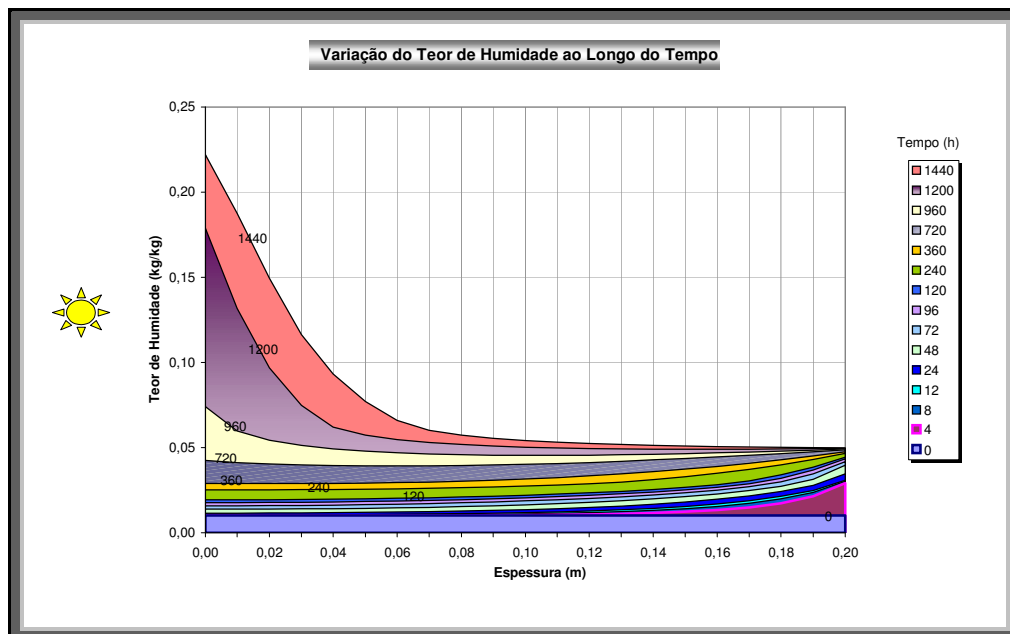
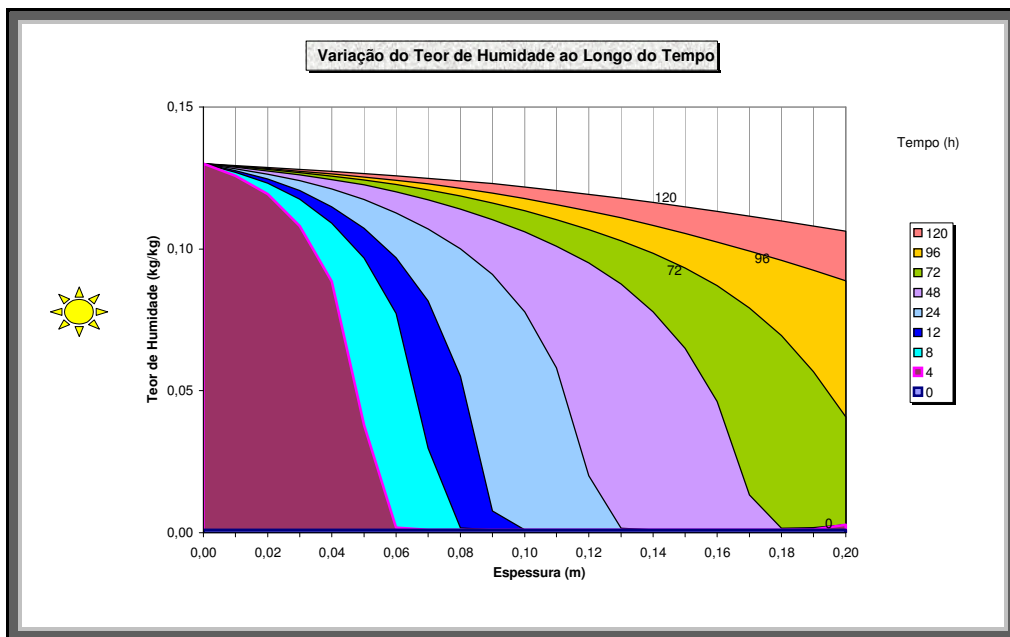
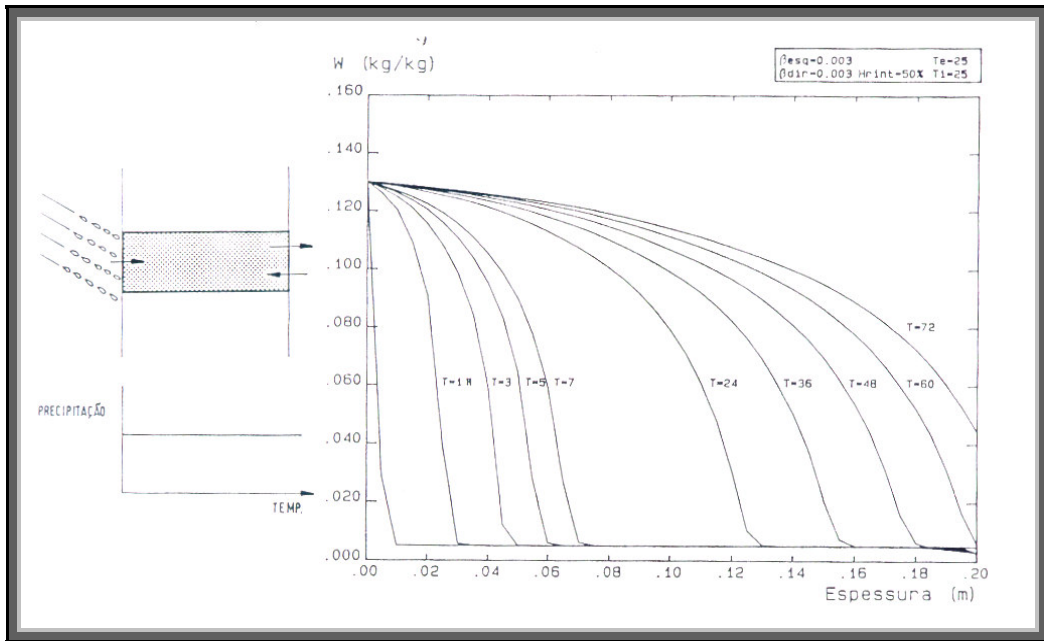


Fig. 74 – Embebição diferencial do betão celular em função da humidade relativa (10%-90%) e da temperatura (0.2°C – 45°C) – TrHum 98

Foram feitas, para validação, simulações de secagem, de embebição, e de embebição e secagem, sendo os valores obtidos aproximados aos disponíveis em gráficos, como pode ser verificado na sequência de gráficos seguintes. As diferenças ligeiras entre os gráficos apresentados nas Fig. 73, Fig. 74. São justificáveis pelo facto de não ser possível determinar com rigor o valor da totalidade dos parâmetros de caracterizados usados na simulação.



Os gráficos das Fig. 75 e Fig. 76, que representam o comportamento do material sob chuva incidente são Praticamente iguais.

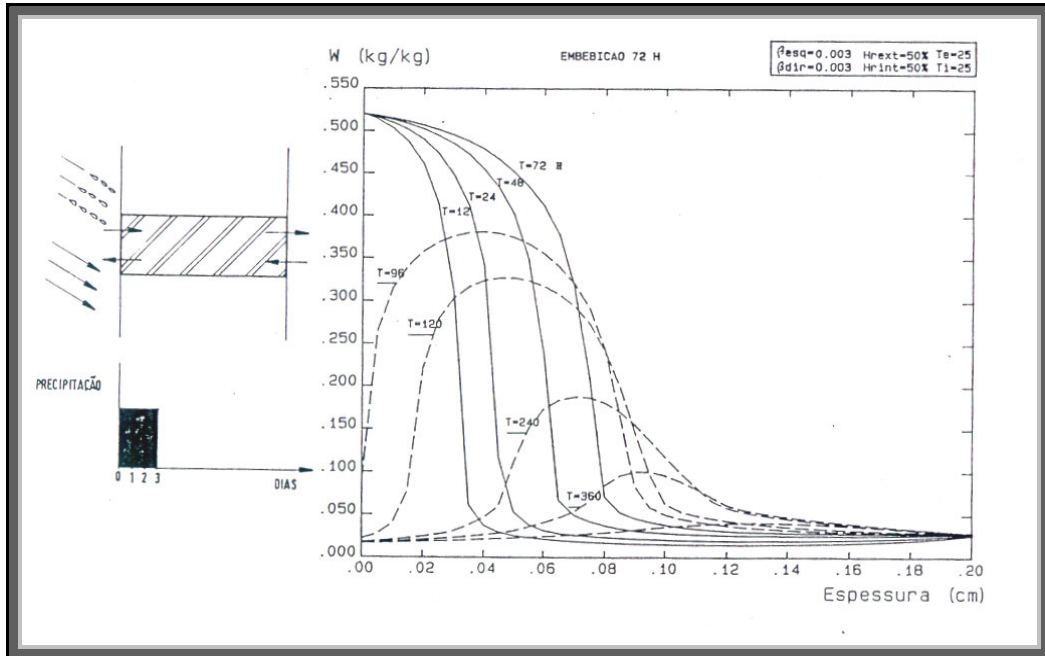


Fig. 77 – Processo de embebição e secagem – Trhumidade [1]

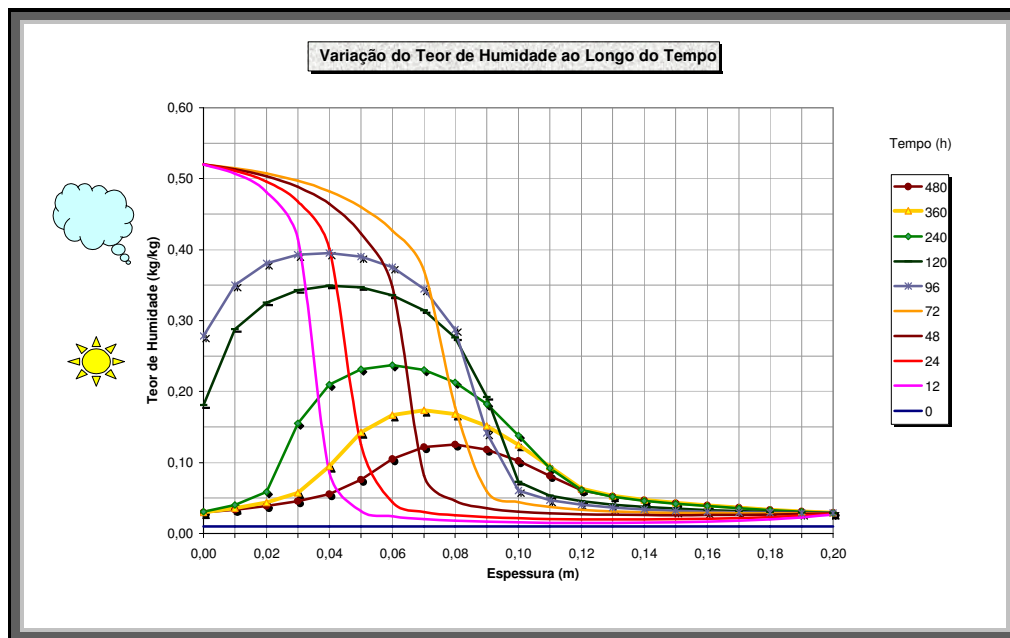


Fig. 78 – Processo de embebição e secagem – TrHum 98

Nas Fig. 77 e Fig. 78 estão representados os perfis de teor de umidade num processo de embebição e secagem. Existe uma variação nos valores correspondentes à fase de secagem, que pode ter a mesma razão atrás apontada.

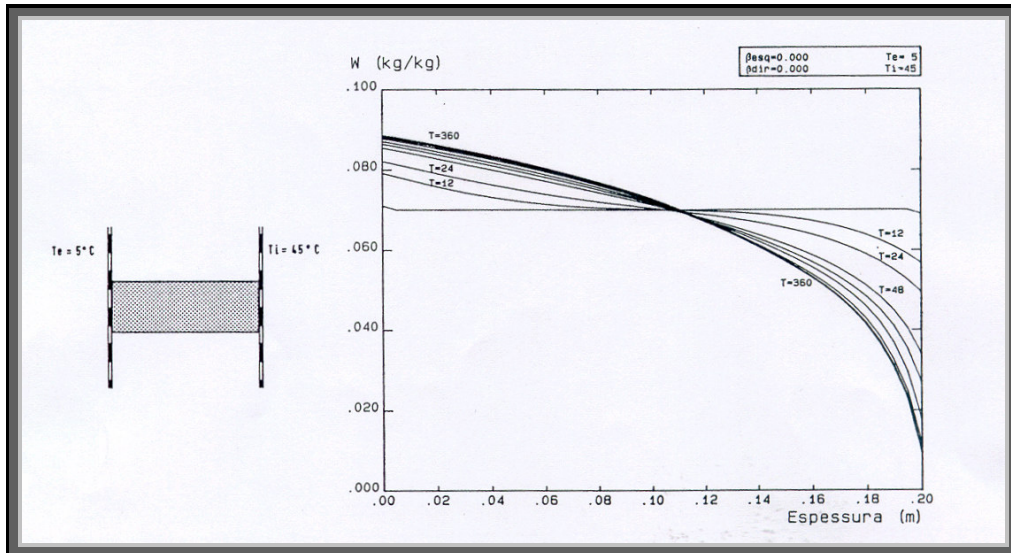


Fig. 79 – Humedecimento por condensação junto à superfície fria – Trhumidade [1]

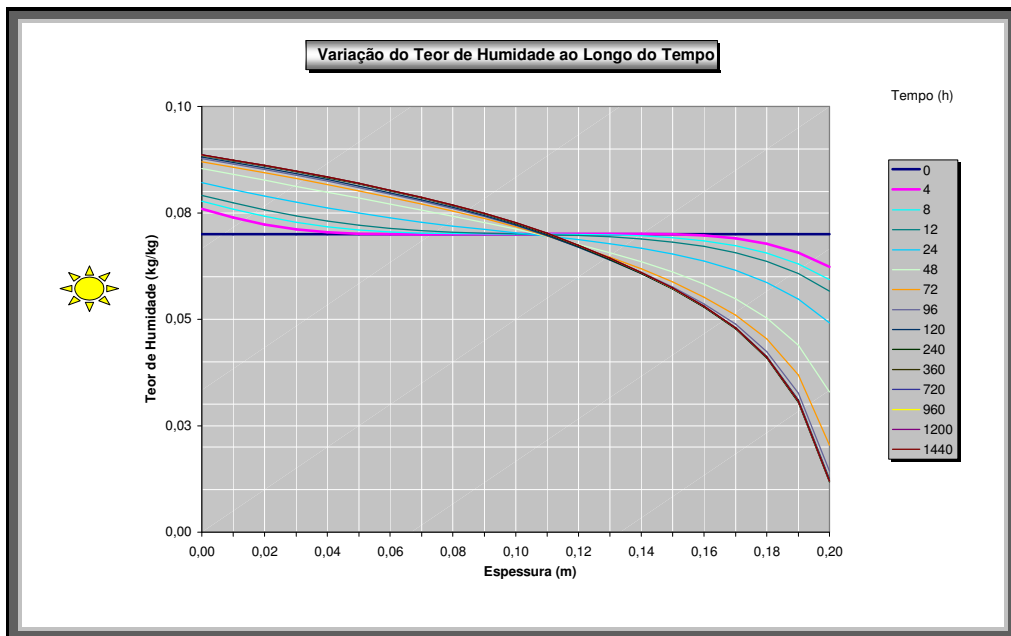


Fig. 80 – Humedecimento por condensação junto à superfície fria – TrHum 98

As Fig. 79 e Fig. 80 mostram as curvas do teor de umidade do material quando se anulam as trocas de umidade com a envolvente, por anulação dos coeficientes de transferência

superficial de humidade interior e exterior. A diferença acentuada de temperatura associada determina o aparecimento de condensações junto à face fria do material.

5.1.6 Análise dos resultados da simulação

Das diversas simulações numéricas efectuadas, foi possível tirar algumas conclusões, que passamos a explicitar:

- O intervalo de tempo do passo da simulação, quando a mesma é feita com clima constante, não tem praticamente influência sobre o resultado final da simulação, isto é, sobre os perfis instalados quando o material atinge o equilíbrio, ou sobre a determinação do tempo que o material leva a atingir o equilíbrio. Pode-se aproveitar para rentabilizar o tempo da simulação, utilizando um período de curta duração, cerca de 1 minuto, no início da simulação e aumentando-o sucessivamente até atingir no máximo 1 hora;
- Para efectuar simulações com clima variável, quanto menor for o intervalo de tempo, melhor será a aproximação à realidade. Deverá ser sempre tomado em conta o tempo necessário para efectuar a simulação, que aumenta consideravelmente com a diminuição do passo;
- Os perfis iniciais de temperatura do material, não tem praticamente influência na evolução dos perfis hídricos, para simulações com duração superior a 24 horas, para os materiais testados;
- Os perfis iniciais de teor de humidade são determinantes na evolução dos perfis hídricos, do material em secagem. Essa influência sente-se tanto mais quanto maior for a humidade relativa da ambiência. Para materiais em embebição essa influencia não é tão determinante;
- Num processo de secagem, a humidade relativa ambiente tem uma influência determinante na evolução dos perfis hídricos do material. Num processo de embebição essa influência também se faz sentir ainda que de forma mais reduzida.

6 CONCLUSÕES 102

6.1	CONSIDERAÇÕES FINAIS	102
6.2	SÍNTESE CRÍTICA DOS RESULTADOS	102
6.3	DESENVOLVIMENTO FUTURO DA INVESTIGAÇÃO NESTE DOMÍNIO.....	103

6 Conclusões

6.1 Considerações Finais

O estudo da transferência de humidade e calor nas paredes dos edifícios tem um profundo interesse. Com este trabalho consideramos que foi dado mais um pequeno passo no longo caminho que ainda falta percorrer.

Esperamos que o programa *TrHum 98* se venha a revelar um auxiliar importante na modelização do comportamento de materiais de construção, e uma ferramenta de ajuda na concepção.

O trabalho realizado assentou fundamentalmente na elaboração do programa *TrHum 98*, para ambiente *Windows*, encarado como um programa de fácil utilização e divulgação, e no seu teste e validação. Assim foram considerados e ponderados todos os aspectos de relação com o utilizador, entrada de dados, saída de resultados, visualização do cálculo, que nos pareceram ser os mais correctos para facilidade de utilização pretendida. Contudo só um uso sistemático por diferentes pessoas, com diferentes formações, poderá testar efectivamente a versatilidade, a facilidade e a eficácia do programa.

O testes sistemáticos feitos ao *TrHum 98* permitiram aferir a facilidade de utilização e a versatilidade do mesmo.

6.2 Síntese crítica dos resultados

A análise dos resultados das simulações efectuadas permite-nos considerar que a descrição matemática proposta por Luikov e Philip /De Vries, bem como a aplicação da mesma no programa *TrHum 98* constituem uma ferramenta para o estudo do comportamento de materiais de construção quando sujeitos à transferência conjunta de calor e humidade, com as restrições indicadas no corpo deste trabalho, designadamente o facto de só se aplicar a elementos constituídos por uma única camada. Apesar das restrições apontadas o campo de aplicação do é muito vasto.

6.3 Desenvolvimento futuro da investigação neste domínio

Um dos trabalhos que se nos afigura como continuação possível do presente estudo é a exploração sistemática do programa, para a sua completa validação e para a inclusão de funcionalidades acrescidas em versões futuras. Outro será a recolha ou determinação experimental dos diversos parâmetros de caracterização para outros materiais que não os estudados, para o enriquecimento da base de dados do programa e o alargamento do âmbito de simulações possíveis. Terá também todo o interesse a caracterização do clima para vários locais de Portugal, e mesmo da Europa, para possibilitar um maior rigor na simulação do comportamento de por exemplo materiais postos em obra, fundamentalmente no que se refere a ciclos de humedecimento e secagem.

Ainda como linha de trabalho futuro, julgamos oportuno referenciar a revisão do modelo para paredes de múltiplas camadas, com a determinação das condições de continuidade entre os materiais, na sequência do programa *Trhumidade*.

7 LISTA DE REFERÊNCIAS BIBLIOGRÁFICAS 105

7 Lista de Referências Bibliográficas

- [1] FREITAS, Vasco Peixoto de - Transferência de Humidade em Paredes de Edifícios - Análise do Fenómeno de Interface - Tese de Doutoramento - FEUP 1992.
- [2] CRAUSSE, Pierre - Etude fondamentale des transferts couplés de chaleur et d'humidité en milieux poreux non saturé, Thèse d'état, Institut National Polytechnique de Toulouse, Toulouse, Janeiro 1983.
- [3] HENRIQUES, Fernando Manuel Anjos - Acção da Humidade em Paredes - Formas de manifestação, critérios de quantificação e análise de soluções de reparação - Tese de Doutoramento - IST 1992.
- [4] FEDERATION NATIONAL DU BATIMENT - Etudes des Transferts Simultanés de Chaleur et d'Humidité dans les Parois Multicouches. – Application à la modélisation du comportement hygrothermique des locaux en régime varie – Direction de la recherche
- [5] HOLM, Andreas - Moisture Migration in Building Elements - Analysis of the moisture transmission throughout an interface – Report - FEUP 1995
- [6] INTERNATIONAL SYMPOSIUM ON MOISTURE PROBLEMS IN BUILDING WALLS - Humidade em Paredes de Edifícios - Porto 1995.
- [7] INTERNATIONAL ENERGY AGENCY - Enquiry on HAMCaT Codes - Report Annex 24, Task 1, Modelling
- [8] RCCTE - Regulamento das Características do Comportamento Térmico dos Edifícios, decreto-lei n.º 40/90, de 6 de Fevereiro de 1990.
- [9] PINA, Heitor – Métodos Numéricos – McGraw-Hill – 1995
- [10] CONTE, Samuel D.;BOOR, Carl de – Elementary Numerical Analysis – An Algorithmic Approach – McGraw-Hill International Editions – 1987
- [11] KRISCHER – Technique du Séchage - Trad. CETIAT

- [12] VOS,B.H. – Internal Condensation in Structures - Building Sci. - Vol.3 - Pergamon Press - Great Britain - 1969
- [13] LUIKOV, A. – Heat and Mass Transfer in Capillary Porous Bodies - Pergamon Press – 1966
- [14] PHILIP, J.; DE VRIES, D. – Moisture Movement in Porous Materials Under Temperature Gradients – Trans. American Geophysical Union – USA – 1957
- [15] DE VRIES, D. – Simultaneous Transfer of Heat and Moisture in Porous Media – Trans. American Geophysical Union – USA – 1958
- [16] DE VRIES, D. – The Theory of Heat and Moisture Transfer in Porous Media Revisited – J. Heat an Mass Transfer – Vol. 30 – 1987
- [17] ABRANTES, Vítor – Análise Numérica e Experimental do Comportamento Térmico de Coberturas em Desvão – Tese de Doutoramento – Porto – FEUP - 1984

Outras Fontes Bibliográficas

- [18] FREITAS, Vasco Peixoto de – Transferts d’humidité dans les systèmes d’isolation extérieure avec peau étanche en régime stationnaire – Provas de aptidão pedagógica e capacidade científica – FEUP 1985
- [19] BURCH, D.M.; TSONGAS, G.A.; WALTON, G.N: - Mathematical Analysis of practices to control moisture in the roof cavities of manufactured houses, NIST 1996
- [20] BURCH, D.M.; TSONGAS, G.A.; WALTON, G.N; THORNTON, B.A.: - A computer analysis of the moisture performance of roof constructions in U.S. DOE Moisture Control Handbook – NIST 1996
- [21] BURCH, D.M; ZAR, R.R.; FANNEY,A.H:.- Experimental verification of a moisture and heat transfer model in the hygroscopic regime – Cleawater Beach 1995
- [22] BURCH, D.M.; TSONGAS, G.A.; WALTON, G.N: - An Analysis of moisture accumulation in the roof cavities of manufactured housing, NIST 1995

- [23] PAIVA, J. Vasconcelos – Compilação de dados climáticos directamente ligados aos problemas da humidade – Cap. VI do relatório “Humidade nas Edificações” – LNEC 1970
- [24] FAUCONNIER, R.; FLORENCE, B.; LAUGIER, D. – Le transfert de l’humidité dans les matériaux isolants – Etudes Thermiques et Aérauliques – Junho 1979
- [25] FEDERATION NATIONAL DU BATIMENT – Transport d’humidité isotherme dans les matériaux poreux de construction – Institut de Physique appliquée de la société Fraunhofer - 1976

ANEXO A - FICHEIROS DO PROGRAMA TRHUM 98 109

A.1. FICHEIROS DE PROJECTO – (*.HUM).....	109
A.2. FICHEIROS DE RESULTADOS - (*.RES)	115
A.3. FICHEIROS DE RESULTADOS - (*.CSV).....	119
A.4. FICHEIROS DA BASE DE DADOS DE MATERIAIS (*.DBK).....	120
A.5. FICHEIROS DE CLIMA (*.WET).....	125

Anexo A Ficheiros do programa TrHum 98

A.1 Ficheiros de projecto – (*.hum)

Os dados necessários à execução do TrHum 98 estão contidos num ficheiro de texto, com a forma a seguir apresentada. A opção por um ficheiro de texto, que é facilmente editável num programa de processamento de texto, tem como principal razão o facto de constituir mais uma forma alternativa de introduzir dados no programa.

```
1 98
2 Project Name:
3 TESTE - 6-24
4 Project Description:
5 ciclo de humedecimento e secagem
6 Project Notes:
7 Ti=Te=25 Be=Bi=0.003 HRi=HRe=50
8 Material Name:
9 BetaoCelular
10 Material Code:
11 1
12 Total Thickness:
13 0,2
14 Number of Layers:
15 1
16 Layer Number 1:
17 1
18 Thickness layer 1:
19 20
20 Number of Steps layer 1:
21 20
22 Step of Layer 1:
23 0,01
24 Layer Number 2:
25 2
26 Thickness layer 2:
27 0
28 Number of Steps layer 2:
29 2
30 Step of Layer 2:
31 0
32 Layer Number 3:
33 3
34 Thickness layer 3:
35 0
36 Number of Steps layer 3:
37 4
38 Step of Layer 3:
39 0
40 Project start date:
41 10-01-1999
42 Project Finish date:
43 11-02-1999
44 Project start hour :
45 0:00:00
46 Project Finish hour:
47 0:00:00
48 Starting Year:
49 1999
50 Day of the Year:
51 10
52 Hour of the day:
53 0
54 Total Calculation Time:
55 500
56 Total weather file Time:
57 1440
58 Indoor superficial conductance:
59 25
60 Outdoor superficial conductance:
61 25
62 Indoor moisture coefficient:
63 0,003
64 Outdoor moisture coefficient:
65 0,003
66 Maximum error:
67 0,001
68 City:
```

```

35 Porto
36 Country:
Portugal
36 Latitude:
37 42
North:
38 1
Longitude:
39 8
West:
40 1
Wind direction:
41 80
Wind speed:
42 3
Azimuth of the perpendicular plane to the surface:
43 180
Initial temperature's Profile:
44 25;25;25;
Initial moisture's Profile:
45 0,5 ; 0,5 ; 0,5
Absorption coefficient for radiation:
46 0,5
Number of profiles and time:
47 9
48 12 ; 24 ; 48 ;72; 96 ; 120 ; 240 ; 360 ; 480 ;

49 0
Beginning of the Weather file
Number of time intervals:
50 10
Number ;time ;step ; Tempout ;HRout ;Rad ;Rain ;Tempin ;HRin ;
51 1 ; 72 ; 0,01 ; 25 ; 50 ; 0 ; 1 ; 25 ; 50
2 ; 444 ; 0,01 ; 25 ; 50 ; 0 ; 0 ; 25 ; 50
3 ; 144 ; 0,05 ; 25 ; 10 ; 0 ; 0 ; 25 ; 50
4 ; 144 ; 0,1 ; 25 ; 10 ; 0 ; 0 ; 25 ; 50
5 ; 144 ; 0,5 ; 25 ; 10 ; 0 ; 0 ; 25 ; 50
6 ; 144 ; 0,5 ; 25 ; 10 ; 0 ; 0 ; 25 ; 50
7 ; 144 ; 1 ; 25 ; 10 ; 0 ; 0 ; 25 ; 50
8 ; 144 ; 1 ; 25 ; 10 ; 0 ; 0 ; 25 ; 50
9 ; 144 ; 1 ; 25 ; 10 ; 0 ; 0 ; 25 ; 50
10; 144 ; 1 ; 25 ; 10 ; 0 ; 0 ; 25 ; 50
End of weather file
Beginning of the Material Data Bank
Material Name:
5 BetaoCelular
Material code
6 1
Density:
52 525
Heat
53 1050
Hygroscope
54 0,2
Saturation
55 0,52
Number of temperature list
56 3
Number of moist list
57 15
Temperature list:
58 10
30
50
Moisture list:
59 0,011
0,016
0,031
0,062
0,158
0,272
0,388
0,472
0,5
0,511
0,52
0,57
0,62
0,67
0,7
Dt list
60 0,00000000000001
0,00000000000002

```

0,00000000000035
0,00000000000045
0,00000000000017
0,00000000000024
0,00000000000034
0,00000000000032
0,00000000000038
0,00000000000038
0,000000000000385
0,000000000000385
0,000000000000385
0,000000000000398
0,00000000000019
0,000000000000151
0,00000000000001
0,00000000000002
0,00000000000035
0,00000000000045
0,00000000000017
0,00000000000024
0,00000000000034
0,00000000000032
0,00000000000038
0,00000000000038
0,000000000000385
0,000000000000385
0,000000000000385
0,000000000000398
0,00000000000019
0,000000000000151
0,00000000000001
0,00000000000002
0,00000000000035
0,00000000000045
0,00000000000017
0,00000000000024
0,00000000000034
0,00000000000032
0,00000000000038
0,00000000000038
0,000000000000385
0,000000000000385
0,000000000000385
0,000000000000398
0,00000000000019
0,000000000000151

Dw list

61

0,00000008
0,00000004
0,00000008
0,00000001
0,00000000831
0,0000000183
0,0000000671
0,000000268
0,000000586
0,000000887
0,00000144
0,0000009
0,0000006
0,0000003
0,0000005
0,0000008
0,0000004
0,00000008
0,00000001
0,0000000831
0,0000000183
0,0000000671
0,000000268
0,000000586
0,000000887
0,00000144
0,0000009
0,0000006
0,0000003
0,0000005
0,0000008
0,0000004
0,00000008
0,00000001
0,0000000831
0,0000000183
0,0000000671
0,000000268
0,000000586
0,000000887
0,00000144
0,0000009

0,00000006
0,00000003
0,00000005
Capillary list
62 687000000
506000000
261000000
66700000
12400000
4190000
1380000
394000
235000
191000
162000
113000
12800
3930
307
653000000
481000000
248000000
63400000
11800000
3990000
1320000
374000
224000
182000
154000
107000
12200
4790
292
619000000
457000000
235000000
61000000
11200000
3780000
1250000
355000
212000
173000
146000
102000
11500
4540
277
Conduct list
63 0,135
0,138
0,142
0,155
0,194
0,296
0,314
0,328
0,334
0,339
0,334
0,362
0,387
0,41
0,419
0,135
0,138
0,142
0,155
0,194
0,296
0,314
0,328
0,334
0,339
0,334
0,362
0,387
0,41
0,419
0,135
0,138
0,142
0,155
0,194
0,296

	0,314
	0,328
	0,334
	0,339
	0,334
	0,362
	0,387
	0,41
	0,419
	Moisture Storage Function
	Number of points
64	5
65	0
	10
	70
	90
	100
66	0
	0,015
	0,035
	0,075
	0,259

Legenda

- 1 Identifica se é ou não um ficheiro de projecto do TrHum 98 (98 – sim / outro valor - não)
- 2 Nome do projecto
- 3 Descrição do projecto
- 4 Notas sobre o projecto
- 5 Nome do material
- 6 Código do material (m)
- 7 Espessura do material (m)
- 8 Numero de sub-camadas
- 9 Sub-camada 1
- 10 Espessura da sub-camada 1 (m)
- 11 Número de nós na sub-camada 1
- 12 Espaçamento da malha de nós da sub-camada 1 (m)
- 13 Sub-camada 2
- 14 Espessura da sub-camada 2 (m)
- 15 Número de nós na sub-camada 2
- 16 Espaçamento da malha de nós da sub-camada 2 (m)
- 17 Sub-camada 3
- 18 Espessura da sub-camada 3 (m)
- 19 Número de nós na sub-camada 3
- 20 Espaçamento da malha de nós da sub-camada 3 (m)
- 21 Data de início do projecto (dd-mm-aaaa)
- 22 Data de fim do projecto (dd-mm-aaaa)
- 23 Hora de início do projecto (hh:mm:ss)
- 24 Hora de fim do projecto (hh:mm:ss)
- 25 Ano de início do projecto
- 26 Dia desde o início do ano
- 27 Hora de início
- 28 Tempo total de cálculo (h)
- 29 Tempo total em que existe definição de clima (h)
- 30 Condutância térmica superficial interior – h_i ($W/m^2\text{°C}$)
- 31 Condutância térmica superficial exterior – h_e ($W/m^2\text{°C}$)
- 32 Coeficiente de transferência superficial de humidade interior - β_i (m/s)
- 33 Coeficiente de transferência superficial de humidade exterior - β_e (m/s)
- 34 Erro máximo entre iterações sucessivas
- 35 Cidade
- 36 País
- 37 Latitude (°)

- 38 Norte ou Sul (Norte – 1 / Sul - 0)
- 39 Longitude (°)
- 40 Oeste ou Este (oeste – 1/ Este – 0)
- 41 Direção do vento (°) (↑=0 +=sentido horário)
- 42 Velocidade do vento (m/s)
- 43 Orientação (°) (↑=0 +=sentido horário)
- 44 Temperatura inicial do material (°C)
- 45 Teor de humidade inicial do material (kg/kg)
- 46 Coeficiente de absorção do material - α
- 47 Número de perfis a escrever no ficheiro de resultados
- 48 Instante dos perfis que deverão ser escritos no ficheiro de resultados (h)
- 49 Repetição do ficheiro de definição de clima (1 – sim / 0 – não)
- 50 Número de linhas da definição do clima
- 51 Parâmetros de definição do clima
 - A – Identificação do número da linha
 - B – Tempo em que as condições climáticas são mantidas (h)
 - C – Intervalo de tempo (h)
 - D – Temperatura exterior (°C)
 - E – Humidade relativa exterior (%)
 - F – Radiação solar em plano horizontal (W/m^2)
 - G – Ocorrência de precipitação (1 – Sim / 0 – Não)
 - H – Temperatura interior (°C)
 - I – Humidade relativa interior (%)
- 52 Massa volúmica seca (kg/m^3)
- 53 Calor específico ($J/kg.K$)
- 54 Humidade higroscópica (kg/kg)
- 55 Humidade de Saturação (kg/kg)
- 56 Número de valores de temperatura tabelados
- 57 Número de valores de teor de humidade tabelados
- 58 Valores de temperatura tabelados
- 59 Valores de teor de humidade tabelados
- 60 Valores do coeficiente de difusividade higrotérmica ($m^2/s.K$) *
- 61 Valores do coeficiente de difusividade hídrica (m^2/s) *
- 62 Valores da pressão capilar (N/m^2) *
- 63 Valores da condutibilidade térmica ($W/m.K$) *
- 64 Número de pontos tabelados da função que define o comportamento higroscópico do material
- 65 Valores da humidade relativa (%)
- 66 Valores do Teor de humidade (kg/kg)

* A ordenação dos valores é feita do seguinte modo:

Sendo t_1, t_2, t_3 o valor tabelado de temperatura

Sendo m_1, m_2, \dots, m_n o valor tabelado do teor de humidade

$V(t_1, m_1)$

$V(t_1, m_2)$

.....

$V(t_1, m_n)$

$V(t_2, m_1)$

$V(t_2, m_2)$

.....

$V(t_2, m_n)$

$V(t_3, m_1)$

$V(t_3, m_2)$

.....

