

CRISTIANO ÉDIO DANNENHAUER

**DESENVOLVIMENTO DE UM APLICATIVO
COMPUTACIONAL PARA MICROBIOLOGIA PREDITIVA**

**FLORIANÓPOLIS – SC
2010**

**UNIVERSIDADE FEDERAL DE SANTA CATARINA
CENTRO TECNOLÓGICO
CURSO DE PÓS-GRADUAÇÃO EM ENGENHARIA DE
ALIMENTOS**

**DESENVOLVIMENTO DE UM APLICATIVO COMPUTACIONAL
PARA MICROBIOLOGIA PREDITIVA**

Dissertação submetida ao Curso de Pós-Graduação em Engenharia de Alimentos da Universidade Federal de Santa Catarina para a obtenção do grau de mestre em Engenharia de Alimentos.

Cristiano Édio Dannenhauer
Engenheiro Químico e Engenheiro de Alimentos

Orientadora: Prof^a Dr^a. Gláucia Maria Falcão de Aragão
Co-orientador: Prof. Dr. João Borges Laurindo

Florianópolis, fevereiro de 2010

Dannenhauer, Cristiano Édio

Desenvolvimento de um Aplicativo Computacional para Microbiologia Preditiva/Cristiano Édio Dannenhauer – Florianópolis, SC, 2010.

73f. : il.,grafs., tabs. 15 cm

Orientador: Gláucia Maria Falcão de Aragão e
Co-orientador: João Borges Laurindo

Dissertação (mestrado) – Universidade Federal de Santa Catarina, Programa de Pós-Graduação em Engenharia de Alimentos.

Inclui referências

1. Microbiologia Preditiva. 2. Modelos. 3. Aplicativo Computacional. I Aragão, Gláucia M. F. II. Universidade Federal de Santa Catarina. Programa de Pós-Graduação em Engenharia de Alimentos. III. Título

Este trabalho foi desenvolvido nos laboratórios de Engenharia Bioquímica (ENGEBIO) e Propriedades Físicas dos Alimentos (PROFI) do Departamento de Engenharia Química e Engenharia de Alimentos da Universidade Federal de Santa Catarina - Florianópolis, SC.

Dedico este trabalho à minha mãe e à memória de
meu pai.

AGRADECIMENTOS

À professora orientadora Gláucia e o professor co-orientador João, pelo grande apoio, dedicação e ensinamento em toda a minha formação.

Aos ex-colegas e amigos do laboratório Luiz Gustavo, Franciny, Vivian e Jaqueline principalmente a Carmen pelo ensino e por proporcionar um ambiente de trabalho agradável e divertido.

Ao professor José Antônio, por aceitar participar da defesa desta dissertação, enriquecendo este trabalho.

À professora Pilar, pelos ensinamentos repassados durante as aulas ministradas e pelas sugestões dadas no início da implementação do aplicativo.

Ao Bruno, pelos anos de ensinamento, críticas, sugestões e principalmente pela grande amizade, fundamental em toda a minha formação acadêmica.

À Francieli, pela ajuda constante em todas as etapas de elaboração do aplicativo, sugestões, críticas e troca de ensinamentos.

Agradeço ao Rodriguinho por desenhar o logotipo do aplicativo.

Aos professores do departamento, Alcilene, Sandra, Haiko, Julian e aos demais professores pelo ensino, ajuda e incentivo.

À UFSC pela oportunidade do desenvolvimento na mesma.

Aos meus irmãos André e Eliane, que apesar de estarem longe sempre apoiaram toda a minha jornada.

Em especial a minha namorada Ana, pulguinha, sempre presente, sacrificando descansos para me apoiar, e ajudando de todas as formas.

E principalmente a minha mãe, Nelsi, por toda sua luta para me manter estudando todos estes anos, pelo amor incondicional e pelas palavras de consolo.

À empresa ESSS, por apoiar na conclusão do mestrado.

E a todos meus colegas de aula e trabalho que me ajudaram direta ou indiretamente para a elaboração deste trabalho.

RESUMO

A microbiologia preditiva pode ser considerada uma ferramenta, baseada em modelos matemáticos, para prever o crescimento e inativação microbiana em alimentos. Modelos que fazem a predição de crescimento dos microrganismos com o tempo, considerando as variações nas condições ambientais (temperatura, pH, concentração de sais ou de inibidores, entre outras) estão sendo cada vez mais estudados. Isso é fundamental, pois as condições de manipulação, processamento e armazenamento de alimentos não ocorrem à mesma temperatura (condições não isotérmicas). O objetivo deste trabalho foi à implementação de uma ferramenta computacional com interface “amigável” para o usuário, onde se pode realizar a modelagem matemática do crescimento de microrganismos sob diversas condições, usando diferentes modelos matemáticos. Os modelos primários utilizados no aplicativo foram: modelo de Gompertz, modelo Logístico, modelo Logístico modificado e o modelo de Baranyi. Entre os modelos secundários estão: modelo tipo Arrhenius, modelo de Raiz Quadrada e modelo de Weibull (equação da potência). Os modelos não isotérmicos de Micha Peleg e de Van Impe também estão disponíveis no aplicativo computacional. O aplicativo possui uma função de armazenamento de dados de crescimento microbiano (banco de dados) que pode ser alimentada com dados de interesse. Toda a interface gráfica do aplicativo, bem como os objetos presente na mesma, foi construída utilizando a plataforma de desenvolvimento Borland C++ Builder 6.0. Para estimativa dos parâmetros dos modelos primários e secundários, foram utilizados modelos de minimização de funções, usando o método de Gauss-Newton e de Levenberg-Marquardt. Nos cálculos das equações diferenciais ordinárias utilizaram-se discretizações baseadas na série de Taylor e em métodos convencionais de interpolação, como é o caso do método de Euler, bem como procedimentos mais sofisticados de interpolação, como o método de Runge-Kutta. O *software* Statística 7.0 foi utilizado para a validação dos resultados encontrados através do aplicativo desenvolvido. Foram calculados diferenças da ordem de 0,001 %, validando o aplicativo desenvolvido.

Palavras-chave: microbiologia prediva, modelos, aplicativo computacional.

ABSTRACT

The predictive microbiology might be considered a tool, based on mathematical models for predicting microbial growth/inactivation in foods. Models that predict the growth of microorganisms with time, considering the variations in environmental conditions (temperature, pH, salt concentration or inhibitors, among others) are increasingly being studied. This is crucial because the conditions of handling, processing and storage of food do not occur at the same temperature (under non-isothermal conditions). The aim of this study was the implementation of a computational tool interface "friendly" for the user, where it is possible to perform the mathematical modeling of microbial growth under various conditions, using different mathematical models. The primary models used in the application were: the Gompertz model, Logistic model, Modified Logistic model and Baranyi model. Among the secondary models are: the Arrhenius model, Square Root model and Weibull model. Two non-isothermal Models developed by Micha Peleg and Van Impe are also available in computer application. The application has a function for storing data of microbial growth (database) that can be fed with particular data. All the graphical interface and the objects present in it were built using the development platform Borland C ++ Builder 6.0. For the optimization of the primary and secondary model parameters, minimize functions such as Gauss-Newton and Levenberg-Marquardt were used. In the calculations of ordinary differential equations were used discretizations based on Taylor series and conventional methods of interpolation, as the Euler's method, as well as more sophisticated interpolation methods, such as Runge-Kutta. The computational software Statistica 7.0 was used to validate the results obtained using the application developed. Were calculated errors on the order of 0,001%, confirming the applicability of the algorithm.

Keywords: predictive microbiology, microbiological models, computer application.

SUMÁRIO

INTRODUÇÃO	1
1. REVISÃO DA LITERATURA	5
1.1 MICROBIOLOGIA DE ALIMENTOS	5
1.1.1 Fatores que permitem a multiplicação da microflora	5
1.1.2 Fatores extrínsecos ao crescimento microbiano	7
1.2 MICROBIOLOGIA PREDITIVA	8
1.2.1 Planejamento	11
1.2.2 Coleta dos dados e análise	11
1.2.3 Descrição Matemática	13
1.2.4 Validação e Manutenção	13
1.2.5 Método dos Mínimos Quadrados	13
1.2.6 Métodos de Resolução de Equações Diferenciais	16
1.3 CLASSIFICAÇÃO DOS MODELOS USADOS EM MICROBIOLOGIA PREDITIVA	19
1.3.1 Modelos Primários	19
1.3.2 Modelos Secundários	24
1.3.3 Modelo da Weibull (equação da potência)	26
1.3.4 Modelos não Isotérmicos - Modelos Dinâmicos	27
1.3.5 Modelos Terciários	31
1.4 O PROCESSO DE DESENVOLVIMENTO DE SOFTWARE	33
1.4.1 As Incertezas de um Projeto de Software	34
1.4.2 Modelos Terciários de Grande Utilização no Meio Acadêmico	35
2. MATERIAL E MÉTODOS	39
2.1 ESTIMATIVA DOS PARÂMETROS DOS MODELOS PRIMÁRIOS E SECUNDÁRIOS	39
2.1.1 Modelos Primários	39
2.1.2 Modelos Secundários	44
2.1.3 Equações diferenciais ordinárias	47
2.1.4 Índices Estatísticos	52
2.2 ESTRUTURA DO PROGRAMA	56
2.2.1 Interface Gráfica da Aplicação	56
2.2.2 Linguagem de Programação	58
2.2.3 Objetos de Interface	58

3. RESULTADOS E DISCUSSÕES	63
3.1 APLICATIVO COMPUTACIONAL	63
3.1.1 Arquitetura do Aplicativo	63
3.1.2 Módulo de Cadastros	66
3.1.3 Módulo de Entrada de Dados	71
3.1.4 Módulo de Processamento dos Modelos e Saída de Dados	74
3.2 VALIDAÇÕES	98
3.2.1 Dados Experimentais	98
3.2.2 Método dos Mínimos Quadrados	99
4. CONCLUSÕES	109
5. REFERÊNCIAS DA LITERATURA	111
6. ANEXO	117

LISTA DE TABELAS

Tabela 2-1: Características de crescimento dos principais patógenos encontrados em carne.	8
Tabela 4-1 – Parâmetros Constantes do Experimento	98
Tabela 4-2 – Dados Experimentais de razão de crescimento (N/N_0) versus tempo (h)	99
Tabela 4-3 – Comparação dos resultados do modelo de Gompertz utilizando o método de Gauss-Newton	102
Tabela 4-4 – Análise da Variância: Modelo de Gompertz utilizando o método de Gauss-Newton	102
Tabela 4-5 - Comparação dos resultados do modelo de Gompertz utilizando o método de Levenberg-Marquardt	103
Tabela 4-6 - Análise da Variância: Modelo de Gompertz utilizando o método de Levenberg-Marquardt	103
Tabela 4-7 – Parâmetros Primários variando conforme a temperatura	104
Tabela 4-8 - Parâmetros Primários linearizados conforme o modelo da Raiz Quadrada	105
Tabela 4-9 – Valores dos coeficientes de ajuste linear gerados pelo software <i>Statística</i> e o aplicativo em desenvolvimento	107
Tabela 7-1 – Parâmetros Estimados – Modelo Logístico utilizando o método de Gauss-Newton	118
Tabela 7-2 - Análise da Variância: Modelo Logístico utilizando o método de Gauss-Newton	118
Tabela 7-3 - Parâmetros Estimados – Modelo Logístico utilizando o método de Levenberg-Marquardt	119
Tabela 7-4 - Análise da Variância: Modelo Logístico Modificado utilizando o método de Levenberg-Marquardt	119
Tabela 7-5 - Parâmetros Estimados – Modelo Logístico Modificado utilizando o método de Gauss-Newton	121
Tabela 7-6 - Análise da Variância: Modelo Logístico Modificado utilizando o método de Gauss-Newton	121
Tabela 7-7 - Parâmetros Estimados – Modelo Logístico Modificado utilizando o método de Levenberg-Marquardt	122
Tabela 7-8 - Análise da Variância: Modelo Logístico Modificado utilizando o método de Levenberg-Marquardt	122
Tabela 7-9 - Valores dos coeficiente de ajuste linear gerados pelo software <i>Statística</i> e o aplicativo em desenvolvimento	124
Tabela 7-10 - Valores dos coeficientes de ajuste linear gerados pelo software <i>Statística</i> e o aplicativo em desenvolvimento	125

LISTA DE FIGURAS

Figura 1-1 Curva típica de crescimento microbiano. Error! Bookmark not defined.	
Figura 1-2 - Curva de crescimento microbiano com indicação dos parâmetros biológicos.	20
Figura 1-3 - O cone de incerteza de um projeto de software. O desvio nas estimativas de tamanho diminui à medida que o processo de desenvolvimento avança.....	35
Figura 1-4 – Screenshot do GlnaFit	37
Figura 2-1: Fluxograma para resolução dos métodos de mínimos quadrados	47
Figura 2-2 – Interface gráfica da aplicação	57
Figura 2-3 – Componente: Button	58
Figura 2-4 – Componente: CheckBox	59
Figura 2-5 - Componente: RadioButton.....	59
Figura 2-6 – Componente: <i>ComboBox</i>	59
Figura 2-7 – Componente: <i>GroupBox</i>	60
Figura 2-8 – Componente: <i>PageControl</i>	60
Figura 2-9 – Componente: <i>CheckListBox</i>	60
Figura 2-10 – Componente: <i>StringGrid</i>	61
Figura 3-1 – Fluxograma da Arquitetura do Aplicativo	64
Figura 3-2 – Janela Principal do Aplicativo	66
Figura 3-3 - Cadastro de Usuários	67
Figura 3-4 – Cadastro de Micro-organismos	68
Figura 3-5 – Cadastro de Cidades.....	69
Figura 3-6 – Efetuação do <i>Login</i>	70
Figura 3-7 – Confirmação do <i>Login</i>	70
Figura 3-8 – Gerenciamento de Dados	72
Figura 3-9 – Gerenciamento de Dados: Aba Dados Experimentais	73
Figura 3-10 – Gerenciador de dados Aba: Modelo Primário.....	74
Figura 3-11 – Gerenciador de dados: Tela Exportar	74
Figura 3-12 – Escolha do modelo primário.....	75
Figura 3-13 – Dados Experimentais do Modelo Primário.....	76
Figura 3-14 – Configurações Avançadas dos Métodos de Resolução de Problemas de Mínimos Quadrados	79
Figura 3-15 – Configurações Avançadas: Inicializador	80
Figura 3-16 – Modelo Primário: Aba Resultados.....	81
Figura 3-17 – Cálculo do velocidade máxima de crescimento.....	82
Figura 3-18 – Molelo Primário: Aba Gráficos	83
Figura 3-19 – Modelos Primários: Ajuste de Gráficos.....	83

Figura 3-20 – Modelos Primários: Aba Estatística	84
Figura 3-21 – Modelo Primário: Aba Gráfico - Observado x Predito.	85
Figura 3-22 – Gerenciador de Dados: Modelos Primários – aba Seleção de Dados.....	86
Figura 3-23 – Gerenciador de Dados: Modelos Primários – aba Tabela de Dados.....	87
Figura 3-24 - – Gerenciador de Dados: Modelos Primários – aba Gráficos.....	88
Figura 3-25 - Dados do Modelo Secundário	89
Figura 3-26 – Gráficos do Modelo Secundário.....	90
Figura 3-27 – Resultados do Modelo Secundário	90
Figura 3-28 – Banco de Dados do Modelo Secundário	91
Figura 3-29 – Gerenciador de Dados: Modelos Secundários – aba Seleção de Dados.....	92
Figura 3-30 – Gerenciador de Dados: Modelos Secundários – aba Tabela de Dados.....	93
Figura 3-31 - – Gerenciador de Dados: Modelos Secundários – aba Gráficos.....	93
Figura 3-32 - Modelos Não isotérmicos – aba Seleção de Dados...	94
Figura 3-33 - Modelos Não isotérmicos – aba Variação de Temperatura.....	95
Figura 3-34 - Modelos Não-Isotérmicos – aba Tabela de Dados Secundários	96
Figura 3-35 - Modelos Não isotérmicos – aba Constantes do Modelo	97
Figura 3-36 - Modelos Não isotérmicos – aba Gráficos	97
Figura 3-37 – Curvas de crescimento geradas pelos parâmetros primários calculados pelo Statística 7.0 e o aplicativo em desenvolvimento	101
Figura 3-38 – Histograma de erros percentuais dos parâmetros matemáticos dos modelos primários	104
Figura 3-39 – Curvas de ajuste dos parâmetros primários pela temperatura usando o <i>Statística 7.0</i> e o aplicativo em desenvolvimento	106
Figura 3-40 – Histograma de erros relativos dos coeficientes gerados pela linearização do modelo secundário.	107
Figura 6-1 - Curvas de crescimento geradas pelos parâmetros primários calculados pelo Statística 7.0 e o aplicativo em desenvolvimento	117
Figura 6-2 - Histograma de erros percentuais dos parâmetros matemáticos dos modelos primários – Modelo Logístico.....	120

Figura 6-3 - Curvas de crescimento geradas pelos parâmetros primários calculados pelo Statistica 7.0 e o aplicativo em desenvolvimento	120
Figura 6-4 - Histograma de erros percentuais dos parâmetros matemáticos dos modelos primários – Modelo Logístico Modificado	123
Figura 6-5 – Histograma de erros relativos dos coeficientes gerados pela linearização do modelo secundário – Modelo de Arrhenius	124
Figura 6-6 – Histograma de erros relativos dos coeficientes gerados pela linearização do modelo secundário.....	125

NOMENCLATURA

Símbolo	Descrição	Unidade
A	Logaritmo da densidade máxima da população	adimensional
A	Parâmetro matemático do modelo primário	adimensional
a	Parâmetro obtido pelo ajuste do modelo de Arrhenius	adimensional
a_w	Atividade de água	adimensional
B	Parâmetro matemático do modelo primário	h^{-1}
D	Parâmetro matemático do modelo primário	h^{-1}
E_a	Energia de Ativação do crescimento microbológico	kJ.mol^{-1}
J	<i>Matriz Jacobiana</i>	
k	Parâmetro de crescimento do modelo (μ , λ ou A)	adimensional
k_1	Parâmetro de interpolação do modelo de Runge-Kutta	adimensional
k_2	Parâmetro de interpolação do modelo de Runge-Kutta	adimensional
k_3	Parâmetro de interpolação do modelo de Runge-Kutta	adimensional
k_4	Parâmetro de interpolação do modelo de Runge-Kutta	adimensional
n	Número de observações	
M	Parâmetro matemático do modelo primário	adimensional
MSE	<i>Mean Square Error</i> ou Erro Quadrado Médio	adimensional
p	Pressão de vapor da água no alimento	Pa
p_o	Pressão de valor da água pura	Pa
R	Constante dos gases ideais	$\text{J.K}^{-1}.\text{mol}^{-1}$
t	Distribuição de t-Student	adimensional

t	Tempo	h
T	Temperatura	$^{\circ}C$
T_{min}	Temperatura mínima teórica para o crescimento	$^{\circ}C$
t_c	Ponto de inflexão na curva	h
x	Concentração celular	UFC/mL
x	Absorbância final	UFC/mL
x_{max}	Máxima concentração celular	UFC/mL
x_o	Concentração celular inicial	UFC/mL
x_o	Absorbância inicial	UFC/mL
y_{max}	Máxima concentração celular	UFC/mL
u	Transição para a fase estacionária da curva de crescimento microbiológico	
UFC	Unidade formadora de colônia	adimensional
V	Variável independente com distribuição chi-quadrado	adimensional
Z	Variável independente com distribuição normal	adimensional
α	Nível de confiança	adimensional
Γ	Função gama	adimensional
γ	Intervalo de confiança	adimensional
λ	Duração da fase lag	h
σ	Desvio Padrão	adimensional
ν	Velocidade específica de crescimento	h^{-1}
ν	Graus de liberdade	adimensional
μ	Velocidade específica de crescimento	h^{-1}
μ_m	Velocidade de crescimento exponencial média	h^{-1}

INTRODUÇÃO

Cada vez mais pesquisas são realizadas buscando o aumento de produção de alimentos para suprir a demanda atual e futura exigida pelo grande crescimento da população. Cientistas e especialistas em desenvolvimento em todo o mundo estão buscando formas de aumentar a produção de alimentos em 50% nas próximas duas décadas, para alimentar a população crescente do planeta. Apesar do consenso amplo de que há terra, água e conhecimento suficiente, muitos duvidam das chances de sucesso. (*New York Times*, Jan/2010).

A busca de melhorar a distribuição de alimentos no mundo interno criou a necessidade de estudos massivos para incrementar o tempo que o alimento permanece próprio para o consumo. A deterioração dos alimentos é um processo complexo e quantidades expressivas de alimentos são perdidas devido à deterioração microbiana. A velocidade das reações físico-químicas no alimento depende da temperatura, do pH e da atividade de água do alimento. O controle destes parâmetros durante o armazenamento é utilizado pela maioria das indústrias para preservar a qualidade do alimento e desta maneira estender a vida de prateleira dos mesmos (PEXARA et al., 2002).

A microbiologia preditiva descreve, de forma quantitativa, os efeitos das condições ambientais (temperatura, pH, concentrações de sais ou de inibidores, entre outras) no crescimento e inativação microbiana nos alimentos. Para isso, faz uso de modelos matemáticos derivados de estudos quantitativos dos microrganismos sob dadas condições ambientais. Em outras palavras, a microbiologia preditiva é baseada em curvas experimentais de crescimento/inativação microbiana sob dadas condições de temperatura, pH, concentração de nutrientes, concentração de sais, presença de inibidores, etc. O conhecimento do crescimento de um dado micro-organismo sob diferentes condições permite modelar a influencia de cada parâmetro ambiental sobre este crescimento.

A necessidade de garantir a segurança microbiológica e a qualidade dos alimentos tem estimulado a aplicação da microbiologia preditiva (NAKASHIMA et al., 2000), pois os

modelos preditivos podem ser usados para avaliar a vida-de-prateleira (*shelf-life*) de um determinado produto e a segurança ou risco alimentar que este produto pode oferecer.

Devido à quantidade de modelos de crescimento microbiano e o grande acúmulo de informações que são geradas, cada vez mais é necessário o uso de aplicativos computacionais que permitam o tratamento adequado dos dados. Os modelos apresentados nos aplicativos permitem que o usuário modele curvas de crescimento microbiano sob diferentes condições ambientais, com base em dados experimentais obtidos nas condições de estudo (faixa de temperatura, pH, etc). Recentemente, a modelagem matemática do crescimento microbiano está orientada para obtenção de modelos dinâmicos, ou seja, modelos que permitam prever a segurança ou vida útil dos alimentos sob condições que variam com o tempo (PELEG, 2006).

Atualmente, há inúmeros *softwares* para o uso em microbiologia preditiva disponíveis para a utilização tanto para pesquisadores como para empresas e entre eles se destacam duas ferramentas gratuitas, o DMFit que é parte de um sistema utilizado no *Institut of Food Research* (IFR) para modelar a concentração celular logarítmica de lotes de culturas bacterianas pelo tempo. (DMFit Manual). E a outra ferramenta utilizada é o GlnaFit, que tem como objetivo eliminar a lacuna hoje existente entre os desenvolvedores de modelos de predição e os usuários finais na indústria de alimentos que não estão familiarizados com ferramentas avançadas de regressão não linear. O GlnaFit é uma ferramenta útil para testar diferentes tipos de modelos de inativação microbiana (GEERAERD, 2005). Ambas as ferramentas possuem características similares, da utilização junto com o aplicativo Microsoft Excel, isto é, não podem ser usadas de maneira independente ao aplicativo da Microsoft.

Deste modo, o objetivo deste trabalho foi a construção de um aplicativo computacional de interface autônoma e amigável que permita a seleção criteriosa de: i) modelos matemáticos de crescimento microbiano sob condições ambientais constantes, (condições isotérmicas, por exemplo) (modelos primários), ii) modelos matemáticos que representem a influência das condições ambientais sobre o crescimento microbiano (modelos secundários) e iii) prever o crescimento microbiano em

condições ambientais variáveis (condições não isotérmicas, por exemplo, nos chamados modelos dinâmicos).

1. REVISÃO DA LITERATURA

1.1 MICROBIOLOGIA DE ALIMENTOS

A microbiologia de alimentos estuda a influência dos microrganismos nas características dos produtos alimentícios de consumo humano ou animal.

A qualidade microbiológica dos alimentos relaciona primeiramente à quantidade e ao tipo de microrganismos inicialmente presentes (contaminação inicial) e depois à multiplicação destes no alimento. (FL Hoffmann - Brasil Alimentos, São Paulo, 2001).

A microbiologia de alimentos adotou métodos modernos e conceitos inovadores com alguma resistência. Muitos microbiologistas de alimentos, não familiarizados com as ferramentas matemáticas, seguem a abordagem "antiquada" de enumerar os micro-organismos em diferentes estágios de conservação de alimentos, identificando as principais frações da microflora por seus caracteres fenotípicos e, gradualmente, construindo uma compreensão da vida útil e a segurança dos alimentos (BARANYI et. al, 2004). A utilização da microbiologia preditiva ou da modelagem matemática auxilia na compreensão e tomada de decisões sobre os processo e vida útil de alimentos.

1.1.1 Fatores que permitem a multiplicação da microflora

Segundo Massaguer (2005), os parâmetros que influenciam na seleção da microflora e que permitem sua multiplicação são:

1.1.1.1 Fatores intrínsecos

Compreendem a expressão das propriedades físicas e a composição do alimento. Os fatores intrínsecos são atividade de água, acidez e poder tampão do alimento, potencial redox, nutrientes, agentes antibacterianos naturalmente presentes e estruturas.

1.1.1.2 Atividade de Água

Atividade de água, a_w , é uma propriedade física que tem uma implicação direta na segurança microbiológica de alimentos e também influencia a estabilidade no armazenamento de alimentos visto que alguns processos deterioradores em alimentos são propagados pela água. (GIBBS et al, 2001)

Segundo Massaguer (2005), “A influência da atividade de água depende de outros fatores, tais como a adição de sal, que altera a a_w junto com a temperatura de estocagem, e estes junto com o pH agem como inibidores”.

A a_w de um alimento ou solução é a relação entre a pressão de vapor de água do alimento ou solução (p) e a da água pura (p_0) na mesma temperatura:

$$a_w = \frac{p}{p_0} \quad (1-1)$$

A a_w está relacionada com os pontos de ebulição e congelamento, umidade relativa de equilíbrio e pressão osmótica. A a_w de uma solução é uma propriedade coligativa, ou seja, depende do número de partículas (moléculas ou íons) em solução. Aumentos na concentração de solutos diminuem a a_w . (GIBBS et al, 2001)

1.1.1.3 Acidez

A grande maioria dos micro-organismos crescem na faixa de pH entre 6,6 e 7,5. Os fungos e leveduras e bactérias lácticas crescem a pH < 4,0.

O efeito inibidor do pH depende também do tipo de ácido presente no alimento e dos nutrientes, a_w , temperatura e pressão parcial de O_2 .

1.1.1.4 Potencial Redox (Eh)

O potencial redox é o índice do grau de oxidação de um sistema biológico. Quando um elemento perde elétrons, o substrato é oxidado e o que ganha elétrons é reduzido. Uma substância que cede elétrons é um agente redutor e, por outro lado, a que captura elétrons é um agente oxidante.

A classificação dos micro-organismos em aeróbios, que necessitam da presença de oxigênio para obter energia; anaeróbios, que realizam as reações químicas na ausência de oxigênio, se faz em relação ao Eh crítico para o crescimento. Quando o Eh é baixo, os micro-organismos anaeróbios crescem e os aeróbios são inibidos.

