國 立 交 通 大 學

生物資訊研究所

碩 士 論 文

FASTR3D: 一個快速且準確搜尋相似RNA三級結構的工具

FASTR3D: A Fast and Accurate Search Tool for Similar RNA 3D Structures

研究生：賴慶恩

指導教授：盧錦隆 博士

中華民國　九十八　年　六　月

FASTR3D: 一個快速且準確搜尋相似 RNA 三級結構的工具

FASTR3D: A Fast and Accurate Search Tool for Similar RNA 3D Structures

研究生：賴慶恩　　Student：Chin-En Lai
指導教授：盧錦隆 博士　Advisor：Dr. Chin Lung Lu

國　立　交　通　大　學

生　物　資　訊　研　究　所

碩　士　論　文

A Thesis Submitted to Institute of Bioinformatics
College of Biological Science and Technology
National Chiao Tung University in partial Fulfillment of the
Requirements for the Degree of Master in
Biological Science and Technology
June 2009
Hsinchu, Taiwan

中文摘要

FASTR3D 是一個架構在網路上的搜尋工具，它允許使用者快速且精確的搜尋 PDB 資料庫中相似的 RNA 結構。目前，它允許使用者輸入三種查詢的格式： (i) 一個 RNA 三級結構的 PDB 代碼(預設值)，可伴隨一段特定的殘基(residue)範圍， (ii) 一段使用點與括號表示法(dot-bracket notation) 的二級結構，可伴隨輸入其一級序列，與 (iii) 一段 FASTA 格式的 RNA 一級序列。另外，使用者也可以指定一些額外的過濾選項來執行 FASTR3D： (i) PDB 資料庫中 RNA 結構的發佈日期，(ii) 用來決定 RNA 結構的實驗方法，與它們最小的解析度(resolution)。FASTR3D 是以雜湊表當作搜尋相似 RNA 結構的核心演算法，來增加搜尋時的速度。我們預先將現有的 PDB 代碼轉換為二級結構，同時將它們分割成許多不重疊的 k-tuples 並儲存在一個表格中。然後 FASTR3D 可利用輸入的二級結構重疊 k-tuples 去搜尋雜湊表並找出二級結構完全一樣的 RNAs。最後，如果查詢的是一個 RNA 三級結構，FASTR3D 會更進一步地透過 3D 過濾器來過濾掉一些三級結構與輸入結構不相似的 RNAs。在輸出的頁面，FASTR3D 將會標示出使用者查詢的 RNA 分子以及使用者設定的選項，接著再顯示出找到結構相似 RNA 的詳細列表。特別的是，當查詢的 RNA 是三級結構時，FASTR3D 提供一個圖形畫面以顯示出每一個找到 RNA 的三級結構與輸入結構之間的重疊。FASTR3D 目前可以在線上讓使用者使用，其網址在 http://bioalgorithm.life.nctu.edu.tw/FASTR3D/。

Abstract

FASTR3D is a web-based search tool that allows the user to fast and accurately search the PDB database for structurally similar RNAs. Currently, it allows the user to input three types of queries: (i) a PDB code of an RNA tertiary structure (default), optionally with specified residue range, (ii) an RNA secondary structure, optionally with primary sequence, in the dot-bracket notation and (iii) an RNA primary sequence in the FASTA format. In addition, the user can run FASTR3D with specifying additional filtering options: (i) the released date of RNA structures in the PDB database, and (ii) the experimental methods used to determine RNA structures and their least resolutions. FASTR3D uses hash table as kernel algorithm to search for similar RNA structures, and it can make the searching more efficiently. We transform PDB codes into secondary structures, and store them in a table of non-overlapping k-tuples beforehand. Then, FASTR3D can find RNAs whose secondary structures are exactly equal to that of the query by searching overlapping k-tuples of the query secondary structure against the hash tables. Finally, FASTR3D further screens out those RNAs whose tertiary structures are not similar to that of the query by a 3D-filter, if the query is a tertiary structure. In the output page, FASTR3D will show the user-queried RNA molecule, as well as user-specified options, followed by a detailed list of identified structurally similar RNAs. Particularly, when queried with RNA tertiary structures, FASTR3D provides a graphical display to show the structural superposition of the query structure and each of identified structures. FASTR3D is now available online at http://bioalgorithm.life.nctu.edu.tw/FASTR3D/.

Acknowledgement

經過兩年的碩士班訓練，終於畢業並且邁向人生下一個階段了。在這邊我要感謝我的指導教授盧錦隆老師，謝謝老師在這兩年當中，時常耐心的與我討論研究中所遭遇的各種難題，讓我學習到一個好的研究態度以及如何分析問題、修正方向和解決問題。

感謝我的家人，謝謝爸爸、媽媽、妹妹在這段時間的支持與鼓勵，全力在經濟上、精神上以及感情上支持我，讓我在做研究當中沒有後顧之憂，並且也時常透過電話關心我，讓我可以更投入在我的研究上。

感謝彥菱學姊，謝謝你這兩年不論在程式的寫法上、講解演算法或是和我討論問題解決辦法，都給了我很大的幫助。也從你身上學習到樂觀開朗的個性、家庭主婦勤儉持家的精神以及折價卷的重要。

感謝明原學長，謝謝你在我研究中網站工具的架設上給予很多的協助，讓我們美輪美奐的 FASTR3D 順利發表。也讓我在程式的撰寫上學習許多不同的技巧與方法。

感謝志偉，謝謝你和我分享的生物知識，讓我在研究中突破了不少瓶頸。這兩年來我們大多一起行動，早上一起進實驗室，一起熬夜趕進度，一起修課寫程式，一起住，一起採買，一起游泳，一起吃也一起肥，和我一起驗證了身材不是一天造成的道理，這些我都會記得的。

感謝智先，謝謝你常常招呼我們一起去運動，維持實驗室的健康不遺餘力，也讓我看到了一個男子漢該有的魄力。我會記得你維持筆電桌布多樣性的堅持，讓我知道原來台灣電影的上檔速度是這樣的快。

感謝忠翰，謝謝你在我報告生物的時候給予的諮詢也介紹各式各樣

不同種類的蔬菜水果給我認識，以及示範如何一次喝光家庭號的豆漿，一大罐的巧克力，還有嗑光像山一樣高的青菜塔，讓我佩服不已。

感謝芸蓁協助我論文研究中的 3D 秀圖介面，使得我可以放心把這一部份分出去並且專心完成核心程式的撰寫。也很謝謝那次舉辦的單車之旅，讓我能放鬆研究的緊繃，欣賞新竹的海岸風光。另外，也讓我認識到嘉義與新竹來回的通勤費是這麼的昂貴，以後要住近一點的道理。

感謝昆澤幫忙測試研究開發的程式，雖然找到的錯誤千奇百怪匪夷所思，但是卻是讓整個研究正確性大大提高的重要推手。此外也讓我認識變種人類的無限潛能：會高速衝刺、不怕子彈、喜歡芭樂以及長大以後會亂丟石頭的專長，由此可見生物科技的重要性。

感謝晟宸，讓我見識到大自然的奇妙，像是各種不同種族的野生動物會互相分工合作收集資源、會互相佔領地盤，或者進化成使用能量互相攻擊等等的特殊行為。

特別要感謝我最愛的女朋友悅晴，謝謝妳在這段時間給我的支持與鼓勵，也在這段時間讓我感受到妳的包容以及體貼，和我一起度過許多喜怒哀樂的研究生活，謝謝妳的陪伴。

最重要的，向我信仰的上帝獻上感恩，不論是當初學校系所的選擇、入學後遭遇各樣跨領域的困難、在研究上的瓶頸、一波三折終於成功發表的論文或是找到理想的工作，這一路上都看到上帝豐富的恩典跟保守，讓我不為明天而憂慮，常常喜樂，心裡有力量，勇往直前。

最後，謹以畢業的喜悅與眾多身邊的師長朋友分享，謝謝你們。

慶恩

2009/07/09 於實驗室

Contents

List of tables

List of figures

Chapter 1

Introduction

In recent years, there is a fast growing interest in non-coding RNAs (ncRNAs) because, although their transcripts are not translated into proteins, they play essential roles in many cellular processes, including gene regulation, RNA modification and chromosome replication [1–4]. However, the function of most ncRNAs has yet to be determined. Likewise to proteins, a common and useful approach for annotating the function of an ncRNA is by searching databases for similar RNA molecules whose functions are already known. For this purpose, several databases of ncRNAs have been proposed, such as NONCODE [5], RNAdb [6], miRBase [7], fRNAdb [8] and ncRNAdb [9]. For these databases, however, the search is performed solely by querying keywords, accession numbers, transcript/organism names and/or sequences. Compared with the 20-letter protein alphabet, the 4-letter RNA alphabet is smaller and less informative, leading to that searching for similar RNA molecules based on sequence comparison/alignment is not as accurate and powerful as it does for proteins.

