МЕТОДЫ ВЫЧИСЛИТЕЛЬНОГО ИНТЕЛЛЕКТА И ИХ ПРИЛОЖЕНИЕ К ОБНАРУЖЕНИЮ СЕТЕВЫХ АТАК

Браницкий А.А.

Федеральное государственное бюджетное учреждение науки Санкт-Петербургский институт информатики и автоматизации Российской академии наук

IM&CTCPA

Санкт-Петербург, 25 октября 2018 г.

Содержание

Вычислительный интеллект

Вычислительный интеллект

Искусственный интеллект

- «Сильный» ИИ
  - ✓ Тест Тьюринга (А. Тьюринг)
  - ✓ Алгоритм, генерирующий новые алгоритмы (М.Л. Минский)
  - ✓ Универсальная машина Тьюринга с бесконечным набором алгоритмов (А. Тьюринг, А. Черч)
- «Слабый» ИИ
  - ✓ Символьные вычисления (Д. Маккарти)
  - ✓ Перцептрон (У. Мак-Каллок, У. Питтс)
  - ✓ Вычислительный интеллект (Д. Бездек, Р. Маркс, Д. Фогель)

Вычислительный интеллект: история

- 1983 г. — «International Journal of Computational Intelligence» (N. Cercone, G. McCalla)
- 1992 г. — Bezdek J. C. On the relationship between neural networks, pattern recognition and intelligence // International journal of approximate reasoning. – 1992. – Vol. 6, no. 2. – Pp. 85–107.
- 1993 г. — Marks R. J. Intelligence: Computational versus artificial // IEEE Transactions on Neural Networks. – 1993. – Vol. 4, no. 5. – Pp. 737–739.
- 1994 г. — сборник статей WCCI symposium Computational Intelligence: Imitating Life (Орландо, Флорида)

ВИ: определения

J. Bezdek (1992 г., 1994 г.): ВИ система:

- обрабатывает низкоуровневые данные об объекте
- обладает компонентами распознавания образов
- не базируется на экспертных знаниях
- обладает свойствами:
  - ✓ вычислительной адаптивности
  - ✓ устойчивости к наличию шумов
  - ✓ высокой скоростью функционирования
  - ✓ низким уровнем ошибок

ВИ: определения

R. Marks (1993 г.): ВИ:

- нейронные сети
- генетические алгоритмы
- нечеткие системы
- эволюционное программирование
- искусственная жизнь

ВИ: определения

R. Eberhart (1996 г., 2007 г.): «ВИ — методология, включающая вычисления, которые наделяют систему возможностью обучения и/или разрешения новых ситуаций таким образом, что система воспринимается как обладающая одним или несколькими атрибутами разума, а именно обобщением, обнаружением, ассоциативностью и абстракцией».

Основные свойства:

- адаптивность
- самоорганизация
- способность обобщения
- биологически инспирированная сущность

ВИ: определения

D. Fogel (2006 г.): «Любая система (углеродная, кремниевая, человек, общество или особь), которая воспроизводит адаптивное поведение для достижения целей в различных средах, можно сказать, является интеллектуальной».
«Методы ВИ — методы, которые могут быть использованы для адаптации решений к новым задачам без базирования на явных человеческих знаниях».

ВИ: определения

W. Duch (2003 г.): Область ВИ охватывает разработку интеллектуальных агентов и покрывает исследование тех задач, для которых отсутствует явный алгоритм их решения.

ИИ и ВИ: отличия

	ИИ	ВИ
Представление исходных данных	Символьное	Числовое
Вид анализа структуры объекта	Сверху-вниз	Снизу-вверх
Тип моделируемых процессов	Моделирование интеллектуального (человеческого) поведения посредством извлечения экспертных знаний	Моделирование естественных процессов и систем, связанных с интеллектуальным поведением
Метод поиска решения	Экспертные знания, логический вывод, ...	Эвристический поиск (градиентный спуск, эволюционные вычисления, ...)

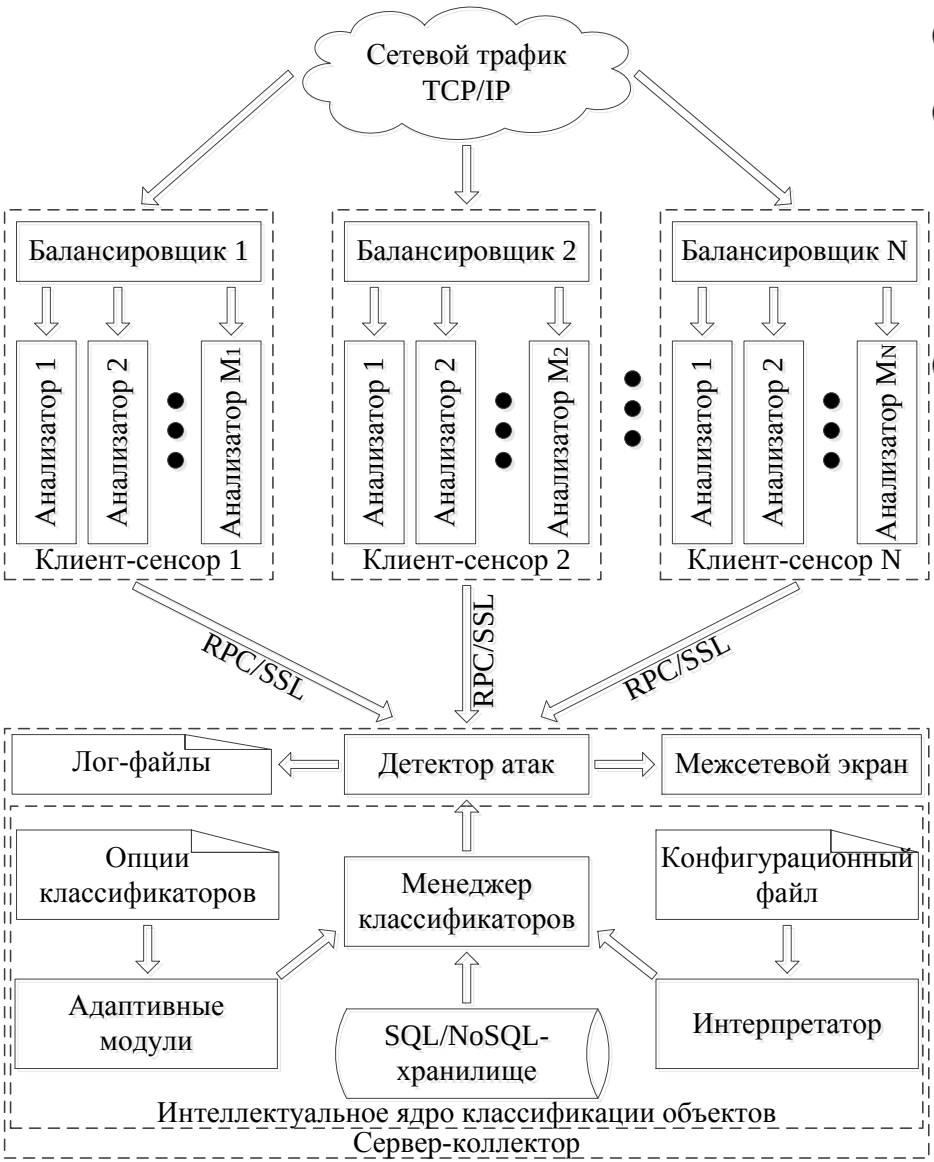
Требования, предъявляемые к ВИ системам

- низкий уровень ошибок
- высокая скорость функционирования
- вычислительная адаптивность
- способность обобщения
- устойчивость к наличию шумов