O Eh depende do grau de oxidação e do pH do próprio alimento e também da circulação de outras substâncias oxidantes e redutoras. (MASSAGUER, 2005)

1.1.2 Fatores extrínsecos ao crescimento microbiano

Partindo do princípio da disponibilidade de nutrientes, o crescimento microbiano em alimentos é controlado principalmente pelo pH, a_w e temperatura de estocagem, com fatores adicionais, como presença de conservantes, embalagem com atmosfera modificada ou tratamento térmico que também contribuem para o controle. Na prática, muitos desses fatores atuam combinados, geralmente em níveis que, isoladamente, não seriam suficientes para controlar o crescimento microbiano. (GIBSON et al., 1987)

1.1.2.1 Temperatura

Quanto maior a temperatura, mais rapidamente crescem os micro-organismos convergindo, possivelmente, a um aumento para um nível associado com alto risco de doenças que podem chegar aos consumidores de produtos contaminados. A Tabela 1-1 apresenta as temperaturas mínimas para o crescimento de cada espécie (temperatura abaixo das quais o micro-organismo não cresce), o tempo de latência (tempo no qual as bactérias se ajustam a um novo ambiente e não aumentam em número) e do tempo de geração (o tempo para que as bactérias dobrem em número). Acima de temperaturas mínimas de crescimento, as bactérias eventualmente crescem para níveis perigosos (COATES, 2004).

Tabela 1-1: Características de crescimento dos principais patógenos encontrados em carne. (ICMSF, 1996).

Patógeno	Min. temperatura de crescimento (oC)
<i>Salmonella</i>	10
<i>EHEC</i>	9
<i>Listeria monocytogenes</i>	<4

Fonte: (SUPPLY CHAINS TO SINGAPORE K. COATES, *Department of Primary Industries*, 475 Mickleham Rd, Attwood, Vic 3049).

Cada microrganismo possui sua temperatura ótima de crescimento. A maioria cresce bem entre 15 e 40 °C. Os microrganismos são classificados de acordo com as temperaturas ótimas de crescimento, conforme segue (MASSAGUER, 2005):

Psicrófilos: Crescem bem abaixo de 20 °C (leveduras, fungos e algumas bactérias – *Pseudomonas*, *Achromobacter*, *Micrococcus*, *Lactobacillus*, *Streptococcus*, *Leuconostoc*, *Pediococcus*, *Flavobacterium* e *Proteus*), sendo que *Pediococcus* e *Flavobacterium* e *Proteus* podem produzir substâncias histaminiformes em pescado sem alterar o frescor do produto;

Psicrotrófilos: Crescem bem entre 20 e 30 °C

Mesófilos: Crescem bem entre 30 e 37 °C;

Termófilos: Crescem acima de 50 °C até 85 °C.

1.2 MICROBIOLOGIA PREDITIVA

O objetivo da microbiologia preditiva é minimizar ou prevenir o crescimento microbiano ao invés de otimizá-lo. Conseqüentemente, efeitos dos fatores inibidores do meio, baixas temperaturas e controle da cadeia de frio têm sido investigados mais intensivamente (BARANYI & ROBERTS, 1994).

McMeekin et al. (1993) define Microbiologia Preditiva como uma ciência quantitativa que habilita os usuários a avaliar objetivamente o efeito das operações de processamento, distribuição e estocagem na segurança microbiológica e qualidade de alimentos.

A microbiologia preditiva unifica a microbiologia, a engenharia e a estatística para obter predições sobre o

comportamento microbiológico em sistemas alimentares, usando ferramentas matemáticas. Pode-se fazer uso de expressões para avaliar a população microbiana com o tempo, avaliar como as condições ambientais afetam a velocidade de crescimento ou inativação, também como o comportamento dos alimentos concorre para o desenvolvimento microbiológico. Desta forma, informações importantes na tomada de decisão, na análise de risco, na segurança e qualidade dos alimentos, na avaliação da vida de prateleira, bem como no desenvolvimento de novos produtos ou processos são fornecidas (SARMENTO, 2006).

Para estabelecer a segurança alimentar são feitos tradicionalmente *challenge tests* com os patógenos de interesse. Esta abordagem tem sido considerada cara, lenta e exigente sobre as instalações e as competências microbiológicas. O conhecimento adquirido não é cumulativo e quando uma formulação do produto, ou a temperatura do perfil temporal estão sujeitos a mudanças, os testes de controle devem ser repetidos. Uma alternativa é compreender melhor as respostas dos microorganismos de interesse sob determinadas condições, construindo uma base cumulativa de conhecimento, e desenvolvendo meios para interpolar as respostas microbianas calculadas. (BARANYI & ROBERTS, 1995).

Os modelos preditivos são usados atualmente como ferramenta de investigação e avaliação dos processos de conservação de alimentos. A modelagem preditiva proporciona um caminho rápido e ainda relativamente econômico para se obter dados confiáveis de crescimento, inativação e sobrevivência no desenvolvimento microbiano. A microbiologia preditiva poderá prever parâmetros como velocidade máxima de crescimento, duração da fase lag e população microbiana máxima atingida, no desenvolvimento de novos produtos, na fase de produção, armazenamento e distribuição, podendo assim, avaliar a vida de prateleira dos produtos alimentícios (SARMENTO, 2006).

Com o objetivo de estimar os parâmetros cinéticos de crescimento, foi proposto uma série de modelos matemáticos. O desenvolvimento de modelos preditivos requer uma grande quantidade de dados de crescimento. O tempo natural, relativamente longo, de técnicas de contagem, levou à necessidade de métodos de coleta mais rápidos e convenientes,

que representariam consideráveis economias em esforço e recursos. (COLE, 1991 apud CARRASCO, 2006)

As variáveis construídas no modelo matemático são divididas em três classes (BARANYI & ROBERTS, 1994):

Condições extracelulares não afetadas, ou insignificamente afetadas pelo crescimento de células que caracterizam o crescimento independente do meio, que pode mudar com o tempo. O componente mais importante do meio de crescimento independente é geralmente a temperatura. Desde que as células produzam calor crescente, a temperatura não deve ser considerada uma variável de crescimento independente. No entanto, uma pequena contribuição do calor produzido é normalmente negligenciada.

Condições extracelulares que são alteradas pelo crescimento da cultura. Estas serão denominadas de crescimento dependente do meio e incluem, por exemplo, a concentração de vários químicos como substratos para crescimento e produtos metabólicos que cercam as células.

Concentração intracelular de determinada substância, como DNA, RNA, que mudam durante o crescimento e caracterizam o estado fisiológico das células. É importante compreender que a fronteira entre o crescimento independente e dependente não é fixa para todos os modelos, e um crescimento variável em um determinado modelo pode representar um crescimento dependente em outro, mais sofisticado.

Apesar de empíricos, os modelos matemáticos definidos para avaliar a segurança e qualidade dos alimentos são baseados em técnicas de regressão linear e não linear. Como em qualquer aplicação estatística de regressão, suposições quanto à aleatoriedade, normalidade, interpolação dentro da faixa de variação dos fatores testados ao invés de extrapolações e especificações estocásticas devem ser estabelecidas (McMEEKIN et al., 1993, RATKOWSKY, 1993b apud NAKASHIMA et al., 2000).

Os vários estágios do desenvolvimento de um modelo podem ser definidos da seguinte forma (DRAPER, SMITH, 1981 e FARBER, 1986 apud NAKASHIMA et al., 2000):

1.2.1 *Planejamento*

Nessa etapa, devem ser definidas algumas características relacionadas à(s) variável(eis) independente(s), que compreendem basicamente pH, temperatura, atividade de água; à variável dependente, ao inóculo e ao modelo experimental.

A variável dependente é a resposta primária medida. É usualmente a mudança da densidade da população bacteriana ao longo do tempo. A velocidade de mudança é frequentemente expressa como sendo a velocidade de crescimento, podendo também ser representada pelo tempo de duração da fase lag, tempo de geração, ou o tempo para se atingir uma determinada densidade.

O número de pontos, ou seja, a quantidade de valores da variável dependente, necessários para ajustar um modelo de crescimento ou inativação depende da complexidade desse modelo. Whiting (1997) indica que 7 a 10 pontos de qualidade apropriadamente distribuídos são provavelmente suficientes, enquanto outros autores recomendam o uso de 10 a 15 pontos (GIBSON et al., 1987, BRATCHEL et al., 1989, McMEEKIN et al., 1993). A quantidade de pontos experimentais depende também da distribuição dos mesmos, de nada adianta uma grande amostragem na fase lag, como por exemplo, e poucos pontos nas fases log e estacionária.

Quanto ao inóculo, uma mistura de 3 a 5 cepas típicas de um mesmo patógeno é frequentemente utilizada para inocular o meio de crescimento que será modelado. Esta prática é experimentalmente viável e resulta em modelos para as cepas de crescimento mais rápido ou de maior resistência nas condições ambientais de estudo. O uso de uma mistura de cepas aumenta a probabilidade de se criar um modelo representativo da situação real.

1.2.2 *Coleta dos dados e análise*

Faz-se necessário, nesta etapa, a análise da curva de crescimento microbiano, bem como o conhecimento de cada uma das fases nela apresentadas, conforme apresentado na Figura 1-1:

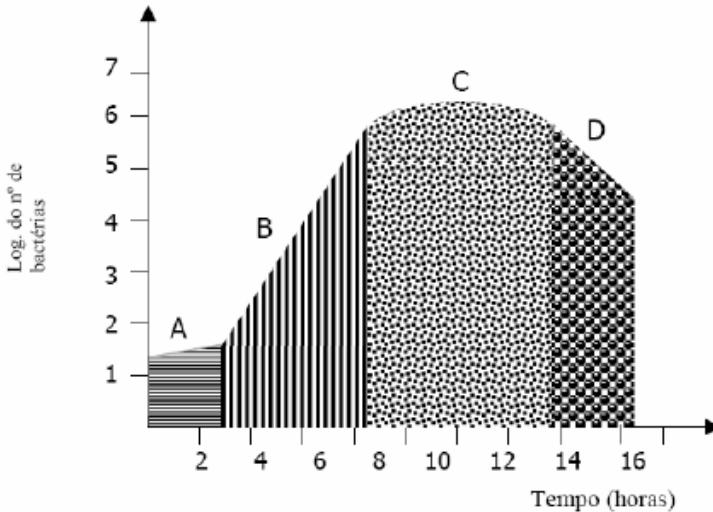


Figura 1-1 Curva típica de crescimento microbiano.

FONTE:

(www.fop.unicamp.br/micro/aulas/fisiologia_bacteriana.pdf)

Fase lag (A): ajuste da fisiologia e bioquímica das células para que possam ser capazes de explorar o ambiente onde se encontram.

Fase exponencial (B): fase de crescimento balanceado, ou seja, a síntese de cada componente celular (enzimas, moléculas estruturais, DNA, etc.) é ajustada para que não exista síntese além do necessário para a produção de novas células. Todo o metabolismo está direcionado para a reprodução. Nesta fase, todos os componentes celulares estão presentes em proporções constantes e as células são consideradas, para aspectos práticos, fisiologicamente idênticas.

A velocidade de crescimento, na fase exponencial, pode ser expressa como velocidade de crescimento absoluta, que indica o aumento da densidade da população com o tempo, ou como velocidade de crescimento relativa ou específica, que indica o aumento da densidade da população em um determinado intervalo de tempo, dividido pela densidade de células no tempo final deste intervalo. O tempo necessário para a população microbiana dobrar é denominado tempo de geração.

Fase estacionária (C): morte celular e lise devido ao acúmulo de metabólitos tóxicos no meio de crescimento. Nesta fase, a velocidade de morte celular é equivalente à velocidade de crescimento. Em condições que permitam o crescimento celular, a densidade microbiana nesta fase é tipicamente 10^9 UFC/mL.

Fase de declínio (D): Maior acúmulo de substâncias tóxicas, determinando uma velocidade de morte celular maior do que a capacidade do meio em proporcionar a reprodução ou divisão celular.

1.2.3 *Descrição Matemática*

Vários modelos matemáticos foram criados para estimar os efeitos de fatores antimicrobianos e outros como temperatura, atividade de água e pH. (CARRASCO et. al, 2006).

Os modelos matemáticos podem ser classificados de acordo com os parâmetros em: lineares ou não-lineares. Como consequência da não-linearidade de um modelo de regressão é o uso de métodos de mínimos quadrados para otimização dos seus parâmetros. (NAKASHIMA et. al, 2000).

Existem vários procedimentos numéricos interativos para a obtenção das estimativas de mínimos quadrados dos parâmetros de um modelo não-linear. Entre eles pode ser citado o método de Gauss-Newton e Levenberg-Marquardt (NETER et al., 1996 apud NAKASHIMA et.al, 2000).

1.2.4 *Validação e Manutenção*

A aplicação de uma técnica estatística adequada permite uma avaliação da capacidade de previsão do modelo que está sendo validado. Independente do quanto um modelo se ajusta ao conjunto de dados que o geraram, o seu valor real está baseado no quão bem ele pode prever a variável de resposta, sob condições não testadas especificamente para derivá-lo (NAKASHIMA et.al., 2000).

1.2.5 *Método dos Mínimos Quadrados*

A estimativa de parâmetros não-linear consiste em um problema de otimização, também conhecido como regressão não linear. Na estimativa desses parâmetros, é comum o emprego do método dos mínimos quadrados.

Seja F a função modelo (que representa matematicamente o modelo a ser ajustado aos dados) e $\{x_i, y_i\}$ o conjunto de n observações das variáveis dependente (y) e independente (x) do modelo. A função-objetivo ($E(\bar{\alpha})$) é definida como:

$$E(\bar{\alpha}) = \sum_{i=1}^n [y_i - F(\bar{\alpha}, x_i)]^2 \quad (1-2)$$

onde $\bar{\alpha}$ = parâmetro otimizado.

Em muitos casos, a função-modelo F é uma função não linear dos parâmetros desconhecidos, de modo que a função – objetivo definida na Equação 1-2 também se torna uma função não linear dos parâmetros a serem estimados. Assim o processo de minimização da função-objetivo requer algum tipo de linearização dessa função antes que o parâmetro $\bar{\alpha}$ possa ser estimado. Vários métodos podem ser utilizados nesse processo, serão destacados aqui os métodos de Gauss-Newton e Levenberg-Marquardt. (ROSA, 2006)

1.2.5.1 Método de Gauss - Newton

O método de Gauss – Newton é um método utilizado para resolver problemas de mínimos quadrados para equações não-lineares. Pode ser visto como uma modificação do método de Newton para encontrar um mínimo de uma função.

Ao contrário do método de Newton, Gauss-Newton pode somente ser usado para minimizar o somatório de valores de funções quadradas, mas tem a vantagem de que derivadas secundárias não são necessárias.

Problemas de mínimos quadrados não-lineares surgem, por exemplo, na regressão não-linear, onde parâmetros do modelo são procurados de tal forma que o modelo esteja em boa concordância com as observações disponíveis.

Dadas m funções r_1, \dots, r_m de n variáveis $\beta = (\beta_1, \dots, \beta_n)$, com $m \geq n$, o algoritmo de Gauss–Newton encontra o mínimo de somatório dos quadrados, segundo a Equação 1-3. .

$$S(\beta) = \sum_{i=1}^m r^2(\beta) \quad (1-3)$$

Iniciando com uma hipótese inicial $\beta^{(0)}$ para o mínimo, o método avança pelas iterações

$$\beta^{(s+1)} = \beta^{(s)} + \Delta \quad (1-4)$$

Onde o incremento Δ é a solução para a equação normal:

$$(J_r^T J_r) \Delta = -J_r^T r \quad (1-5)$$

Neste caso, r é o vetor das funções r_i , e J_r é a Matriz Jacobiana de r com respectivo β , ambos avaliados em β^s . O sobrescrito T denota a matriz transposta (BJÖRCK, 1996).

1.2.5.2 Método de Levenberg–Marquardt

Em matemática e computação, o método de Levenberg–Marquardt (LMA) fornece uma solução numérica para o problema de minimizar uma função, geralmente não-linear, ao longo de um espaço de parâmetros da função. Esses problemas de minimização surgem especialmente em ajustes de curvas de mínimos quadrados e programação não linear.

O LMA interpola entre o algoritmo de Gauss – Newton (GNA) e o método de gradiente descendente. O LMA é mais robusto que o GNA, que significa que em muitos casos encontra uma solução mesmo iniciando muito longe de um mínimo final.

Por outro lado, para funções bem comportadas, como por exemplo funções polinomiais de baixa ordem, e parâmetros iniciais razoáveis, o LMA tem o tempo de convergência mais devagar que o GNA.

A principal aplicação do método de Levenberg – Marquardt é no problema de ajuste da curva de mínimos quadrados; dado um conjunto de pares de dados experimentais de variáveis dependentes e independentes, (x_i, y_i) . Dado o conjunto de dados (x,y) e os parâmetros β da curva do modelo $f(x, \beta)$, é feita a otimização dos destes parâmetros de modo que a soma dos

quadrados dos desvios torne-se mínima, como mostra a Equação 1-6.

$$S(\beta) = \sum_{i=1}^m [y_i - f(x_i, \beta)]^2 \quad (1-6)$$

À semelhança de outros métodos de minimização numérica, o método de Levenberg – Marquardt é um processo iterativo. Para iniciar a minimização, o usuário precisa fornecer um valor inicial para o vetor parâmetro, β .

A cada iteração, o vetor parâmetro β é substituído pelo novo estimado, $\beta + \delta$. Para determinar δ , as funções $f(x_i, \beta + \delta)$ são aproximadas pelas suas linearizações $f(x_i, \beta + \delta) \approx f(x_i, \beta) + J_i \delta$.

Onde o incremento Δ é a solução para a equação normal:

$$\left((J^T J) + \lambda \text{diag}(J^T J) \right) \Delta = J^T r \quad (1-7)$$

Neste caso, r é o vetor das funções r_i , e J é a Matriz Jacobiana de r com respectivo β , ambos avaliados em β^s . O sobrescrito T denota a matriz transposta (BJÖRCK, 1996).

1.2.6 Métodos de Resolução de Equações Diferenciais

Em linhas gerais, qualquer situação concernente à taxa de variações de uma variável com respeito à outra leva a uma equação diferencial (DORN, 1978).

Equações diferenciais aparecem com grande frequência em modelos que descrevem quantitativamente fenômenos em diversas áreas, como mecânicas dos fluidos, fluxo de calor, reações químicas e nucleares, economia e biologia e em microbiologia preditiva, condições ambientais variáveis como temperatura e pH dependentes do tempo .

Serão descritos a seguir dois métodos de resolução de equações diferenciais ordinárias, Método de Euler e Método de Runge – Kutta.

1.2.6.1 Método de Euler

Método numérico que se pode aplicar à solução aproximada de um PVI (problema de valor inicial), que consiste em x_0 e $y_0 = y(x_0)$. Com isso se pode calcular $y'(x_0) = f(x_0, y_0)$. Assim a reta que passa por (x_0, y_0) com coeficiente angular $y'(x_0)$, $r_0(x)$ é dada pela Equação 1-8.

$$r_0(x) = y(x_0) + (x - x_0)y'(x_0) \quad (1-8)$$

O raciocínio é repetido com (x_1, y_1) e $y_2 = y_1 + h*f(x_1, y_1)$ e assim, sucessivamente. Assim, método de Euler fornece a Equação 1-9.

$$y_{k+1} = y_k + h.f(x_k, y_k) \quad (1-9)$$

1.2.6.2 Método de Runge – Kutta

Os diferentes métodos nesta categoria envolvem maior ou menor esforço computacional e paralelamente têm maior ou menor precisão. Os métodos de Runge – Kutta possuem três propriedades distintas:

São métodos de um estágio: Para determinar y_{m+1} , precisa-se somente da informação disponível no ponto precedente x_m , y_m .

Eles concordam com a série de Taylor até os termos em h^p , onde p é diferente para métodos diferentes e é chamado a ordem do método.

Eles não exigem o cálculo de quaisquer derivadas de $f(x,y)$, mas somente da própria função.

Um membro da família de Runge-Kutta é tão comumente usado que é muitas vezes referido como RK4, Runge-Kutta de 4ª ordem, ou simplesmente como o método Runge-Kutta.

Um PVI é especificado pela Equação 1-10.

$$y' = f(t, y), y(t_0) = y_0 \quad (1-10)$$

Então, o método RK4 para este problema é dado pela Equação 1-10, 1-11 e 1-12

$$y_{n+1} = y_n + \frac{1}{6}h(k_1 + 2k_2 + 2k_3 + k_4) \quad (1-11)$$

$$t_{n+1} = t_n + h \quad (1-12)$$

Onde y_{n+1} é a aproximação RK4 de $y(t_{n+1})$, e

$$k_1 = f(t_n, y_n) \quad (1-13)$$

$$k_2 = f\left(t_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_1\right) \quad (1-14)$$

$$k_3 = f\left(t_n + \frac{1}{2}h, y_n + \frac{1}{2}hk_2\right) \quad (1-15)$$

$$k_4 = f(t_n + h, y_n + hk_3) \quad (1-16)$$

Assim, o valor seguinte (y_{n+1}) é determinado pelo valor presente (y_n), mais o produto do tamanho do h (intervalo) e uma inclinação estimada. A inclinação é uma média ponderada de inclinações:

k_1 é a inclinação no início do intervalo;

k_2 é a inclinação no ponto médio do intervalo, com inclinação k_1 para determinar o valor de y no ponto $t + h / 2$ e utilizando o método de Euler;

k_3 é novamente a inclinação no ponto médio, mas agora usando a inclinação k_2 para determinar o valor de y ; k_4 é a inclinação no final do intervalo, com o seu valor y determinado usando k_3 .

Na média das quatro inclinações, maior peso é dado para as inclinações no ponto médio, segundo a Equação 1-17.

$$slope = \frac{1}{6}(k_1 + 2k_2 + 2k_3 + k_4) \quad (1-17)$$

O método RK4 é um método de quarta ordem, significando que o erro por passo é da ordem de h^5 , enquanto o erro total acumulado tem ordem h^4 . É importante se observar que as Equações 1-13 a 1-16 são válidas para escalares e funções vetoriais (isto é, y pode ser um vetor e f um operador).

1.3 CLASSIFICAÇÃO DOS MODELOS USADOS EM MICROBIOLOGIA PREDITIVA

Segundo Gibbs et al (2001), os modelos preditivos são classificados como primário, secundário e terciário de acordo com os seguintes critérios:

Primário – Mede a resposta do micro-organismo com o tempo para um conjunto único de condições. Cada curva, população *versus* tempo pode ser descrita por um conjunto de valores específicos dos parâmetros nos modelos.

Secundário – Descreve a resposta dos parâmetros do modelo primário para variações das condições ambientais. Podem ser classificados como diretos, que descrevem diretamente o efeito das variações nos parâmetros do modelo primário, ou indiretos, que relaciona uma variável ambiental de um ou mais parâmetros do modelo primário.

Terciário – O nível terciário é constituído por aplicativos com interface amigável utilizados para resolver os modelos de nível primário e secundário.

Este nível inclui algoritmos para calcular mudanças nas condições, comparar comportamento microbiano sob diferentes condições, ou graficar o crescimento de vários micro-organismos, simultaneamente.

1.3.1 Modelos Primários

Os modelos primários indicam a variação do número de micro-organismos ou qualquer outra resposta microbiana com o tempo, num ambiente específico. Estes modelos podem avaliar a quantidade de unidades formadoras de colônias por mililitro (UFC/mL), a formação de toxinas, os níveis de substratos e produtos metabólicos.

Um modelo primário pode descrever matematicamente a curva de crescimentos gerada pelos microrganismos, sob condições ambientais definidas, com o intuito de estimar os parâmetros cinéticos que caracterizam esta curva. São eles, tempo de latência (λ (h)), velocidade específica máxima de crescimento (μ (h^{-1})) e densidade máxima da população (A). (CAYRÉ et al., 2007)

Os parâmetros de crescimento citados são apresentados Figura 1-2.

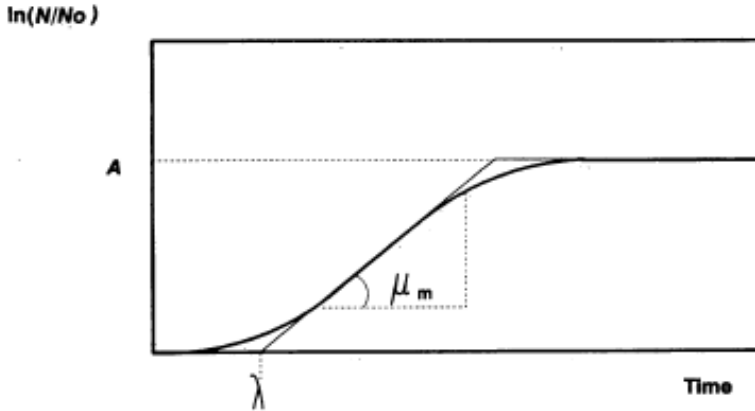


Figura 1-2 - Curva de crescimento microbiano com indicação dos parâmetros biológicos (Fonte: Zwietering, 1990).

Onde:

μ_m = velocidade de crescimento exponencial média

λ = duração da fase lag

A = log da densidade máxima da população

Baty e Delignette-Müller (2004) classificaram os modelos primários como modelos sigmoidais, entre os quais se encontram as versões modificadas dos modelos Logístico e de Gompertz; modelos com uma função de ajuste, como os propostos por Baranyi e Roberts (1994) e modelos de compartimentos como o apresentado por Hills e Wrigth (1994) (CAYRE et al., 2007). Segundo Buchanan, 1997, os dois modelos matemáticos mais amplamente utilizados são a Equação de Gompertz e o Modelo de Baranyi. A primeira é uma relação empírica sigmoideal e o segundo uma equação diferencial baseada em parte no conceito de que a velocidade de crescimento é controlada pela velocidade que é o gargalo da reação bioquímica. Os principais modelos primários são Modelo de Baranyi, Modelo de Gompertz, Logístico e Logístico Modificado.

1.3.1.1 Equação de Baranyi

Segundo Baranyi (2004) um modelo matemático é um conjunto de considerações, possivelmente não explícitas, algumas das quais são formuladas por equações não

diferenciais. Em biologia, um sistema real é extremamente complexo, então o modelo deve inevitavelmente incluir simplificações.

Para propostas práticas, algumas simplificações foram concebidas de acordo com as considerações biológicas:

A limitação de nutrientes, em alimentos, geralmente ocorre apenas em altas concentrações celulares. Considerando o nutriente infinito, pode-se considerar crescimento ilimitado.

Por razões de simplicidade e porque, em microbiologia de alimentos, modelar o efeito da limitação de nutrientes é menos significativo, $u(t)$, que descreve a transição para a fase estacionária, é aproximada para uma função de simples limitação dependendo de um parâmetro de máxima densidade celular.

$$u(t) \approx 1 - \left(\frac{x(t)}{x_{\max}} \right)^m \quad (1-18)$$

Onde x_{\max} é a máxima concentração celular e m é um parâmetro de curvatura que caracteriza a transição da curva de crescimento para a fase estacionária. (BARANYI & ROBERTS, 1995)

O modelo de Baranyi é representado pela Equação 1-19 (BARANYI & ROBERTS, 1994).

$$\ln\left(\frac{x}{x_o}\right) = D + B t + \frac{\ln(e^{-n B t} + e^{-C} - e^{-n B t - C})}{B}$$

$$\frac{\ln\left(1 + \frac{e^{m B t + \frac{\ln(e^{-n B t} + e^{-C} - e^{-n B t - C})}{B}} - 1}{e^{m(A-D)}}\right)}{m}$$

(1-19)

onde A, B, C e D são constantes matemáticas que rearranjadas possuem significado físico:

$$A = y_{\max} \quad (1-20)$$

$$B = \mu_{\max} \quad (1-21)$$

$$C = h_o = \lambda \cdot \mu_{\max} \quad (1-22)$$

$$D = v \quad (1-23)$$

$$n = \frac{\mu_{\max}}{v} \quad (1-24)$$

1.3.1.2 Equação de Gompertz

O modelo de Gompertz tem sido utilizado com sucesso para descrever o comportamento microbiano em diferentes alimentos. Este modelo foi introduzido na microbiologia de alimentos com intuito de realizar a parametrização de uma curva de crescimento de *C. botulinum* (GIBSON et al., 1987).