Actually, a more reliable way for determining the functions of ncRNAs is from the analysis on the structure level, since structures of molecules are typically more evolutionarily conserved than their sequences. In this regard,

a series of recent efforts and studies has led to a substantial increase in both the number and the size of solved RNA structures deposited in the PDB and NDB databases [10,11]. Therefore, it has become more and more crucial to develop automatic tools that are able to efficiently and accurately search for structurally similar RNA substructures and motifs against the PDB/NDB database. Although it is easy to detect structural similarities in two RNA molecules at the secondary structural level, however, doing it at the tertiary structural level would lead to a nondeterministic polynomial time (NP)-hard problem. What's more, even if we find a constant ratio approximation algorithm to compute a pair of maximal substructures with exhibiting the highest degree of similarity from two RNA (or protein) tertiary [three-dimensional (3D)] structures [12]. Therefore, currently available tools, such as ARTS [13,14], DIAL [15], SARSA [16] and SARA [17], are all based on some heuristic approaches for comparing the similarities of two RNA tertiary structures. All these methods, however, at least have quadratic-time complexity and hence are impractical for searching ever-increasing databases of RNA tertiary structures. Currently, there are several tools that can be used to search motifs in RNA structures, including FR3D [18], PRIMOS [19] and RNAMotif [20]. FR3D uses a base-centred method to perform a geometric search of RNA local/composite 3D motifs. PRIMOS searches for locally structural similarities of consecutive RNA fragments by comparing their pseudotorsion angles. RNAMotif finds the fragments of an RNA sequence that conform to a predefined descriptor of defining a particular motif of secondary structure.

In this study, we have developed a web server, called FASTR3D ("Fast and Accurate Search Tool for RNA 3D structures"), based on a hashing

algorithm that is able to fast and accurately find structural similarities for a query of RNA molecule in the PDB database. In principle, this hashing algorithm consists of three main procedures as follows. The first procedure is to derive the primary sequence, secondary structure and tertiary structure information of all RNA molecules currently deposited in the PDB database and then store the derived second structures in a hash table. The secondary procedure is to derive some possible secondary structures of the query RNA if it is a primary sequence or tertiary structure. The third procedure is to search the hash table for all candidate RNAs whose secondary structures exactly match that of the query RNA, followed by primary sequence filter and/or tertiary structure filter to screen out those candidates whose primary sequences and/or tertiary structures are not equal to that of the query RNA. The FASTR3D web server is now available online at http://bioalgorithm.life.nctu.edu.tw/FASTR3D/ for public access.

In addition, our FASTR3D was tested with a number of RNA primary sequences, secondary structures and tertiary structures, and its experimental results on querying RNA primary sequences and secondary structures were also compared with those obtained by the search tool of RNA FRABASE (http://rnafrabase.ibch.poznan.pl/), which was developed by Popenda et al. [21] on the basis of RNA primary sequences and/or secondary structures using the methods of regular expression and pattern recognition. The comparison of experimental results on querying secondary structures reveals that FASTR3D has a comparable performance as RNA FRABASE, both with returning the search results in a short time. However, our FASTR3D is able to find more structurally similar RNAs for a query of RNA primary sequence, when compared with RNA FRABASE, because FASTR3D

searches for structurally similar RNAs using the secondary structure derived from the query sequence, while RNA FRABASE searches them solely based on the primary sequence. In addition, the function of querying RNA tertiary structures in FASTR3D, as well as the online graphical display of showing the structural superposition of the query and identified structures, is not available in RNA FRABASE.

Chapter 2

Method

Our FASTR3D was implemented based on a hashing algorithm whose procedure flowchart, as shown in Figure 1, consists of three major procedures. The first procedure is a preprocessing job that is to derive the primary sequence, secondary structure and tertiary structure information of all RNAs in the PDB database and particularly store the derived secondary structures (i.e. standard Watson–Crick and wobble base pairs) in a hash table. Note that the secondary structure information was derived using the RNAView program [22], while the tertiary structure information of pseudotorsion angles η and θ values was derived using the AMIGOS program [23]. Basically, an RNA 3D structure can be simply represented by a worm that is defined to be an order set of η and θ coordinates for all nucleotides in the RNA structure [23]. Duarte and Pyle [23] have shown that the pseudotorsion angles η and θ are at least as descriptive of backbone morphology as standard torsion angles (α, β, γ, δ, ε and ζ) and can be used to specify the backbone conformation of an individual nucleotide. They have also demonstrated that two RNA molecules can be very similar on the 3D structure level if the average Euclidean distance between their corresponding *worms* is small, which is useful for us to design a 3D filter for filtering out

those database hits whose 3D structures are not similar to the query RNA molecule. The second procedure is to derive the secondary structure information for the RNA queried by the user. Currently, the user can input any of the following three types of queries: (i) a PDB code of an RNA tertiary structure optionally with specified residue range, (ii) an RNA secondary structure, optionally with primary sequence, in the dot-bracket notation, and (iii) an RNA primary sequence in the FASTA format. If the query is a PDB code of an RNA tertiary structure, then its secondary structure is derived from its PDB file, which is downloaded from the PDB database, using the RNAView program [22].

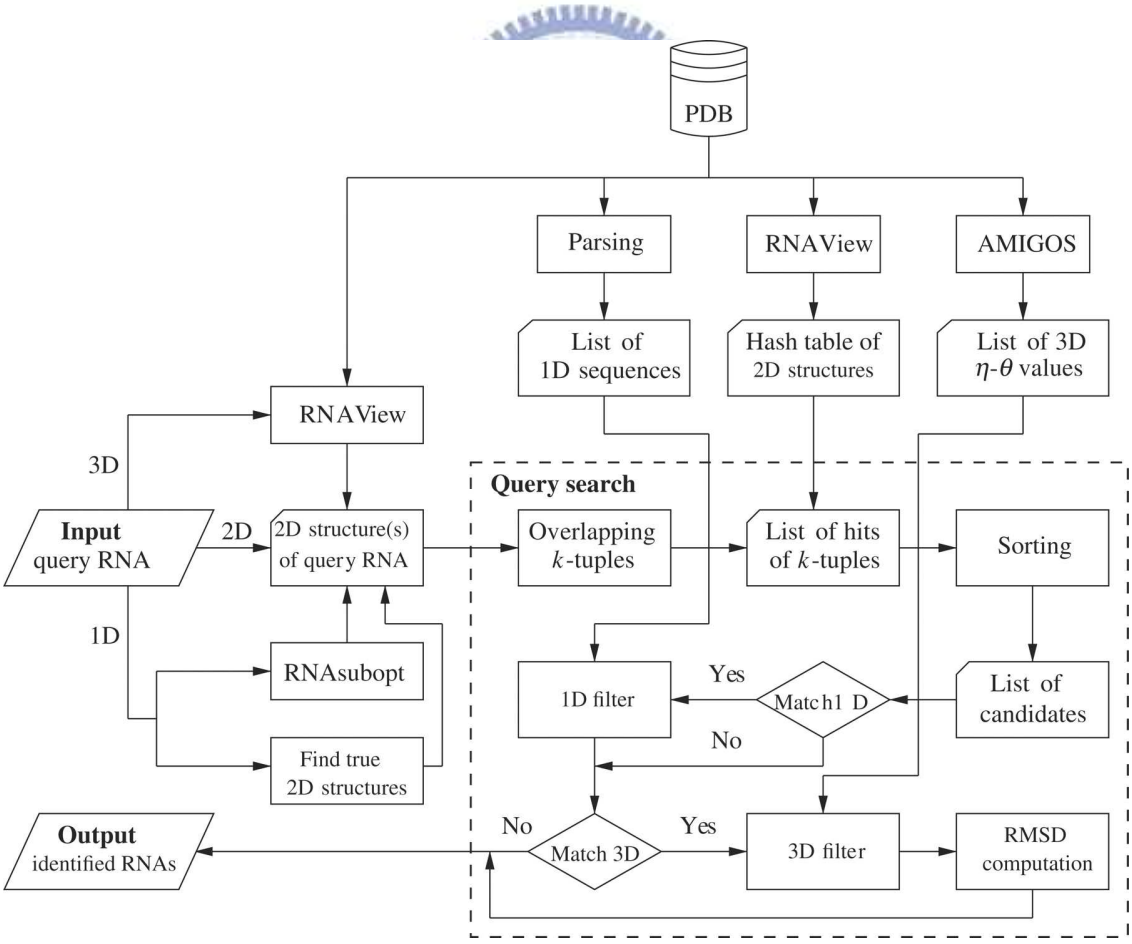

Figure 2-1. The procedure flowchart of FASTR3D, 2D and 3D refer to primary, secondary and tertiary, respectively.

