



Tutorial experimental para determinação da estrutura tridimensional da enzima adenosina deaminase de *Edwardsiella tarda* por modelagem comparativa

*Experimental tutorial for determining the three-dimensional structure of the *Edwardsiella tarda* adenosine deaminase enzyme by comparative modeling*



Artigo

Submetido em 17.04.15 | Aceito em 25.06.15 | Disponível on-line em 01.01.16

Paulo Henrique Matayoshi Calixto*

Instituto Federal de Educação, Ciência e Tecnologia de Santa Catarina | *phcalixto@gmx.com

RESUMO

A *Edwardsiella tarda* é responsável pela edwardsielose. O tratamento da edwardsielose não é totalmente eficaz e nenhuma vacina comercial está disponível no mercado. Nesse sentido, o conhecimento dos processos bioquímicos da *E. tarda* é importante para a determinação de moléculas alvo, sobretudo de proteínas, para o tratamento ou eliminação da bactéria. Uma das proteínas capitais de *E. tarda* é a adenosina deaminase (EtAD), responsável pela deaminação da adenosina em inosina no processo de maturação do RNA transportador. Mesmo apresentando grande importância biológica em *E. tarda*, nenhum estudo relacionado à EtAD. Diante desse cenário, o objetivo desse trabalho é de gerar a estrutura tridimensional EtAD, bem como, oferecer ao leitor um tutorial facilitado sobre o uso do Modeller, o principal software de modelagem comparativa. Após a modelagem, verificou-se que as estruturas de EtAD e EcAD (adenosina deaminase de *Escherichia coli*) compartilham o mesmo padrão de enovelamento. A semelhança estrutural se deve ao percentual de resíduos compartilhados entre as duas proteínas, em torno de 77%. O emprego da modelagem molecular de proteínas pode auxiliar na escolha de novos alvos terapêuticos em potencial. Ademais, este trabalho fornece ao leitor uma fácil e compreensível introdução sobre o processo de modelagem comparativa.

Palavras-chaves: *Edwardsiella tarda*, adenosina deaminase, Modelagem comparativa, Modeller.

ABSTRACT

The *Edwardsiella tarda* is responsible for edwardsiellosis. Treatment of edwardsiellosis is not completely effective and no commercial vaccine is available. In this sense, knowledge of biochemical processes of *E. tarda* is important for the determination of target molecules, particularly proteins, for treatment or elimination of the bacterium. One of the proteins of *E. tarda* is adenosine deaminase (EtAD), responsible for the deamination of adenosine to inosine in transfer RNA maturation process. Even with great biological importance in *E. tarda*, no study was performed with EtAD. Given this scenario, the objective of this work is to generate EtAD three-dimensional structure, as well as offer the reader a tutorial on using facilitated Modeller, the main comparative modeling software. After modeling, it was found that the structures of EcAD (*Escherichia coli* adenosine deaminase) and EtAD share the same pattern of folding. The structural similarity is due to the percentage of residues shared between the two proteins, around 77%. The use of molecular modeling of proteins can help in choosing potential new therapeutic targets. In addition, this paper provides the reader with an easy and understandable introduction to the comparative modeling process.

Keywords: *Edwardsiella tarda*, adenosine deaminase, comparative modeling, Modeller.

1.Introdução

A *Edwardsiella tarda*, um bacilo gram-negativo, é uma bactéria responsável por uma doença hemorrágica, que acomete tanto peixes de água doce, quanto marinhos. A edwardsiiose, como é conhecida, gera grandes perdas econômicas em todo o planeta, seja em criadouros artificiais ou em ambientes naturais (HU *et al.*, 2014). Eventualmente, a *E. tarda* pode infectar humanos, suscitando um quadro de gastroenterite de diferentes graus, porém de fácil resolução (JANDA; ABBOTT, 1993).

Apesar da grande importância econômica, a *E. tarda* é uma bactéria pouco estudada. Embora existam vacinas experimentais, nenhuma delas é comercializada, indicando que seu papel protetivo ainda não atingiu o patamar desejado (NELSON *et al.*, 2009). O uso de antibióticos na água acarreta uma série de transtornos, especialmente quanto à resistência aos fármacos (ZHENG *et al.*, 2012). Dessa maneira, é mister o desenvolvimento de novas vacinas e/ou fármacos que apresentem resultados satisfatórios. Nesse sentido, o conhecimento da maquinaria bioquímica da bactéria é de suma importância na escolha de macromoléculas, tais como as proteínas, como alvos para o tratamento e/ou profilaxia da infecção. Uma dessas moléculas é a adenosina deaminase, responsável pela maturação da molécula de RNA transportador (tRNA), através da deaminação da adenosina em inosina, essencial ao estabelecimento e manutenção de qualquer organismo vivo (LUO; SCHRAMM, 2008). Mesmo sendo uma molécula potencialmente importante como alvo para medidas terapêuticas e/ou profiláticas, nenhum estudo sobre a adenosina deaminase ainda foi empreendido em *E. tarda*.

