# Relatório descritivo do modelo cinético

Como versão inicial da estruturação matemática, tem-se um modelo que considera três componentes: substrato, biomassa e produto (*S, X* e *P*). Destaca-se aqui que, embora saiba-se que, na realidade, a composição do substrato seja variada, contendo diferentes elementos como açúcares, lipídios, proteínas, sólidos e minerais, na interpretação do modelo, o substrato é considerado como um componente único e homogêneo, caracterizando o substrato *S* como uma variável .

Essa mesma noção é aplicada à variável de estado que representa a biomassa *X*, visto que as três populações microbianas participantes do processo de digestão anaeróbica – acidogênicos, acetogênicos e metanogênicos- são agregados em uma única variável de estado.

Por fim, essa estruturação ignora os produtos intermediários formados no processo, focando apenas em descrever a geração de biogás no sistema. A Figura 1 apresenta o diagrama simplificado das reações consideradas no modelo, onde a adição de substrato e biomassa resulta na produção de mais biomassa e produto.

**Figura 1 –** Diagrama de reações simplificado.

Substrato

Biomassa

Biomassa

Produto

Dessa forma, a partir da consideração de um sistema isotérmico, homogêneo, em processo contínuo operado em um quimiostato, o balanço dos componentes do modelo é descrito pelas Equações 1, 2 e 3:

( 1 )

( 2 )

( 3 )

onde Q é a vazão no reator, V é o volume de operação no reator, os termos S0, X0 e P0 com o subscrito 0 representam a concentração de substrato, biomassa e produto na corrente de alimentação, enquanto as variáveis S1, X1 e P1 são a concentração de substrato, biomassa e produto dentro do reator. Já os termos e representam, respectivamente, o somatório das reações de geração e destruição de cada um dos componentes.

No balanço de substrato, desconsidera-se o termo de geração, enquanto o termo de destruição é descrito através da cinética de Monod, conforme apresentada na Equação 4, onde μ é a taxa específica de crescimento celular, μmax é o parâmetro que representa a taxa máxima de crescimento e Ks é uma constante de saturação de Monod, uma constante empírica que representa a concentração de substrato para qual .

( 4 )

Destaca-se aqui que apesar de extensivamente utilizada pela sua praticidade, a equação de Monod é uma formulação empírica, baseada na consideração de crescimento baseado no consumo de um único substrato limitante, descrevendo apenas as fases exponencial e estacionária do crescimento.

Sendo μ uma taxa específica, o crescimento celular pode ser modelado pela Equação 5, ao passo que essa formulação permite descrever a taxa de variação na concentração do substrato segundo a Equação 6, onde o parâmetro YX/S representa o rendimento de biomassa por unidade de substrato consumida.

( 5 )

( 6 )

Substituindo a Equação 6 como o termo de geração na Equação 1, obtém-se a Equação 7.

( 7 )

Por vez, no balanço de biomassa, optou-se por descrever a reação de geração em função de dois processos: o consumo de substrato e o decaimento celular. Tal qual feito no balanço anterior, a taxa de consumo de substrato foi modelada através da equação de Monod, substituindo a Equação 4 na Equação 5. Já o processo de decaimento de biomassa é descrito através de uma reação de primeira ordem que depende apenas da concentração de biomassa presente no reator, conforme apresentado na Equação 8.

( 8 )

Dessa forma, substituindo os processos de consumo de substrato e decaimento celular como termos de geração e destruição na Equação 2, respectivamente, obtém-se a Equação 9, que pode ser simplificada na Equação 10.

( 9 )

( 10 )

Por fim, no balanço do componente *P*, não são consideradas reações de destruição e descreve-se a geração de produto associado ao crescimento celular com a introdução de um parâmetro de rendimento de biogás por substrato consumido, *YP/S*, obtendo-se a Equação 7:

( 11 )

Com o intuito de diminuir o número de parâmetros necessários para descrição do modelo, considerou-se volume constante no reator, possibilitando a substituição da razão entre vazão e volume (*Q/V)* por um parâmetro de diluição (*D*), resultando no seguinte sistema de equações:

( 12 )

Dessa forma, a estruturação matemática do modelo trata-se de um sistema de equações diferenciais ordinárias para três componentes (*S*, *X* e *P*), utilizando seis parâmetros: *μmax*, *KS*, *YX/S*, *YP/S*, *kd* e *D*

A Tabela 1 apresenta a Matriz de Petersen para as reações ocorrentes no sistema. Dessa forma, pode-se escrever as equações de balanço de forma resumida tal qual na Equação 13. Por fim, a partir da consideração de concentrações nulas de biomassa e produto na corrente de entrada, pode-se reescrever o sistema de equações conforme apresentado na Equação 14.

