```
Sum = Sum + Sqr(Базовая_Последовательность[j] – Все_Записи[i])
i++
  End For
  Distance = Sqrt(Sum)
End While
```

Затем полученные расстояния сортируются по возрастанию и выбирается К наименьших расстояний. Итоговый прогноз вычисляется как среднее значение К образцов для выбранного атрибута и дня.

Данный метод показывает очень точные результаты, которые могут использоваться в реальной жизни. Так, для дискретных значений таких показателей как туман, град, снег, гололедица, гроза и т.д. на небольших наборах данных точность составила свыше 90%. На больших массивах данных точность повышается до 95%.

Недостатком данного метода является тот факт, что он не учитывает глобальные изменения климата (ENSOevents). Однако, данный метод прекрасно работает для областей, которые не подвержены таким изменениям.

Список использованных источников:

1. Data Mining for Climate Change and Impacts – A.R.Ganguly, K.Steinhaeuser - 2008 IEEE International Conference on Data Mining Workshops.

2. Метод ближайших соседей [Электронный ресурс]. – 2018. – Режим доступа: https://www.ibm.com/support/knowledgecenter/ru/SSLVMB_24.0.0/spss/base/idh_idd_knn_variables.html Дата доступа: 21.03.2018.

МЕТОДЫ ВОЗВЕДЕНИЯ ЧИСЛА В СТЕПЕНЬ ПО МОДУЛЮ

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

Короткевич А.В.

Ярмолик В.Н. – д.т.н., профессор

Одной из наиболее важных операций в ассиметричной криптографии является возведение числа в степень. Т.к. операция выполняется в конечном поле, то фактически задача сводится к нахождению $g^e(modp)$. Очевидно, что простейшим способом решения является выполнение $e-1$ умножения, однако, такой способ неприемлем, когда речь идет о числах большой разрядности. Эффективные методы выполнения возведения числа в степень по модулю будут рассмотрены в данной работе.

Существует два способа уменьшения времени выполнения операции возведения в степень в конечном поле. Во-первых, это уменьшение времени выполнения умножения двух элементов группы. Вторым способом является уменьшение требуемого числа операций умножения. В идеале оба подхода должны быть использованы одновременно [1].

В качестве решения, использующего уменьшение числа операций умножения при возведении в степень, можно привести следующий алгоритм:

Алгоритм 1 –Бинарное возведение в степень справа налево

INPUT: Целые числа g и $e \geq 1$.

OUTPUT: g^e.

1) $A \leftarrow 1, S \leftarrow g$.

2) While $e \neq 1$ do:

2.1) If e нечетное then $A \leftarrow A \cdot S$.

2.2) $e \leftarrow \lfloor e/2 \rfloor$.

2.3) If $e \neq 0$ then $S \leftarrow S \cdot S$.

3) Return A.

Приведенный алгоритм использует последовательные операции умножения и возведения в квадрат, что фактически тоже является умножением. Пусть $t+1$ – длина в битах бинарного представления числа e, а l– число единиц в данном представлении. Тогда, согласно алгоритму, потребуется t возведений в степень и $l-1$ операция умножения. Если e является случайным числом, то число возведений в степень будет приблизительно $\lfloor \log_2 e \rfloor$, а умножений $-0.5 \cdot (\lfloor \log_2 e \rfloor + 1)$. Таким образом, алгоритм позволяет сократить число операций с $e-1$ для самого простого решения до приблизительно $1.5 \cdot \log_2 e$, что является хорошим показателем.

Алгоритм 1 вычисляет $A \cdot S$ всякий раз, когда eявляется нечетным. Для некоторых значений g выражение $A \cdot g$может быть вычислено более эффективно, чем $A \cdot S$ для случайного S. Алгоритм 2 выполняет бинарное возведение в степень слева направо, которое заменяет операцию $A \cdot S$ (для случайного S) на $A \cdot g$(для фиксированного g).

Алгоритм 2 –Бинарное возведение в степень слева направо

INPUT: Целое g и положительное целое $e = (e_t e_{t-1} \dots e_1 e_0)_2$.

OUTPUT: g^e.

1) $A \leftarrow 1$.

2) For i from t down to 0 do:

2.1) $A \leftarrow A \cdot A$.

2.2) If $e_i = 1$, then $A \leftarrow A \cdot g$.

3) Return A.

Пусть $t+1$ – длина в битах бинарного представления числа e, а l– число единиц в данном представлении. Тогда, согласно алгоритму, потребуется t возведений в степень и $l-1$ операция умножения на g. Число возведений и умножений остается таким же, как и в алгоритме 1, но в алгоритме 2 умножение всегда производится на фиксированное значение g. Если g имеет особую структуру, то это умножение может быть гораздо более легким, чем умножение двух случайных значений.