A caracterização de uma proteína é melhor conduzida de posse da estrutura tridimensional (NADEAU; CARLSON, 2012). A determinação da estrutura proteica pode ser alcançada por vários métodos experimentais, como é o caso da difração de raios-X e a ressonância nuclear magnética, bem como, por métodos teóricos, também referidos como *in silico*. Destes, se destaca a modelagem comparativa ou por homologia, como o método teórico mais acurado para a determinação de estruturas proteicas (SCHMIDT *et al.*, 2014).

A modelagem comparativa se baseia na observação geral, onde sequências de aminoácidos evolutivamente relacionadas apresentam estruturas tridimensionais similares, sendo a conservação em nível estrutural maior que a conservação da estrutura primária (sequência de aminoácidos) (WILTGEN; TILZ, 2009). Consequentemente, o modelo estrutural de uma proteína pode ser construído a partir de estruturas concebidas por métodos experimentais, cujas sequências de aminoácidos compartilham identidade estatisticamente significativa (CALIXTO *et al.*, 2013). Tradicionalmente, o processo de modelagem comparativa compreende uma série de estágios consecutivos e repetidos iterativamente, até que seja obtido um modelo estrutural com parâmetros satisfatórios (UNAJAK *et al.*, 2014). A grosso modo, esses passos se resumem a: i) busca por estruturas (proteína molde) relacionadas à proteína-alvo; ii) alinhamento entre as sequências de aminoácidos das proteínas alvo e molde; iii) construção do modelo. Etapas subsequentes à construção do modelo devem ser consideradas, como o refinamento da estrutura, isto é, modelagem dos *loops* e dos rotâmeros, minimização da energia e validação da estrutura (MARTI-RENO, *et al.*, 2000).

O Modeller (SALI *et al.*, 1995), é de longe o *software* mais popular e eficiente de modelagem comparativa. No entanto, muitos usuários, especialmente aqueles da área de Ciências Biológicas e da Saúde, apresentam dificuldades em se introduzirem no uso do Modeller. Isso se deve ao fato do programa não apresentar interface gráfica, sendo cada comando inserido através de linhas de comando num terminal, exigindo prévio conhecimento de informática, sobretudo na linguagem de programação Python. Diante desse cenário, os objetivos deste trabalho são de proporcionar aos leitores, um tutorial detalhado e facilitado sobre o uso do Modeller, bem como, de realizar a determinação estrutural da adenosina deaminase de *E. tarda*.

2. Materiais e métodos

2.1 Obtenção da sequência e determinação estrutural da EtAD

A estrutura da adenosina deaminase de *E. tarda* (EtAD) (GenBank: WP_015683243.1) (TEKEDAR *et al.*, 2013) foi determinada por modelagem comparativa, através do método de restrições espaciais, empregando o *software* Modeller 9.13v (SALI *et al.*, 1995). Inicialmente, foram gerados 10 modelos, sendo o que apresentou melhor *score* DOPE, eleito para este trabalho.

2.2 Aporte de *software* e de *hardware*

Neste trabalho foram utilizados os seguintes *softwares*: UCSF Chimera (PETTERSEN *et al.*, 2004) disponível em: <<https://www.cgl.ucsf.edu/chimera/>>, GNUPLOT, disponível em: <<http://www.gnuplot.info/>> e Modeller 9.13v, disponível em <<http://salilab.org/modeller/>>. O Modeller é um

software gratuito para fins acadêmicos. No entanto, é necessário que o usuário se cadastre no *website* do programa para obter a licença de uso. Todos os *softwares* foram executados no sistema operacional Linux (Ubuntu 13.10v), devido sua robustez, estabilidade e eficiência. Caso o usuário tenha dificuldade em instalar os programas, o mesmo poderá instalá-los com o auxílio do Central de Programas do Ubuntu. A configuração de *hardware*, no qual o estudo foi realizado é a seguinte: processador Intel® Core™ i7-4790, 32 Gb de memória RAM (1600 MHz), GPU VGA GigaByte GeForce GTX660 Ti 2048MB, HD com capacidade de 1 Tb.

3. Resultados e discussão

3.1 Busca por estruturas relacionadas à EtAD

A primeira etapa da modelagem comparativa é a busca por estruturas que possam servir como molde (*template*). Diferentes abordagens podem ser adotadas para esse fim. No entanto, todas se resumem a uma exaustiva pesquisa no *Protein Data Bank* (PDB). A abordagem mais simplificada para a busca de estruturas molde é através de alinhamento local, via BLAST (*Basic Local Alignment Search Tool* - <<http://www.ncbi.nlm.nih.gov/blast/>>), utilizando como entrada a sequência a ser modelada contra um banco de dados restrito, no caso, “*Protein Data Bank proteins (pdb)*”. Contudo, neste trabalho, a busca por estruturas molde será empreendida pelo Modeller 9.13v, não apenas como demonstrativo de todo o aporte do *software*, mas também para propiciar ao usuário, um contato com as principais ferramentas do programa.