$V(t_3, m_n)$


```
1 Profiles at [h]: 48
25 ; 0,52
25 ; 0,513195749341412
25 ; 0,503294794927291
25 ; 0,488266913015006
25 ; 0,464563035191374
25 ; 0,422634865681373
25 ; 0,347656643739272
25 ; 8,18817295519294E-02
25 ; 4,57971536592549E-02
25 ; 3,53593505089587E-02
25 ; 3,05017040651434E-02
25 ; 0,028203040366123
25 ; 0,026924836680973
25 ; 2,61579065687051E-02
25 ; 2,57383984783403E-02
25 ; 2,55909791105288E-02
25 ; 0,025675359890566
25 ; 2,59685885686432E-02
25 ; 2,64574440198731E-02
25 ; 2,71341286335212E-02
25 ; 0,027992040620701
1 Profiles at [h]: 72
25 ; 0,52
25 ; 0,514596199377979
25 ; 0,507336447129229
25 ; 0,497011390276734
25 ; 0,48222872322617
25 ; 0,460348228607715
25 ; 0,426535798344786
25 ; 0,370418349910005
25 ; 0,178201255833377
25 ; 0,059632797520577
25 ; 4,40584649265003E-02
25 ; 3,71644619760551E-02
25 ; 3,31250426268587E-02
25 ; 3,07614790080254E-02
25 ; 2,94978435148961E-02
25 ; 2,87894013566597E-02
25 ; 2,83860877308433E-02
25 ; 2,81862657536086E-02
25 ; 2,81320661603422E-02
25 ; 2,81805061620312E-02
25 ; 2,82914270246865E-02
1 Profiles at [h]: 96
25 ; 0,27820515681344
25 ; 0,349896023761221
25 ; 0,380900196154895
25 ; 0,393030816555067
25 ; 0,395363834860207
25 ; 0,390296292816303
25 ; 0,37514441384434
25 ; 0,344677749668384
25 ; 0,287382222727921
25 ; 0,141677911470101
25 ; 6,03896901650247E-02
25 ; 0,046952841179601
25 ; 4,05059006790661E-02
25 ; 3,64234182457891E-02
25 ; 3,36647952239972E-02
25 ; 3,17559992419333E-02
25 ; 3,05145535949322E-02
25 ; 2,97274864447035E-02
25 ; 2,91882013217851E-02
25 ; 2,87812372298691E-02
25 ; 2,84436128317129E-02
```

```
1 Profiles at [h]: 120
25 ; 0,180650124987868
25 ; 0,287575369801744
25 ; 0,325355217131351
25 ; 0,34273195732618
25 ; 0,349000029812928
25 ; 0,34652149275517
25 ; 0,335594140765256
25 ; 0,314648718336016
25 ; 0,276097977126198
25 ; 0,192076134301775
25 ; 7,26541228711687E-02
25 ; 5,29950638334084E-02
25 ; 4,52838724769298E-02
25 ; 4,05914281029872E-02
25 ; 3,72548387675759E-02
25 ; 3,47426248654906E-02
25 ; 3,27934148675575E-02
25 ; 3,12402886669261E-02
25 ; 0,030098636939721
25 ; 2,92502981163046E-02
25 ; 2,85529805318466E-02
1 Profiles at [h]: 240
25 ; 3,03221856245933E-02
25 ; 4,01614990368303E-02
25 ; 5,84065308780016E-02
25 ; 0,154992444145755
25 ; 0,209495550198708
25 ; 0,231552032793242
25 ; 0,237254894154024
25 ; 0,230642246013238
25 ; 0,212711690942274
25 ; 0,182232133591139
25 ; 0,137867009669563
25 ; 9,23027193620856E-02
25 ; 6,11011173969374E-02
25 ; 5,15144912414804E-02
25 ; 4,60245950604411E-02
25 ; 4,18637973484669E-02
25 ; 3,84252268389544E-02
25 ; 3,54501090684958E-02
25 ; 3,27968063087065E-02
25 ; 3,05161179624132E-02
25 ; 2,88112514452601E-02
1 Profiles at [h]: 360
25 ; 2,96402862108167E-02
25 ; 3,52106674628608E-02
25 ; 0,043580953594617
25 ; 5,70192496065423E-02
25 ; 9,47940367277969E-02
25 ; 0,142516120953426
25 ; 0,167089279721693
25 ; 0,173842759290644
25 ; 0,168350944970611
25 ; 0,151483291052571
25 ; 0,124774239237121
25 ; 9,42357216933727E-02
25 ; 6,31470536410406E-02
25 ; 5,31239114922831E-02
25 ; 4,74236708059764E-02
25 ; 4,30715692219412E-02
25 ; 3,94207366055072E-02
25 ; 3,62167351604631E-02
25 ; 3,33282302645909E-02
25 ; 3,07555066757338E-02
25 ; 0,028854557410328
```

```
1 Profiles at [h]: 480
25 ; 2,93033620504681E-02
25 ; 3,31826074790799E-02
25 ; 3,87106756648749E-02
25 ; 4,55265264432279E-02
25 ; 5,54655406709909E-02
25 ; 7,56345043179658E-02
25 ; 0,104730881472416
25 ; 0,121607990829912
25 ; 0,125471161824601
25 ; 0,118037051199229
25 ; 0,102252104686929
25 ; 8,11945238296469E-02
25 ; 6,06505659747195E-02
25 ; 5,21785798219744E-02
25 ; 0,046937928984614
25 ; 4,27961647495844E-02
25 ; 3,92607780673215E-02
25 ; 3,61244555131734E-02
25 ; 0,033276946939462
25 ; 3,07353787016691E-02
25 ; 2,88510391187732E-02
```

Legenda

- 1 Instante de tempo considerado (h)
- 2 Valores da temperatura (°C) e do teor de humidade (kg/kg) para cada nó, partindo do nó exterior.

A.3 Ficheiros de resultados - (*.csv)

O segundo ficheiro de resultados (com extensão .csv) é escrito quando o cálculo termina e corresponde a uma organização dos valores quer para uma melhor facilidade de leitura quer para o seu tratamento subsequente por outros programas, como por exemplo, o excel.

Apresenta os valores de temperatura e de teor de humidade organizados numa tabela de duas entradas, correspondendo uma à posição do nó e a outra ao instante de tempo considerado.

	Posição do nó				Instante de tempo				Temperatura	
Result file D:\1-tese\b_testes trhum\humidificacao\6-24-500-st001.csv										
Time profile - temperature profile										
	0;	12;	24;	48;	72;	96;	120;	240;	360;	480;
0	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,01	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,02	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,03	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,04	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,05	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,06	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,07	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,08	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,09	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,1	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,11	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,12	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,13	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,14	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,15	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,16	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,17	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,18	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,19	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
0,2	25;	25;	25;	25;	25;	25;	25;	25;	25;	25;
time profile - Moisture profile										
	0;	12;	24;	48;	72;	96;	120;	240;	360;	480;
0;	0,01;	0,52;	0,52;	0,52;	0,52;	0,278205;	0,18065;	3,03E-02	2,96E-02;	2,93E-02;
0,01;	0,01;	0,507108;	0,510897;	0,513196;	0,514596;	0,349896;	0,287575;	4,02E-02	3,52E-02;	3,32E-02;
0,02;	0,01;	0,480881;	0,495599;	0,503295;	0,507336;	0,3809;	0,325355;	5,84E-02	0,043581;	3,87E-02;
0,03;	0,01;	0,414723;	0,467986;	0,488267;	0,497011;	0,393031;	0,342732;	0,154992	5,70E-02;	4,55E-02;
0,04;	0,01;	8,61E-02;	0,4016;	0,464563;	0,482229;	0,395364;	0,349;	0,209496	9,48E-02;	5,55E-02;
0,05;	0,01;	3,12E-02;	0,125104;	0,422635;	0,460348;	0,390296;	0,346521;	0,231552	0,142516;	7,56E-02;
0,06;	0,01;	2,40E-02;	4,39E-02;	0,347657;	0,426536;	0,375144;	0,335594;	0,237255	0,167089;	0,104731;
0,07;	0,01;	2,05E-02;	3,00E-02;	8,19E-02;	0,370418;	0,344678;	0,314649;	0,230642	0,173843;	0,121608;
0,08;	0,01;	1,82E-02;	2,55E-02;	4,58E-02;	0,178201;	0,287382;	0,276098;	0,212712	0,168351;	0,125471;
0,09;	0,01;	1,66E-02;	2,31E-02;	3,54E-02;	0,059633;	0,141678;	0,192076;	0,182232	0,151483;	0,118037;
0,1;	0,01;	1,57E-02;	2,16E-02;	3,05E-02;	4,41E-02;	6,04E-02;	7,27E-02;	0,137867	0,124774;	0,102252;
0,11;	0,01;	1,51E-02;	2,06E-02;	0,028203;	3,72E-02;	0,046953;	5,30E-02;	9,23E-02	9,42E-02;	8,12E-02;
0,12;	0,01;	0,014937	2,01E-02;	0,026925;	3,31E-02;	4,05E-02;	4,53E-02;	6,11E-02	6,31E-02;	6,07E-02;
0,13;	0,01;	1,50E-02	1,98E-02;	2,62E-02;	3,08E-02;	3,64E-02;	4,06E-02;	5,15E-02	5,31E-02;	5,22E-02;
0,14;	0,01;	1,53E-02	1,99E-02;	2,57E-02;	2,95E-02;	3,37E-02;	3,73E-02;	4,60E-02	4,74E-02;	0,046938;
0,15;	0,01;	1,59E-02	2,03E-02;	2,56E-02;	2,88E-02;	3,18E-02;	3,47E-02;	4,19E-02	4,31E-02;	4,28E-02;
0,16;	0,01;	1,69E-02	2,09E-02;	0,025675;	2,84E-02;	3,05E-02;	3,28E-02;	3,84E-02	3,94E-02;	3,93E-02;
0,17;	0,01;	1,83E-02	2,19E-02;	2,60E-02;	2,82E-02;	2,97E-02;	3,12E-02;	3,55E-02	3,62E-02;	3,61E-02;
0,18;	0,01;	2,02E-02	2,32E-02;	2,65E-02;	2,81E-02;	2,92E-02;	0,030099;	3,28E-02	3,33E-02;	0,033277;
0,19;	0,01;	2,27E-02	0,024897;	2,71E-02;	2,82E-02;	2,88E-02;	2,93E-02;	3,05E-02	3,08E-02;	3,07E-02;
0,2;	0,01;	2,62E-02	2,72E-02;	0,027992;	2,83E-02;	2,84E-02;	2,86E-02;	2,88E-02	0,028855;	2,89E-02;
	Teor de Humidade									

A.4 Ficheiros da base de dados de materiais (*.dbk)

O programa possuiu ainda a possibilidade de leitura e escrita de ficheiros de arquivo das características dos materiais. Esses ficheiros são independentes de qualquer projecto e servem para guardar todos os parâmetros de caracterização de vários materiais que ficam assim disponíveis para simulações. O utilizador terá que escolher o ficheiro de base de dados de materiais que lhe interessa, carrega-lo no programa e escolher, dentre os disponíveis, o material que pretende. O ficheiro tem a seguinte estrutura:

```
1  MATERIAL LIST NUMBER:
2  2
3  Material List:
4  Betão Celular
5  Barro Vermelho
6  Density List:
7  525
8  1925
9  Heat List:
10 1050
11 920
12 Hygroscope List:
13 0,2
14 0,009
15 Saturation List:
16 0,52
17 0,13
18 Number of temperature list:
19 3
20 3
21 Number of moist list:
22 15
23 10
24 Temperature List:
25 10
26 30
27 50
28 10
29 30
30 50
31 Moisture List:
32 0,011
33 0,016
34 0,031
35 0,062
36 0,158
37 0,272
38 0,388
39 0,472
40 0,500
41 0,511
42 0,520
43 0,570
44 0,620
45 0,670
46 0,700
47 0,000
48 0,005
49 0,010
50 0,020
51 0,040
52 0,070
53 0,090
54 0,110
55 0,120
56 0,130
57 Dt List:
58 1,00E-13
59 2,00E-13
60 3,50E-13
61 4,50E-12
62 1,70E-11
63 2,40E-11
64 3,40E-11
65 3,20E-11
66 3,80E-11
67 3,80E-11
68 3,85E-11
69 3,85E-11
70 3,98E-11
71 1,90E-11
72 1,51E-12
73 1,00E-13
```

2,00E-13
3,50E-13
4,50E-12
1,70E-11
2,40E-11
3,40E-11
3,20E-11
3,80E-11
3,80E-11
3,85E-11
3,85E-11
3,98E-11
1,90E-11
1,51E-12
1,00E-13
2,00E-13
3,50E-13
4,50E-12
1,70E-11
2,40E-11
3,40E-11
3,20E-11
3,80E-11
3,80E-11
3,85E-11
3,85E-11
3,98E-11
1,90E-11
1,51E-12
4,00E-12
7,00E-12
1,00E-11
1,50E-11
2,00E-11
2,00E-11
1,70E-11
1,15E-11
7,80E-11
4,60E-12
4,00E-12
7,00E-12
1,00E-11
1,50E-11
2,00E-11
2,00E-11
1,70E-11
1,15E-11
7,80E-11
4,60E-12
4,00E-12
7,00E-12
1,00E-11
1,50E-11
2,00E-11
2,00E-11
1,70E-11
1,15E-11
7,80E-11
4,60E-12
Dw List:
8,00E-08
4,00E-08
8,00E-09
1,00E-09
8,31E-10
1,83E-09
6,71E-09
2,68E-08
5,86E-08
8,87E-08
1,44E-07
9,00E-08
6,00E-08
3,00E-08
5,00E-08
8,00E-08
4,00E-08
8,00E-09
1,00E-09
8,31E-10
1,83E-09
6,71E-09
2,68E-08
5,86E-08
8,87E-08
1,44E-07
9,00E-08
6,00E-08

12

3,00E-08
5,00E-08
8,00E-08
4,00E-08
8,00E-09
1,00E-09
8,31E-10
1,83E-09
6,71E-09
2,68E-08
5,86E-08
8,87E-08
1,44E-07
9,00E-08
6,00E-08
3,00E-08
5,00E-08
3,23E-10
6,00E-10
1,20E-09
2,60E-09
6,88E-09
2,29E-08
5,00E-08
1,14E-07
1,80E-07
4,30E-07
3,23E-10
6,00E-10
1,20E-09
2,60E-09
6,88E-09
2,29E-08
5,00E-08
1,14E-07
1,80E-07
4,30E-07
3,23E-10
6,00E-10
1,20E-09
2,60E-09
6,88E-09
2,29E-08
5,00E-08
1,14E-07
1,80E-07
4,30E-07

Capillary List:

6,87E+08
5,06E+08
2,61E+08
6,67E+07
1,24E+07
4,19E+06
1,38E+06
3,94E+05
2,35E+05
1,91E+05
1,62E+05
1,13E+05
1,28E+04
3,93E+03
3,07E+02
6,53E+08
4,81E+08
2,48E+08
6,34E+07
1,18E+07
3,99E+06
1,32E+06
3,74E+05
2,24E+05
1,82E+05
1,54E+05
1,07E+05
1,22E+04
4,79E+03
2,92E+02
6,19E+08
4,57E+08
2,35E+08
6,10E+07
1,12E+07
3,78E+06
1,25E+06
3,55E+05
2,12E+05
1,73E+05

13

1,46E+05
1,02E+05
1,15E+04
4,54E+03
2,77E+02
1,024E+09
3,823E+07
3,075E+06
1,794E+06
5,822E+05
1,138E+05
3,936E+04
1,36E+04
8,199E+03
3,075E+03
9,75E+08
3,636E+07
2,924E+06
1,706E+06
5,537E+05
1,082E+05
3,743E+04
1,296E+04
7,798E+03
2,92E+03
9,257E+08
3,449E+07
2,774E+06
1,618E+06
5,252E+05
1,026E+05
3,55E+04
1,23E+04
7,397E+03
2,744E+03

Conduct List:

14

1,35E-01
1,38E-01
1,42E-01
1,55E-01
1,94E-01
2,96E-01
3,14E-01
3,28E-01
3,34E-01
3,39E-01
3,34E-01
3,62E-01
3,87E-01
4,10E-01
4,19E-01
1,35E-01
1,38E-01
1,42E-01
1,55E-01
1,94E-01
2,96E-01
3,14E-01
3,28E-01
3,34E-01
3,39E-01
3,34E-01
3,62E-01
3,87E-01
4,10E-01
4,19E-01
1,35E-01
1,38E-01
1,42E-01
1,55E-01
1,94E-01
2,96E-01
3,14E-01
3,28E-01
3,34E-01
3,39E-01
3,34E-01
3,62E-01
3,87E-01
4,10E-01
4,19E-01
9,86E-01
1,034E+00
1,081E+00
1,176E+00
1,386E+00
1,532E+00
1,688E+00

```

1,844E+00
1,922E+00
2,00E+00
9,86E-01
1,037E+00
1,088E+00
1,189E+00
1,424E+00
1,568E+00
1,712E+00
1,856E+00
1,928E+00
2,00E+00
9,86E-01
1,035E+00
1,084E+00
1,083E+00
1,446E+00
1,60E+00
1,73E+00
1,87E+00
1,933E+00
2,00E+00
Moisture storage function:
15 Point list:
5
3
16 0
10
70
90
100
17 0
0,015
0,035
0,075
0,259
16 0
70
100
17 0
0,004
0,009
End

```

Legenda

- 1 Número de materiais na base de dados
- 2 Nome do material
- 3 Massa volúmica seca (kg/m³)
- 4 Calor específico (J/kg.K)
- 5 Humidade higroscópica (kg/kg)
- 6 Humidade de Saturação (kg/kg)
- 7 Número de valores de temperatura tabelados
- 8 Número de valores de teor de humidade tabelados
- 9 Valores de temperatura tabelados
- 10 Valores de teor de humidade tabelados
- 11 Valores do coeficiente de difusividade higtotérmica (m²/s.K)
- 12 Valores do coeficiente de difusividade hídrica (m²/s)
- 13 Valores da pressão capilar (N/m²)
- 14 Valores da condutibilidade térmica (W/m.K)
- 15 Número de pontos tabelados da função que define o comportamento higroscópico do material
- 16 Valores da humidade relativa (%)
- 15 Valores do Teor de humidade (kg/kg)

A.5 Ficheiros de clima (*.wet)

È possível também guardar e ler ficheiros de definição de clima, independentes de qualquer projecto. Estes ficheiros, tal como os de base de dados de materiais, servem para arquivar informações que podem ser utilizadas em diferentes simulações. Estes ficheiros poderão ser em qualquer altura carregados pelo programa e utilizados na simulação.

Os ficheiros de clima que o programa possuiu actualmente são de dois tipos:

- Ficheiros tipo A - Ficheiros para teste de sensibilidade de factores, com duração de tempo variável durante o qual a quase totalidade dos parâmetros se mantém constante.
- Tipo B - Ficheiros de aproximação à realidade, com duração de um dia (24 horas) que por repetição ao longo da simulação, descrevem as condicionantes climáticas.

Este é um exemplo de um ficheiro tipo A, em que o único parâmetro variável é o passo da simulação.

```
1 7
2 Number ;time ;step ; Tempout ;HRout ;Rad ;Rain ;Tempin ;HRin ;
1 ; 72 ; 0,01 ; 25 ; 50 ; 0 ; 1 ; 25 ; 50
2 ; 444 ; 0,01 ; 25 ; 50 ; 0 ; 0 ; 25 ; 50
3 ; 144 ; 0,05 ; 25 ; 10 ; 0 ; 0 ; 25 ; 50
4 ; 144 ; 0,1 ; 25 ; 10 ; 0 ; 0 ; 25 ; 50
5 ; 144 ; 0,5 ; 25 ; 10 ; 0 ; 0 ; 25 ; 50
6 ; 144 ; 0,5 ; 25 ; 10 ; 0 ; 0 ; 25 ; 50
7 ; 576 ; 1 ; 25 ; 10 ; 0 ; 0 ; 25 ; 50
```

A	B	C	D	E	F	G	H	I
Number	time	step	Tempout	HRout	Rad	Rain	Tempin	HRin

Legenda

- 1 Numero de linhas no ficheiro de clima
- 2 Parâmetros de definição do clima
- A Identificação do número da linha do ficheiro
- B Tempo em que as condições climáticas são mantidas (h)
- C Intervalo de tempo (h)
- D Temperatura exterior (°C)
- E Humidade relativa exterior (%)
- F Radiação solar em plano horizontal (W/m²)
- G Ocorrência de precipitação (1 – Sim / 0 – Não)
- H Temperatura interior (°C)
- I Humidade relativa interior (%)

Este é um exemplo de um ficheiro tipo B, que reproduz as condicionantes climáticas ao longo de 24 horas.

```
1 7
2 Number ;time ;step ; Tempout ;HRout ;Rad ;Rain ;Tempin ;HRin ;
1 ; 7 ; 0,01 ; 25 ; 50 ; 0 ; 0 ; 25 ; 50
2 ; 1 ; 0,01 ; 25 ; 50 ; 34 ; 0 ; 25 ; 50
3 ; 1 ; 0,01 ; 25 ; 50 ; 140 ; 0 ; 25 ; 50
4 ; 1 ; 0,01 ; 25 ; 50 ; 230 ; 0 ; 25 ; 50
5 ; 1 ; 0,01 ; 25 ; 50 ; 292 ; 0 ; 25 ; 50
6 ; 1 ; 0,01 ; 25 ; 50 ; 316 ; 0 ; 25 ; 50
7 ; 1 ; 0,01 ; 25 ; 50 ; 297 ; 0 ; 25 ; 50
8 ; 1 ; 0,01 ; 25 ; 50 ; 226 ; 0 ; 25 ; 50
9 ; 1 ; 0,01 ; 25 ; 50 ; 159 ; 0 ; 25 ; 50
10 ; 1 ; 0,01 ; 25 ; 50 ; 44 ; 0 ; 25 ; 50
11 ; 8 ; 0,01 ; 25 ; 50 ; 0 ; 0 ; 25 ; 50
```

B.1.	DEFINIÇÃO DO PROGRAMA	127
B.2.	LISTAGEM DE “MODULES”	128
1.	Módulo “trh98.bas”	128
2.	Módulo “functions.bas”	130
3.	Módulo “n_functions.bas”	135
4.	Módulo “o_functions.bas”	140
5.	Módulo “DataFunctions.bas”	147
6.	Módulo “conv.bas”	158
B.3.	LISTAGEM DE “FORMS”	161
1.	“Form início”	162
2.	“Form frmmodules “	163
3.	“Form frmsat”	164
4.	“Form frmmoisture”	166
5.	“Form frmheat”	168
6.	“Form frmtemp”	170
7.	“Form frmsol”	171
8.	“Form frmconvert”	173
9.	“Form frmmain”	176
10.	“Form frmminwindow”	182
11.	“Form frmobjectname”	189
	“Form frmobjectdefinition”	190
13.	“Form frmtime-r”	194
14.	“Form frminitialc”	196
15.	“Form frminitialprofil”	197
16.	“Form frmmainweather”	200
17.	“Form frmsteadyweather”	202
18.	“Form frmdynamicweather”	203
19.	“Form frmimport”	207
20.	“Form material”	209
21.	“Form frmreview”	210
22.	“Form frmresults”	214

Anexo B Listagem parcial do programa

B.1 Definição do programa

```
Type=Exe
Reference=*G{00020430-0000-0000-C000-000000000046}#2.0#0#C:\WINDOWS\SYSTEM\StdOle2.tlb#OLE Automation

Object={F9043C88-F6F2-101A-A3C9-08002B2F49FB}#1.1#0; COMDLG32.OCX
Object={6B7E6392-850A-101B-AFC0-4210102A8DA7}#1.1#0; COMCTL32.OCX
Object={02B5E320-7292-11CF-93D5-0020AF99504A}#1.0#0; MSCHART.OCX
Object={A8B3B723-0B5A-101B-B22E-00AA0037B2FC}#1.0#0; GRID32.OCX
Object={3B7C8863-D78F-101B-B9B5-04021C009402}#1.1#0; RICHTX32.OCX
Object={00028C01-0000-0000-0000-000000000046}#1.0#0; DBGRID32.OCX

Module=functions; functions.bas
Module=trh98; tr981.bas
Module=globaledit; GlobalEdit.bas
Module=DataFunctions; data.bas
Module=n_functions; n_functions.bas
Module=o_functions; o_functions.bas
Module=conv; conversions.bas

Form=frmmain.frm
Form=inicio.frm
Form=initialc.frm
Form=mainw.frm
Form=mainwea.frm
Form=objecdef.frm
Form=objectn.frm
Form=steadywe.frm
Form=weather.frm
Form=gweather.frm
Form=frmimport.frm
Form=time-r.frm
Form=frmweath.frm
Form=review.frm
Form=formmaterial.frm
Form=frmSat.frm
Form=frmmodules.frm
Form=frmmoisture.frm
Form=frmconvert.frm
Form=frmheat.frm
Form=frmsol.frm
Form=frmtemp.frm
Form=frmresults.frm
Form=frmabout.frm
Form=frmhelp.frm
Form=frminitialprofil.frm

IconForm="FrmInicio"
Startup="FrmInicio"
HelpFile=""
Title="TrHum 98"
ExeName32="TrHum98.exe"
Command32=""
Name="TrHum98"
HelpContextID="0"
CompatibleMode="0"
MajorVer=1
MinorVer=0
RevisionVer=0
AutoIncrementVer=0
ServerSupportFiles=0
VersionCompanyName="LFC - FEUP"
VersionFileDescription="Numerical Simulation of Heat and Moisture Transfer in Building Materials"
VersionLegalCopyright="Clara Pimenta do Vale"
VersionProductName="TrHum 98"
CompilationType=0
OptimizationType=0
FavorPentiumPro(tm)=0
CodeViewDebugInfo=0
NoAliasing=0
BoundsCheck=0
OverflowCheck=0
FlPointCheck=0
FDIVCheck=0
UnroundedFP=0
StartMode=0
```

```

Unattended=0
ThreadPerObject=0
MaxNumberOfThreads=1

```

B.2 Listagem de “Modules”

B.2.1. Módulo “trh98.bas”

```

Option Explicit
'-----
'global Variable declaration
'-----
Public line$
Public Colum(12) As Double
Public wetline As Integer
Public Filename As String
Public FilenameTmp As String
Public FilenameLng As String
Public message As String
Public complete As Boolean
Public Errorr As Double
Public previousform As String
Public nextform As String
'-----
'time variables
'-----
Public gTotalTime As Double 'Project Total Time
Public gYears As Double 'Starting Year
Public gDayS As Double 'Starting Day of the Year
Public gHourS As Double 'Starting Hour
Public gTotalWeather As Double 'Total time of the Weather File
Public gTotalSteps As Double 'Number of Time Intervals
Public gWeatherFileType ' Type of the original weather file
Public gDateS As Date 'Date Start
Public gDateF As Date 'Date Finish
Public gHoursS As Date 'Hour start
Public gHoursF As Date 'Hour finish
'-----
'weather variables
'-----
Public gN As Integer 'number of steps in weather file
Public gnumber(1000) As Integer
Public gTime(1000) As Double 'time in hours
Public gstep(1000) As Double 'time step in hours
Public gRain(1000) As Single 'Weather rain yes or no
Public gTempIn(1000) As Single 'indoor temperature
Public gTempOut(1000) As Single 'outdoor temperature
Public grhin(1000) As Single 'indoor relative humidity
Public grhout(1000) As Single 'outdoor relative humidity
Public grad(1000) As Single 'radiation
Public Nint(1000)
Public Wetfilename As String
Public wstep As Double
'-----
'initial conditions variables
'-----
Public gHe As Double 'outdoor superficial conductance
Public gHI As Double 'indoor superficial conductance
Public gBetaI As Double 'indoor moisture coefficient
Public gBetaE As Double 'outdoor moisture coefficient
Public gErrorMax As Double 'maximum error
Public galpha As Double 'Absorption coeficgTotalNodesnt for radiation
Public gLatitude As Single
Public gLongitude As Single
Public gNorS As Boolean 'latitude North or South
Public gEorW As Boolean 'longitude east or west
Public gWindDirection As Single
Public gWindSpeed As Single
Public gAzimute As Single
Public gCity As String
Public gCountry As String
'-----
'initial Profiles variables
'-----
Public gTempStart(3) 'initial Temperature
Public gRHStart(3) 'initial relative humidity
'-----
'object definition variables

```

```

'-----
Public gObjName           As String
Public gObjDescription    As String
Public gObjNotes         As String

Public gNumberOfLayers   As Integer   ' total number of layers
Public gLayerNumber(3)   As Integer   ' presente layer number
Public gThickness(3)     As Double    ' thickness os presente layer
Public gSStep(3)         As Integer   ' number of steps in present layer
Public.gdx(3)            As Double    ' dx used in present layer
Public dx(51) As Double   'dx for each step
Public gmaterialname     As String    ' material name of present layer
Public gmaterialcode     As Integer   'code for identify the material
Public gthickness       As Double    'material total thickness
'material data bank variables
'-----
Public gmlistnumber As Double
Public gmatname(10) As String
Public gdensity(10) As Double
Public gheat(10) As Double
Public ghygroskop(10) As Double
Public gsaturation(10) As Double
Public gtlist(10) As Double
Public gmlist(10) As Double
Public gtemperature(10, 5) As Double
Public gmoisture(10, 20) As Double
Public gdt(10, 5, 20) As Double
Public gdw(10, 5, 20) As Double
Public gcapillar(10, 5, 20) As Double
Public gconduct(10, 5, 20) As Double
Public gSorpCount(10) As Integer
Public GSorpHR(10, 10) As Double
Public GSorpWHR(10, 10) As Double
'material variables
'-----
Public gsdensity As Double
Public gsheat As Double
Public gshygroskop As Double
Public gssaturation As Double
Public gstlist As Double
Public gsmlist As Double
Public gstemperature(5) As Double
Public gsmoisture(20) As Double
Public gsdt(5, 20) As Double
Public gsdw(5, 20) As Double
Public gscapillar(5, 20) As Double
Public gsconduct(5, 20) As Double
Public gSSorpCount As Integer
Public gSSorpHR(10) As Double
Public gSSorpWHR(10) As Double
Public teste As String
Public Table(25, 25) As Double
Public tipo As Integer
'calculation variables
'-----
Public gtStep As Integer   'total number of steps
Public gEndingNode(3) As Integer 'number of layer starting node
Public gTotalNodes As Integer 'total calculation nodes
Public gNTempStart(61) As Double 'node initial Temperature
Public gNRHStart(61) As Double 'node initial relative humidity
Public IT As Integer 'iteration as int
Public gClock As Double ' Clock for measuring time
Public LLayers As Integer
Public gmoist(61, 30) As Double 'Moisture profils
Public gtemp(61, 30) As Double 'Temperature profiles
Public DTnw(61) As Double
Public DWnw(61) As Double
Public CAPInow(61) As Double
Public Condnw(61) As Double
Public t(61, 1010) As Double 'temperature over time step
Public w(61, 1010) As Double 'moisture over time step
Public ta(61, 50) As Double 'temperature over iterations
Public wa(61, 50) As Double 'moisture over iterations
Public st As Double 'steps of profils
Public WF(61) As Double
Public TF(61) As Double
Public chour As Double ' clock hour
Public cday As Integer 'clock day
Public cyear As Integer 'clock year
Public TSOL As Double
Public Tempoutfic As Double
Public Temp As Double
Public hum As Double
Public Rad_ As Double
Public temp_out As Double

```



```

Public temp_in As Double
Public RH_out As Double
Public RH_in As Double
Public Rain_ As Single
Public pos As Single
Public time_profile(25) As Single
Public nprofile As Integer
Public gtdx As Double
Public AA(61, 61) As Double
Public BB(61) As Double
Public b(61) As Double
Public lv As Double
Public MX As Double
Public anotheriteration As Boolean
Public result(61) As Double
Public distance(61) As Double
Public genweather As Boolean

```

B.2.2. Módulo “functions.bas”

```
Option Explicit
```

```
Public fMainForm As frmMain
```

```

Static Function Log10(x)
Log10 = Log(x) / Log(10#)
End Function

```

```

Static Function convert_point$(v As Double)
Dim aux1 As Integer
convert_point$ = LTrim$(Str$(v))
For aux1 = 1 To Len(convert_point$)
    If Mid$(convert_point$, aux1, 1) = "," Then
        Mid$(convert_point$, aux1, 1) = "."
    End If
Next
If Mid$(convert_point$, 1, 1) = "." Then
    convert_point$ = "0" + convert_point$
End If
End Function

```

```

Static Function txt2double(v As String) As Double
Dim aux1 As Integer
v = LTrim$(v)
For aux1 = 1 To Len(v)
    If Mid$(v, aux1, 1) = "," Then
        Mid$(v, aux1, 1) = "."
    End If
Next
txt2double = Val(v)
End Function

```

```

Public Sub NNODES()
Dim aux1 As Integer
gEndingNode(1) = 0
gEndingNode(2) = 0
gEndingNode(3) = 0
gtStep = 0
For aux1 = 1 To gNumberOFLayers
    gtStep = gtStep + gSStep(aux1)
    gEndingNode(aux1) = gtStep + 1
Next aux1
gTotalNodes = gtStep + 1
End Sub

```

```

Public Sub calcdx()
Dim aux1 As Integer
Dim aux2 As Integer
Dim ii As Integer
Dim sum As Double
ii = 0
gtStep = 0
For aux1 = 1 To gNumberOFLayers
    gtStep = gtStep + gSStep(aux1)
    For aux2 = 1 To gSStep(aux1)
        ii = ii + 1
        dx(ii) = gdx(aux1)
    Next aux2
Next aux1
sum = 0
distance(1) = 0
For ii = 2 To gTotalNodes

```

```

        sum = sum + dx(ii - 1)
        distance(ii) = sum
Next ii
End Sub

```

```

Public Sub startprofiles()
Dim aux3 As Integer
Dim aux1 As Integer
Dim aux2 As Integer
aux3 = 1
For aux1 = 1 To gNumberOfLayers
    For aux2 = aux3 To gEndingNode(aux1)
        gNTempStart(aux2) = gTempStart(aux1)
        gNRHStart(aux2) = gRHStart(aux1)
    Next aux2
    aux3 = gEndingNode(aux1) + 1
Next aux1
Llayers = gNumberOfLayers
End Sub

```

```

Public Sub initialprofil()
Dim i As Integer
For i = 1 To gTotalNodes
    gtemp(i, 0) = gNTempStart(i)
    gmoist(i, 0) = gNRHStart(i)
    time_profile(0) = 0
Next i
End Sub

```

```

Public Sub extractcolum()
Dim FgTotalNodeslds$
FgTotalNodeslds$ = line$
Dim extract(20, 2) As Integer
Dim ps As Integer
Dim f As Integer
Dim nfgTotalNodeslds
Dim aux1 As Integer
For aux1 = 1 To Len(FgTotalNodeslds$)
    If Mid$(FgTotalNodeslds$, aux1, 1) = "," Then Mid$(FgTotalNodeslds$, aux1, 1) = "."
Next
ps = 1
f = 0
extract(1, 1) = 1
Do
    f = f + 1
    extract(f, 1) = ps
    ps = InStr(ps, FgTotalNodeslds$, ";")
    extract(f, 2) = ps - extract(f, 1)
    ps = ps + 1
Loop While ps <> 1
extract(f, 2) = Len(FgTotalNodeslds$) + 1 - extract(f, 1)
nfgTotalNodeslds = f
For aux1 = 1 To nfgTotalNodeslds
    Colum(aux1) = Val(Mid$(FgTotalNodeslds$, extract(aux1, 1), extract(aux1, 2)))
Next
End Sub

```

```

Public Sub passweather96() 'trhum 96
    gTime(wetline) = Colum(1)
    gTempOut(wetline) = Colum(2)
    gRHOut(wetline) = Colum(3)
    grad(wetline) = Colum(4)
    gTempIn(wetline) = Colum(5)
    gRHIn(wetline) = Colum(6)
    gstep(wetline) = Colum(7)
    gRain(wetline) = Colum(8)
    Nint(wetline) = gTime(wetline) / gstep(wetline)
End Sub

```

```

Public Sub passweather98() 'trhum 98
    gnumber(wetline) = Colum(1)
    gTime(wetline) = Colum(2)
    gTempOut(wetline) = Colum(4)
    gRHOut(wetline) = Colum(5)
    grad(wetline) = Colum(6)
    gTempIn(wetline) = Colum(8)
    gRHIn(wetline) = Colum(9)
    gstep(wetline) = Colum(3)
    gRain(wetline) = Colum(7)
End Sub