Гибридная СОА

Гибридная СОА

Гибридная СОА

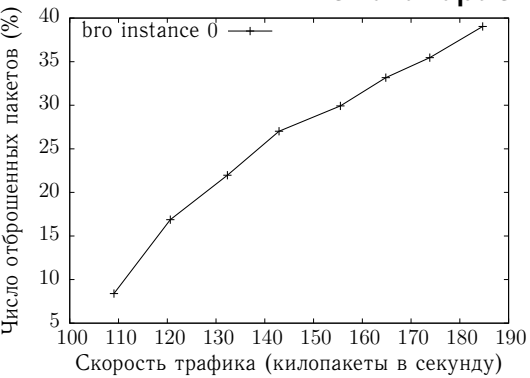


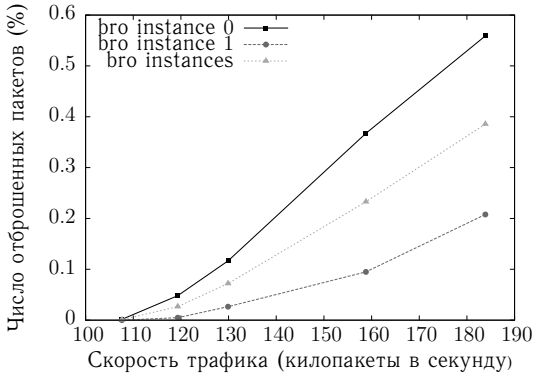
- Клиент-серверная архитектура
- Сенсоры
  - ✓ **Балансировщик трафика**
  - ✓ **Анализаторы трафика**
- Коллектор
  - ✓ **Интерпретатор**
  - ✓ **Адаптивные модули (so-библиотеки)**
  - ✓ **Менеджер классификаторов**
  - ✓ Детектор атак
  - ✓ Межсетевой экран (iptables)
  - ✓ SQL/NoSQL-хранилище (MySQL, MongoDB, CSV)
- Канал на основе RPC/SSL

Балансировщик трафика

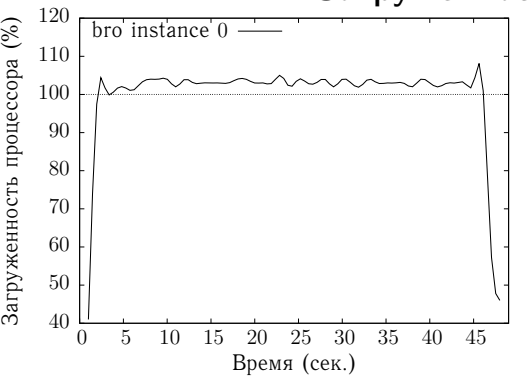
1 выходной интерфейс | 2 выходных интерфейса

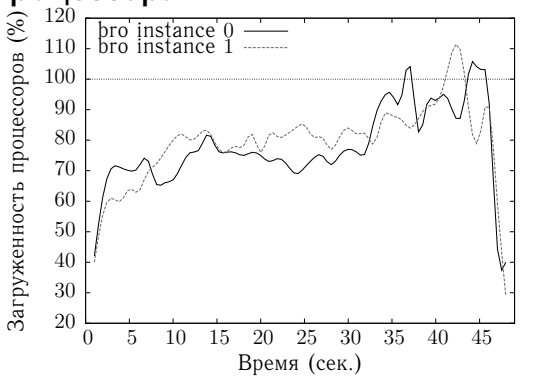
Число отбрасываемых пакетов



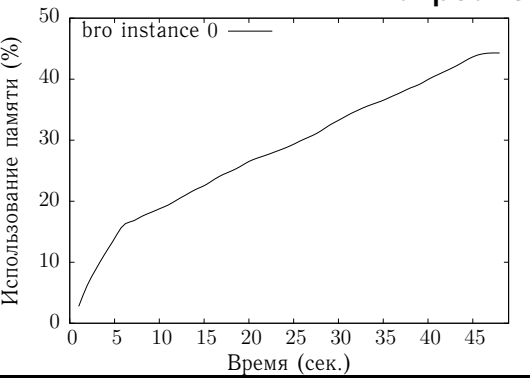


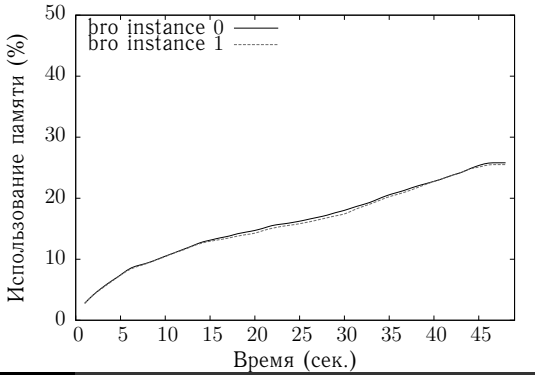
Загруженность процессора





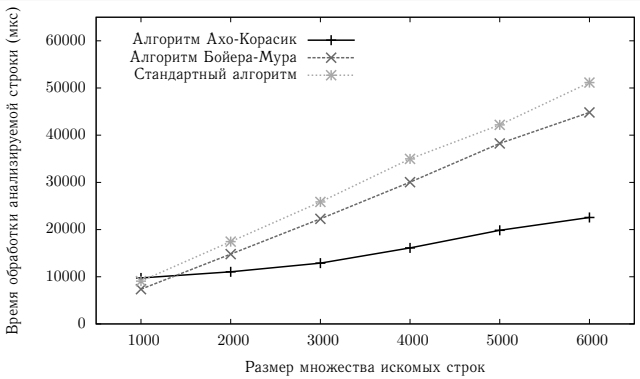
Потребление памяти

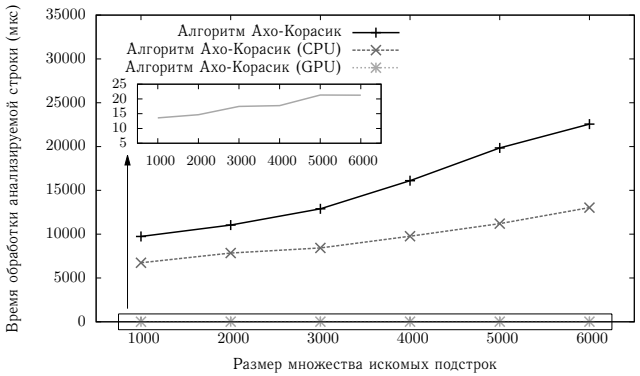


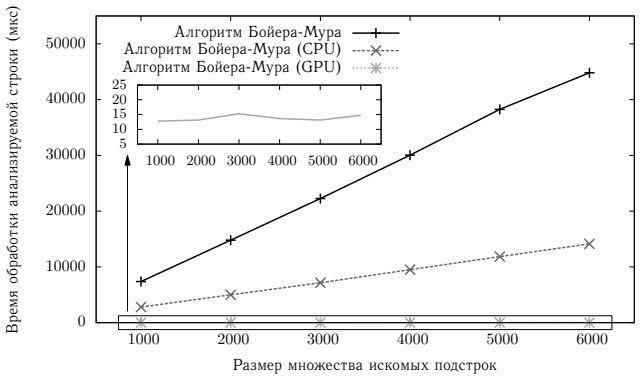


- Сетевая нагрузка распределяется между несколькими сетевыми анализаторами
- Пакеты, передаваемые внутри одной TCP-сессии, обрабатываются одним и тем же анализатором
- Номер выходного интерфейса вычисляется на основе неупорядоченной пары IP-адресов отправителя и получателя

Анализатор трафика







- Поддержка событийно-ориентированного механизма обработки сетевых соединений (32 события)
- Реализация алгоритмов IP-дефрагментации и TCP-реассемблирования, свойственных сетевому стеку ОС Linux (RFC 791, 3128, 793, 7413, 1858, 1071, ...)
- Параллельные модификации алгоритмов поиска шаблонных подстрок в сигнатурных правилах СОА (технологии OpenMP, CUDA)
- 106 сетевых параметров (продолжительность соединения, служба, интенсивность отправки пакетов, число активных соединений, признак изменения масштабирования TCP-окна, состояние соединения, ...)
- Алгоритм преаллокации памяти для хранения fnv-хешированных записей о соединениях
- Наличие API для обнаружения атак со скрытием и со вставкой (Ptacek & Newsham)

Интерпретатор, адаптивные модули, менеджер классификаторов

```
classifier_tree: {
  vars: {
    in_dim: {
      "33" }
    out_dim: {
      "6" } }
  classifier: {
    name: {
      "meta_classifier" }
    identifier: {
      "7" }
    so_library: {
      "libmeta_classifier.so" }
    prf_file: {
      "meta_classifier_results.prf" }
    sls_file: {
      "meta_classifier_struct.sls" }
    opt_file: {
      "meta_classifier_options.opt" }
    input_dim: {
      "7" }
    output_dim: {
      "$out_dim" }
    input_data: {
      "concat(sum(idx#4,idx#5),idx#3)" }
    output_data: {
      type: {
        "array" } }
    nested_classifiers: {
      classifier:{
        ... } } } }
```

