Ciphers and Encryption in MATLAB

A Brief Investigation...

Sasha Egan (900307509)
New Mexico Institute of Mining and Technology
segan@nmt.edu

Supervisor: Billy Oxford
New Mexico Institute of Mining and Technology

(Dated: April 30, 2013)

A brief investigation of encryption and ciphers in MATLAB was attempted by adapting a classic poly-alphabetic cryptographic cipher, Vigenère , and a relatively modern stream cipher, RC4, into the MATLAB coding environment.

I. INTRODUCTION

"The urge to discover secrets is deeply ingrained in human nature; even the least curious mind is roused by the promise of sharing knowledge withheld from others. Some are fortunate enough to find a job which consists in the solution of mysteries, but most of us are driven to sublimate this urge by the solving of artificial puzzles devised for our entertainment. Detective stories or crossword puzzles cater for the majority; the solution of secret codes may be the pursuit of a few. " - John Chadwick (*TheDeciphermentofLinearB*) [2]

The desire for the ability to protect secrets is a centuries old preoccupation. As language and technology has evolved so has the development of codes and ciphers, an everlasting battle between those who create codes and those who break them. The struggle for dominion over information has become the most terrifying war ever to touch mankind to date, for too many lives depend on the proper people, in their proper places, knowing what they absolutely need to know. When that process fails, disaster and death ensue. Technology has become a most effective weapon. MATLAB is export controlled for a reason. It allows, with absolute ease, for the implementation of complex and resource intensive routines without the fuss of mainframes and high-power computing. Two ciphers were investigated using the MATLAB runtime, Vigenère , and RC4.

II. POLY-ALPHABETIC SUBSTITUTION CIPHERS

A. Transposition Ciphers

To understand the inner workings of Vigenère , it would be helpful to understand what a mono-alphabetic substitution cipher consists of. A mono-alphabetic substitution cipher, also known as a *transpositioncipher*, is for lack of a better explanation–*aword jumble*. It may be difficult to believe but the same thing our parents gave us out of the Saturday morning *Times* to keep us from plaguing them to death, was once considered the pinnacle of encryption technology. A transposition cipher is a rearrangement of the letters of the message in a way that is designed to confuse the attacker but can be 'unjumbled' by the recipient (or any other eight year old child). The concept is important as it still forms the basis of modern ciphers.

Here is an example of a transpostion cipher:

THIS IS SECRET MATE
TICM HSRA ISET SETE

TABLE I: Transpostion Cipher of Key(4)

The cipher in Table(I) is an example of Scytale. The earliest recorded uses of cryptography was the Spartan *scytale* circa (500 B.C.). A thin strip of parchment was wrapped helically around a cylindrical rod and the message was written across the rod, with each letter on a successive turn of the parchment.[1] For the Scytale cipher, the key is the rod or more specifically the *diameter* of the rod, which in this case is 4. This example also fits the definition of Columnar Transposition where putting the plaintext into rows of some given dimension (4-by-4) reveals the message.

$$\begin{bmatrix} T & I & C & M \\ H & S & R & A \\ I & S & E & T \\ S & E & T & E \end{bmatrix}$$

As MATLAB is exceptionally good at manipulating matrices and vectors, application of such cipher systems is straightforward. In-built routines like 'circshift', will cycle entire vectors n-spaces giving other means of enciphering by shifting characters in a cyclic manner. Cipher methods used in the days of USENET, such as rot13, did just this and require only a single line of code in MATLAB to rearrange the contents of a vector. Expanding from a mono-alphabetic substitution to poly-alphabetic substitution simply builds upon this method.

The Vigenère cipher builds upon the method of mono-alphabetic substitution by expanding the number of alphabets to twenty-six, if English is the language being encoded. In Table(II) it can be seen that an application of a large vector with values of 1 to 26 and singularly shifted, results in producing the permuation table from which a message using Vigenère can be encoded. Vigenère works by using a key or keyword which is expanded to the length of the message using the modulus index of the key. Each letter within the ex-

panded key becomes the index character for which the message is encoded. This method greatly reduces the frequency of which commonly used characters from within the English language appear and makes breaking a message a bit more difficult. It was not until the early 1900s that a method of attacking Vigenère was clearly defined. While working at the Riverbank Laboratory, William F. Friedman developed the *indexofcoincidence*. The IOC was a powerful definition of the probability that two randomly selected letters in the ciphertext represented the same symbol. (See the book Applied Cryptanalysis page 14 for the mathematical details).

