Нулевое разглашение для языков класса NP

Лекция N 6 курса “Современные задачи криптографии”

Юрий Лифшиц
yura@logic.pdmi.ras.ru

СПбГУ - SPRINT Lab

Осень’2005

План лекции

1. **Еше раз об определении нулевого разглашения**
   - Вспоминая прошлую лекцию
   - Доказательство NISO
2. **Нулевое разглашение для языков класса NP**
   - Формулировка теоремы
   - Доказательство теоремы
3. **Задача**

План лекции

1. **Еше раз об определении нулевого разглашения**
    - Вспоминая прошлую лекцию
    - Доказательство NISO
2. Нулевое разглашение для языков класса NP
    - Формулировка теоремы
    - Доказательство теоремы
3. Задача

Интерактивные доказательства

Инфраструктура

Два участника: **P** и **V**, строка x, язык L

P хочет убедить **V**, что $x \in L$

Они по очереди посылают сообщения друг другу

Через конечное число раундов **V** принимает или отвергает доказательство

Требования

Полнота $\forall x \in L, \exists \mathbf{P} : [\mathbf{P}(x), \mathbf{V}(x)] = 1$

Корректность $\forall x \notin L, \forall \mathbf{P}' : Pr([\mathbf{P}'(x), \mathbf{V}(x)] = 1) = \nu(|x|)$

Обычно считают, что **V** пользуется полиномиальным вероятностным алгоритмом, а **P** вычислительно не ограничен.

Нулевое разглашение

Нулевое разглашение:

$$\forall \mathbf{V'}\ \exists \mathsf{S}_{PPT}\ \forall x \in L : VIEW_{\mathbf{P},\mathbf{V'}}[x] \cong \mathsf{S'}[x]$$

Нулевое разглашение

Нулевое разглашение:

$$\forall \mathbf{V'}\ \exists \mathrm{S}_{PPT}\ \forall x \in L : VIEW_{\mathbf{P},\mathbf{V'}}[x] \cong \mathrm{S'}[x]$$

Следствие:

Все свойства x, которые **V** сможет вычислить за полиномиальное время после разговора с **P**, он мог вычислить и до разговора

История определения

Сначала хотели сделать определением:

Все свойства x, которые **V** сможет вычислить за полиномиальное время после разговора с **P**, он мог вычислить и до разговора

История определения

Сначала хотели сделать определением:

Все свойства x, которые **V** сможет вычислить за полиномиальное время после разговора с **P**, он мог вычислить и до разговора

Возникла трудность: как убедиться, что данное интерактивное доказательство обладает таким свойством?

История определения

Сначала хотели сделать определением:

Все свойства x, которые **V** сможет вычислить за полиномиальное время после разговора с **P**, он мог вычислить и до разговора

Возникла трудность: как убедиться, что данное интерактивное доказательство обладает таким свойством?

Естественный выход:
доказать, что **V** может сам "изобразить" диалог с воображаемым **P** (не зная ничего об x!)

NISO доказательство

P собирается доказать $G_0 \not\cong G_1$.

NISO доказательство

P собирается доказать $G_0 \ncong G_1$.

1. **V** выбирает случайное b и случайную перестановку π и посылает $\pi \circ G_b$

NISO доказательство

P собирается доказать $G_0 \not\cong G_1$.

1. V выбирает случайное b и случайную перестановку π и посылает $\pi \circ G_b$
2. P пытается угадать b

NISO доказательство

P собирается доказать $G_0 \ncong G_1$.

1. **V** выбирает случайное b и случайную перестановку π и посылает $\pi \circ G_b$
2. **P** пытается угадать b
3. Шаги 1-2 повторяются 1000 раз

Разглашение в NISO

Попытайтесь доказать нулевое разглашение для NISO.
Какие трудности?

Разглашение в NISO

Попытайтесь доказать нулевое разглашение для NISO.
Какие трудности?

Факт: NISO не обладает нулевым разглашением!

Что можно узнать у **P**?

Разглашение в NISO

Попытайтесь доказать нулевое разглашение для NISO. Какие трудности?

Факт: NISO не обладает нулевым разглашением!

Что можно узнать у **P**?

Ответ: например, для данного графа C, какому из двух G и H он не изоморфен.

План лекции

1. Еше раз об определении нулевого разглашения
   - Вспоминая прошлую лекцию
   - Доказательство NISO
2. **Нулевое разглашение для языков класса NP**
   - Формулировка теоремы
   - Доказательство теоремы
3. Задача

NP-полнота

NP — класс языков. Язык L принадлежит **NP**, если существует полиномиальный алгоритм P, такой что $x \in L \Leftrightarrow \exists y : P(x, y) = 1$.