Tabela 1 – Matriz de Petersen do sistema

|  |  |  |  |  |  |
| --- | --- | --- | --- | --- | --- |
| Componente → | i  j | 1 | 2 | 3 | Taxa *ρj* |
| Reação ↓ | S | X | P |
| Consumo de substrato | 1 |  |  |  |  |
| Decaimento de biomassa | 2 |  |  |  |  |

( 13 )

( 14 )

A Figura 2 apresenta um fluxograma conceitual detalhado das reações consideradas no sistema.

**Figura 2 –** Fluxograma de reações do modelo.

Substrato

Biomassa

Biomassa

Produto

Fonte: dos autores.

## Fluxo computacional

Para a construção do código do modelo proposto em linguagem *python*, utilizou-se das seguintes bibliotecas: numpy (1.24.3), scipy.integrate e scipy.optimize (1.10.1), matplotlib.pyplot (3.7.1), pandas (2.0.1), datetime (5.1), lmfit.minimize, lmfit.Parameters, lmfit.Parameter, lmfit.report\_fit e lmfit.fit\_report (1.2.1).

O fluxo computacional a ser seguido é definido na função *main*, sendo que inicialmente os dados experimentais são importados do arquivo .xlsx por meio da chamada da função ‘ajustarXlsx()’, a qual recebe como argumentos o caminho do arquivo e uma lista de parâmetros para tratamento dos dados (tempo inicial e final, número de pontos e algarismos significativos) e retorna um *DataFrame* com os dados importados do arquivo *excel* e de tempos.

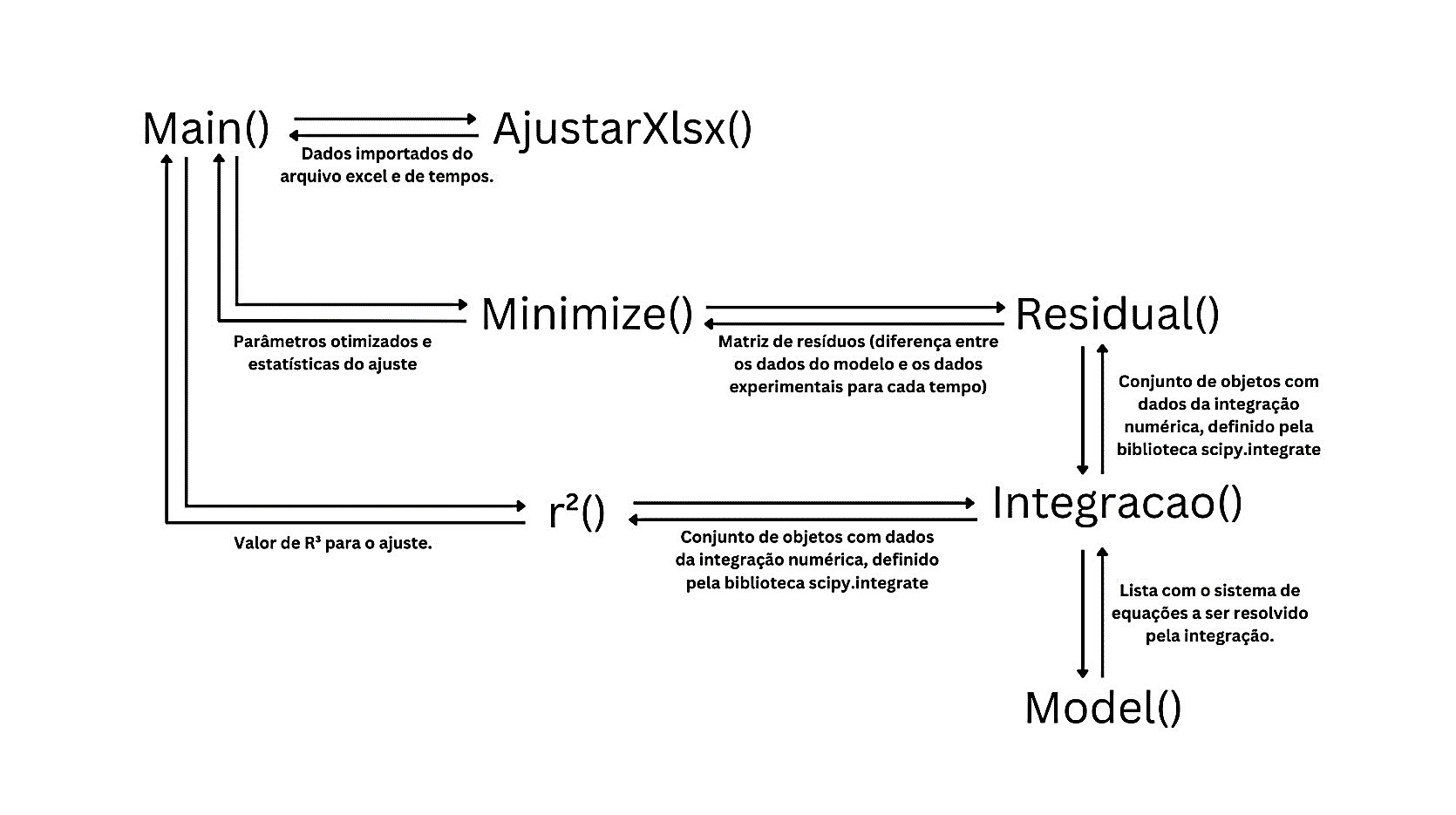
Em seguida são definidas as condições iniciais das variáveis dos balanços, o intervalo de integração, o método de integração a ser usado para a resolução do modelo matemático e as tolerâncias relativas e absolutas (*rtol* e *atol*). São também definidos os parâmetros para o ajuste das curvas do modelo, sendo esses armazenados em uma variável do tipo *Parameters* contendo, para cada parâmetro, seu nome, valor inicial e, para o caso de possibilidade de variação deste parâmetro, a faixa de valores aceitável – mínimo e máximo). Define-se então o método de minimização a ser utilizado e, a partir deste ponto, é realizada a modelagem propriamente dita para três casos distintos: otimizando os dados para produto, otimizando os dados para substrato e otimizando os dados para ambos produto e substrato. Os resultados do processo são então armazenados nas variáveis *resultProduto*, *resultSubstrato* e *resulGeral*, respectivamente.