Este estudo será conduzido com a determinação da estrutura terciária da EtAD. A sequência de aminoácidos da EtAD está disponível no GenBank (<<http://www.ncbi.nlm.nih.gov/genbank/>>),

sob o número de acesso WP_015683243.1 (TEKEDAR *et al.*, 2013). A sequência recuperada no *GenBank* está no formato FASTA (Fig. 1A). No entanto, o formato FASTA não é admitido como entrada pelo Modeller, sendo necessário a conversão da sequência de aminoácidos para o formato PIR (Fig. 1B). A

conversão pode ser feita através do *website* <<http://genome.ncbi.nlm.nih.gov/tools/reformat.html>>. Após a conversão para o formato PIR, a sequência deverá ser transferida para o editor de texto Gedit, nomeada como “EtAD” e salva no formato .ali.

Figura 1. Diferentes formatos de sequência para EtAD. A) Sequência em formato FASTA. O nome da sequência é precedido pelo sinal “>”, que indica a identificação da sequência. Imediatamente abaixo, localiza-se o início da sequência. B) Sequência em formato PIR. A identificação da sequência é precedida por “>P1”. A linha imediatamente abaixo da identificação traz informações acerca da sequência. O fim da sequência é determinado pelo asterisco.

A) >EtAD
MTQRDDSDWMRHALQLAEKARAQGEVPVGAVLVLNDRVIGEGWNRPIRHDPTAHAEIMALQQGGAIMQN
YRLLEATLYVTLEPCVMCAGAMVHSRIRRLVYGAPDLKTGAAGSLDLVGHGPMNHRIEVCGGVLADACA
AQLSDFFRQRRRAQKRAERDARRRAERDDAE

B) >P1;EtAD
sequence:EtAD:::::0.00: 0.00
MTQRDDSDWMRHALQLAEKARAQGEVPVGAVLVLNDRVIGEGWNRPIRHDPTAHAEIMALQQGGAIMQN
YRLLEATLYVTLEPCVMCAGAMVHSRIRRLVYGAPDLKTGAAGSLDLVGHGPMNHRIEVCGGVLADACA
AQLSDFFRQRRRAQKRAERDARRRAERDDAE*

Para iniciar o processo de busca por estruturas relacionadas, o usuário deverá adquirir o arquivo pdb_95 no *website* <<https://salilab.org/modeller/supplemental.html>>. O arquivo pdb_95 reúne as sequências de aminoácidos disponíveis no PDB que compartilham até 95% de identidade entre si, ou seja, um banco de dados não redundante. Outra particularidade do banco de dados pdb_95 é que todas as sequências estão no formato PIR. Os comandos do Modeller são realizados por intermédio de *scripts* escritos em Python, ou seja, de arquivos que contêm as informações acerca de cada comando. O *script* que iniciará a busca por estruturas relacionadas está contido na figura 2 e deverá ser transcrito para o editor de texto, nomeado como “script1” e salvo com a extensão .py. De posse dos três arquivos mencionados até aqui (EtAD.ali; pdb_95.pir; script1.py), o usuário deverá criar a pasta

“Tutorial Modeller” e colocá-los na pasta recém-criada. O script1 deverá ser executado a partir de um terminal endereçado ao diretório “Tutorial Modeller”, através do comando “Mod9.13 script1.py”. A entrada “Mod9.13” inicia o *environment* do Modeller e é exclusiva para cada versão, por exemplo, na versão 9.7 o *environment* é carregado pela entrada “Mod9.7”. Após a execução do script1, serão gerados 4 arquivos: i) script1.log, arquivo de registro que contém todas as informações sobre a execução do script1; ii) build_profile.ali, apresenta o alinhamento entre as sequência alvo e as sequências do pdb_95; iii) build_profile.prf, possui informações importantes a respeito dos alinhamentos, tais como percentual de identidade entre os resíduos e *e-value* (valor de E); iv) pdb_95.bin, conversão do arquivo pdb_95.pir para o formato binário.