Zwietering, et. al. (1990) propuseram a reparametrização da função de Gompertz para obter a representação direta dos parâmetros de interesse: a velocidade específica máxima de crescimento e o tempo de duração da fase lag, resultando no Modelo de Gompertz Modificado. O Modelo de Gompertz modificado apresenta os parâmetros microbiológicos incluídos na equação, não sendo necessárias equações auxiliares (Equação 1-18) para realizar os cálculos dos parâmetros, porém há mais estabilidade numérica da otimização dos parâmetros quando utilizados o modelos tradicional de Gompertz. A Equação 1-25 apresenta o modelo de Gompertz.

$$\ln\left(\frac{x}{x_o}\right) = A e^{-e^{-B(t-M)}} \quad (1-25)$$

onde $\ln(x/x_o)$ é o logaritmo neperiano da densidade celular no tempo t , sendo que x é a absorbância final e x_o é a absorbância inicial, e a absorbância é proporcional a concentração celular, A é o aumento logaritmo da população, B é a velocidade relativa máxima de crescimento no tempo M [h^{-1}] e M é o tempo necessário para atingir a velocidade máxima de crescimento [h].

A , B e M são constantes matemáticas que rearranjadas possuem significado físico:

$$\mu = \frac{B.A}{e} \quad (1-26)$$

$$\lambda = M \left(\frac{1}{B} \right) \quad (1-27)$$

onde:

μ = velocidade específica máxima de crescimento [h^{-1}];

λ = duração da fase lag [h];

e = 2,7182

1.3.1.3 Equação Logística

As curvas de crescimento são geralmente bem descritas pelo modelo Logístico. A Equação 1-28 é utilizada para descrever o modelo Logístico (ERKMEN, 2000).

$$\ln \left(\frac{x}{x_o} \right) = \frac{A}{1 + e^{-B(t-M)}} \quad (1-28)$$

onde:

$\ln (X/X_o)$ e A possuem o mesmo significado que os apresentados no modelo de Gompertz;

B é um parâmetro adimensional;

M é o crescimento relativo na metade do tempo da fase exponencial (h^{-1}).

Estes parâmetros estão relacionados com os parâmetros microbiológicos de crescimento μ e λ .

$$\mu = \frac{A.B}{4} \quad (1-29)$$

$$\lambda = \frac{(M-2)}{B} \quad (1-30)$$

1.3.1.4 Equação Logística Modificado

Uma nova versão do modelo Logístico foi proposta por Corradini e Peleg (2005), como modelo Logístico Modificado Equação 1-31.

$$\ln\left(\frac{x}{x_0}\right) = \frac{A}{1 + e^{-B(t-M)}} - \frac{A}{1 + e^{BM}} \quad (1-31)$$

Onde o segundo termo no lado direito da equação foi introduzido para considerar, por definição, $Y(0) = \ln\left(\frac{x}{x_0}\right) = 0$, para normalizar que o início da curva de crescimento é a partir do ponto de origem (0,0). Pela Equação 1-29 e Equação 1-30 obtêm-se os parâmetros de crescimento μ e λ .

1.3.2 Modelos Secundários

Modelos que descrevem o crescimento isotérmico e padrões de inativação são considerados primários, pois são derivados diretamente de dados experimentais. Os modelos que descrevem a variação de seus parâmetros com a variação de uma condição ambiental, são conhecidos como Modelos Secundários.

O nível secundário envolve equações que descrevem como as respostas dos modelos primários (duração da fase lag, velocidade de crescimento e densidade máxima de população) mudam com alterações nos fatores ambientais. Quando a temperatura for o fator primário de interesse, como é frequentemente o caso, estas equações podem ser baseadas nas equações de Arrhenius ou de Bêlerádek (modelo da raiz quadrada) (McMEEKIN et al., 1993, SKINNER et al., 1994 apud NAKASHIMA et. al., 2000).

Normalmente, os modelos secundários são modelos mais simples que os modelos primários onde as funções são: lineares, exponenciais, potenciais ou polinomiais, com isso reduzindo também o número de parâmetros a serem otimizados.

Além dos modelos da Raiz Quadrada e de Arrhenius, bastante utilizados na literatura, a influência da variação de fatores ambientais sobre os parâmetros primários pode ser descrita através de equações exponenciais, polinomiais, lineares,

potência, conforme o melhor ajuste aos dados propostos. Corradini e Peleg (2005) demonstraram que modelos empíricos podem ser utilizados para descrever a variação dos parâmetros dos modelos primários com a temperatura e outros fatores como atividade de água, pH, concentração de sal, entre outros.

Os principais modelos secundários apresentados na literatura são os modelos de Arrhenius, o Modelo da Raiz Quadrada e o Modelo de Weibull (equação da potência).

1.3.2.1 Equação do tipo Arrhenius

A equação de Arrhenius assume que a velocidade de crescimento é controlada pela velocidade limite de uma única reação enzimática (WHITING, BUCHANAN, 1997). É amplamente utilizada para descrever a variação da taxa de reação química com temperatura e, em microbiologia preditiva, é adaptada para descrever o mesmo comportamento, porém com parâmetros microbiológicos. É representada pela Equação 1-32.

$$k = a \cdot \exp\left(\frac{-Ea}{RT}\right) \quad (1-32)$$

Onde:

k é o parâmetro de crescimento do modelo (μ , λ ou A);

a é um parâmetro obtido pelo ajuste do modelo (h^{-1});

T é a temperatura absoluta (K);

Ea é a energia de ativação para crescimento bacteriano ($kJ \cdot mol^{-1}$);

R a constante universal dos gases ($8,314 J \cdot K^{-1} \cdot mol^{-1}$).

Em muitos casos, a clássica equação de Arrhenius é inapropriada para descrever o efeito da temperatura nas velocidades de crescimento dos microrganismos porque a própria energia de ativação (aparente) é dependente da temperatura (GIL et al., 2006). Quando se utiliza a equação do “tipo” Arrhenius (Equação 1-32) os valores dos parâmetros não ficam restritos às constantes definidas pelo modelo inicial.

$$k = a \exp\left(b \left(\frac{1}{T} - \frac{1}{T_{ref}}\right)\right) \quad (1-33)$$

onde:

k = parâmetro de crescimento de interesse

a e b são parâmetros obtidos pelo ajuste do modelo, na T_{ref} ;

T = temperatura (°C).

T_{ref} = temperatura de referência °C.

1.3.2.2 Modelo da Raiz Quadrada – Modelo de Bélerádek

O modelo da Raiz Quadrada é um modelo empírico que descreve a dependência do parâmetro de crescimento bacteriano com a temperatura. Este modelo foi inicialmente utilizado para modelar o efeito da temperatura na velocidade específica máxima de crescimento (WHITING, BUCHANAN, 1997).

O modelo de Bélerádek está baseado na relação linear que existe entre a raiz quadrada da velocidade de crescimento abaixo da velocidade de crescimento ótima e a temperatura. Porém, o modelo também pode ser extrapolado para os demais parâmetros microbiológicos, como A e λ . Esse modelo ainda assume que os fatores ambientais são independentes, ou seja, que não existe interação entre os fatores (WHITING, BUCHANAN, 1997).

$$\sqrt{k} = b(T - T_{\min}) \quad (1-34)$$

onde:

k é o parâmetro de crescimento do modelo (μ , λ ou A);

b representa o coeficiente de regressão;

T_{\min} é a temperatura mínima teórica para o crescimento ou é um parâmetro do modelo;

T é a temperatura.

1.3.3 Modelo da Weibull (equação da potência)

O modelo de Weibull apresenta a Equação 1-35 (PELEG e COLE, 1998)

$$k = b.T^n \quad (1-35)$$

onde:

k é um parâmetro de crescimento do modelo (μ , λ e A);

b é o parâmetro de escala;

n é o parâmetro de forma, ambos obtidos pelo ajuste do modelo aos dados experimentais. Quando $n < 1$ a concavidade da curva é para cima, $n > 1$ concavidade da curva para baixo e se $n = 1$ é uma equação linear.

1.3.4 Modelos não Isotérmicos - Modelos Dinâmicos

Muitos modelos são desenvolvidos e validados sob condições de temperatura constante. Na prática, entretanto, as variações de temperatura ocorrem frequentemente, especialmente durante a estocagem e distribuição de alimentos. Assim, a validação sob mudanças de temperatura (condições dinâmicas) é de grande importância para avaliar o desempenho do modelo na previsão de vida útil sob condições reais da cadeia de frio (KOUTSOUMANIS et. al, 2006).

Ao contrário do pH e da atividade de água, a temperatura pode variar amplamente em todo o processo de produção e cadeia de distribuição. Compreende-se que uma abordagem mais geral de modelagem é necessária, considerando a transferência de calor, carga microbiana e até mesmo variáveis de outro processo (como por exemplo, a qualidade sensorial), sendo que estes fatores são combinados e interligados (VAN IMPE et.al, 1992).

Para que os modelos possam ser aplicáveis a alimentos armazenados em condições reais, é necessário considerar, dentro do modelo, o efeito das mudanças das variáveis externas sobre o crescimento microbiano a fim de obter predições mais precisas em relação à inocuidade e à vida útil dos mesmos (CAYRÉ et al., 2004).

Alguns modelos dinâmicos são propostos na literatura entre eles o modelo de Corradini e Peleg (2005) e o modelo de Van Impe et al. (1992)

1.3.4.1 Modelo de Corradini e Peleg (2005)

Foi demonstrado por Corradini e Peleg, 2005 que, enquanto os modelos empíricos primários e secundários descrevem os padrões de crescimento isotérmico, expressões matemáticas derivadas destes podem ser utilizadas para prever os padrões de crescimento em condições não-isotérmicas.

Assume-se que, em condições não isotérmicas, a velocidade de crescimento no instante t é a velocidade de

crescimento isotérmico referente à temperatura do instante t e o tempo t corresponde ao tamanho da população momentânea. Mais uma vez, a velocidade de crescimento momentâneo em condições não isotérmicas depende não só da mudança de temperatura (como o modelo de Arrhenius tradicional, por exemplo), mas também da mudança do estado fisiológico da população em crescimento (PELEG, 2006).

O modelo proposto por Corradini e Peleg (2005) pode ser utilizado a partir de dois modelos primários sendo o primeiro uma equação empírica simples e a segunda o modelo Logístico modificado. Uma nova versão do modelo Logístico foi proposta, como modelo Logístico Modificado, que está representado pela Equação 1-31 (modelos primários). A derivada da Equação 1-31 está representada na Equação 1-36.

$$\frac{dy(t)}{dt} \Big|_{(T_{cte})} = \frac{k(T)a(T)e[k(T)(t_c(T)-t)]}{[1 + e(k(T)(t_c(T)-t))]^2} \quad (1-36)$$

onde os parâmetros do modelo são representados por:

a = população máxima atingida;

k = é relacionado ao crescimento na fase exponencial (h^{-1});

t_c = ponto de inflexão na curva (h).

Para aplicação deste modelo dinâmico é necessário que se conheça como os parâmetros do modelo primário variam com a temperatura, ou seja, o modelo secundário para $a(T)$, $k(T)$ e $t_c(T)$. Quando há variações da temperatura com o tempo, é necessário que se conheça os modelos secundários para cada um dos parâmetros do modelo primário. Ou seja, quando a temperatura varia, ou seja, $T = T(t)$, segue-se que:

$$a(T) = a[T(t)],$$

$$k(T) = k[T(t)] \text{ e}$$

$$t_c(T) = t_c[T(t)].$$

Com a inclusão dos modelos secundários na derivada da equação do modelo de Logístico modificado, Equação 1-31, pode-se obter a Equação 1-37 que representa o modelo não isotérmico.

$$\frac{dy(t)}{dt} = \frac{k[T(t)]a'[T(t)]\exp(k[T(t)]\{t_c[T(t)]-t^*\})}{[1 + \exp(k[T(t)]\{t_c[T(t)]-t^*\})]^2} \quad (1-37)$$

onde:

$$t^* = \frac{1}{k(T)} \log e \left[\frac{\exp[k(T)t_c(T)](a'(T) + y(T)\{1 + \exp[k(T)t_c(T)]\})}{a'(T)\exp[k(T)t_c(T)] - y(t)\{1 + \exp[k(T)t_c(T)]\}} \right] \quad (1-38)$$

Apesar da aparência pesada do modelo, ele é uma equação diferencial ordinária, que pode ser facilmente resolvida numericamente por *softwares* comerciais. Sua solução, a curva de crescimento procurado $Y(t)$ vs t , pode ser calculada para quase qualquer história térmica, desde que $k(T)$, $T_c(T)$ e $T(t)$ possam ser expressados algebricamente, dentro da faixa de temperatura estudada.

A partir da equação do modelo não isotérmico, podem-se definir diferentes perfis de temperatura, obtendo-se a predição das curvas de crescimento nas condições não isotérmicas.

O modelo de Corradini e Peleg (2005) foi aplicado aos dados de crescimento isotérmicos e não isotérmicos de *Pseudomonas* spp em peixe refrigerado armazenado de 0°C a 8°C (Koutsoumanis (2001)) e *Escherichia coli* em caldo nutriente de 27,6°C a 36°C (Fujikawa et al (2004)). Demonstrou-se que o procedimento pode ser usado para gerar as curvas de crescimento sob histórias térmicas complicadas que podem incluir oscilações de temperaturas regulares e irregulares, visto que as curvas de crescimento preditas estavam coerentes com os dados experimentais obtidos.

1.3.4.2 Modelo de Van Impe

Van Impe et al (1995) também apresenta um modelo não isotérmico que descreve o crescimento microbiano onde a possível influência de pH e atividade de água não são consideradas.

A construção do modelo dinâmico proposto por Van Impe et al. (1995) ocorre em duas etapas. Primeiro é desenvolvido um modelo dinâmico somente para o crescimento microbiano, e a

segunda etapa, inclui a inativação e a possível transição do crescimento até a inativação. O modelo não isotérmico parte da derivada (Equação 1-39) de um modelo explícito, neste caso o modelo de Gompertz que está representada pela Equação 1-25.

$$\frac{dy}{dt} = a(\exp(-\exp(b-ct)))(-\exp(b-ct))(-c) \quad (1-39)$$

Substituindo a Equação 1-25 na Equação 1-39 e rearranjando matematicamente os termos se obtém a Equação 1-40.

$$\frac{dy}{dt} = M \cdot y \cdot \ln\left(\frac{A}{y}\right) \quad (1-40)$$

onde M = parâmetro matemático.

Os arranjos matemáticos necessários para suprir a ausência do parâmetro matemático B baseiam-se na solução numérica proposta para a condição inicial, sendo o valor de $y = 0$ como o limite inicial da integração, obtém:

$$n = \ln(N)$$

$$y = n - \ln(N_0)$$

$$\frac{dy}{dt} = \frac{dn}{dt}$$

Como $n_0 = \ln(N_0)$, o modelo dinâmico para n é representado pela Equação 1-41.

$$\frac{dn}{dt} = M \cdot (n - n_0) \cdot \ln\left(\frac{A}{n - n_0}\right) \quad (1-41)$$

onde, $n - n_o = y$, e sabendo que como condição inicial $y(0) = 0$, podemos substituir estes valores na Equação 1-41 e obter como condição inicial da equação diferencial ordinária a Equação 1-42.

$$n(t = 0) = n_o + A.e^{-e^B} \quad (1-42)$$

Desta maneira e considerando as equações para a obtenção dos parâmetros de crescimento A , μ e λ (apresentadas no modelo de Gompertz), juntamente com as equações dos modelos secundários (equações da Raiz Quadrada e da equação do tipo Arrhenius), a Equação 1-41 permite prever o comportamento da população microbiana sob as condições de temperatura variáveis.

A validação do modelo de Van Impe foi realizada através de dados experimentais de *Brochothrix thermosphacta* e *Lactobacillus plantarum*, obtidos sob condições de variação de temperatura.

Segundo Nicolai et al. (1995), para *Brochothrix thermosphacta* os dados experimentais de crescimento foram coletados sob condições isotérmicas e não isotérmicas, já as curvas de sobrevivência foram obtidas para temperaturas constantes. Além disso, diferentes valores de pH e concentração de sal foram considerados. Constatou-se excelente acordo entre o modelo dinâmico e os dados experimentais sob as diferentes condições estudadas.

1.3.5 Modelos Terciários

Os modelos terciários são ferramentas desenvolvidas em que rotinas dos *softwares* computacionais transformam os modelos primários e secundários em interfaces amigáveis que permitem ao usuário determinar curvas de crescimento microbiano (WHITING, 1995).

Os dados gerados, através de respostas microbianas de acordo com as condições ambientais, e equações estatísticas serão aplicados com o objetivo de produzir uma interface de fácil acesso, com o intuito de comparar o efeito das diferentes condições e contrastar o comportamento de vários microorganismos.

Muitos modelos de predição foram desenvolvidos, tanto para micro-organismos deteriorantes como para patogênicos, que apresentam seus respectivos modelos de crescimento e inativação.

Estes modelos normalmente incluirão variáveis como:

Temperatura de estocagem e suas oscilações;

pH;

Atividade de água equivalente;

Teor de Sal

Alguns programas mais específicos podem trazer outras variáveis como teor de nitrito (conservantes), taxa de CO₂ (atmosfera modificadas), acidulantes entre outros.

Como vantagem, estes modelos apresentam respostas rápidas e eficientes ao avaliar o crescimento microbiano potencial em um determinado produto sob condições específicas, não se fazendo necessário o uso de recursos experimentais.

Alguns *softwares* comerciais, que podem ser adquiridos livres de encargos, fornecem predições do crescimento microbiano ou da taxa de crescimento ou da fase lag, sob condições definidas (McMEEKIN & ROSS, 2002). Entre eles estão:

- USDA's *Pathogen Modeling Program* (BUCHANAN, 1993), disponível no site <http://www.arserrc.go/mfs/pathogen.htm>;

- Food MicroModel (McCLURE et al., 1994; PANCIELLO & QUANTICK, 1998), *software* desenvolvido por um grupo de empresas em conjunto com o governo do Reino Unido, disponível no site <http://www.lfra.co.uk/micromodel/index.html>;

- *Seafood Spoilage Predictor* (DALGAARD et al., 2002), desenvolvido para prever a vida de prateleira de alimentos de origem marinha estocados à temperatura constante ou sob flutuações de temperatura. O *software* disponível no site <http://dfu.min.dk/micro/ssp/>.

- *Food Spoilage Predictor* (NEUMEYER et al., 1997), *software* comercial que modela o efeito da atividade de água e de flutuações de temperatura no crescimento de *Pseudomonas* psicrotolerantes.

- Bugdeath 1.0 (PEREIRA et al., 2005) desenvolvido numa linhagem computacional chamada de Real Basic 5.2. O *software* apresenta recursos em que o usuário pode optar pelo tipo de processo térmico (seco / úmido), propriedades do ar e

parâmetros do equipamento, regime de aquecimento, tempo total de processo, tempo de aquecimento, temperatura e tempo de equalização e espessura do produto.

- Sym Previous (LEPORQ et al., 2004). Após inserir os dados experimentais, é necessário que o usuário fixe as condições nas quais foram realizados os experimentos. Para a simulação da curva de crescimento de determinado microrganismo, valores de μ , λ e o log da população máxima devem ser especificados, bem como o micro-organismo e o meio de crescimento.

- *Food Product Modeller*, software desenvolvido pelo departamento AgResearch da MIRINZ (órgão ligado ao governo da Nova Zelândia) com função de determinação preditiva do crescimento microbiano em determinados tipos de alimento. Disponível em www.agresearch.co.nz.

1.4 O PROCESSO DE DESENVOLVIMENTO DE SOFTWARE

A engenharia de *software* preocupa-se com o desenvolvimento de aplicações úteis, utilizando uma variedade de ferramentas e técnicas que cobrem uma série de problemas. A principal ideia é construir uma cultura de desenvolvimento de *software* capaz de guiar o projeto sem surpresas, garantindo a presença de qualidade em todos os níveis, indivíduo, projeto e organização. (WIEGERS, 2006)

Utilizando-se metodologias ágeis no processo de desenvolvimento, os requisitos são elaborados em uma escala menor de projeto, sendo os resultados acompanhados através de *builds* (geração dos arquivos executáveis) frequentes com as funcionalidades planejadas, permitindo mudanças ou ajustes o mais cedo possível. (SCHWABER, 2004)

Além disso, durante todo o processo de codificação, o ideal é o desenvolvimento de testes automatizados de unidade, que garantem que cada unidade do código funcione como esperado (BECK, 2003). Outro benefício desta técnica é que quaisquer modificações, sejam por correções de *bugs* ou implementações de novas características, não vão causar novos *bugs* ou quebrar funcionalidades já existentes, pois os testes automatizados permitem ao desenvolvedor detectar qualquer problema imediatamente.

Baseando-se nas metodologias ágeis de desenvolvimento, utilizou-se como base as seguintes práticas do processo de desenvolvimento de *software*:

Comunicação: uma maior colaboração e comunicação frequente durante todas as fases do desenvolvimento entre equipe de desenvolvimento e os usuários buscam minimizar problemas relacionados ao levantamento de requisitos inicial.

Desenvolvimento iterativo e incremental: o projeto é dividido em janelas de tempo (*timebox* ou *sprint*) sobre as quais as atividades são priorizadas e agrupadas. Ao final de cada janela, o objetivo é disponibilizar um *build* funcional da aplicação de *software* para uso e teste do usuário, permitindo antecipar divergências sobre a expectativa do usuário e corrigi-las em tempo hábil. Ao final de uma janela de tempo, a situação do projeto é reavaliada com a participação ativa do usuário, permitindo redefinir requisitos, prioridades e estimativas com menor impacto para as janelas subsequentes.

1.4.1 As Incertezas de um Projeto de Software

Projetos de desenvolvimento de *software* diferem de projetos convencionais por seu elevado grau de risco. Projetos de *software* são, por sua natureza, um processo de aprendizagem e descobrimento.

Nos estágios iniciais do desenvolvimento, tanto o usuário quanto os desenvolvedores não possuem uma noção clara do sistema a ser construído. À medida que o projeto avança e os requisitos vão sendo especificados, uma visão comum começa a ser compartilhada pelas partes envolvidas, nem sempre contemplada na sua totalidade nas projeções iniciais. Por isso, mesmo após a completa especificação dos requisitos do sistema, a experiência mostra que muito ainda resta para ser descoberto durante as etapas de projeto e implementação.

Esta incerteza se reflete no projeto na forma de uma grande variação entre o esforço dimensionado no início do mesmo, e aquele estimado mais tarde, durante o decorrer do projeto. A Figura 1-3 ilustra o cone de incerteza típico de um projeto de software (MCDONNELL, 1998)

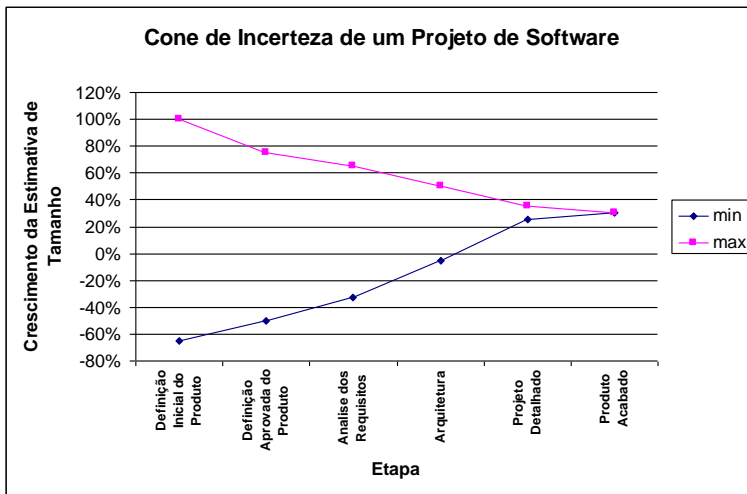


Figura 1-3 - O cone de incerteza de um projeto de software. O desvio nas estimativas de tamanho diminui à medida que o processo de desenvolvimento avança (Fonte: McConell - Software Project Survival Guide)

1.4.2 Modelos Terciários de Grande Utilização no Meio Acadêmico

1.4.2.1 DmFit

O *Institut of Food Research* (IFR) dos Estados Unidos desenvolveu um protocolo padrão de armazenamento de dados de microbiologia no início de 1990. O protocolo foi inicialmente usado para permitir que os dados fossem acessados de qualquer laboratório e pudessem ser processados pelos programas de modelação do IFR, DMFit e MicroFit.

DMFit Excel é uma ferramenta, utilizada para se adequar curvas onde uma fase linear é precedida e seguida por uma fase estacionária. A principal diferença entre este modelo e outras curvas sigmoidais como Gompertz e Logístico é que a fase intermediária é realmente muito próxima da linear, ao contrário das curvas clássicas sigmoidais, que têm uma curvatura acentuada. (DMFit Manual)

DMFit é parte de um sistema utilizado no Institute of Food Research (IFR) para modelar a concentração celular logarítmica de lotes de culturas bacteriana pelo tempo. (DMFit Manual)

1.4.2.2 *GlnaFit*

Outra ferramenta utilizada é o GlnaFit (Geeraerd and Van Impe Inactivation Model Fitting Tool), uma ferramenta gratuita, *Add-in* para Microsoft® Excel (Figura 1-4), que tem como objetivo de eliminar a lacuna hoje existente entre os desenvolvedores de modelos de predição e os usuários finais na indústria de alimentos que não estão familiarizados com ferramentas avançadas de regressão não linear. (GEERAERD, 2005). A ferramenta é útil para testar nove diferentes tipos de modelos de inativação microbiana, que são:

- Regressão Log-Linear;
- Modelo Log-Linear + Ombro;
- Modelo Log-Linear + Cauda;
- Modelo Log-Linear + Ombro + Cauda;
- Modelo de Weibull;
- Modelo de Weibull corrigindo parâmetro p
- Modelo de Weibull + Cauda;
- Modelo Bifásico;
- Modelo Bifásico + Ombro

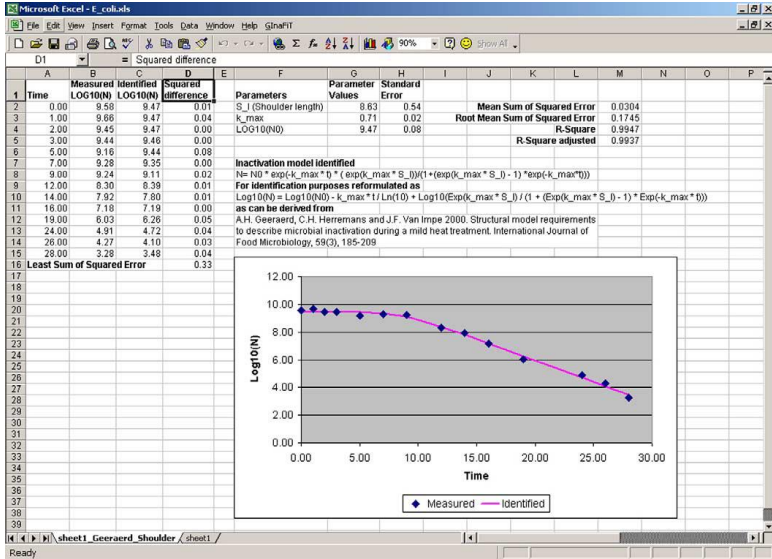


Figura 1-4 – Screenshot do GinaFit

2. MATERIAL E MÉTODOS

Neste capítulo, serão apresentadas as descrições das metodologias que foram utilizados para realização deste trabalho.