- Интерпретатор
  - ✓ КС грамматика
  - ✓ Левосторонняя рекурсия
  - ✓ Построение дерева классификаторов
  - ✓ Поддержка операторов условного ветвления, векторной конкатенации, ...
- Адаптивные модули
  - ✓ 20 плагинов
  - ✓ 5 АЯВУ (C, C++, Perl, Python, R)
  - ✓ Маршалинг основных типов данных C в скриптовые объекты и обратно
  - ✓ Динамическое встраивание плагинов в ядро СОА без необходимости предлинковки
- Менеджер классификаторов
  - загрузка классификаторов из плагинов
  - вызов функций, определенных в плагинах
  - иерархический обход дерева классификаторов

Эксперименты

- Уровень ошибок
- Быстродействие
- Вычислительная адаптивность
- Способность обобщения
- Устойчивость к наличию шумов

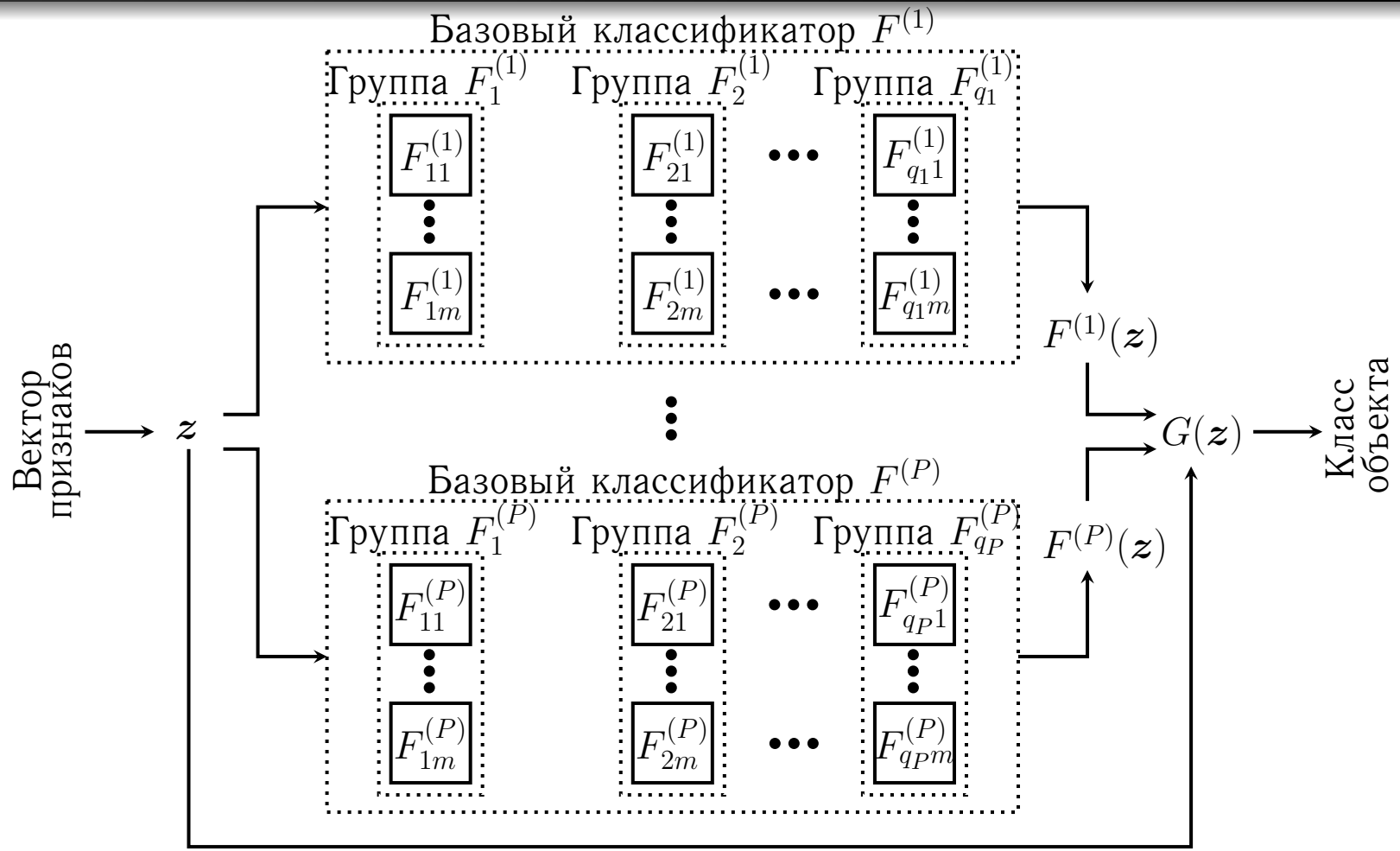
Уровень ошибок

Уровень ошибок

Показатели детектирования

- TPR, FPR — показатели корректности обнаружения и ложных срабатываний
- CCR, ICR — показатели корректности классификации и ошибочной классификации
- GPR, GCR — показатели обобщающей способности при обнаружении и классификации

Дерево классификаторов



- Каждая группа объединяет m или больше детекторов $F_{jk}^{(i)}$ $(k{=}1,\ldots,m)$
- Каждый классификатор $F^{(i)}$ содержит q_i групп $F_j^{(i)}$ $(j=1,\ldots,q_i)$
- Группа детекторов $F_j^{(i)}$ обучается на разных случайных подвыборках

Оценка показателей детектирования

Подход (авторы работы, год)	$TPR(\%)$	$FPR(\%)$	$TPR–FPR$
PCA, нейронечеткая сеть и нечеткая кластеризация (He H.T., Luo X.N., Liu B.L., 2005)	90.78	3.06	87.72
Многослойная иерархическая сеть Кохонена (СК) (Sarasamma S.T., Zhu Q.A., Huff J., 2005)	93.46	2.19	91.27
ANFIS, генетический алгоритм (ГА) и нечеткий вывод Мамдани (Toosi A. N., Kahani M., 2007)	95.3	1.6	93.7
Иммунная система и СК (Powers S.T., He J., 2008)	96.8	0.6	96.2
SVM, РБФ сети и бэггинг с простым голосованием (Govindarajan M., Chandrasekaran R. M., 2011)	99.03	1.12	97.91
Метод нормализованной энтропии и SVM (Agarwal B., Mittal N., 2012)	97.25	2.75	94.5
Комбинирование нейросетей с помощью бустинга (Akhgar M.S., Ghamgin H., Jafari M.T., 2013)	99.95	12.9	87.05
РБФ сети, нечеткая кластеризация и нейросеть (Amini M., Rezaeenoor J., Hadavandi E., 2014)	93	5.1	87.9
Нечеткий логический вывод Мамдани и ГА (Li J., Qu Y., Chao F., Shum H.P.H., Ho E.S.L., Yang L., 2018)	97.84	6.9	90.94
Иерархическая гибридизация бинарных классификаторов (один-ко-всем, МФХ)	**99.7**	**0.3**	**99.4**