```

```

Public Sub passweather90() 'trhumidade
    Nint(wetline) = Colum(1)
    gTempOut(wetline) = Colum(2)
    gRHOut(wetline) = Colum(3)
    grad(wetline) = Colum(4)

```

```

    gTempIn(wetline) = Colum(5)
    gRHIn(wetline) = Colum(6)
    gstep(wetline) = Colum(8)
    gRain(wetline) = Colum(10)
    gTime(wetline) = gstep(wetline) * Nint(wetline)
End Sub

```

```

Public Sub tpassvalue()
Dim i As Integer
Dim j As Integer
Select Case tipo
Case 1
    For i = 0 To gSTLIST
        For j = 0 To gSMLIST
            Table(i, j) = gSDT(i, j)
        Next
    Next
Case 2
    For i = 0 To gSTLIST
        For j = 0 To gSMLIST
            Table(i, j) = gSDW(i, j)
        Next
    Next
Case 4
    For i = 0 To gSTLIST
        For j = 0 To gSMLIST
            Table(i, j) = gSCONDUCT(i, j)
        Next
    Next
Case 3
    For i = 0 To gSTLIST
        For j = 0 To gSMLIST
            Table(i, j) = gSCAPILLAR(i, j)
        Next
    Next
Case 11
    For i = 0 To gSTLIST
        For j = 0 To gSMLIST
            gSDT(i, j) = Table(i, j)
        Next
    Next
Case 12
    For i = 0 To gSTLIST
        For j = 0 To gSMLIST
            gSDW(i, j) = Table(i, j)
        Next
    Next
Case 14
    For i = 0 To gSTLIST
        For j = 0 To gSMLIST
            gSCONDUCT(i, j) = Table(i, j)
        Next
    Next
Case 13
    For i = 0 To gSTLIST
        For j = 0 To gSMLIST
            gSCAPILLAR(i, j) = Table(i, j)
        Next
    Next
End Select
End Sub

```

```

Public Sub validation() 'verify if all the values needed for calculation are entered
Dim aux1 As Integer
Dim i As Integer
Dim ii As Integer

complete = True
'object definition
If gNumberOfLayers = 0 Or gthickness = 0 Then
    complete = False
    message = "error in object definition"
    messagebox
End If
For aux1 = 1 To gNumberOfLayers
    If gLayerNumber(aux1) = 0 Then
        complete = False
        message = "error in object definition"
    End If
    If gThickness(aux1) = 0 Then
        complete = False
        message = "error in object definition"
        messagebox
    End If
    If gdx(aux1) = 0 Then
        complete = False
        message = "error in object definition"
    End If

```

```

    messagebox
End If
If gSStep(aux1) = 0 Then
    complete = False
    message = "error in object definition"
    messagebox
End If
If gTempStart(aux1) = Empty Or gRHStart(aux1) = Empty Then
    complete = False
    message = "error in initial conditions - profil"
    messagebox
End If
Next
'time variables
If Not IsDate(gDateS) Then
    complete = False
    message = "error in time calculation"
    messagebox
End If
If Not IsDate(gDateF) Then
    complete = False
    message = "error in time calculation"
    messagebox
End If
If Not IsDate(gHoursS) Then
    complete = False
    message = "error in time calculation"
    messagebox
End If
If Not IsDate(gHoursF) Then
    complete = False
    message = "error in time calculation"
    messagebox
End If
'weather variables
For i = 1 To gN
    If gnumber(i) = Empty Then
        complete = False
        message = "error in weather"
        messagebox
    End If
Next
'initial conditions variables
If gHe < 0 Or gHI < 0 Or gBetaI < 0 Or gBetaE < 0 Then
    complete = False
    message = "error in initial conditions"
    messagebox
ElseIf gErrorMax = 0 Then
    complete = False
    message = "error in initial conditions"
    messagebox
ElseIf galpha = 0 Then
    complete = False
    message = "error in initial conditions"
    messagebox
ElseIf gLatitude = Empty Or gLongitude = Empty Then
    complete = False
    message = "error in initial conditions"
    messagebox
ElseIf gWindDirection < 0 Or gWindSpeed < 0 Or gAzimute < 0 Then
    complete = False
    message = "error in initial conditions"
    messagebox
End If
'material variables
If gSDENSITY = 0 Or gSHEAT = 0 Or gSHYGROSCOP = 0 Or gSSATURATION = 0 Then
    complete = False
    message = "error in material definition - density ...."
    messagebox
ElseIf gSTLIST = 0 Or gSMLIST = 0 Then
    complete = False
    message = "error in material definition - list ...."
    messagebox
End If
For i = 1 To gSTLIST
    If gSTEMPERATURE(i) = Empty Then
        complete = False
        message = "error in material definition - temperature ..."
        messagebox
    End If
    For ii = 1 To gSMLIST
        If gSMOISTURE(ii) = Empty Then
            complete = False
            message = "error in material definition - moisture ..."
            messagebox
        End If
        If gSDT(i, ii) = Empty Or gSDW(i, ii) = Empty Or gSCAPILLAR(i, ii) = Empty Or gSCONDUCT(i, ii) = 0
Then

```

```

        complete = False
        message = "error in material definition - dt,dw,cap,suc ....."
        messagebox
    End If
Next
Next
End Sub

```

```

Public Sub messagebox() 'general
    Const MB_OK = 0
    Const MB_OKCANCEL = 1
    Const MB_ABORTRETRYIGNORE = 2
    Const MB_YESNOCANCEL = 3
    Const MB_YESNO = 4
    Const MB_RETRYCANCEL = 5
    Const MB_ICONSTOP = 16
    Const MB_ICONQUESTION = 32
    Const MB_ICONEXCLAMATION = 48
    Const MB_ICONINFORMATION = 64
    Const IDOK = 1
    Const IDCANCEL = 2
    Const IDABORT = 3
    Const IDRETRY = 4
    Const IDIGNORE = 5
    Const IDYES = 6
    Const IDNO = 7
    Dim Buttons As Integer
    Dim Title As String
    Dim Answer As Integer
    Dim x As Integer
    Dim Y As Integer
    Buttons = MB_OK + MB_ICONEXCLAMATION
    Answer = MsgBox(message, Buttons, "TRHUM 98")
End Sub

```

```

Public Sub creatProjectdatabank() 'trhum 98
Dim i As Integer
Dim ii As Integer
Dim iii As Integer
i = gmaterialcode
gSDENSITY = gDENSITY(i)
gSHEAT = gHEAT(i)
gSHYGROSCOP = gHYGROSCOP(i)
gSSATURATION = gSATURATION(i)
gSTLIST = gTLIST(i)
gSMLIST = gMLIST(i)
For ii = 1 To gSTLIST
    gSTEMPERATURE(ii) = gTEMPERATURE(i, ii)
Next ii
For ii = 1 To gSMLIST
    gSMOISTURE(ii) = gMOISTURE(i, ii)
Next ii
For ii = 0 To gSTLIST
    For iii = 0 To gSMLIST
        gSDT(ii, iii) = gDT(i, ii, iii)
        gSDW(ii, iii) = gDW(i, ii, iii)
        gSCAPILLAR(ii, iii) = gCAPILLAR(i, ii, iii)
        gSCONDUCT(ii, iii) = gCONDUCT(i, ii, iii)
    Next iii
Next ii
gSSorpCount = gSorpCount(i)
For ii = 1 To gSSorpCount
    gSSorpHR(ii) = GSorpHR(i, ii)
    gSSorpWHR(ii) = GSorpWHR(i, ii)
Next ii
ordering2
zeroundredcorrection
tipo = 1
tpassvalue
ordering
tipo = 11
tpassvalue
tipo = 2
tpassvalue
ordering
tipo = 12
tpassvalue
tipo = 3
tpassvalue
ordering
tipo = 13
tpassvalue
tipo = 4
tpassvalue
ordering
tipo = 14
tpassvalue

```

```

End Sub

Public Sub cleanMatrix()
Dim ttt As Integer
Dim sss As Integer
For ttt = 0 To 2 * gTotalNodes + 4
    For sss = 0 To 2 * gTotalNodes + 4
        AA(ttt, sss) = 0
    Next
Next
b(ttt) = 0
Next
End Sub

```

B.2.3. Módulo “n_functions.bas”

Option Explicit

```

Public Sub startclean()
Dim ii As Integer
Dim iii As Integer
For ii = 1 To 61
    For iii = 1 To 30
        gtemp(ii, iii) = 0
        gmoist(ii, iii) = 0
    Next
Next
For iii = 1 To 1010
    w(ii, iii) = 0
    t(ii, iii) = 0
Next
WF(ii) = 0
DTnw(ii) = 0
DWnw(ii) = 0
CAPInw(ii) = 0
Condnw(ii) = 0
Next
End Sub

```

```

Public Sub ordering() 'trhum 98
'to order the values in a matrix
'number of temperature list
'number of moisture list
Dim aux1 As Integer
Dim aux2 As Integer
Dim aux3 As Integer
For aux1 = 0 To gSTLIST - 1
    For aux2 = aux1 + 1 To gSTLIST
        If Table(aux1, 0) > Table(aux2, 0) Then
            For aux3 = 0 To gSMLIST
                swap Table(aux1, aux3), Table(aux2, aux3)
            Next
        End If
    Next
Next
For aux1 = 0 To gSMLIST - 1
    For aux2 = aux1 + 1 To gSMLIST
        If Table(0, aux1) > Table(0, aux2) Then
            For aux3 = 0 To gSTLIST
                swap Table(aux3, aux1), Table(aux3, aux2)
            Next
        End If
    Next
Next
End Sub

```

```

Public Sub ordering2() 'trhum 98
Dim aux1 As Integer
Dim aux2 As Integer
Dim aux3 As Integer
For aux1 = 0 To gSSorpCount - 1
    For aux2 = aux1 + 1 To gSSorpCount
        If gSSorpHR(aux1) > gSSorpHR(aux2) Then
            swap gSSorpHR(aux1), gSSorpHR(aux2)
            swap gSSorpWHR(aux1), gSSorpWHR(aux2)
        End If
    Next
Next
End Sub

```

```

Public Sub zeroundredcorrection()
Dim i As Integer
If gSSorpHR(1) <> 0 Then
    gSSorpCount = gSSorpCount + 1
    For i = gSSorpCount To 2 Step -1
        gSSorpHR(i) = gSSorpHR(i - 1)
    Next
End Sub

```

```

        gSSorpWHR(i) = gSSorpWHR(i - 1)
    Next i
    gSSorpHR(1) = 0
    gSSorpWHR(1) = 0
End If
End Sub

```

```

Function LinearInterpolation(X, X1, Y1, X2, Y2)
    LinearInterpolation = Y1 + (X - X1) * (Y1 - Y2) / (X1 - X2)
End Function

```

```

Function Hurelat2(w)
Dim i As Integer
Dim h
i = 0
Do
i = i + 1
    If gSSorpWHR(i) > w And gSSorpWHR(i - 1) < w Then
        h = LinearInterpolation(w, gSSorpWHR(i - 1), gSSorpHR(i - 1), gSSorpWHR(i), gSSorpHR(i))
    ElseIf gSSorpWHR(i) = w Then
        h = gSSorpHR(i)
    End If
Loop While i < gSSorpCount
If w > gSSorpWHR(gSSorpCount) Then h = 100
Hurelat2 = h
End Function

```

```

Function Higroscop2(h)
Dim i As Integer
Dim w
i = 0
Do
i = i + 1
    If gSSorpHR(i) > h And gSSorpHR(i - 1) < h Then
        w = LinearInterpolation(h, gSSorpHR(i - 1), gSSorpWHR(i - 1), gSSorpHR(i), gSSorpWHR(i))
    ElseIf gSSorpHR(i) = h Then
        w = gSSorpWHR(i)
    End If
Loop While i < gSSorpCount
Higroscop2 = w
End Function

```

```

Function interpolation(Temp, hum) 'trhum 98
Dim lmin, lmax As Integer
Dim cmin, cmax As Integer
Dim L1, L2 As Double
Dim c1, c2 As Double
Dim V1, v2 As Double
Dim v As Double
Dim aux1 As Integer
If Temp = 0 And hum = 0 Then
    v = Table(1, 1)
    GoTo FINAL
End If
If Temp <= Table(1, 0) Then
    lmin = 1
    lmax = 1
ElseIf Temp >= Table(gSTLIST, 0) Then
    lmin = gSTLIST
    lmax = gSTLIST
Else
    aux1 = 1
    Do While Table(aux1, 0) < Temp
        aux1 = aux1 + 1
    Loop
    If Table(aux1, 0) = Temp Then
        lmax = aux1
        lmin = aux1
    Else
        lmin = aux1 - 1
        lmax = aux1
    End If
End If
If hum <= Table(0, 1) Then
    cmin = 1
    cmax = 1
ElseIf hum >= Table(0, gSMLIST) Then
    cmin = gSMLIST
    cmax = gSMLIST
Else
    aux1 = 1
    Do While Table(0, aux1) < hum
        aux1 = aux1 + 1
    Loop
    If Table(0, aux1) = hum Then
        cmax = aux1
        cmin = aux1
    End If
End If

```

```

Else
    cmin = aux1 - 1
    cmax = aux1
End If
End If
If lmin = lmax Then
    V1 = Table(lmin, cmin)
    v2 = Table(lmin, cmax)
Else
    L1 = Temp - Table(lmin, 0)
    L2 = Table(lmax, 0) - Temp
    V1 = (Table(lmin, cmin) * L1 + Table(lmax, cmin) * L2) / (L1 + L2)
    v2 = (Table(lmin, cmax) * L1 + Table(lmax, cmax) * L2) / (L1 + L2)
End If
If cmin = cmax Then
    v = V1
Else
    c1 = hum - Table(0, cmin)
    c2 = Table(0, cmax) - hum
    v = (V1 * c1 + v2 * c2) / (c1 + c2)
End If
GoTo FINAL
FINAL:
interpolation = v
End Function

```

```

Sub Gauss()
Dim m As Double
Dim m1 As Double
Dim m2 As Double
Dim nn As Integer
Dim K As Integer
Dim ii As Double
Dim j As Integer
Dim FoundPivot As Boolean
Dim iii As Double
nn = 2 * gTotalNodes
For ii = 1 To nn
    For j = 1 To nn
        AA(ii, j) = AA(ii, j) * 100000000
    Next
    b(ii) = BB(ii) * 100000000
Next
For ii = 1 To nn
    If AA(ii, ii) = 0 Then
        If AA(ii + 1, ii) <> 0 Then
            For j = 1 To nn
                swap AA(ii, j), AA(ii + 1, j)
            Next j
            swap b(ii), b(ii + 1)
        Else
            FoundPivot = False
            j = 0
            Do
                j = j + 1
                If j > nn Then Print #3, "error": Stop
                If AA(j, ii) <> 0 Then
                    For K = 1 To nn
                        swap AA(ii, K), AA(j, K)
                    Next K
                    swap b(ii), b(j)
                    FoundPivot = True
                End If
            Loop Until (FoundPivot)
        End If
    End If
Next
For K = 1 To nn - 1
    For ii = K + 1 To nn
        m1 = AA(ii, K)
        m2 = AA(K, K)
        m = m1 / m2
        For j = K To nn
            AA(ii, j) = AA(ii, j) - m1 / m2 * AA(K, j)
        Next
        b(ii) = b(ii) - m1 / m2 * b(K)
    Next
Next
Dim Helpx As Double
Dim X(50) As Double
X(nn) = b(nn) / AA(nn, nn)
For K = nn - 1 To 1 Step -1
    Helpx = 0
    For j = K + 1 To nn
        Helpx = Helpx + AA(K, j) * X(j)
    Next
    X(K) = (b(K) - Helpx) / AA(K, K)

```



```

Next
For iii = 1 To nn
    b(iii) = X(iii)
Next iii
End Sub

```

```

Sub printproject()
Dim K As Integer
Dim Header, i, Y
Dim ii As Integer
Dim iii As Integer
Dim c As Integer
Header = "Project Definitions - Page " ' Set header string.
Printer.FontName = "arial"
Printer.FontBold = False
Printer.FontSize = 9
Printer.Print Header; ' Print header.
Printer.Print Printer.Page ' Print page number.
Y = Printer.CurrentY + 10 ' Set position for line.
Printer.Line (0, Y)-(Printer.ScaleWidth, Y) ' Draw line.
On Error Resume Next
Printer.Print String(10, " "); "Project Name: " , gObjName
Printer.Print String(10, " "); "Project Description: " , gObjDescription
Printer.Print String(10, " "); "Project Notes: " , gObjNotes
Printer.Print String(10, " "); "Material Name: " , gMATERIALNAME
Printer.Print String(10, " "); "Material Code: " , gmaterialcode
Printer.Print String(10, " "); "Total Thickness: " , gtthickness
Printer.Print String(10, " "); "Number of Layers: " , gNumberOFLayers
Printer.Print String(10, " "); "Layer Number 1: " , gLayerNumber(1)
Printer.Print String(10, " "); "Thickness layer 1: " , gThickness(1)
Printer.Print String(10, " "); "N. of Steps layer 1: " , gSStep(1)
Printer.Print String(10, " "); "Step of Layer 1: " , gdx(1)
Printer.Print String(10, " "); "Layer Number 2: " , gLayerNumber(2)
Printer.Print String(10, " "); "Thickness layer 2: " , gThickness(2)
Printer.Print String(10, " "); "N. of Steps layer 2: " , gSStep(2)
Printer.Print String(10, " "); "Step of Layer 2: " , gdx(2)
Printer.Print String(10, " "); "Layer Number 3: " , gLayerNumber(3)
Printer.Print String(10, " "); "Thickness layer 3: " , gThickness(3)
Printer.Print String(10, " "); "N. of Steps layer 3: " , gSStep(3)
Printer.Print String(10, " "); "Step of Layer 3: " , gdx(2)
Dim n As Integer
Printer.Print String(10, " ")
Printer.Print String(10, " "); "Project start date: " , gDateS, gHoursS
Printer.Print String(10, " "); "Project Finish date: " , gDateF, gHoursF
Printer.Print String(10, " "); "Total Calculation Time: " , gTotalTime
Printer.Print String(10, " "); "Total weather file Time: " , gTotalWeather
Printer.Print String(10, " "); "Number of profiles: " , nprofile
Printer.Print String(10, " "); "Time of the profiles: " , ;
For n = 1 To nprofile
    Printer.Print String(1, " "); time_profile(n); "h";
Next
Printer.Print String(10, " ")
Printer.Print String(10, " ")
Printer.Print String(10, " "); "Indoor superficial conductance: " , gHI
Printer.Print String(10, " "); "Outdoor superficial conductance: " , gHe
Printer.Print String(10, " "); "Indoor moisture coefficient: " , gBetaI
Printer.Print String(10, " "); "Outdoor moisture coefficient: " , gBetaE
Printer.Print String(10, " "); "Maximum error: " , gErrorMax
Printer.Print String(10, " "); "City: " , gCity
Printer.Print String(10, " "); "Country: " , gCountry '#71
Printer.Print String(10, " "); "Latitude: " , gLatitude
Printer.Print String(10, " "); "North or South: " , gNorS
Printer.Print String(10, " "); "Longitude: " , gLongitude
Printer.Print String(10, " "); "East or West: " , gEorW
Printer.Print String(10, " "); "Wind direction: " , gWindDirection '#81
Printer.Print String(10, " "); "Wind speed: " , gWindSpeed
Printer.Print String(10, " "); "Azimut (perp.to surface): " , gAzimute
Printer.Print String(10, " "); "Absorption coef.for radiation: " , galpha
Printer.Print String(10, " ")
Printer.Print String(10, " "); "Initial Profile of temperature: " , gTempStart(1); ";"; gTempStart(2); ";";
gTempStart(3)
Printer.Print String(10, " "); "Initial profile of moisture: " , gRHStart(1); ";"; gRHStart(2); ";";
gRHStart(3)
Printer.NewPage
Printer.Print Header; ' Print header.
Printer.Print Printer.Page ' Print page number.
Y = Printer.CurrentY + 10 ' Set position for line.
Printer.Line (0, Y)-(Printer.ScaleWidth, Y) ' Draw line.
Printer.Print String(10, " ")
Printer.Print String(10, " "); "Begining of the Weather file"
Printer.Print String(10, " "); "Number of time intervals: " , gTotalSteps
Printer.Print String(10, " "); "Number ;"; "time ;"; "step ;"; "Tempout ;"; "HRout ;"; "Rad ;"; "Rain ;";
"Tempin ;"; "HRin ;" '#100
For c = 1 To gTotalSteps

```

```

Printer.Print String(10, " "); gnumber(c); ";"; gTime(c); ";"; gstep(c); ";"; gTempOut(c); ";"; gRHOut(c);
";"; grad(c); ";"; gRain(c); ";"; gTempIn(c); ";"; gRHIn(c) '
Next
Printer.Print String(10, " "); "End of weather file"
Printer.Print String(10, " "); "Begining of the Material Data Bank"
Printer.Print String(10, " "); "Material name:          ", gMATERIALNAME
Printer.Print String(10, " "); "Material code:         ", gmaterialcode
Printer.Print String(10, " "); "Density:                ", gSDENSITY
Printer.Print String(10, " "); "Heat:                   ", gSHEAT
Printer.Print String(10, " "); "Hygroskop:             ", gSHYGROSCOP
Printer.Print String(10, " "); "Saturation:           ", gSSATURATION
Printer.Print String(10, " "); "Number of Temperature: ", gSTLIST
Printer.Print String(10, " "); "Number of moisture list: ", gSMLIST
Printer.Print String(10, " "); "Temperature list:"
For ii = 1 To gSTLIST
    Printer.Print String(1, " "); gSTEMPERATURE(ii); ";";
Next ii
Printer.Print String(10, " ")
Printer.Print String(10, " "); "MOISTURE LIST:"
Printer.Print String(10, " ");
For ii = 1 To gSMLIST
    Printer.Print String(1, " "); gSMOISTURE(ii); ";";
Next ii
Printer.Print String(10, " ")
Printer.NewPage
Printer.Print Header; ' Print header.
Printer.Print Printer.Page ' Print page number.
Y = Printer.CurrentY + 10 ' Set position for line.
Printer.Line (0, Y)-(Printer.ScaleWidth, Y) ' Draw line.
Printer.Print String(10, " "); "DT LIST:"
For iii = 1 To gSMLIST
Printer.Print String(10, " ");
    For ii = 1 To gSTLIST
        Printer.Print String(2, " "); gSDT(ii, iii); ";";
    Next ii
    Printer.Print String(10, " ")
Next iii

Printer.Print String(10, " "); "DW LIST:"
For iii = 1 To gSMLIST
Printer.Print String(10, " ");

    For ii = 1 To gSTLIST
        Printer.Print String(2, " "); gSDW(ii, iii); ";";
    Next ii
    Printer.Print String(10, " ")
Next iii
Printer.Print String(10, " "); "CAPILLAR LIST"
For iii = 1 To gSMLIST
Printer.Print String(10, " ");

    For ii = 1 To gSTLIST
        Printer.Print String(2, " "); gSCAPILLAR(ii, iii); ";";
    Next ii
    Printer.Print String(10, " ")
Next iii
Printer.Print String(10, " "); "CONDUCT LIST"
For iii = 1 To gSMLIST
Printer.Print String(10, " ");

    For ii = 1 To gSTLIST
        Printer.Print String(2, " "); gSCONDUCT(ii, iii); ";";
    Next ii
    Printer.Print String(10, " ")
Next iii
Printer.Print String(10, " "); "Moisture Storage Function - ";
Printer.Print String(2, " "); "Number of points:", gSSorpCount
Printer.Print String(10, " ");

For ii = 1 To gSSorpCount
    Printer.Print String(2, " "); gSSorpHR(ii); ";";
Next ii
Printer.Print String(10, " ")
Printer.Print String(10, " ");
For ii = 1 To gSSorpCount
    Printer.Print String(2, " "); gSSorpWHR(ii); ";";
Next ii
Printer.Print String(10, " ")
Printer.EndDoc
End Sub

Sub totalweathertime()
Dim sum As Double
Dim i As Integer
sum = 0
For i = 1 To gTotalSteps

```

```

sum = sum + gTime(i)
Next
Print #3, sum; "Total Weather Time"
Print #3, gTotalTime, "Total Project time"
End Sub

```

B.2.4. Módulo "o_functions.bas"

Option Explicit

```

Static Function PSATURA(tsat)
'-----
'  CALCULATION OF SATURATION PRESSURE   Kg/m3
'-----
Dim aux
Dim aux1
Dim ak
Dim akk
Dim TT
Dim PVS
TT = 273.16
If tsat < 0 Then
    ak = -9.09685 * (TT / (TT + tsat) - 1) - 3.56654 * Log10(TT / (TT + tsat)) +
        0.87682 * (1 - (TT + tsat) / TT) + 0.78614
    PVS = (10 ^ ak * 100) / (461.5 * (tsat + 273))
ElseIf tsat > 0 Then
    aux = Log10((TT + tsat) / TT)
    aux1 = (1 - TT / (TT + tsat))
    akk = 10.79574 * aux1 - 5.028 * aux + 1.50475 * (10 ^ (-4)) * (1 - 10 ^ (-8.2969 *
        ((TT + tsat) / TT - 1))) + 0.42873 * (10 ^ (-3)) *
        (10 ^ (4.76955 * aux1) - 1) + 0.78614
    PVS = (10 ^ akk * 100) / (461.5 * (tsat + 273))
Else
    PVS = (10 ^ 0.78614 * 100) / (461.5 * (tsat + 273))
End If
PSATURA = PVS
End Function

```

```

Static Function PSATURAmHg(tsat)
'-----
'  CALCULATION OF SATURATION PRESSURE   mmHg
'-----
Dim aux
Dim aux1
Dim ak
Dim akk
Dim TT
Dim PVS
TT = 273.16
If tsat < 0 Then
    ak = -9.09685 * (TT / (TT + tsat) - 1) - 3.56654 * Log10(TT / (TT + tsat)) +
        0.87682 * (1 - (TT + tsat) / TT) + 0.78614
    PVS = (10 ^ ak) / 1.333
ElseIf tsat > 0 Then
    aux = Log10((TT + tsat) / TT)
    aux1 = (1 - TT / (TT + tsat))
    akk = 10.79574 * aux1 - 5.028 * aux + 1.50475 * (10 ^ (-4)) *
        (1 - 10 ^ (-8.2969 * ((TT + tsat) / TT - 1))) + 0.42873 * (10 ^ (-3)) *
        (10 ^ (4.76955 * aux1) - 1) + 0.78614
    PVS = 10 ^ akk / 1.3333
Else
    PVS = 10 ^ 0.78614 / 1.3333
End If
PSATURAmHg = PVS
End Function

```

```

Function SOL()
'-----
'  WALL TEMPERATURE CALCULATION
'-----
Dim SC, RDC, PI As Double
Dim SIG, TAMB, DECL As Double
Dim ECC, DSTAR As Double
Dim SOL_ET As Double
Dim TANDEC, TANLAT, HRANG As Double
Dim COSDEC As Double
Dim SINDEC As Double
Dim COSLAT, SINLAT As Double
Dim SINHR, COSHR As Double
Dim COSAZM, SINAZM As Double
Dim X1, WS, WSMAX, COSTZ As Double
Dim HO, XKT As Double

```

```

Dim HD, HB As Double
Dim X, V1 As Double
Dim CTC, AZM1 As Double
Dim RD, RR As Double
Dim RB, COSTT As Double
Dim HTT, TSOL As Double
Dim aux2 As Double
Dim TC As Double
Dim HDV, HDT, HBV As Double
SC = 1353
RDC = 0.0174533
PI = 3.1415927
SIG = 0.000000056698
TAMB = temp_out + 273.16
ECC = Cos(95)
DECL = 23.45 * Sin((284 + cday) / 365)
ECC = 1 + 0.033 * Cos(360 / 365 * cday)
DSTAR = 1 / 365.242 * (cday - 1)
SOL_ET = -(0.1236 * Sin(DSTAR) - 0.004289 * Cos(DSTAR) + 0.1539 * Sin(2 * DSTAR) + 0.06078 * Cos(2 *
DSTAR))
COSDEC = Cos(DECL * RDC)
COSLAT = Cos(gLatitude * RDC)
SINDEC = Sin(DECL * RDC)
SINLAT = Sin(gLatitude * RDC)
TANDEC = SINDEC / COSDEC
TANLAT = SINLAT / COSLAT
TC = chour + SOL_ET - (gLongitude / 15)
HRANG = (12 - TC) * 15 * RDC
X1 = -TANDEC * TANLAT
WS = Atn(Abs(Sqr(1 - (X1 * X1))) / X1)
WSMAX = WS - 7.5 * RDC
If (Abs(HRANG) > WSMAX) Then HRANG = WSMAX * (HRANG >= 0)
COSHR = Cos(HRANG)
SINHR = Sin(HRANG)
COSTZ = COSLAT * COSDEC * COSHR + SINLAT * SINDEC
If (COSTZ > 0) Then
    HO = SC * ECC * COSTZ
    XKT = Rad_ / HO
    aux2 = (1.0045 + 2.6313 * XKT ^ 3 - 3.5227 * XKT ^ 2 + 0.04349 * XKT)
    HD = Rad_ * aux2
    If (HD <= 0) Then HD = 0
    HB = Rad_ - HD
    X = Abs(gWindDirection - gAzimute)
    If Not (X < 90 Or X > 270) Then
        V1 = 0.3 * 0.05 * gWindSpeed
    Else
        V1 = 0.25 * gWindSpeed
        If (V1 < 0.5) Then
            V1 = 0.5
        End If
    End If
    CTC = 11.4 + 4.2 * V1
    AZM1 = gAzimute
    gAzimute = 180 - gAzimute
    COSAZM = Cos(gAzimute * RDC)
    SINAZM = Sin(gAzimute * RDC)
    RD = 0.5
    RR = 0.1
    RB = 0
    COSTT = -SINDEC * COSLAT * COSAZM + COSDEC * SINLAT * COSAZM * COSHR + COSDEC *
        SINAZM * SINHR
    If (COSTT > 0) Then
        RB = COSTT / COSTZ
    End If
    HBV = HB * RB
    HDV = HD * RD
    HDT = HDV + RR * Rad_
    HTT = HBV + HDT
    TSOL = temp_out + galpha * HTT / CTC
    gAzimute = AZM1
ElseIf (COSTZ <= 0) Then
    TSOL = temp_out
End If
SOL = TSOL
End Function

```

```

Sub GAUSSBACON()

```

```

Dim aux7
Dim DIVB As Double
Dim COEFF As Double
Dim SIGMA As Double
Dim K, j, ii
Dim LMOI As Double
Dim ICODE As Double
Dim MX2, MX1, M22, LMX As Double
Dim L As Double

```

```

Dim MI22, I11, i, I1 As Double
Dim K1, k2, k3, K4, K5, LF
Dim llc, l11, LMAT
Dim iq, iv, sa
Dim IM, m1, m2
Dim a(1000)
'transform the 2 dimension matrix in 1 dimension one
For j = 1 To 1000
    a(j) = 0
Next
LMAT = 2 * gTotalNodes
K1 = 3
LF = 3
For K = 1 To LMAT
    b(K) = BB(K)
Next
a(1) = 0
a(2) = 0
a(3) = 0
a(8) = 0
a(9) = 0
For K = 1 To 50
    a(K * 14) = 0
    a(K * 14 + 1) = 0
Next K
k2 = 6
For llc = 1 To 3
    k2 = k2 + 1
    If llc = 2 Then LF = LF + 2
    If llc = 3 Then LF = LF + 1
    For l11 = 4 To k2
        LF = LF + 1
        a(LF) = AA(llc, l11)
    Next l11
Next llc '
k3 = 9
For llc = 4 To LMAT - 3
    K1 = K1 + 1
    k3 = k3 + 1
    For l11 = K1 To k3
        LF = LF + 1
        a(LF) = AA(llc, l11)
    Next l11
Next llc '
K4 = LMAT - 3
K5 = LMAT + 3
For llc = LMAT - 2 To LMAT
    K4 = K4 + 1
    If llc = LMAT - 1 Then LF = LF + 1
    If llc = LMAT Then LF = LF + 2
    For l11 = K4 To K5
        LF = LF + 1
        a(LF) = AA(llc, l11)
    Next l11
Next llc
a(LMAT * 7) = 0
a(LMAT * 7 - 1) = 0
a(LMAT * 7 - 2) = 0
a(LMAT * 7 - 7) = 0
a(LMAT * 7 - 8) = 0
For iq = 1 To 82
    iv = iq * 7
Next
MX = 3
lv = LMAT
sa = 4
For aux7 = 1 To LMAT
    If a(sa) = 0 Then Print #3, "FOUND A ZERO PIVOT"
    sa = sa + 7
Next
If Rain_ <> 0 Then
    LMAT = LMAT - 1
    lv = LMAT
    For aux7 = 8 To 14 * gTotalNodes
        a(aux7 - 7) = a(aux7)
    Next
    For aux7 = 2 To 2 * gTotalNodes
        b(aux7 - 1) = b(aux7)
    Next
End If
'start gaussbacon
LMOI = lv - MX
ICODE = 1
MX2 = 2 * MX + 1
MX1 = MX + 1
M22 = MX2 - 1
L = MX1
For K = 1 To lv

```

```

DIVB = 1 / a(L)
LMX = L + MX
If (K > LMOI) Then
    ICODE = ICODE + 1
    LMX = LMX - ICODE + 1
End If
For j = L To LMX
    a(j) = a(j) * DIVB
Next j
b(K) = b(K) * DIVB
If (K <> lv) Then
    MI22 = 0
    For ii = ICODE To MX
        MI22 = MI22 + M22
        If (a(L + MI22) <> 0) Then
            COEFF = a(L + MI22)
            For j = L To LMX
                a(j + MI22) = a(j + MI22) - a(j) * COEFF
            Next j
            II1 = ii - ICODE + 1
            b(K + II1) = b(K + II1) - b(K) * COEFF
        End If
    Next ii
    L = L + MX2
Else: GoTo FIM1
End If
Next K '5
GoTo FIM1
FIM1:
ICODE = 0
i = lv
L = MX1 + 1 + (lv - 1) * MX2 '

```

```

Do
    SIGMA = 0
    L = L - MX2
    LMX = L + MX - 1
    If (i > LMOI) Then
        ICODE = ICODE + 1
        LMX = L + ICODE - 1
    End If
    I1 = i
    For j = L To LMX
        SIGMA = SIGMA + a(j) * b(I1)
        I1 = I1 + 1
    Next
    i = i - 1
    b(i) = b(i) - SIGMA
Loop While i > 1
End Sub

```

```

Public Sub passgauss()
Dim IM As Integer
Dim m1 As Integer
Dim m2 As Integer
For IM = 1 To gTotalNodes
    m1 = IM * 2 - 1
    m2 = IM * 2
    WF(IM) = b(m1)
    TF(IM) = b(m2)

    If Rain_ <> 0 Then
        TF(IM) = b(m1)
        If IM = 1 Then
            WF(IM) = gSSATURATION
        Else
            WF(IM) = b(m2 - 2)
        End If
    End If
Next IM
End Sub

```

```

Public Sub MATAMB(LZ)
'-----
'   CALCULATION OF FIRST AND LAST NODES
'-----
Dim FTERMI, am, bm, cm, dm, em As Double
Dim tsat, psat, psat1 As Double
Dim PS_out As Double
Dim PV_out As Double
Dim HRL As Double
Dim conte As Double
Dim fiabs As Double
Dim fm As Double
Dim FTERME As Double
Dim PSATLZ As Double
Dim PSINT As Double
Dim HRLZ As Double