If the query is an RNA primary sequence, then a set of at most X suboptimal secondary structures is derived using the RNAsubopt program [24], where the default value of X is 16. It is often observed that the suboptimal secondary structure predicted by RNAsubopt for an RNA molecule may not be the true secondary structure. Therefore, we design an alternative approach as follows to derive a set of at most X most frequently occurring true secondary structures for the query RNA sequence. First, we search the PDB database for all the RNAs whose primary sequences are equal to the query sequence. Then, we use RNAView to derive all the secondary structures from the PDB files of these RNAs and from them we finally select at most X most frequently occurring secondary structures. The third procedure is to use the hash table to quickly search for all candidate RNAs whose secondary structures exactly match that of the query RNA (or any of X predicted/true secondary structures for the query RNA), followed by primary sequence filter (if the query RNA has primary sequence information) and/or tertiary structure filter (if the query is an RNA tertiary structure) to screen out those candidates whose primary sequences and/or tertiary structures are not equal to that of the query RNA.

In the following, we describe the details of the significant steps in the above procedures, including how to prepare the hash table of the secondary structures of all RNA molecules currently deposited in the PDB database, how to use this hash table to search for RNA structural similarities and how to utilize the η and θ values to efficiently screen out structurally non-similar candidates. For simplicity, we let $D = \{S_1, S_2, \ldots, S_m\}$ denote the database of the secondary structures derived from the PDB database using the RNAView program [22], and let Q be the secondary structure of the query

RNA. Note that in the structural database *D*, each structure S_i is labelled with an integer i, to which we refer as the index of Si. Moreover, we denote by the *k-tuple* a consecutive sequence of *k* nt (residues) within an RNA molecule. Clearly, there are ($|S| - k + 1$) *overlapping k*-tuples for a given RNA secondary structure *S* with $|S|$ residues. The *offset* of a *k*-tuple within *S* is defined to be the position of its first residue with respect to the first residue of *S*. For convenience, we use the letter *j* to denote offset and use the notation $w_j(S)$ to denote the *k*-tuple of *S* that has offset *j*. Therefore, the position of each occurrence of each *k*-tuple within a structure S_i of *D* can be represented by an (*i*, *j*) pair.

2.1 Hash table construction for a structural database

Here, we reorganize the structural database *D* by using a hash table to store the position of each occurrence of each k-tuple. Note that each RNA tertiary structure S_i in the structural database *D* is represented by its secondary structure in the *dot-bracket* format, where an unpaired nucleotide is denoted by a dot and a Watson–Crick (e.g. AU, UA, CG, GC) or wobble (e.g. GU and UG) base pair by a pair of opening and closing round brackets (e.g. '(' and ')'). Moreover, to correctly represent complicated secondary structures in RNA molecules, the bracket notation used in this study is extended by allowing the user to use additional squared brackets (e.g. '[' and ']') to represent simple pseudoknots and kissing loops, and curly brackets (e.g. '{' and '}') to represent high-order pseudoknotted structures.

To simplify our implementation, all the brackets appearing in an RNA secondary structure are transformed into the round brackets, since their exact

pairing relationships between the opening and closing brackets are already recorded in advance using a data structure of 1D array. For each secondary structure S_i with $|S_i|$ residues, we break it into $\left\lceil \frac{|S_i|}{k} \right\rceil$ non-overlapping k-tuples and store the position of each occurrence of each k-tuple in the hash table. Recall that for any k-tuple w = $r_1 r_2 \ldots r_k$, each residue r_x, where $1 \le x \le k$, can be either a dot, opening bracket or closing bracket. Therefore, each of these three possible symbols is then encoded as a base-3 digit as follows: $e(.) = 0_3$, $e(() = 1_3$ and $e()) = 2_3$. Using this encoding, w can be represented uniquely by a decimal integer $E(w) = \sum_{x=1}^{k} 3^{x-1} e(r_x)$. Finally, the hash table of the structural database D is represented by two data structures, a list of positions L and an array A of pointers into L. Basically, there are 3^k pointers in A, with one pointer corresponding to each of the 3^k possible k-tuples. More clearly, the pointer at position $E(w)$ of A points to the entry of L that describes the positions of the first occurrence of the k-tuple w in the database D. Then we can obtain the positions of all occurrences of w in D by traversing L from this position until we reach the location pointed by the pointer located at position $E(w) + 1$ of A. Below, we illustrate the above hash table construction with a simple example. For simplicity, we let $k = 2$ and D consist of two RNAs S_1 and S_2 whose secondary structures are S_1 = '(((...).))' and S_2 = '.((....)).', respectively. In Table 1, each row contains the list of the positions of all occurrences for each of the nine possible 2-tuples, denoted by w. Then the pointer at $E(w)$ of A points to the beginning of the position list corresponding to w and the concatenation of the nine position lists in the order from top to bottom forms L.

Table 2-1. A 2-tuple hash table for S_1= '(((...).))' and S_2 = '.((....)).'

2-tuple *w*	***E*(*w*)**	**Position lists**
..	0	(1, 5), (2, 5)
.(	1	(2, 1)
.)	2	(2, 7)
(.	3	(1, 3), (2, 3)
((	4	(1, 1)
()	5	
).	6	(1, 7), (2, 9)
)(	7	
))	8	(1, 9)

2.2 Query substructure search

In the following, we describe how to use the hash table of the structural database D as constructed above to search for all occurrences of a query Q of an RNA secondary structure. Suppose that the length of Q is n. Then we can proceed position-by-position along Q from position 1 to $n - k + 1$. At position p, where $1 \le p \le n-k+1$, we obtain the list of the positions of all the occurrences of the k-tuple $w_p(Q)$ from the hash table of D via the pointer of $E(w_p(Q))$. Let this list contain q positions, say (i_1, j_1), (i_2, j_2), ... , (i_q, j_q). From this list, we derive a list of *hits* $H_1 = (i_1, j_1 - p, j_1)$, $H_2 = (i_2, j_2 - p, j_2)$, ... , $H_q = (i_q, j_q - p, j_q)$. This list of hits is then added to a master list M of hits that accumulates all the hits we derived when p runs from 1 to $n - k + 1$. For convenience, the elements of a hit are referred to as the *index*, *shift* and *offset*. Next, we sort all the elements in M first by index and then by shift. Finally, we scan through M by looking for *runs* of hits for which the index and shift are identical. Clearly, by further sorting each of these runs by offset, we can determine the region of some structure in D that exactly matches the

query structure *Q*. For example, we search for the query of an RNA secondary structure *Q* = '(...).' within the hash table of *D* as constructed in Table 1. In Table 2, column 3 displays the occurrence positions in *D* for each 2-tuple of *Q*, with corresponding hits shown in column 4, and column 5 shows the sorted *M* in which the run of three hits highlighted in bold indicates that there is a match between *Q* and S_1 that starts at the third nucleotide and ends at the eighth nucleotide. Basically, the search speed of the above hashing algorithm is proportional to the size of the master list *M*, which falls off rapidly with increasing the value of *k*. Although a greater *k* increases the search speed, the condition $|Q| \geq 2k-1$ should be satisfied to guarantee that the hashing algorithm will find a hit at some point in the matching region. For example, suppose that *S* = '((....))' and *Q* = '(....)'. If *k* = 4, then none of three overlapping 4-tuples in *Q* is able to match any of two non-overlapping 4-tuples in *S*. In addition, the hash table is generated in advance for a fixed *k* in our algorithm. Therefore, to achieve the best search speed and reduce the storage requirement, we set the value of *k* as $\left\{20, \left\lceil \frac{|Q|}{2} \right\rceil \right\}$.