2.1 ESTIMATIVA DOS PARÂMETROS DOS MODELOS PRIMÁRIOS E SECUNDÁRIOS

A solução numérica dos modelos primários e secundários é um problema de mínimos quadrados. Os métodos utilizados para solucionar este problema utilizam o princípio de minimização da função através de derivadas da função em relação a cada parâmetro a ser otimizado.

2.1.1 Modelos Primários

Os modelos primários inseridos no aplicativo foram os modelos de Baranyi, de Gompertz, Logístico e Logístico Modificado (BARANYI & ROBERTS, 1994; ZWIETERING, 1990; ERKMEN, 2000; CORRADINI & PELEG, 2005).

2.1.1.1 Modelo de Baranyi

O modelo de Baranyi (BARANYI & ROBERTS, 1994) é descrito pela Equação 1-19. Por simplificação matemática com o intuito de diminuir os termos das equações a Equação 1-19 foi reorganizada considerando-se a Equação 2-1:

$$f = \ln\left(\frac{x}{x_o}\right) \quad (2-1)$$

Otimização dos parâmetros pelo método das derivadas

A minimização do resíduo ocorre através do método das derivadas. Para o uso deste método, é necessário se determinar a derivada da função f em relação a cada parâmetro a ser otimizado.

A derivada da função f em relação ao parâmetro A é apresentada na Equação 2-2.

$$\frac{df}{dA} = \frac{e^{mBt + \frac{\ln(e^{-nBt} + e^{-C} - e^{-nBt-C})}{B}} - 1}{e^{m(A-D)} \left(1 + \frac{e^{mBt + \frac{\ln(e^{-nBt} + e^{-C} - e^{-nBt-C})}{B}} - 1}{e^{m(A-D)}} \right)}$$

(2-2)

A derivada da função f em relação ao parâmetro B é apresentada na Equação 2-3.

$$\begin{aligned} \frac{df}{dB} = & t - \frac{\ln(e^{-nBt} + e^{-C} - e^{-nBt-C})}{B^2} \\ & + \frac{-nt e^{-nBt} + nte^{-nBt-C}}{B(e^{-nBt} + e^{-C} - e^{-nBt-C})} - \left(\left(mt \right. \right. \\ & \left. \left. - \frac{\ln(e^{-nBt} + e^{-C} - e^{-nBt-C})}{B^2} \right. \right. \\ & \left. \left. + \frac{-nt e^{-nBt} + nte^{-nBt-C}}{B(e^{-nBt} + e^{-C} - e^{-nBt-C})} \right) \right) \\ & e^{mBt + \frac{\ln(e^{-nBt} + e^{-C} - e^{-nBt-C})}{B}} \left/ \left(m e^{m(A-D)} \left(1 \right. \right. \right. \\ & \left. \left. + \frac{e^{mBt + \frac{\ln(e^{-nBt} + e^{-C} - e^{-nBt-C})}{B}} - 1}{e^{m(A-D)}} \right) \right) \end{aligned}$$

(2-3)

A derivada da função f em relação ao parâmetro C é apresentada na Equação 2-4.

$$\frac{df}{dC} = \frac{-e^{-C} + e^{-nBt-C}}{B(e^{-nBt} + e^{-C} - e^{-nBt-C})} - \left((-e^{-C} + e^{-nBt-C}) e^{mBt + \frac{\ln(e^{-nBt} + e^{-C} - e^{-nBt-C})}{B}} \right) / \left(mB(e^{-nBt} + e^{-C} - e^{-nBt-C}) e^{m(A-D)} \left(1 + \frac{e^{mBt + \frac{\ln(e^{-nBt} + e^{-C} - e^{-nBt-C})}{B}}}{e^{m(A-D)}} - 1 \right) \right)$$

(2-4)

A derivada da função f em relação ao parâmetro D é dada pela Equação 2-5.

$$\frac{df}{dD} = 1 - \frac{e^{mBt + \frac{\ln(e^{-nBt} + e^{-C} - e^{-nBt-C})}{B}} - 1}{e^{m(A-D)} \left(1 + \frac{e^{mBt + \frac{\ln(e^{-nBt} + e^{-C} - e^{-nBt-C})}{B}}}{e^{m(A-D)}} - 1 \right)}$$

(2-5)

2.1.1.2 Modelo de Gompertz

O modelo de Gompertz (ZWIETERING, 1990) é descrito pela Equação 1-25. Por simplificação matemática com o intuito de diminuir os termos das equações a Equação 1-25 foi reorganizada considerando-se a Equação 2-1

Otimização dos parâmetros pelo método das derivadas

A minimização do resíduo ocorre através de métodos de derivada. Para o uso destes métodos, é necessário se determinar a derivada da função f em relação a cada parâmetro a ser otimizado. A derivada da função f em relação ao parâmetro A , para o modelo de Gompertz, é apresentada na Equação 2-6.

$$\frac{df}{dA} = e^{-e^{-B(t-M)}} \quad (2-6)$$

A derivada da função f em relação ao parâmetro B , para o modelo de Gompertz, é apresentada na Equação 2-7.

$$\frac{df}{dB} = -A(-t+M)e^{-B(t-M)}e^{-e^{-B(t-M)}} \quad (2-7)$$

A derivada da função f em relação ao parâmetro M é apresentada na Equação 2-8.

$$\frac{df}{dM} = -A B e^{-B(t-M)}e^{-e^{-B(t-M)}} \quad (2-8)$$

2.1.1.3 Modelo Logístico

O modelo Logístico (ERKMEN, 2000) é descrito pela Equação 1-28. Por simplificação matemática com o intuito de diminuir os termos das equações a Equação 1-28 foi reorganizada considerando-se a Equação 2-1

Otimização dos parâmetros pelo método das derivadas

A minimização do resíduo ocorre através de métodos de derivada. Para o uso destes métodos, é necessário se determinar a derivada da função f em relação a cada parâmetro a ser otimizado. A derivada da função f em relação ao parâmetro A , para o modelo de Logístico, é apresentada na Equação 2-9.

$$\frac{df}{dA} = \frac{1}{1 + e^{-B(t-M)}} \quad (2-9)$$

A derivada da função f em relação ao parâmetro B , para o modelo de Logístico, é apresentada na equação 2-10.

$$\frac{df}{dB} = - \frac{A(-t+M)e^{-B(t-M)}}{(1 + e^{-B(t-M)})^2} \quad (2-10)$$

A derivada da função f em relação ao parâmetro M é apresentada na equação 2-11.

$$\frac{df}{dM} = - \frac{ABe^{-B(t-M)}}{(1 + e^{-B(t-M)})^2} \quad (2-11)$$

2.1.1.4 Modelo Logístico Modificado

O modelo Logístico Modificado (CORRADINI E PELEG, 2005) é descrito pela Equação 1-31. Por simplificação matemática com o intuito de diminuir os termos das equações a Equação 1-31 foi reorganizada considerando-se a Equação 2-1

Otimização dos parâmetros pelo método das derivadas

A minimização do resíduo ocorre através de métodos de derivada. Para o uso destes métodos, é necessário se determinar a derivada da função f em relação a cada parâmetro a ser otimizado. A derivada da função f em relação ao parâmetro A , para o modelo de Logístico, Modificado é apresentada na Equação 2-12.

$$\frac{df}{dA} = \frac{1}{1 + e^{-B(t-M)}} - \frac{1}{1 + e^{BM}} \quad (2-12)$$

A derivada da função f em relação ao parâmetro B , para o modelo de Logístico, Modificado é apresentada na Equação 2-13.

$$\frac{df}{dB} = -\frac{A(-t+M)e^{-B(t-M)}}{(1+e^{-B(t-M)})^2} + \frac{AMe^{BM}}{(1+e^{BM})^2} \quad (2-13)$$

A derivada da função f em relação ao parâmetro M é apresentada na Equação 2-14.

$$\frac{df}{dM} = -\frac{ABe^{-B(t-M)}}{(1+e^{-B(t-M)})^2} + \frac{ABe^{BM}}{(1+e^{BM})^2} \quad (2-14)$$

2.1.2 Modelos Secundários

Para descrever o efeito da temperatura sobre os parâmetros dos modelos primários, foram utilizados os seguintes modelos secundários: Modelo de Arrhenius, Modelo da raiz quadrada, modelo de Weibull (equação da potência).

2.1.2.1 Modelo de Arrhenius

O modelo de Arrhenius (WHITING, BUCHANAN, 1997) é descrito pela Equação 1-32.

Otimização dos parâmetros pelo método das derivadas

Assim como foi realizado para os modelos primários, a otimização dos parâmetros dos modelos secundários foi realizada pelo método das derivadas.

A derivada da função em relação ao parâmetro A para o modelo de Arrhenius é apresentada pela Equação 2-15.

$$\frac{df}{dA} = e^{-\frac{Ea}{RT}} \quad (2-15)$$

A derivada da função em relação ao parâmetro Ea , para o modelo de Arrhenius é apresentada pela Equação 2-16.

$$\frac{df}{dEa} = - \frac{A e^{-\frac{Ea}{RT}} \ln(e)}{RT} \quad (2-16)$$

2.1.2.2 Modelo da Raiz Quadrada

O modelo da Raiz Quadrada (WHITING, BUCHANAN, 1997) é descrito pela Equação 1-34.

Otimização dos parâmetros pelo método das derivadas

Assim como foi realizado para os modelos primários, a otimização dos parâmetros do modelo da Raiz Quadrada foi realizada pelo método das derivadas.

A derivada da função em relação ao parâmetro b para o modelo da Raiz Quadrada é apresentada pela Equação 2-17.

$$\frac{df}{db} = 2 b (T - T_{\min})^2 \quad (2-17)$$

A derivada da função em relação ao parâmetro T_{\min} para o modelo da Raiz Quadrada é apresentada pela Equação 2-18.

$$\frac{df}{dT_{\min}} = -2 b^2 (T - T_{\min}) \quad (2-18)$$

2.1.2.3 Modelo de Weibull (Equação da potência)

O modelo de Weibull (PELEG e COLE, 1998) é descrito pela Equação 1-35.

Otimização dos parâmetros pelo método das derivadas

A derivada da função em relação ao parâmetro b para o modelo de Weibull é apresentada pela Equação 2-19.

$$\frac{df}{db} = T^n \quad (2-19)$$

A derivada da função em relação ao parâmetro n para o modelo de Weibull é apresentada pela Equação 2-20.

$$\frac{df}{dn} = b T^n \ln(T) \quad (2-20)$$

3.1.3 Resolução dos modelos pelo método dos mínimos quadrados

A resolução dos modelos pelo método de mínimos quadrados é feita calculando o vetor resíduo através dos dados iniciais dos pontos experimentais e com a primeira estimativa dos parâmetros a serem calculados. Através das derivadas dos modelos com relação aos parâmetros matemáticos, é possível obter a matriz Jacobiana. O vetor incremento é a solução do sistema linear proposto pelos métodos de Gauss-Newton ou Levenberg-Marquardt. A soma deste vetor com os parâmetros iniciais (estimativa inicial) é a solução para a próxima iteração, ou se o erro, que é a norma dos resíduos, estiver abaixo do mínimo proposto, a soma é a própria solução do problema.

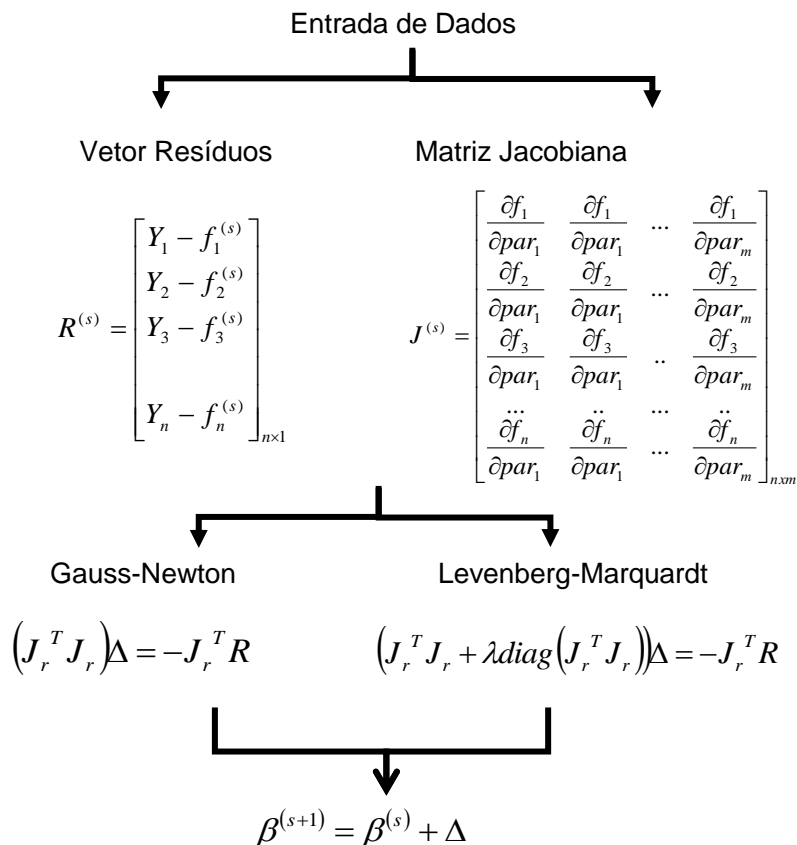


Figura 2-1: Fluxograma para resolução dos métodos de mínimos quadrados

2.1.3 Equações diferenciais ordinárias

Os modelos não isotérmicos são problemas de equações diferenciais ordinárias com condições iniciais. Para a solução dos modelos são utilizados métodos baseados na série de Taylor como é o caso de método de Euler e método de Runge-Kutta.

2.1.3.1 Modelos não isotérmicos - Modelos Dinâmicos

Foram inseridos no aplicativo dois métodos dinâmicos: o método de Corradini e Peleg (Corradini e Peleg, 2005) e o método de Van Impe (Van Impe et al, 1995)

2.1.3.2 Modelo de Corradini e Peleg (2005)

Este método utiliza o modelo logístico modificado (Equação 1-31) como modelo primário base para a modelagem do crescimento não isotérmico. A Equação 2-21 apresenta a equação diferencial em relação ao tempo da Equação 1-31.

$$\frac{dy(t)}{dt} = \frac{B(T).A(T).e^{(B(T).(M(T)-t))}}{\left[1 + e^{(B(T).(M(T)-t))}\right]^2} \tag{2-21}$$

Onde:

$$t^* = \frac{1}{B(T(t))} \ln \left[\frac{e^{[B(T(t))M(T(t))]} \left(A'(T(t)) + y \cdot \left(1 + e^{[k(T(t))t_c(T(t))]} \right) \right)}{A'(T(t))e^{[B(T(t))M(T(t))]} - y \cdot \left(1 + e^{[B(T(t))M(T(t))]} \right)} \right] \tag{2-22}$$

que foi obtido isolando-se t na Equação 1-31 e considerando-se condição de temperatura variável.

Método de Euler

A solução da equação diferencial ordinária (EDO) pode ser realizada pelo método de Euler, onde se inicia pelo truncamento da série de Taylor (Equação 2-23) no primeiro termo.

$$y = y^0 + \Delta t \left(\frac{dy}{dt} \right) \rightarrow \left(\frac{dy}{dt} \right) = \frac{y - y^0}{\Delta t} \tag{2-23}$$

Substituindo a Equação 2-23 na equação do modelo de Corradini e Peleg (Equação 2-21) se obtém equação discretizada no tempo (Equação 2-24)

$$\frac{y - y^0}{\Delta t} = \frac{B(T(t)).A(T(t)).e^{(B(T(t)).(M(T)-n\Delta t))}}{\left[1 + e^{(B(T(t)).(M(T(t))-t^*))}\right]^2} \quad (2-24)$$

rearranjando os termos da Equação 2-25 obtemos:

$$y = y^0 + \Delta t \frac{B(T(t)).A(T(t)).e^{(B(T(t)).(M(T)-n\Delta t))}}{\left[1 + e^{(B(T(t)).(M(T(t))-t^*))}\right]^2} \quad (2-25)$$

onde,

$$t^* = \frac{1}{B(T(t))} \ln \left[\frac{e^{[B(T(t))M(T(t))]} \left(A'(T(t)) + y^0 \cdot (1 + e^{[k(T(t))t_c(T(t))]} \right) \right)}{A'(T(t))e^{[B(T(t))M(T(t))]} - y^0 \cdot (1 + e^{[B(T(t))M(T(t))]} \right)} \right] \quad (2-26)$$

Condição Inicial

$$y(t = 0) = \ln \left(\frac{x_o}{x_o} \right) = \ln(1) = 0$$

Método de Runge-Kutta

A solução da EDO pelo método de Runge-Kutta é realizada através da discretização da equação do modelo de Corradini e Peleg (Equação 2-21) como mostra a Equação 2-27.

$$f(y, t) = \frac{B(T(t)).A(T(t)).e^{(B(T(t)).(M(T)-t))}}{\left[1 + e^{(B(T(t)).(M(T(t))-t^*))}\right]^2} \quad (2-27)$$

Após a discretização da equação a aplicação do método é feita em etapas usando os coeficientes de interpolação. Estes coeficientes variam em número e valor dependendo da ordem de interpolação usado no método. O método de Runge-Kutta de 4^a ordem possui quatro coeficientes: k_1 , k_2 , k_3 e k_4 .

Método de 4ª ordem:

$$y^{n+1} = y^n + \frac{\Delta t}{6} (k_1 + 2k_2 + 2k_3 + k_4)$$

$$t^{n+1} = t^n + \Delta t$$

Onde, y^{n+1} é a aproximação por RK4 de $y(t_{n+1})$. Os coeficientes são calculados a partir das Equações 1-13, 1-14, 1-15 e 1-16. Onde a variável f é a função da Equação 3-27.

2.1.3.3 Modelo de Van Impe

O modelo proposto por Van Impe et al. (1995) diferentemente do modelo de Corradini e Peleg utiliza o modelo de Gompertz (Equação 1-25) como modelo primário base da modelagem do crescimento de micro-organismos não isotérmico. A Equação 2-28 apresenta a derivada da Equação 1-25 pelo tempo

$$\frac{dy}{dt} = A \left(e^{-e^{B(t-M)}} \right) \left(-e^{B(t-M)} \right) (-M) \quad (2-28)$$

Substituindo a Equação 1-25 na Equação 2-28 e rearranjando matematicamente os termos se obtém a Equação 2-29.

$$\frac{dy}{dt} = M \cdot y \cdot \ln \left(\frac{A}{y} \right) \quad (2-29)$$

Os arranjos matemáticos necessários para suprir a ausência do parâmetro matemático B baseiam-se na solução numérica proposta para a condição inicial, sendo o valor de $y = 0$ como o limite inicial da integração, obtém:

$$n = \ln(N)$$

$$y = n - \ln(N_o)$$

$$\frac{dy}{dt} = \frac{dn}{dt}$$

Como $n_o = \ln(N_o)$, o modelo dinâmico para n é representado pela:

$$\frac{dn}{dt} = M \cdot (n - n_o) \cdot \ln\left(\frac{A}{n - n_o}\right) \quad (2-30)$$

onde, $n - n_o = y$, e sabendo que como condição inicial $y(0) = 0$, podemos substituir estes valores na Equação 1-25 e obter como condição inicial da equação diferencial ordinária a Equação 2-31.

$$n(t = 0) = n_o + A \cdot e^{-e^B} \quad (2-31)$$

Método de Euler

A solução da EDO pelo método de Euler parte do truncamento da série de Taylor (Equação 2-23) no primeiro termo.

Substituindo a Equação 2-23 na equação do modelo de Van Impe (Equação 2-30) se obtém equação discretizada no tempo (Equação 2-32)

$$\frac{n - n^0}{\Delta t} = M(n^0 - n_o) \ln\left(\frac{A}{n^0 - n_o}\right) \quad (2-32)$$

rearranjando os termos da equação 2-32 obtemos:

$$n = n^0 + \Delta t \cdot M(n^0 - n_o) \ln\left(\frac{A}{n^0 - n_o}\right) \quad (2-33)$$

Condição inicial

$$n(t = 0) = n_o + A.e^{-e^B}$$

Método de Runge-Kutta

A solução da EDO pelo método de Runge-Kutta é realizada através da discretização da equação do modelo de Van Impe (Equação 2-30) como mostra a Equação 2-34.

$$f(y, t) = M(y - n_o) \ln\left(\frac{A}{y - n_o}\right) \quad (2-34)$$

Após a discretização (Equação 2-34) a aplicação do método é feita em etapas usando os coeficientes de interpolação. Estes coeficientes variam em número e valor dependendo da ordem de interpolação usado no método. O método de Runge-Kutta de 4ª ordem possui quatro coeficientes: k_1 , k_2 , k_3 e k_4 .

Método de 4ª ordem

$$y^{n+1} = y^n + \frac{\Delta t}{6} (k_1 + 2k_2 + 2k_3 + k_4) \quad (2-35)$$

$$t^{n+1} = t^n + \Delta t \quad (2-36)$$

onde, y^{n+1} é a aproximação por RK4 de $y(t_{n+1})$. Os coeficientes são calculados a partir das Equações 1-13, 1-14, 1-15 e 1-16. Onde a variável f é a função da Equação 2-34.

2.1.4 Índices Estatísticos

Os parâmetros cinéticos foram estimados a partir dos dados experimentais, selecionando o conjunto de parâmetros que minimizam a soma dos desvios quadrados entre dados experimentais e valores calculados. Para medir a precisão do modelo, uma análise de variância foi realizada utilizando-se como

hipótese nula de que os resíduos da regressão não linear são diferentes de zero apenas por causa da uniforme imprecisão na medição da variável resposta (número de células) e não devido à falta de ajuste. (CHARALAMPOPOULOS et.al, 2002). Os índices estatísticos fornecem métodos que auxiliam na interpretação de dados coletados e estimados dos experimentos.

2.1.4.1 Soma dos erros Residuais (RSS)

A soma dos erros residuais (RSS) mede a discrepância entre os dados experimentais e o modelo estimado.

$$RSS = \sum (\text{Valor}_{\text{observado}} - \text{Valor}_{\text{predito}})^2 \quad (2-37)$$

2.1.4.2 Média dos erros ao quadrado (MSE)

Mean Square Error (MSE) ou Erro Quadrado Médio — é à medida que calcula a média dos erros observados entre o valor gerado dos parâmetros estimados ($\text{Valor}_{\text{predito}}$) e o valor observado nos dados experimentais ($\text{Valor}_{\text{observado}}$).

$$MSE = \frac{RSS}{v} = \frac{\sum (\text{Valor}_{\text{observado}} - \text{Valor}_{\text{predito}})^2}{v} \quad (2-38)$$

onde:

v = graus de liberdade.

2.1.4.3 Desvio Padrão (σ)

O desvio padrão é a medida mais comum da dispersão estatística, define-se como a raiz quadrada da variância.

$$\sigma = \sqrt{\frac{1}{n-1} \sum (\text{Valor}_{\text{observado}} - \text{MédiaValores}_{\text{observados}})^2} \quad (2-39)$$

onde:

n = número de observações

Ao se usar o método de mínimos quadrados o desvio padrão pode ser calculado pela Equação 2-40:

$$\sigma = \sqrt{\text{MSE}(\mathbf{J}^T \mathbf{J})^{-1}} \quad (2-40)$$

onde:

J = Matriz Jacobiana dos parâmetros a ser otimizados

2.1.4.4 Intervalo de Confiança (IC)

Intervalo de confiança é um intervalo estimado de um parâmetro estatístico. Ao invés de se estimar o parâmetro por um único valor, é dado um intervalo de estimativas prováveis. O coeficiente de confiança determina quão prováveis são estas estimativas.

$$\gamma_k < \gamma < \gamma_{k+1}$$

IC - Inferior

$$\gamma_k = \gamma - t_{(1-\alpha)} \sigma \quad (2-41)$$

IC - Superior

$$\gamma_{k+1} = \gamma + t_{(1-\alpha)} \sigma \quad (2-42)$$

2.1.4.5 Level-p (Nível de significância (α))

O nível de significância é uma medida da quantidade de provas que se tem contra a hipótese nula. A hipótese nula, tradicionalmente representada pelo símbolo H_0 , representa a hipótese de qualquer alteração ou nenhum efeito.

$$\text{Nível de significância} = \frac{\mu}{\sigma} - t_{(1-\alpha, gl)} \quad (2-43)$$

onde:

μ = média ou próprio parâmetro estimado
 σ = desvio padrão
 t = distribuição de t-Student
 α = nível de confiança (normalmente igual 0.05)

2.1.4.6 Distribuição t-Student (t)

A distribuição t-Student é uma distribuição de probabilidade teórica, pois não é conhecido o valor da variância. O único parâmetro que a define e caracteriza a sua forma é o número de graus de liberdade (v). Quanto maior for esse parâmetro, mais próxima da distribuição normal ela será.

$$t = \frac{Z}{\sqrt{V/v}} \quad (2-44)$$

$$f(t) = \frac{\Gamma\left(\frac{v+1}{2}\right)}{\sqrt{v\pi}\Gamma\left(\frac{v}{2}\right)} \left(1 + \frac{t^2}{v}\right)^{-\left(\frac{v+1}{2}\right)} \quad (2-45)$$

onde:

Z = variável independente com distribuição normal
 V = variável independente com distribuição chi-quadrado
 v = graus de liberdade;
 Γ = função gama;
 t = variável dependente, normalmente é fração de área igual a: 1.0 – nível de confiança.

2.1.4.7 Função Gama (Γ)

A função gama é uma extensão da função fatorial aos números complexos (após uma reparametrização). Aparece muitas vezes com cálculos estatísticos, como no caso da distribuição t-Student.

$$\Gamma(z) = \int_0^{\infty} t^{z-1} e^{-t} dt \quad (2-46)$$

2.1.4.8 Fator Bias

O fator bias representa a diferença média entre os valores observados e preditos.

$$\text{fator bias} = 10 \left(\sum \frac{\log(\text{Valor}_{\text{predito}} / \text{Valor}_{\text{observado}})}{n} \right) \quad (2-47)$$

onde:

n = número de observações

2.1.4.9 Fator de Exatidão

O fator de exatidão é uma medida da diferença média absoluta entre os valores preditos e observados

$$\text{fator exatidão} = 10 \left(\sum \frac{|\log(\text{Valor}_{\text{observado}} / \text{Valor}_{\text{predito}})|}{n} \right) \quad (2-48)$$

onde:

n = número de observações

2.2 ESTRUTURA DO PROGRAMA

2.2.1 Interface Gráfica da Aplicação

A construção de interface é realizada a partir de componentes visuais incorporados nas linguagens de programação e devido à complexidade no desenvolvimento destes, normalmente são utilizados componentes já existentes no mercado.

Apesar do crescimento no número de ferramentas para construção de interfaces, como Java™ Swing, Qt, HTML e JavaScript, Flash, ainda não é fácil projetar boas interfaces gráficas (conjunto de componentes visuais). A expectativa dos usuários cresceu proporcionalmente com os novos componentes de construção de aplicações gráficas, e a interface deve ser

concebida para ser de fácil uso, obedecer ao fluxo de trabalho natural e possuir comportamento intuitivo ao usuário.

A aplicação em desenvolvimento usa VCL (*Visual Components Library*), biblioteca de componentes visuais, a desenvolvida originalmente para o Delphi, ferramenta produzida pela mesma empresa com características semelhantes ao C++ Builder, só que utilizando a linguagem ObjectPascal como linguagem base. E a interface gráfica possui os seguintes elementos principais:

Menu de Ferramentas contém diversos menus relacionados às configurações da aplicação em geral, como abrir/salvar/fechar o projeto atual.

Editor de Dados permite ao usuário efetivamente entrar com os dados necessários à aplicação.

Área de Trabalho possui todas as janelas flutuantes da aplicação, como visualização de gráficos e curvas. Ela provém ao usuário total liberdade na manipulação e disposição das janelas.