Para a obtenção dos resultados do modelo é empregada a função *minimize()* da biblioteca *lmfit*, a qual recebe como parâmetro uma função objetivo, um objeto *Parameters* e outros argumentos opcionais como o método a ser usado na minimização dos dados, para retornar um objeto do tipo *MinimizeResult*, que contém os parâmetros otimizados e estatísticas de qualidade do ajuste. A função objetivo utilizada pela minimize é definida na função residual() e seus parâmetros são passados no argumento *args* na chamada de minimize(). Basicamente, a função residual realiza a chamada da função integração() que utiliza do módulo scipy.integrate.solve\_ivp para realizar a integração dos dados, recebendo como parâmetro a função *model()* que define o modelo matemático e os parâmetros de integração, para retornar à função residual a variável *solve\_ivp* (conjunto de objetos com dados da integração numérica, definido pela biblioteca *scipy.integrate*) salva na variável *model*, a função residual acessa então *model.y*, que contém os dados de resolução da integração para cada tempo. Por fim, é calculado a diferença entre os dados do modelo e os dados experimentais para cada tempo, sendo realizada a normalização do erro por meio de sua divisão pelo maior valor dos dados experimentais. Assim, a função residual retorna uma matriz de resíduos, que será efetivamente minimizada pela *minimize()*, permitindo a obtenção do objeto *MinimizeResult*. A função residual() e residual2() seguem os mesmos princípios, sendo que a residual2() foi apenas adaptada para calcular erro considerando tanto os dados de produto quanto de substrato, realizando o somatório de ambos.

O modelo matemático é definido na função *model()*, recebendo como parâmetros uma lista com as condições iniciais das variáveis dos balanços (x) e os valores de parâmetros do modelo. Assim, essa função registra os valores dos parâmetros para cada iteração e retorna a lista com o sistema de equações a ser resolvido pela integração.