Быстродействие

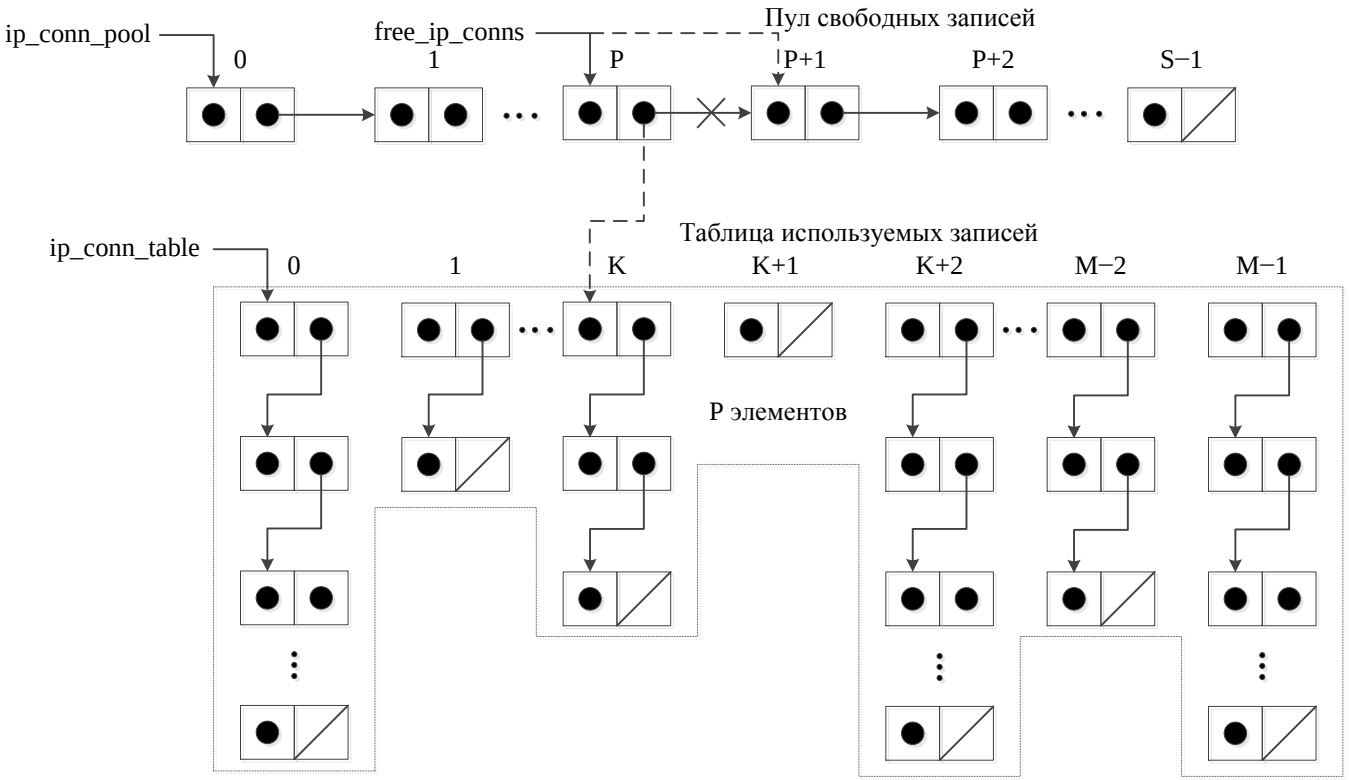
Быстродействие

Сравниваемые характеристики СОА

- Запуск каждой СОА — 100 раз
- Размер тестового pcap-файла — 10^7 пакетных записей, 709.047 MB
- Сравниваемые характеристики:
  - ✓ $T^{(real)}$ — общее время функционирования
  - ✓ $T^{(proc)}$ — время обработки пакетов
  - ✓ $V^{(pkts)}$ — скорость обработки килопакетов данных
  - ✓ $V^{(data)}$ — скорость обработки килобайтов данных
  - ✓ $L^{(CPU)}$ — загрузка CPU
  - ✓ $C^{(virt)}$ — размер потребляемой виртуальной памяти
  - ✓ $C^{(resd)}$ — размер потребляемой резидентной памяти
  - ✓ $C^{(real)}$ — размер потребляемой реальной памяти
- Сравниваемые СОА:
  - ✓ Snort 2.9.8.2
  - ✓ Suricata 3.0.1
  - ✓ Bro 2.4.1

Хранение записей о сетевых соединениях

Механизм выделения и хранения памяти



Настройки сенсора:

- Размер списка ip_conn_pool: $MAX\_IP\_POOL\_SIZE{=}S{=}300000$
- Размер таблицы ip_conn_table: $MAX\_IP\_TABLE\_SIZE{=}M{=}100000$
- Хранение 5-секундной истории захваченных пакетов
- Хеширование 13-байтового ключа сетевого соединения

Оперативность и ресурсопотребление

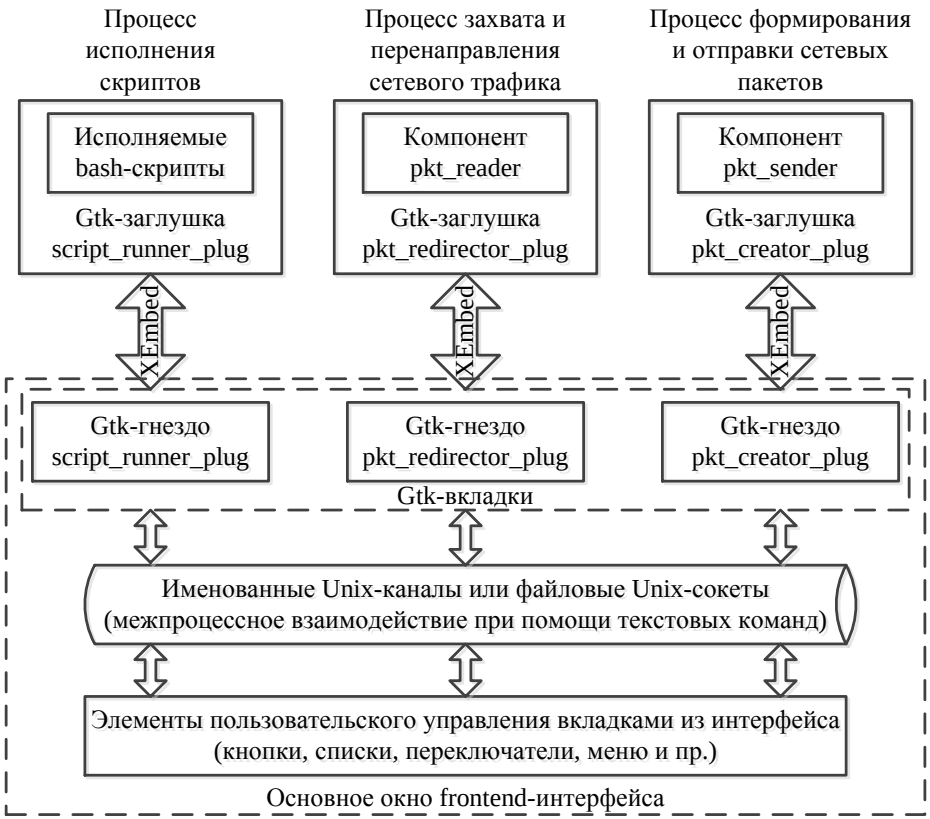
Характеристика		СОА				
		netpcap_sensor	Snort	Suricata (single)	Suricata (autofp)	Bro
$T^{(real)}$ (сек.)	min	30	64	48	87	178
	max	31	72	60	120	196
	avg	31	66	53	103	185
$T^{(proc)}$ (сек.)	min	29	49	42	80	154
	max	31	56	54	114	175
	avg	30	51	47	96	161
$V^{(pkts)}$ (Кпакеты/сек.)	min	320	174	182	86	56
	max	333	199	234	122	63
	avg	325	190	209	102	61
$V^{(data)}$ (Кбайты/сек.)	min	18 675	10 180	10 637	5013	3255
	max	19 445	11 602	13 632	7110	3696
	avg	18 949	11 099	12 223	5924	3544
$L^{(CPU)}$ (%)	min	52	0	0	0	7
	max	100	100	180	266	213
	avg	84	89	142	193	116
$C^{(virt)}$ (Мбайты)	min	176	78	0	0	0
	max	256	800	675	1026	1529
	avg	233	706	617	914	972
$C^{(resd)}$ (Мбайты)	min	144	15	0	0	0
	max	224	527	332	360	992
	avg	202	422	304	319	634
$C^{(real)}$ (Мбайты)	min	141	11	0	0	0
	max	221	522	325	353	983
	avg	198	417	297	312	625