Оптимизацией приведенных алгоритмов являются оконные методы возведения в степень, которые на одной итерации обрабатывают более одного бита значения степени одновременно [2]. Разновидность такого алгоритма может иметь следующий вид:

Алгоритм 3 –Возведение в степень слева направо с помощью окна ширины k

INPUT: Целые g и $e = (e_t e_{t-1} \dots e_1 e_0)_b$, где $b = 2^k$ для некоторого $k \geq 1$.

OUTPUT: g^e.

1) Предварительные вычисления.

1.1) $g_0 \leftarrow 1$.

1.2) For i from 1 to $(2^k - 1)$ do: $g_i \leftarrow g_{i-1} \cdot g$. (Таким образом, $g_i = g^i$)

2) $A \leftarrow 1$.

3) For i from t down to 0 do:

3.1) $A \leftarrow A^{2^k}$.

3.2) $A \leftarrow A \cdot g^{e_i}$.

4) Return A.

В алгоритме 4, по сравнению с алгоритмом 3, уменьшается количество предварительных вычислений, а также сокращается среднее число используемых операций умножения (исключая возведение в квадрат). k называется шириной окна.

Алгоритм 4 –Возведение в степень с использованием скользящего окна

INPUT: Целые g, $e = (e_t e_{t-1} \dots e_1 e_0)_2$ с $e_t = 1$, $k \geq 1$.

OUTPUT: g^e.

1) Предварительные вычисления.

1.1) $g_1 \leftarrow g$, $g_2 \leftarrow g^2$.

1.2) For i from 1 to $(2^{k-1} - 1)$ do: $g_{2i+1} \leftarrow g_{2i-1} \cdot g_2$.

2) $A \leftarrow 1$, $i \leftarrow t$.

3) While $i \geq 0$ do:

3.1) If $e_i = 0$ then do: $A \leftarrow A \cdot A$, $i \leftarrow i - 1$.

3.2) Else найти наибольший набор бит $e_i e_{i-1} \dots e_l$ такой, что $i - l + 1 \leq k$ и $e_l = 1$, и do: $A \leftarrow A^{i-l+1} \cdot g(e_i e_{i-1} \dots e_l)_2$, $i \leftarrow l - 1$.

4) Return$A$.

Отметим, что метод скользящего окна для данного возведения в степень требует трех операций умножения, соответствующих $i = 7,\ 4$ и 0. А алгоритму 2 понадобится 4 операции умножения для тех же самых k и e.

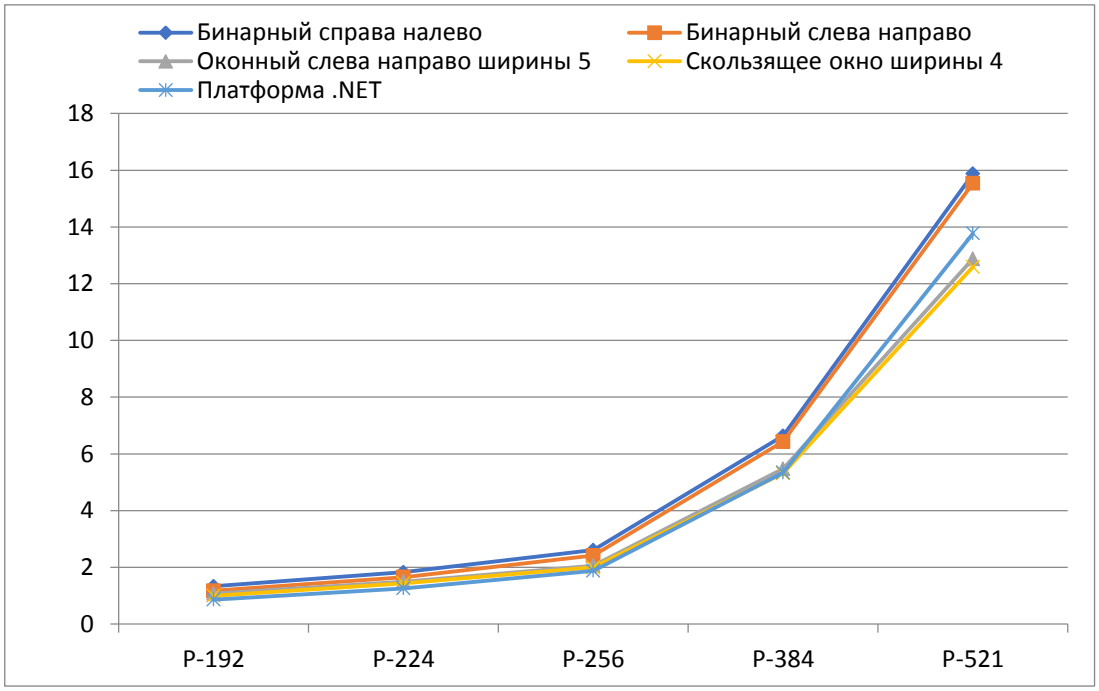

Рис. 1 – Производительность алгоритмов возведения числа в степень по модулю

Два алгоритма из рассмотренных являются оконными, потому для них необходимо определить оптимальную ширину окна. Все исследования проводились для чисел различных размерностей, которые ограничиваются размерами эллиптических групп, рекомендованных NIST кривых P-192, P-224, P-256, P-384, P-521 (число обозначает размер поля в битах). Из полученных результатов можно сделать вывод о том, что оптимальной шириной окна для алгоритма 3 является значение 5. Оптимальное значение ширины скользящего окна (алгоритм 4) приблизительно равняется 4.