Table 2-2. A The search of the query secondary structure Q = '(...).'

p	$w_p(Q)$	**Positions**	***H***	***M***
1	(.	(1, 3)	(1, 2, 3)	**(1, 2, 3)**
		(2, 3)	(2, 2, 3)	**(1, 2, 5)**
2	..	(1, 5)	(1, 3, 5)	**(1, 2, 7)**
		(2, 5)	(2, 3, 5)	(1, 3, 5)
3	..	(1, 5)	(1, 2, 5)	(2, 2, 3)
		(2, 5)	(2, 2, 5)	(2, 2, 5)
4	.)	(2, 7)	(2, 3, 7)	(2, 3, 5)
5	).	(1, 7)	(1, 2, 7)	(2, 3, 7)
		(2, 9)	(2, 4, 9)	(2, 4, 9)

2.3 Tertiary structure filter using pseudotorsion angles

Basically, the comparison of RNA conformation is a high-dimensional problem, because six standard torsion angles (α, β, γ, δ, ε and ζ) are needed to specify the backbone conformation of a single nucleotide. Duarte and Pyle [23], however, pointed out that the pseudotorsion angles η $(C4'_{i-1} - P_i - C4'_i - P_{i+1})$ and θ $(P_i - C4'_i - P_{i+1} - C4'_{i+1})$ are at least as descriptive of backbone morphology as standard torsion angles and they may be even superior in terms of specifying the backbone conformation of an individual nucleotide. This suggests that the η–θ plot can provide us a 2D representation of the conformation properties of an entire RNA molecule, so that we can carry out the rapid and accurate comparison of RNA conformations. Duarte *et al.* [19] further called such an ordered set of η–θ coordinates as an RNA *worm*. As was used by Duarte *et al.* [19], we can detect the conformation difference of two RNAs by comparing their worms based on a Euclidean metric as follows. Let *Q*' denote an identified candidate RNA whose secondary structure matches that of the query RNA *Q* with *n*

residues, and let the worms of Q and Q' denoted by $\{(\eta_{1,1}, \theta_{1,1}), \ldots, (\eta_{1,n}, \theta_{1,n})\}$ and $\{(\eta_{2,1}, \theta_{2,1}), \ldots, (\eta_{2,n}, \theta_{2,n})\}$, respectively. The *conformational difference* between two residues $(\eta_{1,i}, \theta_{1,i})$ and $(\eta_{2,i}, \theta_{2,i})$ is defined to be $\Delta(\eta, \theta)_i = \sqrt{\Delta\eta_i^2 + \Delta\theta_i^2}$ where $\Delta\eta_i = \min\{|\eta_{1,i} - \eta_{2,i}|, 360 - |\eta_{1,i} - \eta_{2,i}|\}$ and $\Delta\theta_i = \min\{|\theta_{1,i} - \theta_{2,i}|, 360 - |\theta_{1,i} - \theta_{2,i}|\}$ (since 0° and 360° are the same). As was also pointed out by Duarte *et al.* (19), two residues $(\eta_{1,i}, \theta_{1,i})$ and $(\eta_{2,i}, \theta_{2,i})$ can be considered structurally identical if $\Delta(\eta, \theta)_i < 25°$. Therefore, based on this property, we design our tertiary structure filter to discard the identified RNA Q' from consideration if the average conformation difference $\overline{\Delta(\eta, \theta)}$ between Q and Q' is greater than or equal to a predefined cutoff, where $\overline{\Delta(\eta, \theta)} = \sqrt{\left(\sum_{i=1}^{n} (\Delta(\eta, \theta)_i)^2\right)/(n)}$ and for our purpose, the cutoff value is set as 55°.

Chapter 3

Implementation of Software Tools

Based on the hashing algorithm described in the previous chapter, we have developed a novel web-based tool, called FASTR3D (short for "Fast and Accurate Search Tool for RNA 3D structures"), which allows user to find similar tertiary structure, secondary structure and/or primary sequence when user queries them respectively. In the following, we will detail the usage of FASTR3D, including its input and output.

3.1 Usage of FASTR3D

3.1.1 Input of FASTR3D

FASTR3D provides an intuitive user interface as illustrated in Figure 3-1. In basic search, the user can submit a job by entering or pasting one of the following three types of queries to search for structurally similar RNA structures: (i) a PDB code of an RNA tertiary structure (1) (default), optionally with specified residue range, (ii) an RNA secondary structure (2), optionally with primary sequence, in the RNA FRABASE format (i.e. a kind of dot-bracket notation) and (iii) an RNA primary sequence (3) in the FASTA format.

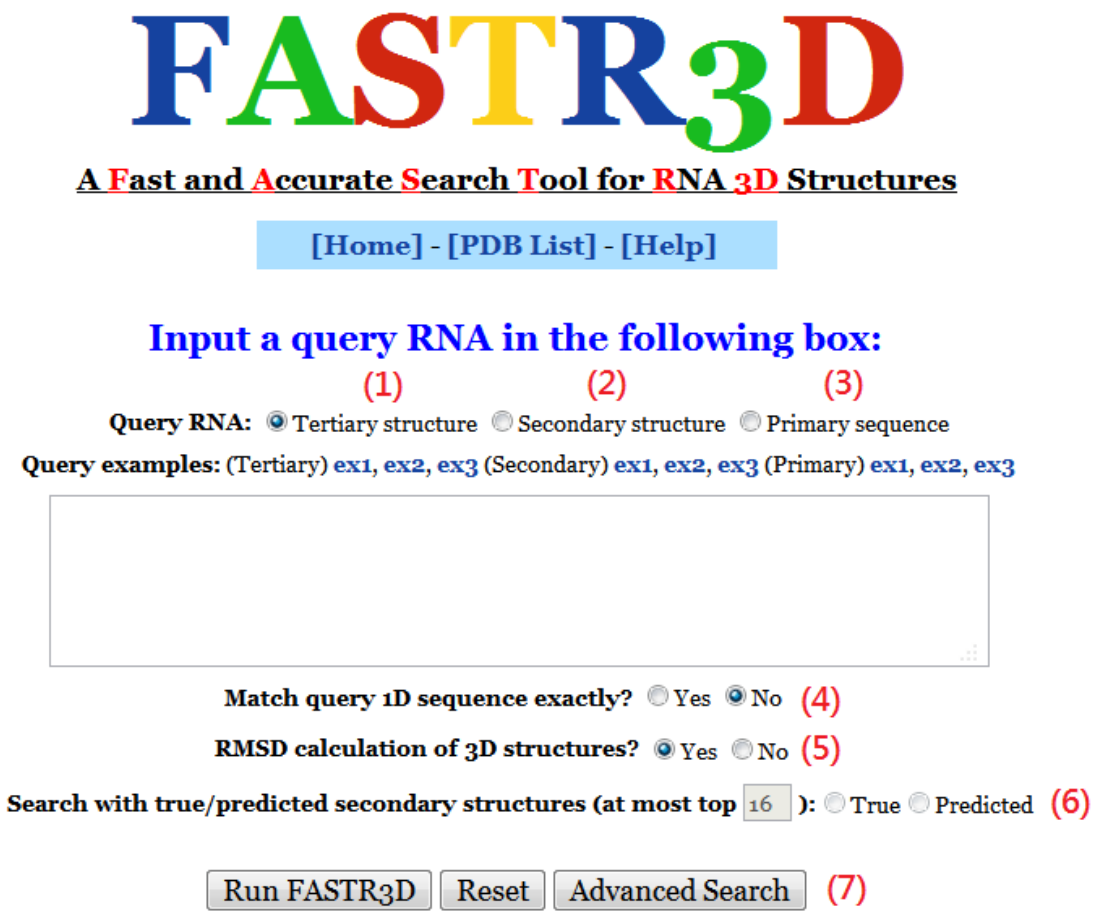

Figure 3-1. The web interface of FASTR3D.

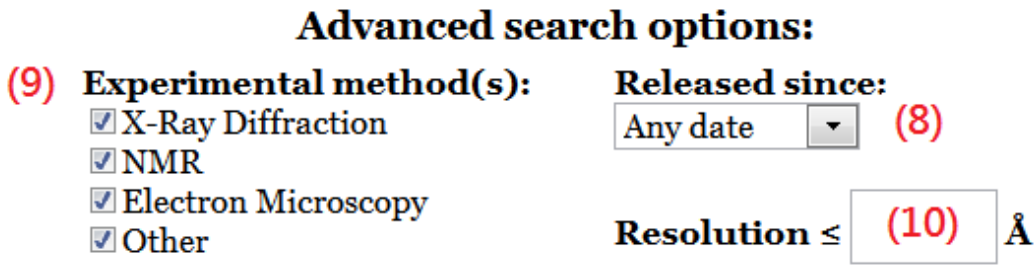

Figure 3-2. The advanced search options of FASTR3D.