NP-полнота

NP — класс языков. Язык L принадлежит **NP**, если существует полиномиальный алгоритм P, такой что $x \in L \Leftrightarrow \exists y : P(x, y) = 1.$

Язык L из класса **NP** называется **NP**-полным, если для любого другого языка L' из **NP** существует полиномиально вычислимая функция f такая, что $x \in L' \Leftrightarrow f(x) \in L$

Использование сведений

Факт: язык, состоящий из графов, раскрашиваемых правильным образом в три цвета является **NP**-полным

Использование сведений

Факт: язык, состоящий из графов, раскрашиваемых правильным образом в три цвета является **NP**-полным

Идея: если мы построим нулевое разглашение для 3-раскрашиваемости, мы сможем доказывать принадлежность любому языку из **NP**

Использование сведений

Факт: язык, состоящий из графов, раскрашиваемых правильным образом в три цвета является **NP**-полным

Идея: если мы построим нулевое разглашение для 3-раскрашиваемости, мы сможем доказывать принадлежность любому языку из **NP**

Как?

Доказательство для 3-раскраски

1. P случайным образом переставляет три цвета между собой

Доказательство для 3-раскраски

1. P случайным образом переставляет три цвета между собой
2. P коммитит (т.е. использует привязку к биту) цвета всех вершин

Доказательство для 3-раскраски

1. **P** случайным образом переставляет три цвета между собой
2. **P** коммитит (т.е. использует привязку к биту) цвета всех вершин
3. **V** выбирает случайную пару вершин

Доказательство для 3-раскраски

1. **P** случайным образом переставляет три цвета между собой
2. **P** коммитит (т.е. использует привязку к биту) цвета всех вершин
3. **V** выбирает случайную пару вершин
4. **P** открывает цвета этих вершин

Доказательство для 3-раскраски

1. **P** случайным образом переставляет три цвета между собой
2. **P** коммитит (т.е. использует привязку к биту) цвета всех вершин
3. **V** выбирает случайную пару вершин
4. **P** открывает цвета этих вершин
5. Шаги 1-4 повторяются $1000n^2$ раз

Доказательство для 3-раскраски

1. **P** случайным образом переставляет три цвета между собой
2. **P** коммитит (т.е. использует привязку к биту) цвета всех вершин
3. **V** выбирает случайную пару вершин
4. **P** открывает цвета этих вершин
5. Шаги 1-4 повторяются $1000n^2$ раз

Доказательство для 3-раскраски

1. **P** случайным образом переставляет три цвета между собой
2. **P** коммитит (т.е. использует привязку к биту) цвета всех вершин
3. **V** выбирает случайную пару вершин
4. **P** открывает цвета этих вершин
5. Шаги 1-4 повторяются $1000n^2$ раз

Какую схему привязки к биту надо использовать: с безусловной секретностью или с безусловной связанностью?

Вычислительно-нулевое разглашение

Семейство распределений:

Последовательность $\{A_k\}_{k \in \mathbb{N}}$

Каждое A_i — распределение на конечном множестве

Вычислительно-нулевое разглашение

Семейство распределений:

Последовательность $\{A_k\}_{k\in\mathbb{N}}$

Каждое A_i — распределение на конечном множестве

Вычислительная неразличимость

$\forall F_{poly} : \quad |Pr[x \to A_k;\ F(x) = 1] - Pr[x \to B_k;\ F(x) = 1]| = \nu(k)$

Вычислительно-нулевое разглашение

Семейство распределений:

Последовательность $\{A_k\}_{k\in\mathbb{N}}$

Каждое A_i — распределение на конечном множестве

Вычислительная неразличимость

$\forall F_{poly}: \quad |Pr[x \to A_k;\ F(x) = 1] - Pr[x \to B_k;\ F(x) = 1]| = \nu(k)$

Интерпретация: никакой полиномиальный алгоритм не может с *хорошей* вероятностью отличить одно семейство распределений от другого.

Конструкция симулятора

Алгоритм симулятора:

1. Генерeруем ключи для привязки к биту

Конструкция симулятора

Алгоритм симулятора:

1. Генерeруем ключи для привязки к биту
2. Генерируем случайные биты для **V'**

Конструкция симулятора

Алгоритм симулятора:

1. Генерeруем ключи для привязки к биту
2. Генерируем случайные биты для **V'**
3. Выбираем разные цвета для случайной пары вершин, остальные красим в красный цвет

Конструкция симулятора

Алгоритм симулятора:

1. Генерерируем ключи для привязки к биту
2. Генерируем случайные биты для **V'**
3. Выбираем разные цвета для случайной пары вершин, остальные красим в красный цвет
4. Посылаем зашифрованные цвета **V'**

Конструкция симулятора

Алгоритм симулятора:

1. Генереруем ключи для привязки к биту
2. Генерируем случайные биты для **V'**
3. Выбираем разные цвета для случайной пары вершин, остальные красим в красный цвет
4. Посылаем зашифрованные цвета **V'**
5. Если **V'** просит показать нашу пару вершин — показываем, если другую — сбрасываем память **V'** и пробуем еще раз

Конструкция симулятора

Алгоритм симулятора:

1. Генерeруем ключи для привязки к биту
2. Генерируем случайные биты для **V'**
3. Выбираем разные цвета для случайной пары вершин, остальные красим в красный цвет
4. Посылаем зашифрованные цвета **V'**
5. Если **V'** просит показать нашу пару вершин — показываем, если другую — сбрасываем память **V'** и пробуем еще раз
6. Цикл по шагам 1-3 повторяем до 1000 успешных итераций