Вычислительная адаптивность

Вычислительная адаптивность

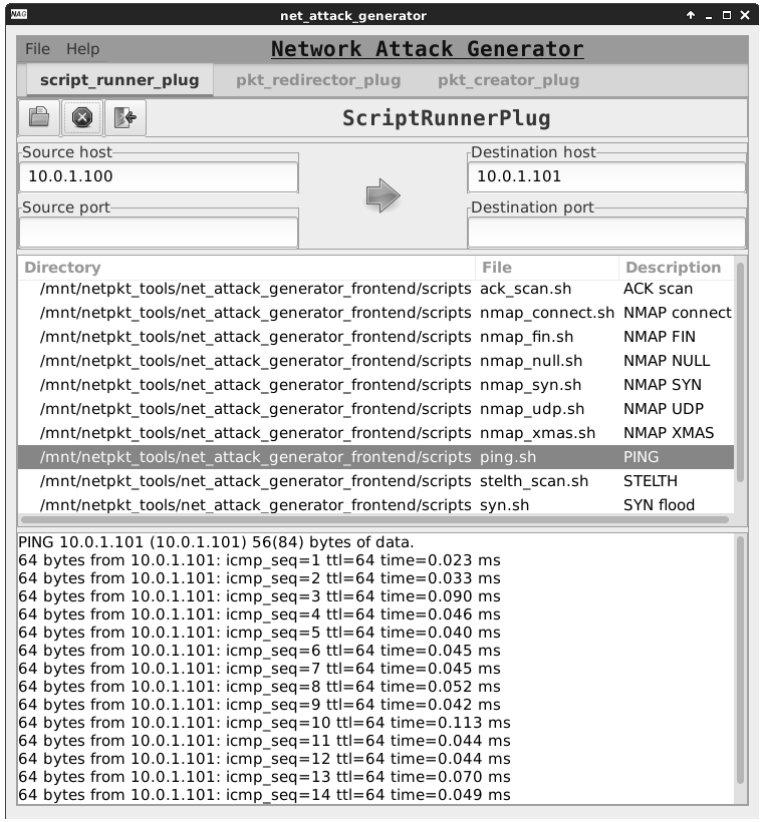
Генератор атак и набор данных

Архитектура генератора атак



- Взаимодействие процессов с основным окном на системном уровне через протокол XEmbed
- Управление процессами на пользовательском уровне через именованные Unix-каналы или файловые Unix-сокеты

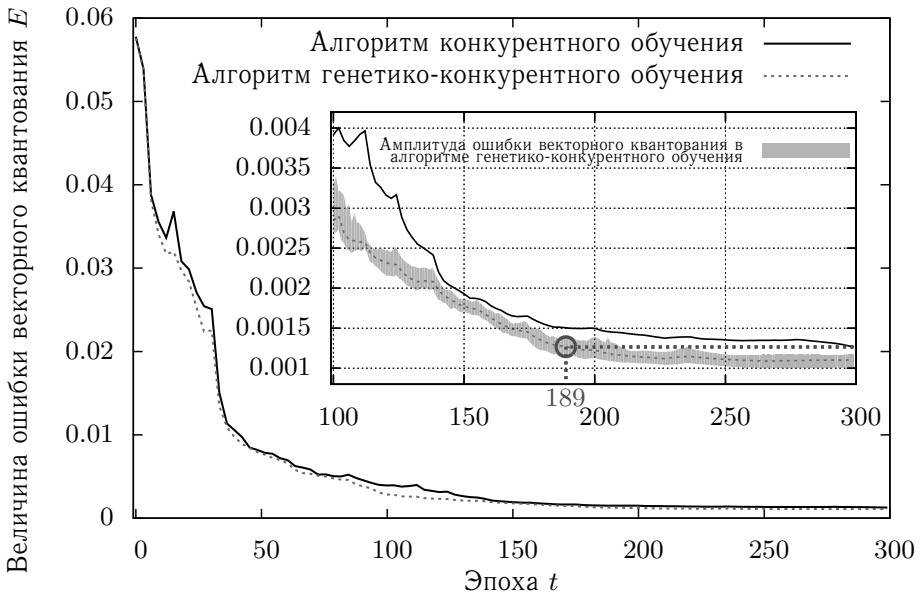
Интерфейс генератора атак

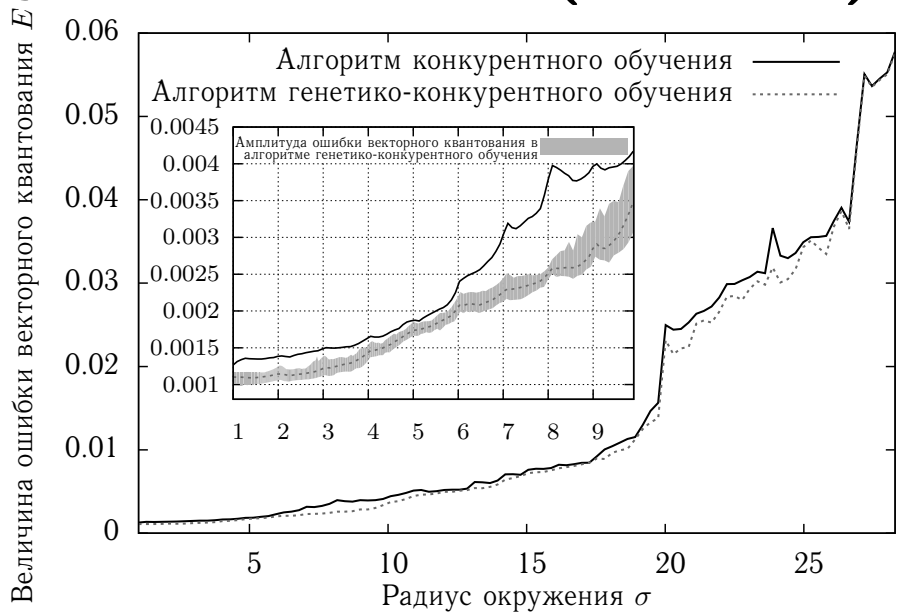


- Отказоустойчивое исполнение загруженных внутри вкладок процессов
- Возможность независимого запуска и прерывания скриптов (nmap, hping3, ...)
- Набор данных с нормальными сетевыми соединениями — MAWILab Data Set 2015/07/01

Вычислительная адаптивность (класс scan)

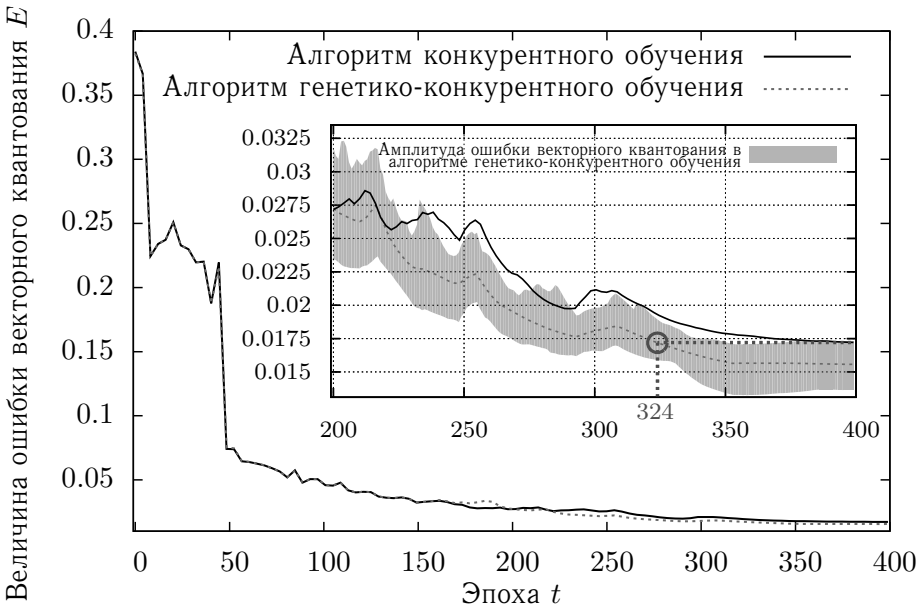
- Начальный параметр скорости обучения: $\kappa_0 = 0.3$
- Мощность обучающей выборки: $M = 1000$
- Размерность входного вектора: $n = 10$
- Размер выходной решетки: $I \times J = 20 \times 20$
- Число эпох обучения: $T = 300$
- Начальный радиус окружения: $\sigma_0 = 28.28$
- **Сокращение числа эпох обучения на 110 ($\delta_t \approx 37\%$)**