A Figura 2-2 ilustra a interface gráfica padrão utilizada no desenvolvimento do aplicativo.

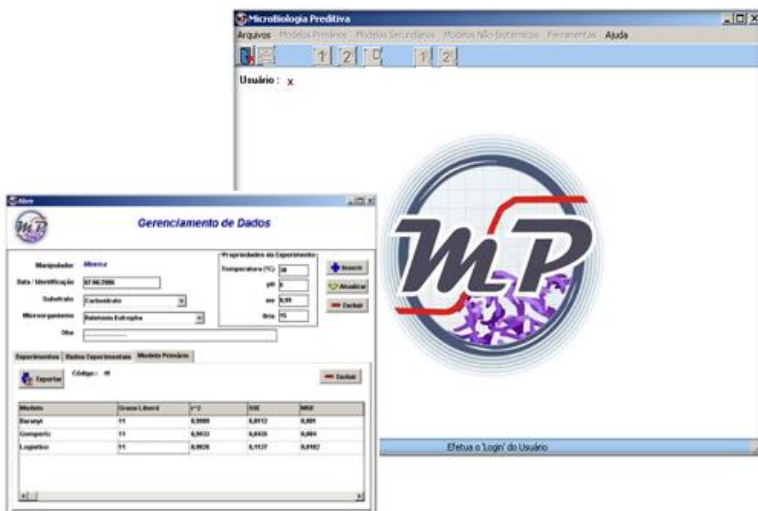


Figura 2-2 – Interface gráfica da aplicação

2.2.2 Linguagem de Programação

C++ *Builder* é um ambiente de desenvolvimento para aplicações cliente/servidor produzido pela Inprise/Borland, tradicional empresa que desenvolve compiladores. O C++ *Builder* utiliza os conceitos de programação visual e dirigida por eventos para proporcionar uma ferramenta RAD (*Rapid Application Development*) extremamente poderosa, que permite desenvolver aplicações eficientes rapidamente. Algumas de suas características utilizadas são:

 Uso de uma biblioteca de componentes visuais, VCL (*Visual Components Library*);

Orientação a objetos: o C++ *Builder* utiliza a linguagem de programação orientada a objetos C++, onde se podem utilizar plenamente os conceitos de POO (Programação Orientada a Objetos) e obter os seus benefícios.

 Ambiente de desenvolvimento (IDE) com ferramentas de produtividade que auxiliam a programação, além de um depurador (*debugger*);

 Acesso a banco de dados: Microsoft Access.

2.2.3 Objetos de Interface

Os objetos de interface são utilizados para melhorar a utilização do software pelos usuários. Cada objeto tem sua característica específica

2.2.3.1 *Button*

É um objeto de interface do tipo botão de empurrão (*push*) (Figura 2-3) também conhecido com botão de comando. O *Button* é o principal acionador de processos dentro do código e é utilizado praticamente em todas as janelas.

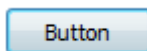


Figura 2-3 – Componente: Button

2.2.3.2 *CheckBox*

O *CheckBox* (ou *checkbox*) (Figura 2-4) é um elemento de seleção de múltipla escolha. É gerada uma lista onde é possível selecionar um ou mais itens.

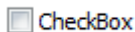


Figura 2-4 – Componente: CheckBox

2.2.3.3 *RadioButton*

O *RadioButton*, (Figura 2-5) também conhecido como *Option Button*, é um elemento de seleção de única escolha. De uma lista só é possível selecionar um item.

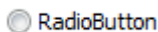


Figura 2-5 - Componente: RadioButton

2.2.3.4 *ComboBox*

ComboBox (Figura 2-6) é uma caixa de combinação onde os dados podem ser digitados ou uma seleção pode ser feita a parti da lista suspensa. É utilizado para escolha de modelos ou campos já definidos pelo aplicativo.

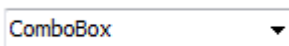


Figura 2-6 – Componente: *ComboBox*

2.2.3.5 *GroupBox*

GroupBox (Figura 2-7) é um contêiner que serve para agrupar os itens semelhantes em um formulário. Normalmente não é utilizado para disparar algum processo, tem papel puramente estético.



Figura 2-7 – Componente: *GroupBox*

2.2.3.6 PageControl

PageControl (Figura 2-8) cria múltiplos componentes de caixa de diálogo. Através deste componente é possível separar em abas inúmeras caixa de diálogo minimizando o uso de outras janelas. Ele amplia a área total da janela.

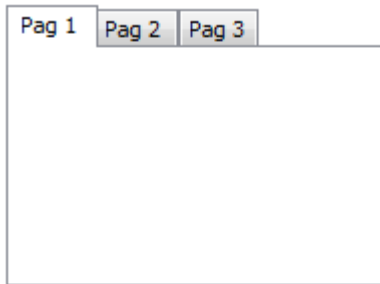


Figura 2-8 – Componente: *PageControl*

2.2.3.7 CheckListBox

CheckBoxList (Figura 2-9) é uma caixa de listagem que contém uma caixa de seleção ao lado de cada item.

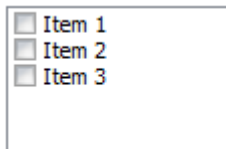


Figura 2-9 – Componente: *CheckListBox*

2.2.3.8 *StringGrid*

StringGrid (Figura 2-10) cria uma grade que exibe os dados (*strings*) em forma de planilha. Os dados são agrupados em colunas e linhas, facilitando o cruzamento de informações.

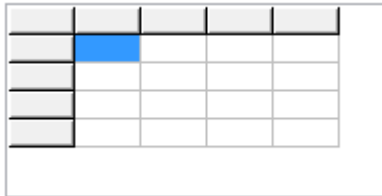


Figura 2-10 – Componente: *StringGrid*

3. RESULTADOS E DISCUSSÕES

3.1 APLICATIVO COMPUTACIONAL

O aplicativo computacional desenvolvido é dividido em três grandes módulos:

- Módulo de cadastro;
- Módulo de entrada de dados;
- Módulo de processamento dos modelos e saída de dados.

3.1.1 Arquitetura do Aplicativo

A Figura 3-1 ilustra a arquitetura do aplicativo. Todos os módulos estão diretamente ligados a uma estrutura principal, que é o banco de dados (*database*), tornando possível gravar e selecionar toda a informação gerada e registrada pelos modelos.

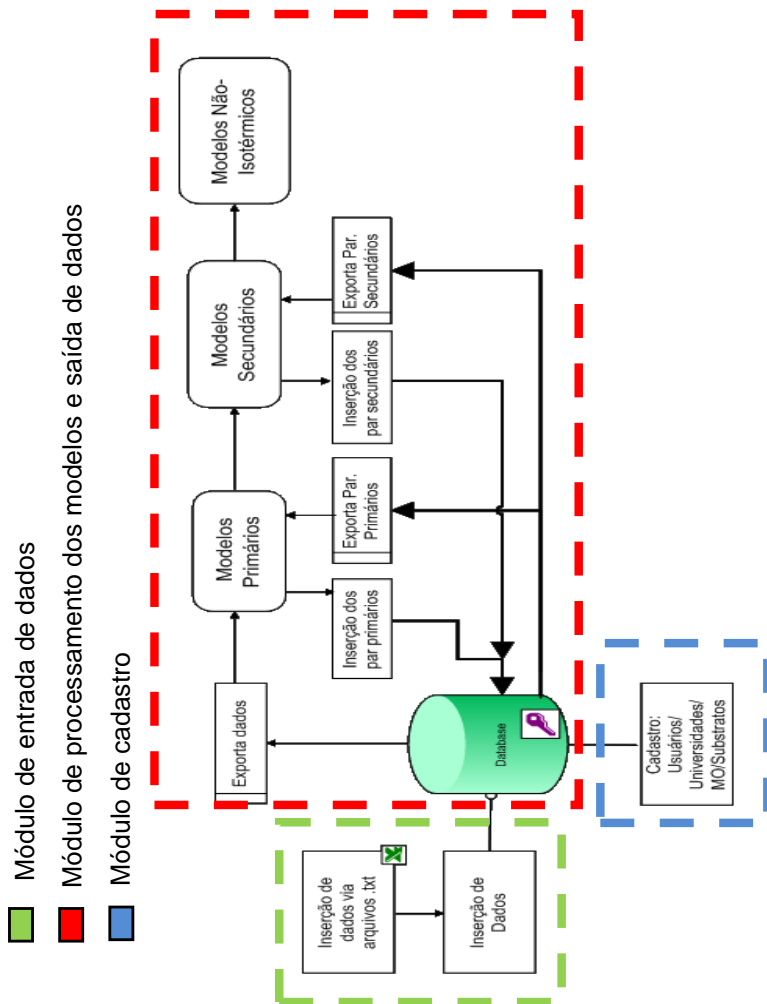


Figura 3-1 – Fluxograma da Arquitetura do Aplicativo

3.1.1.1 Tela Principal

A tela principal (Figura 3-2) é composta por seis menus principais e cada qual possui vários *sub-menus* que acionam os módulos que compõem o aplicativo. O conteúdo de cada menu está detalhado abaixo.

Arquivos

Entrar: Entrada do usuário (*Login*)

Cadastro: Cadastro/Edição/Exclusão de cada item

abaixo

Usuários

Micro-organismos

Cidade

Dados Experimentais: entrada no gerente de dados

Sair: Saída do programa

Modelos Primários

Modelos Primários

Gráficos

Equações

Modelos Secundários

Modelos Secundários

Gráficos

Equações

Modelos Não isotérmicos

Modelos Não isotérmicos

Ajuda

Sobre

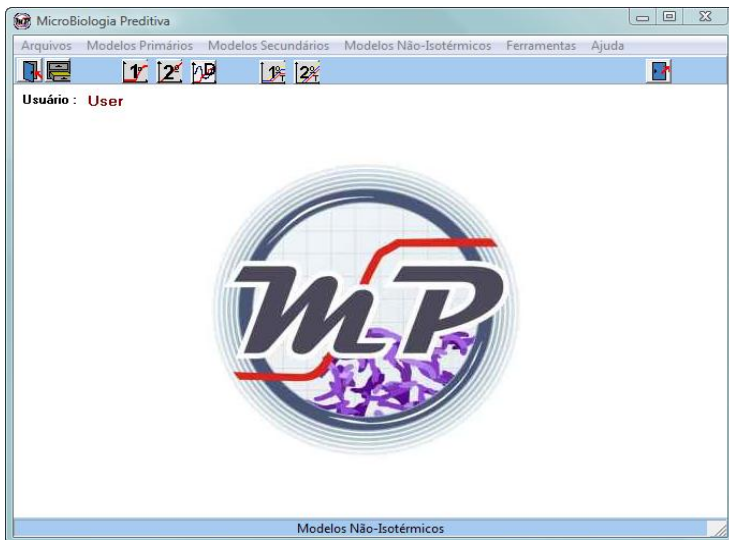


Figura 3-2 – Janela Principal do Aplicativo

3.1.2 Módulo de Cadastros

O módulo de cadastros é a primeira parte do programa, onde são cadastrados todos os dados que serão utilizados durante a vida útil da aplicação. Este módulo é dividido em quatro grupos principais de cadastros: usuários, instituições, microrganismos e cidades. O objetivo dos cadastros é agrupar o maior número de informações que são interligadas a partir de um banco de dados relacional. Isto torna possível fornecer relatórios com filtros específicos que possam ajudar os usuários a cruzar informações que auxiliem em seus trabalhos.

O aplicativo possui uma estrutura padrão de botões e planilhas usadas em todos os formulários. Entre as estruturas usadas para o cadastro/edição/exclusão estão os botões:

Botão 'Inserir': responsável pela inserção dos dados no banco de dados;

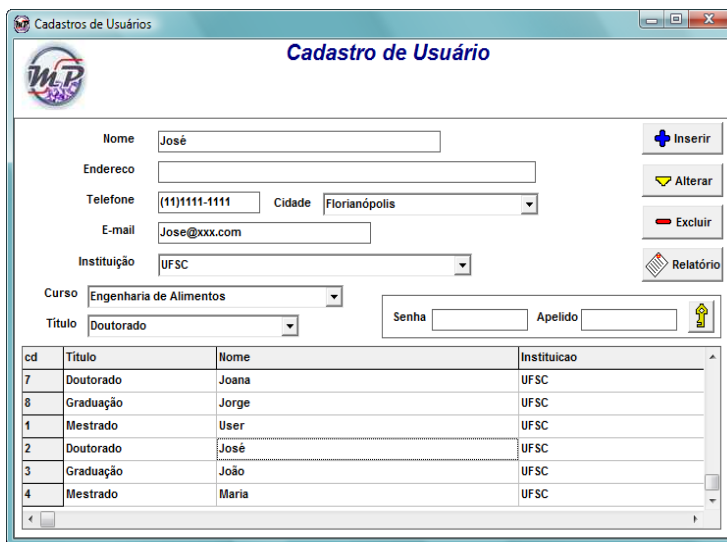
Botão 'Alterar': altera os dados já existentes. Ao acionar este botão aparecerá uma janela com o código referente ao dado que se deseja alterar.

Botão 'Excluir': exclui o conjunto de dados selecionados do banco de dados;

Botão 'Relatório', que gera um relatório que é dividido conforme os campos e que contém todos os dados inseridos no banco de dados (não disponível em todos os cadastros).

3.1.2.1 Cadastro de Usuários

No cadastro dos usuários, é fornecido um formulário para o preenchimento do cadastro com os seguintes campos (Figura 3-3): nome completo, endereço, telefone, email, cidade, instituição, curso título e ainda apelido e senha. Os dois últimos campos, Apelido e Senha, são fundamentais para a sequência do aplicativo, pois é através deles que o aplicativo ativa todos os demais módulos, permitindo ao usuário inserir todo o conjunto de informações de seus experimentos.



cd	Título	Nome	Instituição
7	Doutorado	Joana	UFSC
8	Graduação	Jorge	UFSC
1	Mestrado	User	UFSC
2	Doutorado	José	UFSC
3	Graduação	João	UFSC
4	Mestrado	Maria	UFSC

Figura 3-3 - Cadastro de Usuários

3.1.2.2 Cadastro de Micro-organismos

No cadastro de micro-organismos (Figura 3-4) são inseridos no formulário os campos:

- Espécie: nome científico do microrganismo
- Observações Adicionais
- Classificação: Fungo/Bactéria
- Linhagem: A linhagem do microrganismo
- Faixa de pH ótimo para o crescimento
- Faixa de temperatura ótima (°C) para o crescimento

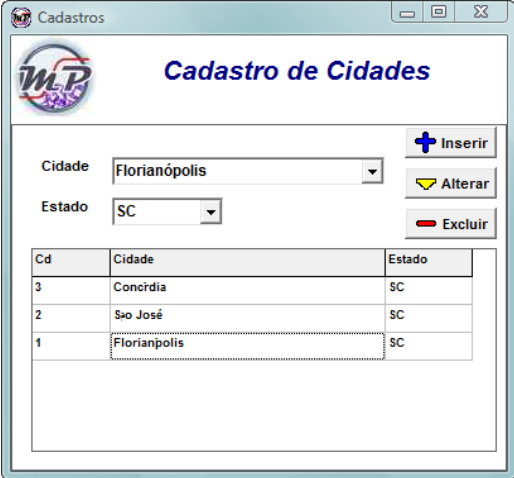
O objetivo desse cadastro é detalhar o máximo possível as características do micro-organismo a ser utilizado no aplicativo.

Cd	Espécie	Linhagem	Características	z(C)
1	Neosartorya fischeri		xxx	0
2	Byssoschlamys nive		Fungo	0
3	Lactobacillus plantarum			0
4	Lactobacillus sakei			4
5	Lactobacillus viridescens			5

Figura 3-4 – Cadastro de Micro-organismos

3.1.2.3 Cadastro de Cidades

A última janela do módulo de cadastros é o cadastro de cidades (Figura 3-5), em que o campo nome da cidade é inserido.



Cd	Cidade	Estado
3	Concórdia	SC
2	Sao José	SC
1	Florianópolis	SC

Figura 3-5 – Cadastro de Cidades

3.1.2.4 Login

Após todos os cadastros realizados, o programa está pronto para prosseguir com os cálculos dos modelos microbiológicos. O *Login* é responsável pela abertura do banco de dados para as futuras inserções e/ou buscas. Após o usuário inserir o *login* (apelido) e a sua senha (Figura 3-6) o aplicativo realiza um filtro no banco de dados verificando se o usuário já está cadastrado e, em caso afirmativo, é feita a validação da senha. Validados o apelido e a senha inseridos, o aplicativo dispara uma mensagem confirmando o nome do usuário e ativa todos os botões de atalho e menus da tela principal Figura 3-7.

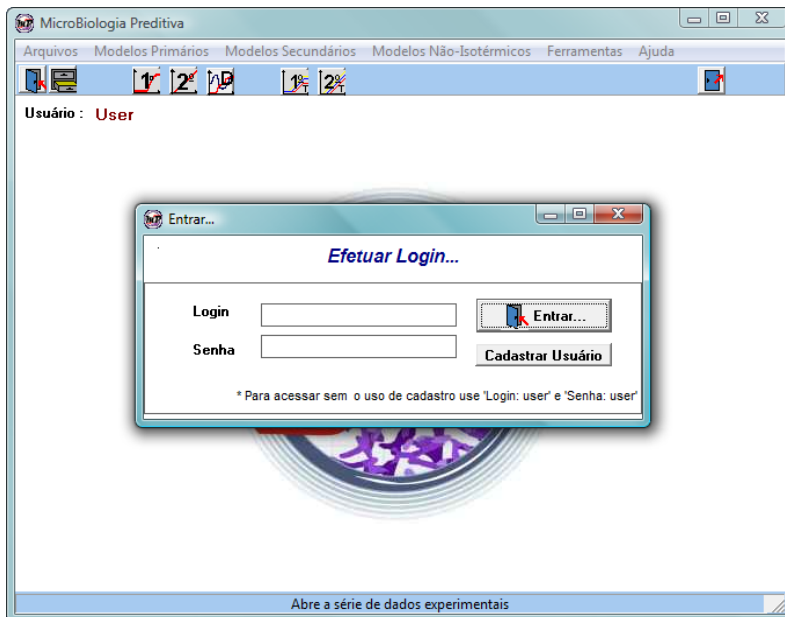


Figura 3-6 – Efetuação do Login

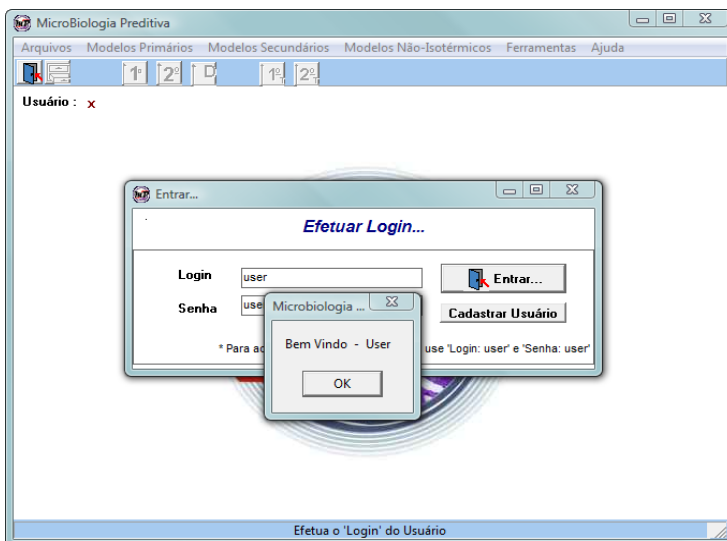


Figura 3-7 – Confirmação do Login

3.1.3 *Módulo de Entrada de Dados*

O módulo de entrada de dados é o mais importante módulo do programa, pois nele são inseridas as entradas dos valores do experimento realizado como:

Micro-organismo;

Substrato;

Data de realização;

Observações;

Curvas de crescimento;

E dados físicos como: temperatura, pH, atividade de água (a_w) e grau Brix.

Com os dados inseridos, é possível trabalhar com todos os modelos microbiológicos presentes no aplicativo.

Na aba 'Experimentos', como mostra Figura 3-8, há uma planilha onde são impressos todos os dados já cadastrados no banco de dados gravados pelo manipulador do aplicativo. Cada linha da planilha corresponde a um dado inserido pelo usuário e cada coluna é um campo do experimento. Ao clicar em alguma coluna, o programa identifica internamente o código correspondente ao experimento, e o usuário tem a opção de enviar os dados diretamente ao módulo de modelos primários.

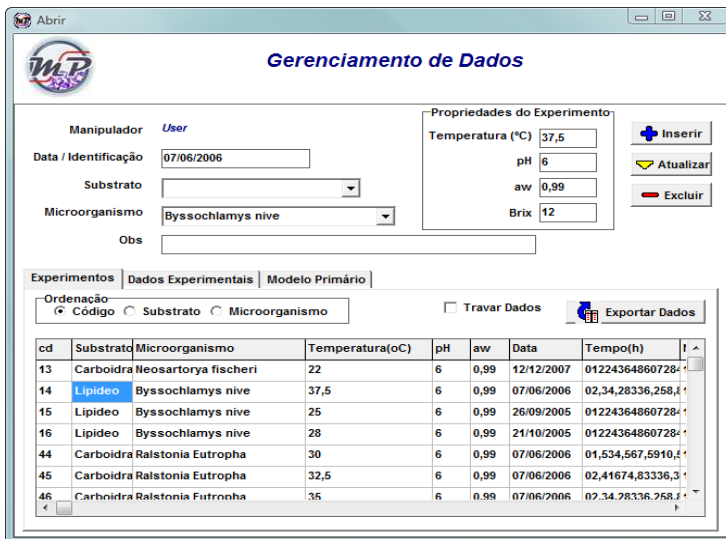


Figura 3-8 – Gerenciamento de Dados

Na aba 'Dados Experimentais', Figura 3-9, é realizada a entrada dos pontos experimentais (dados de crescimento dos microrganismos (Unidade Formadora de Colônia (UFC/mL)) com o tempo) através de um arquivo de texto ou a entrada direta de dados na planilha. Selecionando o *checkbox* 'Diâmetro' a possibilidade de entrar com a variação do diâmetro da colônia pelo tempo. Esta opção é importante para o trabalho com fungos, pois normalmente os dados experimentais são coletados medindo o aumento do diâmetro. Após inserir a série de dados, há mais dois campos que podem ser preenchidos: Nome da Série de Dados e a Cor da Série, que servem para diferenciar quando há mais uma curva importada.

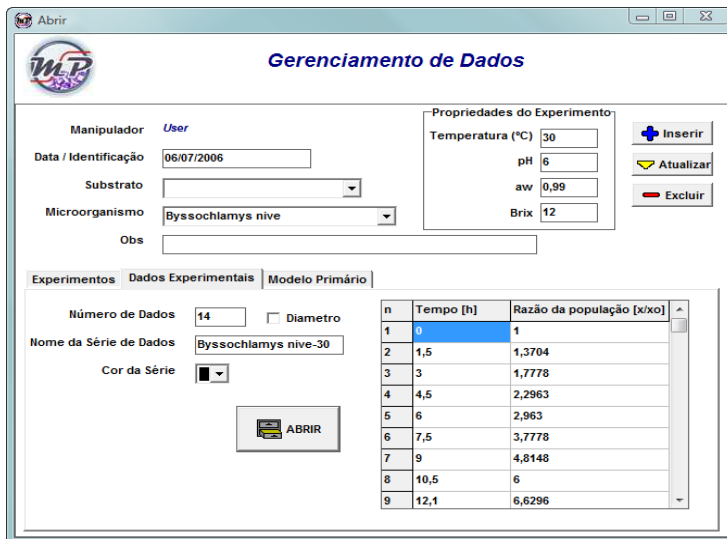


Figura 3-9 – Gerenciamento de Dados: Aba Dados Experimentais

Ainda no gerenciamento de dados a última aba 'Modelo Primário' (Figura 3-10) são enviados na planilha os dados do modelo primário salvos no banco de dados para o experimento previamente selecionado através na aba 'Experimentos'. Nesta aba, o usuário tem a opção de exportar o modelo primário diretamente para o módulo gerenciador dos modelos primário. A Figura 3-11 mostra a janela que é acionada ao acionar o botão exportar, nela o usuário pode editar o texto para distinguir das demais curvas.

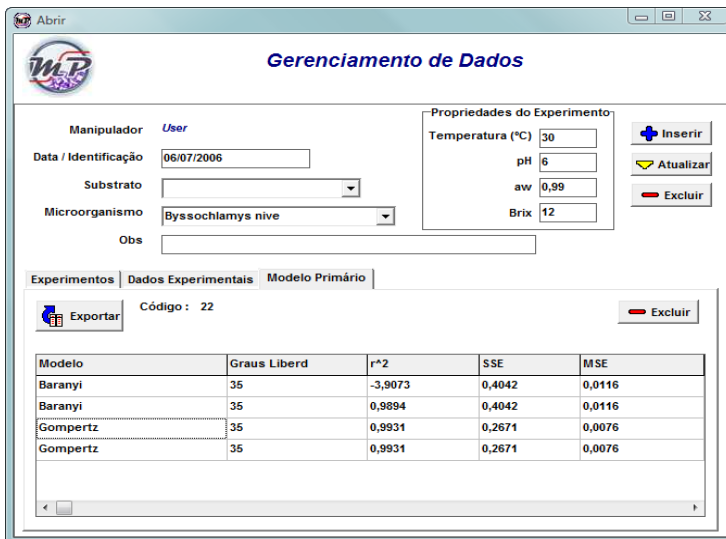


Figura 3-10 – Gerenciador de dados Aba: Modelo Primário

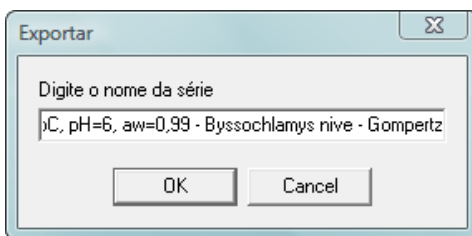


Figura 3-11 – Gerenciador de dados: Tela Exportar

3.1.4 Módulo de Processamento dos Modelos e Saída de Dados

3.1.4.1 Modelos Primários

Os modelos primários são ajustados através do módulo de modelo primário que é dividido em duas partes:

Seleção do modelo primário

Cálculo do modelo e saída de resultados

Seleção do modelo primário

A primeira janela (Figura 3-12) determina a escolha do modelo primário através de um objeto de interface chamado *Combobox*. Ao clicar neste objeto, aparecerão todos os modelos primários disponíveis para o cálculo. Após a escolha do modelo, o *software* imprime uma breve descrição do modelo primário e permite ao usuário prosseguir com o cálculo ativando uma nova janela ao acionar o botão 'Ir'.

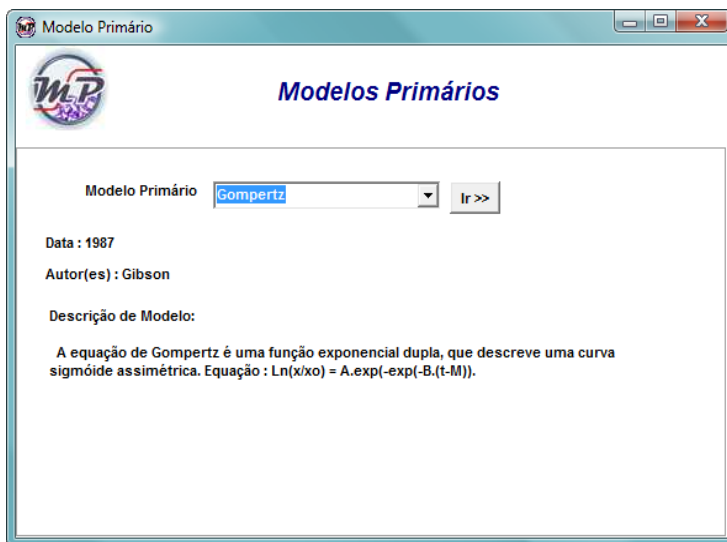


Figura 3-12 – Escolha do modelo primário

Cálculo do modelo e saída de resultados

A segunda janela (4.13) é acionada após a escolha do modelo primário e acionamento do botão ‘Ir’. No construtor da janela, que é o método de inicialização da classe dos modelos primários, há a incorporação dos dados inseridos no gerenciador dos dados, que podem ser visualizados na aba ‘Dados Experimentais’. Também são apresentadas as informações sobre as condições dos dados experimentais bem como o nome do micro-organismo estudado, data do experimento e o código do número de identificação do banco de dados (ID).