Figura 2. Script 1, escrito em Python. As informações desta figura deverão ser fielmente transcritas para o bloco de notas e, posteriormente, o arquivo deverá ser salvo como “script2.py”. A execução do script depende da presença dos arquivos “pdb_95.pir” e “EtAD.ali”. As cores azul, rosa e preto são padrão do Python.

```
from modeller import *

log.verbose()
env = environ()

#-- Preparacao dos arquivos de entrada

#-- Leitura no banco de dados de sequencia
sdb = sequence_db(env)
sdb.read(seq_database_file='pdb_95.pir', seq_database_format='PIR',
        chains_list='ALL', minmax_db_seq_len=(30, 4000), clean_sequences=True)

#-- Transcrever o banco de dados de sequencia no formato binario
sdb.write(seq_database_file='pdb_95.bin', seq_database_format='BINARY',
        chains_list='ALL')

#-- Leitura do banco de dados binario
sdb.read(seq_database_file='pdb_95.bin', seq_database_format='BINARY',
        chains_list='ALL')

#-- Leitura da sequencia/alinhamento alvo
aln = alignment(env)
aln.append(file='EtAD.ali', alignment_format='PIR', align_codes='ALL')

#-- Conversao da sequencia/alinhamento de entrada
# profile format
prf = aln.to_profile()

#-- Pesquisa no banco de dados de sequencia e escolha das sequencias homologas
prf.build(sdb, matrix_offset=-450, rr_file='${LIB}/blosum62.sim.mat',
        gap_penalties_1d=(-500, -50), n_prof_iterations=1,
        check_profile=False, max_aln_evalue=0.01)

#-- Escrever o arquivo de saida do perfil no formato texto
prf.write(file='build_profile.prf', profile_format='TEXT')

#-- Converter o arquivo de saida do profile para o formato alinhamento
aln = prf.to_alignment()

#-- Escrever o arquivo de saida do arquivo de alinhamento
aln.write(file='build_profile.ali', alignment_format='PIR')
```

3.2 Seleção da estrutura molde

A seleção da estrutura molde envolve uma série de considerações, tais como, percentual de resíduos idênticos e similares, cobertura do molde, confiabilidade do alinhamento (expresso pelo *e-value*), entre outros (DAI *et al.*, 2014). Após executar o script1, o usuário deverá abrir o arquivo “build_profile.prf”, através do Gedit, para

acessar as informações sobre o alinhamento. Sobre o arquivo “build_profile.prf” (Fig. 3), chama a atenção as informações contidas nas colunas de número 2, 11 e 12, as quais correspondem ao identificador da proteína no PDB, ao percentual de identidade entre a sequência de EtAD e sequência do PDB e à confiabilidade do alinhamento (quanto mais próximo de zero, menor é a possibilidade do alinhamento ter ocorrido ao acaso),

respectivamente. O percentual de resíduos idênticos entre a sequência alvo e o molde é diretamente proporcional à qualidade do modelo gerado (MOBAREC *et al.*, 2009). No entanto, do ponto de vista prático, sequências alvo e

molde que compartilham pelo menos 25% de identidade podem gerar modelos de boa qualidade (VALLAT *et al.*, 2008).

Figura 3. Representação do “build_profile.prf”. As sequências foram omitidas para melhor apresentação dos dados. A segunda coluna se refere ao código de acesso da proteína no PDB. A décima coluna aponta a cobertura do *template* sobre a proteína alvo. Por último, a décima primeira e décima segunda colunas, correspondem ao percentual de resíduos idênticos e ao *e-value*, respectivamente. As proteínas marcadas em vermelho se referem às potenciais candidatas a proteínas molde.

```
# Number of sequences:      17
# Length of profile :      170
# N_PROF_ITERATIONS :      1
# GAP_PENALTIES_1D :      -500.0      -50.0
# MATRIX_OFFSET :      -450.0
# RR_FILE :      ${LIB}/blosum62.sim.mat

1 EtAD S 0 170 1 170 0 0 0 0. 0.0
2 2a8nA X 1 130 9 133 6 130 125 46. 0.0
3 2b3jA X 1 151 5 152 3 150 148 47. 0.0
4 2b3zA X 1 359 9 147 5 137 125 36. 0.56E-08
5 3dh1A X 1 162 2 149 7 162 146 31. 0.0
6 2g6vA X 1 355 3 150 5 139 133 36. 0.23E-08
7 2g84A X 1 189 10 146 30 177 135 34. 0.62E-07
8 2hvwA X 1 147 8 102 7 114 93 28. 0.78E-05
9 2hvxvA X 1 349 9 155 9 151 135 36. 0.74E-09
10 2nx8A X 1 168 9 155 14 160 147 41. 0.0
11 3ocqA X 1 146 5 148 2 145 144 76. 0.0
12 1p6oA X 1 156 1 147 4 151 146 29. 0.12E-08
13 1vq2A X 1 173 28 102 24 129 75 29. 0.89E-03
14 1wkqA X 1 158 9 113 8 113 105 37. 0.0
15 1wwrA X 1 154 9 151 9 150 142 39. 0.0
16 1z3aA X 1 156 9 153 10 154 145 77. 0.0
17 3zpgA X 1 354 4 136 4 135 124 35. 0.0
```

Baseado no percentual de identidade, *e-value* e cobertura (Fig. 3), sete estruturas foram selecionadas para a próxima etapa desse estudo. As proteínas selecionadas são as seguintes e estão representadas pelo número de acesso no PDB: 2B3J; 3DH1; 2NX8; 3OCQ; 1WKQ; 1WWR e 1Z3A. Como esperado, todas as sequências selecionadas correspondem à deaminases de diferentes espécies. Todas as estruturas mencionadas deverão ser recuperadas no PDB e armazenadas na pasta “Tutorial Modeller”. O próximo passo será comparar as estruturas e avaliar qual estrutura é mais apropriada para ser empregada como molde. Antes de proceder com a avaliação, o usuário deverá transcrever fielmente as informações contidas na figura 4 para o editor de texto, nomear o arquivo como “script2” e salvá-lo com

a extensão .py. Para realizar a comparação o usuário deverá executar o script2 através da seguinte linha de comando “**mod9.13 script2.py**”.