```

```

Dim CONTI As Double
Dim tempoutpro As Double
Dim fiabs1 As Double
Dim FLUX1 As Double
'ReDim WSAT(5) As Double
Dim kf As Integer
Dim roc, at, bt, ct As Double
kf = LZ * 2
'FIRST NODE
If (LZ = 1) Then
    fiabs = 0
    roc = gSHEAT * gSDENSITY + 4180 * w(1, st) * gSDENSITY
    at = (WSTEP * 3600) / roc * ((Condnw(2) - 3 * Condnw(1)) / (Condnw(1) * dx(1))) *
        gHe
    bt = ((Condnw(2) - 3 * Condnw(1)) / dx(1) * fiabs / Condnw(1)) *
        (WSTEP * 3600) / roc
    ct = (2 * Condnw(1) / (dx(1) ^ 2)) * (WSTEP * 3600) / roc
    AA(kf, kf + 2) = 0
    AA(kf, kf + 3) = 1 - at / 2 + ct / 2
    AA(kf, kf + 4) = 0
    AA(kf, kf + 5) = -ct / 2
    If (Rad_ <> 0) Then
        If (TSOL <> temp_out) Then
            tempoutpro = temp_out
            temp_out = Tempoutfic
        End If
    End If
    BB(kf) = t(1, st) + at / 2 * t(1, st) - at * temp_out + ct / 2 * t(2, st) -
        ct / 2 * t(1, st) + bt
    FTERMI = Condnw(1) / gHe * (t(1, st) - temp_out)
    am = 2 * DWnw(1) / (dx(1) ^ 2)
    bm = 2 * DTnw(1) / (dx(1) ^ 2)
    cm = (DWnw(2) - 3 * DWnw(1)) / dx(1)
    dm = (DTnw(2) - 3 * DTnw(1)) / dx(1)
    em = -(dm * gHe / Condnw(1)) - (cm * DTnw(1) / DWnw(1)) * (gHe / Condnw(1))
    tsat = t(1, st)
    psat1 = PSATURA(tsat)
    tsat = temp_out
    If (TSOL <> temp_out) Then
        tsat = tempoutpro
    ElseIf (TSOL = temp_out) Then
        tsat = temp_out
    End If
    PS_out = PSATURA(tsat)
    PV_out = PS_out * RH_out / 100
    HR1 = Hurelat2(w(1, st))
    If (Rain_ <> 0) Then
        RH_out = 100
        PS_out = psat1
    End If
    conte = gBetaE * (psat1 * HR1 / 100 - PS_out * RH_out / 100)
    fm = cm * (gBetaE / (gSDENSITY * DWnw(1)) * (psat1 * HR1 / 100 - PS_out *
        RH_out / 100) + Rain_ / (gSDENSITY * DWnw(1))) - cm * fiabs / (Condnw(1) *
        DWnw(1) * DTnw(1)) + dm * fiabs / Condnw(1)
    AA(kf - 1, kf + 2) = 1 / (WSTEP * 3600) + am / 2
    AA(kf - 1, kf + 3) = bm / 2 - em / 2
    AA(kf - 1, kf + 4) = -am / 2
    AA(kf - 1, kf + 5) = -bm / 2
    BB(kf - 1) = (1 / (WSTEP * 3600) - am / 2) * w(1, st) + (em / 2 - bm / 2) *
        t(1, st) + am / 2 * w(2, st) + bm / 2 * t(2, st) + fm - em * temp_out
    If Rad_ <> 0 Then
        temp_out = tempoutpro
    End If
    If (TSOL <> temp_out) Then AA(kf - 1, kf + 3) = 0
    If (TSOL <> temp_out) Then
        BB(kf - 1) = (1 / (WSTEP * 3600) - am / 2) * w(1, st) - (bm) *
            TSOL + am / 2 * w(2, st) + bm / 2 * t(2, st) + fm
    End If
'LAST NODE
ElseIf (LZ = gTotalNodes) Then
    fiabs1 = 0
    FLUX1 = 0
    dx(gTotalNodes) = dx(gTotalNodes - 1)
    roc = gSHEAT * gSDENSITY + 4180 * w(LZ, st) * gSDENSITY
    at = -(WSTEP * 3600) / roc * ((3 * Condnw(LZ) - Condnw(LZ - 1)) /
        (Condnw(LZ) * dx(LZ))) * gHI
    bt = -((3 * Condnw(LZ) - Condnw(LZ - 1)) / dx(LZ) * fiabs1) *
        (WSTEP * 3600) / (roc * Condnw(LZ))
    ct = (2 * Condnw(LZ) / (dx(LZ) ^ 2)) * (WSTEP * 3600) / roc
    AA(kf, kf) = 0
    AA(kf, kf + 1) = -ct / 2
    AA(kf, kf + 2) = 0
    AA(kf, kf + 3) = (1 - at / 2 + ct / 2)
    BB(kf) = (1 + at / 2) * t(LZ, st) - at * temp_in + ct / 2 * t(LZ - 1, st) -
        ct / 2 * t(LZ, st) + bt
    FTERME = Condnw(LZ) / gHI * (t(gTotalNodes, st) - temp_in)
    am = 2 * DWnw(LZ) / (dx(LZ) ^ 2)
    bm = 2 * DTnw(LZ) / (dx(LZ) ^ 2)

```

```

cm = -(3 * DWnw(LZ) - DWnw(LZ - 1)) / dx(LZ)
dm = -(3 * DTnw(LZ) - DTnw(LZ - 1)) / dx(LZ)
em = -dm * gHI / Condnw(LZ) - (cm * DTnw(LZ) / (DWnw(LZ))) * (gHe / Condnw(LZ))
tsat = t(LZ, st)
PSATLZ = PSATURA(tsat)
PSINT = PSATURA(temp_in)
HRLZ = Hurelat2(w(LZ, st))
CONTI = gBetaI * (PSATLZ * HRLZ / 100 - PSINT * RH_in / 100)
fm = (cm * (CONTI / (gSDENSITY * DWnw(LZ)) + FLUX1 / (gSDENSITY * DWnw(LZ)) -
      cm * fiabs1 / (Condnw(LZ) * DWnw(LZ)) * DTnw(LZ)) + dm * fiabs1 / Condnw(LZ))
AA(kf - 1, kf + 2) = 1 / (WSTEP * 3600) + am / 2
AA(kf - 1, kf + 3) = bm / 2 - em / 2
AA(kf - 1, kf) = -am / 2
AA(kf - 1, kf + 1) = -bm / 2
BB(kf - 1) = (1 / (WSTEP * 3600) - am / 2) * w(LZ, st) + (em / 2 - bm / 2) *
             t(LZ, st) + am / 2 * w(LZ - 1, st) + bm / 2 * t(LZ - 1, st) + fm -
             em * temp_in
End If
End Sub

```

```

Public Sub MATCOR(LZ)
'-----
'  CALCULATION OF INSIDE NODES
'-----
Dim KK As Integer
Dim wlaux As Double
Dim aux As Double
Dim auxbb1, auxbb2, auxbb3 As Double
KK = LZ * 2
dx(gTotalNodes) = dx(gTotalNodes - 1)
AA(KK - 1, KK) = ((DWnw(LZ + 1) - DWnw(LZ - 1)) / (8 * dx(LZ) ^ 2)) -
                DWnw(LZ) / (2 * dx(LZ) ^ 2)
If (Rain_ <> 0 And LZ = 2) Then
    wlaux = AA(KK - 1, KK)
    AA(KK - 1, KK) = 0
End If
AA(KK - 1, KK + 1) = ((DTnw(LZ + 1) - DTnw(LZ - 1)) / (8 * dx(LZ) ^ 2)) -
                    DTnw(LZ) / (2 * dx(LZ) ^ 2)
AA(KK - 1, KK + 2) = 1 / (WSTEP * 3600) + DWnw(LZ) / (dx(LZ) ^ 2)
AA(KK - 1, KK + 3) = DTnw(LZ) / (dx(LZ) ^ 2)
AA(KK - 1, KK + 4) = -((DWnw(LZ + 1) - DWnw(LZ - 1)) / (8 * dx(LZ) ^ 2)) -
                    DWnw(LZ) / (2 * dx(LZ) ^ 2)
AA(KK - 1, KK + 5) = -((DTnw(LZ + 1) - DTnw(LZ - 1)) / (8 * dx(LZ) ^ 2)) -
                    DTnw(LZ) / (2 * dx(LZ) ^ 2)
AA(KK, KK + 1) = ((Condnw(LZ + 1) - Condnw(LZ - 1)) / (8 * dx(LZ) ^ 2)) -
                Condnw(LZ) / (2 * dx(LZ) ^ 2)
aux = gSDENSITY * gSHEAT + 4180 * w(LZ, st) * gSDENSITY
AA(KK, KK + 3) = aux / (WSTEP * 3600) + Condnw(LZ) / (dx(LZ) ^ 2)
AA(KK, KK + 5) = -((Condnw(LZ + 1) - Condnw(LZ - 1)) / (8 * dx(LZ) ^ 2)) -
                Condnw(LZ) / (2 * dx(LZ) ^ 2)
AA(KK, KK) = 0
AA(KK, KK + 2) = 0
AA(KK, KK + 4) = 0
BB(KK - 1) = -(AA(KK - 1, KK) * w(LZ - 1, st)) - (AA(KK - 1, KK + 1) * t(LZ - 1, st))
            + ((1 / (WSTEP * 3600) - DWnw(LZ) / dx(LZ) ^ 2) * w(LZ, st)) -
            (AA(KK - 1, KK + 3) * t(LZ, st)) - (AA(KK - 1, KK + 4) * w(LZ + 1, st)) -
            (AA(KK - 1, KK + 5) * t(LZ + 1, st))
BB(KK) = -(AA(KK, KK + 1) * t(LZ - 1, st)) + (aux / (WSTEP * 3600) -
        Condnw(LZ) / (dx(LZ) ^ 2)) * t(LZ, st) - (AA(KK, KK + 5) * t(LZ + 1, st))
If (Rain_ <> 0 And LZ = 2) Then
    BB(KK - 1) = BB(KK - 1) - 2 * wlaux * w(LZ - 1, st)
End If
End Sub

```

```

Public Sub FindError()
Dim ect(50) As Double
Dim ecw(50) As Double
Dim Mem1 As Double
Dim Mem2 As Double
Dim rrr As Integer
Mem1 = 0
Mem2 = 0
ERRORT = 0
For rrr = 1 To gTotalNodes
    ect(rrr) = Abs(TF(rrr) - ta(rrr, IT))
    ecw(rrr) = Abs(WF(rrr) - wa(rrr, IT))
    If ect(rrr) > Mem1 Then Mem1 = ect(rrr)
    If ecw(rrr) > Mem2 Then Mem2 = ecw(rrr)
Next
If Mem1 > Mem2 Then ERRORT = Mem1
If Mem1 <= Mem2 Then ERRORT = Mem2
End Sub

```

```

Function cofenon(Temp, hum)
Dim L, LL, K, KK, LL1, KK1 As Integer
Dim P As Double
Dim BD, BE, CD, CE, PW, PE As Double

```



```

Dim X1, X2, XTOT As Double
Dim Y1, Y2, YTOT As Double
Dim i As Integer
L = 0
LL = 0
K = 0
KK = 0
If hum >= gSMOISTURE(1) And hum <= gSMOISTURE(gSMLIST) Then
    For i = 1 To gSMLIST
        If (hum = gSMOISTURE(i)) Then L = i
    Next
    For i = 1 To gSMLIST - 1
        If (hum > gSMOISTURE(i) And hum < gSMOISTURE(i + 1)) Then LL = i
    Next
ElseIf hum < gSMOISTURE(1) Then L = 1
ElseIf hum > gSMOISTURE(gSMLIST) Then L = gSMLIST
End If
If Temp >= gSTEMPERATURE(1) And Temp <= gSTEMPERATURE(3) Then
    For i = 1 To 3
        If Temp = gSTEMPERATURE(i) Then K = i
    Next
    For i = 1 To 2
        If Temp > gSTEMPERATURE(i) And Temp < gSTEMPERATURE(i + 1) Then K = i
    Next
ElseIf (Temp > gSTEMPERATURE(3)) Then K = 3
ElseIf (Temp < gSTEMPERATURE(1)) Then K = 1
End If
If LL = 0 And KK = 0 Then
    P = Table(K, L)
ElseIf KK = 0 And LL <> 0 Then
    CE = Table(K, LL)
    LL1 = LL + 1
    CD = Table(K, LL1)
    If (CE <> CD) Then
        X1 = hum - gSMOISTURE(LL)
        XTOT = gSMOISTURE(LL1) - gSMOISTURE(LL)
        X2 = XTOT - X1
        P = (CE * X2 + CD * X1) / XTOT
    ElseIf (CE = CD) Then
        P = CE
    End If
ElseIf KK <> 0 And LL = 0 Then
    CE = Table(KK, L)
    KK1 = KK + 1
    BE = Table(KK1, L)
    If (CE <> BE) Then
        Y1 = Temp - gSMOISTURE(KK)
        YTOT = gSTEMPERATURE(KK1) - gSTEMPERATURE(KK)
        Y2 = YTOT - Y1
        P = (CE * Y2 + BE * Y1) / YTOT
    ElseIf (CE = BE) Then
        P = CE
    End If
Else
    CE = Table(KK, LL)
    KK1 = KK + 1
    LL1 = LL + 1
    CD = Table(KK, LL1)
    BE = Table(KK1, LL)
    BD = Table(KK1, LL1)
    Y1 = Temp - gSTEMPERATURE(KK)
    YTOT = gSTEMPERATURE(KK1) - gSTEMPERATURE(KK)
    X1 = hum - gSMOISTURE(LL)
    XTOT = gSMOISTURE(LL1) - gSMOISTURE(LL)
    X2 = XTOT - X1
    Y2 = YTOT - Y1
    If (CE <> BE) Then
        PW = (CE * Y2 + BE * Y1) / YTOT
    ElseIf (CE = BE) Then
        PW = CE
    End If
    If (CD <> BD) Then
        PE = (CD * Y2 + BD * Y1) / YTOT
    ElseIf (CD = BD) Then
        PE = CD
    End If
    P = (PW * X2 + PE * X1) / XTOT
End If

cofenon = P
End Function

```

B.2.5. Módulo “DataFunctions.bas”

Option Explicit
Dim c As Integer

```
Function CleanValue() 'trhum 98
'Sets all Variables to nul value
'time variables
'-----
gDateS = 1 - 1 - 1900 'Date Start
gDateF = 1 - 1 - 1900 'Date Finish
gHoursS = "0:00:00"
gHoursF = "0:00:00"
gTotalWeather = 0 'Total time of the Weather File
gTotalSteps = 0 'Number of Time Intervals
'weather variables
'-----
gN = 0 'number of steps in weather file
Dim aux1 As Integer
For aux1 = 1 To 100
  gnumber(aux1) = 0
  gRain(aux1) = 0 'Weather rain yes or no
  gTempIn(aux1) = 0 'indoor temperature
  gTempOut(aux1) = 0 'outdoor temperature
  gRHIn(aux1) = 0 'indoor relative humidity
  gRHOut(aux1) = 0 'outdoor relative humidity
  grad(aux1) = 0 'radiation
  gTime(aux1) = 0 'time step in hours
  gstep(aux1) = 0 'step in hours
Next aux1
'initial conditions variables
'-----
gHe = 0 'outdoor superficial conductance
gHI = 0 'indoor superficial conductance
gBetaI = 0 'indoor moisture coefficient
gBetaE = 0 'outdoor moisture coefficient
gErrorMax = 0 'maximum error
gLatitude = 0
gLongitude = 0
gNorS = 0 'latitude North or South
gEorW = 0 'longitude east or west
gWindDirection = 0
gWindSpeed = 0
gAzimute = 0
gCity = ""
gCountry = ""
'initial Profiles variables
'-----
gTempStart(1) = 0 'initial Temperature
gTempStart(2) = 0 'initial Temperature
gTempStart(3) = 0 'initial Temperature
gRHStart(1) = 0 'initial relative humidity
gRHStart(2) = 0 'initial relative humidity
gRHStart(3) = 0 'initial relative humidity
'object definition variables
'-----
gObjName = ""
gObjDescription = ""
gObjNotes = ""
gNumberOfLayers = 1 ' total number of layers
gLayerNumber(1) = 1 ' presente layer number
gThickness(1) = 1 ' thickness os presente layer
gSStep(1) = 1 ' number of steps in present layer
gdx(1) = 1 ' dx used in present layer
gLayerNumber(2) = 2 ' presente layer number
gThickness(2) = 1 ' thickness os presente layer
gSStep(2) = 1 ' number of steps in present layer
gdx(2) = 1 ' dx used in present layer
gLayerNumber(3) = 3 ' presente layer number
gThickness(3) = 1 ' thickness os presente layer
gSStep(3) = 1 ' number of steps in present layer
gdx(3) = 1 ' dx used in present layer
gMATERIALNAME = "" ' material name of present layer
gmaterialcode = 0 'code for identify the material
gtthickness = 1 'material total thickness
End Function
```

```
Function Open_File() 'trhum 98 files
```

```
Open Filename For Input As #1
On Error Resume Next
```

```

'project definitions variables
'-----
Dim offset(1000) As Long
Dim P As Integer
Dim line$
Dim FgTotalNodeslds$
Dim version As Integer
Dim aux2 As Integer
Dim aux1 As Integer
Dim aux3 As Integer
Dim aux4 As Integer
P = 1
Do
    offset(P) = Seek(1)
    Line Input #1, line$
    P = P + 1
Loop Until EOF(1)

offset(0) = P - 1
For P = 1 To offset(0)
    Seek #1, offset(P)
    Line Input #1, line$
Next
For P = offset(0) To 1 Step -1
    Seek #1, offset(P)
    Line Input #1, line$
Next
aux2 = 1
Seek #1, offset(aux2)
    Line Input #1, line$
    version = Val(line$)
    If version <> 98 Then
        message = "This is not a Trhum 98 File! Chose another file!"
        messagebox
        Exit Function
    End If
aux2 = 3
Seek #1, offset(aux2)
    Line Input #1, gObjName
aux2 = aux2 + 2 '5
Seek #1, offset(aux2)
    Line Input #1, gObjDescription
aux2 = aux2 + 2 '7
Seek #1, offset(aux2)
    Line Input #1, gObjNotes
aux2 = aux2 + 2 '9
Seek #1, offset(aux2)
    Line Input #1, gMATERIALNAME
aux2 = aux2 + 2 '#11
Seek #1, offset(aux2)
    Line Input #1, line$
    gmaterialcode = txt2double(line$)
aux2 = aux2 + 2 '13
Seek #1, offset(aux2)
    Line Input #1, line$
    gtthickness = txt2double(line$)
aux2 = aux2 + 2 '15
Seek #1, offset(aux2)
    Line Input #1, line$
    gNumberOfLayers = txt2double(line$)
aux2 = aux2 + 2 '17
Seek #1, offset(aux2)
    Line Input #1, line$
    gLayerNumber(1) = txt2double(line$)
aux2 = aux2 + 2 '19
Seek #1, offset(aux2)
    Line Input #1, line$
    gThickness(1) = txt2double(line$)
aux2 = aux2 + 2 '#21
Seek #1, offset(aux2)
    Line Input #1, line$
    gSStep(1) = txt2double(line$)
aux2 = aux2 + 2 '23
Seek #1, offset(aux2)
    Line Input #1, line$
    gdx(1) = txt2double(line$)
aux2 = aux2 + 2 '#25
Seek #1, offset(aux2)
    Line Input #1, line$
    gLayerNumber(2) = txt2double(line$)
aux2 = aux2 + 2
Seek #1, offset(aux2)
    Line Input #1, line$
    gLayerNumber(1) = txt2double(line$)
aux2 = aux2 + 2
Seek #1, offset(aux2)
    Line Input #1, line$
    gSStep(2) = txt2double(line$)

```

```

aux2 = aux2 + 2 '#31
Seek #1, offset(aux2)
    Line Input #1, line$
   .gdx(2) = txt2double(line$)
aux2 = aux2 + 2
Seek #1, offset(aux2)
    Line Input #1, line$
   .gLayerNumber(3) = txt2double(line$)
aux2 = aux2 + 2 '#35
Seek #1, offset(aux2)
    Line Input #1, line$
   .gThickness(3) = txt2double(line$)
aux2 = aux2 + 2
Seek #1, offset(aux2)
    Line Input #1, line$
   .gSStep(3) = txt2double(line$)
aux2 = aux2 + 2
Seek #1, offset(aux2)
    Line Input #1, line$
   .gdx(3) = txt2double(line$)
'time variables
'-----
aux2 = aux2 + 2 '#41
Seek #1, offset(aux2)
    Line Input #1, line$
   .gDateS = line$
aux2 = aux2 + 2
Seek #1, offset(aux2)
    Line Input #1, line$
   .gDateF = line$
aux2 = aux2 + 2 '#45
Seek #1, offset(aux2)
    Line Input #1, line$
   .gHoursS = line$
aux2 = aux2 + 2
Seek #1, offset(aux2)
    Line Input #1, line$
   .gHoursF = line$
aux2 = aux2 + 2
Seek #1, offset(aux2)
    Line Input #1, line$
   .gYears = txt2double(line$)
aux2 = aux2 + 2 '#51
Seek #1, offset(aux2)
    Line Input #1, line$
   .gDayS = txt2double(line$)
aux2 = aux2 + 2
Seek #1, offset(aux2)
    Line Input #1, line$
   .gHourS = txt2double(line$)
aux2 = aux2 + 2 '#55
Seek #1, offset(aux2)
    Line Input #1, line$
   .gTotalTime = txt2double(line$)
aux2 = aux2 + 2
Seek #1, offset(aux2)
    Line Input #1, line$
   .gTotalWeather = txt2double(line$)
'Initial conditions variables
'-----
aux2 = aux2 + 2
Seek #1, offset(aux2)
    Line Input #1, line$
   .gHI = txt2double(line$)
aux2 = aux2 + 2 '#61
Seek #1, offset(aux2)
    Line Input #1, line$
   .gHe = txt2double(line$)
aux2 = aux2 + 2 '#63
Seek #1, offset(aux2)
    Line Input #1, line$
   .gBetaI = txt2double(line$)
aux2 = aux2 + 2
Seek #1, offset(aux2)
    Line Input #1, line$
   .gBetaE = txt2double(line$)
aux2 = aux2 + 2
Seek #1, offset(aux2)
    Line Input #1, line$
   .gErrorMax = txt2double(line$)
aux2 = aux2 + 2 '#69
Seek #1, offset(aux2)
    Line Input #1, line$
   .gCity = line$
aux2 = aux2 + 2 '#71
Seek #1, offset(aux2)
    Line Input #1, line$
   .gCountry = line$

```

```

aux2 = aux2 + 2 '#77
Seek #1, offset(aux2)
    Line Input #1, line$
    gLatitude = txt2double(line$)
aux2 = aux2 + 2 '79
Seek #1, offset(aux2)
    Line Input #1, line$
    gNorS = txt2double(line$)
aux2 = aux2 + 2 '81
Seek #1, offset(aux2)
    Line Input #1, line$
    gLongitude = txt2double(line$)
aux2 = aux2 + 2 '83
Seek #1, offset(aux2)
    Line Input #1, line$
    gEorW = txt2double(line$)
aux2 = aux2 + 2 '#85
Seek #1, offset(aux2)
    Line Input #1, line$
    gWindDirection = txt2double(line$)
aux2 = aux2 + 2 '87
Seek #1, offset(aux2)
    Line Input #1, line$
    gWindSpeed = txt2double(line$)
aux2 = aux2 + 2 '89
Seek #1, offset(aux2)
    Line Input #1, line$
    gAzimute = txt2double(line$)
'initial profiles
-----
aux2 = aux2 + 2 '#91
Seek #1, offset(aux2)
    Line Input #1, line$
    FgTotalNodeslds$ = line$
    Dim extract(50, 2) As Integer
    Dim ps As Integer
    Dim f As Integer
    Dim nfgTotalNodeslds
    For P = 1 To Len(FgTotalNodeslds$)
        If Mid$(FgTotalNodeslds$, P, 1) = "," Then Mid$(FgTotalNodeslds$, P, 1) = "."
    Next
    ps = 1
    f = 0
    extract(1, 1) = 1
    Do
        f = f + 1
        extract(f, 1) = ps
        ps = InStr(ps, FgTotalNodeslds$, ";")
        extract(f, 2) = ps - extract(f, 1)
        ps = ps + 1
    Loop While ps <> 1
    extract(f, 2) = Len(FgTotalNodeslds$) + 1 - extract(f, 1)
    nfgTotalNodeslds = f
    For P = 1 To nfgTotalNodeslds
        gTempStart(P) = Val(Mid$(FgTotalNodeslds$, extract(P, 1), extract(P, 2)))
    Next
aux2 = aux2 + 2 '#93
Seek #1, offset(aux2)
    Line Input #1, line$
    FgTotalNodeslds$ = line$
    For P = 1 To Len(FgTotalNodeslds$)
        If Mid$(FgTotalNodeslds$, P, 1) = "," Then Mid$(FgTotalNodeslds$, P, 1) = "."
    Next
    ps = 1
    f = 0
    extract(1, 1) = 1
    Do
        f = f + 1
        extract(f, 1) = ps
        ps = InStr(ps, FgTotalNodeslds$, ";")
        extract(f, 2) = ps - extract(f, 1)
        ps = ps + 1
    Loop While ps <> 1
    extract(f, 2) = Len(FgTotalNodeslds$) + 1 - extract(f, 1)
    nfgTotalNodeslds = f
    For P = 1 To nfgTotalNodeslds
        gRHStart(P) = Val(Mid$(FgTotalNodeslds$, extract(P, 1), extract(P, 2)))
    Next
aux2 = aux2 + 2 '#95
Seek #1, offset(aux2)
    Line Input #1, line$
    galpha = txt2double(line$)
aux2 = aux2 + 2 '#97
Seek #1, offset(aux2)
    Line Input #1, line$
    nprofile = txt2double(line$)
aux2 = aux2 + 1 '#98
Seek #1, offset(aux2)

```

```

Line Input #1, line$
FgTotalNodeslds$ = line$
For P = 1 To Len(FgTotalNodeslds$)
    If Mid$(FgTotalNodeslds$, P, 1) = "," Then Mid$(FgTotalNodeslds$, P, 1) = "."
Next
ps = 1
f = 0
extract(1, 1) = 1
Do
    f = f + 1
    extract(f, 1) = ps
    ps = InStr(ps, FgTotalNodeslds$, ";")
    extract(f, 2) = ps - extract(f, 1)
    ps = ps + 1
Loop While ps <> 1
extract(f, 2) = Len(FgTotalNodeslds$) + 1 - extract(f, 1)
nfgTotalNodeslds = f
For P = 1 To nprofile
    time_profile(P) = Val(Mid$(FgTotalNodeslds$, extract(P, 1), extract(P, 2)))
Next

aux2 = 100
aux2 = aux2 + 2 '#102
Seek #1, offset(aux2)
    Line Input #1, line$
    gTotalSteps = txt2double(line$)
aux2 = aux2 + 2 '#104
For aux3 = 1 To gTotalSteps
Seek #1, offset(aux2)
    Line Input #1, line$
    FgTotalNodeslds$ = line$
    For P = 1 To Len(FgTotalNodeslds$)
        If Mid$(FgTotalNodeslds$, P, 1) = "," Then Mid$(FgTotalNodeslds$, P, 1) = "."
    Next
    ps = 1
    f = 0
    extract(1, 1) = 1
    Do
        f = f + 1
        extract(f, 1) = ps
        ps = InStr(ps, FgTotalNodeslds$, ";")
        extract(f, 2) = ps - extract(f, 1)
        ps = ps + 1
    Loop While ps <> 1
    extract(f, 2) = Len(FgTotalNodeslds$) + 1 - extract(f, 1)
    nfgTotalNodeslds = f
    For P = 1 To nfgTotalNodeslds
        Colum(P) = txt2double(Mid$(FgTotalNodeslds$, extract(P, 1), extract(P, 2)))
    Next
    wetline = aux3
    passweather98
    aux2 = aux2 + 1
Next aux3
'material
'-----
aux2 = aux2 + 3 '107
Seek #1, offset(aux2)
    Line Input #1, line$
    gMATERIALNAME = line$
aux2 = aux2 + 2 '109
Seek #1, offset(aux2)
    Line Input #1, line$
    gmaterialcode = txt2double(line$)
aux2 = aux2 + 2 '111
Seek #1, offset(aux2)
    Line Input #1, line$
    gSDENSITY = txt2double(line$)
aux2 = aux2 + 2 '113
Seek #1, offset(aux2)
    Line Input #1, line$
    gSHEAT = txt2double(line$)
aux2 = aux2 + 2 '115
Seek #1, offset(aux2)
    Line Input #1, line$
    gSHYGROSCOP = txt2double(line$)
aux2 = aux2 + 2 '117
Seek #1, offset(aux2)
    Line Input #1, line$
    gSSATURATION = txt2double(line$)
aux2 = aux2 + 2 '119
Seek #1, offset(aux2)
    Line Input #1, line$
    gSTLIST = txt2double(line$)
aux2 = aux2 + 2 '121
Seek #1, offset(aux2)
    Line Input #1, line$
    gSMLIST = txt2double(line$)
aux2 = aux2 + 1 '122

```

```

For aux3 = 1 To gSTLIST
aux2 = aux2 + 1
Seek #1, offset(aux2)
Line Input #1, line$
gSTEMPERATURE(aux3) = txt2double(line$)
Next
aux2 = aux2 + 1 '126
For aux3 = 1 To gSMLIST
aux2 = aux2 + 1
Seek #1, offset(aux2)
Line Input #1, line$
gSMOISTURE(aux3) = txt2double(line$)
Next
aux2 = aux2 + 1 '126+mlist+1
For aux3 = 1 To gSTLIST
For aux4 = 1 To gSMLIST
aux2 = aux2 + 1
Seek #1, offset(aux2)
Line Input #1, line$
gSDT(aux3, aux4) = txt2double(line$)
Next
Next
aux2 = aux2 + 1 '126+mlist+1+mlist*tlist+1
For aux3 = 1 To gSTLIST
For aux4 = 1 To gSMLIST
aux2 = aux2 + 1
Seek #1, offset(aux2)
Line Input #1, line$
gSDW(aux3, aux4) = txt2double(line$)
Next
Next
aux2 = aux2 + 1 '126+mlist+1+2*mlist*tlist+2
For aux3 = 1 To gSTLIST
For aux4 = 1 To gSMLIST
aux2 = aux2 + 1
Seek #1, offset(aux2)
Line Input #1, line$
gSCAPILLAR(aux3, aux4) = txt2double(line$)
Next
Next
aux2 = aux2 + 1 '126+mlist+1+3*mlist*tlist+3
For aux3 = 1 To gSTLIST
For aux4 = 1 To gSMLIST
aux2 = aux2 + 1
Seek #1, offset(aux2)
Line Input #1, line$
gSCONDUCT(aux3, aux4) = txt2double(line$)
Next
Next
aux2 = aux2 + 3
Seek #1, offset(aux2)
Line Input #1, line$
gSSorpCount = txt2double(line$)
For aux3 = 1 To gSSorpCount
aux2 = aux2 + 1
Seek #1, offset(aux2)
Line Input #1, line$
gSSorpHR(aux3) = txt2double(line$)
Next
For aux3 = 1 To gSSorpCount
aux2 = aux2 + 1
Seek #1, offset(aux2)
Line Input #1, line$
gSSorpWHR(aux3) = txt2double(line$)
Next
Close #1
End Function

```

```
Function save_file() 'trhum 98 files
```

```
Dim ii As Integer
Dim iii As Integer
```

```
On Error Resume Next
Open Filename For Output As #1
```

```
'project definitions variables
```

```
-----
Print #1, "98"
Print #1, "Project Name:" '#2
Print #1, gObjName
Print #1, "Project Description:"
Print #1, gObjDescription
Print #1, "Project Notes:"
Print #1, gObjNotes
Print #1, "Material Name:"
Print #1, gMATERIALNAME

```

```

Print #1, "Material Code:"
Print #1, gmaterialcode '#11
Print #1, "Total Thickness:"
Print #1, gtthickness
Print #1, "Number of Layers:"
Print #1, gNumberOFLayers
Print #1, "Layer Number 1:"
Print #1, gLayerNumber(1)
Print #1, "Thickness layer 1:"
Print #1, gThickness(1)
Print #1, "Number of Steps layer 1:"
Print #1, gSStep(1) '#21
Print #1, "Step of Layer 1:"
Print #1, gdx(1)
Print #1, "Layer Number 2:"
Print #1, gLayerNumber(2)
Print #1, "Thickness layer 2:"
Print #1, gThickness(2)
Print #1, "Number of Steps layer 2:"
Print #1, gSStep(2)
Print #1, "Step of Layer 2:"
Print #1, gdx(2) '#31
Print #1, "Layer Number 3:"
Print #1, gLayerNumber(3)
Print #1, "Thickness layer 3:"
Print #1, gThickness(3)
Print #1, "Number of Steps layer 3:"
Print #1, gSStep(3)
Print #1, "Step of Layer 3:"
Print #1, gdx(2)
'time variables
'-----
Print #1, "Project start date: "
Print #1, gDateS '#41
Print #1, "Project Finish date:"
Print #1, gDateF
Print #1, "Project start hour : "
Print #1, gHoursS
Print #1, "Project Finish hour:"
Print #1, gHoursF
Print #1, "Starting Year:"
Print #1, gYears
Print #1, "Day of the Year:"
Print #1, gDayS '#51
Print #1, "Hour of the day:"
Print #1, gHourS
Print #1, "Total Calculation Time:"
Print #1, gTotalTime
Print #1, "Total weather file Time:"
Print #1, gTotalWeather
'Initial conditions variables
'-----
Print #1, "Indoor superficial conductance:"
Print #1, gHI
Print #1, "outdoor superficial conductance:"
Print #1, gHe '#61
Print #1, "Indoor moisture coefficient:"
Print #1, gBetaI
Print #1, "outdoor moisture coefficient:"
Print #1, gBetaE
Print #1, "Maximum error:"
Print #1, gErrorMax
Print #1, "City:"
Print #1, gCity
Print #1, "country:"
Print #1, gCountry '#71
Print #1, "latitude:"
Print #1, gLatitude
Print #1, "North or South:"
Print #1, gNorS
Print #1, "longitude:"
Print #1, gLongitude
Print #1, "East or West:"
Print #1, gEorW
Print #1, "wind direction:"
Print #1, gWindDirection '#81
Print #1, "wind speed:"
Print #1, gWindSpeed
Print #1, "Azimut of the perpendicular plane to the surface:"
Print #1, gAzimute
'initial profiles
'-----
Dim n As Integer
Print #1, "Initial Profile of temperature:"
Print #1, gTempStart(1); ";"; gTempStart(2); ";"; gTempStart(3)
Print #1, "Initial profile of moisture:"
Print #1, gRHStart(1); ";"; gRHStart(2); ";"; gRHStart(3)
Print #1, "Absorption coefficient for radiation"

```



```

Print #1, galpha
Print #1, "number of profiles"
Print #1, nprofile
For n = 1 To nprofile
    Print #1, time_profile(n); ";";
Next
Print #1,
Print #1,
Print #1,
Print #1,
Print #1,
Print #1,
Print #1,
'weather
'-----
Print #1, "Beginning of the Weather file"
Print #1, "Number of time intervals:"
Print #1, gTotalSteps
Print #1, "Number ;"; "time ;"; "step ;"; "Tempout ;"; "HRout ;"; "Rad ;"; "Rain ;"; "Tempin ;"; "HRin ;"
'#100
For c = 1 To gTotalSteps
Print #1, gnumber(c); ";"; gTime(c); ";"; gstep(c); ";"; gTempOut(c); ";"; gRHOut(c); ";"; grad(c); ";";
gRain(c); ";"; gTempIn(c); ";"; gRHIn(c) '
Next
Print #1, "End of weather file" '100+3+gtotalSteps
'material data bank
'-----
Print #1, "Beginning of the Material Data Bank"
Print #1, "MATERIAL NAME:"
Print #1, gMATERIALNAME
Print #1, "MATERIAL CODE"
Print #1, gmaterialcode
Print #1, "DENSITY:"
Print #1, gSDENSITY
Print #1, "HEAT"
Print #1, gSHEAT
Print #1, "HYGROSCOP"
Print #1, gSHYGROSCOP
Print #1, "SATURATION"
Print #1, gSSATURATION
Print #1, "NUMBER OF TEMPERATURE LIST"
Print #1, gSTLIST
Print #1, "NUMBER OF MOIST LIST"
Print #1, gSMLIST
Print #1, "TEMPERATURE LIST:"
For ii = 1 To gSTLIST
    Print #1, gSTEMPERATURE(ii)
Next ii
Print #1, "MOISTURE LIST:"
    For ii = 1 To gSMLIST
        Print #1, gSMOISTURE(ii)
Next ii
Print #1, "DT LIST"
    For ii = 1 To gSTLIST
        For iii = 1 To gSMLIST
            Print #1, gSDT(ii, iii)
        Next iii
    Next ii
Print #1, "DW LIST"
    For ii = 1 To gSTLIST
        For iii = 1 To gSMLIST
            Print #1, gSDW(ii, iii)
        Next iii
    Next ii
Print #1, "CAPILLAR LIST"
    For ii = 1 To gSTLIST
        For iii = 1 To gSMLIST
            Print #1, gSCAPILLAR(ii, iii)
        Next iii
    Next ii
Print #1, "CONDUCT LIST"
    For ii = 1 To gSTLIST
        For iii = 1 To gSMLIST
            Print #1, gSCONDUCT(ii, iii)
        Next iii
    Next ii
Print #1, "moisture Storage Function"
Print #1, "number of points"
Print #1, gSSorpCount
For ii = 1 To gSSorpCount
    Print #1, gSSorpHR(ii)
Next ii
For ii = 1 To gSSorpCount
    Print #1, gSSorpWHR(ii)
Next ii
ii = 329 + 1 + 2 * gSSorpCount
ii = 350 - ii
For iii = 1 To ii
    Print #1, ""