In addition, the user can further restrict FASTR3D to return those RNAs whose primary sequences exactly match that of the query RNA by clicking on the “Yes” button of “Match query 1D sequence exactly?” (4). if the query

RNA contains the information of its primary sequence. If the query is an RNA tertiary structure, then the user can determine whether to calculate the RMSD between the query RNA and identified candidate RNAs with the considerations of computational performance by clicking on the "No" button of "RMSD calculation of 3D structures?" (5). If the query is a primary sequence, then the user can choose to use either at most *X* true, frequently occurring secondary structures or predicted suboptimal secondary structures to perform the PDB database search (6). The default value of *X* is 16 and can be changed by the user. The user can click on the "Advanced Search" bottom in Figure 3-2 (7) to activate the function of advanced search. In advanced search (refer to Figure 3-2), the user can run FASTR3D with specifying additional filtering options: (i) the released date of identified RNA structures in the PDB database (8), and (ii) the experimental methods used to determine identified RNA structures (such as X-Ray Diffraction, electron microscopy, and NMR) (9) and their least resolutions (10).

3.1.2 Output of FASTR3D

In the output page, FASTR3D will first show the user-queried RNA molecule, as well as user-specified options (See Figure 3-3). Next, it will show a detailed list of identified structurally similar RNAs, including corresponding PDB ID, primary sequence, secondary structure, tertiary structure, RMSD between the query and identified structures, chain ID, starting and ending nucleotide numbers, experimental method used to determine the structure, classification of RNA molecule (based on function, metabolic role, molecule type, cellular location and so on), released date in the PDB database and solved resolution (refer to Figure 3-4 for an

example). In addition, FASTR3D allows the user to save the search result in the Excel or CSV format for later process. Particularly note that if the query RNA is a tertiary structure, then the FASTR3D allows the user to visually view, rotate and enlarge the superposition of the query RNA and each of identified RNA by clicking on the link of "Jmol 3D" (Figure 3-5). If the query RNA is a primary sequence or secondary structure, then the user still can visually view, rotate and enlarge the tertiary structure of each identified RNA candidate.

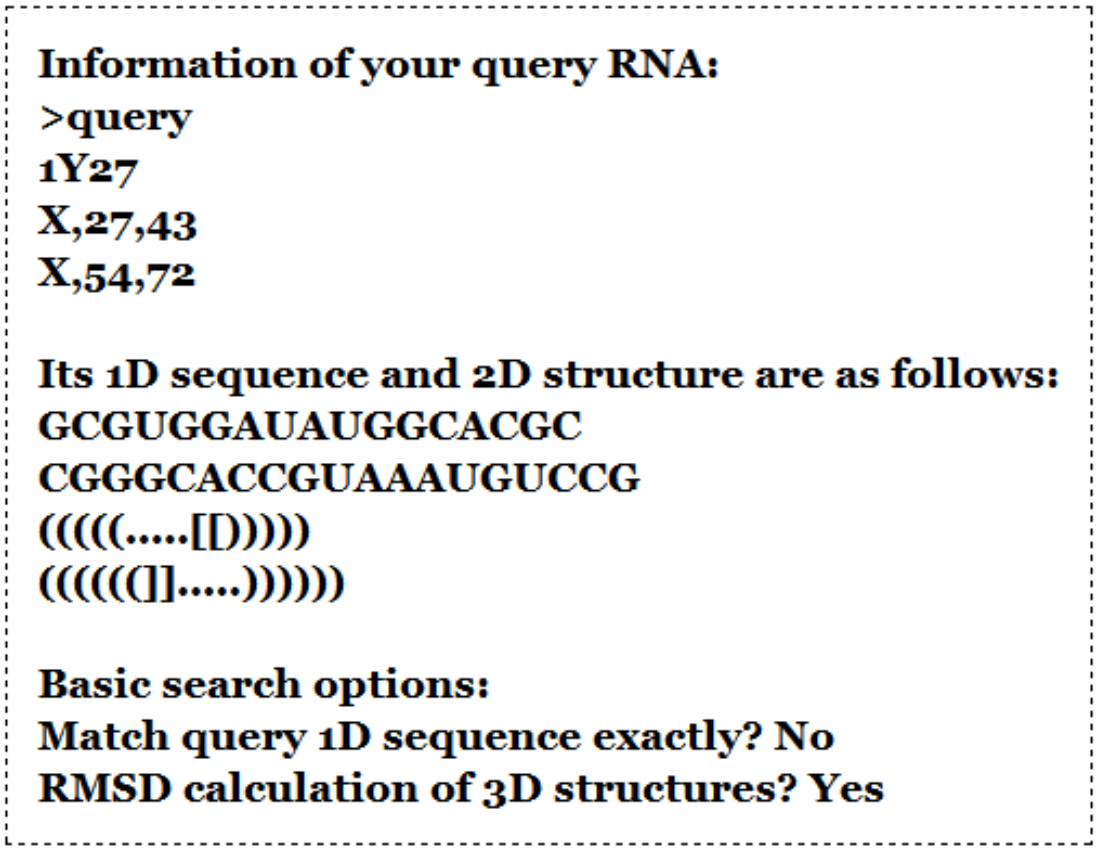

Figure 3-3. The user-queried RNA molecule and user-specified options.

No.	PDB id	Primary Sequence	Secondary Structure	Tertiary Structure	RMSD	Chain	Start	End	Method	Class	Released Date	Å
1	1Y27	GCGUGGAUAUGGCACGC CGGGCACCGUAAAUGUCCG	(((((.....[[))))) (((((([].....))))))	Jmol 3D	0.000	X X	27 54	43 72	X-Ray Difraction	RIBONUCLEIC ACID	28-DEC-04	2.4
2	2G9C	GCGUGGAUAUGGCACGC CGGGCACCGUAAAUGUCCG	(((((.....[[))))) (((((([].....))))))	Jmol 3D	0.981	A A	27 54	43 72	X-Ray Difraction	RIBONUCLEIC ACID	21-NOV-06	1.7
3	2B57	GCGUGGAUAUGGCACGC CGGGCACCGUAAAUGUCCG	(((((.....[[))))) (((((([].....))))))	Jmol 3D	0.987	A A	27 54	43 72	X-Ray Difraction	RIBONUCLEIC ACID	23-MAY-06	2.15
4	2EEW	GCGUGGAUAUGGCACGC CGGGCACCGUAAAUGUCCG	(((((.....[[))))) (((((([].....))))))	Jmol 3D	0.989	A A	27 54	43 72	X-Ray Difraction	RIBONUCLEIC ACID	13-NOV-07	2.25
5	2EEU	GCGUGGAUAUGGCACGC CGGGCACCGUAAAUGUCCG	(((((.....[[))))) (((((([].....))))))	Jmol 3D	0.991	A A	27 54	43 72	X-Ray Difraction	RIBONUCLEIC ACID	13-NOV-07	1.95
6	2EES	GCGUGGAUAUGGCACGC CGGGCACCGUAAAUGUCCG	(((((.....[[))))) (((((([].....))))))	Jmol 3D	0.992	A A	27 54	43 72	X-Ray Difraction	RIBONUCLEIC ACID	13-NOV-07	1.75
7	1U8D	GCGUGGAUAUGGCACGC CGGGCACCGUAAAUGUCCG	(((((.....[[))))) (((((([].....))))))	Jmol 3D	1.008	A A	27 54	43 72	X-Ray Difraction	RIBONUCLEIC ACID	23-NOV-04	1.95
8	2EET	GCGUGGAUAUGGCACGC CGGGCACCGUAAAUGUCCG	(((((.....[[))))) (((((([].....))))))	Jmol 3D	1.011	A A	27 54	43 72	X-Ray Difraction	RIBONUCLEIC ACID	13-NOV-07	1.95
9	2EEV	GCGUGGAUAUGGCACGC CGGGCACCGUAAAUGUCCG	(((((.....[[))))) (((((([].....))))))	Jmol 3D	1.014	A A	27 54	43 72	X-Ray Difraction	RIBONUCLEIC ACID	13-NOV-07	1.95
10	3DS7	GCGUGGAUAUGGCACGC CGGGCACCGUAAAUGUCCG	(((((.....[[))))) (((((([].....))))))	Jmol 3D	1.036	A A	27 54	43 72	X-Ray Difraction	RNA	17-FEB-09	1.85

Figure 3-4. The output of FASTR3D for a query of RNA tertiary structure.