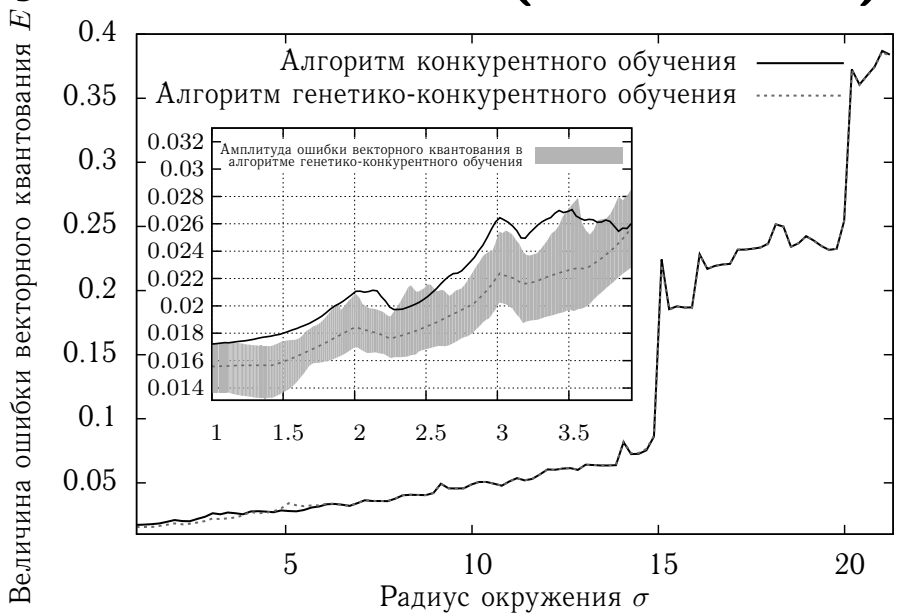




Вычислительная адаптивность (класс normal)

- Начальный параметр скорости обучения: $\kappa_0 = 0.7$
- Мощность обучающей выборки: $M = 900$
- Размерность входного вектора: $n = 20$
- Размер выходной решетки: $I \times J = 15 \times 15$
- Число эпох обучения: $T = 400$
- Начальный радиус окружения: $\sigma_0 = 21.21$
- **Сокращение числа эпох обучения на 74 (δ_t=18.5%)**





Способность обобщения

Способность обобщения

Пороги активации детекторов

$$h_{ij} = \begin{cases} h_{ij}^{-} - \beta \cdot \left(h_{ij}^{-} - h_{ij}^{+} \right), \text{ если } \forall \boldsymbol{x_k} \in C_0 \bigcap \mathcal{X}_{\mathcal{C}}^{(LS)} \; d_{ijk}^{-1} < h_{ij}^{-} \\ h_{ij}^{-}, \text{ если } \exists \boldsymbol{x_k} \in C_0 \bigcap \mathcal{X}_{\mathcal{C}}^{(LS)} \; d_{ijk}^{-1} \geqslant h_{ij}^{-}, \end{cases}$$

$$h_{ij}^{-} = \min_{\boldsymbol{x_k} \in C_l \bigcap \mathcal{X}_{\mathcal{C}}^{(LS)}} \left\{ d_{ijk}^{-1} \,|(i,j) = F_{IJ}(\boldsymbol{x_k}) \right\},$$

$$h_{ij}^{+} = \max_{\boldsymbol{x_k} \in C_0 \bigcap \mathcal{X}_{\mathcal{C}}^{(LS)}} \left\{ d_{ijk}^{-1} | (i,j) = F_{IJ}(\boldsymbol{x_k}) \right\},$$

$$d_{ijk} = D(\boldsymbol{x_k}, \boldsymbol{w_{ij}}) = \sqrt{\sum_{l=1}^{n} \left(x_{kl} - w_{ijl} \right)^2}, F_{IJ}(\boldsymbol{x_k}) = \arg \min_{\substack{1 \leqslant i \leqslant I \\ 1 \leqslant j \leqslant J}} d_{ijk},$$

C_0 — класс нормальных сетевых соединений,

C_l — l-ый класс аномальных сетевых соединений$(l = 1, 2)$.

- $\beta = \frac{1}{2}$
- $\beta = \arg \max_{\beta = 0, 0.01, \ldots, 1} GPR - FPR + GCR - ICR$

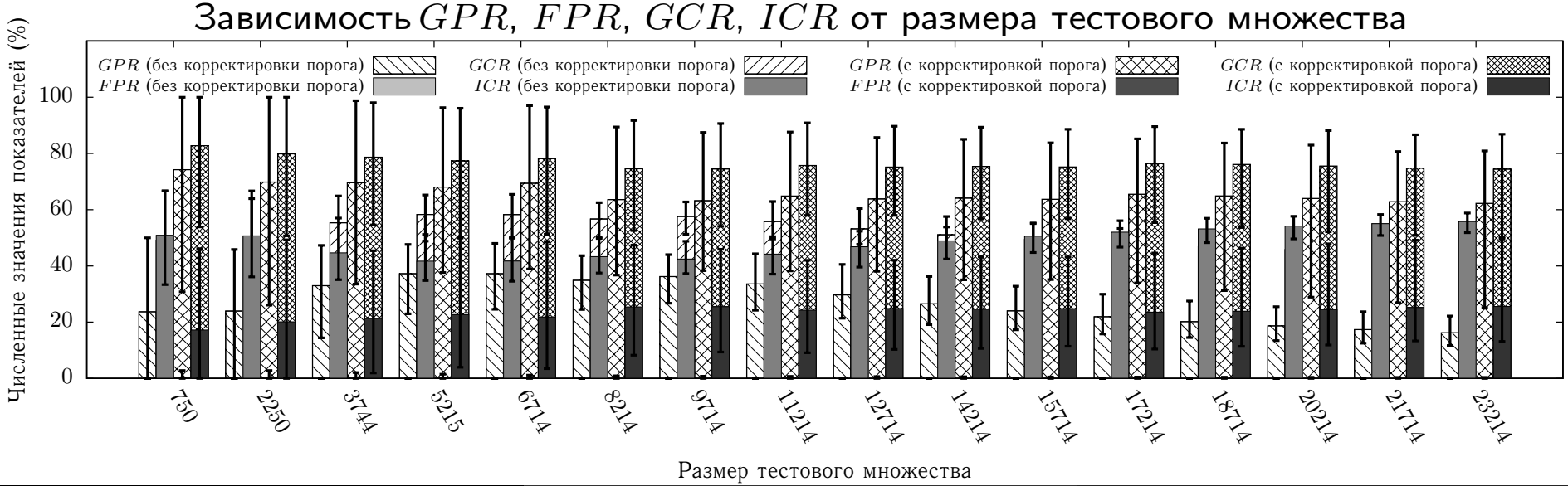
Способность обобщения ($\beta = 1/2$)

- $\kappa_0 = 0.1$
- $M = 4500$
- $n = 25$
- $T = 500$
- $I \times J$:10–15$\times$10–15
- 6 детекторов
- $\bar{M}^*$=750,...,23214

$\Delta_l^{(95\%)} \approx 2.83$

$\delta_g \approx 30.2\%$

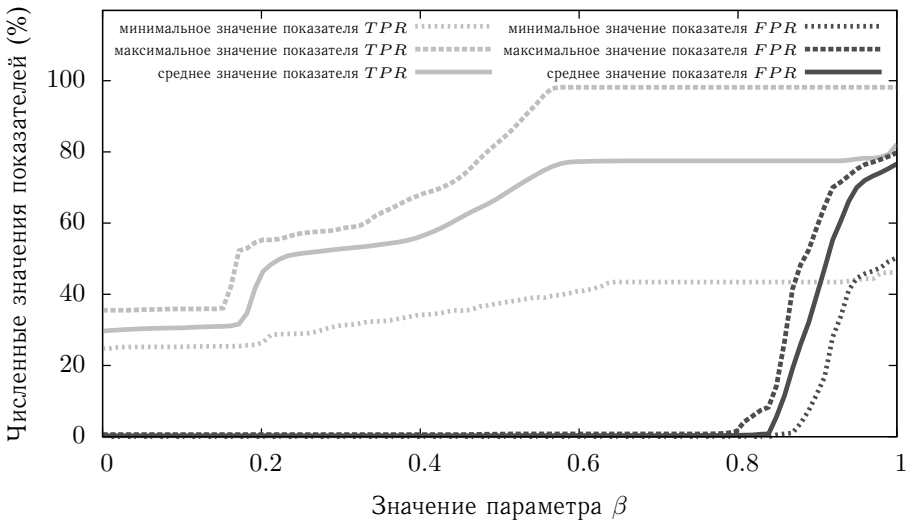
Порог	Средние значения показателей (%)			
	GPR	FPR	GCR	ICR
Размер тестового множества: M^*=750				
h_{ij}^-	23.68	0	49.12	50.88
h_{ij}	74.21	0.05	82.79	17.21
Размер тестового множества: M^*=23214				
h_{ij}^-	16.23	0	44.21	55.79
h_{ij}	62.22	0.01	74.41	25.59