Para realizar o cálculo do modelo é necessário escolher qual o método de minimização: Gauss-Newton ou Marquardt, no *groupbox* ‘Métodos’.

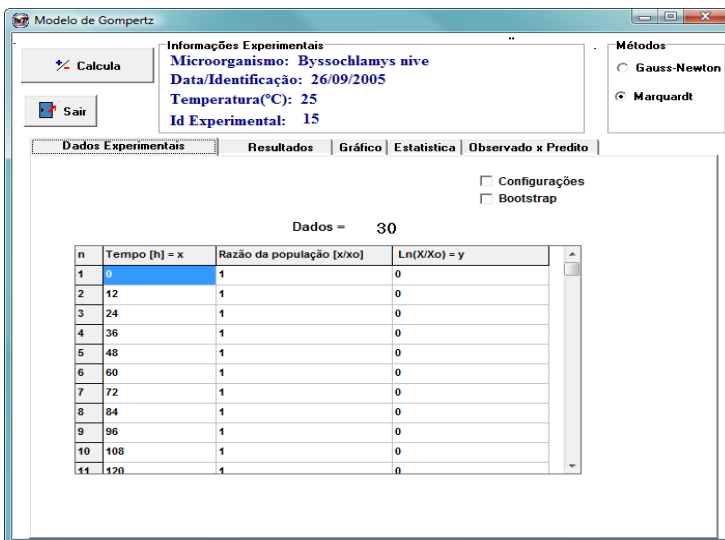


Figura 3-13 – Dados Experimentais do Modelo Primário

Ainda na Aba ‘Dados Experimentais’ há o objeto *Checkbox* chamado ‘Configuração’ que, quando selecionado, ativa mais uma aba também chamada ‘Configurações’ onde são ajustadas entradas avançadas dos métodos utilizados para o ajuste não-linear. Os campos mostrados nesta tela (Figura 3-14) são

descritos abaixo. As configurações avançadas somente são necessárias quando os métodos utilizados não convergirem para o valor esperado, normalmente utilizam valores padrões para os campos apresentados abaixo.

Marquardt

Configurações Avançadas

In[0] : Valor mínimo aceitável (Precisão da máquina)

In[3]: Tolerância relativa para a diferença entre a norma da última e penúltima iteração;

In[4]: Tolerância absoluta para a diferença entre a norma da última e penúltima iteração;

O processo é terminado se o ganho da soma dos quadrados é menor que $\text{In}[3] \cdot (\text{SSE}) + \text{In}[4] \cdot \text{In}[4]$;

In[6]: é o valor inicial usado para relação entre o gradiente e a direção Gauss-Newton; se o problema é bem condicionado o valor padrão é 0.01; se o problema mal condicionado o parâmetro deve assumir valores maiores.

Saída

Out[1]: fornece informação sobre o processo de finalização do método;

- Out[1] = 0: finalização normal;
- Out[1] = 1: o processo foi finalizado, pois atingiu o número Máximo de iterações;
- Out[1] = 2: o processo foi finalizado devido à função tender a zero;
- Out[1] = 3: o processo foi finalizado devido os parâmetros estimados atribuírem valor zero a função;
- Out[1] = 4: houve quebra do processo, pois não pode ser atingida a tolerância especificada;

Out[2]: valor da norma do último vetor resíduo calculado;

Out[3]: valor da norma do primeiro vetor resíduo calculado;

Out[4]: número de chamadas das funções;

Out[5]: número de iterações;

Out[6]: o valor da soma do vetor de incrementos no passo da última iteração;

Out[7]: o número de condição $J^T J$.

Gauss-Newton

Configurações Avançadas

In[0]: valor mínimo aceitável (Precisão da máquina);

In[1]: valor da tolerância relativa para o vetor incremento;

In[2]: valor da tolerância absoluta para o vetor incremento;

O processo é terminado se em alguma iteração a norma calculada é menor que $\ln[1] * ||\text{par}|| + \ln[2]$;

In[4]: valor da tolerância absoluta para a diferença entre a norma da última e penúltima iteração;

In[6]: valor máximo permitido de reduções dos passos do vetor incremento.

Saída

Out[1]: fornece informação sobre o processo de finalização do método

- Out[1] = 0: finalização normal;
- Out[1] = 1: o processo foi finalizado, pois atingiu o número máximo de iterações;
- Out[1] = 2: o processo foi finalizado devido à função tender a zero;
- Out[1] = 3: o processo foi finalizado devido os parâmetros estimados atribuírem valor zero a função;
- Out[1] = 4: houve quebra do processo, pois não pode ser atingida a tolerância especificada;

Out[2]: valor da norma do último vetor resíduo calculado;

Out[3]: valor da norma do primeiro vetor resíduo calculado;

Out[4]: número de chamadas das funções;

Out[5]: número de iterações;

Out[6]: o valor da soma do vetor de incrementos no passo da última iteração;

Out[7]: o número de condição $J^T J$.

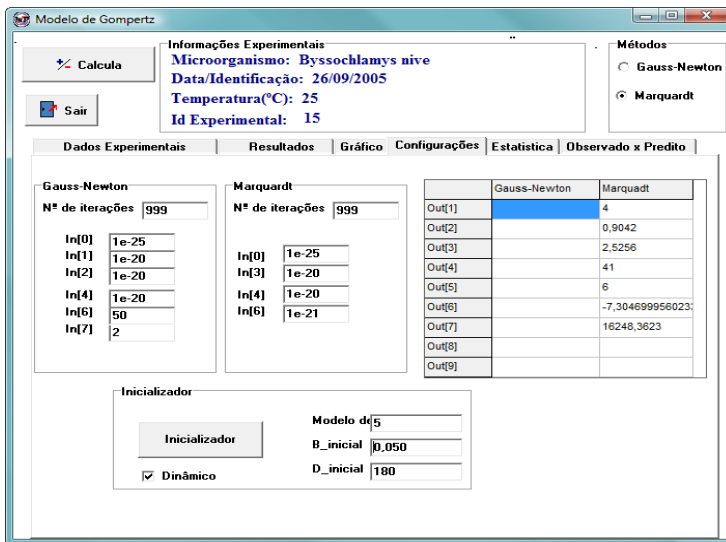


Figura 3-14 – Configurações Avançadas dos Métodos de Resolução de Problemas de Mínimos Quadrados

Através do botão 'Inicializador' é feita uma varredura completa no domínio dos parâmetros, procurando quais os valores possuem o menor erro absoluto em relação aos dados experimentais. Esta inicialização dos parâmetros é necessária quando o aplicativo não encontra a solução desejada, isto é, quando o parâmetro de saída *Out[1]* (configurações avançadas) for diferente de zero. Portanto como os métodos de ajuste utilizam as derivadas partindo de um ponto inicial, muitas vezes é necessário o uso do inicializador dos parâmetros, que evita que os pontos iniciais caiam em um mínimo local tornando impossível a obtenção de valores microbiologicamente coerentes.

The image shows a software window titled "Achar parametros iniciais". It contains a table with three columns: "Incial", "Precisão", and "Final". There are three rows of input fields. The first row is labeled "A" and has values 1, 0,5, and 15. The second row is labeled "B" and has values 0,01, 0,01, and 0,5. The third row is labeled "D ou M" and has values 1, 1, and 15. Below the table is a button labeled "Procurar". At the bottom right of the window, there is a note: "Obs: Final > inical".

	Incial	Precisão	Final
A	1	0,5	15
B	0,01	0,01	0,5
D ou M	1	1	15

Obs: Final > inical

Figura 3-15 – Configurações Avançadas: Inicializador

Os parâmetros do modelo são obtidos pelo acionamento do botão 'Calcula', onde serão realizados todos os cálculos necessários para a obtenção dos resultados do modelo.

A visualização das saídas do modelo (parâmetros obtidos) é apresentada na aba 'Resultados' como ilustra a Figura 3-16, onde são apresentados os parâmetros matemáticos, o índice de correlação alcançado e também os parâmetros microbiológicos A , μ e λ .

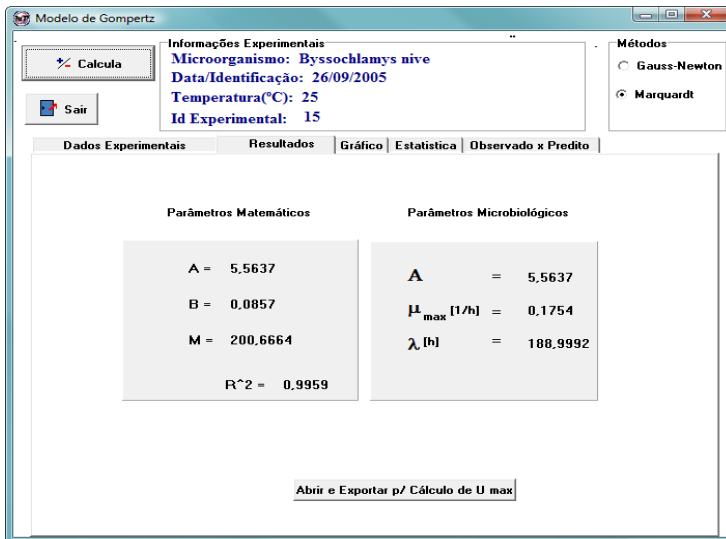


Figura 3-16 – Modelo Primário: Aba Resultados

Após o cálculo dos parâmetros microbiológicos, o programa possibilita a obtenção da velocidade máxima de crescimento usando a derivada da equação do modelo primário. O cálculo do μ_{max} é feito usando o método de Newton para encontrar o valor máximo da função da velocidade de crescimento pelo tempo (Figura 3-17).

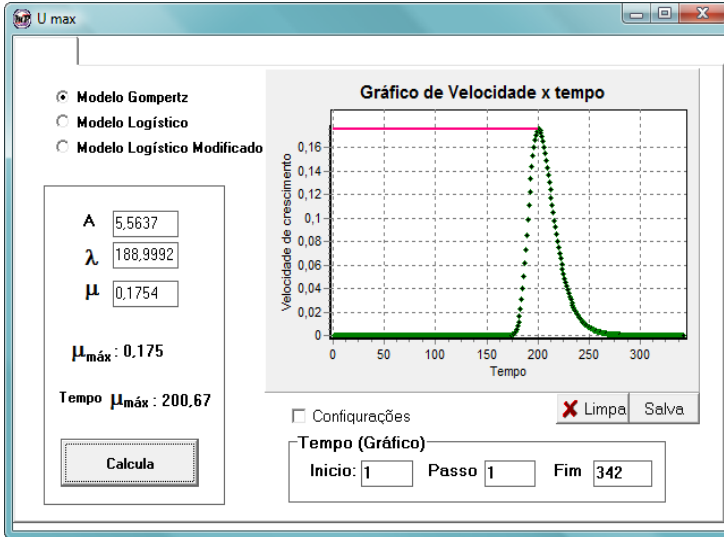


Figura 3-17 – Cálculo da velocidade máxima de crescimento

Na aba 'Gráfico' são apresentados a curva do modelo primário otimizado e os pontos experimentais, como mostra a Figura 3-18. Nesta mesma aba, há mais dois botões de acionamento: o botão 'Salva gráfico .bmp', que permite que a figura do gráfico gerado seja salva em formato bitmap (.bmp) e o botão 'Ajuste' que configura os ajustes nos valores e nomes dos eixos do gráfico (Figura 3-19).

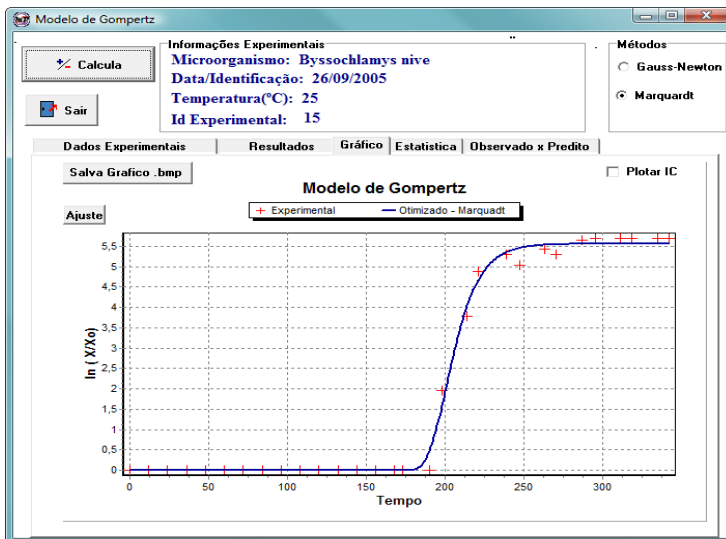


Figura 3-18 – Modelo Primário: Aba Gráficos

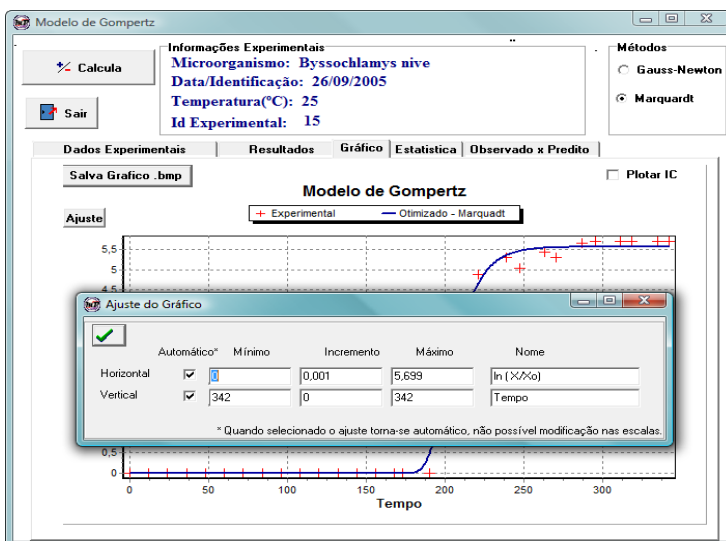


Figura 3-19 – Modelos Primários: Ajuste de Gráficos

Outros resultados importantes ilustrados na Figura 3-20, são os dados estatísticos gerados na otimização. Cada linha da *stringgrid* apresenta uma variável estatística. Nas colunas estão agrupados os valores estatísticos totais do modelo primário e também os valores de cada parâmetro matemático. Os dados estatísticos calculados são:

- SSE: Sum of squared errors
- MSE: *Mean squared error*
- Desvio Padrão
- Valor-t (g)
- Intervalo de confiança acima (Up IC)
- Intervalo de confiança abaixo (Down IC)
- Fator Bias
- Fator de Exatidão (Accuracy)

O usuário ainda pode escolher o nível de confiança (ou nível de significância). Como padrão, o aplicativo utiliza 95%.

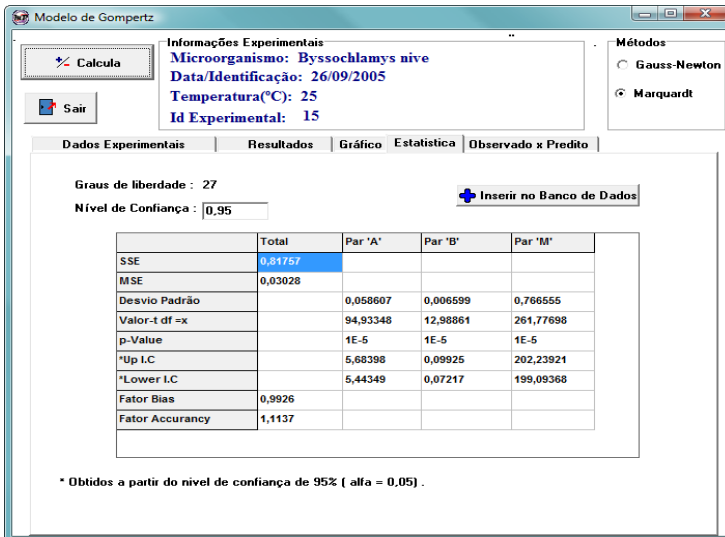


Figura 3-20 – Modelos Primários: Aba Estatística

O último gráfico do modelo primário, Figura 3-21, é a comparação entre as dados de crescimento preditos e observados. Como nos demais gráficos, possui botões de ajuste e de salvar em formato bitmap.

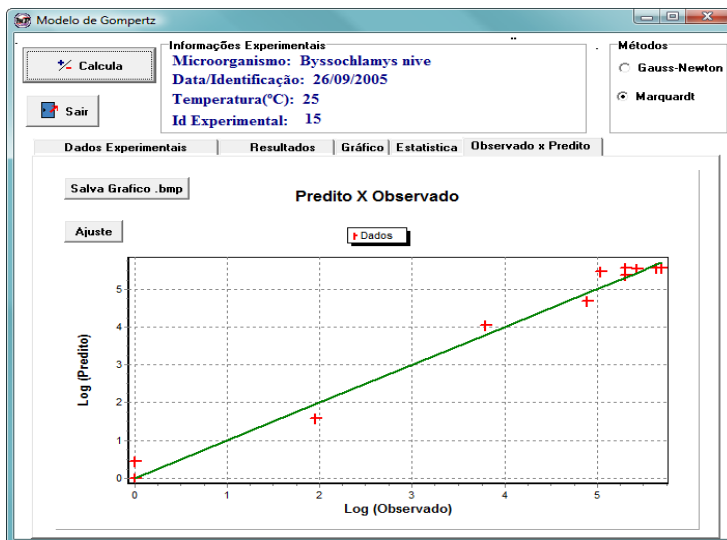


Figura 3-21 – Modelo Primário: Aba Gráfico - Observado x Predito.

3.1.4.2 Gerenciador de Dados - Modelos Primários

O gerente de gráficos dos modelos primários tem a finalidade de graficar as curvas selecionadas, comparar as variáveis estatísticas de cada modelo ou de cada conjunto de parâmetros físicos (temperatura, pH e a_w) e exportar os dados para o modelo secundário.

Conforme são inseridos os dados através da aba 'Modelo Primário' mostrado na Figura 3-10, os dados podem ser visualizados e também selecionados, como o usuário desejar. A Figura 3-22 mostra a aba 'Seleção de Dados' no gerente de gráficos dos modelos primários.

O botão 'Exportar para o Modelo Secundário', juntamente com os botões de seleção dos parâmetros físicos (temperatura, pH e a_w), permite que sejam enviados os dados selecionados ao módulo secundário, descrito adiante.

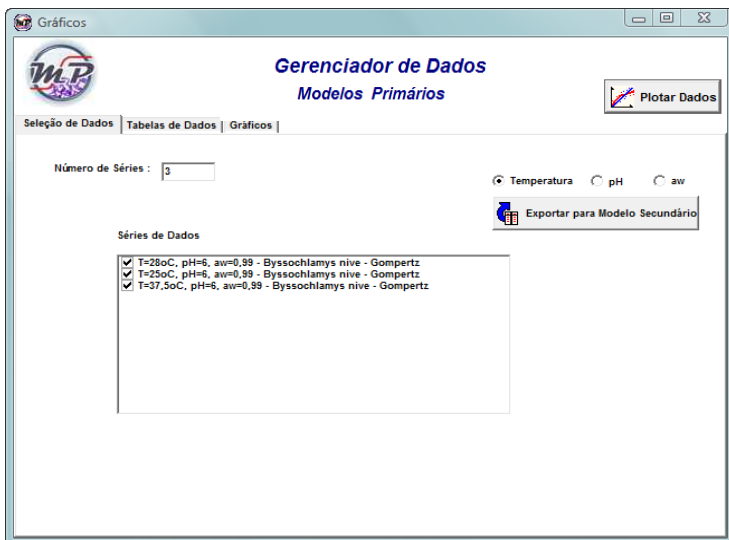


Figura 3-22 – Gerenciador de Dados: Modelos Primários – aba Seleção de Dados

Todos os dados inseridos no gerente de gráficos são agrupados em planilhas ilustradas na Figura 3-23. Estes dados são divididos em dois grupos: o primeiro grupo contém os dados gerais do experimento, como código, o modelo primário utilizado, os dados de tempo e razão de crescimento, graus de liberdade e índices estatísticos. O segundo grupo contém os valores dos parâmetros matemáticos do modelo e os índices estatísticos calculados.

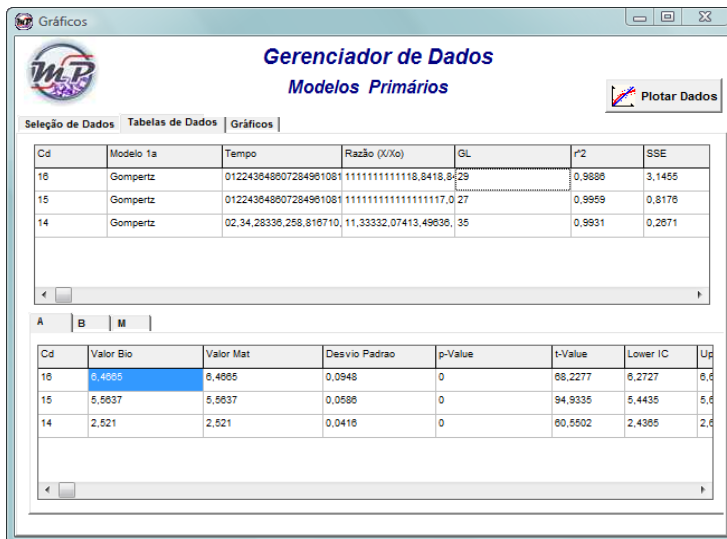


Figura 3-23 – Gerenciador de Dados: Modelos Primários – aba Tabela de Dados

Ao acionar o botão ‘Plotar Gráfico’ o programa gera os gráficos com os dados que foram selecionados permitindo a comparação de inúmeras combinações de experimentos, modelos e variações dos parâmetros físicos como mostra a Figura 3-24.

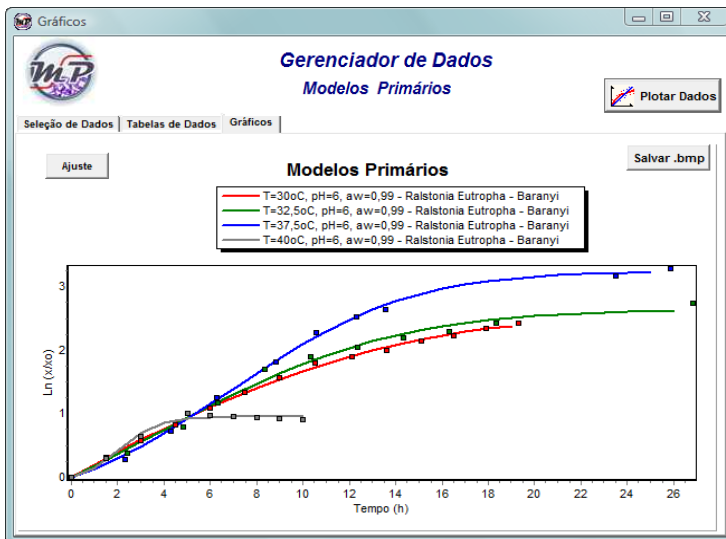


Figura 3-24 -- Gerenciador de Dados: Modelos Primários – aba Gráficos

3.1.4.3 Modelo Secundário

No módulo dos modelos secundários estão agrupados todos os modelos que permitem descrever variações das variáveis microbiológicas, obtidas nos modelos primários, em função de algum parâmetro físico como: temperatura, pH e atividade de água.

O aplicativo faz com que a alimentação de valores seja feita de forma automática no modelo secundário, isto é, conforme são calculados os parâmetros microbiológicos no modelo primário o aplicativo já envia os resultados com o respectivo valor do parâmetro físico para as planilhas de dados do módulo ilustrados na Figura 3-25. Os dados também podem ser alimentados através da seleção realizada no 'gerenciador de gráficos do modelo primário' e ainda na edição direta da planilha.

A escolha do modelo secundário é realizada selecionando o próprio modelo no *menu* 'Modelos Secundários'. Após a escolha, o programa está pronto para o processamento do

modelo que é acionado pelo botão 'Calcula'. As Figuras 3-26 e 3-27 apresentam o gráfico e os resultados do modelo secundário, respectivamente.

Modelo de Arrhenius

Dados

Parâmetro 'A'

n	Temperatura(°C)	A	Cd
0	37,5	2,521	14
1	25	5,5837	15
2	28	6,4665	16

Parâmetro 'U max'

n	Temperatura(°C)	U max	Cd
0	37,5	0,0695	14
1	25	0,1754	15
2	28	0,1363	16

Parâmetro 'Lambda'

n	Temperatura(°C)	Lambda	Cd
0	37,5	293,1162	14
1	25	188,9992	15
2	28	111,6367	16

Figura 3-25 - Dados do Modelo Secundário

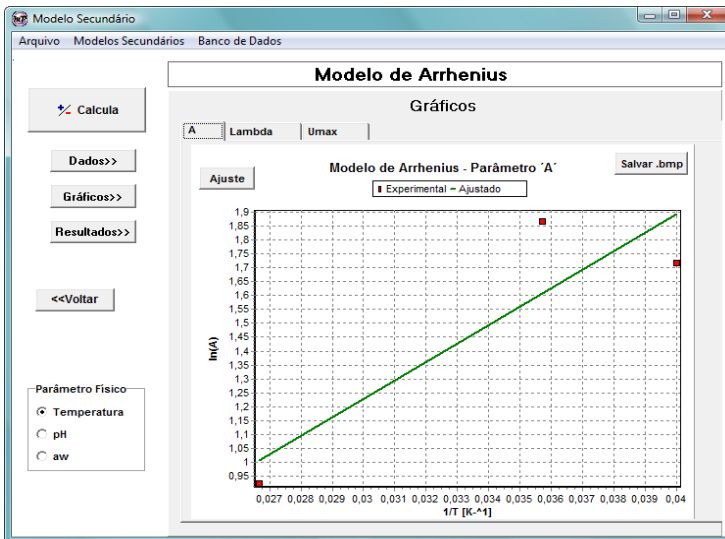


Figura 3-26 – Gráficos do Modelo Secundário

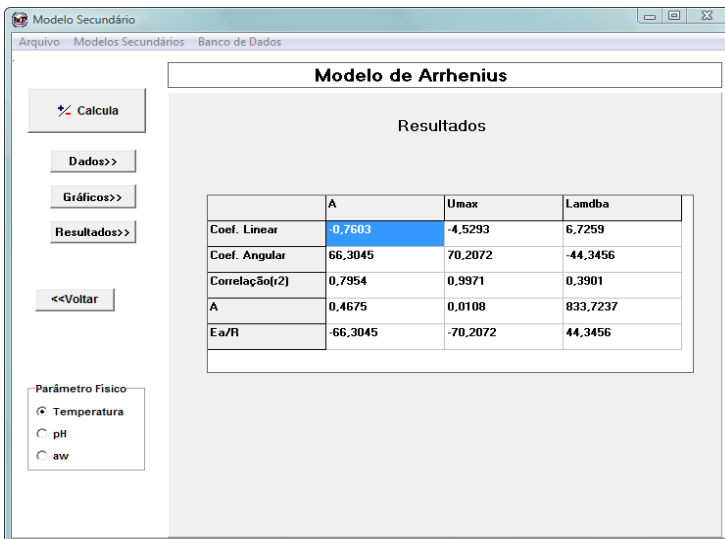


Figura 3-27 – Resultados do Modelo Secundário

Após realizar todos os cálculos do modelo secundário, há a opção de inserir os resultados na base de dados do usuário. O procedimento para adicionar os resultados é clicar no *menu* 'Banco de Dados' e em seguida no *sub-menu* 'Inserir', então aparecerá uma janela (Figura 3-28) com botões de inserir os dados e botão de excluir os dados já existentes. O botão 'Exportar' tem a função de enviar os dados selecionados da planilha para o gerenciador dos dados secundários para futura análise e/ou exportar para o modelo dinâmico.