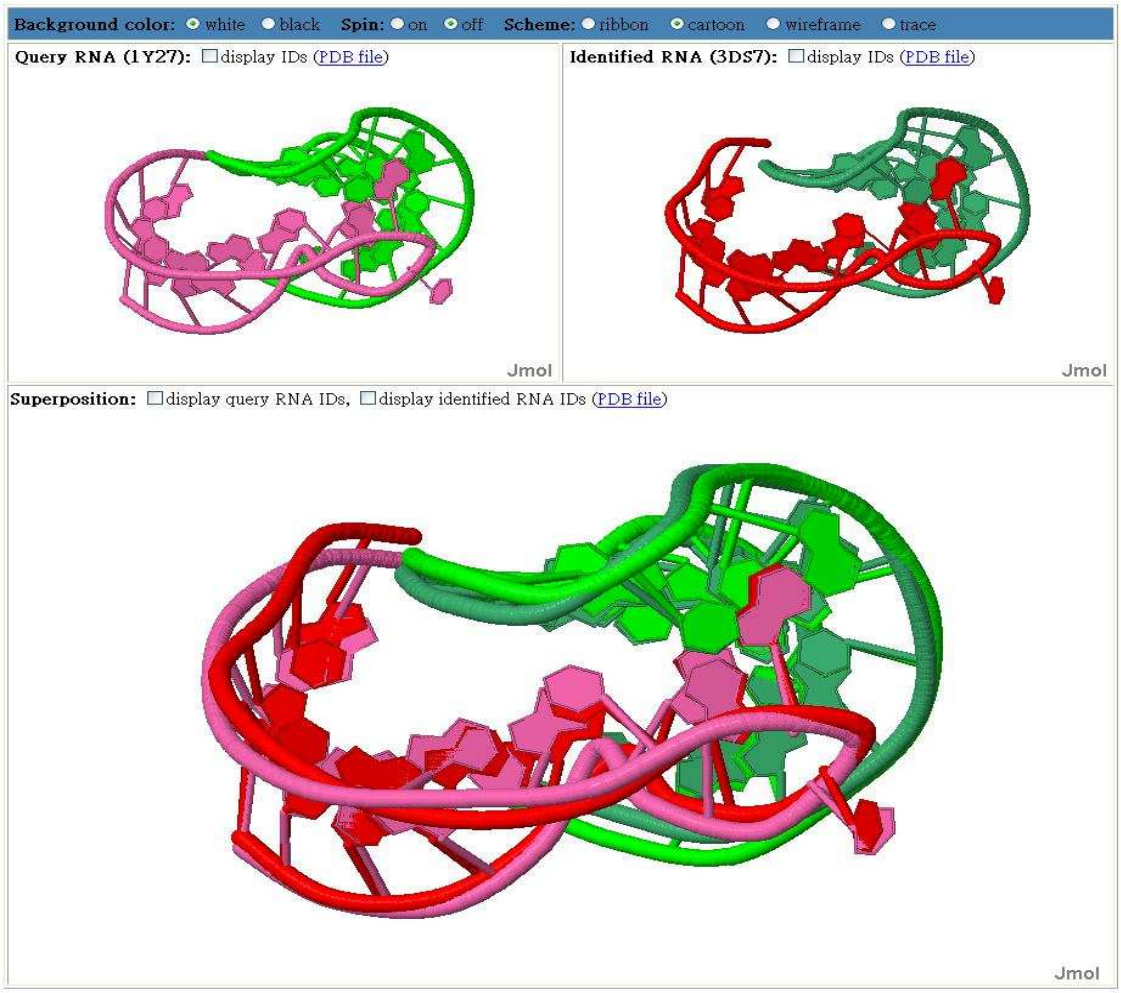

Figure 3-5. The visual display of query RNA (top left panel), an identified RNA (top right panel) and their superposition (bottom panel).

Chapter 4

Experimental Results

For the purpose of evaluation, our FASTR3D was tested with a number of RNA primary sequences and secondary/tertiary structures, and its experimental results on querying RNA primary sequences and secondary structures were also compared with those obtained by RNA FRABASE search engine, developed by Popenda et al [21]. RNA FRABASE on the basis of RNA primary sequence and/or secondary structures using the methods of *regular expression* and *pattern recognition*. It should be noted here that the function of querying RNA tertiary structures in our FASTR3D, as well as the online graphical display of showing the tertiary structures of the identified RNAs and their structural superposition with the query tertiary RNA, is not available in the current RNA FRABASE search program. Basically, our FASTR3D has a comparable performance as RNA FRABASE on querying RNA secondary structures, because the basic principles behind these two tools are the same, even though they were implemented based on different algorithms. As to the queries of RNA primary sequences, the search result of our FASTR3D is greatly different from those obtained by RNA FRABASE. Recall that, when queried with an RNA primary sequence, our FASTR3D searches for query-matching substructures (fragments) within

RNA molecules using the secondary structure information of the query sequence, while RNA FRABASE searches them solely based on the query sequence. As mentioned before, RNA structures are more evolutionarily conserved than their sequences and, therefore, it can be commonly observed that different RNA sequences have the same/similar structures. This indicates that our FASTR3D may be able to find more structurally similar RNA fragments, when compared with RNA FRABASE. For the purpose of demonstration, we selected a fragment from the large subunit of the ribosome in *Haloarcula marismortui* (PDB ID: 1FFK, chain: 0, nucleotide number: 2558–2575) and applied its sequence (GGGGCUG AAGAAGGUCCC) to RNA FRABASE (with default parameters) and our FASTR3D (with searching frequently occurring true secondary structures and without matching the query sequence). Consequently, RNA FRABASE found 51 candidate RNAs that have the same primary sequence as the query, while our FASTR3D found 304 candidates that have the same secondary structure as that of the query derived by the program RNAsubopt. By further verification, we found that 94 out of the 304 tertiary substructures returned by our FASTR3D are highly similar to that of the query. Actually, the above verification was done by running our FASTR3D using the tertiary structure information of the above query sequence (Figure 4-1), where in this resulting list, the first 97 RNAs, which are also in the above list of 304 substructures queried using the primary sequence of the query, have ver similar structures to the query one (with RMSDs less than or equal to 2 angstrom). This experiment demonstrates that the number of structurally similar substructures identified by our FASTR3D is greater than that by RNA FRABASE.

No.	PDB id	Primary Sequence	Secondary Structure	Tertiary Structure	RMSD	Chain	Start	End	Method	Class	Released Date	Å
1	1FFK	GCCCGUGCAGAAGCGGGC	((((((.....))))))	Jmol 3D	0.000	0	2558	2575	X-Ray Difraction	RIBOSOME	14-AUG-00	2.4
2	1S1I	GAGUCGUCACUCGCAAGA	((((((.....))))))	Jmol 3D	0.001	3	2558	2575	Electron Microscopy	RIBOSOME	25-MAY-04	11.7
3	1FG0	GCCCGUGCAGAAGCGGGC	((((((.....))))))	Jmol 3D	0.128	A	2558	2575	X-Ray Difraction	RIBOSOME	28-AUG-00	3
4	1M90	GCCCGUGCAGAAGCGGGC	((((((.....))))))	Jmol 3D	0.135	A	2558	2575	X-Ray Difraction	RIBOSOME	06-SEP-02	2.8
5	1JJ2	GCCCGUGCAGAAGCGGGC	((((((.....))))))	Jmol 3D	0.136	0	2558	2575	X-Ray Difraction	RIBOSOME	01-AUG-01	2.4
...	...	...	...	...	...	...	...	...	...	...	...	...
91	1VOW	GGGGCUGAAGAAGGUCCC	((((((.....))))))	Jmol 3D	1.794	B	2502	2519	X-Ray Difraction	RIBOSOME	16-NOV-04	11.5
92	1VOY	GGGGCUGAAGAAGGUCCC	((((((.....))))))	Jmol 3D	1.794	B	2502	2519	X-Ray Difraction	RIBOSOME	16-NOV-04	11.5
93	1VOU	GGGGCUGAAGAAGGUCCC	((((((.....))))))	Jmol 3D	1.796	B	2502	2519	X-Ray Difraction	RIBOSOME	16-NOV-04	11.5
94	1VP0	GGGGCUGAAGAAGGUCCC	((((((.....))))))	Jmol 3D	1.796	B	2502	2519	X-Ray Difraction	RIBOSOME	16-NOV-04	11.5
95	2GYC	GGUUGGGUAACACUAACU	((((((.....))))))	Jmol 3D	2.562	0	707	724	Electron Microscopy	RIBOSOME	26-SEP-06	2

Figure 4-1. The output of FASTR3D for querying an RNA tertiary structure (PDB ID: 1FFK, chain: 0, nucleotide numbers: 2558-2575).