A execução do script2 conduzirá a criação do arquivo “script2.log”. Este arquivo apresenta um comparativo entre as proteínas candidatas à estrutura molde, além de gerar um dendograma (Fig. 5). As informações contidas no “script.log” e no “build_profile.prf” auxiliarão o leitor na escolha da proteína. Como demonstrado no dendograma as proteínas 3OCQ e 1Z3A são praticamente idênticas. No entanto 1Z3A apresenta maior cobertura de sequência com EtAD do que 3OCQ, além de apresentar percentual de identidade ligeiramente maior. Dessa forma, a estrutura 3OCQ será excluída do estudo. A resolução da estrutura,

junto com o percentual de identidade, *e-value* e cobertura, é um dos pontos capitais que contribuem para a qualidade final do modelo. Nesse sentido, a estrutura 3DH1 também foi excluída do estudo por apresentar baixa resolução (2,8 Å), quando comparada às demais. Mesmo apresentando resolução satisfatória, as estruturas 1WKQ, 1WWR, 2NX8, 2B3J foram

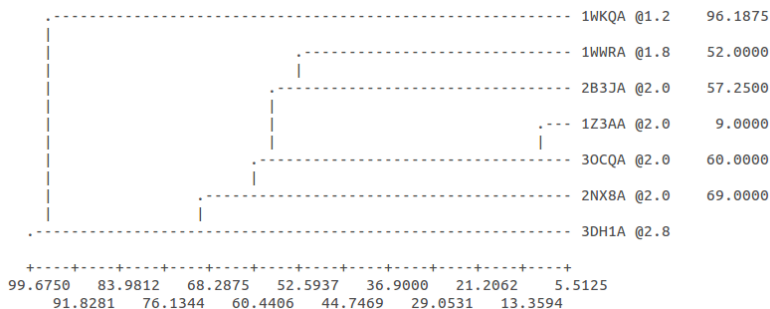
excluídas do estudo por apresentarem baixo percentual de identidade e/ou cobertura com EtAD. Consequentemente, a estrutura eleita como proteína molde foi 1Z3A, correspondente à adenosina deaminase de *Escherichia coli* (EcAD), que apresenta 77% de identidade com EtAD (LUO, M.; SCHRAMM, 2008).

Figura 4. Script 2, escrito em Python. As informações desta figura deverão ser fielmente transcritas para o bloco de notas e, posteriormente, o arquivo deverá ser salvo como “script2.py”.

```
from modeller import *

env = environ()
aln = alignment(env)
for (pdb, chain) in (('1WKQ', 'A'), ('1WWR', 'A'), ('1Z3A', 'A'),
                    ('2B3J', 'A'), ('2NX8', 'A'), ('3DH1', 'A'),
                    ('3OCQ', 'A')):
    m = model(env, file=pdb, model_segment=('FIRST:'+chain, 'LAST:'+chain))
    aln.append_model(m, atom_files=pdb, align_codes=pdb+chain)
aln.malign()
aln.malign3d()
aln.compare_structures()
aln.id_table(matrix_file='family.mat')
env.dendrogram(matrix_file='family.mat', cluster_cut=-1.0)
```

Figura 5. Dendograma com as sete estruturas candidatas a *template*. O dendograma pode ser encontrado no arquivo “script2.log”. A resolução de cada estrutura é indicada pelo caractere “@” em Å (ångström).



3.3 Alinhamento da EtAD com a estrutura molde

Tendo em vista que o Modeller se baseia nas restrições espaciais para a construção do modelo, a qualidade do alinhamento está diretamente relacionada com o sucesso da modelagem (ESWAR *et al.*, 2008). Preferencialmente, o alinhamento entre a sequência alvo e a estrutura molde deve ser realizado pelo comando align2d do Modeller. O align2d é diferente dos demais métodos de alinhamento, pois considera as informações estruturais do modelo na construção do

alinhamento. Para realizar o alinhamento entre EtAD e 1Z3A, o usuário deverá transcrever fielmente as informações contidas na figura 6 para o editor de texto, nomear o arquivo como “script3” e salvar com a extensão .py na pasta “Tutorial Modeller”. Para realizar a comparação o usuário deverá executar o script3 através da seguinte linha de comando “**mod9.13 script3.py**”. Após a execução do script3, será gerado o arquivo EtAD-1Z3A.pap (Figura 7). Este arquivo apresenta o alinhamento entre a sequências de EtAD e 1Z3A.