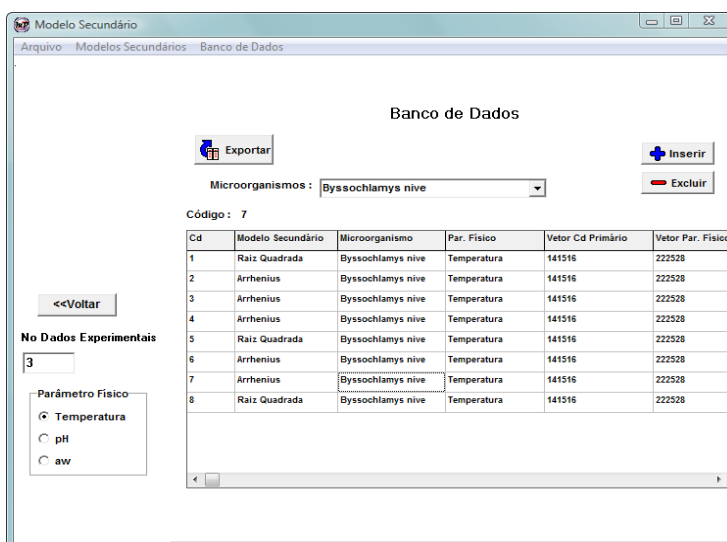


Figura 3-28 – Banco de Dados do Modelo Secundário

3.1.4.4 Gerenciador de Dados - Modelo Secundário

O gerenciador de dados do modelo secundário possui as mesmas características e funcionalidades do gerenciador do modelo primário. A única diferença está na exportação dos dados secundários para o modelo dinâmico, pois no modelo secundário é exportado somente um dado selecionado e a visualização do gráfico é feita de acordo com a seleção do parâmetro secundário ativado no *Radiobutton* (botão de seleção individual), após o acionamento do botão 'Plotar Dados' mostrados na Figura 3-29.

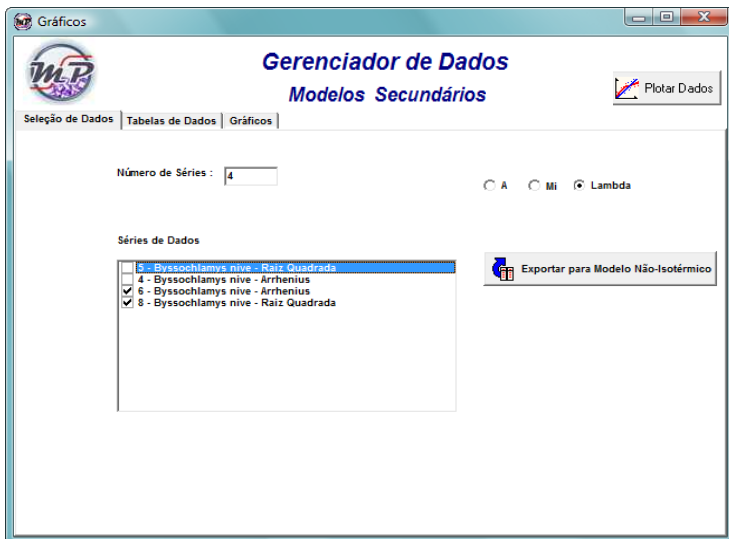


Figura 3-29 – Gerenciador de Dados: Modelos Secundários – aba Seleção de Dados

A Figura 3-30 e a Figura 3-31 ilustram a impressão em planilhas dos valores dos parâmetros gerados e os gráficos da seleção de dados do modelo secundário.



Figura 3-30 – Gerenciador de Dados: Modelos Secundários – aba Tabela de Dados

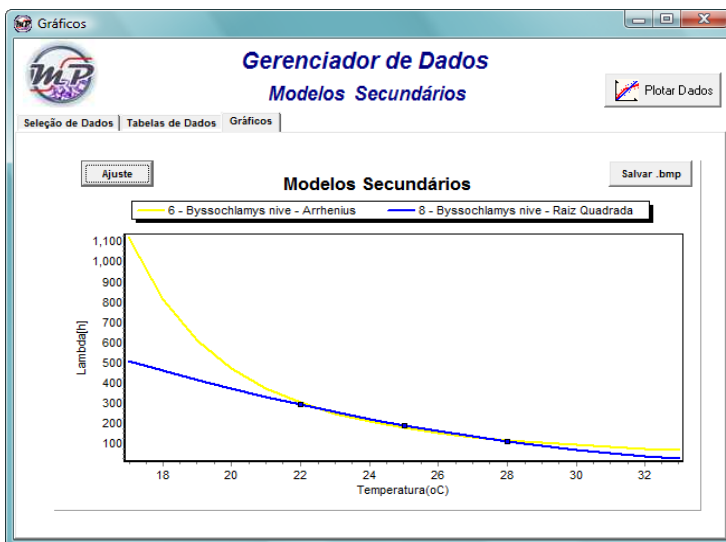


Figura 3-31 -- Gerenciador de Dados: Modelos Secundários – aba Gráficos

3.1.4.5 Modelo Não isotérmico

No módulo do modelo não isotérmico são inseridos os dados dos modelos secundários que são utilizados para prever o crescimento dos microrganismos à determinada variação do parâmetro físico especificado.

A Figura 3-32 mostra a tela de configuração de entrada do modelo dinâmico onde são definidos: o parâmetro físico (temperatura, pH ou a_w), o método de resolução da equação diferencial ordinária (EDO) que pode ser realizada pelo método de Euler ou Runge-Kutta e o modelo dinâmico (Van Impe ou Corradini- Peleg), além de selecionar os valores dos parâmetros e o tipo do modelo secundário.

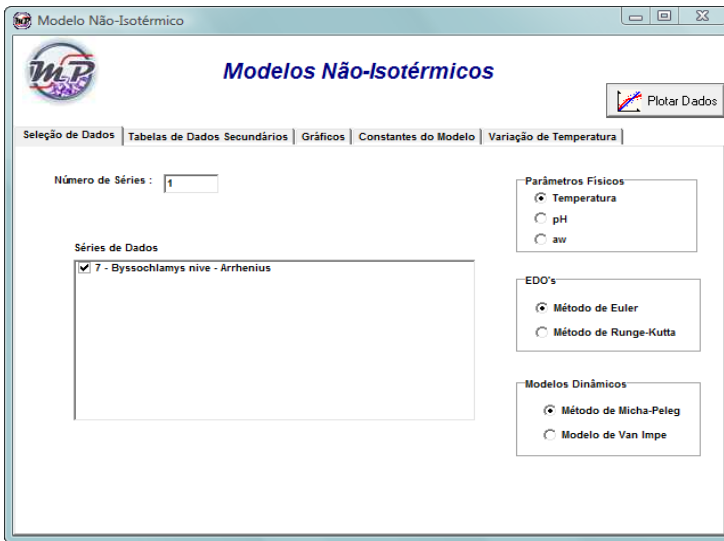


Figura 3-32 - Modelos Não isotérmicos – aba Seleção de Dados

Após a seleção dos modelos, é necessário definir a variação de temperatura (ou qualquer outro parâmetro físico) ao qual o microrganismo simulado será submetido. Na Figura 3-33, na aba 'Variação de Temperatura', há duas opções de entrada para a variação do parâmetro físico. A primeira é uma função

com inúmeras possibilidades de tendência, que vão desde uma função constante até variações logarítmicas. E a segunda forma é a inserção direta de um conjunto de dados da variação da temperatura com o tempo.

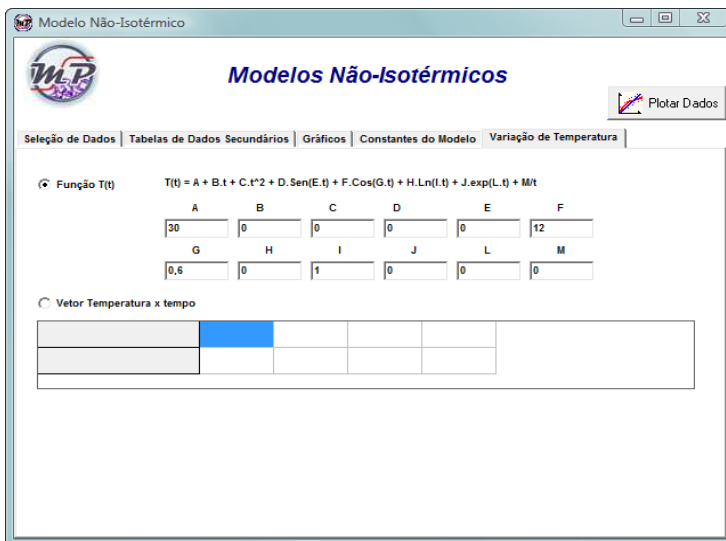


Figura 3-33 - Modelos Não isotérmicos – aba Variação de Temperatura

A Figura 3-34 exibe os dados e equação do modelo secundário enviados para o modelo não isotérmico.

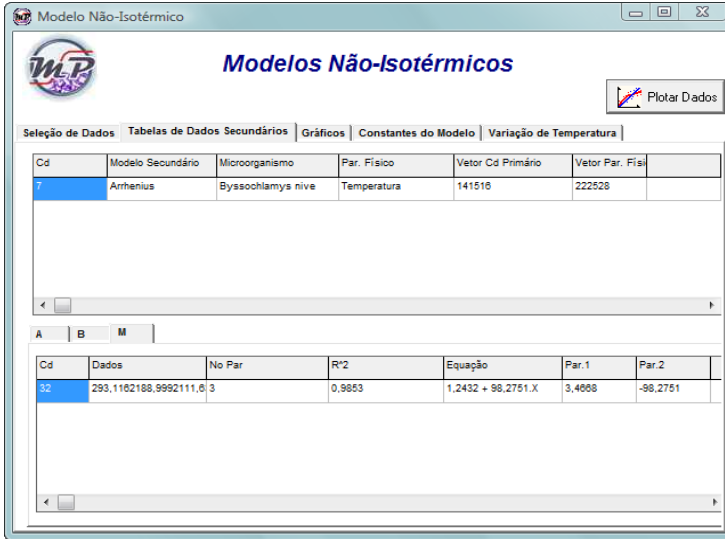


Figura 3-34 - Modelos Não isotérmicos – aba Tabela de Dados Secundários

Na aba 'Constantes do Modelo', isto é, parâmetros matemáticos (Figura 3-35) é feita a entrada do tempo total de simulação juntamente com o número total de passos discretos, usados para resolução numérica da EDO. Há ainda uma planilha que poderá ser preenchida com dados experimentais inseridos via arquivos de texto que serão ilustrados juntamente com os valores simulados. A Figura 3-36 mostra a saída gráfica da simulação do crescimento dos microrganismos com o tempo, dada uma variação de temperatura.

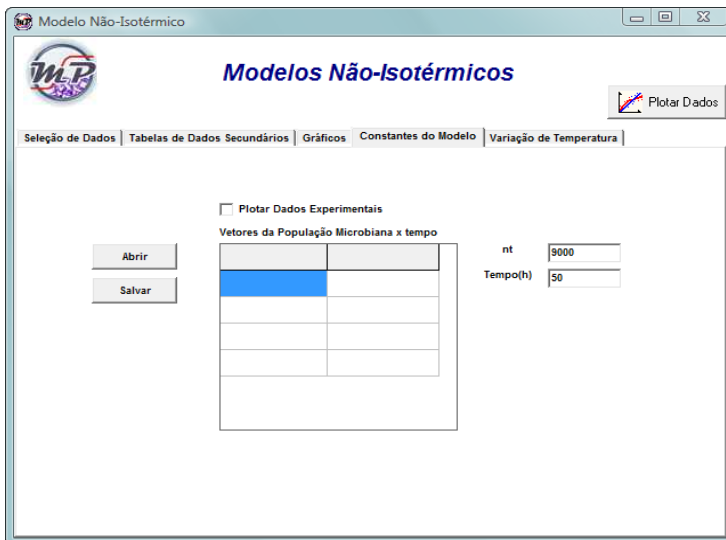


Figura 3-35 - Modelos Não isotérmicos – aba Constantes do Modelo

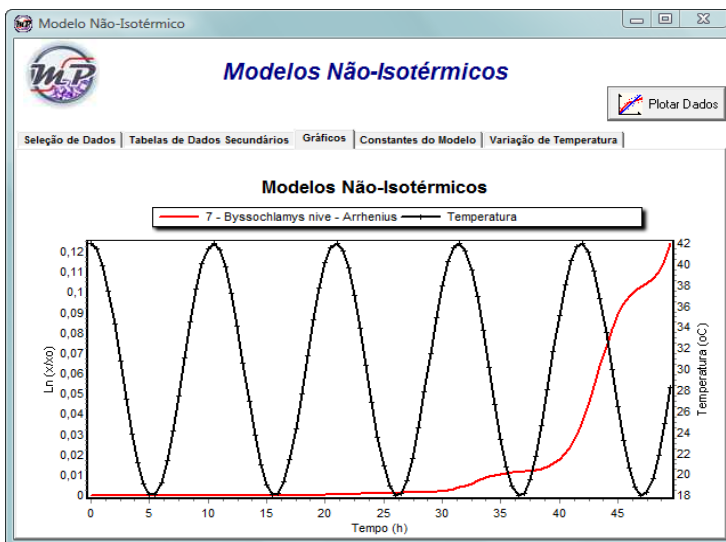


Figura 3-36 - Modelos Não isotérmicos – aba Gráficos

3.2 VALIDAÇÕES

Todos os valores encontrados pelos modelos calculados foram comparados com *softwares* comerciais comuns no mercado. O *software* Statística 7.0 foi utilizado para as validações dos valores encontrados no aplicativo computacional em desenvolvimento.

3.2.1 Dados Experimentais

Os dados experimentais utilizados para a validação dos modelos foram retirados da literatura. Foi utilizada somente uma curva de crescimento (Tabela 3-2) e os parâmetros físicos utilizados para obtenção da curva são todos constantes como ilustra a Tabela 3-1. Os resultados obtidos pelo aplicativo proposto, utilizando os dados experimentais, foram comparados com os resultados gerados pelo *software* comercial Statística 7.0. O microrganismo selecionado foi a bactéria *Ralstonia eutropha*.

Tabela 3-1 – Parâmetros Constantes do Experimento

Temperatura (°C)	37,5
pH	6
Aw	0,99
Bactéria	<i>Ralstonia eutropha</i>

Tabela 3-2 – Dados Experimentais de razão de crescimento (N/N_0) versus tempo (h)

N/No	Tempo (h)
1,000	0,00
1,333	2,30
2,074	4,28
3,496	6,25
6,141	8,82
9,778	10,58
12,444	12,32
13,889	13,57
23,778	23,48
26,556	25,88

3.2.2 Método dos Mínimos Quadrados

Na validação dos resultados obtidos dos modelos primários e secundários do aplicativo foram comparados desde os parâmetros estimados, usando os métodos de otimização, até os dados estatísticos gerados (somente no modelo primário).

3.2.2.1 Modelos Primários

O modelo primário utilizado para validação foi: modelo de Gompertz. Os resultados das validações do modelo Logístico e do modelo Logístico Modificado estão no Anexo A. O modelo de Baranyi não foi comparado, pois ainda há necessidade de confirmar quantos e quais são os parâmetros a serem otimizados.

Para o modelo primário, foram gerados dados obtidos pelos métodos de Gauss-Newton e de Levenberg-Marquardt e comparados com o *software Statistica*. Os dados comparados foram:

Estimativa dos parâmetros

- Os valores estimados dos parâmetros matemáticos: A, B e M;

- Erro Padrão (desvio padrão);
- t-value;
- Nível de significância (p-level);
- E intervalos de confiança (superior e inferior).

Análise de Variância

- Soma dos erros Residuais (RSS ou SSE)
- Média dos erros ao quadrado (MSE)

Modelo de Gompertz

Para validação do modelo de Gompertz foram utilizados os dados experimentais citados acima e gerados os resultados no *software Statística 7.0* e no aplicativo MBP que são impressos nas tabelas: Tabela 3-4,

Tabela 3-5 e Tabela 3-6.

Os resultados foram separados em dois grupos: o primeiro grupo de dados utilizando o método de Gauss-Newton (Tabela 3-3 e Tabela 3-4) e o segundo grupo de dados utilizando o método de Levenberg-Marquardt (Tabela 3-5). Para todos os dados obtidos foi calculado o erro relativo entre os dois *softwares* em estudo.

Os dados gerados pelo aplicativo desenvolvido foram truncados na quinta casa decimal. Alguns erros relativos são encontrados devido ao erro de truncamento e não somente pela diferença entre os resultados dos *softwares*.

Analisando os erros relativos calculados, observa-se que o erro máximo encontrado foi de 0.01579% para o desvio padrão do parâmetro matemático B e um erro médio menor que 0.008%. A Figura 3-37 apresenta os gráficos para cada um dos métodos de resolução e o resultado de ajuste utilizando-se o *software Statística*. Nota-se claramente que as curvas se sobrepõem devido à pequena diferença entre os valores estimados pelo aplicativo desenvolvido e pelo *Statística*.

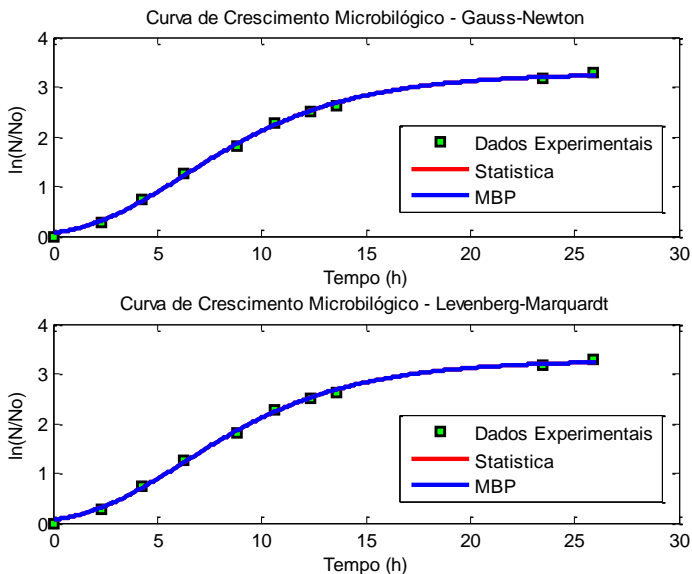


Figura 3-37 – Curvas de crescimento geradas pelos parâmetros primários calculados pelo Statistica 7.0 e o aplicativo em desenvolvimento

Tabela 3-3 – Comparação dos resultados do modelo de Gompertz utilizando o método de Gauss-Newton

	Valor Estimado	Erro Padrão	t-value (gl=7)	p-level	IC Down	IC Up
Statística 7.0						
A	3,273255	0,040489	80,84262	0,000000	3,177514	3,368997
B	0,219383	0,008215	26,70608	0,000000	0,199958	0,238807
M	6,211760	0,120125	51,71065	0,000000	5,927708	6,495811
MPB						
A	3,2733	0,040495	80,8319	0,000000	3,177510	3,369000
B	0,2194	0,008216	26,7027	0,000000	0,199960	0,238810
M	6,2118	0,120141	51,7038	0,000000	5,927710	6,495830
Erro Relativo						
A	0,00136%	0,01425%	0,01330%	0,00000%	0,00011%	0,00008%
B	0,00797%	0,01579%	0,01263%	0,00000%	0,00108%	0,00117%
M	0,00065%	0,01302%	0,01318%	0,00000%	0,00003%	0,00029%

Tabela 3-4 – Análise da Variância: Modelo de Gompertz utilizando o método de Gauss-Newton

	SSE	MSE
Statística 7.0		
A	0,01708	0,00244
MPB		
A	0,01709	0,00244
Erro Relativo		
A	-0,03380%	0,02473%

Tabela 3-5 - Comparação dos resultados do modelo de Gompertz utilizando o método de Levenberg-Marquardt

	Valor Estimado	Erro Padrão	t-value (gl=7)	p-level	IC Down	IC Up
Statística 7.0						
A	3,273256	0,040489	80,84377	0,000000	3,177515	3,368996
B	0,219382	0,008215	26,70536	0,000000	0,199957	0,238807
M	6,211757	0,120123	51,71160	0,000000	5,927711	6,495803
MPB						
A	3,273300	0,040495	80,8319	0,000000	3,177510	3,369000
B	0,219400	0,008216	26,7027	0,000000	0,199960	0,238810
M	6,211800	0,120142	51,7036	0,000000	5,927690	6,495820
Erro Relativo						
A	0,00135%	0,01566%	0,01473%	0,00952%	0,00017%	0,00011%
B	0,00813%	0,01327%	0,00998%	0,01865%	0,00151%	0,00112%
M	0,00070%	0,01575%	0,01542%	0,01291%	0,00035%	0,00027%

Tabela 3-6 - Análise da Variância: Modelo de Gompertz utilizando o método de Levenberg-Marquardt

	SSE	MSE
Statística 7.0		
A	0,01708	0,00244
MPB		
A	0,01709	0,00244
Erro Relativo		
A	-0,03380%	0,02473%

Observando a Figura 3-38 é possível verificar que o parâmetro matemático B possui o maior erro relativo em relação aos demais parâmetros (A e M). Isso ocorre devido à característica que cada parâmetro fornece à curva de crescimento. O parâmetro B é responsável pela inclinação da curva, conseqüentemente ao se usar um método de derivadas, os maiores desvios ocorrem principalmente em regiões da função otimizada, onde há a maior taxa de variação (maior inclinação).

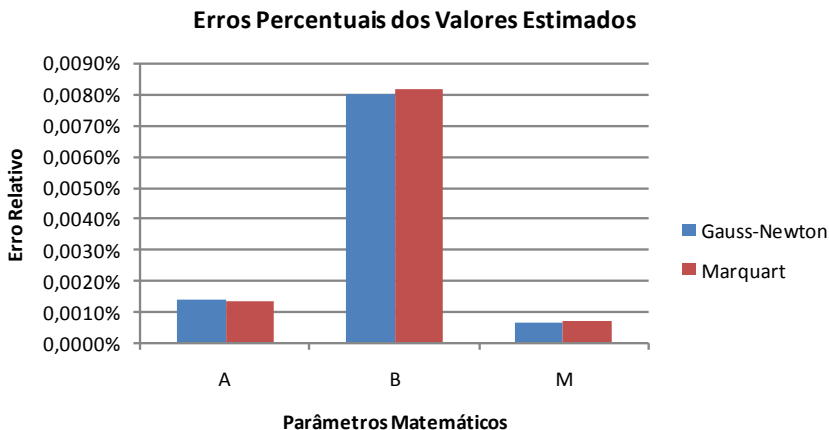


Figura 3-38 – Histograma de erros percentuais dos parâmetros matemáticos dos modelos primários

3.2.2.2 Modelos Secundários

Para a validação dos modelos secundários foram usados parâmetros primários obtidos de três diferentes temperaturas, apresentados na Tabela 3-7. O micro-organismo utilizado no experimento foi *Byssochlamys nivea*. Todos os dados foram retirados da literatura, e serão comparados entre os dados gerados pelo aplicativo em desenvolvimento e o software Statistica 7.0.

Tabela 3-7 – Parâmetros Primários variando conforme a temperatura

Parâmetro Físico (Temperatura [K])	A	μ	λ
310,65	2,5210	0,0695	293,1162
298,15	5,5637	0,1754	188,9992
301,15	6,4655	0,1363	111,6367

Modelo da Raiz Quadrada

Para efetuar os cálculos do modelo primário no *Statistica* 7.0, foi linearizada a equação do modelo da Raiz Quadrada e gerados os cálculos mostrados na Tabela 3-8.

Tabela 3-8 - Parâmetros Primários linearizados conforme o modelo da Raiz Quadrada

Raiz Quadrada			
Parâmetro Físico (Temperatura [K])	\sqrt{A}	$\sqrt{\mu}$	$\sqrt{\lambda}$
310,65	1,58777	0,26363	17,12064
298,15	2,35875	0,41881	13,74770
301,15	2,54273	0,36919	10,56583

Os resultados obtidos pelos dois aplicativos estão ilustrados na Tabela 3-9, onde são comparados os dois coeficientes da linearização: coeficiente angular e coeficiente linear, e para cada parâmetro primário obtidos nas isotérmicas de crescimento.

As curvas geradas pelos parâmetros estimados através dos dois aplicativos foram apresentadas na Figura 3-39. Observa-se que as retas praticamente se sobrepõem indicando que o erro relativo entre os resultados dos dois aplicativos é muito baixo.

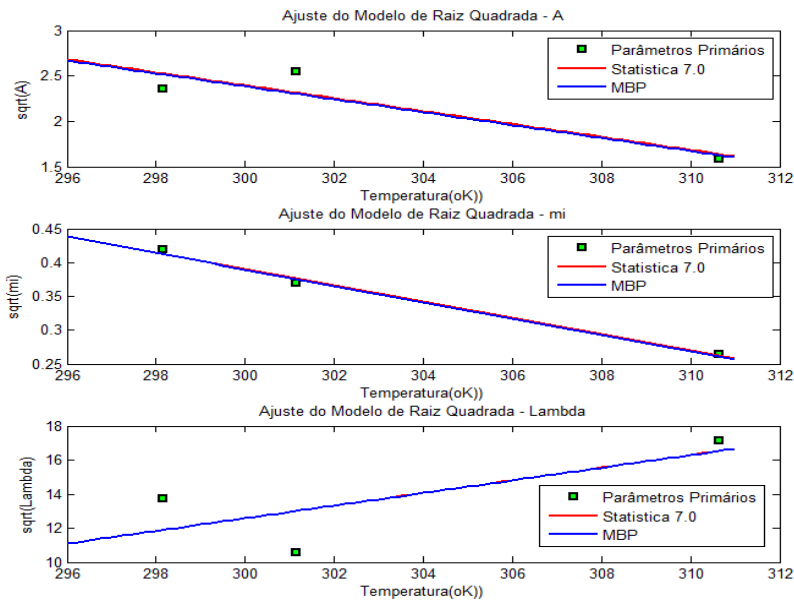


Figura 3-39 – Curvas de ajuste dos parâmetros primários pela temperatura usando o *Statística 7.0* e o aplicativo em desenvolvimento

Tabela 3-9 – Valores dos coeficientes de ajuste linear gerados pelo software *Statistica* e o aplicativo em desenvolvimento

	A	μ	λ
Statistica 7.0			
Coeficiente Linear	23,71881	4,02052	-98,83345
Coeficiente Angular	-0,07107	-0,01210	0,37138
MPB			
Coeficiente Linear	23,71540	4,01980	-98,80900
Coeficiente Angular	-0,07110	-0,01210	0,37130
Erro Relativo			
Coeficiente Linear	0,01437%	0,01782%	0,02475%
Coeficiente Angular	0,04679%	0,00428%	0,02074%

A Figura 3-40 mostra o histograma de erros relativos entre os coeficientes da reta obtidos no *Statistica* e no aplicativo desenvolvido. Pode se observar que o erro relativo de maior valor é do coeficiente angular do parâmetro microbiológico 'A', mesmo assim seu valor é muito baixo na ordem de 0,045 %.