In the following, we demonstrate the utility of our FASTR3D on querying RNA tertiary structures, which is currently not available in RNA FRABASE. First of all, we used the tertiary substructure of a riboswitch (PDB ID: 1Y27, chain: X, nucleotide numbers: 27–43 and 54–72), as shown in Figure 4-2, to test our FASTR3D for its capability of searching the PDB database for structurally similar riboswitches. The so-called riboswitches are genetic regulatory elements typically found in the non-coding regions of various bacterial mRNAs. They are to regulate the expression of the genes encoded by their downstream mRNAs, via the binding of small metabolites that do not require the assistance of any protein factor [25]. More importantly, it has been suggested by recent studies that riboswitches can serve as antibacterial drug targets, due to their importance to the control of genes in many bacteria [26].

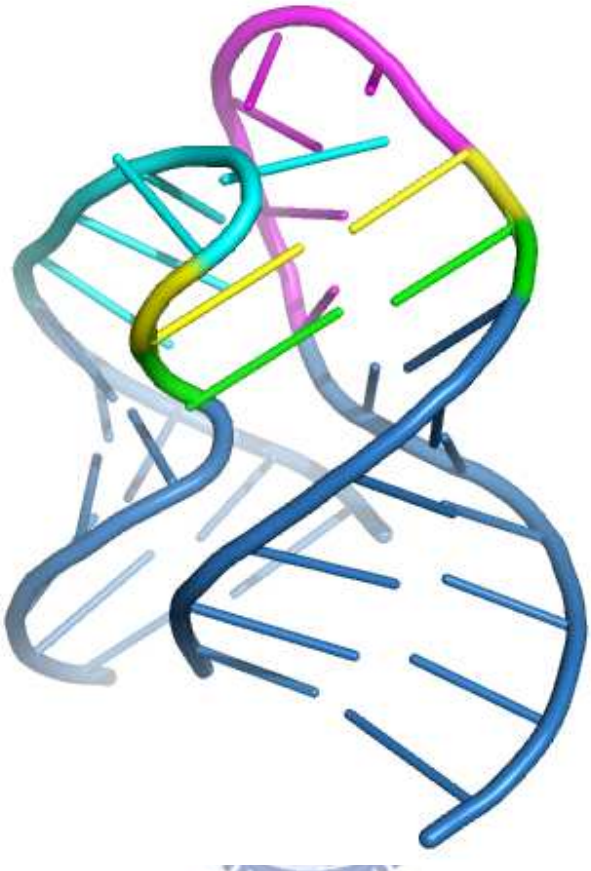

Figure 4-2. The interaction between two hairpin loops from the guanine-responsive riboswitch (PDB ID: 1Y27, chain: X, nucleotide numbers: 27-43 and 54-72). One loop is in cyan and the other is in magenta, with interacting residues in the loops colored yellow and green. Helical stems of the hairpin loops are in blue. This figure was prepared using the program PyMoL (http://www.pymol.org/).

Basically, riboswitches are composed of a ligand binding aptamer domain and an expression platform that interfaces with RNA elements involved in gene expression. Particularly, the aptamer domain for guanine-responsive riboswitches consists of three stems and two hairpin loops. It has been reported that the interaction between these two hairpin loops, as was illustrated in Figure 4-3, is required for the biological function of the guanine-responsive riboswitches [27]. In this experiment, FASTR3D quickly

found other nine riboswitches (PDB IDs: 2G9C, 2B57, 2EEW, 2EEU, 2EES, 1U8D, 2EET, 2EEV and 3DS7) that possess substructures highly similar to the query, where their RMSDs to the query range from 0.98 Å to 1.04 Å (Figure 3-4 for other details). The superposition of the query and the identified substructure in 3DS7 is shown in the bottom panel in Figure 3-5.

Next, we tested our FASTR3D using a frameshifting pseudoknot (PDB ID: 1YG3, chain: A, nucleotide numbers: 3–30) from sugarcane yellow leaf virus (ScYLV), as shown in Figure 4-3a. Programmed –1 ribosomal frameshifting (–1 PRF) is a recoding mechanism by which the translational ribosome switches from the zero reading frame to the –1 reading frame at a specific position and continues its translation in the new frame. The recording of –1 PRF leads to an expression of an alternative protein, which is different from that produced by standard translation. To date, this recoding mechanism has been found to occur in many viruses, as well as a few cellular genes [28 ,29]. The mechanism allows viruses to produce different proteins from the same mRNA and hence increases the diversity of their proteins. In most cases (but not all), the –1 PRF is commonly stimulated by an RNA pseudoknot located downstream from a heptanucleotide slip site where the –1 PRF event takes place. It has been shown that the absence or destabilization of a stable pseudoknot can eliminate efficient stimulation of –1 PRF in ScYLV [30]. In this experiment, FASTR3D quickly found other three RNA pseudoknots (PDB IDs: 1YG4, 2AP0 and 2AP5) in the PDB database whose 3D structures are very similar to that of the query, where their RMSDs to the query is between 1.71 Å and 2.97 Å. Figure 4-3b displays the superposition of the query and the identified pseudoknot 2AP5 whose RMSD is 2.97 Å.

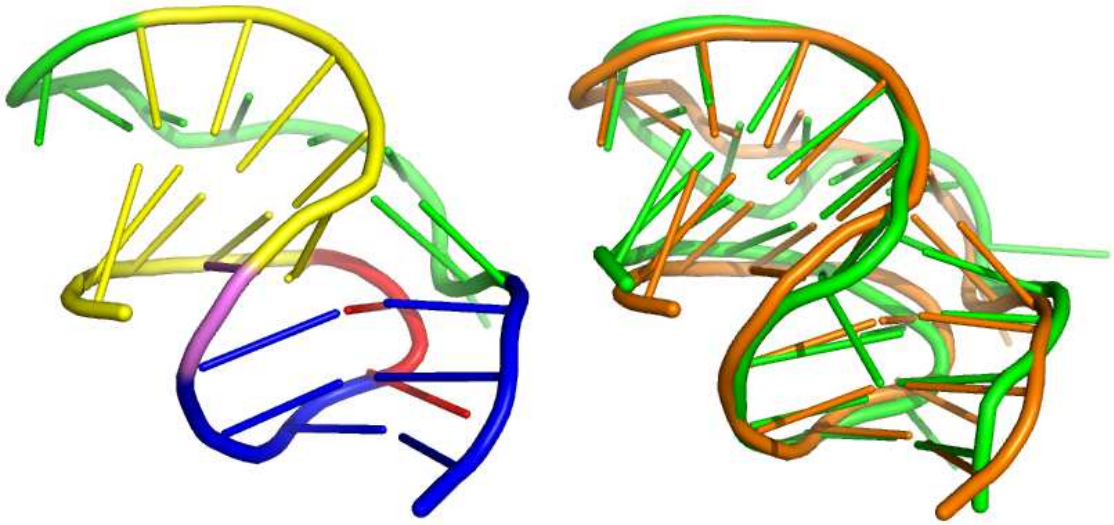

Figure 4-3. (a) Tertiary structure of a frameshifting pseudoknot (PDB ID: 1YG3, chain: A, nucleotide numbers: 3-30). Stem 1 is in yellow, stem 2 is in blue, loop 1 is in red, loop 2 is in green and the nucleotide (A13) between the two stems is in violet. (b) The superposition between the query pseudoknot (1YG3) colored orange and an identified pseudoknot (2AP5) colored green with an RMSD of 2.97 Å.

For more detailed experimental results of our FASTR3D and their comparisons with RNA FRABASE, refer to the following Table 4-1 on querying RNA primary sequences, Table 4-2 on querying RNA secondary structures, and Table 4-3 on querying RNA tertiary structures.

Table 4-1. Comparison of FASTR3D and RNA FRABASE on querying RNA primary sequences.