Figura 6. Script 3, escrito em Python. As informações desta figura deverão ser fielmente transcritas para o bloco de notas e, posteriormente, o arquivo deverá ser salvo como “script3.py”.

```
from modeller import *

env = environ()
aln = alignment(env)
mdl = model(env, file='1Z3A', model_segment=('FIRST:A','LAST:A'))
aln.append_model(mdl, align_codes='1Z3AA', atom_files='1Z3A.pdb')
aln.append(file='EtAD.ali', align_codes='EtAD')
aln.align2d()
aln.write(file='EtAD-1Z3A.ali', alignment_format='PIR')
aln.write(file='EtAD-1Z3A.pap', alignment_format='PAP')
```

Figura 7: Alinhamento entre os resíduos de EtAD e 1Z3A (EcAD) realizado pelo align2d. Os asteriscos apontam os resíduos idênticos.

```
_aln.pos      10      20      30      40      50      60
1Z3AA      SEVEFSHEYWMRHALTLAKRAWDEREVPVGAVLVHNNRVIGEGWNRPIGRHDPTAHAEIMALRQGGLV
EtAD      -MTQRDSDWMRHALQLAEKARAQGEVPVGAVLVNDRVIGEGWNRPIGRHDPTAHAEIMALQGGAI
_consrvd      ***** ** * ***** * ***** * ***** * ***** *

_aln.p      70      80      90      100     110     120     130
1Z3AA      MQNYRLIDATLYVTLEPCVMCAGAMHSRIGRVVFGARDAKTGAAGSLMDVLHHPGMNHRVEITEGIL
EtAD      MQNYRLLEATLYVTLEPCVMCAGAMVHSRIRRLVYGAPDLKTGAAGSLDVLGHPGMNHRIEVCGGVL
_consrvd      ***** ***** ***** * * * * * ***** * * * * *

_aln.pos     140     150
1Z3AA      ADECAALLSDFFRMRQEIK
EtAD      ADACAAQLSDFFRQRAQKR
_consrvd      ** ** * ***** **
```


3.4 Construção do modelo

Após o alinhamento entre a sequência alvo e a proteína molde, o Modeller calculará automaticamente as coordenadas do modelo, através do comando “automodel”. Devido às diferentes conformações possíveis que uma proteína pode adotar, sobretudo nas regiões das alças, é importante a geração de pelo menos 10 modelos estruturais. Posteriormente, o modelo que apresentar os melhores parâmetros será eleito para o refinamento, isto é, minimização da estrutura, modelagem dos *loops* e dos rotâmeros. A construção do modelo estrutural da EtAD foi alcançado após a execução dos dados presentes no script4. Os dados que compõe o script4 estão disponíveis na figura 8 e deverão ser transcritos para o editor de texto,

nomeado como “script4” e salvo com a extensão .py na pasta “Tutorial Modeller”. Para executar o script4, o usuário deverá inserir a seguinte linha de comando no terminal “**mod9.13 script4.py**”. Após a geração dos modelos, será criado o arquivo script4.log, que apresenta os registros da construção dos modelos. Dentre as informações, destaca-se o valor do DOPE (*Discrete Optimized Protein Energy*) score, usado para escolher o melhor modelo entre os 10 gerados. De acordo com o script4.log, a estrutura que alcançou o menor DOPE score e, portanto, o melhor score, foi a estrutura EtAD.B99990009.pdb (DOPE score - 16681.77930).

Figura 8. Script 4, escrito em Python. As informações desta figura deverão ser fielmente transcritas para o bloco de notas e, posteriormente, o arquivo deverá ser salvo como “script4.py”.

```
from modeller import *
from modeller.automodel import *
#from modeller import soap_protein_od

env = environ()
a = automodel(env, alnfile='EtAD-1Z3A.ali',
              knowns='1Z3AA', sequence='EtAD',
              assess_methods=(assess.DOPE,
                             #soap_protein_od.Scorer(),
                             assess.GA341))

a.starting_model = 1
a.ending_model = 10
a.make()
```

O valor do DOPE para cada resíduo em individual pode ser calculado através do script5. Antes de executar o script5, o usuário deverá transcrever fielmente as informações contidas na figura 9 para o editor de texto, nomear o arquivo como “script5” e salvá-lo com a extensão .py no diretório “Tutorial Modeller”. A

execução do script5 é realizada através do comando “**mod9.13 script5.py**”. O resultado do DOPE poderá ser convertido em gráfico, por intermédio do programa GNUPLOT, através do comando ‘**plot "EtAD.profile" using 1:42 with lines**’.

