Применение взаимодействующих нейронных сетей в криптографии

Ходаков Дмитрий.

March 14, 2014

1. Постановка задачи
   - Передача ключей
   - Tree Parity Machines

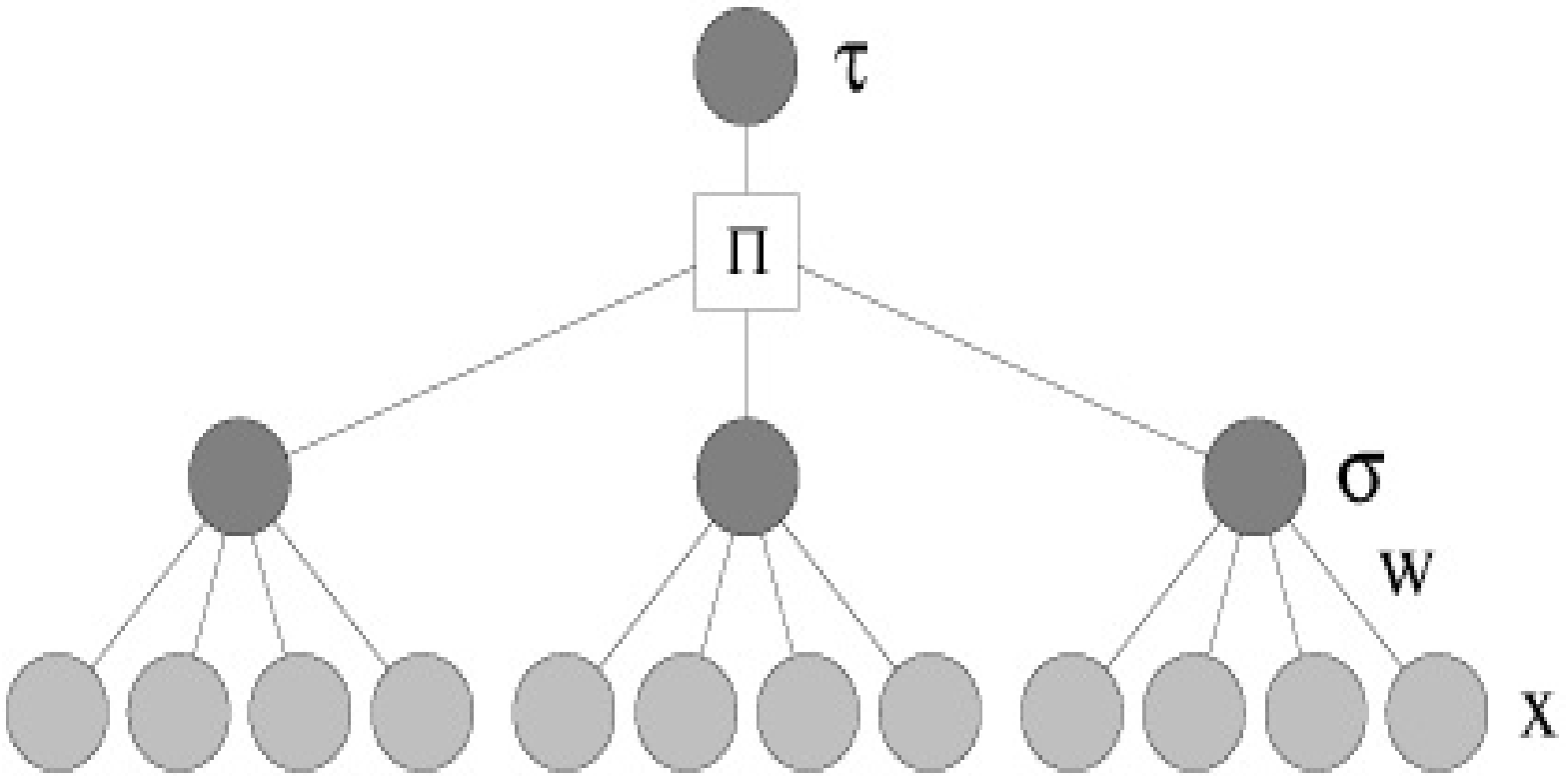
2. Результаты
   - Зависимость скорости сходимости от N
   - Оптимальное время синхронизации
   - Устойчивость к атакам
   - Эффективность по ср. с другими способами обмена ключами

Передача ключей

- Безопасная замена алгоритма Диффи-Хеллмана
- Основана на синхронизации двух нейросетей
- Они называются древовидных машин четности (TPM, tree parity machines)