Para cada um dos três casos também é calculado o valor de r², por meio da função *r2()*, a qual realiza novamente a integração do sistema de equações, agora a partir dos parâmetros retornados pela minimização e compara os dados obtidos com os dados experimentais. Tem-se ainda as funções de plotagem() e writeReport(), as quais servem para realizar a plotagem dos gráficos tanto com a curva obtida pela modelagem quanto com os dados experimentais e para atualizar automaticamente o arquivo .txt de relatório, armazenando as informações obtidas ao longo da execução para cada um dos casos analisados.

A Figura 3 apresenta um fluxograma básico do fluxo computacional e da comunicação entre a função principal Main() e as demais funções do código, com destaque para o retorno de cada função.

**Figura 3 –** Fluxo computacional do código para o modelo implementado em *Python*.

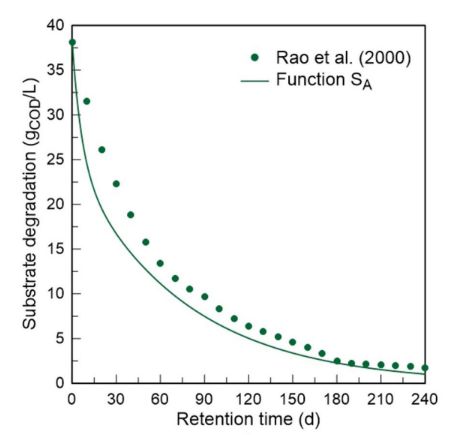
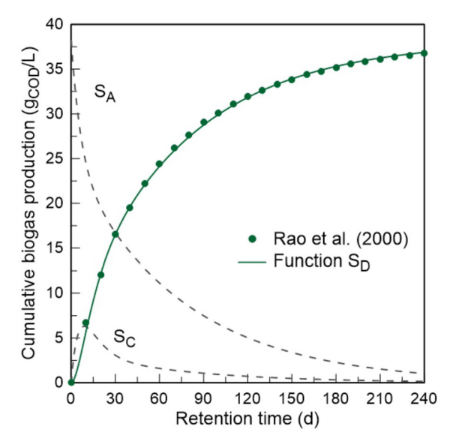
A saída final do ensaio com o código pode ser visualizada tanto em console quanto por meio do arquivo *report.txt*, o qual é atualizado automaticamente, gravando os dados do teste realizado. No console é possível acompanhar o *status* das chamadas das funções minimize e plotagem para cada variável a ser otimizada (Produto, Substratos ou ambas) e, ao final da execução da função minimize para cada caso é apresentado uma tabela com os valores dos parâmetros ajustados, a faixa de variação definida para os parâmetros ou se são parâmetros fixos (nesse caso valor ajustado será o valor especificado no objeto paras no código. Também é exibido o *stderr*, o erro padrão estimado para o valor de melhor ajuste, sendo que valores menores são melhores porque indicam que as observações estão mais próximas da curva ajustada. Além disso, é exibido a mensagem da função minimize indicando se ocorreu algum problema no ajuste, o valor de r² e o tempo de execução do ajuste.

No arquivo *report.txt* são salvos automaticamente ao final da execução do código (pela função *writeReport()*) os dados do ensaio como os dados e estatísticas de ajuste (método de minimização utilizado, número de funções avaliadas, número de pontos de dados, número de variáveis, qui-quadrado, qui-quadrado reduzido, índice de akaike e índice bayesiano), as variáveis ajustadas (valor final, incerteza e valor inicial testado) e as correlações entre as variáveis, caso seja possível a realização dessa estimativa.

## Otimização e simulação a partir de dados experimentais

Extraiu-se os dados do artigo de Gouveia et al. a partir da ferramenta gráfica WebPlotDigitizer. A Figura 3 apresenta os gráficos utilizados na extração.  
Em seu artigo, Gouveia et al. 2022 buscaram aplicar seu modelo de duas fases, descritas como processos de primeira ordem, nos dados experimentais de diferentes autores. Dentre esses extraiu-se os dados do ajuste do modelo de Gouveia et al. (2022) aos experimentos de Rao et al. (2000).

**Figura 3 –** Gráficos utilizados para extração de dados de produção de biogás e substrato.



a)

b)

Fonte: Gouveia et al. (2022).

Utilizou-se os dados extraídos para realizar processo de otimização de parâmetros com o modelo inicial proposto nesse trabalho. Considerando que, em seus experimentos, Rao et al. não mensuraram a concentração de biomassa, a otimização baseou-se na redução de uma função objetivo comparando os dados experimentais de produção de biogás e consumo de substrato com as predições das variáveis S e P do modelo.