```

```

Next iii
Close #1

End Function

```

```

Function Save_MATFile() 'trhum 98 files

```

```

Dim i As Integer
Dim ii As Integer
Dim iii As Integer

Open Filename For Output As #1

Print #1, "MATERIAL LIST NUMBER:" '#1
Print #1, gMLISTNUMBER
Print #1, "MATERIAL LIST:"
gMLISTNUMBER = gMLISTNUMBER

For i = 1 To gMLISTNUMBER
    Print #1, gMATNAME(i)
Next i
    Print #1, "DENSITY LIST" '4+A
For i = 1 To gMLISTNUMBER
    Print #1, gDENSITY(i)
Next i
    Print #1, "HEAT LIST" '5+2*gMLISTNUMBER
For i = 1 To gMLISTNUMBER
    Print #1, gHEAT(i)
Next i
Print #1, "HYGROSCOP LIST" '6+3*gMLISTNUMBER
For i = 1 To gMLISTNUMBER
    Print #1, gHYGROSCOP(i)
Next i
Print #1, "SATURATION LIST" '7+4*gMLISTNUMBER
For i = 1 To gMLISTNUMBER
    Print #1, gSATURATION(i)
Next i
Print #1, "NYMBER OF TEMPERATURE LIST" '8+5*gMLISTNUMBER
For i = 1 To gMLISTNUMBER
    Print #1, gTLIST(i) 'B(gMLISTNUMBER)
Next i
Print #1, "NYMBER OF MOIST LIST" '9+6*gMLISTNUMBER
For i = 1 To gMLISTNUMBER
    Print #1, gMLIST(i) 'C(gMLISTNUMBER)
Next i
Print #1, "TEMPERATURE LIST:" '10+7*gMLISTNUMBER -31
For i = 1 To gMLISTNUMBER
    For ii = 1 To gTLIST(gMLISTNUMBER)
        Print #1, gTEMPERATURE(i, ii)
    Next ii
Next i
Print #1, "MOISTURE LIST:" '11+7*A+N -41
For i = 1 To gMLISTNUMBER
    For ii = 1 To gMLIST(i)
        Print #1, gMOISTURE(i, ii)
    Next ii
Next i
Print #1, "DT LIST" '12+7*A+N+NN
For i = 1 To gMLISTNUMBER
    For ii = 1 To gTLIST(i)
        For iii = 1 To gMLIST(i)
            Print #1, gDT(i, ii, iii)
        Next iii
    Next ii
Next i
Print #1, "DW LIST" '13+7*A+N+NN+NNN
For i = 1 To gMLISTNUMBER
    For ii = 1 To gTLIST(i)
        For iii = 1 To gMLIST(i)
            Print #1, gDW(i, ii, iii)
        Next iii
    Next ii
Next i
Print #1, "CAPILLAR LIST" '14+7*A+N+NN+2*NNN
For i = 1 To gMLISTNUMBER
    For ii = 1 To gTLIST(i)
        For iii = 1 To gMLIST(i)
            Print #1, gCAPILLAR(i, ii, iii)
        Next iii
    Next ii
Next i
Print #1, "CONDUCT LIST" '15+7*A+N+NN+3*NNN
For i = 1 To gMLISTNUMBER
    For ii = 1 To gTLIST(i)
        For iii = 1 To gMLIST(i)
            Print #1, gCONDUCT(i, ii, iii)
        Next iii
    Next ii

```

```

        Next ii
    Next i
    Print #1, "MOISTURE STORAGE FUNCTION"
    Print #1, "NUMBER OF POINTS"
    For i = 1 To gMLISTNUMBER
    Print #1, gSorpCount(i)
    Next
    For i = 1 To gMLISTNUMBER
        For ii = 1 To gSorpCount(i)
        Print #1, GSorpPHR(i, ii)
        Next
        For ii = 1 To gSorpCount(i)
        Print #1, GSorpWHR(i, ii)
        Next
    Next
    Print #1, TESTE
    Close #1
    End Function

```

```

Function Open_MATFile() 'trhum 98 files

Open Filename For Input As #1

Dim offset(2000) As Long
Dim VI As Integer
Dim ii As Integer
Dim iii As Integer
Dim line$
Dim n As Integer
Dim aux5 As Integer

VI = 1
Do
    offset(VI) = Seek(1)
    Line Input #1, line$
    VI = VI + 1
Loop Until EOF(1)
offset(0) = VI - 1
For VI = 1 To offset(0)
    Seek #1, offset(VI)
    Line Input #1, line$
Next
aux5 = 2
    Seek #1, offset(2)
    Line Input #1, line$
    gMLISTNUMBER = txt2double(line$)
aux5 = aux5 + 2 '4
For VI = 1 To gMLISTNUMBER
    Seek #1, offset(aux5)
    aux5 = aux5 + 1
    Line Input #1, line$
    gMATNAME(VI) = line$
Next VI
aux5 = aux5 + 1 '8
For VI = 1 To gMLISTNUMBER
    Seek #1, offset(aux5)
    aux5 = aux5 + 1
    Line Input #1, line$
    gDENSITY(VI) = txt2double(line$)
Next VI
aux5 = aux5 + 1 '12
For VI = 1 To gMLISTNUMBER
    Seek #1, offset(aux5)
    aux5 = aux5 + 1
    Line Input #1, line$
    gHEAT(VI) = txt2double(line$)
Next VI
aux5 = aux5 + 1 '16
For VI = 1 To gMLISTNUMBER
    Seek #1, offset(aux5)
    aux5 = aux5 + 1
    Line Input #1, line$
    gHYGROSCOP(VI) = txt2double(line$)
Next VI
aux5 = aux5 + 1 '20
For VI = 1 To gMLISTNUMBER
    Seek #1, offset(aux5)
    aux5 = aux5 + 1
    Line Input #1, line$
    gSATURATION(VI) = txt2double(line$)
Next VI
aux5 = aux5 + 1 '24
For VI = 1 To gMLISTNUMBER
    Seek #1, offset(aux5)
    aux5 = aux5 + 1
    Line Input #1, line$
    gTLIST(VI) = txt2double(line$)

```

```

Next VI
aux5 = aux5 + 1 '28
For VI = 1 To gMLISTNUMBER
  Seek #1, offset(aux5)
  aux5 = aux5 + 1
  Line Input #1, line$
  gMLIST(VI) = txt2double(line$)
Next VI
aux5 = aux5 + 1 '32
For VI = 1 To gMLISTNUMBER
  For ii = 1 To gTLIST(VI)
    Seek #1, offset(aux5)
    aux5 = aux5 + 1
    Line Input #1, line$
    gTEMPERATURE(VI, ii) = txt2double(line$)
  Next ii
Next VI
aux5 = aux5 + 1 '42
For VI = 1 To gMLISTNUMBER
  For ii = 1 To gMLIST(VI)
    Seek #1, offset(aux5)
    aux5 = aux5 + 1
    Line Input #1, line$
    gMOISTURE(VI, ii) = txt2double(line$)
  Next ii
Next VI
aux5 = aux5 + 1
For VI = 1 To gMLISTNUMBER
  For ii = 1 To gTLIST(VI)
    For iii = 1 To gMLIST(VI)
      Seek #1, offset(aux5)
      aux5 = aux5 + 1
      Line Input #1, line$
      gDT(VI, ii, iii) = txt2double(line$)
      gDT(VI, 0, iii) = gMOISTURE(VI, iii)
    Next iii
    gDT(VI, ii, 0) = gTEMPERATURE(VI, ii)
  Next ii
Next VI
aux5 = aux5 + 1
For VI = 1 To gMLISTNUMBER
  For ii = 1 To gTLIST(VI)
    For iii = 1 To gMLIST(VI)
      Seek #1, offset(aux5)
      aux5 = aux5 + 1
      Line Input #1, line$
      gDW(VI, ii, iii) = txt2double(line$)
      gDW(VI, 0, iii) = gMOISTURE(VI, iii)
    Next iii
    gDW(VI, ii, 0) = gTEMPERATURE(VI, ii)
  Next ii
Next VI
aux5 = aux5 + 1
For VI = 1 To gMLISTNUMBER
  For ii = 1 To gTLIST(VI)
    For iii = 1 To gMLIST(VI)
      Seek #1, offset(aux5)
      aux5 = aux5 + 1
      Line Input #1, line$
      gCAPILLAR(VI, ii, iii) = txt2double(line$)
      gCAPILLAR(VI, 0, iii) = gMOISTURE(VI, iii)
    Next iii
    gCAPILLAR(VI, ii, 0) = gTEMPERATURE(VI, ii)
  Next ii
Next VI
aux5 = aux5 + 1
For VI = 1 To gMLISTNUMBER
  For ii = 1 To gTLIST(VI)
    For iii = 1 To gMLIST(VI)
      Seek #1, offset(aux5)
      aux5 = aux5 + 1
      Line Input #1, line$
      gCONDUCT(VI, ii, iii) = txt2double(line$)
      gCONDUCT(VI, 0, iii) = gMOISTURE(VI, iii)
    Next iii
    gCONDUCT(VI, ii, 0) = gTEMPERATURE(VI, ii)
  Next ii
Next VI
aux5 = aux5 + 2
For VI = 1 To gMLISTNUMBER
  Seek #1, offset(aux5)
  aux5 = aux5 + 1
  Line Input #1, line$
  gSorpCount(VI) = txt2double(line$)
Next VI
For VI = 1 To gMLISTNUMBER
  For ii = 1 To gSorpCount(VI)
    Seek #1, offset(aux5)

```

```

        aux5 = aux5 + 1
        Line Input #1, line$
        GSorpHR(VI, ii) = txt2double(line$)
    Next
    For ii = 1 To gSorpCount(VI)
        Seek #1, offset(aux5)
        aux5 = aux5 + 1
        Line Input #1, line$
        GSorpWHR(VI, ii) = txt2double(line$)
    Next
Next
aux5 = aux5
    Seek #1, offset(aux5)
    Line Input #1, line$
    TESTE = line$
Close #1
End Function

```

```

Function Open_WET98File() 'trhum 98 files
Open Filename For Input As #1
Dim offset(500) As Long
Dim i As Integer
Dim lines As Integer
Dim aux1 As Integer
i = 1
Do
    offset(i) = Seek(1)
    Line Input #1, line$
    i = i + 1
Loop Until EOF(1)
offset(0) = i - 1
lines = offset(0)

Seek #1, offset(1)
    Line Input #1, line$
    gN = txt2double(line$)

For aux1 = 3 To lines
    Seek #1, offset(aux1)
    wetline = aux1 - 2
    Line Input #1, line$
    extractcolumn
    passweather98
Next
Close #1
End Function

```

```

Function Save_WET98File() 'trhum 98 files

Dim i As Integer
On Error Resume Next
Open Filename For Output As #1
On Error Resume Next

Print #1, gN
Print #1, "Time(h);TempOut(°C);RHOut(°);Rad(W/m2);TempIn(°C);RHIn(°); gstep; Rain(1/0)"
For i = 1 To gN
    Print #1, gTime(i); ";"; gTempOut(i); ";"; gRHOut(i); ";"; grad(i); ";"; gTempIn(i); ";"; gRHIn(i);
";"; gstep(i); ";"; gRain(i)
Next i
Close #1
End Function

```

B.2.6. Módulo “conv.bas”

```

Attribute VB_Name = "conv"
Option Explicit

```

```

Static Function cv_AtmmHg(invalue)
    cv_AtmmHg = invalue * 760
End Function

```

```

Static Function cv_AtmmB(invalue)
    cv_AtmmB = invalue * 1013.25
End Function

```

```

Static Function cv_AtmmPA(invalue)
    cv_AtmmPA = invalue * 101325
End Function

```

```

Static Function cv_AtmmH2o(invalue)
    cv_AtmmH2o = invalue * 101325 / 9806

```

```

End Function

Static Function cv_Atm_nm2(invalue)
  cv_Atm_nm2 = invalue * 101325
End Function

Static Function cv_Atm_bar(invalue)
  cv_Atm_bar = invalue * 1.01325
End Function

Static Function cv_mmhg_pa(invalue)
  cv_mmhg_pa = invalue * 1.333 * 10 ^ (-2)
End Function

Static Function cv_mb_pa(invalue)
  cv_mb_pa = invalue * 100
End Function

Static Function cv_mb_bar(invalue)
  cv_mb_bar = invalue * 10 ^ (-3)
End Function

Static Function cv_mb_mh2o(invalue)
  cv_mb_mh2o = invalue * 100 / 9806
End Function

Static Function cv_bar_pa(invalue)
  cv_bar_pa = invalue * 10 ^ 5
End Function

Static Function cv_bar_mb(invalue)
  cv_bar_mb = invalue * 10 ^ 3
End Function

Static Function cv_bar_atm(invalue)
  cv_bar_atm = invalue * (10 ^ 5) / 101325
End Function

Static Function cv_bar_mh2o(invalue)
  cv_bar_mh2o = invalue * (10 ^ 5) / 9806
End Function

Static Function cv_bar_nm2(invalue)
  cv_bar_nm2 = invalue * 10 ^ 5
End Function

Static Function cv_bar_mmhg(invalue)
  cv_bar_mmhg = invalue * (10 ^ 2) / 1.333 * 9806
End Function

Static Function cv_nm2_pa(invalue)
  cv_nm2_pa = invalue
End Function

Static Function cv_pa_atm(invalue)
  cv_pa_atm = invalue * 1 / 101325
End Function

Static Function cv_pa_mmhg(invalue)
  cv_pa_mmhg = invalue * 10 ^ 2 / 1.333
End Function

Static Function cv_pa_mb(invalue)
  cv_pa_mb = invalue * 0.01
End Function

Static Function cv_pa_bar(invalue)
  cv_pa_bar = invalue * 10 ^ (-5)
End Function

Static Function cv_pa_mh2o(invalue)
  cv_pa_mh2o = invalue * (1 / 9806)
End Function

Static Function cv_pa_nm2(invalue)
  cv_pa_nm2 = invalue
End Function

Static Function cv_mh2o_pa(invalue)
  cv_mh2o_pa = invalue * 9806
End Function

Static Function cv_mh2o_atm(invalue)
  cv_mh2o_atm = invalue * 9806 / 101325

```

```

End Function

Static Function cv_mh2o_mb(invalue)
  cv_mh2o_mb = invalue * 9806 * 100
End Function

Static Function cv_mh2o_mmhg(invalue)
  cv_mh2o_mmhg = invalue * (9806 * 10 ^ 2) / 1.333
End Function

Static Function cv_mh2o_nm2(invalue)
  cv_mh2o_nm2 = invalue * 9806
End Function

Static Function cv_mh2o_bar(invalue)
  cv_mh2o_bar = invalue * 9806 * 10 ^ 5
End Function

Static Function cv_mmhg_atm(invalue)
  cv_mmhg_atm = invalue * 1 / 760
End Function

Static Function cv_mb_atm(invalue)
  cv_mb_atm = invalue * 1 / 1013.25
End Function

Static Function cv_mb_mmhg(invalue)
  cv_mb_mmhg = invalue * 1 / 1.33322
End Function

Static Function cv_mb_nm2(invalue)
  cv_mb_nm2 = invalue * 100
End Function

Static Function cv_mmhg_mb(invalue)
  cv_mmhg_mb = invalue * 1.33322
End Function

Static Function cv_mmhg_bar(invalue)
  cv_mmhg_bar = invalue * 1.33322 * 10 ^ 3
End Function

Static Function cv_mmhg_nm2(invalue)
  cv_mmhg_nm2 = invalue * 1.33322 * 10 ^ (-2)
End Function

Static Function cv_mmhg_mh2o(invalue)
  cv_mmhg_mh2o = invalue * 1.33322 / 9806 * 10 ^ 2
End Function

Static Function cv_nm2_atm(invalue)
  cv_nm2_atm = invalue * 1 / 101325
End Function

Static Function cv_nm2_mmhg(invalue)
  cv_nm2_mmhg = invalue * 10 ^ 2 / 1.333
End Function

Static Function cv_nm2_mb(invalue)
  cv_nm2_mb = invalue * 0.01
End Function

Static Function cv_nm2_bar(invalue)
  cv_nm2_bar = invalue * 10 ^ (-5)
End Function

Static Function cv_nm2_mh2o(invalue)
  cv_nm2_mh2o = invalue * (1 / 9806)
End Function

```

B.3 Listagem de “Forms”

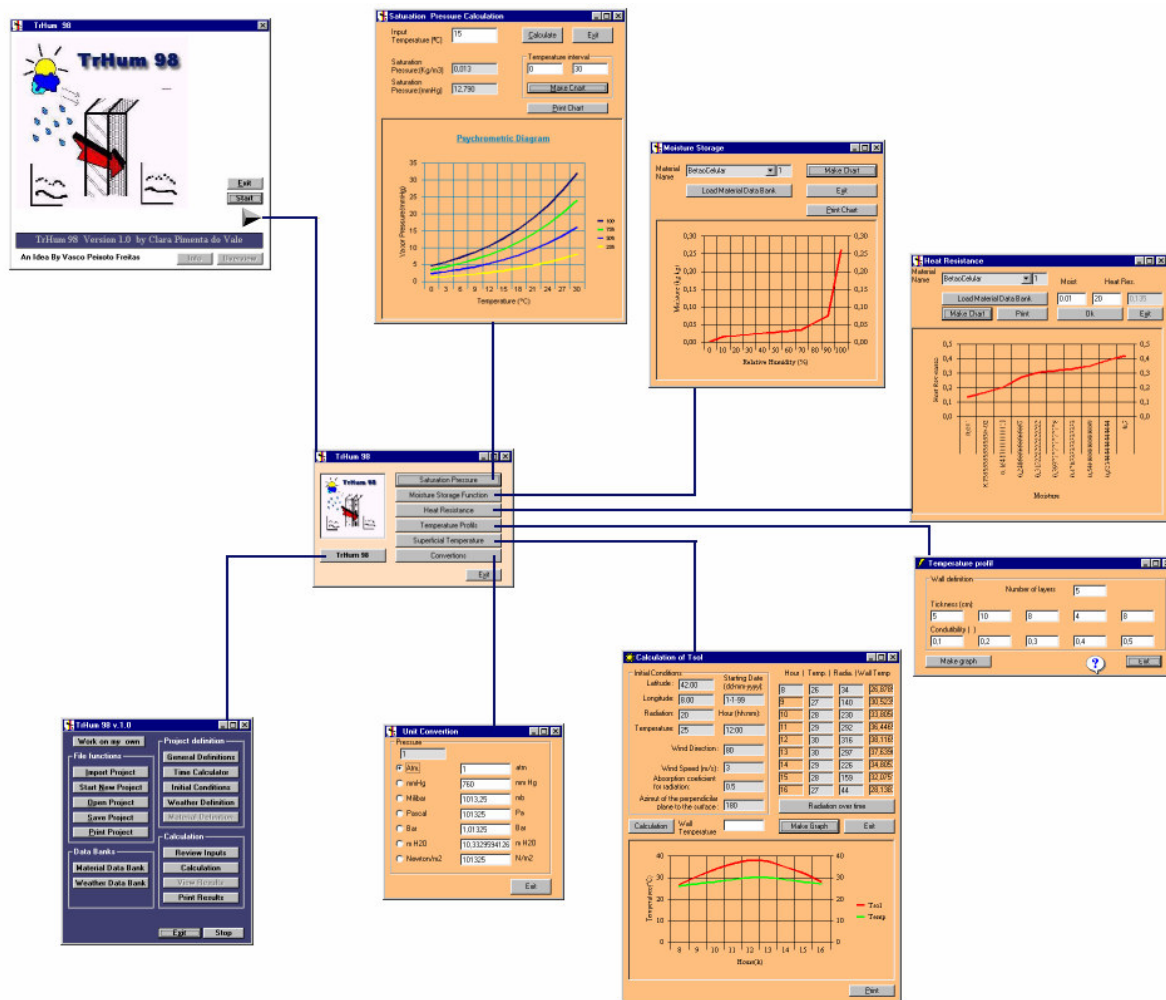


Fig. 81 - Esquema de ligações entre os diferentes “forms”

B.3.1. "Form início"

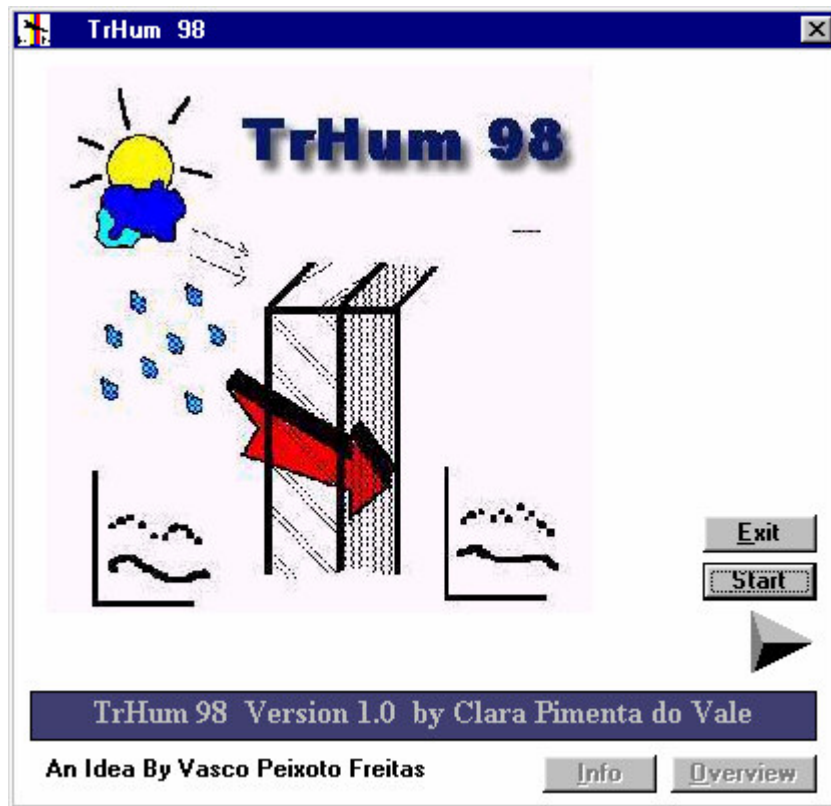


Fig. 82 – "Form início"

Option Explicit

```
Private Sub CmdExit_Click()  
    End  
End Sub
```

```
Private Sub cmdStart_Click()  
    Unload Me 'unload this form  
    frmMain.Show  
    Frmmodules.Show  
    previousform = "frminicio"  
End Sub
```

```
Private Sub Form_Load()  
    CleanValue  
End Sub
```

```
Private Sub imgNext_Click()  
    Unload Me 'unload this form  
    frmMain.Show  
    Frmmodules.Show  
    previousform = "frminicio"  
End Sub
```

B.3.2. "Form frmmodules"

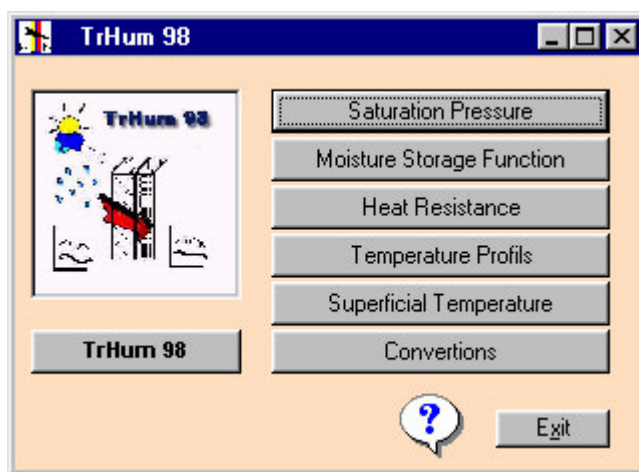


Fig. 83 – "Form frmmodules"

Option Explicit

```
Private Sub cmdconv_Click()  
    Unload Me  
    frmConvert.Show  
End Sub
```

```
Private Sub cmdExit_Click()  
    Unload Me  
End  
End Sub
```

```
Private Sub cmdheat_Click()  
    Unload Me  
    frmHeat.Show  
End Sub
```

```
Private Sub cmdmoisture_Click()  
    Unload Me  
    frmMoisture.Show  
End Sub
```

```
Private Sub cmdsat_Click()  
    Unload Me  
    frmSat.Show  
End Sub
```

```
Private Sub cmdsuptemp_Click()  
    Unload Me  
    frmsol.Show  
End Sub
```

```
Private Sub cmdtemp_Click()  
    Unload Me  
    frmtemp.Show  
End Sub
```

```
Private Sub cmdtrhum_Click()  
    Unload Me 'unload this form  
    frmMain.Show  
    frmMainWindow.Show  
End Sub
```

B.3.3. "Form frmsat"

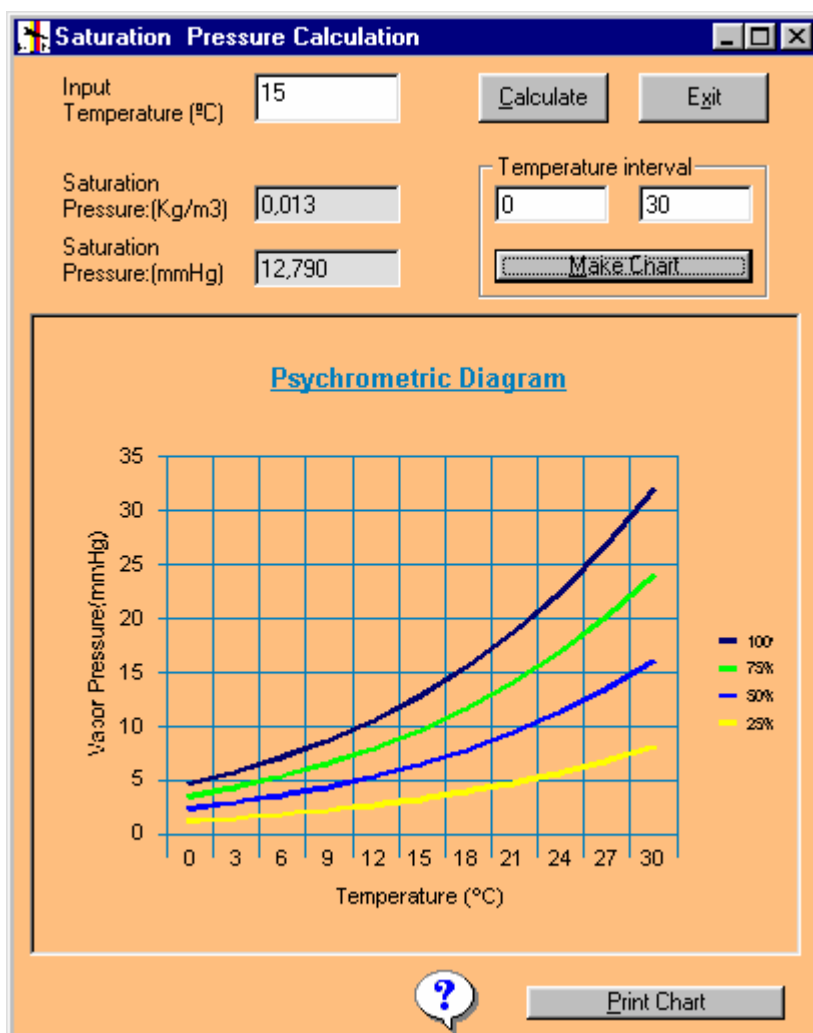


Fig. 84 – "Form frmsat"

Option Explicit

```
Dim Temp As Double
Dim PVS As Double
Dim PVSHG As Double
Dim hr(11) As Double
Dim temper(11) As Double
Dim pvshr(11) As Double
Dim ii As Integer
Dim t1, t2, dxt As Double
Dim axisId
```

```
Sub calcul()
    For ii = 0 To 10
        hr(ii) = ii * 10
        pvshr(ii) = PVSHG * hr(ii) / 100
    Next
End Sub
```

```
Sub calcultemp()
    t1 = txtt1.Text
    t2 = txtt2.Text
    dxt = (t2 - t1) / 10
    temper(0) = t1
    For ii = 0 To 10
        temper(ii) = temper(0) + ii * dxt
    Next
End Sub
```

```

        pvshr(ii) = PSATURAmhg(temper(ii))
    Next
End Sub

```

```

Function charthr()
    Dim Column As Integer
    Dim Row As Integer

    With frmSat2.MSChart1
        .Visible = True
        .chartType = VtChChartType2dLine
        .Title = True
        .TitleText = "Vapour Pressure"
        .ColumnCount = 1
        .RowCount = 11
        .ShowLegend = True
    End With

    Column = 1
    For Row = 1 To 11
        .Row = Row
        .Column = Column
        .Data = pvshr(Row - 1)
        .RowLabel = (Row - 1) * 10
    Next

    For axisId = VtChAxisIdX To VtChAxisIdY
        With MSChart1.Plot.Axis(axisId, 1).AxisTitle
            .Visible = True
            Select Case axisId
                Case 0
                    .Text = "Relative Humidity (%)"
                Case 1
                    .Text = "Vapor Pressure(mmHg) "
                Case 2
                    .Text = ""
            End Select
        End With
    Next
End Function

```

```

Function charttemp()
    Dim Column As Integer
    Dim Row As Integer

    With frmSat2.MSChart1
        .Visible = True
        .chartType = VtChChartType2dLine
        .Title = True
        .TitleText = "Psychrometric Diagram"
        .ColumnCount = 4
        .RowCount = 11
        .ShowLegend = True
    End With

    Column = 1
    For Row = 1 To 11
        .Row = Row
        .Column = Column
        .ColumnLabel = "100%"
        .Data = pvshr(Row - 1)
        .RowLabel = temper(Row - 1)
    Next

    Column = 2
    For Row = 1 To 11
        .Row = Row
        .Column = Column
        .ColumnLabel = "75%"
        .Data = (pvshr(Row - 1) * 0.75)
    Next

    Column = 3
    For Row = 1 To 11
        .Row = Row
        .Column = Column
        .ColumnLabel = "50%"
        .Data = (pvshr(Row - 1) * 0.5)
    Next

    Column = 4
    For Row = 1 To 11
        .Row = Row
        .Column = Column
        .ColumnLabel = "25%"
        .Data = (pvshr(Row - 1) * 0.25)
    Next

    For axisId = VtChAxisIdX To VtChAxisIdY
        With MSChart1.Plot.Axis(axisId, 1).AxisTitle
            .Visible = True
            Select Case axisId

```

```

        Case 0
        .Text = "Temperature (°C)"
        Case 1
        .Text = "Vapor Pressure (mmHg)"
        Case 2
        .Text = ""
        End Select
    End With
Next
End Function

Private Sub cmdCalculate_Click()
    frmSat2.Height = 7545
    Temp = txttemp.Text
    PVS = PSATURA(Temp)
    PVSHG = PSATURAmHg(Temp)
    txtPVS.Text = Format(PVS, "###,##0.000")
    pvsmhg = Format(PVSHG, "###,##0.000")
    calcul
    charthr
End Sub

```

B.3.4. "Form frmmoisture"

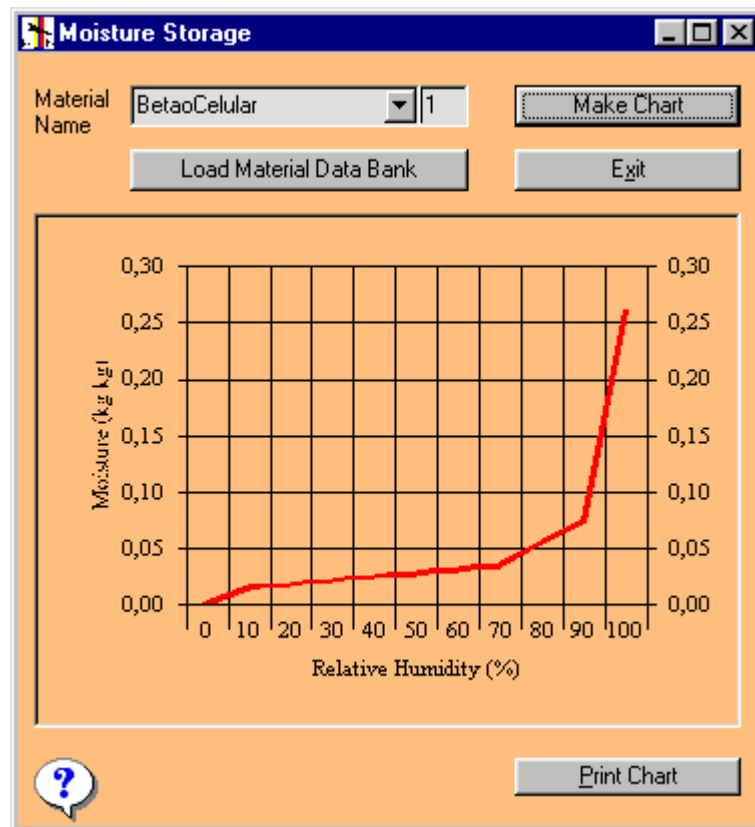


Fig. 85 – "Form frmmoisture"

Option Explicit

```

Dim i As Integer
Dim n As Integer
Dim Tick As Double
Dim Tickt As Double
Dim lTick(3) As Double
Dim NumberOfLayer As Integer
Dim Matcod(5)
Dim whr(11)

```

```

Function prepare()
    For i = 0 To 10
        whr(i) = Higriscop2(i * 10)
        Print #3, whr(i)
    Next i
End Function

```

```

Next i
End Function

```

```

Function chart()
    Dim Column As Integer
    Dim Row As Integer
    Dim axisId
    With frmMoisture.MSChart1
        .Visible = True
        .chartType = VtChChartType2dLine
        .Title = True
        .TitleText = ""
        .ColumnCount = 1
        .RowCount = 11
        .ShowLegend = False
    End With
    Column = 1
    For Row = 1 To 11
        .Row = Row
        .Column = Column
        .Data = whr(Row - 1)
        .RowLabel = ((Row - 1) * 10)
        .ColumnLabel = ""
    Next Row
End With
For axisId = VtChAxisIdX To VtChAxisIdY
    With MSChart1.Plot.Axis(axisId, 1).AxisTitle
        .Visible = True
        Select Case axisId
            Case 0
                .Text = "Relative Humidity (%)"
            Case 1
                .Text = "Moisture (kg/kg)"
            Case 2
                .Text = ""
        End Select
    End With
Next
End Function

```

```

Private Sub Combomaterial_Click()
    On Error Resume Next
    gMATERIALNAME = Combomaterial.Text
    TxtCode.Text = Matcod(Combomaterial.ListIndex)
    gmaterialcode = Matcod(Combomaterial.ListIndex)
    On Error Resume Next
End Sub

```

```

Private Sub cmdacmat_Click()
    frmMoisture.Height = 6165
    creatProjectdatabank
    prepare
    chart
End Sub

```

```

Private Sub CmdLoadDBK_Click()
    Dim sFile As String
    With dlgCommonDialog
        .Filter = "Files (*.dbk)|*.dbk"
        .ShowOpen
        If Len(.Filename) = 0 Then
            Exit Sub
        End If
        sFile = .Filename
        Filename = sFile
    End With
    Open_MATFile
    Combomaterial.Clear
    gMATNAME(0) = "none"
    For i = 0 To gMLISTNUMBER
        Matcod(i) = i
        Combomaterial.AddItem gMATNAME(i)
    Next i
    cmdacmat.Enabled = True
End Sub

```

```

Private Sub Combomaterial_Change()
    On Error Resume Next
    gMATERIALNAME = Combomaterial.Text
    TxtCode.Text = Matcod(Combomaterial.ListIndex)
    gmaterialcode = Matcod(Combomaterial.ListIndex)
    On Error Resume Next
End Sub