Figura 9. Script 5 escrito em Python. As informações desta figura deverão ser fielmente transcritas para o bloco de notas e, posteriormente, o arquivo deverá ser salvo como “script5.py”

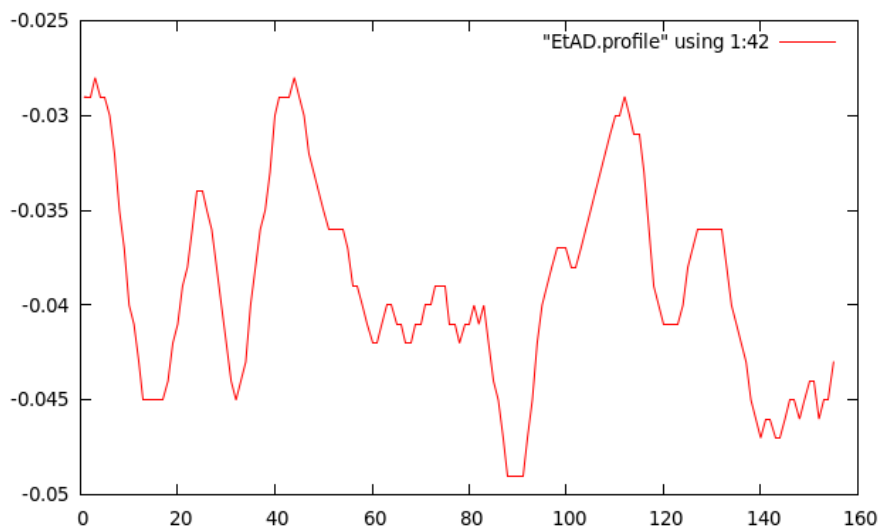
```
from modeller import *
from modeller.scripts import complete_pdb

log.verbose() # request verbose output
env = environ()
env.libs.topology.read(file='${LIB}/top_heav.lib') # read topology
env.libs.parameters.read(file='${LIB}/par.lib') # read parameters

# read model file
mdl = complete_pdb(env, 'EtAD.B99990009.pdb')

# Assess with DOPE:
s = selection(mdl) # all atom selection
s.assess_dope(output='ENERGY_PROFILE NO_REPORT', file='EtAD.profile',
              normalize_profile=True, smoothing_window=15)
```

Figura 10. DOPE para cada resíduo em individual. A energia de cada resíduo permanece relativamente baixa



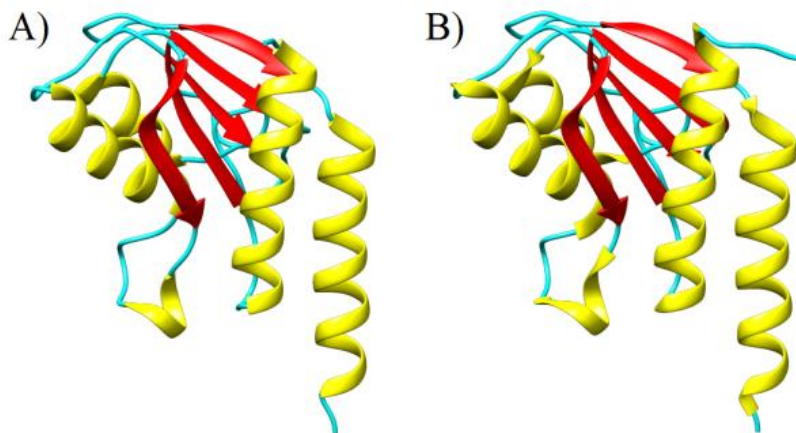
Após a avaliação, vários outros passos deverão ser realizados para alcançar a validação da estrutura. Via de regra, a estrutura, após sua determinação, deverá ser refinada, através da minimização da energia e modelagem dos *loops* e rotâmeros. Uma vez refinada, a estrutura

passará por testes de validação, tais como PROCHECK (LASKOWSKI *et al.*, 1996), Verify3D (EISENBERG *et al.*, 1997), ProSa (WIEDERSTEIN; SIPPL; 2007) e Erratv2 (KOH *et al.*, 2003), a fim de certificar a confiabilidade da estrutura. No entanto, este

trabalho se limita à concepção do modelo. A estrutura tridimensional da EtAD, bem como do molde (EcAD) (Fig. 11) pode ser visualizada por intermédio de programas de visualização, tais como UCSF Chimera e VMD. EtAD e

EcAD compartilham o mesmo padrão de enovelamento e são compostas por 4 α -hélices e cinco fitas- β .

Figura 11. Estrutura tridimensional da adenosina deaminase. A) Representação em fitas da EtAD (EtAD.B99990009.pdb). B) Representação em fitas da EcAD (1Z3A) usada como molde. Amarelo, vermelho e ciano correspondem às α -hélices, fitas- β e alças (*loops*), respectivamente.