ZABCDEFGHIJKLMNOPQRSTUVWXY
YZABCDEFGHIJKLMNOPQRSTUVWX
XYZABCDEFGHIJKLMNOPQRSTUVW
WXYZABCDEFGHIJKLMNOPQRSTUV
VWXYZABCDEFGHIJKLMNOPQRSTU
UVWXYZABCDEFGHIJKLMNOPQRST
TUVWXYZABCDEFGHIJKLMNOPQRS
STUVWXYZABCDEFGHIJKLMNOPQR
RSTUVWXYZABCDEFGHIJKLMNOPQ
QRSTUVWXYZABCDEFGHIJKLMNOP
PQRSTUVWXYZABCDEFGHIJKLMNO
OPQRSTUVWXYZABCDEFGHIJKLMN
NOPQRSTUVWXYZABCDEFGHIJKLM
MNOPQRSTUVWXYZABCDEFGHIJKL
LMNOPQRSTUVWXYZABCDEFGHIJK
KLMNOPQRSTUVWXYZABCDEFGHIJ
JKLMNOPQRSTUVWXYZABCDEFGHI
IJKLMNOPQRSTUVWXYZABCDEFGH
HIJKLMNOPQRSTUVWXYZABCDEFG
GHIJKLMNOPQRSTUVWXYZABCDEF
FGHIJKLMNOPQRSTUVWXYZABCDE
EFGHIJKLMNOPQRSTUVWXYZABCD
DEFGHIJKLMNOPQRSTUVWXYZABC
CDEFGHIJKLMNOPQRSTUVWXYZAB
BCDEFGHIJKLMNOPQRSTUVWXYZA
ABCDEFGHIJKLMNOPQRSTUVWXYZ

TABLE II: Vigenère Poly-alphabetic Permutation Table

The goal of this project was to duplicate the encoding algorithm in MATLAB to encipher plaintext and then to subsequently decipher it. The enciphering algorithm was slightly more complex than was initially anticipated strictly due to the reason that MATLAB indexes characters by the numerical index of their ASCII value. Where 26 might refer to *zed* in the English language to MATLAB this represents a character which cannot be displayed. The key to making Vigenère work is that the index of key's letter and where the corresponding ciphertext letter resides must have an index which can be readily reversed. If a shift in the alphabet occurs, then the index of the keyword will not correspond with the correct piece of ciphertext or plaintext, the index of the enciphering alphabet is incorrect and the sequence of the message is disrupted. An offset was determined to translate between the ASCII value of the plaintext, ciphertext, and the indexing method of the individual alphabets. This offset allowed a 26-by-26 matrix to serve as the index of the alphabet while allowing the creation of a table of ASCII values that corresponded with the characters to be encoded or decoded. Another aspect which proved to be a difficulty was computing the modulus of the key with respect to the indexing. Within MATLAB the array index requires positive integers from 1 to $1+n$ while modulus addition requires the value of zero to be included as the index and it repetitively crops up at regular intervals depending upon the length of the key being computed. A solution was found to this problem by carefully calculating the jumps using the modulus of the key but either adding or subtracting a value of 1 before the contents of a specific index was modified.

B. Stream Cipher

RC4 is a stream cipher. In cryptography, a stream cipher is a symmetric key cipher where plaintext bits are combined with a pseudorandom cipher bit stream. RC4 was invented by Ron Rivest in 1987 and is without a doubt the most widely used cipher in the world today. It is used in the Secure Sockets Layer (SSL), which is the defacto standard for secure transactions over the Internet.[1, p.103]The RC4 algorithm is considered secure, if used properly. WEP, or Wired Equivalent Privacy, a networking protocol which has been deemed unsafe, managed to implement nearly all of its security functions insecurely, including RC4. Preventing the attack on RC4 is implemented in by generating 256 or more keystream bytes and discarding them prior to beginning encryption of the bytestream. As long as both the sender and receiver follow this, no modification of the inner workings of RC4 is required. There are several other ways to prevent the attack one of which is implemented in RC4Plus.

The goal was to implement the RC4 algorithm in MATLAB with these modifications. RC4 begins by populating a 256 byte table of linear values. These values are pseudorandomised using an initialisation vector provided by the user, pseudo randomising all the values destroys most, if not all, of the linearity of the original population. A byte value from this State (S) table is then used to encipher the plaintext byte by the xor operation, bit-by-bit. The resulting output becomes the key-lookup value which is then found in the state table and becomes the cipherbyte value. Like the permutation table used in Vigenère , the symmetry of the xor operation can be reversed and the original plaintext value can be recovered. However, unlike Vigenère , in this algorithm the state table constantly shifts the values around continually randomising the state of the table. As long as the initialisation vector remains secure then reconstruction of the exact state of the table used to generate the cipherbyte is extremely difficult. Stream ciphers are very efficient and fast however they have a drawback in that they encrypt every single byte individually and as the keylength increases so does the time it takes to complete an encryption process.

C. Conclusion

Two ciphering systems were implemented in MATLAB, one classic and one modern. This proves that the versatility and power of the MATLAB runtime is not limited to mindless calculator operations. The only limitation is derived from the restrictions and semantics of the language itself and the ingenuity of the programmer. The RC4 algorithm as is implemented in this project also cannot be utilised to encrypt specialised or packed binary files the resulting output results in a corrupted file which cannot be opened by the parent program. The algorithm doe encrypt text files and strings quite well. Short animation sequences can be turned on by setting the animation flags in specific procedures; however I found them quite boring after the initialisation procedure. I also am not a cryptographer therefore I cannot vouch for the strength of the implemented algorithm and no security vectors were evaluated.

[1] Richard M. Low Mark Stamp. *Applied Cryptanalysis*. Wiley-Interscience, John Wiley [and] Sons INC., Hoboken, New Jersey, 2007.

[2] Simon Singh. *The Code Book*. Random House, Inc, Fourth Estate, London, UK, 1999.