```

B.3.5. "Form frmheat"

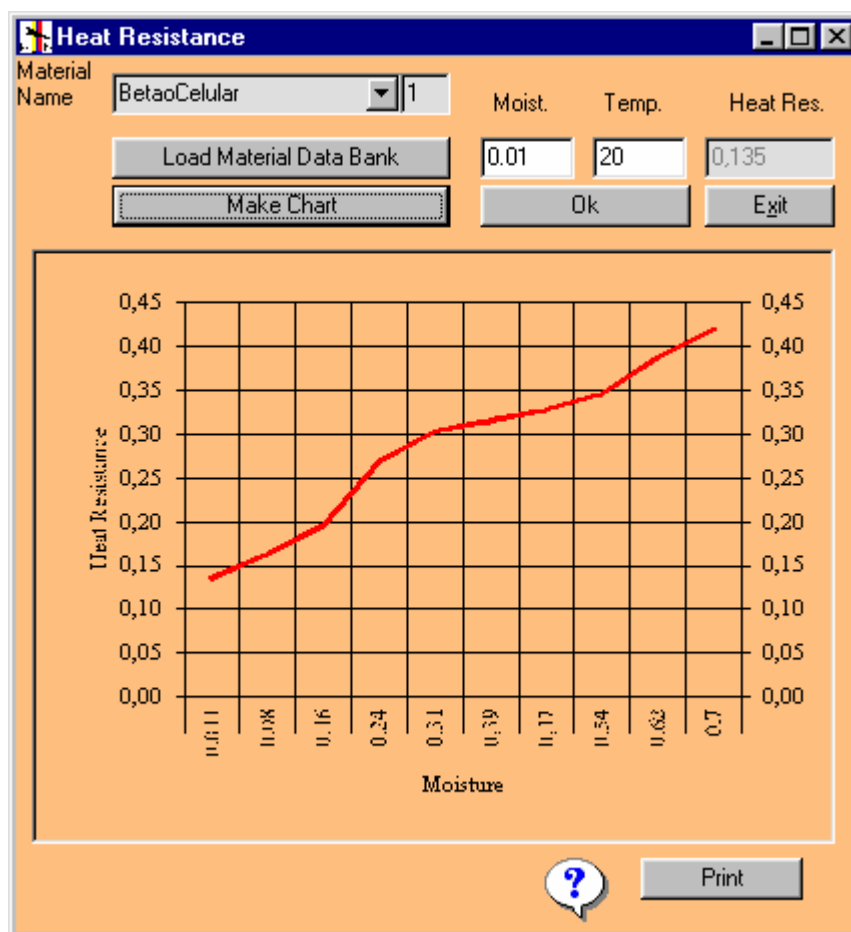


Fig. 86 – "Form frmheat"

Option Explicit

```

Dim i As Integer
Dim n As Integer
Dim Matcod(5)
Dim lambda(11) As Double
Dim moist(11) As Double
Dim s As Double

Function prepare()
moist(1) = gSMOISTURE(1)
moist(10) = gSMOISTURE(gSMLIST)
Temp = txt2double(Text3.Text)
s = (moist(10) - moist(1)) / 9
For i = 2 To 9
    moist(i) = moist(1) + (i - 1) * s
    moist(i) = Int(moist(i) * 100)
    moist(i) = moist(i) / 100
Next
For i = 1 To 10
    Temp = txt2double(Text3.Text)
    hum = moist(i)
    tipo = 4
    tpassvalue
    lambda(i) = cofenon(Temp, hum)
    Print #3, lambda(i); moist(i)
Next i
End Function

Function chart()
    Dim Column As Integer
    Dim Row As Integer

```

```

    Dim axisId
With frmHeat.MSChart1
    .Visible = True
    .chartType = VtChChartType2dLine
    .Title = False
    .TitleText = ""
    .ColumnCount = 1
    .RowCount = 10
    .ShowLegend = False
Column = 1
    For Row = 1 To 10
        .Row = Row
        .Column = Column
        .Data = lambda(Row)
        .RowLabel = moist(Row)
        .ColumnLabel = ""
    Next
End With
For axisId = VtChAxisIdX To VtChAxisIdY
    With MSChart1.Plot.Axis(axisId, 1).AxisTitle
        .Visible = True
        Select Case axisId
            Case 0
                .Text = "Moisture"
            Case 1
                .Text = "Superficial conductance"
            Case 2
                .Text = ""
        End Select
    End With
Next
End Function

```

```

Private Sub Combomaterial_Click()
On Error Resume Next
    gMATERIALNAME = Combomaterial.Text
    TxtCode.Text = Matcod(Combomaterial.ListIndex)
    gmaterialcode = Matcod(Combomaterial.ListIndex)
On Error Resume Next
End Sub

```

```

Private Sub cmdacmat_Click()
    creatProjectdatabank
    prepare
    chart
End Sub

```

```

Private Sub CmdLoadDBK_Click()
    Dim sFile As String
    With dlgCommonDialog
        .Filter = "Files (*.dbk)|*.dbk"
        .ShowOpen
        If Len(.Filename) = 0 Then
            Exit Sub
        End If
        sFile = .Filename
        Filename = sFile
    End With
    Open_MATFile
    Combomaterial.Clear
    gMATNAME(0) = "nome"
    For i = 0 To gMLISTNUMBER
        Matcod(i) = i
        Combomaterial.AddItem gMATNAME(i)
    Next
    cmdacmat.Enabled = True
End Sub

```

```

Private Sub Combomaterial_Change()
On Error Resume Next
    gMATERIALNAME = Combomaterial.Text
    TxtCode.Text = Matcod(Combomaterial.ListIndex)
    gmaterialcode = Matcod(Combomaterial.ListIndex)
On Error Resume Next
End Sub

```

```

Private Sub Command1_Click()
    creatProjectdatabank
    hum = txt2double(Text1.Text)
    Temp = txt2double(Text3.Text)
    tipo = 4
    tpassvalue
    Text2.Text = interpolation(Temp, hum)
End Sub

```


B.3.6. "Form frmtemp"

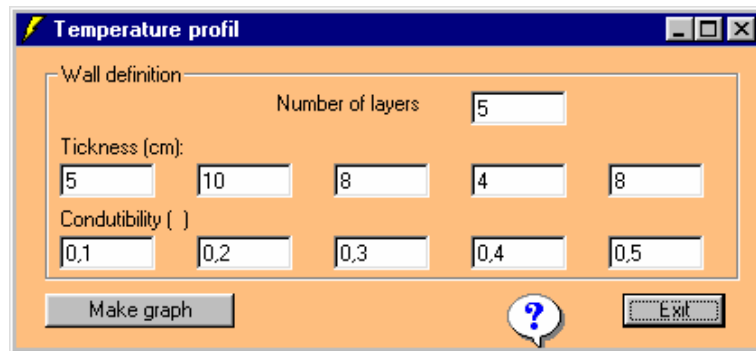


Fig. 87 – "Form frmtemp"

```
Option Explicit
Dim nl As Integer
Dim tt1, tt2, tt3, tt4, tt5 As Double
Dim cc1, cc2, cc3, cc4, cc5 As Double
Dim i As Integer
```

```
Sub visibility()
Select Case nl
Case 1
t1.Enabled = True
t2.Enabled = False
t3.Enabled = False
t4.Enabled = False
t5.Enabled = False
c1.Enabled = True
c2.Enabled = False
c3.Enabled = False
c4.Enabled = False
c5.Enabled = False
Case 2
t1.Enabled = True
t2.Enabled = True
t3.Enabled = False
t4.Enabled = False
t5.Enabled = False
c1.Enabled = True
c2.Enabled = True
c3.Enabled = False
c4.Enabled = False
c5.Enabled = False
Case 3
t1.Enabled = True
t2.Enabled = True
t3.Enabled = True
t4.Enabled = False
t5.Enabled = False
c1.Enabled = True
c2.Enabled = True
c3.Enabled = True
c4.Enabled = False
c5.Enabled = False
Case 4
t1.Enabled = True
t2.Enabled = True
t3.Enabled = True
t4.Enabled = True
t5.Enabled = False
c1.Enabled = True
c2.Enabled = True
c3.Enabled = True
c4.Enabled = True
c5.Enabled = False
Case 5
t1.Enabled = True
t2.Enabled = True
t3.Enabled = True
t4.Enabled = True
t5.Enabled = True
c1.Enabled = True
c2.Enabled = True
```

```

c3.Enabled = True
c4.Enabled = True
c5.Enabled = True
End Select
End Sub

Private Sub txtnl_LostFocus()
If txtnl.Text = "" Then txtnl.Text = 1
nl = txtnl
If nl > 5 Then nl = 5
If nl < 1 Then nl = 1
txtnl.Text = nl
visibility
End Sub

```

B.3.7. "Form frmsol"

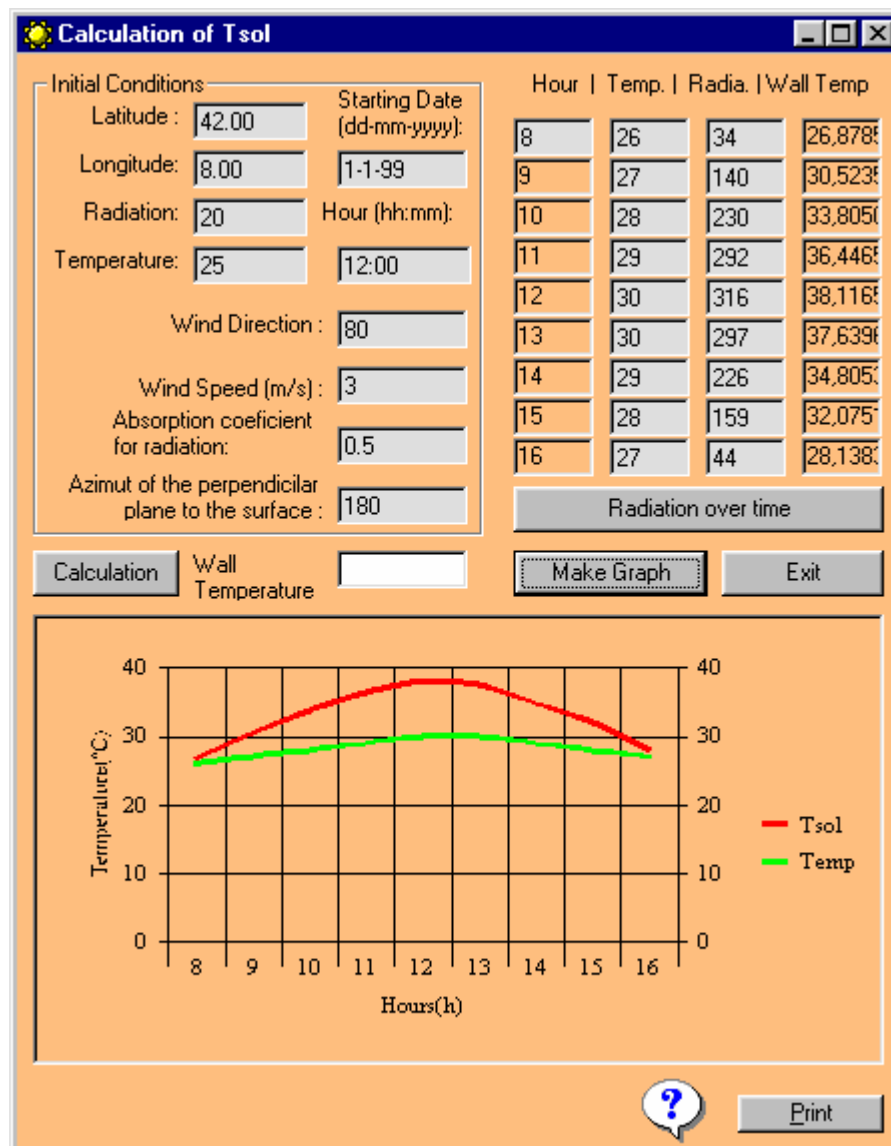


Fig. 88 – "Form frmsol"

```

Option Explicit

Dim txtHoursS
Dim txtYearS
Dim txtDays
Dim timeday
Dim timeyear
Dim inte As Integer
Dim TSOLtemp(10) As Double
Dim tempi(10) As Double
Dim chouri(10) As Double
Dim axisId
Dim i As Integer

Function chart()
    Dim Column As Integer
    Dim Row As Integer
    With frmsol.MSChart1
        .Visible = True
        .chartType = VtChChartType2dLine
        .Title = True
        .TitleText = ""
        .ColumnCount = 2
        .RowCount = 9
        .ShowLegend = True
    Column = 1
        For Row = 1 To 9
            .Row = Row
            .Column = Column
            .Data = TSOLtemp(Row)
            .RowLabel = (chouri(Row))
            .ColumnLabel = "Tsol"
        Next
    Column = 2
        For Row = 1 To 9
            .Row = Row
            .Column = Column
            .Data = tempi(Row)
            .ColumnLabel = "Temp"
        Next
    End With
    For axisId = VtChAxisIdX To VtChAxisIdY
        With MSChart1.Plot.Axis(axisId, 1).AxisTitle
            .Visible = True
            Select Case axisId
                Case 0
                    .Text = "Hours (h) "
                Case 1
                    .Text = "Temperature (°C) "
                Case 2
                    .Text = ""
            End Select
        End With
    Next
End Function

```

```

Sub calculation()
    passvalue
    temp_out = txt2double(txttemp)
    Rad_ = txt2double(txtradiation)
    chour = gHourS
    cday = gDayS
    cyear = gYears
    TSOL = SOL
    txtsol.Text = TSOL
End Sub

```

```

Sub calculationtime()

    passvalue
    chour = 0
    cday = gDayS
    cyear = gYears
    chour = txt2double(Text1) - 1
    For i = 1 To 9
        temp_out = txt2double(Text5(i - 1))
        tempi(i) = temp_out
        Rad_ = txt2double(Text3(i - 1))
        chour = chour + 1
        chouri(i) = chour
        TSOLtemp(i) = SOL
        Labell(i - 1).Caption = TSOLtemp(i)
    Next
End Sub

```

```

Function passvalue()
    gLatitude = txt2double(txtLatitude.Text)
    gLongitude = txt2double(txtLongitude.Text)
    gWindDirection = txt2double(txtWindDirection.Text)
    gWindSpeed = txt2double(txtWindSpeed.Text)
    gAzimute = txt2double(txtAzimut.Text)
    galpha = txt2double(txtalpha.Text)
    gDayS = Format(txtDateS, "y")
    gYears = Format(txtDateS, "yyyy")
    gHourS = Format(txthourss, "hh")
End Function

```

```

Private Sub cmdovertime_Click()
    calculationtime
    Command2.Enabled = True
End Sub

```

```

Private Sub Command1_Click()
    calculation
End Sub

```

```

Private Sub Text1_Change()
    For i = 1 To 8
        Label5(i - 1).Caption = txt2double(Text1) + i
    Next
End Sub

```

```

Private Sub txthourss_LostFocus()
    If Not IsDate(txthourss) Then
        txthourss.Text = "12:00"
    End If
End Sub

```

B.3.8. "Form frmconvert"

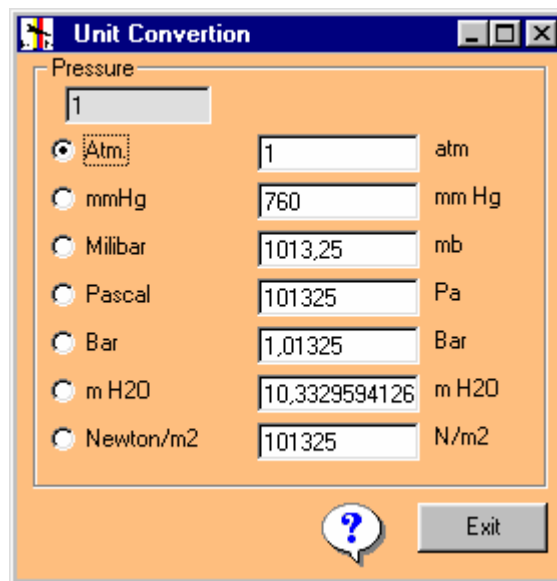


Fig. 89 – "Form frmconvert"

Option Explicit

```

Private Sub cmdExit_Click()
    Unload Me
    Frmmodules.Show
End Sub

```

```

Private Sub Optatm_Click()

    Dim invalue
    invalue = txtinvalue.Text
    txtatm.Text = invalue
    txtmmhg.Text = cv_Atm_mmHg(invalue)
    txtmb.Text = cv_Atm_mb(invalue)

```

```
txtpascal.Text = cv_AtM_PA(invalue)
txtbar.Text = cv_AtM_bar(invalue)
txtmh2o.Text = cv_AtM_mH2o(invalue)
txtnm2.Text = cv_AtM_nm2(invalue)
End Sub
```

```
Private Sub Optbar_Click()
Dim invalue
invalue = txtinvalue.Text
txtatm.Text = cv_bar_atm(invalue)
txtmmhg.Text = cv_bar_mmhg(invalue)
txtmb.Text = cv_bar_mb(invalue)
txtpascal.Text = cv_bar_pa(invalue)
txtbar.Text = invalue
txtmh2o.Text = cv_bar_mh2o(invalue)
txtnm2.Text = cv_bar_nm2(invalue)
End Sub
```

```
Private Sub Optmb_Click()
Dim invalue
invalue = txtinvalue.Text
txtatm.Text = cv_mb_atm(invalue)
txtmmhg.Text = cv_mb_mmhg(invalue)
txtmb.Text = invalue
txtpascal.Text = cv_mb_pa(invalue)
txtbar.Text = cv_mb_bar(invalue)
txtmh2o.Text = cv_mb_mh2o(invalue)
txtnm2.Text = cv_mb_nm2(invalue)
End Sub
```

```
Private Sub Optmh2o_Click()
Dim invalue
invalue = txtinvalue.Text
txtatm.Text = cv_mh2o_atm(invalue)
txtmmhg.Text = cv_mh2o_atm(invalue)
txtmb.Text = cv_mh2o_mb(invalue)
txtpascal.Text = cv_mh2o_pa(invalue)
txtbar.Text = cv_mh2o_bar(invalue)
txtmh2o.Text = invalue
txtnm2.Text = cv_mh2o_nm2(invalue)
End Sub
```

```
Private Sub Optmmhg_Click()
Dim invalue
invalue = txtinvalue.Text
txtatm.Text = cv_mmhg_atm(invalue)
txtmmhg.Text = invalue
txtmb.Text = cv_mmhg_mb(invalue)
txtpascal.Text = cv_mmhg_pa(invalue)
txtbar.Text = cv_mmhg_bar(invalue)
txtmh2o.Text = cv_mmhg_mh2o(invalue)
txtnm2.Text = cv_mmhg_nm2(invalue)
End Sub
```

```
Private Sub Optnm2_Click()
Dim invalue
invalue = txtinvalue.Text
txtatm.Text = cv_nm2_atm(invalue)
txtmmhg.Text = cv_nm2_mmhg(invalue)
txtmb.Text = cv_nm2_mb(invalue)
txtpascal.Text = cv_nm2_mb(invalue)
txtbar.Text = cv_nm2_bar(invalue)
txtmh2o.Text = cv_nm2_mh2o(invalue)
txtnm2.Text = invalue
End Sub
```

```
Private Sub Optpa_Click()
Dim invalue
invalue = txtinvalue.Text
txtatm.Text = cv_pa_atm(invalue)
txtmmhg.Text = cv_pa_mmhg(invalue)
txtmb.Text = cv_pa_mb(invalue)
txtpascal.Text = invalue
txtbar.Text = cv_pa_bar(invalue)
txtmh2o.Text = cv_pa_mh2o(invalue)
txtnm2.Text = cv_pa_nm2(invalue)
End Sub
```

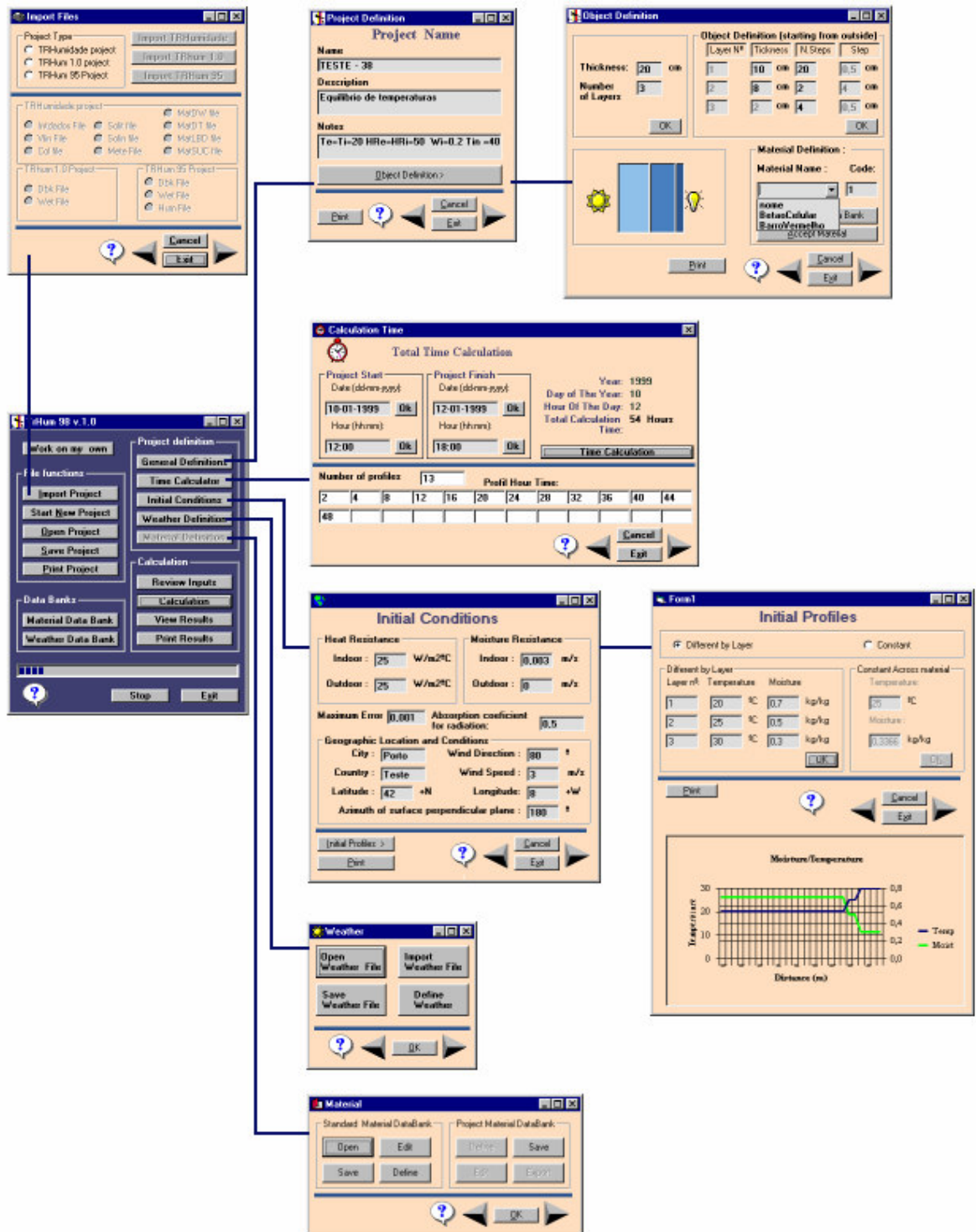


Fig. 90 - Esquema de ligações entre "forms"

B.3.9. "Form frmmain"

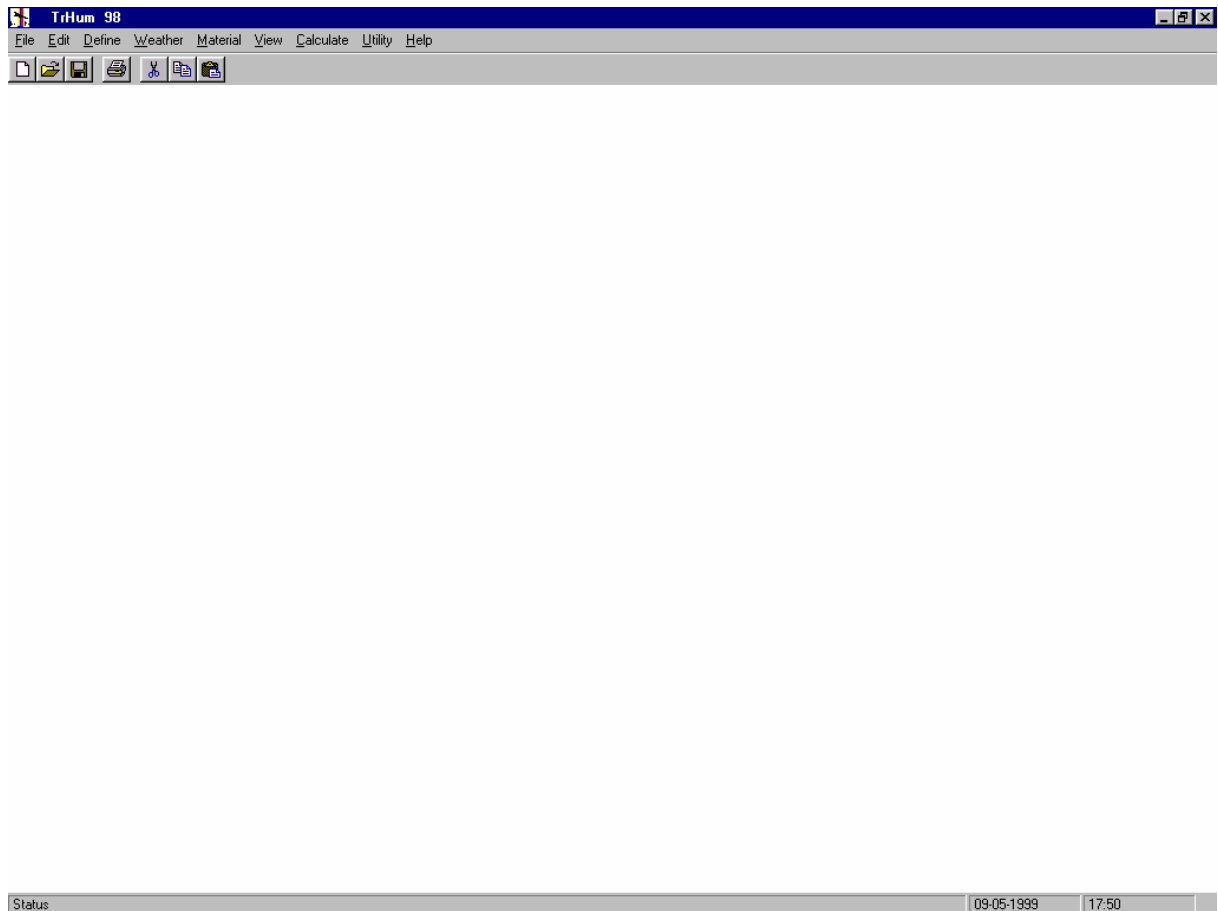


Fig. 91 – "Form frmmain"

Option Explicit

```
Private Sub Form_Load()  
    Me.Left = GetSetting(App.Title, "Settings", "MainLeft", 1000)  
    Me.Top = GetSetting(App.Title, "Settings", "MainTop", 1000)  
    Me.Width = GetSetting(App.Title, "Settings", "MainWidth", 6500)  
    Me.Height = GetSetting(App.Title, "Settings", "MainHeight", 6500)  
    Dim i As Integer          ' Counter variable.  
    Show  
    ChDir App.Path  
    FState.Dirty = False  
End Sub
```

```
Private Sub mnuCalculatePressure_Click()  
    frmSat2.Show  
    previousform = "main"  
End Sub
```

```
Private Sub mnuCalculateTime_Click()  
    FrmTimeR.Show  
End Sub
```

```
Private Sub mnuDBKSave_Click()  
    Dim sFile As String  
    With dlgCommonDialog  
        .Filter = "Data Bank files (*.dbk)|*.dbk"  
        .ShowSave  
        sFile = .Filename  
    End With  
    Filename = sFile  
    Save_MATFile  
End Sub
```

```
Private Sub mnuDefineConditions_Click()  
    FrmInitialC.Show  
End Sub
```

```
Private Sub mnuDefineObject_Click()  
    frmObjectDefinition.Show  
End Sub
```

```
Private Sub mnuDefineProfiles_Click()  
    frminitialprofil.Show  
End Sub
```

```
Private Sub mnuDefineProject_Click()  
    FrmObjectName.Show  
End Sub
```

```
Private Sub mnuDefineTime_Click()  
    FrmTimeR.Show  
End Sub
```

```
Private Sub mnudefineweather_Click()  
    frmMainWeather.Show  
End Sub
```

```
Private Sub mnuFileImport_Click()  
    frmImport.Show  
End Sub
```

```
Private Sub mnuHelpAbout_Click()  
    frmAbout.Show vbModal, Me  
End Sub
```

```
Private Sub mnuMDKOpen_Click()  
    Dim sFile As String  
    With dlgCommonDialog  
        .Filter = "Files (*.dbk)|*.dbk"  
        .ShowOpen  
        If Len(.Filename) = 0 Then  
            Exit Sub  
        End If  
        sFile = .Filename  
        Filename = sFile  
    End With  
    Open_MATFile  
End Sub
```

```
Private Sub mnuMoist_Click()  
    frmMoisture.Show  
End Sub
```

```
Private Sub mnuutilheat_Click()  
    frmHeat.Show  
End Sub
```

```
Private Sub mnuUtilityConversion_Click()  
    frmConvert.Show  
    previousform = "main"  
End Sub
```

```
Private Sub mnuutilsat_Click()  
    frmSat.Show  
End Sub
```

```
Private Sub mnuViewConditions_Click()  
    FrmInitialC.Show  
End Sub
```

```
Private Sub mnuviewmain_Click()  
    frmMainWindow.Show  
End Sub
```

```
Private Sub mnuViewMaterial_Click()  
    MAterial.Show  
End Sub
```

```
Private Sub mnuViewProject_Click()  
    FrmObjectName.Show  
End Sub
```

```
Private Sub mnuViewWeather_Click()  
    frmweather.Show  
End Sub
```

```

Private Sub mnuWeatherDynamic_Click()
    frmDynamicWeather.Show
End Sub

Private Sub mnuWeatherGenerate_Click()
    frmGeneratedWeather.Show
End Sub

Private Sub mnuweatherimport_Click()
    frmImport.Show
End Sub

Private Sub mnuWeatherOpen_Click()
    Dim sFile As String
    With dlgCommonDialog
        .Filter = "Files (*.wet)|*.wet"
        .ShowOpen
        If Len(.Filename) = 0 Then
            Exit Sub
        End If
        sFile = .Filename
        Filename = sFile
    End With
    Open_WET98File
End Sub

Private Sub mnuweathersave_Click()
    Dim sFile As String
    With dlgCommonDialog
        .Filter = "TrHum files (*.wet)|*.wet"
        .ShowSave
        sFile = .Filename
    End With
    Filename = sFile
    Save_WET98File
End Sub

Private Sub mnuWeatherSteady_Click()
    frmSteadyWeather.Show
End Sub

Private Sub tbToolBar_ButtonClick(ByVal Button As ComctlLib.Button)
    Select Case Button.Key
        Case "New"
            mnuFileNew_Click
        Case "Open"
            mnuFileOpen_Click
        Case "Save"
            mnuFileSave_Click
        Case "Print"
            mnuFilePrint_Click
        Case "Cut"
            mnuEditCut_Click
        Case "Copy"
            mnuEditCopy_Click
        Case "Paste"
            mnuEditPaste_Click
    End Select
End Sub

Private Sub mnuFileOpen_Click()
    Dim sFile As String
    With dlgCommonDialog
        .Filter = "TrHum 98 files (*.hum)|*.hum"
        .ShowOpen
        If Len(.Filename) = 0 Then
            Exit Sub
        End If
        sFile = .Filename
        Filename = sFile
    End With
    Open_File
End Sub

Private Sub mnuFileSave_Click()
    Dim sFile As String
    With dlgCommonDialog
        .Filter = "TrHum 98 files (*.hum)|*.hum"
        .ShowSave
        sFile = .Filename
    End With
    Filename = sFile
    save_file
End Sub

```

```

Private Sub mnuFileSaveAs_Click()
    Dim sFile As String
    With dlgCommonDialog
        .Filter = "TrHum files (*.hum)|*.hum"
        .ShowSave
        sFile = .Filename
    End With
    Filename = sFile
    save_file
End Sub

Private Sub mnuFilePrint_Click()
    Dim BeginPage, EndPage, NumCopies, i
    dlgCommonDialog.CancelError = True
    On Error GoTo ErrHandler
    dlgCommonDialog.ShowPrinter
    BeginPage = dlgCommonDialog.FromPage
    EndPage = dlgCommonDialog.ToPage
    NumCopies = dlgCommonDialog.Copies
    For i = 1 To NumCopies
        printproject
    Next i
    Exit Sub
ErrHandler:
Exit Sub
End Sub

```

B.3.10. Sequência de menus do “form frmmain”

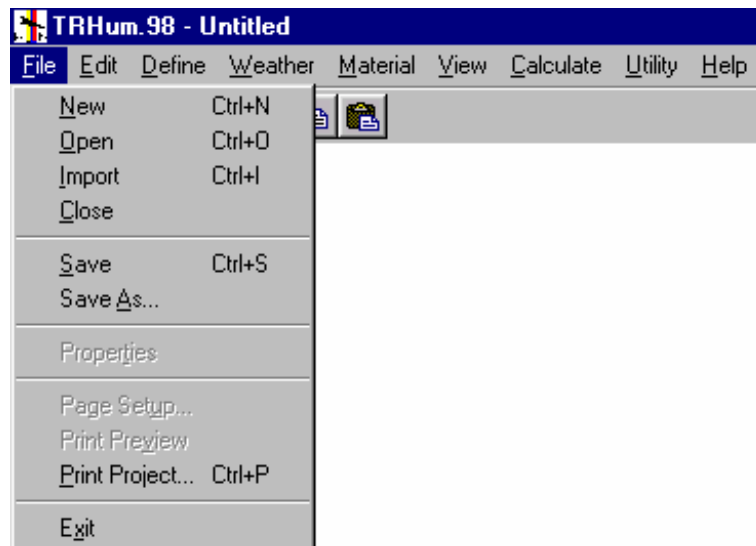


Fig. 92 – menu “File”

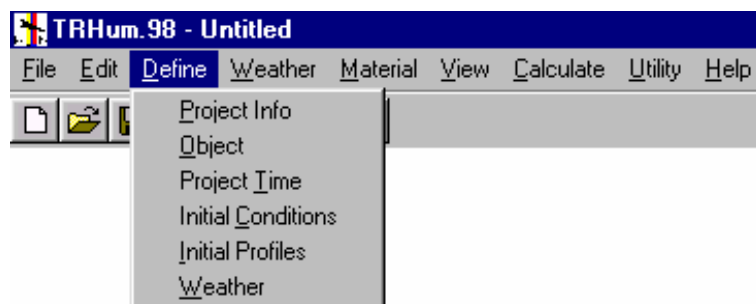


Fig. 93 – menu “Define”

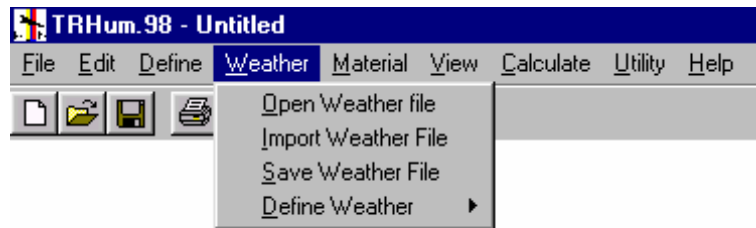


Fig. 94 – menu “Weather”

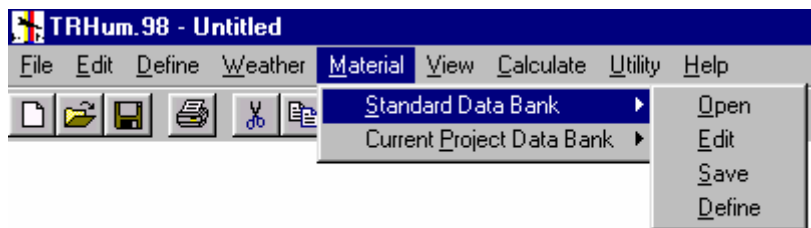


Fig. 95 – menu “Material – Standard data Bank”