4. Conclusão

À luz do conhecimento estrutural, diversos processos envolvendo proteínas podem ser compreendidos e melhor analisados. A determinação da estrutura proteica possibilita, além da caracterização inicial, informações sobre o funcionamento da proteína. Essas informações podem ser úteis no desenvolvimento de estratégias que buscam aumentar a atividade de determinadas enzimas, como no caso das mutações sítio-dirigidas, bem como no desenvolvimento de inibidores e/ou bloqueadores da atividade proteica, como comumente ocorre no desenho racional de fármacos. Ademais, este trabalho possibilita ao leitor lançar-se no campo da Biologia

Computacional, através de um tutorial experimental extremamente facilitado.

5. Referências bibliográficas

CALIXTO, P. H. M. *et al.* Gene identification and comparative molecular modeling of a *Trypanosoma rangeli* major surface protease. *J Mol Model*, v. 19, n. 8, p. 3053-64, Aug 2013.

DAI, W. *et al.* Improvement in low-homology template-based modeling by employing a model evaluation method with focus on topology. *PLoS One*, v. 9, n. 2, p. e89935, 2014.

EISENBERG, D. *et al.* VERIFY3D: assessment of protein models with three-dimensional profiles. *Methods Enzymol*, v. 277, p. 396-404, 1997.

- ESWAR, N. *et al.* Protein structure modeling with MODELLER. *Methods Mol Biol*, v. 426, p. 145-59, 2008.
- HU, Y. H. *et al.* *Edwardsiella tarda* Hfq: impact on host infection and global protein expression. *Vet Res*, v. 45, p. 23, 2014.
- JANDA, J. M.; ABBOTT, S. L. Infections associated with the genus *Edwardsiella*: the role of *Edwardsiella tarda* in human disease. *Clin Infect Dis*, v. 17, n. 4, p. 742-8, Oct 1993.
- KOH, I. Y. *et al.* EVA: Evaluation of protein structure prediction servers. *Nucleic Acids Res*, v. 31, n. 13, p. 3311-5, Jul 1 2003.
- LASKOWSKI, R. A. *et al.* AQUA and PROCHECK-NMR: programs for checking the quality of protein structures solved by NMR. *J Biomol NMR*, v. 8, n. 4, p. 477-86, Dec 1996.
- LUO, M.; SCHRAMM, V. L. Transition state structure of *E. coli* tRNA-specific adenosine deaminase. *J Am Chem Soc*, v. 130, n. 8, p. 2649-55, Feb 27 2008.
- MARTI-RENO, M. A. *et al.* Comparative protein structure modeling of genes and genomes. *Annu Rev Biophys Biomol Struct*, v. 29, p. 291-325, 2000.
- MOBAREC, J. C. *et al.* Modern homology modeling of G-protein coupled receptors: which structural template to use? *J Med Chem*, v. 52, n. 16, p. 5207-16, Aug 27 2009.
- NADEAU, O. W.; CARLSON, G. M. A review of methods used for identifying structural changes in a large protein complex. *Methods Mol Biol*, v. 796, p. 117-32, 2012.
- NELSON, J. J. *et al.* Extraintestinal manifestations of *Edwardsiella tarda* infection: a 10-year retrospective review. *J La State Med Soc*, v. 161, n. 2, p. 103-6, Mar-Apr 2009.
- PETTERSEN, E. F. *et al.* UCSF Chimera--a visualization system for exploratory research and analysis. *J Comput Chem*, v. 25, n. 13, p. 1605-12, Oct 2004.
- SALI, A. *et al.* Evaluation of comparative protein modeling by MODELLER. *Proteins*, v. 23, n. 3, p. 318-26, Nov 1995.
- SCHMIDT, T. *et al.* Modelling three-dimensional protein structures for applications in drug design. *Drug Discov Today*, v. 19, n. 7, p. 890-897, Jul 2014.
- VALLAT, B. K. *et al.* A template-finding algorithm and a comprehensive benchmark for homology modeling of proteins. *Proteins*, v. 72, n. 3, p. 910-28, Aug 15 2008.
- UNAJAK, S. *et al.* Homology modeling and virtual screening for antagonists of protease from yellow head virus. *J Mol Model*, v. 20, n. 3, p. 2116, Mar 2014.
- WIEDERSTEIN, M.; SIPPL, M. J. ProSA-web: interactive web service for the recognition of errors in three-dimensional structures of proteins. *Nucleic Acids Res*, v. 35, n. Web Server issue, p. W407-10, Jul 2007.
- WILTGEN, M.; TILZ, G. P. Homology modelling: a review about the method on hand of the diabetic antigen GAD 65 structure prediction. *Wien Med Wochenschr*, v. 159, n. 5-6, p. 112-25, 2009.

ZHENG, Y. *et al.* Different approaches to expressing *Edwardsiella tarda* antigen GAPDH in attenuated *Vibrio anguillarum* for multivalent fish vaccines. *J Fish Dis*, v. 35, n. 8, p. 569-77, Aug 2012.