ПРИМЕНЕНИЕ ГЕНЕТИЧЕСКИХ АЛГОРИТМОВ ДЛЯ КРИПТОАНАЛИЗА ПОДСТАНОВОЧНЫХ ШИФРОВ

Фролов А.О.

Белорусский государственный университет информатики и радиоэлектроники
г. Минск, Республика Беларусь

Прохорчик Р.В. – ст. преп., м. т. н.

В данной работе рассматривается иной подход к криптоанализу подстановочных шифров, основанный на применении генетических алгоритмов. Показывается, что такие алгоритмы могут быть использованы для обнаружения ключа простых шифров подстановок. В качестве существенного фактора целевой функции генетического алгоритма используется частотный анализ.

В настоящее время безопасность является одной из важнейших проблем в информатике, криптография широко используется для ее реализации. Задачу, противоположную криптографии, решает криптоанализ – процесс, в котором пытаются нарушить безопасность. В свою очередь, сложность этого процесса рассматривается как мера безопасности. Поскольку криптографические алгоритмы открыты для всех, вся мощность алгоритма заключается в сложности ключа, сила которого в общем случае определяется его длиной.

Данная работа посвящена криптоанализу подстановочных шифров, которые являются одними из древнейших. Шифры такого рода явным или неявным образом используют таблицы подстановок. Рассмотрим в качестве шифра подстановки достаточно популярный шифр Виженера и покажем, насколько применим генетический алгоритм для его криптоанализа.

Генетические алгоритмы – это класс эвристических алгоритмов, которые пытаются решить задачу путем моделирования упрощенной версии генетических процессов. Такого рода алгоритмы основаны на концепциях генетики и эволюции. Генетический алгоритм функционирует на основе популяции выбранных организмов, в генном материале которых (генотип) закодирован ответ на поставленный вопрос (фенотип). Для простых задач пространства фенотипов и генотипов одинаковы, однако в большинстве случаев эти пространства различны. Генетический алгоритм начинается с формирования изначального популяционного материала на основании произвольной выборки. В дальнейшем на основании аналитики дается оценка популяции по уровню приспособленности к среде обитания или заданным факторам. Затем самые приспособленные особи скрещиваются. Таким образом, популяционные особи дают потомство, которое имеет высокую вероятность сохранить необходимый генотип в своем и следующих поколениях. Потомство изначальной популяции создает следующую по хронологии популяцию, причем у большинства особей должен сохраняться на протяжении всей жизни выбранный генотип, а минимальное число особей подвержены генной мутации. Отметим, что каждое множество особей популяции в определенный промежуток времени – это поколение. Смена поколений образует популяционную эволюцию. При этом каждый этап эволюционного процесса может иметь разную длительность, что обусловлено различной скоростью поиска правильного решения, ограничением на количество возможных поколений, а также другими факторами.

Стоит отметить, что генетические алгоритмы не всегда дают точный ответ, а скорее дают решение, близкое к правильному. Несмотря на неточность, генетические алгоритмы показывают хорошие результаты при решении оптимизационных задач и поиске ответов на такого рода вопросы. По той причине, что целевая установка взлома шифра – это выявление единственно верного секретного ключа в ряду потенциально возможных ключей, то задача криптографического анализа может быть представлена в виде оптимизационной задачи. Таким образом, в поиске решения такой задачи может быть применен генетический алгоритм.

В первую очередь, при создании генетического алгоритма важно выбрать оптимальный, эффективный способ кодирования и описать основные операторы алгоритма – селекция, кроссинговер и мутация. На качественную составляющую алгоритма ключевое значение оказывает популяционная численность, количество поколений, значения уровня вероятности для базовых операторов.

В случае расшифровки зашифрованного текста каждое решение хранит ключ k длины n, а операторы мутации и кроссинговера изменяют k. Для реализации расшифровки текста была выбрана малая вероятность мутации (0.25), а уровень вероятности наследования генотипа от одного поколения к другому была, в свою очередь, выбрана достаточно большой (0.85). Оператор кроссинговера, используемый в данной работе, представляет собой двухточечный кроссинговер. Для выбора наиболее приспособленных особей для скрещивания была выбрана турнирная селекция.

Одной из главных задач при использовании генетических алгоритмов для расшифрования является выбор целевой функции. Реализовать хорошую целевую функцию довольно нетривиально. Языковая статистика и атаки по словарю должны противостоять этой проблеме. Весьма интересный подход к вычислению целевой функции описан в работе Э. Кларка [1]. Суть этого подхода

заключается в сравнении n-граммной статистики расшифрованного сообщения с данными языка, которые считаются известными. Уравнение, являющееся общей формулой, используемой для определения пригодности предполагаемого ключа k, имеет следующий вид:

$$f(k) = \alpha \cdot \sum_{i \in A} \left| D_i^u(k) - S_i^u \right| + \beta \cdot \sum_{i,j \in A} \left| D_{i,j}^b(k) - S_{i,j}^b \right| + \gamma \cdot \sum_{i,j,l \in A} \left| D_{i,j,l}^t(k) - S_{i,j,l}^t \right|, \quad (1)$$

где A – алфавит языка, D – частота появления n-граммы в анализируемом текстовом фрагменте, который получен из начального зашифрованного текста путем его расшифровки с применением предполагаемого ключа, обозначенного буквой k, S – заданный средний уровень частоты повторения n-граммы в различных текстовых источниках, а индексы u, b и t обозначают статистику униграмм, биграмм и триграмм соответственно. Значения α, β и γ позволяют присваивать различные веса каждому из трех типов n-грамм. Целевая функция опирается на статистическую характеристику языка, чтобы определить пригодность ключа. Таким образом, выбор меры пригодности полностью зависит от языковых характеристик и, следовательно, эти характеристики должны быть известны, как и сам используемый язык.