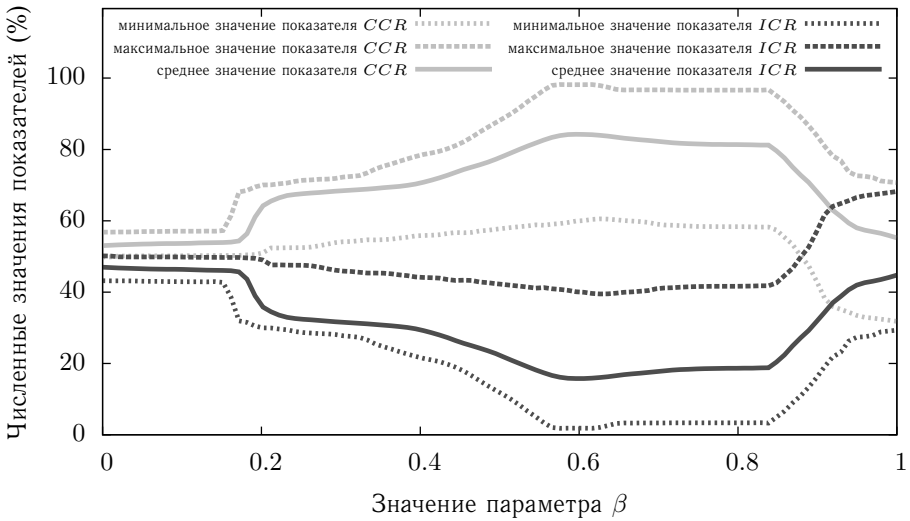


Способность обобщения ($\beta = 3/5$)

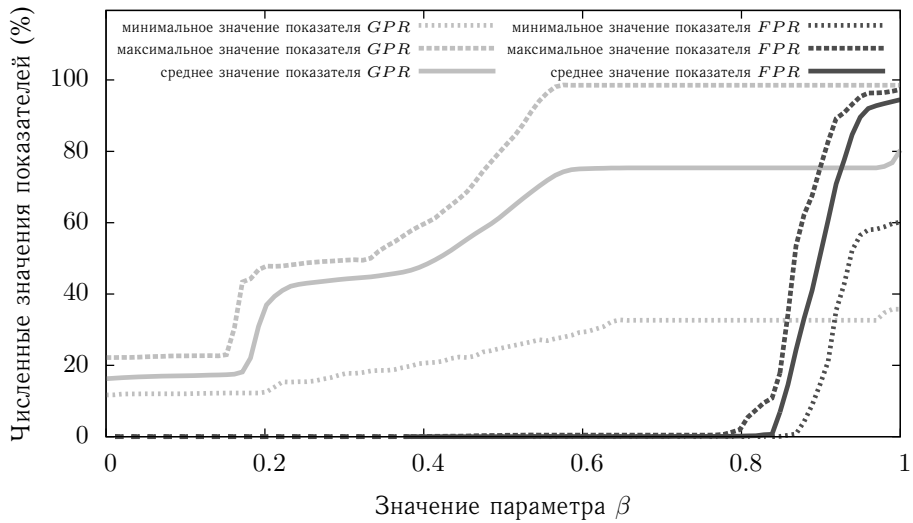
$$\beta_{opt} = \arg \max_{\beta=0,0.01,\ldots,1} GPR - FPR + GCR - ICR = 0.6$$

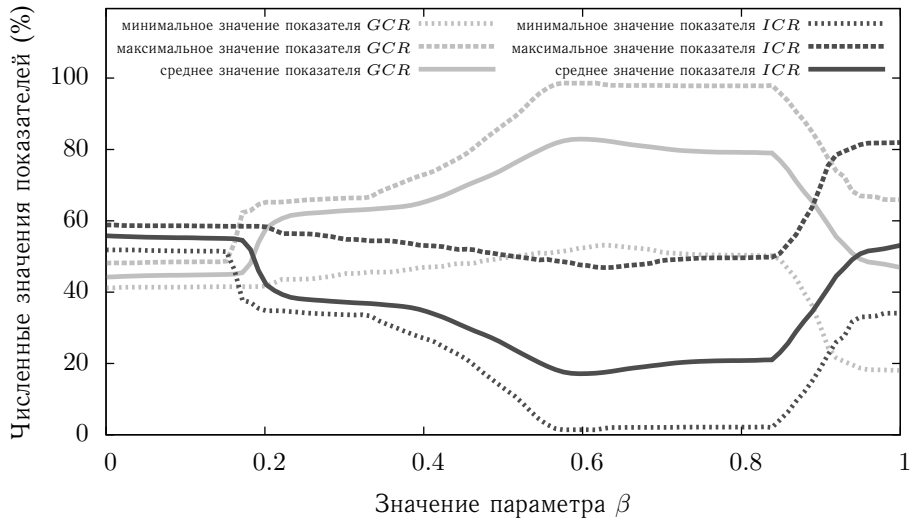
Зависимость TPR, FPR, CCR, ICR от β





Зависимость GPR, FPR, GCR, ICR от β





Устойчивость к наличию шумов

Устойчивость к наличию шумов

Набор данных

5-блочная кросс-валидация

- Разбиение множества $\bar{\mathcal{X}}_{\mathcal{C}}^{(TS)}$:

$$\bar{\mathcal{X}}_{\mathcal{C}}^{(TS)} = \bigvee_{k=1}^{5} \overset{(k)}{\bar{\mathcal{X}}}{}_{\mathcal{C}}^{(TS)}, \quad \#\overset{(1)}{\bar{\mathcal{X}}}{}_{\{C_l\}}^{(TS)} \approx \ldots \approx \overset{(5)}{\bar{\mathcal{X}}}{}_{\{C_l\}}^{(TS)}$$

$(l=0,\ldots,6)$

- Отношение размера обучающей выборки к контрольной: $3:2$
- Число процессов обучения и тестирования базовых классификаторов: $3 \times C_5^3 = 30$ раз
- Вычисление показателей на уникальных элементах, не встречавшихся при обучении:

$$\min_{p=1,2,3} \frac{1}{10} \cdot \sum_{1 \leqslant d < e \leqslant 5} GPR_{(de)_p},$$

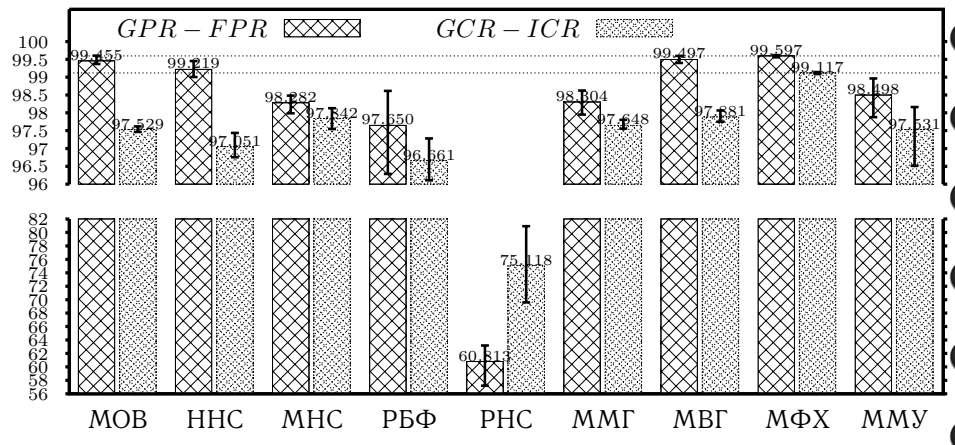
$$\max_{p=1,2,3} \frac{1}{10} \cdot \sum_{1 \leqslant d < e \leqslant 5} GPR_{(de)_p},$$

$$\frac{1}{3} \cdot \sum_{p=1}^{3} \frac{1}{10} \cdot \sum_{1 \leqslant d < e \leqslant 5} GPR_{(de)_p}$$