Конструкция симулятора

Алгоритм симулятора:

1. Генерeруем ключи для привязки к биту
2. Генерируем случайные биты для **V'**
3. Выбираем разные цвета для случайной пары вершин, остальные красим в красный цвет
4. Посылаем зашифрованные цвета **V'**
5. Если **V'** просит показать нашу пару вершин — показываем, если другую — сбрасываем память **V'** и пробуем еще раз
6. Цикл по шагам 1-3 повторяем до 1000 успешных итераций

Конструкция симулятора

Алгоритм симулятора:

1. Генереруем ключи для привязки к биту
2. Генерируем случайные биты для **V'**
3. Выбираем разные цвета для случайной пары вершин, остальные красим в красный цвет
4. Посылаем зашифрованные цвета **V'**
5. Если **V'** просит показать нашу пару вершин — показываем, если другую — сбрасываем память **V'** и пробуем еще раз
6. Цикл по шагам 1-3 повторяем до 1000 успешных итераций

Математическое ожидание времени работы симулятора полиномиально!

Black-box сведение

Наша задача: доказать что симулятор полиномиально неотличим от P

Black-box сведение

Наша задача: доказать что симулятор полиномиально неотличим от **P**

Идея доказательства: если бы можно было отличить **P** от симулятора, то можно было бы и вскрыть привязку к биту.

Black-box сведение

Наша задача: доказать что симулятор полиномиально неотличим от **P**

Идея доказательства: если бы можно было отличить **P** от симулятора, то можно было бы и вскрыть привязку к биту.

Такая идеология называется **black-box reduction**

Гибридное доказательство

Докажем, что реакция **V'** на симулятор будет статистически неотличима от его реакции на зашифрованную правильную раскраску.

Гибридное доказательство

Докажем, что реакция **V’** на симулятор будет статистически неотличима от его реакции на зашифрованную правильную раскраску.

От противного: пусть **V’** реагирует существенно по разному.

Гибридное доказательство

Докажем, что реакция **V'** на симулятор будет статистически неотличима от его реакции на зашифрованную правильную раскраску.

От противного: пусть **V'** реагирует существенно по разному.

Рассмотрим серию промежуточных алгоритмов между симулятором и и **P**. Алгоритм номер i правильно красит $i + 2$ вершины, остальные красит в красный цвет.

Гибридное доказательство

Докажем, что реакция **V'** на симулятор будет статистически неотличима от его реакции на зашифрованную правильную раскраску.

От противного: пусть **V'** реагирует существенно по разному.

Рассмотрим серию промежуточных алгоритмов между симулятором и и **P**. Алгоритм номер i правильно красит $i + 2$ вершины, остальные красит в красный цвет.

Для какого-то i есть существенная разница в реакции **V'** на алгоритмы i и $i + 1$

Гибридное доказательство

Докажем, что реакция **V’** на симулятор будет статистически неотличима от его реакции на зашифрованную правильную раскраску.

От противного: пусть **V’** реагирует существенно по разному.

Рассмотрим серию промежуточных алгоритмов между симулятором и и **P**. Алгоритм номер i правильно красит $i+2$ вершины, остальные красит в красный цвет.

Для какого-то i есть существенная разница в реакции **V’** на алгоритмы i и $i+1$

Значит **V’** способен вскрыть привязку к биту!

План лекции

1. Еще раз об определении нулевого разглашения
   - Вспоминая прошлую лекцию
   - Доказательство NISO
2. Нулевое разглашение для языков класса NP
   - Формулировка теоремы
   - Доказательство теоремы
3. Задача

Задача

Рассмотрим две проблемы.

Первая: даны три графа G, H, C, такие что $G \not\cong H$. Требуется определить, какому из графов G или H не изоморфен C.

Вторая: по графам G и H определить, изоморфны они или нет.

Докажите, что если первая задача решается за полиномиальное время, то и вторая тоже.

Последний слайд

Если не запомните ничего другого:

- Принадлежность любому языку из класса **NP**имеет доказательство с нулевым разглашением

Последний слайд

Если не запомните ничего другого:

- Принадлежность любому языку из класса **NP** имеет доказательство с нулевым разглашением
- Нулевое разглашение выводится из стойкости привязки к биту

Последний слайд

Если не запомните ничего другого:

- Принадлежность любому языку из класса **NP**имеет доказательство с нулевым разглашением
- Нулевое разглашение выводится из стойкости привязки к биту
- Техника доказательства: гибридный метод

Последний слайд

Если не запомните ничего другого:

- Принадлежность любому языку из класса **NP**имеет доказательство с нулевым разглашением
- Нулевое разглашение выводится из стойкости привязки к биту
- Техника доказательства: гибридный метод

Последний слайд

Если не запомните ничего другого:

- Принадлежность любому языку из класса **NP**имеет доказательство с нулевым разглашением
- Нулевое разглашение выводится из стойкости привязки к биту
- Техника доказательства: гибридный метод

Вопросы?