Tree Parity Machines

- Многоуровневая нейронная сети прямого распространения
- Входные нейроны принимают значения $x_i \in \{-1, +1\}$
- Веса принимают значения $w_i \in \{-L, ..., 0, ..., +L\}$
- Значения скрытого нейрона $\sigma_i = sgn(\sum_{j=1}^{N} w_{ij} x_{ij})$
- Значения выходного нейрона $\tau = \prod_{i=1}^{K} \sigma_i$



Tree Parity Machines - алгоритм

У каждого абонента (А или Б) есть своя TPM. Синхронизация:

- Задаём случайные значения весовых коэффициентов
- Выполняем следующие шаги, пока не наступит синхронизация
- Генерируем случайный входной вектор X
- Вычисляем значения скрытых нейронов
- Вычисляем значение выходного нейрона
- Сравниваем выходы двух TPM:
- Выходы разные: переход к п.2.1
- Выходы одинаковые: применяем выбранное правило к весовым коэффициентам

Для обновления весовых коэффициентов могут использоваться следующие правила:

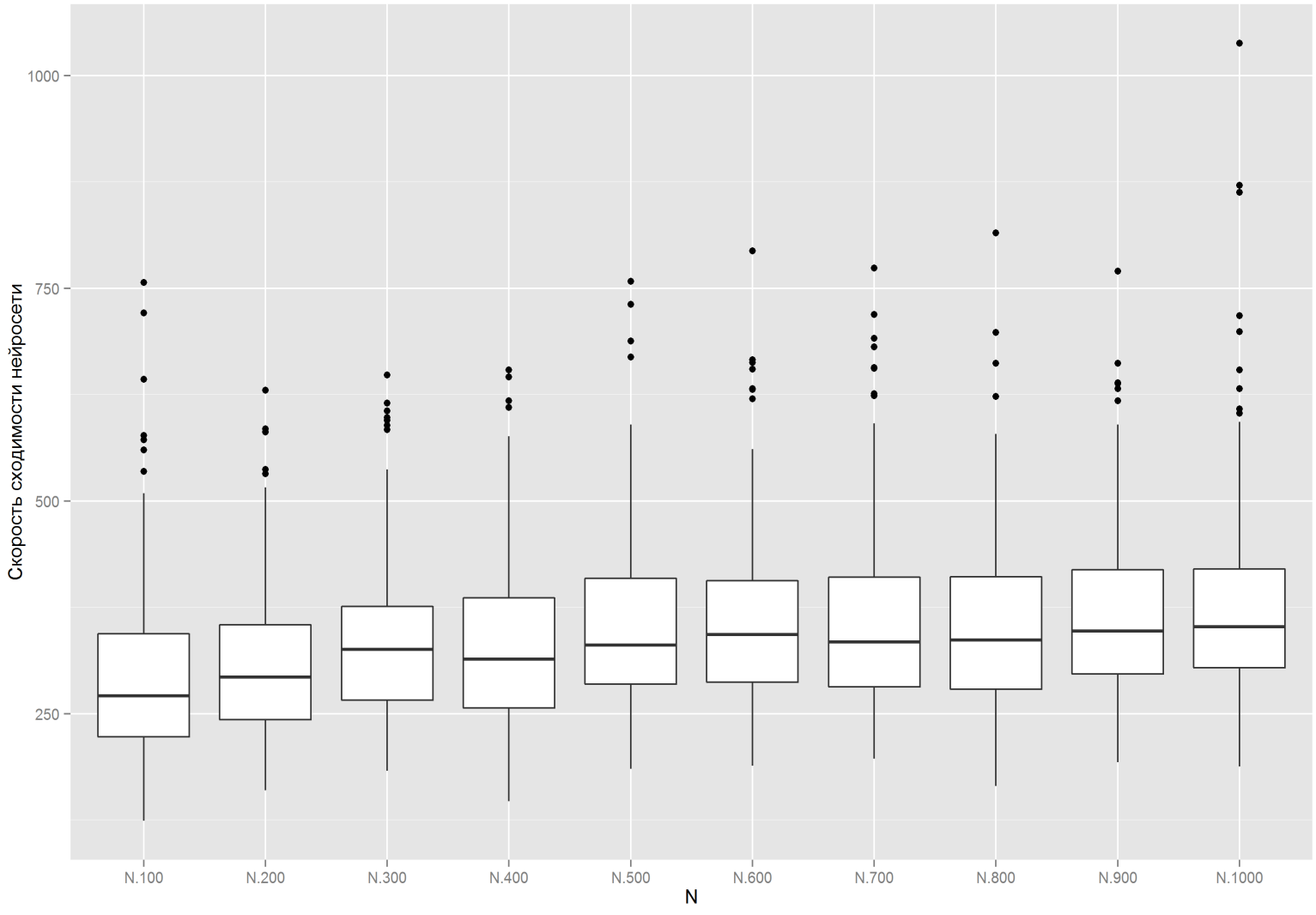
- Правило Хебба: $w_i^+ = w_i + \sigma_i x_i \Theta(\sigma_i \tau)\Theta(\tau^A \tau^B)$
- Обратное правило Хебба: $w_i^+ = w_i - \sigma_i x_i \Theta(\sigma_i \tau)\Theta(\tau^A \tau^B)$
- Случайное блуждание: $w_i^+ = w_i + x_i \Theta(\sigma_i \tau)\Theta(\tau^A \tau^B)$