- Набор данных DARPA 1998
- 101113 записей
- 53733 уникальных записи
- 5 базовых классификаторов: машины опорных векторов (МОВ), нейронечеткие сети (ННС), многослойные нейронные сети (МНС), нейронные сети с радиальными базисными функциями (РБФ), рекуррентные нейронные сети Джордана (РНС)
- 2 ошибочных классификатора: случайный классификатор (СК) и ложный классификатор (ЛК)
- 6 классов аномальных сетевых соединений (probing и DoS)
- 4 типа низкоуровневых схем объединения детекторов
- 4 типа агрегирующих композиций

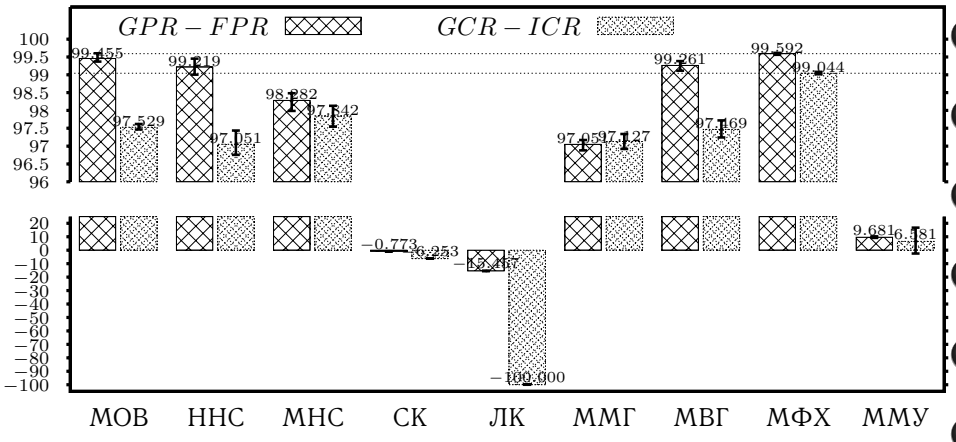
Устойчивость к наличию шумов (one-vs-all)

Схема комбинирования детекторов one-vs-all (90)



- $GPR \Uparrow$ на 0.02% (МФХ vs ННС)
- $FPR \Downarrow$ на 0.07% (ММГ vs МОВ)
- $GCR \Uparrow$ на 0.71% (МФХ vs МНС)
- $ICR \Downarrow$ на 0.57% (МФХ vs МНС)
- $GPR–FPR \Uparrow$ на 0.14% (МФХ vs МОВ)
- $GCR–ICR \Uparrow$ на 1.28% (МФХ vs МНС)

Схема комбинирования детекторов one-vs-all (с СК и ЛК)

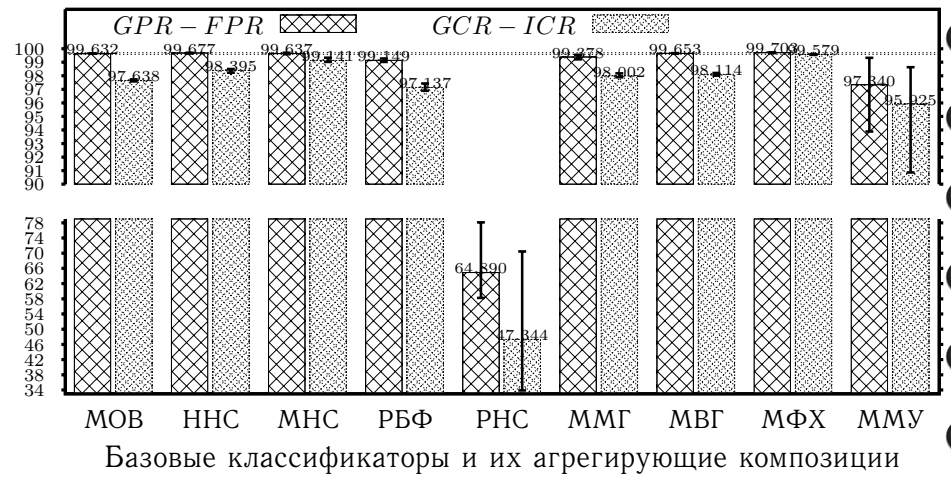


- $GPR \Uparrow$ на 0.02% (МФХ vs ННС)
- $FPR \Downarrow$ на 0.01% (МФХ vs МОВ)
- $GCR \Uparrow$ на 0.67% (МФХ vs МНС)
- $ICR \Downarrow$ на 0.53% (МФХ vs МНС)
- $GPR–FPR \Uparrow$ на 0.14% (МФХ vs МОВ)
- $GCR–ICR \Uparrow$ на 1.2% (МФХ vs МНС)

$$\delta_n = \left(1 - \frac{0.71 - 0.67}{0.71}\right) \cdot 100\% \approx 94\%$$

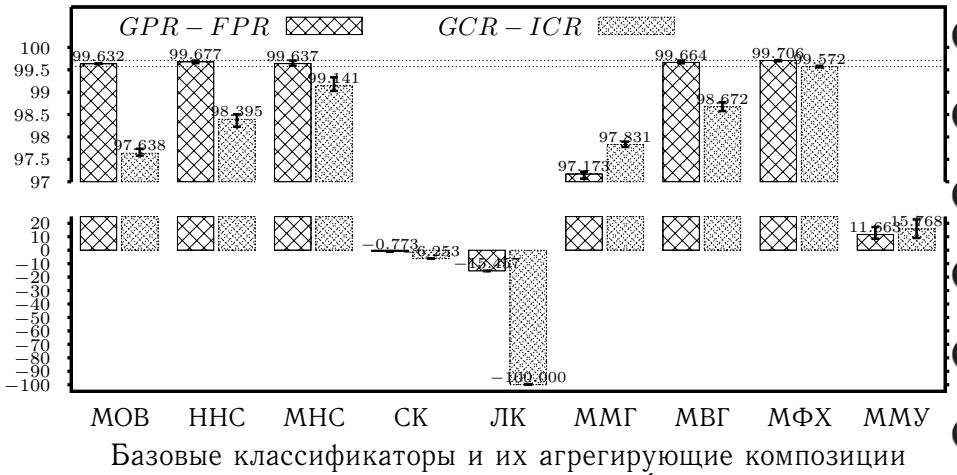
Устойчивость к наличию шумов (one-vs-one)

Схема комбинирования детекторов one-vs-one (315)



- $GPR \Uparrow$ на 0.003% (МФХ vs ННС)
- $FPR \Downarrow$ на 0.023% (МФХ vs ННС)
- $GCR \Uparrow$ на 0.217% (МФХ vs МНС)
- $ICR \Downarrow$ на 0.221% (МФХ vs МНС)
- $GPR{-}FPR \Uparrow$ на 0.025% (МФХ vs ННС)
- $GCR{-}ICR \Uparrow$ на 0.438% (МФХ vs МНС)

Схема комбинирования детекторов one-vs-one (с СК и ЛК)



- $GPR \Uparrow$ на 0.003% (МФХ vs ННС)
- $FPR \Downarrow$ на 0.056% (ММГ vs ННС)
- $GCR \Uparrow$ на 0.213% (МФХ vs МНС)
- $ICR \Downarrow$ на 0.218% (МФХ vs МНС)
- $GPR{-}FPR \Uparrow$ на 0.029% (МФХ vs ННС)
- $GCR{-}ICR \Uparrow$ на 0.431% (МФХ vs МНС)

$$\delta_n = \left(1 - \frac{0.217 - 0.213}{0.217}\right) \cdot 100\% \approx 98\%$$

Заключение

- Рассмотрено приложение методов вычислительного интеллекта к обнаружению сетевых атак
- Представлены архитектура гибридной СОА и эксперименты по оценке ее эффективности