Query RNA	RNA FRABASE		FASTR3D	
Testing example	Result	Time (h/m/s)	Result	Time (h/m/s)
>Query ACUGGCCGUGA AGAUGCGGCC	2.7 sec.		0.34 sec.	
>Query GUCGGGUAAGU UCCGAC	2.9 sec.		0.37 sec.	

Table 4-2. Comparison of FASTR3D and RNA FRABASE on querying RNA secondary structures.

Query RNA	RNA FRABASE	FASTR3D
Testing example	Time (h/m/s)	Time (h/m/s)
>Query GAGGNRACUC (((....)))	3.3 sec.	0.66 sec.
>Query GUCGGGUAAGU UCCGAC	4.2 sec.	0.72 sec.

Table 4-3. Comparison of FASTR3D and RNA FRABASE on querying RNA tertiary structures.

Query RNA	FASTR3D	
Testing example	Result	Time (h/m/s)
>Query 2HHH A,595,612	3.80 sec.	
>Query 1Q82 B,3004,3037 B,3043,3089 B,3094,3119	1.97 sec.	

Basically, when queried with RNA primary sequences, our FASTR3D can provide more unintended structures than RNA FRABASE as the query sequences are not as conserved as their secondary structures. On the other hand, the search results by our FASTR3D using RNA tertiary structures have the intended structures with more various sequences than those by RNA FRABASE using their primary sequences and secondary structures as the input.

Chapter 5

Conclusions

FASTR3D is a web-based search tool that allows the user to quickly and accurately search the PDB database for structural similarities of a query RNA. The user can query this tool by using either an RNA tertiary structure, an RNA secondary structure or an RNA primary sequence. Since the hashing algorithm, as well as tertiary structure filter, behind our FASTR3D is highly efficient, a typical query can be done in a short time. It is worth mentioning again that the function of querying RNA tertiary structures in our FASTR3D, as well as the online graphical display of showing structural superposition, is not available in RNA FRABASE. Therefore, we believe that our FASTR3D can serve as a useful tool in the study of structural biology.

Reference

[1] Doudna, J. A. (2000) Structural genomics of RNA. *Nat. Struct. Biol.*, **7**, 954–956.

[2] Eddy, S. R. (2001) Non-coding RNA genes and the modern RNA world. *Nat. Rev. Gene.t*, **2**, 919–929.

[3] Mattick, J. S, Makunin, I. V. (2006) Non-coding RNA. *Hum. Mol. Genet.*, **15**, R17–R29.

[4] Storz, G. (2002) An expanding universe of noncoding RNAs. *Science*, **296**, 1260–1263.

[5] He, S., Liu, C., Skogerbo, G., Zhao, H., Wang, J., Liu, T., Bai, B., Zhao, Y. and Chen, R. (2008) NONCODE v2.0: decoding the non-coding. *Nucleic Acids Res*., **36**, D170 – D172.

[6] Pang, K. C., Stephen, S., Dinger, M. E., Engstrom, P. G., Lenhard, B. and Mattick, J. S. (2007) RNAdb 2.0–an expanded database of mammalian non-coding RNAs. *Nucleic Acids Res.*, **35**, D178–D182.

[7] Griffiths-Jones, S., Saini, H. K., van Dongen, S. and Enright, A. J. (2008) miRBase: tools for microRNA genomics. *Nucleic Acids Res.*, **36**, D154–D158.

[8] Kin, T., Yamada, K., Terai, G., Okida, H., Yoshinari, Y., Ono, Y., Kojima, A., Kimura, Y., Komori, T. and Asai, K. (2007) fRNAdb: a platform for mining/annotating functional RNA candidates from

non-coding RNA sequences. *Nucleic Acids Res.*, **35**, D145–D148.

[9] Szymanski, M., Erdmann, V. A. and Barciszewski, J. (2007) Noncoding RNAs database (ncRNAdb). *Nucleic Acids Res.*, **35**, D162–D164.

[10] Berman, H. M., Westbrook, J., Feng, Z., Gilliland, G., Bhat, T. N., Weissig, H., Shindyalov, I. N. and Bourne, P. E. (2000) The protein data bank. *Nucleic Acids Res.*, **28**, 235–242.

[11] Berman, H. M., Westbrook, J., Feng, Z., Iype, L., Schneider, B. and Zardecki, C. (2002) The nucleic acid database. *Acta Crystallogr. D Biol. Crystallogr.*, **58**, 889–898.

[12] Kolodny, R. and Linial, N. (2004) Approximate protein structural alignment in polynomial time. *Proc. Natl Acad. Sci. USA*, **101**, 12201–12206.

[13] Dror, O., Nussinov, R. and Wolfson, H. (2005) ARTS: alignment of RNA tertiary structures. *Bioinformatics*, **21 (Suppl. 2)**, 47–53.

[14] Dror, O., Nussinov, R. and Wolfson, H. J. (2006) The ARTS web server for aligning RNA tertiary structures. *Nucleic Acids Res.*, **34**, W412–W415.

[15] Ferrè, F., Ponty, Y., Lorenz, W. A. and Clote, P. (2007) DIAL: a web server for the pairwise alignment of two RNA three-dimensional structures using nucleotide, dihedral angle and base-pairing similarities. *Nucleic Acids Res.*, **35**, W659–W668.

[16] Chang, Y. F., Huang, Y. L. and Lu, C. L. (2008) SARSA: a web tool

for structural alignment of RNA using a structural alphabet. *Nucleic Acids Res.*, **36**, W19–W24.

[17] Capriotti, E. and Marti-Renom, M. A. (2008) RNA structure alignment by a unit-vector approach. *Bioinformatics*, **24**, i112–i118.

[18] Sarver, M., Zirbel, C. L., Stombaugh, J., Mokdad, A. and Leontis, N. B. (2008) FR3D: finding local and composite recurrent structural motifs in RNA 3D structures. *J. Mol. Biol.*, **56**, 215–252.

[19] Duarte, C. M., Wadley, L. M. and Pyle, A. M. (2003) RNA structure comparison, motif search and discovery using a reduced representation of RNA conformational space. *Nucleic Acids Res.*, **31**, 4755–4761.

[20] Macke, T. J., Ecker, D. J., Gutell, R. R., Gautheret, D., Case, D. A. and Sampath, R. (2001) RNAMotif, an RNA secondary structure definition and search algorithm. *Nucleic Acids Res.*, **29**, 4724–4735.

[21] Popenda, M., Blazewicz, M., Szachniuk, M. and Adamiak, R. W. (2008) RNA FRABASE version 1.0: an engine with a database to search for the three-dimensional fragments within RNA structures. *Nucleic Acids Res.*, **36**, D386–D391

[22] Yang, H., Jossinet, F., Leontis, N., Chen, L., Westbrook, J., Berman, H. and Westhof, E. (2003) Tools for the automatic identification and classification of RNA base pairs. *Nucleic Acids Res.*, **31**, 3450–3460.

[23] Duarte, C. M. and Pyle, A. M. (1998) Stepping through an RNA structure: a novel approach to conformational analysis. *J. Mol. Biol.*, **284**, 1465–1478.

[24] Wuchty, S., Fontana, W., Hofacker, I. L. and Schuster, P. (1999) Complete suboptimal folding of RNA and the stability of secondary structures. *Biopolymers*, **49**, 145–165.

[25] Mandal, M. and Breaker, R. R. (2004) Gene regulation by riboswitches. *Nat. Rev. Mol. Cell Biol.*, **5**, 451–463.

[26] Blount, K. F. and Breaker, R. R. (2006) Riboswitches as antibacterial drug targets. *Nat. Biotechnol.*, **24**, 1558–1564.

[27] Batey, R. T., Gilbert, S. D. and Montange, R. K. (2004) Structure of a natural guanine-responsive riboswitch complexed with the metabolite hypoxanthine. *Nature*, **432**, 411–415.

[28] Farabaugh, P. J. (1996) Programmed translational frameshifting. *Microbiol. Rev.*, **60**, 103–134.

[29] Namy, O., Rousset, J. P., Napthine, S. and Brierley, I. (2004) Reprogrammed genetic decoding in cellular gene expression. *Mol. Cell*, **13**, 157–168.

[30] Cornish, P. V., Hennig, M. and Giedroc, D. P. (2005) A loop 2 cytidine-stem 1 minor groove interaction as a positive determinant for pseudoknot-stimulated -1 ribosomal frameshifting. *Proc. Natl Acad. Sci. USA*, **102**, 12694–12699.