Устойчивость к атакам

- Перебор: $(2L+1)^{KN}$ вариантов - неэффективно
- Синхронное обучение: Длительная синхронизация (10-100 и более раз) - неэффективно
- Другие атаки: геометрическая, вероятностный анализ, генетические алгоритмы

Зависимость скорости сходимости от N

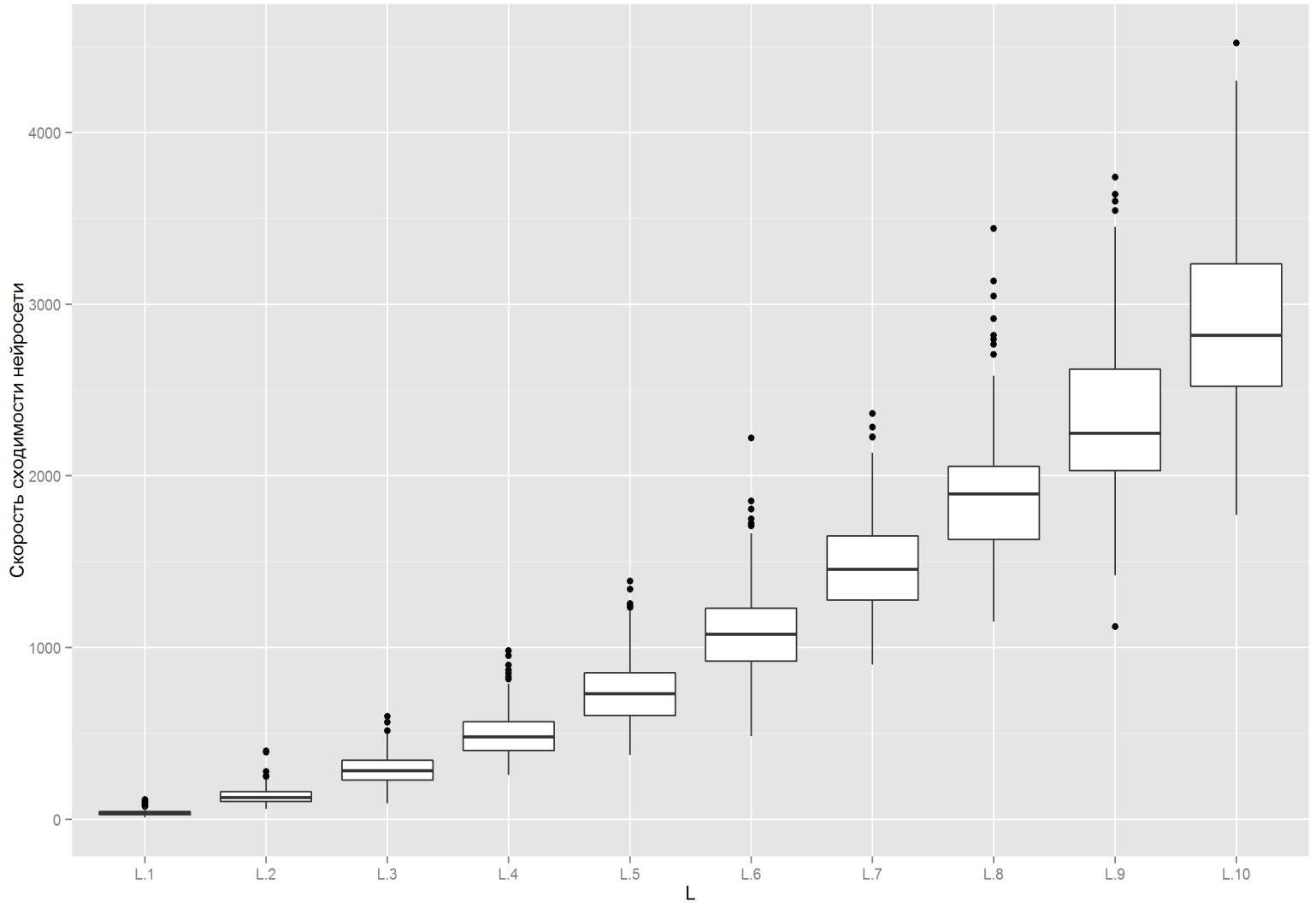
Правило Хебба, L=3, K=3



Зависимость скорость схождения сети от L

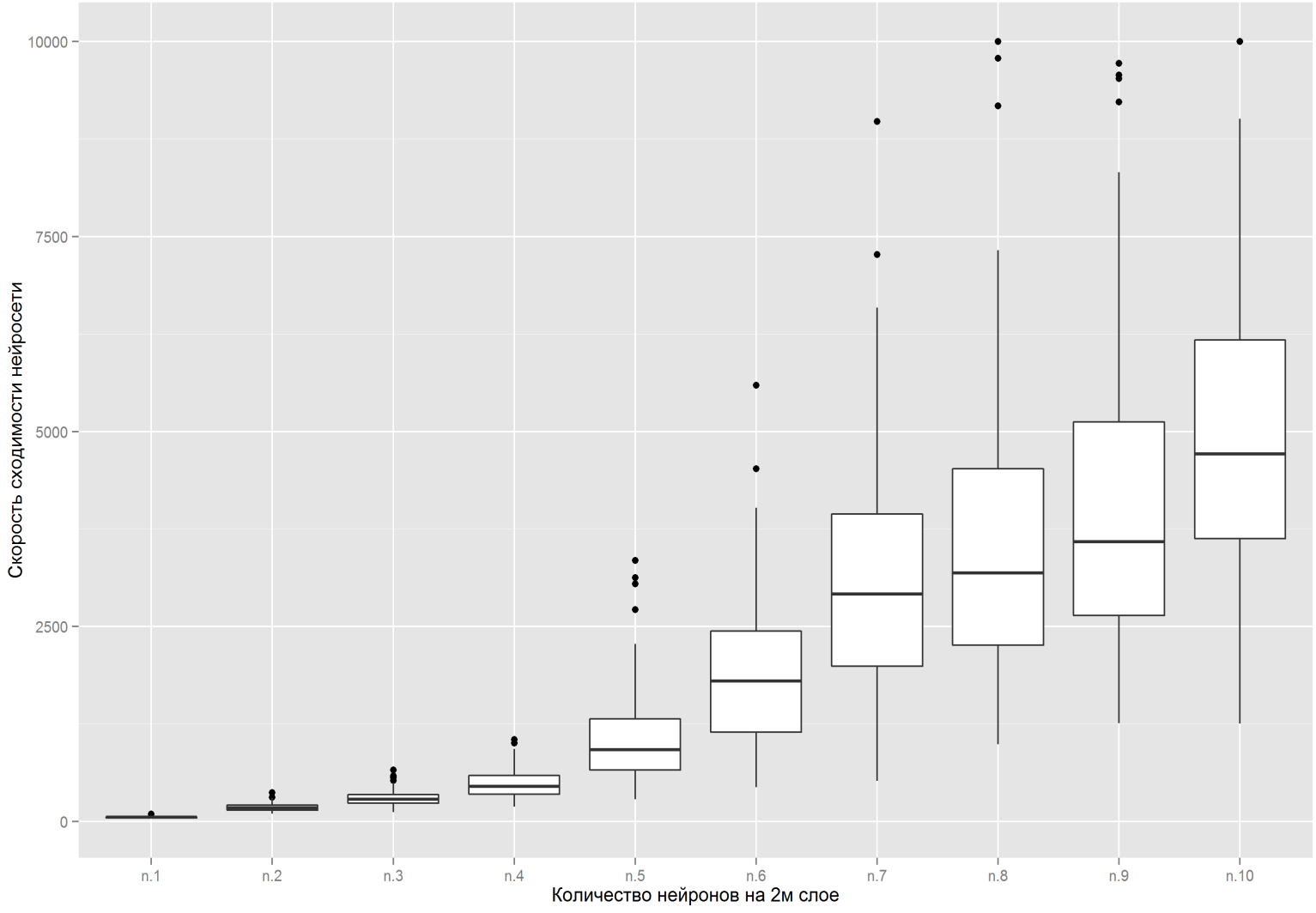
МФТИ

Правило Хебба, N=100, K=3



От кол-ва связей на 1м слое

Правило Хебба, N=100, L=3



Оптимальное время синхронизации

Каковы оптимальные параметры системы (N, L, кол-во нейронов на 1м слое) с точки зрения времени синхронизации и ключа?

Ср. с другими способами обмена ключами

МФТИ

- Что будет если внедрить в TLS
- Насколько велика экономия вычислительных операций и передач по сети по сравнению с Диффи-Хеллманом, задачей дискретного логарифмирования
- Возможные применение