После нахождения оптимальных значений для оконных алгоритмов перейдем к сравнительному анализу производительности возведения в степень по модулю всеми рассмотренными алгоритмами для чисел различной величины. График такой зависимости можно найти на рисунке 1.

Помимо описанных алгоритмов на график также включено возведение в степень по модулю стандартным методом платформы .NET. Полученные результаты позволяют сделать вывод о том, что бинарные методы обладают худшей производительностью, а оконные – лучшей. Самым лучшим методом возведения числа в степень по модулю является алгоритм с использованием скользящего окна ширины 4. Именно он и будет использован в разрабатываемой криптосистеме. Стоит заметить, что данный алгоритм на больших эллиптических группах превосходит по производительности соответствующий метод из платформы .NET, что является хорошим результатом.

Список использованных источников:

1. Menezes, A. Handbook of applied cryptography / A. Menezes, P. Van Oorschot, S. Vanstone – CRC Press, 1997.
2. St Denis, T. BigNum math: implementing cryptographic multiple precision arithmetic / T. St Denis, G. Rose, S. Vanstone – Syngress Publishing, Inc, 2006.

ВНЕДРЕНИЕ МОБИЛЬНЫХ ТЕХНОЛОГИЙ В ПРОЦЕСС ОБУЧЕНИЯ

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

Кротов Е.М.

Данилова Г.В. – м.т.н., ассистент

Отсутствие какой-либо практики использования мобильных телефонов в процессе обучения является упущением. Один мобильный гаджет может с легкость стать заменой сразу нескольким нужным предметам. Акцентирование внимания на внедрение мобильных технологий для оптимизации ведения учебы может повлиять в лучшую сторону на успеваемость учащихся.

В настоящее время абсолютное большинство людей пользуется мобильными телефонами. Гаджет стал неотъемлемой частью нашей жизни и используется ежедневно. Этот простой, функциональный и эффективный атрибут позволяет человеку оставаться на связи, где бы он ни был, не ограничиваясь лишь сотовой связью. Различные мессенджеры, социальные сети стали тем самым фактором, без которого сегодня трудно представить общение молодёжи.

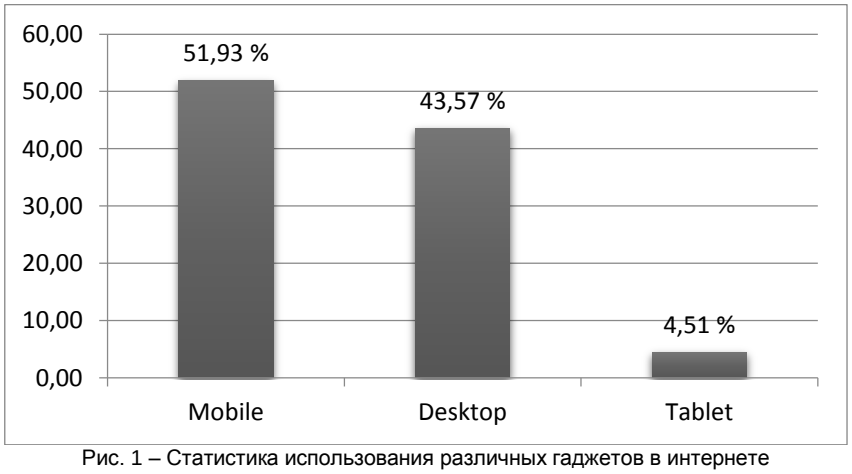

Рис. 1 – Статистика использования различных гаджетов в интернете

В то же время, преподаватели до сих пор вынуждены использовать бумагу для проведения тестов и контрольных работ. В случае, если у преподавателя много студентов, он вынужден таскать кучу макулатуры, что не только повышает риск утери чьих-то работ, но и нагружает излишним физическим трудом. Осуществлять проверку больших объемов тестов достаточно сложно и долго, при этом преподаватель сильно устает и может сам допустить ошибку, что в последствии опечалит студента и вызовет споры.

Беря во внимание данные факторы, встаёт вопрос об интеграции учебного процесса с мобильными гаджетами. Вместо огромных кип листочков с проверочными тестами, студенты могут без особых усилий