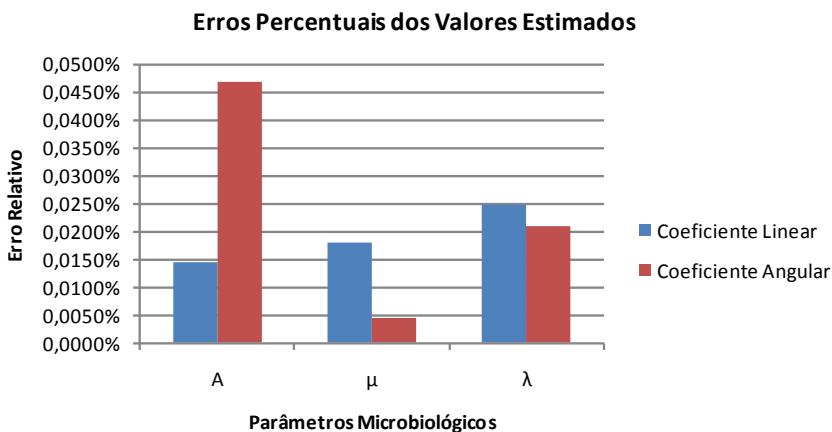


Figura 3-40 – Histograma de erros relativos dos coeficientes gerados pela linearização do modelo secundário.

4. CONCLUSÕES

O aplicativo desenvolvido é versátil, possui os principais modelos de microbiologia preditiva citados na literatura especializada e possui uma interface que permite o uso desses modelos de modo simples e seguro, pois os métodos de cálculo foram validados com um software comercial que possui excelente reputação no meio acadêmico. A confiabilidade do aplicativo permite seu uso para fins de pesquisa científica, na indústria e por órgãos de defesa sanitária da área de alimentos. Com ele, é possível simular o crescimento microbiano para as condições ambientais dinâmicas às quais um produto é submetido durante seu processamento, transporte e armazenamento. Também é possível simular o impacto que abusos de temperatura na vida-de-prateleira de produtos refrigerados.

Não há aplicativo similar com todas as funcionalidades presentes no aplicativo desenvolvido no presente trabalho. Deste modo, espera-se que o aplicativo desenvolvido seja usado extensivamente pelos profissionais da área de alimentos, pois estará disponível sem custos para os usuários interessados.

5. REFERÊNCIAS DA LITERATURA

ANTWI, M.; THEYS, T.E.; BERNAERTS, K.; VAN IMPE, J.F.; GEERAERD, A.H. Validation of a model for growth of *Lactococcus lactis* and *Listeria innocua* in a structured gel system: Effect of monopotassium phosphate. **International Journal of Food Microbiology**, v. 125, p. 320–329, 2008.

BARANYI, J.; ROBERTS, T.A. Mathematics of predictive food microbiology. **International Journal of Food Microbiology**, v.26 (1995) 199-218

BARANYI, J.; ROBERTS, T.A. A dynamic approach to predicting bacterial growth in food. **International Journal of Food Microbiology**, v. 23, p.277-294, 1994.

BARANYI, J.; ROBERTS, T.A. Predictive Microbiology - Quantitative Microbial, 2004.

BARANYI, J.; LE MARC, Y. DMFit Manual Version 2.0. Institute of Food Research, Norwich Research Park, UK.

BECK, K. AND ANDRES, C. “Extreme Programming Explained: Embrace Change,” Addison-Wesley Professional, 2004.

BECK K. “Test-Driven Development - By Example”. Addison-Wesley, 2003. 220 p.

BJÖRCK, A..*Numerical methods for least squares problems*. SIAM, Philadelphia, 1996).

BRUL, S.; GERWEN, S. V.; ZWIETERING, M.(Eds.). Modelling microorganisms in food. Boca Raton: CRC Press LLC, 2007. 294p.

BUCHANAN, R. L. Developing and distributing user-friendly application software. **Journal of Industrial Microbiology**, v.12, p.251-255, 1993.

BUCHANAN, R.L.; WHITING, R.C.; DAMERT, W.C. When is simple good enough: A comparison of the Gompertz, Baranyi, and three-phase linear models for fitting bacterial growth curves. **Food Microbiology**, v.14, p.313-326, 1997.

CARRASCO, E.; GARCÍA-GIMENO, R.; SESELOVSKY, R.; VALERO, A.; PÉREZ, F.; ZURERA, G.; TODD, E. Predictive Model of *Listeria Monocytogenes* Growth Rate Under Different Temperatures and Acids, p. 48-56, 2006.

CAYRÉ, M.E.; VIGNOLO, G.; GARRO, O. Selección de un Modelo Primario para Describir la Curva de Crecimiento de Bacterias Lácticas y *Brochothrix thermosphacta* sobre Emulsiones Cárnicas Cocidas. **Inf. tecnol.**, v.18, p.23-29, 2007.

COATES, K., Supply Chains To Singapore Department of Primary Industries, 475 Mickleham Rd, Attwood, Vic 3049. **Animal Production in Australia**, v.25, 349-356, 2004.

CHARALAMPOPOULOS, D., PANDIELLA, S., WEBB, C., Growth studies of potentially probiotic lactic acid bacteria in cereal-based substrates. **Journal of Applied Microbiology** 92, 851-860, 2002.

CORRADINI, M.G.; PELEG, M. Estimating non-isothermal bacterial growth in foods from isothermal experiments data. **Journal of Applied Microbiology**, v. 99, p. 187-200, 2005.

DALCANTON, F. **Produção, extração e caracterização de Poli(3-Hidroxitbutirato) por *Ralstonia Eutropha* em diferentes substratos** 2006. Dissertação (Mestrado em Engenharia de Alimentos) - Universidade Federal de Santa Catarina, Florianópolis, 2006.

DALGAARD, P., P. BUCH AND S. SILBERBERG. Seafood Spoilage Predictor - development and distribution of a product specific application software. **International Journal of Food Microbiology** 73, 227-233, 2002

ERKMEN, O. Predictive modelling of *Listeria monocytogenes* inactivation under high pressure carbon dioxide. **Food Science and Technology**, v. 33, p. 514-519, 2000.

FERREIRA, L.D. **Utilização da microbiologia preditiva na avaliação do crescimento de bactérias ácido lácticas em presunto fatiado**. 2004. Dissertação (Mestrado em Engenharia de Alimentos) - Universidade Federal de Santa Catarina, Florianópolis, 2004.

FOOD MICROBIOLOGY, A classification of models in predictive microbiology – a reply to K. R. Davey, v. 10, p. 175 – 177, 1993.

FUJIKAWA, H.; KAI, A.; MOROZUMI, S.; A new logistic model for *Escherichia coli* growth at constant and dynamic temperatures. **Food Microbiology**, v.21, p.501-509, 2004.

GEERAERD A. H.; VALDRAMIDIS V. P.; VAN IMPE J. F.; GlnaFIT, a freeware tool to assess non-log-linear microbial survivor curves A.H. **International Journal of Food Microbiology**, v. 102, p.95-105, 2005.

GIBBS, P.; GEKAS V.; Water Activity and Microbiological Aspects of Foods a Knowledge. Leatherhead Food Research Association, Randalls Road, Leatherhead, Surrey UK, University of Hania, Crete, Greece, 2001.

GIBSON, A.M.; BRATCHELL, N.; ROBERTS, T.A. Predicting microbial growth: growth responses of salmonellae in a laboratory medium as affected by pH, sodium chloride and storage temperature. **International Journal of Food Microbiology**, v.6, p.155 – 178, 1978.

GIL, M.M.; BRANDÃO, T.R.S.; SILVA, C.L.M. A modified Gompertz model to predict microbial inactivation under time-varying temperature conditions. **Journal of Food Engineering**, v.76, p.89-94, 2006.

GIL, M.M., PEREIRA, P.M., BRANDÃO, T.R.S., SILVA, C.L.M., KONDOYAN, A., VAN IMPE, J.F.M. AND JAMES, S. Integrated approach on heat transfer and inactivation kinetics of microorganisms on the surface of foods during heat treatments - software development. **Journal of Food Engineering**, 76(1): 95-103. 2006.

KOUTSOUMANIS K, STAMATIOU A, SKANDAMIS P, NYCHAS GJE. Development of a microbial model for the combined effect of temperature and pH on spoilage of ground meat, and validation of the model under dynamic temperature conditions. *App. Env. Microbiol.* 72 (1): 124-134, 2006

KOUTSOUMANIS, K. Predictive Modeling of the Shelf Life of Fish under Nonisothermal Conditions. **Applied and Environmental Microbiology**, v. 67, p.1821-1829, 2001.

MASSAGUER, Pilar Rodriguez. **Microbiologia dos processos alimentares**. São Paulo: Varela, 2005.

MCCONNELL, S. "Software Project Survival Guide". Microsoft Press, 1998. 304 p.

McMEEKIN, T.A.; ROSS T. Predictive microbiology: providing a knowledge-based framework for change management. **International Journal of Food Microbiology**, v.78, p. 133-153, 2002.

McMEEKIN, T.A., BARANYI, J., BOWMAN, J., DALGAARD, P., KIRK, M., ROSS, T.; SCHMID, S.; ZWIETERING, M.H. Information systems in food safety management. **International Journal of Food Microbiology**, 112, 181–194, 2006.

McClure, P. J., BLACKBURN, C. de W, COLE, M., B., CURTIS, P. S, JONES, J. E, LEGAN, J. D, OGDEN, I. D, PECK, M. W, ROBERTS, T. A, SUTHERLAND, J. P, WALKER, S. J. Review Paper. Modelling the growth, survival and death of microorganisms in foods: the UK Food Micromodel approach. **International Journal of Food Microbiology**, v.23, p.265-275, 1994.

MONOD, J. Recherches sur la croissance des. cultures bacteriennes. Hermann et Cie, Paris. 211 p., 1942.

NAKASHIMA, S.M.K.; ANDRÉ, D.S. ; FRANCO, B.D.G.M. Revisão: Aspectos Básicos da Microbiologia Preditiva. **Brazilian Journal of Food Technology**, v. 3, p.41-51, 2000.

NEUMEYER, K.; ROSS, T.; McMEEKIN, T.A. Development of a predictive model to describe the effects of temperature and water activity on the growth of spoilage pseudomonads. **International Journal of Food Microbiology**, v.38, p. 45-54, 1997.

NICOLAI, B.M.; VAN IMPE, J.F.; VERLINDEN, B.; MARTENS, T.; VANDEWALLE, J.; BAERDEMAEKER, J. Predictive modelling of surface growth of lactic acid bacteria in vacuum- packed meat. **Food Microbiology**, v.10, p. 229-238, 1993.

PANCIELLO, P. J. & QUANTICK, P. C. Application of food micro-model predictive software in the development of hazard analysis critical control point (HACCP) systems. **Food Microbiology**, v. 15, p.425-439, 1998.

PELEG, M.; COLE, M. B. Reinterpretation of microbial survival curves. **Crit. Rev. Food Sci. Nutr.** v. 38, p. 353-380, 1998.

PELEG, Micha. Advanced Quantitative Microbiology for Foods and Biosystems: models for Predicting Growth and Inactivation. CRS Series in Contemporary Food Science. Taylor & Francis Group. 2006.

RODRIGUES, R. C, Condições de cultura para a produção de Poli (3-hidroxibutirato) por *Ralstronia Eutropha* a partir de resíduo de industria de alimentos.

ROSA, A. J., CARVALHO, R, XAVIER, J. A D., Engenharia de Reservatório de Petróleo. Rio de Janeiro, Brasil. 1ª edição. Interciência -2006.

PEXARA, E.S.; METAXOPOULOS, J.; DROSINOS, E.H. Evaluation of shelf life of cured, cooked, sliced turkey fillets and cooked pork sausages - piroski - stored under vacuum and modified atmospheres at +4 and +10°C. **Meat Science**. v. 62, p. 33-43, 2002.

SARMENTO, C.M.P. Modelagem do Crescimento Microbiano e Avaliação Sensorial no estudo da Vida de Prateleira de mortadela e lingüiça defumada em armazenamento isotérmico e não isotérmico. 2006. Tese (Doutorado em Engenharia Química) - Universidade Federal de Santa Catarina, Florianópolis, 2006.

SCHWABER, K. "Agile Project Management with Scrum". Microsoft Press. 2004.

VAN IMPE, J.F.; BART, M.N.; SCHELLEKENS, M.; MARTENS, T.; BAERDEMAEKER, J.A. Predictive microbiology in a dynamic environment: a system theory approach. **International Journal of Food Microbiology**, v. 25, p.227-249, 1995.

VAN IMPE, J. F.; NICOLAI, B. M; MARTENS, T.; BAERDEMAEKER, J. & VANDEWALLE, J. Dynamic Mathematical Model to predict microbial growth and Inactivation during food processing. **Applied and Environmental Microbiology**, v.58, n.9, p.2901-2909, 1992

WHITING, R. C. Microbial modelling in foods. **Critical Reviews in Food Science and Nutrition**, 35, 467-494, 1995.

WIEGERS, K. E, "Creating a Software Engineering Culture". Dorset House, 1996.

ZWIETERING, M.H.; JONGENBURGER, I.; ROMBOUTS, F. M.; RIET, K. VAN 'T. Modeling of the Bacterial Growth Curve. **Appl Environ Microbiol**, p. 1875-1881, 1990.

6. ANEXO

Foram realizadas comparações dos modelos primários e secundários com o software Statistica 7.0.

MODELOS PRIMÁRIOS

Modelo Logístico

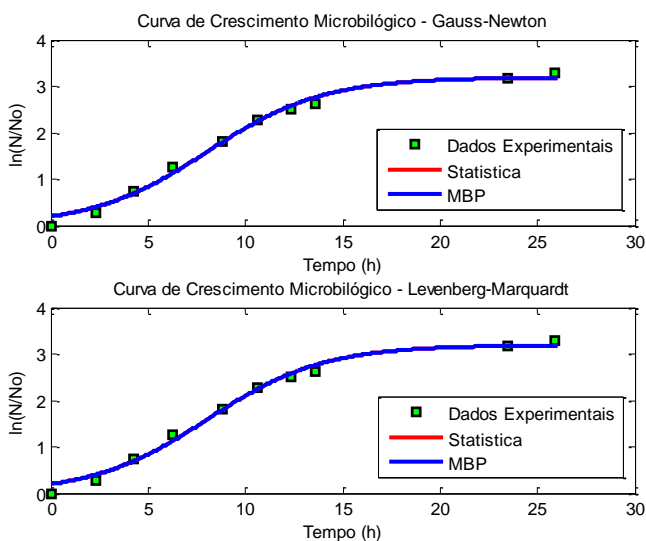


Figura 6-1 - Curvas de crescimento geradas pelos parâmetros primários calculados pelo Statistica 7.0 e o aplicativo em desenvolvimento

Tabela 6-1 – Parâmetros Estimados – Modelo Logístico utilizando o método de Gauss-Newton

		Valor Estimado	Erro Padrão	t-value (gl=7)	p-level	IC Down	IC Up
Statistica 7.0							
	A	3,190053	0,085153	37,46269	0,000000	2,988699	3,391408
	B	0,337387	0,027900	12,09292	0,000006	0,271415	0,403359
	M	8,054537	0,310818	25,91403	0,000000	7,319570	8,789504
MPB							
	A	3,190100	0,085157	37,4609	0,000000	2,988710	3,391400
	B	0,337400	0,027901	12,0925	0,000000	0,271420	0,403360
	M	8,054500	0,217515	12,4935	0,000000	2,203220	3,231810
ERRO RELATIVO							
	A	-0,00147%	-0,00494%	0,00474%	0,000000%	-0,00037%	0,00022%
	B	-0,00389%	-0,00526%	0,00374%	0,000000%	-0,00186%	-0,00030%
	M	0,00046%	42,89478%	107,42029%	0,000000%	232,221%	171,968%

Tabela 6-2 - Análise da Variância: Modelo Logístico utilizando o método de Gauss-Newton

	SSE	MSE
Statistica 7.0		
Resíduo	0,10047	0,01435
MPB		
Resíduo	0,10048	0,01435
ERRO RELATIVO		
Resíduo	-0,00659%	0,02327%

Tabela 6-3 - Parâmetros Estimados – Modelo Logístico utilizando o método de Levenberg-Marquardt

		Valor Estimado	Erro Padrão	t-value (gl=7)	p-level	IC Down	IC Up
Statistica 7.0							
	A	3,190053	0,085153	37,46269	0,000000	2,988699	3,391408
	B	0,337387	0,027900	12,09292	0,000006	0,271415	0,403359
	M	8,054537	0,310818	25,91403	0,000000	7,319570	8,789504
MPB							
	A	3,190100	0,085157	37,4609	0,000000	2,988710	3,391400
	B	0,337400	0,027901	12,0925	0,000000	0,271420	0,403360
	M	8,054500	0,217515	12,4935	0,000000	2,203220	3,231810
ERRO RELATIVO							
	A	-0,00147%	-0,00494%	0,00474%	0,00000%	-0,00037%	0,00022%
	B	-0,00389%	-0,00526%	0,00374%	0,00000%	-0,00186%	-0,00030%
	M	0,00046%	42,89478%	107,42029%	0,00000%	232,221%	171,968%

Tabela 6-4 - Análise da Variância: Modelo Logístico Modificado utilizando o método de Levenberg-Marquardt

	SSE	MSE
Statistica 7.0		
Resíduo	0,10047	0,01435
MPB		
Resíduo	0,10048	0,01435
ERRO RELATIVO		
Resíduo	-0,00659%	0,02327%

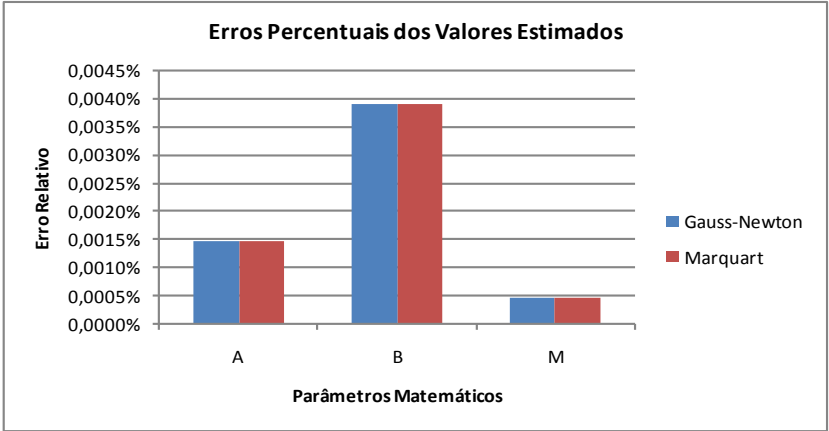


Figura 6-2 - Histograma de erros percentuais dos parâmetros matemáticos dos modelos primários – Modelo Logístico

Modelo Logístico Modificado

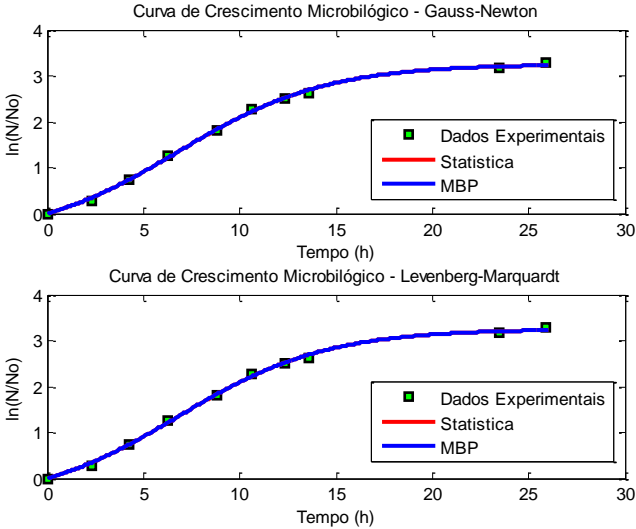


Figura 6-3 - Curvas de crescimento geradas pelos parâmetros primários calculados pelo Statistica 7.0 e o aplicativo em desenvolvimento

Gauss-Newton

Tabela 6-5 - Parâmetros Estimados – Modelo Logístico Modificado utilizando o método de Gauss-Newton

		Valor Estimado	Erro Padrão	t-value (gl=7)	p-level	IC Down	IC Up
Statistica 7.0							
	A	3,805124	0,113960	33,38987	0,000000	3,535651	4,074598
	B	0,260481	0,015135	17,21105	0,000001	0,224694	0,296268
	M	6,809656	0,257855	26,40885	0,000000	6,199926	7,419386
MPB							
	A	3,805100	0,113973	33,3863	0,000000	3,535660	4,074620
	B	0,260500	0,015137	17,2085	0,000000	0,224690	0,296270
	M	6,809700	0,257884	26,4059	0,000000	6,199920	7,419410
ERRO RELATIVO							
	A	0,00064%	-0,01100%	0,01077%	0,00000%	-0,00026%	-0,00054%
	B	-0,00729%	-0,01635%	0,01473%	0,00000%	0,00158%	-0,00052%
	M	-0,00065%	-0,01123%	0,01126%	0,00000%	0,00009%	-0,00032%

Tabela 6-6 - Análise da Variância: Modelo Logístico Modificado utilizando o método de Gauss-Newton

	SSE	MSE
Statistica 7.0		
Resíduo	0,01776	0,00254
MPB		
Resíduo	0,01776	0,00254
ERRO RELATIVO		
Resíduo	-0,00024%	-0,11273%

Tabela 6-7 - Parâmetros Estimados – Modelo Logístico Modificado utilizando o método de Levenberg-Marquardt

	Valor Estimado	Erro Padrão	t-value (gl=7)	p-level	IC Down	IC Up
Statística 7.0						
A	3,805124	0,113960	33,38987	0,000000	3,535651	4,074598
B	0,260481	0,015135	17,21105	0,000001	0,224694	0,296268
M	6,809656	0,257855	26,40885	0,000000	6,199926	7,419386
MPB						
A	3,8051	0,113979	33,3848	0,000000	3,535680	4,074660
B	0,2605	0,015137	17,2082	0,000000	0,224690	0,296260
M	6,8097	0,257897	26,4041	0,000000	6,199770	7,419310
ERRO RELATIVO						
A	0,00064%	-0,01627%	0,01532%	0,000000%	-0,00083%	-0,00152%
B	-0,00729%	-0,01635%	0,01642%	0,000000%	0,00158%	0,00286%
M	-0,00065%	-0,01627%	0,01800%	0,000000%	0,00251%	0,00103%

Tabela 6-8 - Análise da Variância: Modelo Logístico Modificado utilizando o método de Levenberg-Marquardt

	SSE	MSE
Statística 7.0		
Resíduo	0,01776	0,00254
MPB		
Resíduo	0,01776	0,00254
ERRO RELATIVO		
Resíduo	-0,00024%	-0,11273%

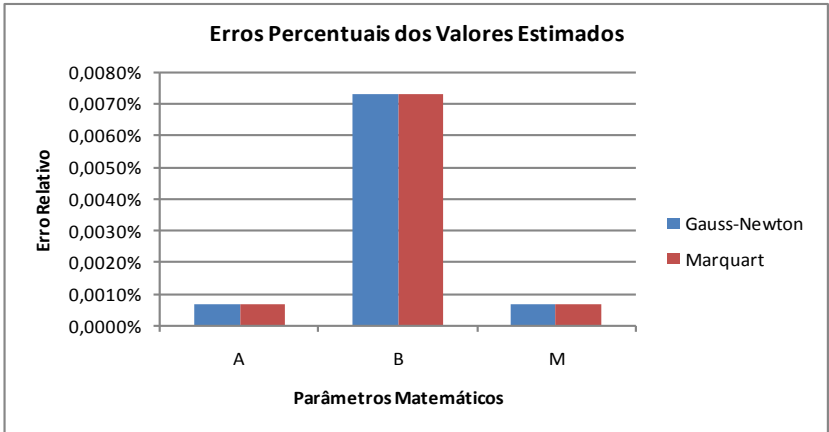


Figura 6-4 - Histograma de erros percentuais dos parâmetros matemáticos dos modelos primários – Modelo Logístico Modificado

MODELOS SECUNDÁRIOS

Modelos de Arrhenius

Tabela 6-9 - Valores dos coeficientes de ajuste linear gerados pelo software Statistica e o aplicativo em desenvolvimento

	A	μ	λ
Statistica 7.0			
Coeficiente Linear	-0,76022	-4,52931	6,72592
Coeficiente Angular	66,30199	70,20746	-44,34626
MPB			
Coeficiente Linear	-0,76030	-4,53930	6,72590
Coeficiente Angular	66,30450	70,20720	-44,34560
ERRO RELATIVO			
Coeficiente Linear	-0,01046%	-0,22017%	0,00037%
Coeficiente Angular	-0,00379%	0,00037%	0,00148%

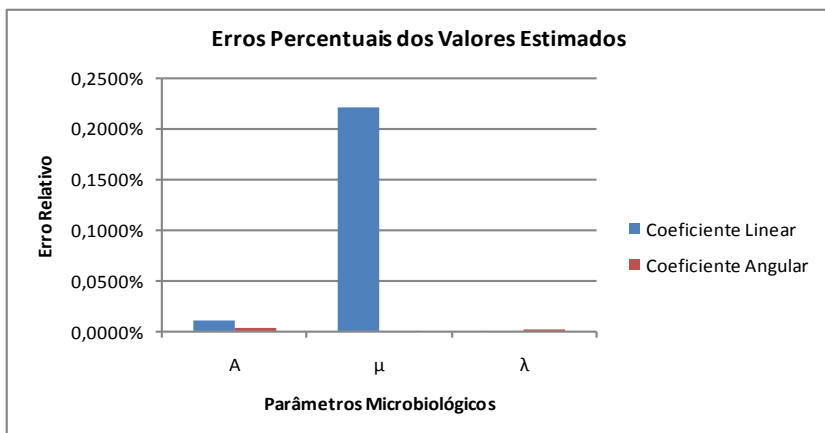


Figura 6-5 – Histograma de erros relativos dos coeficientes gerados pela linearização do modelo secundário – Modelo de Arrhenius

Modelo de Weibull

Tabela 6-10 - Valores dos coeficientes de ajuste linear gerados pelo software Statistica e o aplicativo em desenvolvimento

	A	μ	λ
Statistica 7.0			
Coefficiente Linear	126,64996	125,51711	-83,53606
Coefficiente Angular	-21,89952	-22,33751	15,53008
MPB			
Coefficiente Linear	128,16640	127,03210	-84,55380
Coefficiente Angular	-22,16490	-22,60260	15,70820
ERRO RELATIVO			
Coefficiente Linear	-1,18318%	-1,19261%	-1,20366%
Coefficiente Angular	-1,19732%	-1,17285%	-1,13392%

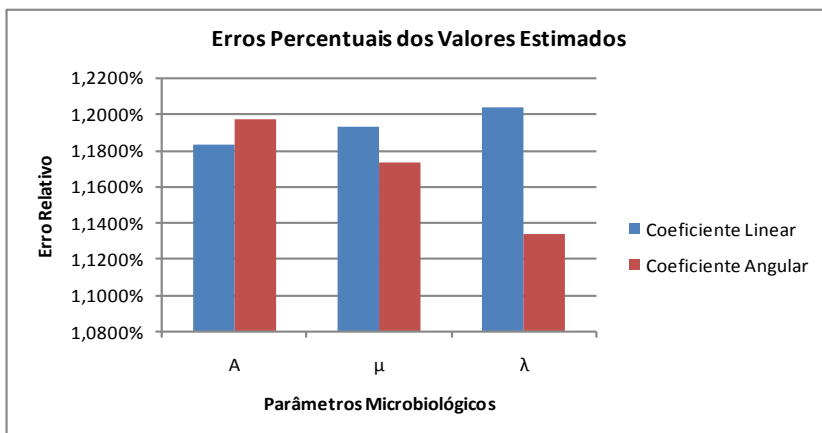


Figura 6-6 – Histograma de erros relativos dos coeficientes gerados pela linearização do modelo secundário