Разработанный на основе вышеизложенного алгоритм, позволяющий проводить криптографический анализ шифра Виженера, работает на заранее заданной длине ключа шифрования и находит с определенной точностью сам секретный ключ. Выбранная функция пригодности является основным фактором алгоритма. Текст, получаемый в результате работы алгоритма, не сильно отличается от исходного текста. В большинстве случаев отдельные символы легко восстанавливаются с помощью контекста. Результаты проведенных опытов дают возможность сказать, что в случае малой численности популяции результаты получаются не оптимальными, что продемонстрировано на рисунке 1. Это связано с тем, что генетический алгоритм достаточно быстро определяет место локального экстремума функции. Очевидно, что число поколений сильно зависит от длины ключа и, следовательно, с увеличением длины ключа нужно увеличивать число последовательно идущих друг за другом поколений. Соблюдение обозначенного условия дает возможность получить наиболее точное решение. Изменения основных генетических операторов алгоритма в допустимых пределах не дали улучшения результатов.

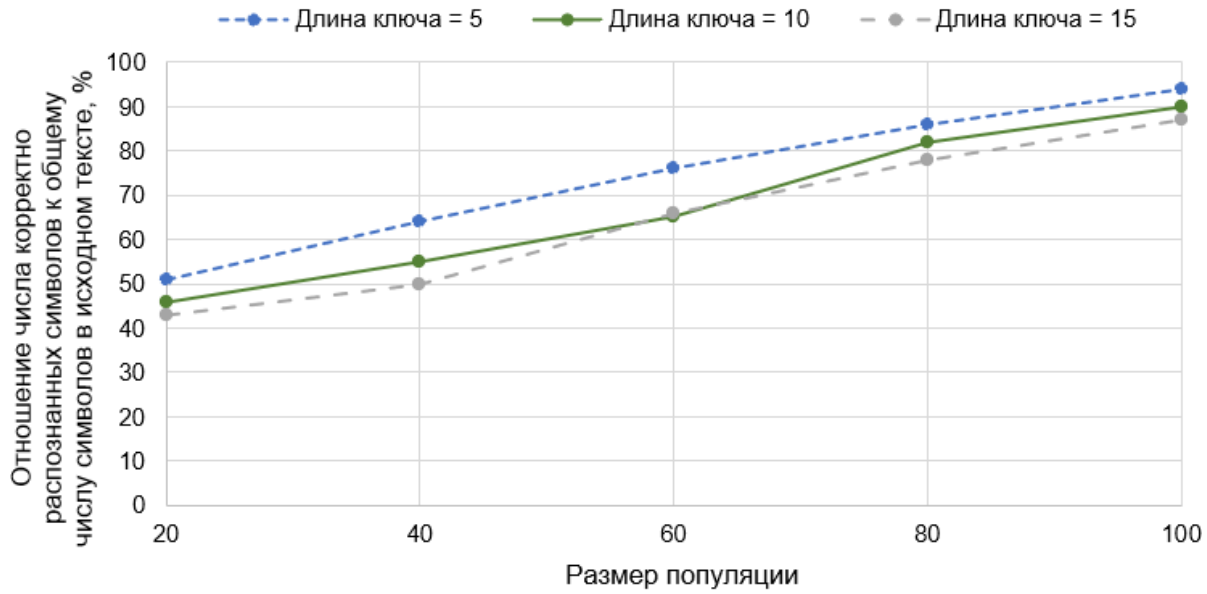

Рисунок 1 – Процентное соотношение корректных символов при различных размерах популяции

В данной работе была успешно реализована атака генетического алгоритма на простой криптографический шифр подстановки. Несмотря на то, что не всегда удается полностью автоматизировать процесс решения задачи криптоанализа, алгоритм в значительной степени помогает расшифровать зашифрованный текст. В ходе работы были протестированы различные параметры, такие как численность популяции, число поколений, а также вероятностные характеристики основных операторов. Использование генетического алгоритма для криптоанализа простого шифра подстановки оказалось эффективным методом криптоанализа, основанным на аспекте сравнения частот встречаемости букв. Так как подход, основанный на применении генетического алгоритма, оказался успешным для простого шифра подстановки, то имеет смысл применить тот же подход к более современным и сложным шифрам.

Список использованных источников:

1. Clark, A. Modern optimisation algorithms for cryptanalysis. In Proceedings of the 1994 Second Australian and New Zealand Conference on Intelligent Information Systems, November 29 – December 2 1994, p. 258-262.