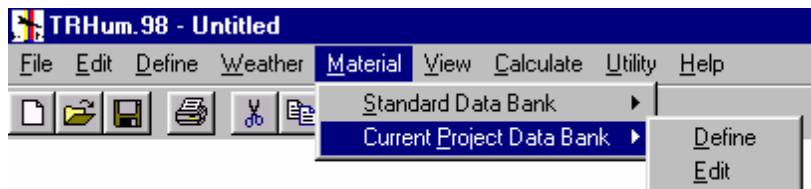


Fig. 96 – menu “Material – Current Project Data Bank”

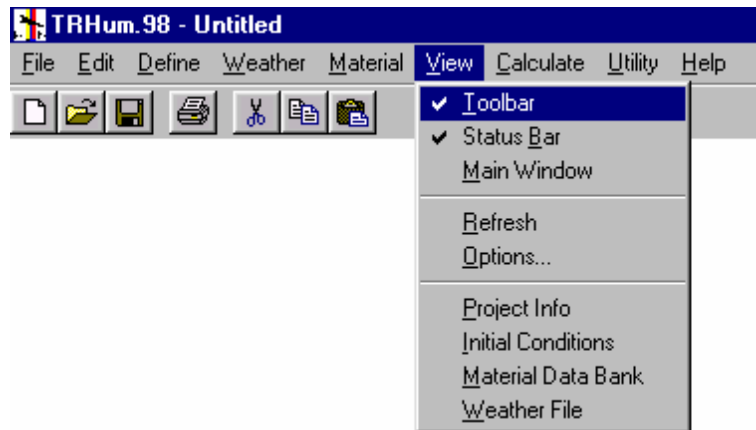


Fig. 97 – menu “View”

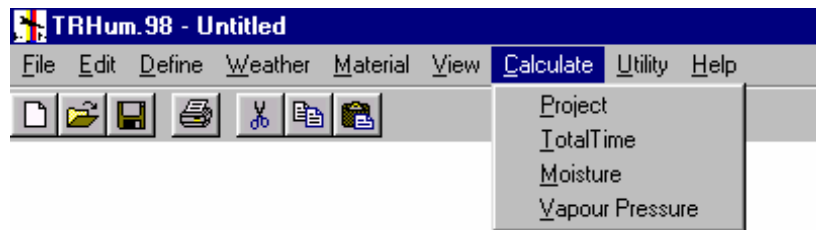


Fig. 98 – menu “Calculate”

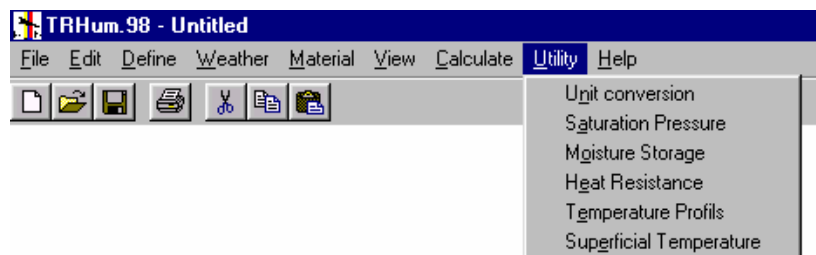


Fig. 99 – menu “Utility”

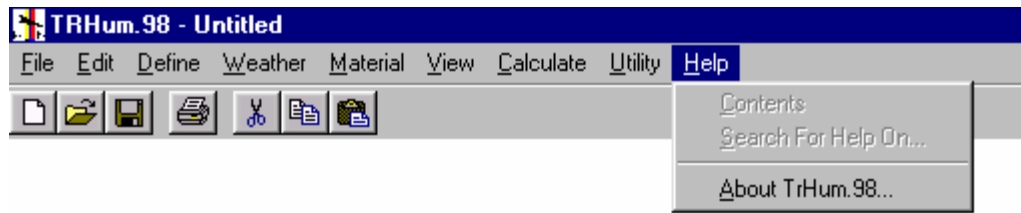


Fig. 100 – menu “Help”

B.3.11. “Form frmmainwindow”

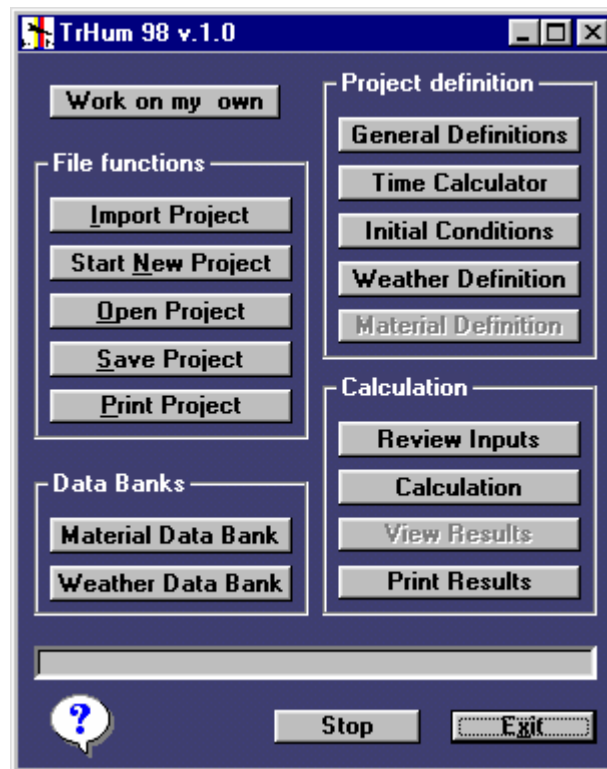


Fig. 101 – “Form frmmainwindow”

Option Explicit

```

Public Sub cleaning(iteration)
Dim ii As Integer
Dim iii As Integer
For ii = 1 To iteration
    For iii = 1 To gTotalNodes
        wa(iii, ii) = 0
        ta(iii, ii) = 0
    Next
Next
End Sub

```

```

Public Sub calculation()
Dim WI As Integer

```



```

sFile = .Filename
sfile2 = .Filename
sfile3 = .Filename
Mid$(sfile2, Len(sfile2) - 2, 3) = "csv"
Mid$(sfile3, Len(sfile3) - 2, 3) = "lng"
Filename = sFile
FilenameTmp = sfile2
FilenameLng = sfile3
End With
ProgressBar1.Enabled = True
ProgressBar1.Value = IT
calculation
Exit Sub
ErrorHandler:
Exit Sub
End Sub

```

```

Private Sub cmdExit_Click()
Unload Me
End
End Sub

```

```

Private Sub cmdimport_Click()
startclean
frmImport.Show
End Sub

```

```

Private Sub cmdOpenProject_Click()
startclean
On Error GoTo errorhandler
Dim sFile As String
With dlgCommonDialog
.Filter = "Files (*.hum)|*.hum"
.ShowOpen
If Len(.Filename) = 0 Then
Exit Sub
End If
sFile = .Filename
Filename = sFile
End With
Open_File
Exit Sub
errorhandler:
Exit Sub
End Sub

```

```

Private Sub cmdPVS_Click()
frmSat2.Show
End Sub

```

```

Private Sub cmdprintproj_Click()
Dim BeginPage, EndPage, NumCopies, I
dlgCommonDialog.CancelError = True
On Error GoTo ErrorHandler
dlgCommonDialog.ShowPrinter

BeginPage = dlgCommonDialog.FromPage
EndPage = dlgCommonDialog.ToPage
NumCopies = dlgCommonDialog.Copies
For I = 1 To NumCopies
printproject
Next I
Exit Sub
ErrorHandler:
'User pressed the Cancel button
Exit Sub
End Sub

```

```

Private Sub CmdPrintresult_Click()
PrintForm
End Sub

```

```

Private Sub cmdReview_Click()
frmReview.Show
End Sub

```

```

Private Sub CmdSave_Click()
On Error GoTo ErrorHandler
Dim sFile As String
With dlgCommonDialog
.Filter = "TrHum files (*.hum)|*.hum"
.ShowSave
sFile = .Filename
End With

```

```

        Filename = sFile
        save_file
        Exit Sub
ErrorHandler:
Exit Sub
End Sub

Private Sub cmdStartNew_Click()
    startclean
    FileNew
    CleanValue
End Sub

Private Sub CmdTime_Click()
    FrmTimeR.Show
End Sub

Private Sub CmdInitialC_Click()
    FrmInitialC.Show
End Sub

Private Sub CmdObjDef_Click()
    FrmObjectName.Show
End Sub

Private Sub cmdMaterialDBK_Click()
    MAterial.Show
End Sub

Private Sub cmdviewresults_Click()
    frmresults.Show
End Sub

Private Sub cmdWeather_Click()
    frmweather.Show
End Sub

Private Sub cmdWeatherDBK_Click()
    frmweather.Show
End Sub

Private Sub Form_Load()
    frmMainWindow.Enabled = True
    frmMain.Enabled = False
    ProgressBar1.Visible = True
End Sub

Private Sub imghelp_Click()
    frmhelp.Show
    helpmessage = " Help file"
End Sub

```

B.3.12. "Form frmobjectname"

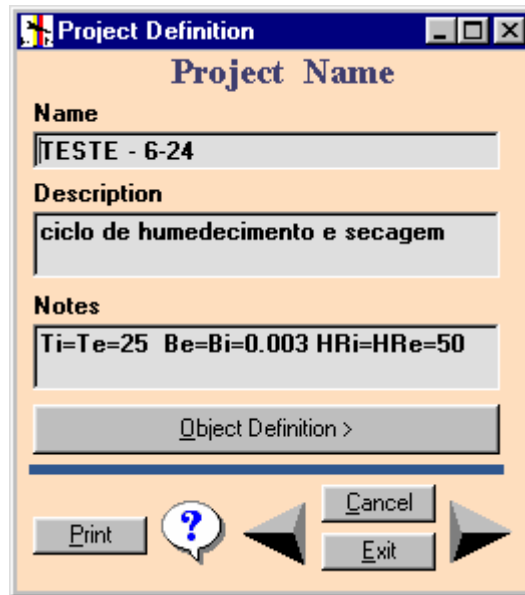


Fig. 102 – "Form frmobjectname"

Option Explicit

```
Private Sub CmdObjDef_Click()  
    passvalue  
    Unload Me  
    frmObjectDefinition.Show  
End Sub
```

```
Private Sub Form_Unload(Cancel As Integer)  
    frmMain.Enabled = True  
    frmMainWindow.Enabled = True  
    frmObjectDefinition.Show  
End Sub
```

```
Private Sub imghelp_Click()  
    frmhelp.Show  
    helpmessage = " Help file"  
End Sub
```

```
Private Sub imgNext_Click()  
    passvalue  
    Unload Me 'unload this form  
    frmObjectDefinition.Show  
End Sub
```

```
Private Sub ImgPrevious_Click()  
    Unload Me 'unload this form  
    frmMainWindow.Show  
    passvalue  
End Sub
```

```
Function passvalue()  
    gObjName = txtObjName.Text  
    gObjDescription = txtObjDescription.Text  
    gObjNotes = txtObjNotes.Text  
End Function
```

```
Function getvalue()  
    txtObjName.Text = gObjName  
    txtObjDescription.Text = gObjDescription  
    txtObjNotes.Text = gObjNotes  
End Function
```

B.3.13. "Form frmobjectdefinition"

Layer N°	Tickness	N.Steps	Step
1	10 cm	20	0,5 cm
2	6 cm	2	3 cm
3	4 cm	4	1 cm

Fig. 103 – "Form frmobjectdefinition"

option Explicit

```
Dim i As Integer
Dim n As Integer
Dim Tick As Double
Dim Tickt As Double
Dim lTick(3) As Double
Dim NumberOfLayer As Integer
Dim Matcod(5)
```

```
Private Sub cmdaccept_Click()
    If Val(txtThickness.Text) > 50 Then txtThickness.Text = 50
    If Val(txtThickness.Text) < 1 Then txtThickness.Text = 1
    If Val(txtNumberOfLayer.Text) > 3 Then txtNumberOfLayer.Text = 3
    If Val(txtNumberOfLayer.Text) <= 0 Then txtNumberOfLayer.Text = 1
    i = Val(txtNumberOfLayer.Text)
    NumberOfLayer = Val(txtNumberOfLayer.Text)
    ReDim LayerNumberx(i)
    ReDim Thicknessx(i)
    ReDim SStepx(i)
    ReDim MaterialNamex(i)
    For n = 1 To i
        LayerNumberx(i) = i
    Next n
    Tickt = Val(txtThickness.Text)
    Tick = Tickt / Val(txtNumberOfLayer.Text)
    For n = 1 To i
        lTick(n) = Tick
    Next n
    txtTick1.Text = convert_point$(lTick(1))
    txttick2.Text = convert_point$(lTick(2))
    txttick3.Text = convert_point$(lTick(3))
    If Val(txtstep1.Text) < 1 Then txtstep1.Text = 1
    If Val(txtstep2.Text) < 1 Then txtstep2.Text = 1
    If Val(txtstep3.Text) < 1 Then txtstep3.Text = 1

    txtdx1.Text = Val(txtTick1.Text) / Val(txtstep1.Text)
    txtdx2.Text = Val(txttick2.Text) / Val(txtstep2.Text)
    txtdx3.Text = Val(txttick3.Text) / Val(txtstep3.Text)
    visibility
    draw
```

```

End Sub

Private Sub cmdacmat_Click()
    creatProjectdatabank
End Sub

Private Sub cmdOk1_Click()
    lTick(1) = Val(txtTick1.Text)
    lTick(2) = Val(txttick2.Text)
    lTick(3) = Val(txttick3.Text)
    Dim total As Double
    total = Tickt
    For i = 1 To NumberOfLayer - 1
        total = total - lTick(i)
        If total < 0 Then lTick(i) = lTick(i) + total: total = 0
    Next
    lTick(NumberOfLayer) = total
    txtTick1.Text = convert_point$(lTick(1))
    txttick2.Text = convert_point$(lTick(2))
    txttick3.Text = convert_point$(lTick(3))
    If Val(txtstep1.Text) < 1 Then txtstep1.Text = 1
    If Val(txtstep2.Text) < 1 Then txtstep2.Text = 1
    If Val(txtstep3.Text) < 1 Then txtstep3.Text = 1
    txtdx1.Text = Val(txtTick1.Text) / Val(txtstep1.Text)
    txtdx2.Text = Val(txttick2.Text) / Val(txtstep2.Text)
    txtdx3.Text = Val(txttick3.Text) / Val(txtstep3.Text)
    draw
End Sub

Private Sub Combomaterial_Click()
    On Error Resume Next
    gMATERIALNAME = Combomaterial.Text
    TxtCode.Text = Matcod(Combomaterial.ListIndex)
    gmaterialcode = Matcod(Combomaterial.ListIndex)
    On Error Resume Next
End Sub

Private Sub CmdLoadDBK_Click()
    Dim sFile As String
    With dlgCommonDialog
        .Filter = "Files (*.dbk)|*.dbk"
        .ShowOpen
        If Len(.Filename) = 0 Then
            Exit Sub
        End If
        sFile = .Filename
        Filename = sFile
    End With
    Open_MATFile
    Combomaterial.Clear
    gMATNAME(0) = "none"
    For i = 0 To gMLISTNUMBER
        Matcod(i) = i
        Combomaterial.AddItem gMATNAME(i)
    Next
End Sub

Private Sub Form_Load()
    frmMain.Enabled = False
    frmMainWindow.Enabled = False
    getvalue
    draw
    visibility
End Sub

Private Sub txtNumberOfLayer_LostFocus()
    If Val(txtNumberOfLayer.Text) > 3 Then txtNumberOfLayer.Text = 3
    If Val(txtNumberOfLayer.Text) <= 0 Then txtNumberOfLayer.Text = 1
End Sub

Private Sub txtstep1_LostFocus()
    If Val(txtstep1.Text) < 1 Then txtstep1.Text = 1
End Sub

Private Sub txtstep2_LostFocus()
    If Val(txtstep2.Text) < 1 Then txtstep2.Text = 1
End Sub

Private Sub txtstep3_LostFocus()
    If Val(txtstep3.Text) < 1 Then txtstep3.Text = 1
End Sub

Private Sub txtTick1_LostFocus()
    If Val(txtTick1.Text) > Val(txtThickness.Text) Then
        txtTick1.Text = Val(txtThickness.Text)
    End If

```

```

End Sub

Private Sub txttick2_LostFocus()
If Val(txttick2.Text) > Val(txtThickness.Text) Then
    txttick2.Text = Val(txtThickness.Text)
End If
End Sub

Private Sub txttick3_LostFocus()
If Val(txttick3.Text) > Val(txtThickness.Text) Then
    txttick3.Text = Val(txtThickness.Text)
End If
End Sub

Private Sub txtThickness_LostFocus()
If Val(txtThickness.Text) > 50 Then txtThickness.Text = 50
If Val(txtThickness.Text) < 1 Then txtThickness.Text = 1
i = Val(txtNumberOfLayer.Text)
End Sub

Sub passvalue()
    gthickness = Val(txtThickness.Text) / 100
    gMATERIALNAME = Combomaterial.Text
    gmaterialcode = Val(TxtCode.Text)
    gNumberOfLayers = Val(txtNumberOfLayer.Text)
    gLayerNumber(1) = Val(txtLayerNumber.Text)
    gThickness(1) = Val(txtTick1.Text)
    gSStep(1) = Val(txtstep1.Text)
    gdx(1) = Val(txdx1.Text) / 100
    gLayerNumber(2) = Val(txtLayerNumber2.Text)
    gThickness(2) = Val(txttick2.Text)
    gSStep(2) = Val(txtstep2.Text)
    gdx(2) = Val(txdx2.Text) / 100
    gLayerNumber(3) = Val(txtLayerNumber3.Text)
    gThickness(3) = Val(txttick3.Text)
    gSStep(3) = Val(txtstep3.Text)
    gdx(3) = Val(txdx3.Text) / 100
    creatProjectdatabank
End Sub

Sub getvalue()
    txtThickness.Text = gthickness * 100
    Combomaterial.Text = gMATERIALNAME
    TxtCode.Text = gmaterialcode

    gMATNAME(0) = "nome"
    For i = 0 To gMLISTNUMBER
        Matcod(i) = i
        Combomaterial.AddItem gMATNAME(i)
    Next
    txtNumberOfLayer.Text = gNumberOfLayers
    txtLayerNumber.Text = gLayerNumber(1)
    txtTick1.Text = gThickness(1)
    txtstep1.Text = gSStep(1)
    txdx1.Text = gdx(1) * 100
    txtLayerNumber2.Text = gLayerNumber(2)
    txttick2.Text = gThickness(2)
    txtstep2.Text = gSStep(2)
    txdx2.Text = gdx(2) * 100
    txtLayerNumber3.Text = gLayerNumber(3)
    txttick3.Text = gThickness(2)
    txtstep3.Text = gSStep(3)
    txdx3.Text = gdx(3) * 100
End Sub

Function draw()
Dim L1 As Double
Dim L2 As Double
Dim L3 As Double
Dim tw As Double
If Tick1 = 0 Then Tick1 = 1
    tw = 1225
    Select Case i
    Case 1
        L1 = lTick(1) / Tick1 * tw
        Txt11.Width = L1
        Txt11.Visible = True
        Txt12.Visible = False
        Txt13.Visible = False
    Case 2
        L1 = lTick(1) / Tick1 * tw
        Txt11.Width = L1
        L2 = lTick(2) / Tick1 * tw
        Txt12.Width = L2
        Txt12.Left = 840 + L1
        Txt11.Visible = True

```

```

        Txt12.Visible = True
        Txt13.Visible = False
    Case 3
        L1 = lTick(1) / Tickt * tw
        Txt11.Width = L1
        L2 = lTick(2) / Tickt * tw
        Txt12.Width = L2
        Txt12.Left = 840 + L1
        L3 = lTick(3) / Tickt * tw
        Txt13.Width = L3
        Txt13.Left = 840 + L1 + L2
        Txt11.Visible = True
        Txt12.Visible = True
        Txt13.Visible = True
    End Select
End Function

```

```

Function visibility()
    i = Val(txtNumberOfLayer.Text)
    Select Case i
    Case 1
        txtLayerNumber.Visible = True
        txtTick1.Visible = True
        txtstep1.Visible = True
        txtdx1.Visible = True
        cmdOk1.Visible = True
        Label10.Visible = False
        txtLayerNumber2.Visible = False
        txttick2.Visible = False
        txtstep2.Visible = False
        txtdx2.Visible = False
        txtLayerNumber3.Visible = False
        txttick3.Visible = False
        txtstep3.Visible = False
        txtdx3.Visible = False
        Label11.Visible = False
        txtLayerNumber.Text = 1
        Label14.Visible = False
        Label15.Visible = False
    Case 2
        txtLayerNumber.Visible = True
        txtTick1.Visible = True
        txtstep1.Visible = True
        cmdOk1.Visible = True
        txtLayerNumber2.Visible = True
        txttick2.Visible = True
        txtstep2.Visible = True
        txtLayerNumber3.Visible = False
        Label10.Visible = True
        txttick3.Visible = False
        txtstep3.Visible = False
        txtLayerNumber.Text = 1
        txtLayerNumber2.Text = 2
        txtdx1.Visible = True
        txtdx2.Visible = True
        txtdx3.Visible = False
        Label11.Visible = False
        Label14.Visible = True
        Label15.Visible = False
    Case 3
        txtLayerNumber.Visible = True
        txtTick1.Visible = True
        txtstep1.Visible = True
        cmdOk1.Visible = True
        txtLayerNumber2.Visible = True
        txttick2.Visible = True
        txtstep2.Visible = True
        txtLayerNumber3.Visible = True
        txttick3.Visible = True
        txtstep3.Visible = True
        txtLayerNumber.Text = 1
        txtLayerNumber2.Text = 2
        txtLayerNumber3.Text = 3
        txtdx1.Visible = True
        txtdx2.Visible = True
        txtdx3.Visible = True
        Label10.Visible = True
        Label11.Visible = True
        Label14.Visible = True
        Label15.Visible = True
    End Select
End Function

```


B.3.14. "Form frmtime-r"

Fig. 104 – "Form frmtime-r"

Option Explicit

```
Dim difdate
Dim difhour
Dim diftime
Dim P(25)
```

```
Private Sub cmdcalculation_Click()
If Trim$(txtDateS) <> "" And Trim$(txtDateF) <> "" And Trim$(txthourss) <> "" And Trim$(txthoursf) <> ""
Then
difdate = DateDiff("h", txtDateS, txtDateF)
difhour = DateDiff("h", txthourss, txthoursf)
diftime = difdate + difhour
lblTotalTime = diftime
lblTotalTime = lblTotalTime + " Hours"
End If
txtHourS = Format(txthourss, "hh")
txtYearS = Format(txtDateS, "yyyy")
txtDays = Format(txtDateS, "y")
End Sub
```

```
Private Sub cmdExit_Click()
If txtDateS > txtDateF Then
message = " Project Finish Date is Before Project Start Date. Exit Anyway?"
Buttons = MB_YESNO + MB_ICONQUESTION
Title = "Project Total Time Calculation"
Answer = MsgBox(message, Buttons, Title)
If Answer = IDYES Then
lblTotalTime = 0
Unload Me
End If
Exit Sub
End If
passvalue
Unload Me
End Sub
```

```
Private Sub cmdoKhourf_Click()
txthoursf = Format(txthoursf, "hh:mm")
If Not IsDate(txthoursf) Then
message = "Not a valid date! Try again!"
messagebox
Exit Sub
End If
End Sub
```

```
Private Sub cmdokdatef_Click()
```

```

txtDateF = Format(txtDateF, "d/m/yyyy")
If Not IsDate(txtDateF) Then
    message = "Not a valid date! Try again!"
    messagebox
Exit Sub
End If
End Sub

```

```

Private Sub cmdokdates_Click()
    txtDateS = Format(txtDateS, "d/m/yyyy ")
    If Not IsDate(txtDateS) Then
        message = "Not a valid date! Try again!"
        messagebox
    Exit Sub
    End If
End Sub

```

```

Private Sub cmdokhours_Click()
    txthourss = Format(txthourss, "hh:mm")
    If Not IsDate(txthourss) Then
        message = "Not a valid date! Try again!"
        messagebox
    Exit Sub
    End If
End Sub

```

```

Private Sub Form_Unload(Cancel As Integer)
    frmMain.Enabled = True
    frmMainWindow.Enabled = True
End Sub

```

```

Function passvalue()
    Dim n As Integer
    gTotalTime = difftime
    gDayS = Format(txtDateS, "y")
    gYears = Format(txtDateS, "yyyy")
    gHourS = Format(txthourss, "hh")
    gDateS = txtDateS
    gDateF = txtDateF
    gHoursS = txthourss
    gHoursF = txthourssf
    nprofile = txt2double(txtprofilnumber.Text)
    P(1) = Val(txtp1.Text)
    P(2) = Val(txtp2.Text)
    P(3) = Val(txtp3.Text)
    P(4) = Val(txtp4.Text)
    P(5) = Val(txtp5.Text)
    P(6) = Val(txtp6.Text)
    P(7) = Val(txtp7.Text)
    P(8) = Val(txtp8.Text)
    P(9) = Val(txtp9.Text)
    P(10) = Val(txtp10.Text)
    P(11) = Val(txtp11.Text)
    P(12) = Val(txtp12.Text)
    P(13) = Val(txtp13.Text)
    P(14) = Val(txtp14.Text)
    P(15) = Val(txtp15.Text)
    P(16) = Val(txtp16.Text)
    P(17) = Val(txtp17.Text)
    P(18) = Val(txtp18.Text)
    P(19) = Val(txtp19.Text)
    P(20) = Val(txtp20.Text)
    P(21) = Val(txtp21.Text)
    P(22) = Val(txtp22.Text)
    P(23) = Val(txtp23.Text)
    P(24) = Val(txtp24.Text)
    For n = 1 To nprofile
        time_profile(n) = P(n)
    Next
End Function

```

```

Function getvalue()
    Dim n As Integer
    txtDateS.Text = gDateS
    txtDateF.Text = gDateF
    txthourss.Text = gHoursS
    txthourssf.Text = gHoursF
    txtprofilnumber.Text = nprofile
    For n = 1 To nprofile
        P(n) = time_profile(n)
    Next
    txtp1.Text = P(1)
    txtp2.Text = P(2)
    txtp3.Text = P(3)
    txtp4.Text = P(4)
    txtp5.Text = P(5)

```

```

txtp6.Text = P(6)
txtp7.Text = P(7)
txtp8.Text = P(8)
txtp9.Text = P(9)
txtp10.Text = P(10)
txtp11.Text = P(11)
txtp12.Text = P(12)
txtp13.Text = P(13)
txtp14.Text = P(14)
txtp15.Text = P(15)
txtp16.Text = P(16)
txtp17.Text = P(17)
txtp18.Text = P(18)
txtp19.Text = P(19)
txtp20.Text = P(20)
txtp21.Text = P(21)
txtp22.Text = P(22)
txtp23.Text = P(23)
txtp24.Text = P(24)
End Function

```

B.3.15. "Form frinitialc"

Fig. 105 – "Form frinitialc"

Option Explicit

```

Function getvalue()
    txtHI.Text = (gHI)
    txtHE.Text = (gHe)
    txtBetaE.Text = gBetaE
    txtBetaI.Text = gBetaI
    txtErrorMax.Text = (gErrorMax)
    txtLatitude.Text = (gLatitude)
    txtLongitude.Text = (gLongitude)
    txtWindDirection.Text = (gWindDirection)
    txtWindSpeed.Text = (gWindSpeed)
    txtAzimut.Text = (gAzimute)
    txtCity.Text = gCity
    txtCountry.Text = gCountry
    txtalpha.Text = galpha
End Function

```

```

Function passvalue()
    gHI = txt2double(txtHI.Text)

```

```

gHe = txt2double(txtHE.Text)
gBetaE = txt2double(txtBetaE.Text)
gBetaI = txt2double(txtBetaI.Text)
gErrorMax = txt2double(txtErrorMax.Text)
gLatitude = txt2double(txtLatitude.Text)
gLongitude = txt2double(txtLongitude.Text)
gWindDirection = txt2double(txtWindDirection.Text)
gWindSpeed = txt2double(txtWindSpeed.Text)
gAzimute = txt2double(txtAzimut.Text)
gCity = txtCity.Text
gCountry = txtCountry.Text
galpha = txt2double(txtalpha.Text)
End Function

```

B.3.16. "Form frinitialprofil"

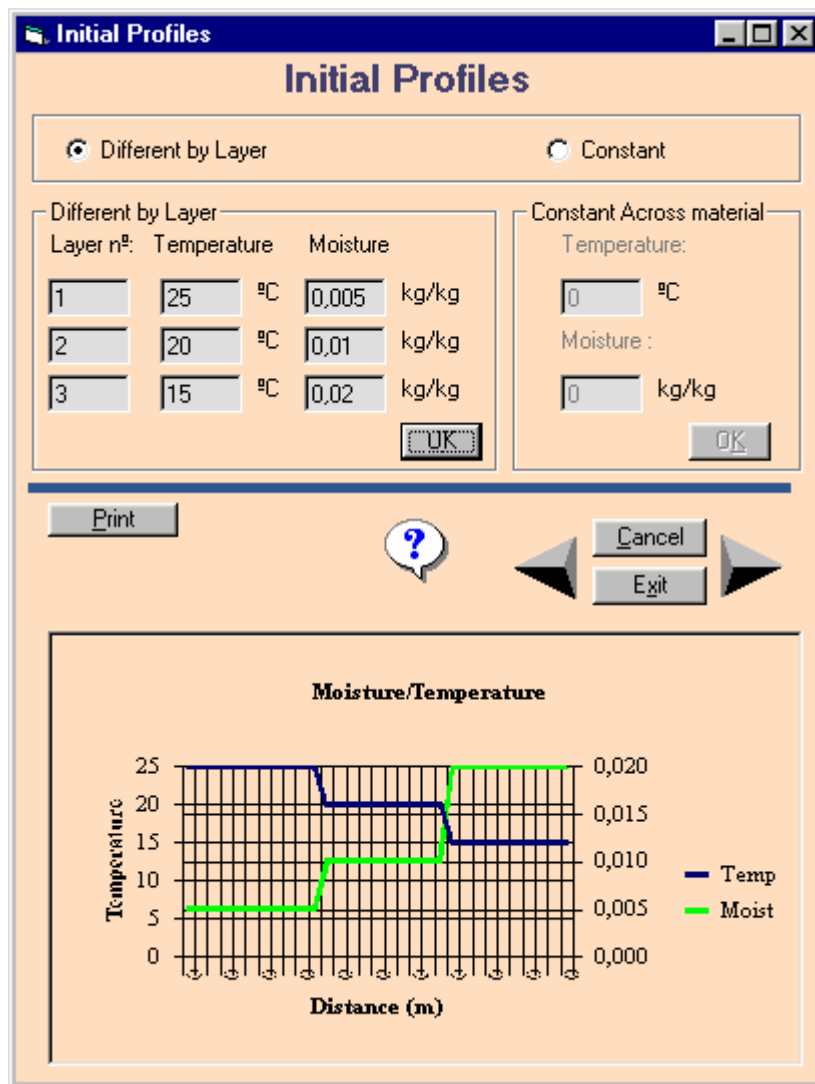


Fig. 106 – "Form frinitialprofil"

```

Option Explicit

Dim TempStart(3)          'initial Temperature
Dim RHStart(3)           'initial relative humidity
Dim axisId



---


Function chartp1()
Dim Column As Integer
Dim Row As Integer

With frminitialprofil.MSChart1
.Visible = True
.chartType = VtChChartType2dLine
.Title = True
.TitleText = "Moisture/Temperature"
.ColumnCount = 2
.RowCount = gTotalNodes
.ShowLegend = True
Column = 1
For Row = 1 To gTotalNodes
.Row = Row
.Column = Column
.ColumnLabel = "Temp"
.Data = gNTempStart(Row)
.RowLabel = distance(Row)
Next
Column = 2
For Row = 1 To gTotalNodes
.Row = Row
.Column = Column
.ColumnLabel = "Moist"
.Data = gNRHStart(Row)

Next
End With
For axisId = VtChAxisIdX To VtChAxisIdY
With MSChart1.Plot.Axis(axisId, 1).AxisTitle
.Visible = True
Select Case axisId
Case 0
.Text = "Distance (m)"
Case 1
.Text = "Temperature"
Case 2
.Text = "Moisture"
End Select
End With
Next
End Function



---


Private Sub cmdconstant_Click()
Dim i
frminitialprofil.Height = 8100
For i = 1 To gNumberOFLayers
TempStart(i) = txt2double(txtITempStart)
RHStart(i) = txt2double(txtRHStart)
Next
passvalue
NNODES 'calculation of the starting node number and total nodes
startprofiles ' Calculation of starting profiles
chartp1
chartp2
End Sub



---


Private Sub cmddifferent_Click()
Dim i
frminitialprofil.Height = 8100
i = gNumberOFLayers

Select Case i
Case 1
TempStart(i) = txt2double(txtTemp1)
RHStart(i) = txt2double(txtRH1)
Case 2
TempStart(i) = txt2double(txtTemp2)
RHStart(i) = txt2double(txtRH2)
TempStart(i - 1) = txt2double(txtTemp1)
RHStart(i - 1) = txt2double(txtRH1)
Case 3
TempStart(i) = txt2double(txtTemp3)
RHStart(i) = txt2double(txtRH3)
TempStart(i - 1) = txt2double(txtTemp2)
RHStart(i - 1) = txt2double(txtRH2)
TempStart(i - 2) = txt2double(txtTemp1)
RHStart(i - 2) = txt2double(txtRH1)
End Select

```

```

passvalue
NNODES 'calculation of the starting node number and total nodes
startprofiles
chartp2
chartp1
End Sub

```

```

Private Sub Form_Load()
    getvalue
    frminitialprofil.Height = 4665
    frmMain.Enabled = False
    frmMainWindow.Enabled = False
    cmdconstant.Enabled = False
    cmddifferent.Enabled = False
    Label6.Enabled = False
    Label7.Enabled = False
    Label8.Enabled = False
    txtRH1.Enabled = False
    txtRH2.Enabled = False
    txtRH3.Enabled = False
    txtLayerNumber1.Enabled = False
    txtLayerNumber2.Enabled = False
    txtLayerNumber3.Enabled = False
    txtTemp1.Enabled = False
    txtTemp2.Enabled = False
    txtTemp3.Enabled = False
    Label3.Enabled = False
    Label4.Enabled = False
    TxtITempStart.Enabled = False
    txtRHStart.Enabled = False
End Sub

```

```

Private Sub OptionProfiles_Click(Index As Integer)
If (Index = 0) Then 'constante
    cmdconstant.Enabled = True
    cmddifferent.Enabled = False
    Label3.Enabled = True
    Label4.Enabled = True
    TxtITempStart.Enabled = True
    txtRHStart.Enabled = True
    Label6.Enabled = False
    Label7.Enabled = False
    Label8.Enabled = False
    txtRH1.Enabled = False
    txtRH2.Enabled = False
    txtRH3.Enabled = False
    txtLayerNumber1.Enabled = False
    txtLayerNumber2.Enabled = False
    txtLayerNumber3.Enabled = False
    txtTemp1.Enabled = False
    txtTemp2.Enabled = False
    txtTemp3.Enabled = False
    Dim i
    Dim auxTempStart
    Dim auxRHStart
    auxTempStart = 0
    auxRHStart = 0
    For i = 1 To gNumberOfLayers
        auxTempStart = auxTempStart + TempStart(i)
        auxRHStart = auxRHStart + RHStart(i)
    Next i
    auxTempStart = auxTempStart / gNumberOfLayers
    auxRHStart = auxRHStart / gNumberOfLayers
    TxtITempStart.Text = auxTempStart
    txtRHStart.Text = auxRHStart
End If

If (Index = 1) Then
    Label3.Enabled = False
    Label4.Enabled = False
    cmdconstant.Enabled = False
    cmddifferent.Enabled = True
    TxtITempStart.Enabled = False
    txtRHStart.Enabled = False
    Label6.Enabled = True
    Label7.Enabled = True
    Label8.Enabled = True
    txtRH1.Enabled = True
    txtRH2.Enabled = True
    txtRH3.Enabled = True
    txtLayerNumber1.Enabled = True
    txtLayerNumber2.Enabled = True
    txtLayerNumber3.Enabled = True
    txtTemp1.Enabled = True
    txtTemp2.Enabled = True
    txtTemp3.Enabled = True
    i = gNumberOfLayers

```

```

Select Case i
Case 1
    txtTemp1.Text = TempStart(i)
    txtRH1.Text = RHStart(i)
    txtLayerNumber2.Enabled = False
    txtTemp2.Enabled = False
    txtRH2.Enabled = False
    Label10.Enabled = False
    Label13.Enabled = False
    txtLayerNumber3.Enabled = False
    txtTemp3.Enabled = False
    txtRH3.Enabled = False
    Label11.Enabled = False
    Label14.Enabled = False
Case 2
    txtTemp2.Text = TempStart(i)
    txtRH2.Text = RHStart(i)
    txtTemp1.Text = TempStart(i - 1)
    txtRH1.Text = RHStart(i - 1)
    txtLayerNumber3.Enabled = False
    txtTemp3.Enabled = False
    txtRH3.Enabled = False
    Label11.Enabled = False
    Label14.Enabled = False
Case 3
    txtTemp3.Text = TempStart(i)
    txtRH3.Text = RHStart(i)
    txtTemp2.Text = TempStart(i - 1)
    txtRH2.Text = RHStart(i - 1)
    txtTemp1.Text = TempStart(i - 2)
    txtRH1.Text = RHStart(i - 2)
End Select
End If
End Sub

```

```

Function passvalue()
Dim i
For i = 1 To gNumberOFLayers
    gTempStart(i) = TempStart(i)
    gRHStart(i) = RHStart(i)
Next i
End Function

```

```

Function getvalue()
Dim i
For i = 1 To gNumberOFLayers
    TempStart(i) = gTempStart(i)
    RHStart(i) = gRHStart(i)
Next i
End Function

```

B.3.17. "Form frmmainweather"

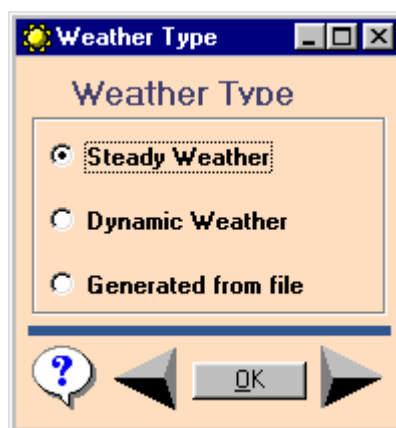


Fig. 107 – "Form frmmainweather"

```

Option Explicit
Public showForm

```

```

Private Sub cmdOK_Click()
Select Case showForm
Case 1

```

```

        frmSteadyWeather.Show
    Case 2
        frmDynamicWeather.Show
    Case 3
    Dim sFile As String
        With dlgCommonDialog
            .Filter = "Files (*.wet)|*.wet"
            .ShowOpen
            If Len(.Filename) = 0 Then
                Exit Sub
            End If
            sFile = .Filename
            Filename = sFile
        End With
        Open_WETFile
        frmGeneratedWeather.Show
End Select
    Unload Me
End Sub

```

```

Private Sub imgNext_Click()
    Select Case showForm
    Case 1
        frmSteadyWeather.Show
    Case 2
        frmDynamicWeather.Show
    Case 3
    Dim sFile As String
        With dlgCommonDialog
            .Filter = "Files (*.wet)|*.wet"
            .ShowOpen
            If Len(.Filename) = 0 Then
                Exit Sub
            End If
            sFile = .Filename
            Filename = sFile
        End With
        Open_WETFile
        frmGeneratedWeather.Show
    End Select
    Unload Me
End Sub

```

```

Public Sub optweather_Click(Index As Integer)
    If (Index = 0) Then
        showForm = 1
    End If
    If (Index = 1) Then
        showForm = 2
    End If
    If (Index = 2) Then
        showForm = 3
    End If
End Sub

```


B.3.18. "Form frmsteadyweather"

Fig. 108 – "Form frmsteadyweather"

Option Explicit

```
Dim Rain      'Weather rain yes or no
Dim TempIn   'indoor temperature
Dim TempOut  'outdoor temperature
Dim RHIn     'indoor relative humidity
Dim RHOut    'outdoor relative humidity
Dim Rad      'radiation
Dim Time     'time step in hours
Dim step     'step in hours
```

```
Sub getvalue()
    txtTempOut.Text = gTempOut(1)
    txtRHOut.Text = gRHOut(1)
    txtRad.Text = grad(1)
    txtRain.Text = gRain(1)
    txtTempIn.Text = gTempIn(1)
    txtRHIn.Text = gRHIn(1)
End Sub
```

```
Sub passvalue()
    gTempOut(1) = txtTempOut.Text
    gRHOut(1) = txtRHOut.Text
    grad(1) = txtRad.Text
    gRain(1) = txtRain.Text
    gTempIn(1) = txtTempIn.Text
    gRHIn(1) = txtRHIn.Text
    gstep(1) = step
End Sub
```

B.3.19. "Form frmdynamicweather"

Project Time

Number of Intervals in Weather File: Total Project Time:

Time

Number	Time [h]	Step [h]
<input type="text" value="1"/>	<input type="text"/>	<input type="text"/>

Outdoor

Temp.[°C] | R.H [%] | Rad [W/m2] | Rain [0/1]

Indoor

Temp.[°C] | R.H [%]

Number	Hours	step	Outd. Ten	Outd. RH	Outd. Rad	Ind. Temp	Ind. RH	Rai
1	72	0,25	25	50	0	25	50	1
2	444	0,01	25	50	0	25	50	0
3	144	0,01	25	10	0	25	50	0
4	144	0,01	25	10	0	25	50	0

Weather Time

Total Weather Time: h

N. Repetitions:

Create remaining weather

Fig. 109 – "Form frmdynamicweather"

Option Explicit

```

Dim Totaltime
Dim TotalWeather 'totaltime of the weather file
Dim totalstep 'number of steps in weather file
Dim n As Integer 'number of steps in weather file
Dim number(100)
Dim Rain(100) 'Weather rain yes or no
Dim TempIn(100) 'indoor temperature
Dim TempOut(100) 'outdoor temperature
Dim RHIn(100) 'indoor relative humidity
Dim RHOut(100) 'outdoor relative humidity
Dim Rad(100) 'radiation
Dim Time(100) 'time in hours
Dim step(100) 'time step in hours
Dim datain(100)
Dim Dataout(100)

```

```

Private Sub Check1_Click()
    genweather = True
End Sub

```

```

Private Sub cmdaccept_Click()
Dim aux2 As Integer
aux2 = TxtTotalSteps.Text
cmdAdd.Enabled = True
grdDynamic.Rows = aux2 + 1
grdDynamic.Cols = 9
totalstep = aux2
End Sub

```

```

Private Sub cmdchangetime_Click()
Totaltime = TxtTotalTime
End Sub

```

```

Private Sub cmdoRad_Click()
Dim aux1 As Integer
For aux1 = 1 To totalstep
Dataout(aux1) = Rad(aux1)
Next
chart2
End Sub

```

```

Private Sub cmdORH_Click()
Dim aux1 As Integer
For aux1 = 1 To totalstep
datain(aux1) = RHIn(aux1)
Dataout(aux1) = RHOut(aux1)
Next
chart
End Sub

```

```

Private Sub cmdRain_Click()
Dim aux1 As Integer
For aux1 = 1 To totalstep
Dataout(aux1) = Rain(aux1)
Next
chart2
End Sub

```

```

Private Sub Form_Load()
Dim aux2 As Integer
getvalue
frmMain.Enabled = False
frmMainWindow.Enabled = False
'headers
Dim aux1 As Integer
grdDynamic.Row = 0
grdDynamic.Col = 0
grdDynamic.Text = "Number"
grdDynamic.Col = 1
grdDynamic.Text = "Hours"
grdDynamic.Col = 2
grdDynamic.Text = "step"
grdDynamic.Col = 3
grdDynamic.Text = "Outd.Temp."
grdDynamic.Col = 4
grdDynamic.Text = "Outd.RH."
grdDynamic.Col = 5
grdDynamic.Text = "Outd.Rad."
grdDynamic.Col = 6
grdDynamic.Text = "Ind.Temp."
grdDynamic.Col = 7
grdDynamic.Text = "Ind.RH."
grdDynamic.Col = 8
grdDynamic.Text = "Rain"
For aux1 = 0 To 8
grdDynamic.ColWidth(aux1) = 800
Next
aux2 = totalstep
cmdAdd.Enabled = True
grdDynamic.Rows = totalstep + 1
grdDynamic.Cols = 9
totalstep = aux2
fillgrid
txtN = 1
End Sub

```

```

Private Sub cmdAdd_Click()
'To add a line to the weather file
n = txtN.Text
number(n) = txtN.Text
Time(n) = txtTime.Text
step(n) = txtstep.Text
TempOut(n) = txtTempOut.Text
RHOut(n) = txtRHoutd.Text
Rad(n) = txtRad.Text

```

```

TempIn(n) = txtTempIn.Text
RHIn(n) = txtRHIn.Text
Rain(n) = txtRain
On Error Resume Next
    grdDynamic.Row = n
    grdDynamic.Col = 0
    grdDynamic.Text = number(n)
    grdDynamic.Col = 1
    grdDynamic.Text = Time(n)
    grdDynamic.Col = 2
    grdDynamic.Text = step(n)
    grdDynamic.Col = 3
    grdDynamic.Text = TempOut(n)
    grdDynamic.Col = 4
    grdDynamic.Text = RHOut(n)
    grdDynamic.Col = 5
    grdDynamic.Text = Rad(n)
    grdDynamic.Col = 6
    grdDynamic.Text = TempIn(n)
    grdDynamic.Col = 7
    grdDynamic.Text = RHIn(n)
    grdDynamic.Col = 8
    grdDynamic.Text = Rain(n)
'To verify if Step number is not exceeded
Dim i As Integer
If n = Val(txtTotalSteps.Text) Then
    TotalWeather = 0
    For i = 1 To n
        TotalWeather = Val(Time(i)) + TotalWeather
    Next i
    txtTTWeather.Text = TotalWeather
    message = " Number of step is achieved! Stop input or change Total step!"
    Buttons = MB_OK + MB_ICONEXCLAMATION
    Title = "Weather definition"
    Answer = MsgBox(message, Buttons, Title)
    cmdAdd.Enabled = False
    timetotal
End If
n = n + 1
End Sub

```

```

Private Sub cmdtemp_Click()
    Dim aux1 As Integer
    For aux1 = 1 To totalstep
        datain(aux1) = TempIn(aux1)
        Dataout(aux1) = TempOut(aux1)
    Next
chart
End Sub

```

```

Function getvalue()
txtTotalTime.Text = gTotalTime
txtTTWeather.Text = gTotalWeather
txtTotalSteps.Text = gTotalSteps
Totaltime = gTotalTime
TotalWeather = gTotalWeather
totalstep = gTotalSteps
Dim aux1 As Integer
For aux1 = 1 To gTotalSteps
    number(aux1) = gnumber(aux1)
    Rain(aux1) = gRain(aux1)
    TempIn(aux1) = gTempIn(aux1)
    TempOut(aux1) = gTempOut(aux1)
    RHIn(aux1) = gRHIn(aux1)
    RHOut(aux1) = gRHOut(aux1)
    Rad(aux1) = grad(aux1)
    Time(aux1) = gTime(aux1)
    step(aux1) = gstep(aux1)
Next
End Function

```

```

Function passvalue()
gTotalWeather = TotalWeather 'Total time of the Weather File
gTotalSteps = totalstep 'Number of Time Intervals
gTotalTime = Totaltime
Dim aux1
For aux1 = 1 To gTotalSteps
    gnumber(aux1) = number(aux1)
    gRain(aux1) = Rain(aux1)
    gTempIn(aux1) = TempIn(aux1)
    gTempOut(aux1) = TempOut(aux1)
    gRHIn(aux1) = RHIn(aux1)
    gRHOut(aux1) = RHOut(aux1)
    grad(aux1) = Rad(aux1)
    gTime(aux1) = Time(aux1)
    gstep(aux1) = step(aux1)
Next

```

```

End Function

Function timetotal()
If Val(TxtTotalTime.Text) > Val(txtTTWeather.Text) Then
    message = " Weather file total time is smaller than Project Total Time! Do you Want to Generate the
remaining Time?"
    Buttons = MB_YESNO + MB_ICONQUESTION
    Title = "Weather definition"
    Answer = MsgBox(message, Buttons, Title)
    If Answer = IDNO Then Exit Function
End If
End Function

Function generate()
    Dim Num As Integer
    Dim Y As Integer
    Dim aux1 As Integer
    Dim aux2 As Integer
    Dim X As Integer
    Num = Totaltime / TotalWeather
    Num = Int(Num + 0.5)
    Y = totalstep
    totalstep = totalstep * Num
    ReDim Ltime(totalstep)
    ReDim lstep(totalstep)
    ReDim lTempOut(totalstep)
    ReDim lRHOut(totalstep)
    ReDim lRad(totalstep)
    ReDim lTempIn(totalstep)
    ReDim lRHIn(totalstep)
    ReDim lRain(totalstep)
    For aux1 = 1 To Num
        For aux2 = 1 To Y
            X = aux2 + aux1 * TotalWeather
            Ltime(X) = Time(aux2)
            lstep(X) = step(aux2)
            lTempOut(X) = TempOut(aux2)
            lRHOut(X) = RHOut(aux2)
            lRad(X) = Rad(aux2)
            lTempIn(X) = TempIn(aux2)
            lRHIn(X) = RHIn(aux2)
            lRain(X) = Rain(aux2)
            If X >= Totaltime Then Stop
        Next
    Next
End Function

Function chart()
    Dim Column As Integer
    Dim Row As Integer
    Dim Index1
    Dim Index2
    Dim Index3
    Dim Index4
    With frmDynamicWeather.MSChart1
        .chartType = VtChChartType2dLine
        .ColumnCount = 2
        .RowCount = totalstep
        Column = 1
        For Row = 1 To totalstep
            .Column = Column
            .Row = Row
            .Data = Dataout(Row)
            .RowLabel = ""
            .ColumnLabel = "Outdoor"
        Next
        Column = 2
        For Row = 1 To totalstep
            .Column = Column
            .Row = Row
            .Data = datain(Row)
            .RowLabel = ""
            .ColumnLabel = "Indoor"
        Next
        .ShowLegend = True
    End With
End Function

Function chart2()
    Dim Column As Integer
    Dim Row As Integer
    With frmDynamicWeather.MSChart1
        .chartType = VtChChartType2dLine
        .ColumnCount = 1
        .RowCount = totalstep
        Column = 1
        For Row = 1 To totalstep

```

```

        .Column = Column
        .Row = Row
        .Data = Dataout (Row)
        .RowLabel = ""
        .ColumnLabel = "Radiation"
    Next
    .ShowLegend = True
End With
End Function

```

```

Private Sub fillgrid()
For n = 1 To gTotalSteps
    grdDynamic.Row = n
    grdDynamic.Col = 0
    grdDynamic.Text = number(n)
    grdDynamic.Col = 1
    grdDynamic.Text = Time(n)
    grdDynamic.Col = 2
    grdDynamic.Text = step(n)
    grdDynamic.Col = 3
    grdDynamic.Text = TempOut(n)
    grdDynamic.Col = 4
    grdDynamic.Text = RHOut(n)
    grdDynamic.Col = 5
    grdDynamic.Text = Rad(n)
    grdDynamic.Col = 6
    grdDynamic.Text = TempIn(n)
    grdDynamic.Col = 7
    grdDynamic.Text = RHIn(n)
    grdDynamic.Col = 8
    grdDynamic.Text = Rain(n)
Next n
End Sub

```

B.3.20. "Form frmimport"

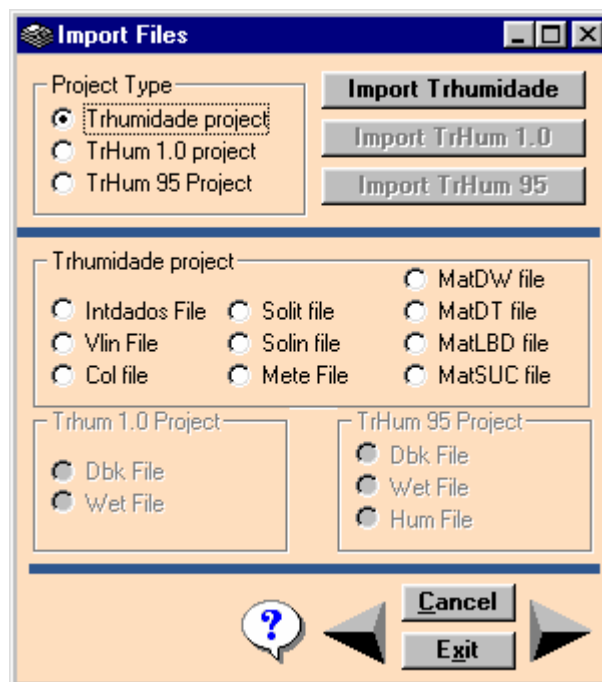


Fig. 110 – "Form frmimport"

Option Explicit

```
Dim AuxType As Integer '0,1,2,3
```

```

Private Sub cmdImportTrhum95_Click()
Select Case AuxType
Case 0
    dlgCommonDialog.Filter = "Files (*.dbk)|*.dbk|"
Case 1

```

```

    dlgCommonDialog.Filter = "Files (*.wet)|*.wet|"
Case 2
    dlgCommonDialog.Filter = "Files (*.hum)|*.hum|"
End Select

Dim sFile As String
With dlgCommonDialog
    .ShowOpen
    If Len(.Filename) = 0 Then
        Exit Sub
    End If
    sFile = .Filename
End With
Select Case AuxType
Case 0
    'open a databank file
Case 1
    Open_WETFile
Case 2
    import_File
End Select
End Sub

```

```

Private Sub Form_Load()
frmMainWindow.Enabled = False
frmMain.Enabled = False
FraTrhumidade.Enabled = False
FraTrhum1.Enabled = False
FraTrhum95.Enabled = False
OptIntdados.Enabled = False
optvlin.Enabled = False
optcol.Enabled = False
optsolit.Enabled = False
optsolin.Enabled = False
optmete.Enabled = False
optdw.Enabled = False
optdt.Enabled = False
optlbd.Enabled = False
optsuc.Enabled = False
optTR1wet.Enabled = False
optTr1dbk.Enabled = False
optTR95dbk.Enabled = False
optTR95wet.Enabled = False
optTR95hum.Enabled = False
cmdImporttrhumidade.Enabled = False
cmdImportTrhum1.Enabled = False
cmdImportTrhum95.Enabled = False
End Sub

```

```

Private Sub optTrhum95_Click()
FraTrhumidade.Enabled = False
FraTrhum1.Enabled = False
FraTrhum95.Enabled = True
OptIntdados.Enabled = False
optvlin.Enabled = False
optcol.Enabled = False
optsolit.Enabled = False
optsolin.Enabled = False
optmete.Enabled = False
optdw.Enabled = False
optdt.Enabled = False
optlbd.Enabled = False
optsuc.Enabled = False
optTR1wet.Enabled = False
optTr1dbk.Enabled = False
optTR95dbk.Enabled = True
optTR95wet.Enabled = True
optTR95hum.Enabled = True
cmdImporttrhumidade.Enabled = False
cmdImportTrhum1.Enabled = False
cmdImportTrhum95.Enabled = True
End Sub

```

```

Private Sub opttrhumidade_Click()
FraTrhumidade.Enabled = True
FraTrhum1.Enabled = False
FraTrhum95.Enabled = False
OptIntdados.Enabled = True
optvlin.Enabled = True
optcol.Enabled = True
optsolit.Enabled = True
optsolin.Enabled = True
optmete.Enabled = True
optdw.Enabled = True
optdt.Enabled = True
optlbd.Enabled = True
optsuc.Enabled = True

```

```

optTRlwet.Enabled = False
optTrlrbk.Enabled = False
optTR95dbk.Enabled = False
optTR95wet.Enabled = False
optTR95hum.Enabled = False
cmdImporttrhumidade.Enabled = True
cmdImportTrhum1.Enabled = False
cmdImportTrhum95.Enabled = False
End Sub

```

```

Private Sub optTrhum1_Click()
FraTrhumidade.Enabled = False
FraTrhum1.Enabled = True
FraTrhum95.Enabled = False
OptIntdados.Enabled = False
optvlin.Enabled = False
optcol.Enabled = False
optsolit.Enabled = False
optsolin.Enabled = False
optmete.Enabled = False
optdw.Enabled = False
optdt.Enabled = False
optlbd.Enabled = False
optsuc.Enabled = False
optTRlwet.Enabled = True
optTrlrbk.Enabled = True
optTR95dbk.Enabled = False
optTR95wet.Enabled = False
optTR95hum.Enabled = False
cmdImporttrhumidade.Enabled = False
cmdImportTrhum1.Enabled = True
cmdImportTrhum95.Enabled = False
End Sub

```

B.3.21. "Form material"

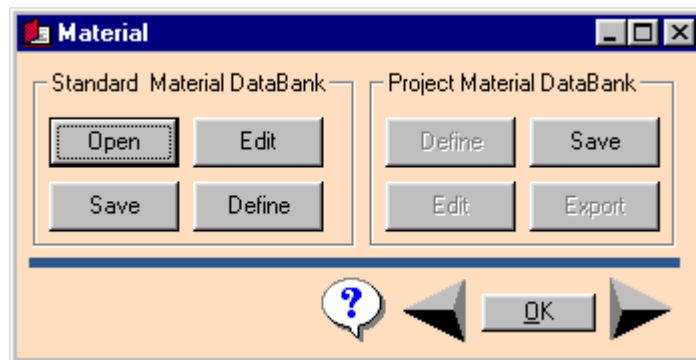


Fig. 111 – "Form material"

Option Explicit

```

Private Sub Command1_Click()
Dim sFile As String
With dlgCommonDialog
.Filter = "Files (*.dbk)|*.dbk"
.ShowOpen
If Len(.Filename) = 0 Then
Exit Sub
End If
sFile = .Filename
Filename = sFile
End With
Open_MATFile
End Sub

```

```

Private Sub Command3_Click()
Dim sFile As String
With dlgCommonDialog
.Filter = "Data Bank files (*.dbk)|*.dbk"
.ShowSave
sFile = .Filename
End With
Filename = sFile
Save_MATFile
End Sub

```


B.3.22. "Form frmreview"

Review Inputs

Print

Object Name: TESTE - 6-24
Object Description: ciclo de humedecimento e secagem
Object Definition: Ti=Te=25 Be=Bi=0.003 HRI=HRe=50
Profil List Number: Label28

Starting date: 0:00:00
Ending date: 0:00:00
Project Total Time: 500
Maximum Error: 0,001

Material name: BetaoCelular
Material Code: 1
Total Thickness: 0,2
N. of layers: 3

Layer Definition:

Number	Thickness	Step	Dx
1	6	10	0
2	7	10	0
3	7	10	0

Initial Conditions:

Outdoor Heat Resistance: 25
Indoor Heat Resistance: 25
Outdoor Moisture Resistance: 0,003
Indoor Moisture Resistance: 0,003
Wind Direction: 80
Wind Speed: 3

City: Porto
Country: Portugal
Latitude: 42
Longitude: 8
Azimuth: 180

Material databank:

Material	Betao1									
30	65300	48100	24800	63400	11800	39900	13200	37400	22400	18200
50	61900	45700	23500	61000	11200	37800	12500	35500	21200	17300
Conduct list										
10	0,135	0,138	0,142	0,155	0,194	0,296	0,314	0,328	0,334	0,339
30	0,135	0,138	0,142	0,155	0,194	0,296	0,314	0,328	0,334	0,339
50	0,135	0,138	0,142	0,155	0,194	0,296	0,314	0,328	0,334	0,339

Weather file:

Number	Hours	step	Outd. Temp	Outd. RH	Outd. Rac	Ind. Temp	Ind. RH	Rain
1	72	0,01	25	50	0	25	50	1
2	444	0,01	25	50	0	25	50	0
3	144	0,01	25	10	0	25	50	0
4	144	0,01	25	10	0	25	50	0
5	144	0,01	25	10	0	25	50	0
6	144	0,01	25	10	0	25	50	0

Material profil:

98
Project Name: TESTE - 6-24
Project Description: ciclo de humedecimento e secagem
Project Notes: Ti=Te=25 Be=Bi=0.003 HRI=HRe=50
Material Name: BetaoCelular
Material Code: 1
Total Thickness: 0,2
Number of Layers: 1
Layer Number 1: 1
Thickness layer 1: 0,2

? Exit

Fig. 112 – "Form frmreview"

Option Explicit

```
Private Sub Form_Load()
    frmMain.Enabled = False
    frmMainWindow.Enabled = False
    RichTextBox1.FileName = Filename
    getvalue
End Sub
```

```
Function getvalue()
    lblObjName.Caption = gObjName
    lblObjDescription.Caption = gObjDescription
    lblObjDef.Caption = gObjNotes
    lblTotalTime.Caption = gTotalTime
    lbhours.Caption = gHoursS
    lbdates.Caption = gDateS
    lbhourf.Caption = gHoursF
    lbdatef.Caption = gDateF
    lbttick.Caption = gThickness
```

```

lbmatcode.Caption = gmaterialcode
lbmatname.Caption = gMATERIALNAME
lbnlayers.Caption = gNumberOFLayers
lblErrorMax.Caption = gErrorMax
lblBetaI.Caption = gBetaI
lblBetaE.Caption = gBetaE
lblHI.Caption = gHI
lblHE.Caption = gHe
lblWindSpeed.Caption = gWindSpeed
lblWindDirection.Caption = gWindDirection
lblLatitude.Caption = gLatitude
lblLongitude.Caption = gLongitude
lblAzimut.Caption = gAzimute
lblCity.Caption = gCity
lblCountry.Caption = gCountry
    weatherDbgrid
    materialdbgrid
    layergrid
End Function

```

```

Sub WeatherDbgrid()
Dim aux1 As Integer
Dim n As Integer
Gridwet.Rows = gTotalSteps + 1
Gridwet.Cols = 9
With Gridwet
    .Row = 0
    .Col = 0
    .Text = "Number"
    .Col = 1
    .Text = "Hours"
    .Col = 2
    .Text = "step"
    .Col = 3
    .Text = "Outd.Temp."
    .Col = 4
    .Text = "Outd.RH."
    .Col = 5
    .Text = "Outd.Rad."
    .Col = 6
    .Text = "Ind.Temp."
    .Col = 7
    .Text = "Ind.RH."
    .Col = 8
    .Text = "Rain"
    For aux1 = 0 To 8
        .ColWidth(aux1) = 800
    Next
    For n = 1 To gTotalSteps
        .Row = n
        .Col = 0
        .Text = gnumber(n)
        .Col = 1
        .Text = gTime(n)
        .Col = 2
        .Text = gstep(n)
        .Col = 3
        .Text = gTempOut(n)
        .Col = 4
        .Text = gRHOut(n)
        .Col = 5
        .Text = grad(n)
        .Col = 6
        .Text = gTempIn(n)
        .Col = 7
        .Text = gRHIn(n)
        .Col = 8
        .Text = gRain(n)
    Next n
End With
End Sub

```

```

Sub layergrid()
Dim aux1 As Integer
Dim n As Integer
Gridlayer.Rows = gNumberOFLayers + 1
Gridlayer.Cols = 4
With Gridlayer
    .Row = 0
    .Col = 0
    .Text = "Number of layer"
    .Col = 1
    .Text = "Thickness"
    .Col = 2
    .Text = "Step"
    .Col = 3
    .Text = "Dx"

```

```

        For n = 1 To gNumberOFLayers
            .Row = n
            .Col = 0
            .Text = gLayerNumber(n)
            .Col = 1
            .Text = gThickness(n)
            .Col = 2
            .Text = gSStep(n)
            .Col = 3
            .Text = gdx(n)
        Next
    End With
End Sub

```

```

Sub materialdbgrid()
Dim aux1 As Integer
Dim aux2 As Integer
Dim count As Integer
Gridmat.Rows = (gSTLIST + 2) * 6
Gridmat.Cols = gSMLIST + 2
With Gridmat
    .Row = 0
    .Col = 0
    .Text = "Material Name"
    .Col = 1
    .Text = gMATERIALNAME
    .Row = 1
    .Col = 0
    .Text = "Material Code"
    .Col = 1
    .Text = gmaterialcode
    .Row = 2
    .Col = 0
    .Text = "Density"
    .Col = 1
    .Text = gSDENSITY
    .Row = 3
    .Col = 0
    .Text = "Heat"
    .Col = 1
    .Text = gSHEAT
    .Row = 4
    .Col = 0
    .Text = "Hygroskop"
    .Col = 1
    .Text = gSHYGROSCOP
    .Row = 5
    .Col = 0
    .Text = "Saturation"
    .Col = 1
    .Text = gSSATURATION
    .Row = 6
    .Col = 0
    .Text = "DT list"
    .Row = 7
    .Col = 0
    .Text = "Mlist/Tlist"
    For aux2 = 1 To gSMLIST
        .Col = aux2
        .Text = gSMOISTURE(aux2)
    Next
    count = 7
    For aux1 = 1 To gSTLIST
        count = count + 1
        .Row = count
        .Col = 0
        .Text = gSTEMPERATURE(aux1)
        For aux2 = 1 To gSMLIST
            .Col = aux2
            .Text = gSDT(aux1, aux2)
        Next
        Next
        count = count + 1
        .Row = count
        .Col = 0
        .Text = "DW list"
        For aux1 = 1 To gSTLIST
            count = count + 1
            .Row = count
            .Col = 0
            .Text = gSTEMPERATURE(aux1)
            For aux2 = 1 To gSMLIST
                .Col = aux2
                .Text = gSDW(aux1, aux2)
            Next
        Next
        count = count + 1

```

```

.Row = count
.Col = 0
.Text = "Capillar list"
For aux1 = 1 To gSTLIST
  count = count + 1
  .Row = count
  .Col = 0
  .Text = gSTEMPERATURE(aux1)
  For aux2 = 1 To gSMLIST
    .Col = aux2
    .Text = gSCAPILLAR(aux1, aux2)
  Next
Next
count = count + 1
.Row = count
.Col = 0
.Text = "Conduct list"
For aux1 = 1 To gSTLIST
  count = count + 1
  .Row = count
  .Col = 0
  .Text = gSTEMPERATURE(aux1)
  For aux2 = 1 To gSMLIST
    .Col = aux2
    .Text = gSCONDUCT(aux1, aux2)
  Next
Next
count = count + 1
.Row = count
.Col = 0
.Text = "Sorp list"
.Col = 1
.Text = gSSorpCount
For aux1 = 1 To gSSorpCount
  count = count + 1
  .Row = count
  .Col = 1
  .Text = gSSorpHR(aux1)
  .Col = 2
  .Text = gSSorpWHR(aux1)
Next
End With
End Sub

```

B.3.23. "Form frmresults"

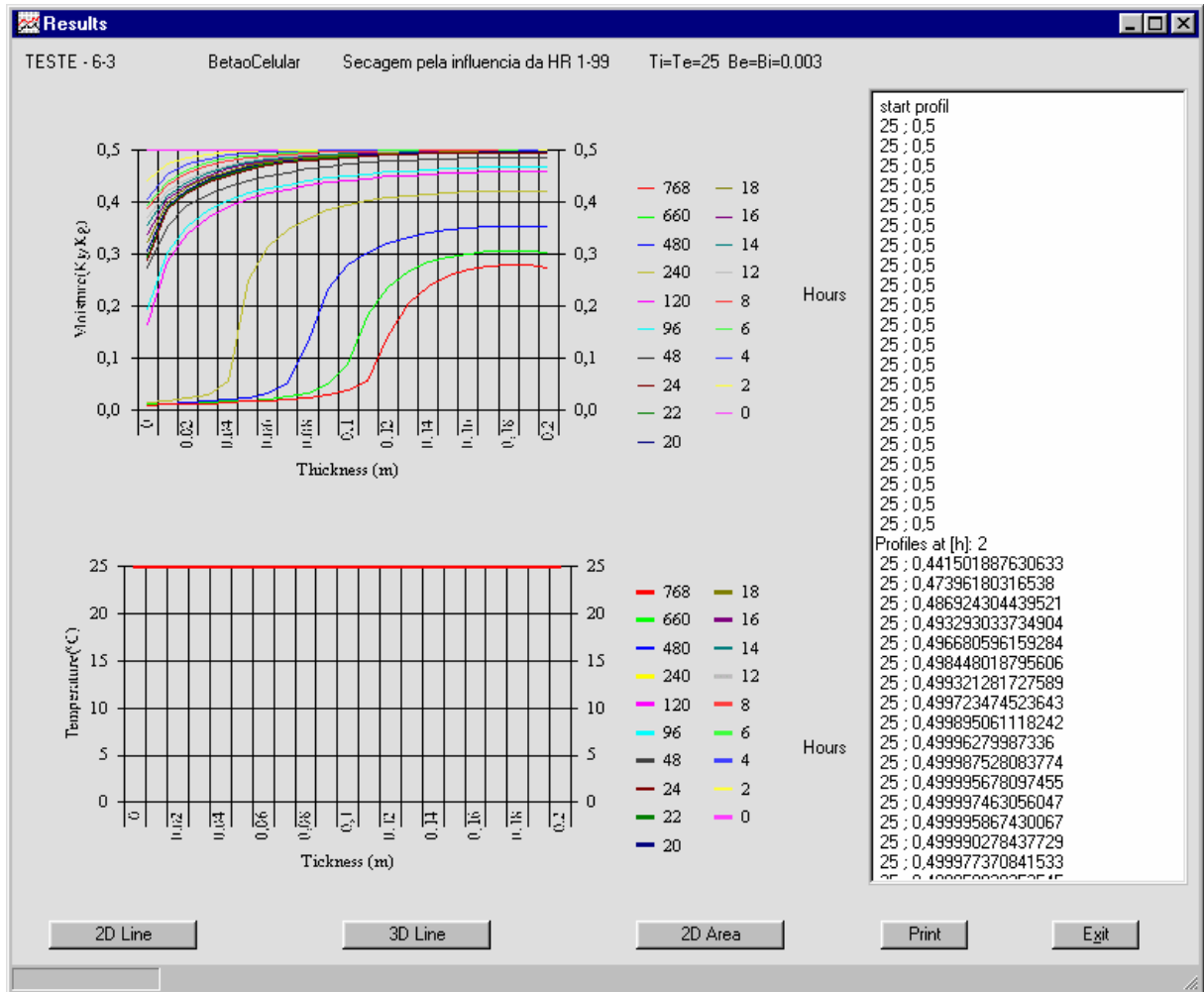


Fig. 113 – "Form frmresults"

Option Explicit

Dim axisId

```

Function chart1()
Dim Column As Integer
Dim Row As Integer
Dim Index1
Dim Index2
Dim Index3
Dim Index4
With frmresults.MSChart1
.ColumnCount = nprofile + 1
.RowCount = gTotalNodes
For Column = 1 To nprofile + 1
For Row = 1 To gTotalNodes
.Row = Row
.Column = Column
.ColumnLabel = time_profile(nprofile + 1 - Column)
.Data = gmoist(Row, nprofile + 1 - Column)
.RowLabel = distance(Row)
Next
Next
.ShowLegend = True
End With
For axisId = VtChAxisIdX To VtChAxisIdY
With MSChart1.Plot.Axis(axisId, 1).AxisTitle
.Visible = True
Select Case axisId
Case 0

```

```

        .Text = "Thickness (m)"
        Case 1
        .Text = "Moisture (Kg/Kg)"
    End Select
End With
Next
End Function

Function chart2()
Dim Column As Integer
Dim Row As Integer
Dim Index1
Dim Index2
Dim Index3
Dim Index4
With frmresults.MSChart2
    .ColumnCount = nprofile + 1
    .RowCount = gTotalNodes

For Column = 1 To nprofile + 1
    For Row = 1 To gTotalNodes
        .Row = Row
        .Column = Column
        .ColumnLabel = time_profile(nprofile + 1 - Column)
        .Data = gtemp(Row, nprofile + 1 - Column)
        .RowLabel = distance(Row)
    Next
Next
.ShowLegend = True
End With
For axisId = VtChAxisIdX To VtChAxisIdY
    With MSChart2.Plot.Axis(axisId, 1).AxisTitle
        .Visible = True
        Select Case axisId
            Case 0
                .Text = "Thickness (m)"
            Case 1
                .Text = "Temperature(°C)"
        End Select
    End With
Next
End Function

```